



**HAL**  
open science

# Automatic Analysis of Macro and Micro Facial Expressions: Detection and Recognition via Machine Learning

Dawood Al Chanti

► **To cite this version:**

Dawood Al Chanti. Automatic Analysis of Macro and Micro Facial Expressions: Detection and Recognition via Machine Learning. Image Processing [eess.IV]. UGA (Université Grenoble Alpes), 2019. English. NNT: . tel-02525707v1

**HAL Id: tel-02525707**

**<https://hal.univ-grenoble-alpes.fr/tel-02525707v1>**

Submitted on 12 Nov 2019 (v1), last revised 31 Mar 2020 (v2)

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

## THÈSE

Pour obtenir le grade de

### **DOCTEUR DE LA COMMUNAUTÉ UNIVERSITÉ GRENOBLE ALPES**

Spécialité : SIGNAL IMAGE PAROLE TELECOMS

Arrêté ministériel : 25 mai 2016

Présentée par

### **DAWOOD AL CHANTI**

Thèse dirigée par **Alice CAPLIER** , Professeur, Communauté  
Université Grenoble Alpes

préparée au sein du **Laboratoire Grenoble Images Parole Signal  
Automatique**  
dans l'**École Doctorale Electronique, Electrotechnique,  
Automatique, Traitement du Signal (EEATS)**

### **Analyse Automatique des Macro et Micro Expressions Faciales: Détection et Reconnaissance par Machine Learning**

### **Automatic Analysis of Macro and Micro Facial Expressions: Detection and Recognition via Machine Learning**

Thèse soutenue publiquement le **5 novembre 2019**,  
devant le jury composé de :

**Madame ALICE CAPLIER**

PROFESSEUR, UNIVERSITE GRENOBLE ALPES, Directeur de thèse

**Monsieur RENAUD SEGUIER**

PROFESSEUR, CENTRALE SUPELEC - RENNES, Rapporteur

**Monsieur MOHAMED DAOUDI**

PROFESSEUR, UNIVERSITE DE LILLE, Rapporteur

**Monsieur OLIVIER ALATA**

PROFESSEUR, UNIVERSITE JEAN MONNET - SAINT ETIENNE,  
Examineur

**Madame MICHELE ROMBAUT**

PROFESSEUR, UNIVERSITE GRENOBLE ALPES, Examineur

Président





---

**Abstract** — Facial expression analysis is an important problem in many biometric tasks, such as face recognition, face animation, affective computing and human computer interface. In this thesis, we aim at analyzing facial expressions using images and video sequences. We divided the problem into three leading parts.

First, we study **Macro Facial Expressions for Emotion Recognition** and we propose three different levels of feature representations. Low-level feature through a Bag of Visual Word model, mid-level feature through Sparse Representation and hierarchical features through a Deep Learning based method. The objective of doing this is to find the most effective and efficient representation that contains distinctive information of expressions and that overcomes various challenges coming from: 1) intrinsic factors such as appearance and expressiveness variability and 2) extrinsic factors such as illumination, pose, scale and imaging parameters, *e.g.*, resolution, focus, imaging, noise. Then, we incorporate the temporal dimension to extract spatio-temporal features with the objective to describe subtle feature deformations to discriminate ambiguous classes.

Second, we direct our research toward transfer learning, where we aim at **Adapting Facial Expression Models to New Domains and Tasks**. Thus we study domain adaptation and zero shot learning for developing a method that solves the two tasks jointly. Our method is suitable for unlabelled target datasets coming from different data distributions than the source domain and for unlabelled target datasets with different label distributions but sharing the same context as the source domain. Therefore, to permit knowledge transfer between domains and tasks, we use Euclidean learning and Convolutional Neural Networks to design a mapping function that maps the visual information coming from facial expressions into a semantic space coming from a Natural Language model that encodes the visual attribute description or uses the label information. The consistency between the two subspaces is maximized by aligning them using the visual feature distribution.

Third, we study **Micro Facial Expression Detection**. We propose an algorithm to spot micro-expression segments including the onset and offset frames and to spatially pinpoint in each image the regions involved in the micro-facial muscle movements. The problem is formulated into Anomaly Detection due to the fact that micro-expressions occur infrequently and thus leading to few data generation compared to natural facial behaviours. In this manner, first, we propose a deep Recurrent Convolutional Auto-Encoder to capture spatial and motion feature changes of natural facial behaviours. Then, a statistical based model for estimating the probability density function of normal facial behaviours while associating a discriminating score to spot micro-expressions is learned based on a Gaussian Mixture Model. Finally, an adaptive thresholding technique for identifying micro expressions from natural facial behaviours is proposed.

Our algorithms are tested over deliberate and spontaneous facial expression benchmarks.

**Keywords:** Macro-Facial Expression Recognition, Micro-Facial Expression Detection, Domain Adaptation, Zero Shot Learning, Recurrent Convolutional Auto-Encoder, 3D-CNN, 2D-CNN, Sparse Representation, Bag of Visual Words.

---

**Résumé** — L'analyse automatique des expressions faciales représente à l'heure actuelle une problématique importante associée à de multiples applications telles que la reconnaissance de visages, l'animation de visages ou encore les interactions homme machine. Dans cette thèse, nous nous attaquons au problème de la reconnaissance d'expressions faciales à partir d'une image ou d'une séquence d'images. Nous abordons le problème sous trois angles.

Tout d'abord, nous étudions les macro-expressions faciales et nous proposons de comparer l'efficacité de trois descripteurs différents tenant compte d'informations de plus ou moins haut niveau. Cela conduit au développement d'un algorithme de reconnaissance d'expressions basé sur des descripteurs bas niveau encodés dans un modèle de type sac de mots, puis d'un algorithme basé sur des descripteurs de moyen niveau associés à une représentation éparsée et enfin d'un algorithme d'apprentissage profond tenant compte de descripteurs haut niveau. Notre objectif lors de la comparaison de ces trois algorithmes est de trouver la représentation des informations de visages la plus discriminante pour reconnaître des expressions faciales en étant donc capable de s'affranchir des sources de variabilités que sont 1) les facteurs de variabilité intrinsèques tels que l'apparence du visage ou encore la manière de réaliser une expression donnée et 2) les facteurs de variabilité extrinsèques tels que les variations d'illumination, de pose, d'échelle, de résolution, de bruit ou d'occultations. Dans le même temps, nous examinons aussi l'apport de descripteurs spatio-temporels capables de prendre en compte des informations dynamiques utiles pour séparer les classes ambiguës.

La grosse limitation des méthodes de classification supervisée est qu'elles sont très coûteuses en termes de labélisation de données. Afin de s'affranchir en partie de cette limitation, nous avons étudié dans un second temps, comment utiliser des méthodes de transfert d'apprentissage de manière à essayer d'étendre les modèles appris sur un ensemble donné de classes d'émotions à des expressions inconnues du processus d'apprentissage. Ainsi nous nous sommes intéressés à l'adaptation de domaine et à l'apprentissage avec peu ou pas de données labélisées. La méthode proposée nous permet de traiter des données non labélisées provenant de distributions différentes de celles du domaine source de l'apprentissage ou encore des données qui ne concernent pas les mêmes labels mais qui partagent le même contexte. Le transfert de connaissance s'appuie sur un apprentissage euclidien et des réseaux de neurones convolutifs de manière à définir une fonction de mise en correspondance entre les informations visuelles provenant des expressions faciales et un espace sémantique issu d'un modèle de langage naturel. La correspondance entre les deux espaces est optimisée par alignement basé sur la distribution des descripteurs visuels.

Dans un troisième temps, nous nous sommes intéressés à la reconnaissance des micro-expressions faciales. Nous proposons un algorithme destiné à localiser ces micro-expressions dans une séquence d'images depuis l'image initiale (onset image) jusqu'à l'image finale (offset image) et à déterminer les régions des images qui sont affectées par les micro-déformations associées aux micro-expressions. Le problème est abordé sous un angle de détection d'anomalies ce qui se justifie par le fait que les déformations engendrées par les micro-expressions sont a priori un phénomène beaucoup plus rare que celles produites par toutes les autres causes

de déformation du visage telles que les macro-expressions, les clignements des yeux, les mouvements de la tête... Ainsi nous proposons un réseau de neurones auto-encodeur récurrent destiné à capturer les changements spatiaux et temporels associés à toutes les déformations du visage autres que celles dues aux micro-expressions. Ensuite, nous apprenons un modèle statistique basé sur un mélange de gaussiennes afin d'estimer la densité de probabilité de ces déformations autres que celles dues aux micro-expressions. Finalement les micro-expressions sont détectées au moyen d'une opération de seuillage sur cette densité de probabilité.

Tous nos algorithmes sont testés et évalués sur des bases d'expressions faciales actées et/ou spontanées.

**Mots clés:** Reconnaissance d'Expressions Macro-Faciales, Détection d'Expressions Micro-Faciales, Adaptation de Domaine, Zero-Shot Learning, Auto-codeur Convolutionnel Récurrent, 3D-CNN, 2D-CNN, Représentation Parcimonieuse, Sac de Mots Visuels.

---

GIPSA-lab Grenoble Images Parole Signal Automatique, adresse du laboratoire  
UMR 5216 CNRS - Grenoble INP - Université Grenoble Alpes  
11 rue des Mathématiques, Grenoble Campus,  
BP46, F-38402 SAINT MARTIN D'HERES Cedex

---

# Acknowledgment

I would like to start this acknowledgements by expressing my sincere gratitude to my supervisor professor Alice CAPLIER. She gave me the freedom to grow as an independent researcher and, at the same time, she provided help when it was needed. Her advice on both research as well as my career have been priceless. I want to thank her for her unlimited patience, understanding and encouragement. This thesis could not be accomplished without her guidance and aid.

I would like to give my heartfelt appreciation to my parents, who brought me up with their love and encouraged me to pursue my dreams. I would also like to express my special appreciation to my beloved family who accompanied me with their love, faith, and unconditional support over years.

My years at GIPSA-lab have been priceless. The friends I have met over the years have made the experience unique. I am especially proud that I got an opportunity to spend such quality time with all of them, especially at DIS department. I am going from GIPSA-lab full of great memories.

Specially, I want to thank a very special person who carries much meaning in my life. She has been a supporter, a friend and a constant source of encouragement. She is my love and my best friend.

Finally, I would like to gratefully acknowledge the members of my thesis committee. I am grateful for their thorough scrutiny of my work. I am very thankful for their time dedicated for participating in the mission of evaluation of this work.





# Contents

|   |            |
|---|------------|
| <b>List of Abbreviations</b>  | <b>xxi</b> |
| <b>1 Introduction</b>   | <b>1</b>   |
| 1.1 Facial Expression of Emotion . . . . .  | 2          |
| 1.2 The Characteristics and Challenges of Facial Expressions . . . . .                      | 4          |
| 1.2.1 Facial Expression Description: Labels versus Action Units . . . . .                   | 4          |
| 1.2.2 Emotion Categories: Basic versus Non-Basic Emotions . . . . .                         | 6          |
| 1.2.3 Facial Expression Spontaneity Variations: Posed versus Spontaneous . . . . .          | 8          |
| 1.2.4 Facial Expressions Types: Macro versus Micro . . . . .                                | 10         |
| 1.2.5 Source of Input for Facial Expressions: Still Images versus Image Sequences . . . . . | 11         |
| 1.2.6 FER Challenges: Intrinsic and Extrinsic Variabilities . . . . .                       | 12         |
| 1.3 Current FER Approaches . . . . .  | 14         |
| 1.3.1 FER Approaches via Message-Judgment . . . . .   | 14         |
| 1.3.2 FER Approaches via Sign-Judgment . . . . .  | 15         |
| 1.3.3 Discussion . . . . .  | 16         |
| 1.4 Thesis Contributions . . . . .  | 16         |
| 1.5 Thesis Outline . . . . .  | 18         |
| <b>2 A Review of Facial Expressions: Registration, Representation, and Recognition</b>      | <b>19</b>  |
| 2.1 Facial Expression Recognition Systems . . . . .   | 19         |
| 2.2 Facial Detection . . . . .  | 20         |
| 2.3 Face Registration . . . . .   | 21         |
| 2.4 Face Representation and Facial Feature Extraction . . . . .                             | 22         |

---

|          |   |           |
|----------|---|-----------|
| 2.4.1    | Spatial Representations and Feature Encoding . . . . .  | 23        |
| 2.4.2    | Spatio-Temporal Representations and Feature Encoding . . . . .  | 29        |
| 2.5      | Feature Discrimination and Dimensionality Reduction . . . . .   | 32        |
| 2.6      | Recognition . . . . .   | 33        |
| 2.7      | Conclusion . . . . .  | 34        |
| <b>3</b> | <b>Macro Facial Expression Analysis</b>   | <b>37</b> |
| 3.1      | Database Description and Training Protocol . . . . .  | 38        |
| 3.2      | A Low Level Feature Encoding Based on BoVW for FER . . . . .  | 41        |
| 3.2.1    | Beyond the Standard Bag of Visual Words Model . . . . .   | 44        |
| 3.2.2    | Facial Expression Classification Algorithm . . . . .  | 47        |
| 3.2.3    | State of the Art BoVW Representation Methods for Comparison Purpose   | 47        |
| 3.2.4    | Experimental Setup and Analysis . . . . .   | 49        |
| 3.2.5    | Conclusion . . . . .  | 53        |
| 3.3      | A Mid Level Feature Encoding Based on Sparse Representation for FER . . . . .                               | 54        |
| 3.3.1    | Formulation of Sparse Representation and Dictionary Learning . . . . .                                      | 57        |
| 3.3.2    | Related Works Regarding Classification Using Sparse Representation . .                                      | 61        |
| 3.3.3    | The SPFER Algorithm . . . . .   | 62        |
| 3.3.4    | Experimental Setup and Analysis . . . . .   | 66        |
| 3.3.5    | Conclusion . . . . .  | 70        |
| 3.4      | Hierarchical Spatial and Spatio-Temporal Feature Encoding Based on Deep<br>Neural Network for FER . . . . . | 72        |
| 3.4.1    | Hierarchical Spatio-Temporal FER Model . . . . .  | 73        |
| 3.4.2    | Network Training and Initialization . . . . .   | 83        |
| 3.4.3    | Data Augmentation for Facial Expressions Databases . . . . .  | 84        |
| 3.4.4    | Network Diagnosis: Visual Debugging . . . . .   | 85        |
| 3.4.5    | Hierarchical Spatio-Temporal to Hierarchical Spatial Feature Encoding<br>Image-Based for FER . . . . .      | 90        |

---

|          |  |            |
|----------|--|------------|
| 3.4.6    | Experimental Setup and Results Analysis Over the Spatio-Temporal Model . . . . . | 92         |
| 3.4.7    | Conclusion . . . . .   | 97         |
| <b>4</b> | <b>Adapting Facial Expression Models to New Domains or Tasks</b>                 | <b>101</b> |
| 4.1      | Introduction to DA and ZSL for FER . . . . .                                     | 101        |
| 4.2      | Joint Formulation of DA and ZSL and Challenges . . . . .                         | 105        |
| 4.3      | Related Works . . . . .  | 107        |
| 4.3.1    | Domain Adaptation . . . . .  | 107        |
| 4.3.2    | Zero Shot Learning . . . . .   | 110        |
| 4.4      | The Proposed Model for DA-FER and ZS-FER . . . . .                               | 114        |
| 4.4.1    | The Visual Model $f(\cdot)$ . . . . .  | 116        |
| 4.4.2    | The Semantic Model $g(\cdot)$ . . . . .  | 121        |
| 4.4.3    | Semantic Space Re-alignment Process . . . . .                                    | 125        |
| 4.4.4    | Inference Stage . . . . .  | 125        |
| 4.5      | Experimental Setup and Analysis Results . . . . .                                | 125        |
| 4.5.1    | DA-FER performance evaluation and analysis . . . . .                             | 125        |
| 4.5.2    | ZS-FER Performance Evaluation and Analysis Results . . . . .                     | 132        |
| 4.6      | Conclusion . . . . .   | 134        |
| <b>5</b> | <b>Micro Facial Expression Analysis</b>  | <b>137</b> |
| 5.1      | Introduction . . . . .   | 137        |
| 5.2      | Related Work . . . . .   | 139        |
| 5.2.1    | Micro Expressions Databases Elicitation and Training Protocol . . . . .          | 139        |
| 5.2.2    | Micro Expression Detection Methods . . . . .                                     | 142        |
| 5.2.3    | Anomaly Detection Based-Methods . . . . .  | 144        |
| 5.3      | Micro Expression Detection Algorithm . . . . .                                   | 145        |
| 5.3.1    | The Learning Stage . . . . .   | 146        |

---

|          |  |            |
|----------|--|------------|
| 5.3.2    | The Inference Stage . . . . .                                  | 154        |
| 5.4      | Experimental Setup & Analysis . . . . .                        | 156        |
| 5.4.1    | Evaluation Metrics . . . . .                                   | 157        |
| 5.4.2    | Parameters Evaluation and Discussion . . . . .                 | 157        |
| 5.4.3    | Detection Results over MiE Databases . . . . .                 | 159        |
| 5.4.4    | Feature Learning Strategies and Evaluation . . . . .           | 160        |
| 5.5      | Conclusion . . . . .   | 160        |
| <b>6</b> | <b>Thesis Summary, Future Work and Publications</b>            | <b>161</b> |
| 6.1      | Summary . . . . .  | 161        |
| 6.2      | Future Work . . . . .  | 163        |
| 6.3      | Publications . . . . .   | 164        |
| <b>7</b> | <b>Brief Review of Table 2.1</b>                               | <b>165</b> |
| 7.1      | Engineered Spatial Appearance Based Features . . . . .         | 165        |
| 7.2      | Engineered Spatial Geometric Based Features . . . . .          | 165        |
| 7.3      | Engineered Spatio-Temporal Appearance Based Features . . . . . | 166        |
| 7.4      | Engineered Spatio-Temporal Geometric Based Features . . . . .  | 166        |
| 7.5      | Learned Spatial Appearance Features . . . . .                  | 167        |
| 7.6      | Learned Spatio-Temporal Appearance Features . . . . .          | 167        |
| 7.7      | Copy Rights . . . . .  | 168        |
|          | <b>Bibliography</b>  | <b>194</b> |

# List of Figures

|     |  |    |
|-----|--|----|
| 1.1 | General scheme on facial expression sources of variabilities and influencing factors.  | 3  |
| 1.2 | Linguistic description of face using upper AUs (a) and lower AUs (b) and their facial appearance variations (adapted from [Ekman and Rosenberg 1997]). . . .   | 5  |
| 1.3 | The facial muscles that regulate facial expression movements [PashovAlex 2019].  | 5  |
| 1.4 | Examples of basic FEs. Images of the top row are taken from the MMI database [Pantic et al. 2005] and the images of the bottom row are taken from the DISFA database [Mavadati et al. 2013]. . . . .   | 6  |
| 1.5 | Non-basic emotional FEs samples. The images come from the DynEmo database [Tcherkassof et al. 2013]. . . . .   | 7  |
| 1.6 | The Duchenne smile (left) versus the social smile (right). Social smiles use only the mouth muscles. Whereas true smiles, known as Duchenne smiles, cause the eyes to twinkle and the cheeks to rise. . . . .  | 9  |
| 1.7 | The first three columns represent an example of “happy” faces performed in MaEs manner (upper row, the MMI database), and in MiEs (bottom row, the SMIC database [Li et al. 2013]). The final three columns represent another example with “disgust” emotion. . . . .  | 10 |
| 1.8 | Morphological steps of an angry facial expression. Images taken from CK+ database [Lucey et al. 2010]. The time is denoted as t and the neutral phase is denoted as NE. Dark green is related to the strongest intensity level. . . . .  | 11 |
| 2.1 | Generic pipeline for FER algorithms. The input is a peak facial expression image for spatial representations or a set of facial frames within a temporal window for spatio-temporal representations. The system outputs a discrete label if it is obtained through classification or a continuous signal if obtained through regression. . . . . | 20 |
| 2.2 | Our choices regarding facial expression recognition process and the input source of variabilities. Our choices are highlighted using colors. . . . .   | 35 |
| 3.1 | Challenges to face with MaE recognition. . . . .   | 38 |
| 3.2 | The databases represented from top to down correspond to: JAFFE, CK+, MMI, DISFA, DynEmo and MUG databases respectively. . . . .   | 39 |
| 3.3 | Standard BoVW representation for facial expressive image. . . . .  | 43 |

|      |  |    |
|------|--|----|
| 3.4  | The steps of computing the relative conjunction matrix in order to take into account the spatial organization of the data. . . . .   | 45 |
| 3.5  | The construction of a three level spatial pyramid. The image has different feature types, indicated by different colors. At the top, the facial image is sliced at two different levels of resolution. Then, for each resolution and each channel, the features that fall in each spatial bin are counted. Finally, each spatial histogram is weighted according to eq. (3.5). . . . .   | 48 |
| 3.6  | Classification accuracies of the FER system starting from the standard BoVW method successively improved either with TF.IDF or RCM to finish with the complete model ImpBoVW (SBoVW + TF.IDF + RCM). SIFT descriptors are used to extract low level features from keypoints. For each configuration of the algorithm, different keypoint detectors and different clustering methods are tested. (a): Over posed basic Ekman's MaEs using the JAFFE database. (b): Over spontaneous non-basic MaEs using the DynEmo database. . . . . | 51 |
| 3.7  | (a) Performances over five databases compared with SBoVW and SP BoVW while using SIFT descriptor. (b): Computational times for generating the BoVW features and classifying them into emotional categories. . . . .  | 52 |
| 3.8  | The raw input image corresponds to a frighten spontaneous expression class along its corresponding HOG features that encode gradient orientations. . . . .   | 52 |
| 3.9  | The recognition rates of the ImpBoVW on different databases. The <i>minimal values</i> correspond to <i>HOG</i> . The <i>red line</i> corresponds to <i>SIFT</i> . The <i>maximum values</i> correspond to their combination, <i>HOG+SIFT</i> . . . . .  | 53 |
| 3.10 | Graphical illustration of sparse representation concept. An input signal can be approximated by a linear combination of a set of atoms that compose a dictionary, such that the sparse code has as few as possible non-zero coefficients. For instance, the input signal here is reconstructed using two atoms out of ten. The sparse code is used as the feature vector for training the classifier. . . . .  | 55 |
| 3.11 | SPFER algorithm composed of three main stages, first the pre-training for dictionary initialization, then the sparse coding and the dictionary refining and finally the classification stage. . . . .  | 57 |
| 3.12 | Performances of the linear SVM classifier for a variety of projected dimensions to evaluate RFFD over acted databases (a) JAFFE, (b) CK+ and (c) MMI. . . . .  | 67 |
| 3.13 | Performances of the linear SVM classifier for a variety of projected dimensions to evaluate RFFD over spontaneous databases (a) DISFA and (b) DynEmo. . . . .  | 67 |
| 3.14 | Different dimensionality reduction technique performances. . . . .   | 68 |

|  |    |
|--|----|
| 3.15 (a) Evaluation of the SPFER algorithm with different sparsity levels. The x-axis represents the sparsity level. (b): Evaluation of different variants of the SPFER algorithm. . . . .   | 69 |
| 3.16 General model architecture for learning local and global hierarchical spatio-temporal features for FER. . . . .   | 73 |
| 3.17 3D Convolutional block architecture for encoding local variations. In this figure, $sl$ :sequence length, $f$ : feature maps, $s$ : stride, $a$ : activation, $h$ : height, $w$ :width, $c$ :channels. . . . .  | 75 |
| 3.18 Convolutional Long Short-Term Memory Cell. The numbers refer to the corresponding equations. The inputs coming from different sources get convoluted with their filters, added up along with bias. The cell operates by learning gate functions that determine whether an input is significant enough to be memorized, to be forgotten or to be sent to the output. By using a gated way for sorting information over short or long time ranges, the discriminant spatio-temporal information is extracted. . . . . | 79 |
| 3.19 ConvLSTM architecture for extracting long term dependencies. . . . .  | 80 |
| 3.20 Weighted spatial pyramid pooling layer. . . . .   | 81 |
| 3.21 Classification stage: mapping the final representation into a probability distribution. The system outputs one label for every input sequence. . . . .  | 82 |
| 3.22 Augmenting image data by introducing geometric deformations. . . . .  | 84 |
| 3.23 Learning curves and accuracy rates during the training and validation phases. The model is trained over a combination of MMI and CK+ databases. It is validated over the CK+ and the MMI validation sets. . . . .   | 86 |
| 3.24 3DConv-1 weight histograms through the learning procedure with an appropriate hyper-parameter choice. . . . .   | 87 |
| 3.25 3DConv-1 weight histograms through the learning procedure with an inappropriate hyper-parameter choice. . . . .   | 87 |
| 3.26 3DConv-1 bias and weight distribution values over time. . . . .   | 87 |
| 3.27 2D visualization for the FC layer outputs over the test image sequences from the combination of MMI and CK+ databases using t-SNE projection and their corresponding confusion matrix. . . . .  | 88 |
| 3.28 One of the feature map responses over time for a temporal sequence that corresponds to an <i>Angry</i> facial expression taken from the first and the second 3D-Conv block. . . . .   | 89 |



|      |  |     |
|------|--|-----|
| 3.29 | General model architecture for learning spatial hierarchical features for FER using static images. . . . .   | 90  |
| 3.30 | The 2D-CNN model results using the same test data protocol over the algorithms SPFER and ImpBoVW. The values are shown in Table 3.8. . . . .   | 91  |
| 3.31 | Frontal and profile face image sequences for Sad expression. In this example, all the frontal face image sequences are correctly classified. For the profile faces, some examples are falsely classified and confused with the Disgust expression. . . . .   | 94  |
| 3.32 | Normalized confusion matrices over the (a) DISFA database, (b): DynEmo database. . . . .   | 94  |
| 3.33 | Facial expression input source variabilities with respect to the average performances over spatial and spatio-temporal models. The reported numbers are computed by averaging the classification rates of each model over either acted or spontaneous databases. . . . .   | 98  |
| 4.1  | Regression model for establishing an embedding space that maps the visual image signatures $\mathbf{x}_i$ onto the semantic prototypes $\mathbf{y}_k$ . . . . .  | 105 |
| 4.2  | (a) Illustration of the domain shift problem where the feature points of the source domain and target domain are both shifted from each other and from their class prototype. (b) We aim to reunify samples from source and target domains in a common invariant space around their semantic prototype and to apply the learned model to new categories never seen before. Different classes are represented by different colors and shapes. . . . .   | 106 |
| 4.3  | An example of visual attribute description. Adapted from [Farhadi et al. 2009]   | 110 |
| 4.4  | A description by high-level attributes allows the transfer of knowledge between object categories: after learning the visual appearance of attributes from any class with training examples, one can detect also object classes that do not have any training images, based on which attribute description a test image fits best. Adapted from [Lampert et al. 2009] . . . . .  | 111 |
| 4.5  | Global overview of the DA-FER and ZS-FER methods. $\mathbf{x}$ is the encoded visual signature obtained via a deep CNN model. $\boldsymbol{\alpha}$ is the visual feature distribution obtained via the Soft Attention Model. $\mathbf{y}$ is the semantic class prototype that describes the class label or its visual attribute description obtained via an NLP model. $\mathbf{sa}$ is the re-aligned semantic vectors. The whole system is trained in an end-to-end fashion using the source domain information and produces the estimated class label $z$ . . . . . | 115 |

- 
- 4.6 Zoom on the visual model modules. Our model is composed of a deep CNN module and another two sequential Soft Attention modules, one for discovering the feature probability distribution and another for discovering the location probability distribution. Our architecture can learn which information to emphasize or suppress and provides a compact visual signature vector. . . . . 116
- 4.7 Module arrangements: a deep CNN integrated with region layers (RL), followed by a feature soft attention (SFA-1) and a location soft attention (SFA-2). The whole model produces  $\mathbf{x}_i$  for an image  $I_i$ . . . . . 117
- 4.8 A feature map is output from Conv1, and uniformly divided into  $4 \times 4$  patches. Each  $60 \times 60$  pixel patch is processed with Instance Normalization (IN), followed by Parametric Rectified Linear Unit (PReLU) activation and then locally convolved with  $3 \times 3$  filters with the same number of Conv1 channels. Afterwards, each original patch is re-weighted by adding it to the convolved one. The output of the RL is the concatenation of all re-weighted patches. . . . . 118
- 4.9 A demonstration for the attended locations for a surprised emotion. . . . . 120
- 4.10 An outline of the proposed semantic representation for the DA-FER and ZS-FER methods respectively. An NLP model encodes the class name or its description into a continuous feature vector(s). For ZS-FER, the class name is used and thus a semantic vector  $\mathbf{y}_k \in \mathbb{R}^{300}$  is produced. For DA-FER, 24 visual attributes that describe each class are provided as an input and the model produces for each attribute entry a vector  $\mathbf{y} \in \mathbb{R}^{300}$ . A set of 24 semantic vectors are then fed to a soft attention model SFA-3 to encode the relationships between those vectors and to provides a compact semantic vector  $\mathbf{y}_k \in \mathbb{R}^{300}$ . . . 121
- 4.11 Attribute entry dependencies among samples either from the same or from different classes. The vertical red lines represent the limit among the different classes: angry, disgust, fear, happiness, sadness, surprised and neutral respectively. Five images per class are shown, *e.g.* the first five columns correspond to the angry class. Each column represents the probability distribution  $\nu_{\mathbf{k}}$  that captures the percentage of importance of each semantic entry. . . . . 124
- 4.12 Feature representation for image visual signatures coming from different classes and projected over their semantic class prototype (black dot) using data coming from: *Source Domain (a)* and *Target Domain (b)*. . . . . 127
- 4.13 t-SNE projection of visual and semantic signatures on the first two dimensions for target domain DISFA dataset and their confusion matrices. . . . . 131
- 4.14 (a) Confusion Matrix over the DynEmo database using our model ZS-FER. (b) Confusion Matrix over the DynEmo database using the VAWE model. . . . . 134

|      |   |     |
|------|---|-----|
| 5.1  | Spatio-temporal MiEs detection. A face with a micro-expression that appears around the lip corners associated with a fast eye blinking at the same time. . .  | 138 |
| 5.2  | An example of frame sequence from CASME-i, including onset frame, apex frame and offset frame. The facial MiE movement associated to this sequence is lip corner depressor. . . . .   | 140 |
| 5.3  | General block diagram of the ADS-MiE. . . . .   | 147 |
| 5.4  | Spatio-temporal feature learning for NFBs. Both the Encoder and the Decoder are made up of multilayered CNNs and ConvLSTMs. . . . .   | 149 |
| 5.5  | Filters responses at convolutional layer number 2. The first column on the left represents the original patch. . . . .  | 150 |
| 5.6  | The reconstructed blocks over the first 5 time instances. . . . .   | 152 |
| 5.7  | Weighted sum of 5 component densities. . . . .  | 153 |
| 5.8  | Decision making process via adaptive thresholding. The two vertical blue lines are the temporal ground truth while the vertical shaded pink areas are our prediction. The horizontal shaded pink areas correspond to the threshold. . . .   | 155 |
| 5.9  | Decision making process over a video without any MiE. (a): weighted log-likelihood score over time for each image patch; (c), (d) & (e): Mean pooling over all blocks in (a); (b): smoothed curve with 1-D Gaussian filtering (c); The horizontal shaded areas represents the threshold. The vertical shaded areas represents the prediction of MiE segments. . . . . | 156 |
| 5.10 | Studying the effects of different time windows $TW = \{10, 20, 30\}$ , temporal Multiscaling $S = \{1, 2, 3\}$ , and thresholding techniques (Adaptive: <i>AdTh</i> or Cross Validation: <i>CV</i> ) over CASME-i database with $M = \{5\}$ . . . . .   | 158 |

# List of Tables

|      |   |    |
|------|---|----|
| 1.1  | Emotions in terms of the most 16 active AUs [Du et al. 2014a]. . . . .  | 5  |
| 1.2  | Emotions categorization [Adolphs 2002]. . . . .   | 7  |
| 1.3  | Properties of an ideal facial expression analysis system. . . . .   | 14 |
| 2.1  | Summary of facial representations and feature extraction for FER systems proposed in the literature. . . . .  | 23 |
| 3.1  | A summary about the macro facial expression databases and their training protocol used in this thesis. . . . .  | 42 |
| 3.2  | Dictionary size for each of the facial expression database and the corresponding size of the sparse matrix. . . . .   | 68 |
| 3.3  | Recognition rates in % over all databases using state of the art methods. . . . .   | 70 |
| 3.4  | Recognition rates in % over all databases using low and mid level features. . . . .   | 70 |
| 3.5  | Low-level versus mid-level feature representation power over class prediction. The first three rows correspond to the DynEmo database. The last four rows correspond to the DISFA database. . . . .   | 71 |
| 3.6  | The total number of image sequences before and after data augmentation considered for training, validation and testing. The database description and its protocol are also provided in Table 3.1, Section 3.1. . . . .  | 85 |
| 3.7  | The total number of images before and after data augmentation considered for the training, validation and testing phases. . . . .   | 91 |
| 3.8  | Recognition rates in % over all databases using low, mid level and hierarchical features using same data protocol. . . . .  | 91 |
| 3.9  | The 2D-CNN model results over test data with data augmentation and over the same test data used for ImpBoVW and SPFER. The protocol is presented in Table 3.7, the last row. . . . .  | 92 |
| 3.10 | Hierarchical spatio-temporal model classification performances over acted and spontaneous databases having short and long image sequences and encompassing various intrinsic and extrinsic variations. The sign $\pm$ indicates the average variance of our model when we repeat the same experiment three times. The variance is directly correlated to the parameters initialization. . . . . | 93 |

|      |   |     |
|------|---|-----|
| 3.11 | State of the art recognition rates in %.  | 95  |
| 3.12 | Recognition rates in % over dynamic acted expressions.  | 96  |
| 3.13 | Recognition rates over dynamic spontaneous expressions.   | 96  |
| 4.1  | Examples of visual domain shift. Performance degradation of the classification method based on 2D-CNN model (Section 3.4.5) when trained and tested on image domains coming from two different distributions: the MMI dataset as the source domain and the DISFA dataset as the target domain.  | 103 |
| 4.2  | Graphical representation of the across-class learning task: dark gray nodes are always observed, light gray nodes are observed only during training, while white nodes are never observed but must be inferred. An ordinary classifier learns one parameter $\theta_k$ for each training class. It cannot generalize to classes $z_k^T$ that are not part of the training set. In an attribute-based classifier (DAP) with fixed class–attribute relations (thick lines), training labels $z_k^S$ imply training values for the attributes $\theta_m$ , from which parameters $\beta_m$ are learned. At test time, attribute values can directly be inferred, and this implies output class labels even for unseen classes. A multi-class based attribute classifier (IAP) combines both ideas: multi-class parameters $\theta_k$ are learned for each training class. At test time, the posterior distribution of the training class labels induces a distribution over the labels of unseen classes by means of the class–attribute relationship. | 112 |
| 4.3  | Emotion in terms of prototypical facial AUs. Adapted from [Du et al. 2014a].  | 122 |
| 4.4  | Basic emotional categories with their visual attribute vector.  | 123 |
| 4.5  | Facial expression databases used for Da-FER and ZS-FER validations  | 126 |
| 4.6  | Ablation study: accuracies % on DA target and source data with different models which are variant of the DA-FER approach.   | 129 |
| 4.7  | Benchmarking our proposed DA-FER with the state of the art in DA.   | 130 |
| 4.8  | Zero-shot performances over variant models of the ZS-FER approach using ZS data protocol.   | 132 |
| 4.9  | Zero-shot performances in comparison with the state of the art on ZS data protocol.   | 132 |
| 5.1  | A summary of the current spontaneous micro-expression databases with the devised data protocol for model validation.  | 140 |
| 5.2  | Performance evaluation over the three micro facial expressions databases.   | 159 |

|     |  |     |
|-----|--|-----|
| 5.3 | Reported AUC values in the state of the art. . . . .                             | 159 |
| 5.4 | Evaluation using different feature representations. AUC values are reported. . . | 160 |



# List of Abbreviations

|               |                                       |
|---------------|---------------------------------------|
| <b>FE</b>     | Facial Expression                     |
| <b>FER</b>    | Facial Expression Recognition         |
| <b>HCI</b>    | Human Computer Interaction            |
| <b>AU</b>     | Action Unit                           |
| <b>FACS</b>   | Facial Action Coding System           |
| <b>EMFACS</b> | Emotional Facial Action Coding System |
| <b>MaE</b>    | Macro-Facial Expression               |
| <b>MiE</b>    | Micro-Facial Expression               |
| <b>AC</b>     | Affective Computing                   |
| <b>LBP</b>    | Local Binary Pattern                  |
| <b>SIFT</b>   | Scale-Invariant Feature Transform     |
| <b>SURF</b>   | Speeded-Up Robust Features            |
| <b>HOG</b>    | Histogram of Oriented Gradient        |
| <b>SVM</b>    | Support Vector Machines               |
| <b>RF</b>     | Random Forest                         |
| <b>k-NN</b>   | k-Nearest Neighbours                  |
| <b>DNN</b>    | Deep Neural Networks                  |
| <b>CNN</b>    | Convolutional Neural Network          |
| <b>RNN</b>    | Recurrent Neural Network              |
| <b>ANN</b>    | Artificial Neural Network             |
| <b>DBN</b>    | Dynamic Bayesian Network              |
| <b>AAM</b>    | Active Appearance Model               |
| <b>ROI</b>    | Region of Interest                    |
| <b>ICA</b>    | Independent Component Analysis        |
| <b>PCA</b>    | Principle Component Analysis          |



|                 |  |
|-----------------|--|
| <b>BoVW</b>     | Bag-of-Visual-Words  |
| <b>DoG</b>      | Difference of Gaussian                                       |
| <b>NMF</b>      | Non-negative Matrix Factorization                            |
| <b>3D-SIFT</b>  | 3 Dimensional Scale-Invariant Feature Transform              |
| <b>3D-HOG</b>   | 3 Dimensional Histogram of Oriented Gradient                 |
| <b>STIP</b>     | Space Time Interest Points                                   |
| <b>3D-CNN</b>   | 3D Convolutional Neural Network                              |
| <b>ConvLSTM</b> | Convolutional Long Short Term Memory                         |
| <b>DCT</b>      | Discrete Cosine Transform                                    |
| <b>RFS</b>      | Random Features Selection                                    |
| <b>RP</b>       | Random Projection  |
| <b>JAFFE</b>    | The Japanese Female Facial Expression                        |
| <b>CK+</b>      | The Extended Cohn-Kanade Facial Expression                   |
| <b>MMI</b>      | The MMI Facial Expression                                    |
| <b>DISFA</b>    | The Denver Intensity of Spontaneous Facial Actions           |
| <b>fps</b>      | frames per second  |
| <b>DynEmo</b>   | The Video database of Natural Facial Expressions of Emotions |
| <b>MUG</b>      | The MUG Facial Expression                                    |
| <b>IDF</b>      | Inverse-Document-Frequency                                   |
| <b>TF.IDF</b>   | Term-Frequency Inverse-Document-Frequency                    |
| <b>SP BoVW</b>  | Spatial Pyramid Bag of Visual Words                          |
| <b>TF</b>       | Term Frequency   |
| <b>PMK</b>      | Pyramid Matching Kernel                                      |
| <b>K-SVD</b>    | K-Singular Value Decomposition                               |
| <b>OMP</b>      | Orthogonal Matching Pursuit                                  |
| <b>MP</b>       | Matching Pursuit   |
| <b>SRC</b>      | Sparse Representation Based Classification                   |
| <b>LC-KSVD</b>  | Label Consistent K-Singular Value Decomposition              |

---

|                |  |
|----------------|--|
| <b>2D-CNN</b>  | 2D Convolutional Neural Network                        |
| <b>NFB</b>     | Natural Facial Behaviour                               |
| <b>SPFER</b>   | Spontaneous Facial Expression Recognition              |
| <b>DA</b>      | Domain Adaptation                                      |
| <b>ZS</b>      | Zero Shot  |
| <b>ZSL</b>     | Zero Shot Learning                                     |
| <b>ADS-MiE</b> | Anomaly Detection System for Micro-Expression Spotting |
| <b>ms</b>      | milliseconds   |
| <b>ReLU</b>    | Rectified Linear Unit                                  |
| <b>tanh</b>    | Hyperbolic tangent                                     |
| <b>t-SNE</b>   | T-Distributed Stochastic Neighbor Embedding            |
| <b>LBP-TOP</b> | Local Binary Pattern Three Orthogonal Planes           |
| <b>ASM</b>     | Active Appearance Model                                |
| <b>HMM</b>     | Hidden Markov Model                                    |
| <b>SOM</b>     | Self-Organizing Maps                                   |
| <b>ImpBoVW</b> | Improved Bag of Visual Words                           |
| <b>TF-IDF</b>  | Term-Frequency Inverse-Document-Frequency              |
| <b>SP BoVW</b> | Spatial Pyramid Bag of Visual Words                    |
| <b>SBoVW</b>   | Standard Bag of Visual Words                           |
| <b>RCM</b>     | Relative Conjunction Matrix                            |
| <b>LSTM</b>    | Long Short Term Memory                                 |
| <b>SPP-Net</b> | Spatial Pyramid Pooling Network                        |
| <b>RFFD</b>    | Random Face Feature Descriptor                         |
| <b>ADAM</b>    | Adaptive Moment Estimation                             |
| <b>PDF</b>     | Probability Density Function                           |
| <b>FC</b>      | Fully Connected  |
| <b>GMM</b>     | Gaussian Mixture Models                                |
| <b>MLP</b>     | Multilayer Perceptron                                  |

|                 |   |
|-----------------|---|
| <b>RCAE</b>     | Recurrent Convolutional AutoEncoder           |
| <b>MDN</b>      | Mixture Density Network                       |
| <b>RBF</b>      | Radial Basis Function                         |
| <b>AUC</b>      | Area Under the ROC Curve                      |
| <b>MAD</b>      | Mean Average Duration                         |
| <b>MAS</b>      | Mean Average Shift                            |
| <b>DWE</b>      | Distributed Word Embeddings                   |
| <b>word2vec</b> | Word to Vector                                |
| <b>GloVe</b>    | Global Vectors for Word Representation        |
| <b>LCN</b>      | Locally Connected Layers                      |
| <b>RL</b>       | Region Layer                                  |
| <b>PReLU</b>    | Parametric Rectified Linear Unit              |
| <b>SimNet</b>   | Similarity-Based Classifier Network           |
| <b>DA-LS</b>    | Domain Adaptation Method using Linear Setting |
| <b>DAP</b>      | Direct Attribute Prediction                   |
| <b>IAP</b>      | Indirect Attribute Prediction                 |
| <b>IN</b>       | Instance Normalization                        |
| <b>VAWE</b>     | Visually Aligned Word Embedding               |
| <b>SynC</b>     | Synthesized Classifiers                       |
| <b>DeViSE</b>   | Deep Visual Semantic Embedding                |
| <b>ConSE</b>    | Convex Combination of Semantic Embeddings     |
| <b>MTL</b>      | Multi-Domain and Multi-Task Learning          |
| <b>LatEm</b>    | Latent Embedding                              |
| <b>VdSA</b>     | Visually-Driven Semantic Augmentation         |
| <b>SFA</b>      | Soft Attention                                |
| <b>NLP</b>      | Natural Language Processing                   |
| <b>CASME</b>    | Chinese Academy of Sciences Micro-expression  |
| <b>SMIC-HS</b>  | Spontaneous Micro-expression High Speed       |

|            |                                |
|------------|--------------------------------|
| <b>TIM</b> | Temporal Interpolation Model   |
| <b>MKL</b> | Multiple Kernel Learning       |
| <b>TMS</b> | Temporal Multiscaling Sampling |
| <b>TW</b>  | Temporal Window                |



# Introduction

---

Faces play a major role in social communication and interactions. They are characterized by multiple of communication channels, such as the auditory channel (carrying speech) and the visual channel (carrying facial expressions), that act on multiple modalities to perceive signals from the outside world [Bruce 1993; Takeuchi and Nagao 1993]. They carry a broad variety of rich information that reveals person's identity [Phillips and O'toole 2014], demographic information (*e.g.* gender, age, race) [Han et al. 2015], emotional states [Koelstra et al. 2010] or other socially relevant categories. While humans are able to detect, identify, and analyse faces in a scene quickly, efficiently and effortlessly, yet building an automated system that carries out these tasks is very problematic. For instance, there are considerable related difficulties with respect to detect a pattern as a face, to identify a face, to analyse of facial expressions, *etc.* The capacity of the human visual system to overcome these problems is suggested by Pantic and Rothkrantz 2000 to be considered as a reference point for designing an automated affective computing system qualified to analyse facial expressions and recognize emotions. A system that executes these operations could be helpful for many applications, *e.g.* pain detection, computer-aided learning for people with autism, and driver assistant for drowsiness detection *etc.*

This chapter is an introductory study to the problem of facial expression analysis where we describe the link between facial expressions and emotion. We introduce the basic structure of any automated facial expression system and the challenges it has to deal with. Afterwards, we introduce the current approaches of facial expression recognition. Eventually the contributions followed by the overall organization of the thesis are described.

## 1.1 Facial Expression of Emotion

The human face permits to a person to send social signals such as Facial Expression (FE)s, vocal, linguistic and other physiological signals. FEs are the correlation of facial muscle changes beneath the skin of the face. They are a prime facet to estimate emotion-related activities such as our current focus of attention, emotional state, signal comprehension or disagreement. They are one of the most powerful ways for communicating and characterizing the displayed emotion over the face and other mental, social, and physiological clues.

The origins of human FE analysis of emotions go back to the earliest scientific exploration by Boulogne 1862 who performed experimental manipulations of FE activations by applying Galvanic electrical stimulations directly to the facial muscles. Boulogne 1862 illustrated his view that there are different muscles in the human face that are separately responsible for each individual emotion. Then, Darwin and Prodger 1872 wondered whether there might be a few set of core emotions that are expressed with great stability worldwide and across cultures and proposed the concept of universal FEs in man and animals. Ekman and Friesen 1978 have performed comprehensive studies of FEs, giving evidence to support this universality theory. The “universal FEs” are those representing prototypical (basic) emotions, namely: (1) *anger*, (2) *disgust*, (3) *fear*, (4) *happiness*, (5) *sadness*, (6) *surprise*, plus (7) *neutral*. Psychological theories on universality and interpretation of facial expressions in terms of basic emotion categories has become the most prevalent model for research on emotion recognition. It possesses the advantage that FEs pertaining to basic emotions are smoothly recognized and described by humans, despite the differences imposed by social rules.

For many decades, FE analysis was primarily a research domain for psychologists, until Suwa 1978 presented a preliminary study on automatic FE analysis from images. In the nineties, automatic Facial Expression Recognition (FER) becomes an active and challenging research topic in computer vision. It has been introduced as a pattern recognition and learning problem that deals with the extraction and representation of facial motions and facial feature deformations using images or sequences for categorization in term of interpretative abstract emotion labels. The basic structure to automatic FER consists of three main stages: face detection, facial feature extraction and representation, and recognition. Mase 1991 pioneered automatic FER using optical flow and then Essa and Pentland 1994 and Iwano et al. 1996 utilized optical flow computation to identify the direction of motions that are caused by FEs and categorized them into basic emotional categories. Picard 1997 built various models for recognizing emotions and presented new application areas of affective wearable computers. The achievements in the complementary areas such as psychological studies, human movement analysis, face detection and tracking, and face recognition facilitate the improvements of automatic FER.

Yet, computers remain emotionally challenged as they neither recognize the user’s emotions nor possess emotions of their own [Sebe et al. 2007]. Aiding the computer to have some perception of the emotional state of humans can manage to create a new level of affection in Human Computer Interaction (HCI). The advantages of automated FER could evolve the level of interaction from unidirectional to bidirectional with computing devices. Therefore it

could provide affect-sensitive HCI systems, which include:

1. A multimodal affective user interface system, such as the emotion assessment proposed by Zhou Jianzhong et al. 2005.
2. A clinical affective vision system, such as the pain detection proposed by Lucey et al. 2011.
3. A driver assistant system, such as the driver drowsiness detection proposed by Assari and Rahmati 2011
4. An affective tutoring system, such as the one proposed by Chao et al. 2012 which uses emotion recognition techniques to improve interest in learning by recognizing the emotional states of students during their learning processes and giving adequate feedbacks.
5. A cognitive emotional system, such as the elder-care robot proposed by Jing et al. 2015 based on the smart home and FER.

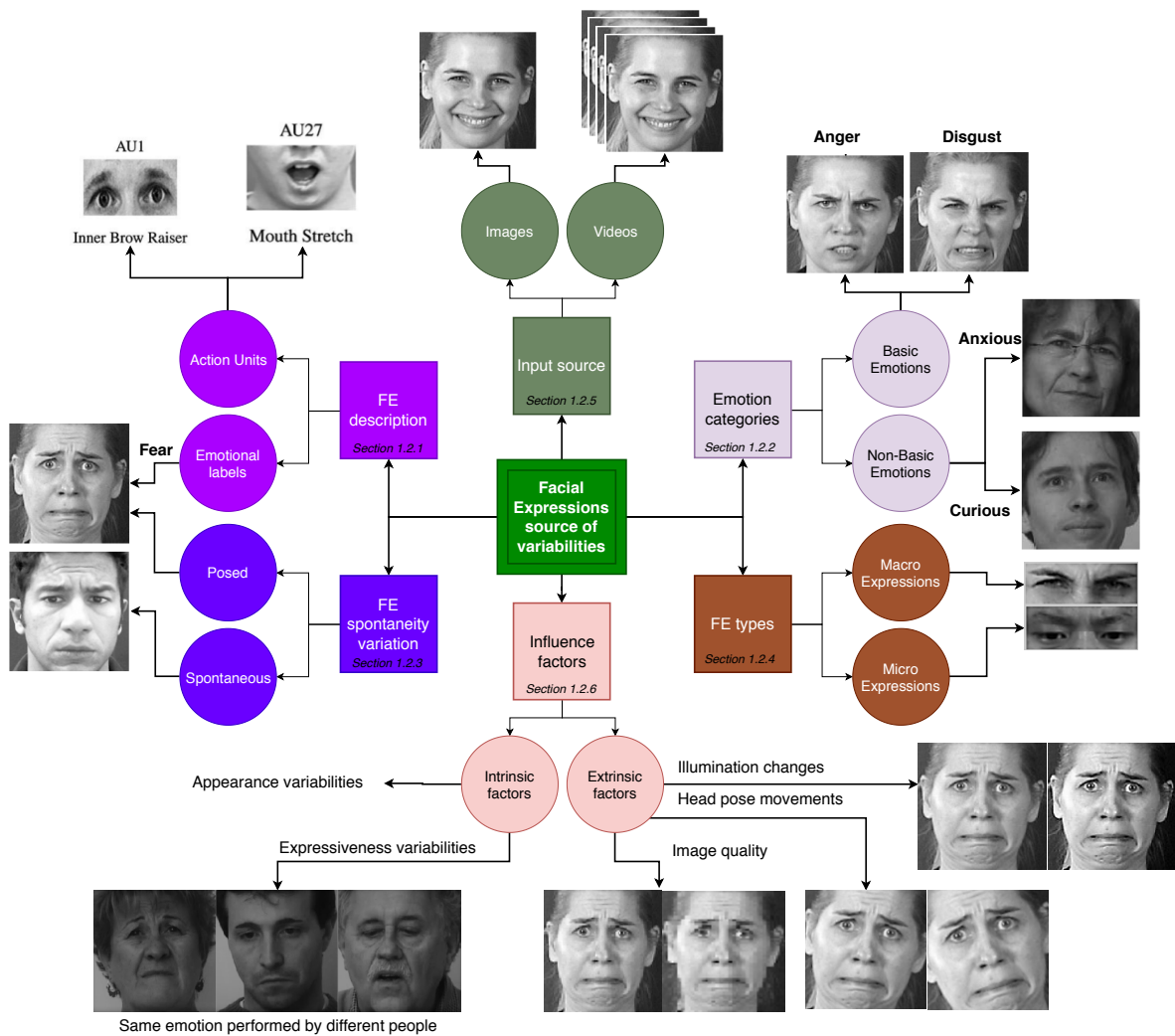


Figure 1.1: General scheme on facial expression sources of variabilities and influencing factors.



## 1.2 The Characteristics and Challenges of Facial Expressions

Automated facial expression analysis is a very challenging task. Several choices have to be specified and several factors that influence its performance have to be taken into account. Figure 1.1 demonstrates those variabilities, which include:

1. The level of description of FE: labels versus facial visual attributes, Section 1.2.1.
2. The type of emotion categories: basic versus non-basic emotions, Section 1.2.2.
3. The spontaneity of FE in terms of visual appearance: posed versus spontaneous, Section 1.2.3.
4. The type of FE in term of subtlety and speed of motion: macro versus micro facial muscle movements, Section 1.2.4.
5. The source of input: images versus videos, Section 1.2.5.
6. The influence factors on recognizing emotions: intrinsic and extrinsic factors, Section 1.2.6.

### 1.2.1 Facial Expression Description: Labels versus Action Units

To interpret FEs, that is to infer or describe what underlies a visual and temporal deformation of a face, emotion detection and facial muscle action (a.k.a Action Unit (AU)) detection are two major approaches that have been dominated [Cohn 2007]. These models are derived from two leading paths in psychological researches that focus on FE measurement: message-judgment and sign-judgment [Pantic and Rothkrantz 2000].

- The message-judgment approach aims to describe FEs in terms of a set of discrete affective labels such as basic emotions or other set of emotional labels.
- The sign-judgment approach aims to describe the displayed FEs (facial movement or facial component shape) in terms of activated facial muscle movements or action units (AUs) through the Facial Action Coding System (FACS) [Ekman and Friesen 1978], former to the final classification.

The FACS is an anatomical based system for describing all visual facial deformations and apparent facial movements. It decomposes FEs into individual components of muscle movements, called AUs. The AUs with their linguistic description are presented in Figure 1.2. In general, there is a total of forty-four predefined AUs, which correspond to the upper and lower facial areas. These AUs describe facial actions with regard to their location as well as their intensity [Fasel and Luetin 2003]. For instance, twelve upper AUs as shown in Figure 1.2a are connected with eye gaze direction and head orientation.

The anatomical structure of the facial muscles that regulate FEs is presented in Figure 1.3. The frontalis muscles pars medialis and pars lateralis, AU1 and AU2 respectively, are responsible for lifting the inner and the outer corner of eyebrows. AU1 and AU2 create horizontal forehead wrinkles on a surprised face. Orbicularis oculi pars orbitalis and pars palpebralis, AU6 and AU7 respectively, are the circular muscles of the eye. AU6 is responsible

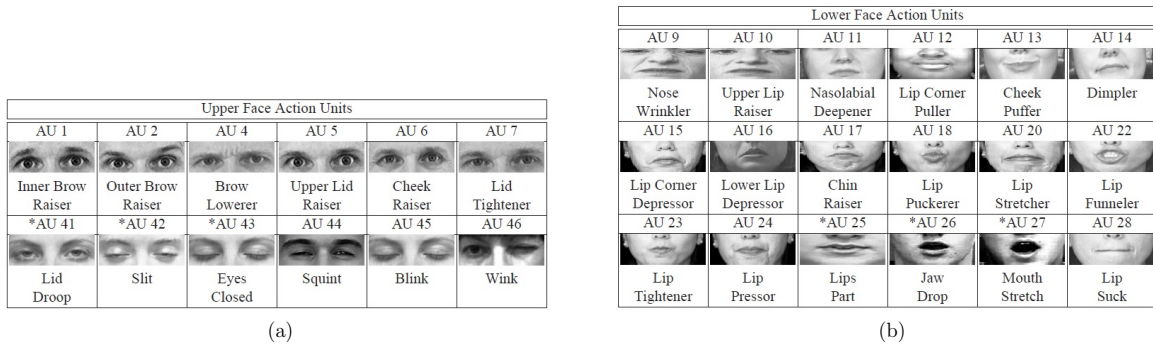


Figure 1.2: Linguistic description of face using upper AU (a) and lower AU (b) and their facial appearance variations (adapted from [Ekman and Rosenberg 1997]).

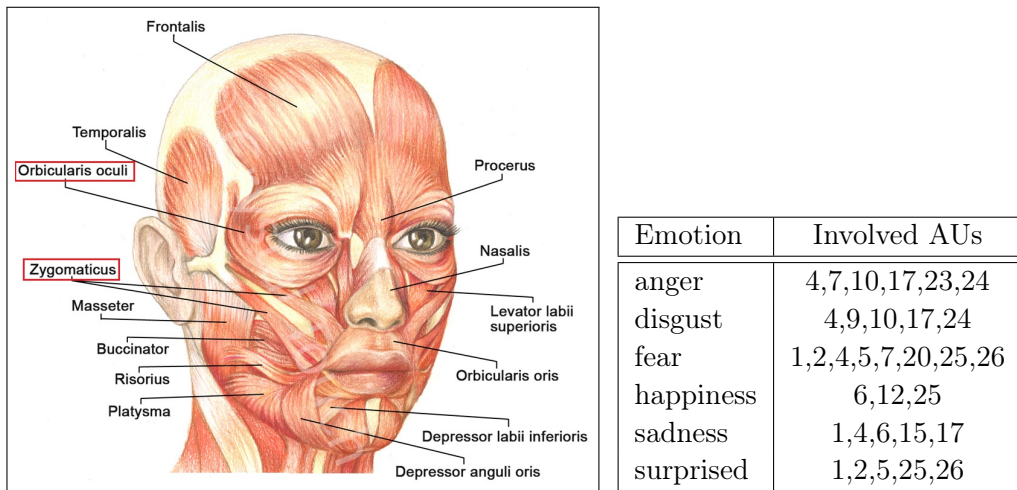


Figure 1.3: The facial muscles that regulate facial expression movements [PashovAlex 2019].

Table 1.1: Emotions in terms of the most 16 active AUs [Du et al. 2014a].

for raising the cheeks and narrowing the eyes while AU7 is responsible for raising the lower eyelid. Procerus is a frown muscle. The corrugator superclii muscle, AU4, pulls the eyebrows together. The major and the minor Zygomatic muscles, AU12 and AU13, move the mouth corners up and outward when someone smiles. The risorius, AU20, is a smile muscle. It pulls mouth corners laterally and forms dimples in the cheeks. This muscle is not always active in all people. The Orbicularis oris muscles, AU22, AU23 and AU24, are the circular muscles of the mouth. They tightly gather the lips and bring mouth corners towards the middle line. The depressor anguli oris muscle, AU15, pulls the mouth corners downward and inward. The levator labii superioris muscle, AU10, and the depressor labii inferioris muscle, AU25, pulls the upper and the lower lips up and down respectively when someone grins. Mentalis, AU17, is the chin muscle and it pulls up the chin as someone expresses disappointment, doubt and some other negative emotions. Platysma, AU20, is a surface muscle of the neck. It is engaged in the expressions of fear, disgust and other negative emotions.

A single FE can be encoded in terms of one AU or several AUs. FACS is considered to provide a solid basis for the formulation of emotion categories. Even though, AUs are only descriptive in terms of facial movements, they do not provide any information about the message they represent. Therefore, to relate and map facial AUs to discrete affective labels, Ekman and Rosenberg 1997 introduced the Emotional Facial Action Coding System (EMFACS) that specifies which facial AUs are common for a particular emotion. Table 1.1 shows that most of the universal FEs can be described using a sub-set of AUs.



Figure 1.4: Examples of basic FEs. Images of the top row are taken from the MMI database [Pantic et al. 2005] and the images of the bottom row are taken from the DISFA database [Mavadati et al. 2013].

As it can be seen in Figure 1.4, the message-judgment approach is all about interpretation of the conveyed message through the FE, that is assigning a discrete emotional label for labeling a particular facial behavior. On the contrary, sign-judgment approach as illustrated in Table 1.1, tries to leave inference to higher order decision-making, such as mapping AUs onto emotion-specified expressions (*e.g.* happiness or anger) or other categories such as positive or negative emotions using EMFACS or similar system. Another example is when a face appears with a brow furrow which can be judged as “anger” in the message-judgment approach while as a facial movement that lowers (AU4) and pulls the eyebrows closer together (AU7) in the sign-judgment approach.

In this thesis, we mainly work with the message-judgment approach for categorizing FEs into discrete affective labels as will be demonstrated in Chapter 3. Nonetheless, the sign-judgment approach is incorporated with the message-judgment approach in Chapter 4 for solving domain adaptation and to allow knowledge transfer to new domains and tasks.

## 1.2.2 Emotion Categories: Basic versus Non-Basic Emotions

There are many schemes of thinking about emotions, ranging from universal categories such as Ekman *basic emotions* to the huge diversity of social affective states such as *non-basic emotions*. Basic emotions include *anger*, *disgust*, *fear*, *happiness*, *sadness*, *surprise*, plus *neutral*, as shown in Figure 1.4. Non-basic emotions include *scared*, *anxious*, *hesitant*, *thinking*, *sus-*

*picious, affected, ashamed, astonished, curious etc.* Some of these mental states are shown in Figure 1.5.

The term *basic* means that emotions are universal and have evolved to help us adapt to our surroundings [Ekman and Cordaro 2011]. Everyone is producing such FE in the same consistent way. For instance the AUs used to produce an angry expression as listed in Table 1.1 are basically the same across populations. Basic emotion theory [Ekman and Cordaro 2011] captures what is unique about emotion, and what emotions have in common that distinguish them from other affective states. Some of the characteristics found in most basic emotions are:

1. Distinctive universal signals.
2. Distinctive physiology.
3. Automatic appraisal.
4. Capable of quick onset.
5. Can be of brief duration.
6. Unbidden occurrence.
7. Distinctive thoughts, memories, and images.
8. Distinctive subjective experience.
9. Target of emotion unconstrained.
10. The emotion can be enacted in either a constructive or destructive fashion.

On contrary, non-Basic emotions portray a wide variety of mental activities and deception expressed by realistic FE. Table 1.2 presents superordinate emotional categories into which individual emotion states can be grouped. Emotion states have more subtle facial deformation than basic FEs. The way to express them can depend on cultural background. So they are not so well defined and described than basic FEs.

|                           |                            |            |               |              |             |          |          |
|---------------------------|----------------------------|------------|---------------|--------------|-------------|----------|----------|
| <b>Basic Emotions</b>     | <b>Universal:</b>          | Happiness  | Fear          | Anger        | Disgust     | Sadness  | Surprise |
|                           | <b>Motivational State:</b> | Reward     | Punishment    | Thirst       | Huger       | Pain     |          |
| <b>Non-Basic Emotions</b> | <b>Moods:</b>              | Depression | Anxiety       | Cheerfulness | Contentment | Worry    |          |
|                           | <b>Emotion System:</b>     | Seeking    | Panic         | Rage         | Fear        |          |          |
|                           | <b>Social Emotion:</b>     | Pride      | Embarrassment | Guilt        | Shame       | Jealousy |          |

Table 1.2: Emotions categorization [Adolphs 2002].



Figure 1.5: Non-basic emotional FEs samples. The images come from the DynEmo database [Tcherkassof et al. 2013].

Current automated FE analysis systems either attempt to identify basic AUs of muscular

facial activity using FACS, or only go for recognizing the set of basic emotions [Zhang and Tjondronegoro 2011, Zhi et al. 2011, Shan et al. 2009, Huang and Tai 2012, Majumder et al. 2014, Yi et al. 2014, Wang et al. 2015, Liliانا et al. 2018]. Certainly, basic expressions are universal, but they comprise only a small subset of the mental states that people can experience, and are arguably not the most frequently occurring in day-to-day interactions [Rozin and Cohen 2003].

Recently, few researchers have started exploring how to model and interpret the subtlety and complexity of non-basic emotional categories [Du et al. 2014a], [Hu et al. 2018] and [Barros and Wermter 2015]. For instance some works extends the list of prototypical expressions to include pain , fatigue [Byrnes and Sturton 2018] and engagement [Kosti et al. 2017]. Du et al. 2014b deals with facial expressions as mixtures of the basic emotions such as “happily surprise” or “sadly fear”. Dealing with non-basic expressions is a challenging problem due to subtle deformations, uncontrolled head motions, and a wide variety of subject expressiveness and appearances. For instance, Figure 1.5 shows that there are some FEs we may or may not immediately recognize and in fact may not be recognizable without context due to the subtlety and fine granularity of the facial deformations.

In this dissertation, we study basic emotions and we also go beyond them to recognize complex non-basic emotions as they are closer to human natural behaviours.

### 1.2.3 Facial Expression Spontaneity Variations: Posed versus Spontaneous

Facial expressions are broadly classified as either deliberate (posed) or spontaneous, as shown in Figure 1.4 top row and bottom row respectively. Spontaneous facial expressions differ from posed expressions in both which muscles are moved, and in the dynamics of the movement. For instance, spontaneous FEs are produced in an involuntary-like manner leading to subtle or intense changes in a face while reflecting the true facial behavior rather than the exaggerated changes that mirror deliberate FE. For example, Figure 1.4 top row shows exaggerated deformations in the facial features which are distinctive among different emotional classes. It is easy to label and collect many samples of such data. On the contrary, Figure 1.4 bottom row shows spontaneous emotions with subtle facial feature deformations. For instance, considering the faces performing happiness and surprise emotions spontaneously as shown in Figure 1.4 bottom row, it is hard to distinguish between them and they look closer to each other, which is not the as when they are performed deliberately as shown in Figure 1.4 upper row.

Anatomically, Rinn 1984 shows that deliberate and spontaneous facial behaviors are mediated by separate motor pathways, the pyramidal and extra-pyramidal motor tracks, respectively. Correspondingly, fine-motor control of deliberate facial actions is often inferior and less symmetrical than what occurs spontaneously [Tian et al. 2005]. For instance, many people are able to raise their outer brows spontaneously while leaving their inner brows at rest, while few can perform this action voluntarily. As most of the people cannot perform some particular AUs deliberately, such as depression of the lip corners (AU15 as shown in Figure 1.2b) or raising and narrowing the inner corners of the brow (the combination of AU1+4), building a

model based on spontaneous expressions is crucial to detect true emotions and to build a real world affective applications such as lie detection [Ekman 1997] for example.

Moreover, Craig et al. 1991 insisted on the limitation of using subjects pose states for building FER systems, as there is a great deal of evidence that people do different things with their faces when posing versus during a spontaneous experience. For example, Ekman 1997 demonstrated an example on the difference between the posed and spontaneous smile. He shows the main differences between social and true smiles with respect to the activated facial muscles movements. Figure 1.6 shows that posed or social smiles only involve movement of the mouth and mainly the zygomaticus muscle (shown in Figure 1.3) because this muscle can be controlled. Whereas, spontaneous smiles (a.k.a Duchenne smiles) are more symmetrical, they are characterized by activating the risorius muscle (shown in Figure 1.3) that pulls mouth corners laterally and forms dimples in the cheeks. Risorius muscle is not always active in all people and it is hard to fake. In addition, spontaneous smiles include the orbicularis oculi muscle, AU6, which is not activated in posed smile. Similar study by Hess and Kleck 1990 have examined spontaneous versus deliberate facial expression of happiness and disgust in two experiments. Analyses of facial behavior in both experiments suggested that a deliberate expression shows more irregularities in timing than a spontaneous expression of the same emotion. Sauter and Fischer 2018 shows that acted expressions patterns are not able to generalize to real-life situations involving spontaneous emotional expressions.

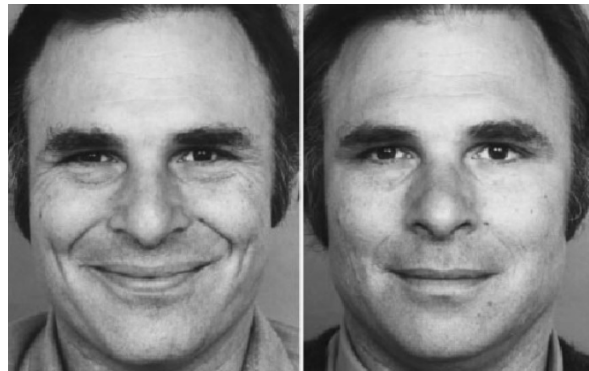


Figure 1.6: The Duchenne smile (left) versus the social smile (right). Social smiles use only the mouth muscles. Whereas true smiles, known as Duchenne smiles, cause the eyes to twinkle and the cheeks to rise.

Despite the fact that posed facial expressions are not natural and they have different characteristics than spontaneous ones, the vast majority of researches into FER works on acted FE materials due to the possible experimental control. The data collected in those studies are typically done by querying subjects to behave a series of basic expressions and are acquired under controlled lab conditions, *i.e.*, small set of posed facial expressions, short pre-segmented videos, portraits or nearly frontal views of faces, no facial hair or glasses, recorded under constant illumination, *etc.* FER is now considered solved for posed FEs; but is still an open question for realistic subtly exhibited expressions where, for example, even a slight tightening of the lips can be a sign that someone is angry. Subtle expressions are important for mental activity analysis and deception detection [Warren et al. 2009].

In this thesis, we put the focus on solving the problem of spontaneous facial expressions. However we also use acted facial expressions for experimental control and models validation. Furthermore, acted FEs materials are used for comparison with state of the art methods.

#### 1.2.4 Facial Expressions Types: Macro versus Micro

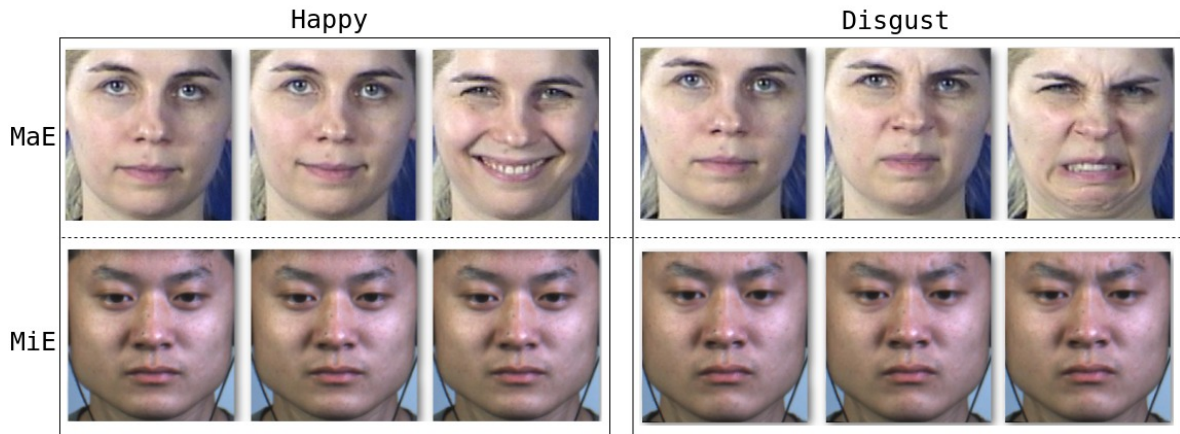


Figure 1.7: The first three columns represent an example of “happy” faces performed in MaEs manner (upper row, the MMI database), and in MiEs (bottom row, the SMIC database [Li et al. 2013]). The final three columns represent another example with “disgust” emotion.

Ekman 1997 proposed to consider two groups of facial expressions: *Macro-Facial Expression (MaE)s* and *Micro-Facial Expression (MiE)s*.

**Macro-Facial Expressions:** Occur when the subject agrees to express a given emotion. As a consequence, MaEs are characterized by lasting over a substantial period of time on several regions of the face. They are frequent and involve more conscious control. As shown in Figure 1.7 top row, they are easy to identify in videos because their facial motion deformations can be seen over the entire face. MaEs can occur spontaneously due to an involuntary manifestation of an emotional state or it can be posed as a result of a deliberate effort of communicating an emotional signal. Posed MaEs induce exaggerated movements and changes in the location and appearance of facial features. On the contrary, spontaneous MaEs are more subtle but have visible facial movements and typically evolve differently over time than posed MaE.

**Micro-Facial Expressions:** Occur when the expresser wants to repress his or her emotions (a.k.a poker face) in order to keep control of his or her face, due to a high stake situation where showing emotions is risky, as observed by Ekman 2009 and Warren et al. 2009. This phenomenon leads to a leakage of involuntary micro facial movements which may last only up to one twenty-fifth of a second as noticed by Shreve et al. 2009 and Porter and Ten Brinke 2008. MiEs rarely cause motion except on particular facial regions with low magnitude, depending on the induced emotion and can commonly go unnoticed. Figure 1.7 (bottom row)

shows MiE sequences of happiness and disgust emotions. It can be noticed that very subtle facial deformations are barely visible. Only when motion is incorporated, it can be perceived.

Three main characteristics differentiate macro- from micro-expressions, mainly: 1) the variations in the duration, 2) the spatial locality of facial deformations, and 3) the subtlety of facial movements.

In this dissertation, first, we dedicate our work on MaEs analysis where expressions are categorized into basic and non basic emotions and we put the focus on spontaneous facial expression associated with less constrained environmental conditions. Second, we study the problem of micro-facial expression detection as it is the primary step for a MaEs recognition system. Herein, we do not attempt to categorize MiEs into a discrete set of emotions. We leave inference about the conveyed message to psychological trained experts as categorizing MiEs is very hard and ambiguous.

### 1.2.5 Source of Input for Facial Expressions: Still Images versus Image Sequences

The salient issue in emotion recognition from faces, is the attempt to define the facial expression of emotion in term of qualitative targets, *i.e.*, static position capable of being displayed in static image or image sequence [Cowie et al. 2001].

Naturally, emotions are made of different parts: they start with a neutral phase, followed by an onset, then a peak or apex, and finally an offset phase. They have a finite duration. As shown in Figure 1.8, the neutral phase is the expressionless phase with no indication of facial muscular activity. The Onset phase expresses the duration upon which muscular contraction starts and inflate in intensity. The Apex or peak state is a plateau where the intensity reaches a stable maximum level of muscular contractions. The Offset phase is the muscular action relaxation.

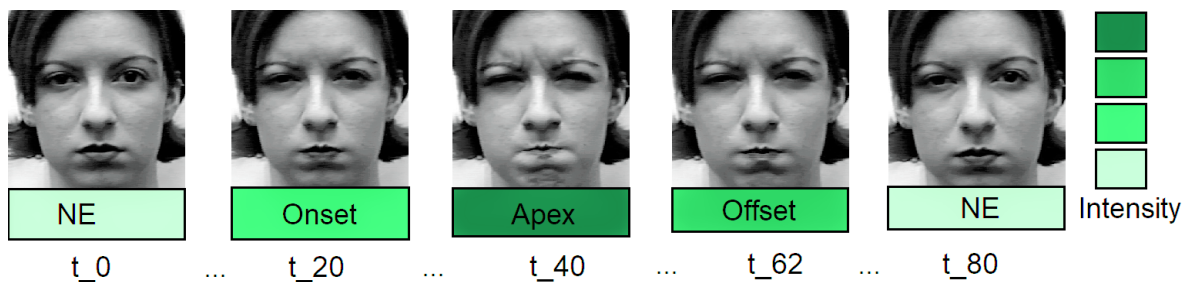


Figure 1.8: Morphological steps of an angry facial expression. Images taken from CK+ database [Lucey et al. 2010]. The time is denoted as  $t$  and the neutral phase is denoted as NE. Dark green is related to the strongest intensity level.

Most of the studies on FER have used static faces [Littlewort et al. 2011, Iqbal et al. 2018, Sajjad et al. 2018, Littlewort et al. 2011, Zafeiriou and Petrou 2009, Munir and Arshid 2018, Dhall et al. 2011, and Zhi et al. 2011] as stimuli of prototypical expressive patterns, considering



the apex of the expression. However, other studies have demonstrated that dynamic stimuli represented in an image sequence [Gu et al. 2017, Liu et al. 2014d, Wang et al. 2013, Hasani and Mahoor 2017, Demirkus et al. 2016, Yau et al. 2015, and Kuo and Sarkis 2018], usually capture the morphological changes of an emotion, that is its unfolding and ending as shown in Figure 1.8. Neglecting the dynamic aspect of a facial expression influences the perception of the facial expression. And taking dynamic into account is even mandatory for MiEs recognition.

In this dissertation, we consider first static images using the most expressive frames and then we extend our methods to spatio-temporal information in which image sequences are used.

### 1.2.6 FER Challenges: Intrinsic and Extrinsic Variabilities

Although there is a vast literature on FER, the design of an automated system is still considered as a complex and challenging problem firstly because of the different kinds of emotion categories and types and also due to the spontaneity of FEs as mentioned earlier. Secondly, facial images can also be affected by different variations such as lighting conditions, face pose, face alignment, and occlusions. Furthermore, it is affected by different appearance changes in face shape, texture, color and hair vary with gender, ethnic background, and age [Tian et al. 2005]. An important requirement for an automating FER is to achieve an optimal preprocessing for feature representation or selection and an accurate classification particularly under various input data conditions ranging from subjects and environmental variability to image acquisition. The sources of variation in facial appearance can be categorized into two groups: Intrinsic and Extrinsic factors.

**Intrinsic factors:** are related to appearance and expressiveness variabilities due to the physical nature of the face. For instance, intra-personal facial changes can occur from one moment to the next, such as eye gaze direction, head pose, or wearing eyeglasses or jewelry that may hide prominent facial features. Beyond individual variations in appearance, there are inter-personal variabilities in expressiveness intensity with respect to the amplitudes of facial AUs. It is often difficult to determine the absolute emotional intensity of a given subject and this leadings to ambiguity to determine the proper emotion. An extreme example of variability in expressiveness occurs in individuals who have incurred damages either to the facial nerve or to the central nervous system [Rinn 1984].

**Extrinsic factors:** these factors cause changes in the appearance of the face due to the interaction of light with the face or due to the data acquisition conditions such as scale and imaging parameters, *e.g.*, resolution, focus, imaging, noise, *etc.*

1. **Illumination.** Illumination changes can vary the overall magnitude of light intensity reflected back from an object (face), in addition to the pattern of shading and shadows visible in an image. Certainly, varying the illumination yields in larger facial image changes than varying either the identity or the viewpoint of a face. For instance, the difference between two images of the same identity taken under varying illuminations is greater than the difference between the images of two different people under the same

illumination.

2. **Pose.** The pose of a face varies with the viewing angle of the observer and the rotation in the head position. These changes impact the identification of the input image. For instance when the rotation angle goes higher and the available images in the database consist only of frontal views of the face, it becomes a challenge to recognize facial expressions.
3. **Occlusion.** It is produced by presence of various occluding objects such as glasses, beard, moustache *etc.* In real world applications, it is a common situation to acquire people talking on the phone or wearing glasses, scarves, hats, *etc.*, or for some reasons having their face covered with hands. Occlusions can severely affect the feature extraction process of the FER, thus leading to inferior classification performances.
4. **Image Quality.** When the face is far from the camera, the facial parts become very small or blurry, resulting a low resolution facial image. Such an image consists of limited feature information as most of the informative details are lost. This can drop down the recognition rate severely and impose difficulty to detect the face.
5. **Ground truth reliability.** During training process of FER, we assume that the training and test data are accurately labeled, which shall be more or less accurate. Querying subjects to behave a given facial expression does not guarantee that they will. To secure the ground truth validity, facial expression data must be manually coded, and to secure the reliability of the coding, it must be verified by trained psychological experts [Tian et al. 2005].
6. **Databases conditions.** Limited FEs datasets consisting of rather very basic facial expressions (*e.g.*, joy or anger) than non-basic expressions (*e.g.*, worried or ashamed) have been relatively used in facial expression analysis research. Subjects have been few in number and homogeneous with respect to age and ethnic background so that the recording conditions have been optimized. As this methodological setup do not reflect real world conditions, it reduces the capability of any model to generalize well over unseen faces or even over seen faces under intrinsic or extrinsic factors changes. In spite of this fact, those databases are still used to give insight on understanding facial expression analysis and behaviors.

Any FER system is typically challenged by these factors that affect its performances. To address all these dimensions and output an accurate recognition result, on the one hand, it is important to collect large databases of adverse intrinsic and extrinsic factors. On the other hand, further research into techniques that extract and represent efficient FE features robust against different intrinsic and extrinsic variabilities of the input data is needed. Finally, it would be also ideal to have approaches that are efficient to transfer knowledge learned from the current available FE materials to scenarios in which expressions, subjects, contexts, or image properties are wilder, and to periodically or continuously adapting their knowledge to avoid models from scratch retraining. A summary of optimal properties of FE analysis system is presented in Table 1.3.

In this thesis, we target these factors using two complementary directions. First, we investigate how to build facial representations and feature extractors that can alleviate the effects of the discussed factors. Then, we investigate how to incorporate the best feature represen-

|                           |  |   |                                    |   |  |  |
|---------------------------|--|---|------------------------------------|---|--|--|
| <b>Robustness:</b>        | Deal with subjects of different ages, genders, and ethnicities | Handle illumination, head motion, and occlusions            | Handle identity bias               | Handle distorted data and low image quality | Recognize a wide range of basic and non-basic emotions | Recognize spontaneous expressions with various intensities |
| <b>Automatic process:</b> | Face acquisition and detection                                 | Facial feature extraction                                   | Expression Recognition             | Analysis of motion and features             |  |  |
| <b>Real-time process:</b> | Acquisition and detection                                      | Feature extraction  | Recognition                        |   |  |  |
| <b>Autonomic process:</b> | Output recognition with confidence                             | Adaptative to different level outputs based on input images | Adaptive to new data distributions | Adaptive to new label distributions         |  |  |

Table 1.3: Properties of an ideal facial expression analysis system.

tation while solving the domain shift problem so that we can allow transferring knowledge using current available FE materials to challenging scenarios such as applications in which expressions, subjects, contexts, or image properties are more variable.

## 1.3 Current FER Approaches

### 1.3.1 FER Approaches via Message-Judgment

To recognize someone’s facial behavioural cues, the majority of efforts in Affective Computing (AC) concerns automatic analysis of facial expressions based on a message-judgment approach due to the ease of interpretation of facial expressions. These methods follow either a Conventional-Based FER approach or a Deep Learning-Based FER approach.

**Conventional FER Approaches.** Various conventional automatic FER approaches have been studied over decades. These approaches implement a decoupled process for feature extraction and for classification, where each stage is done separately. Feature extraction and representation portray the core success of such approaches as it has to deal with the various intrinsic and extrinsic factors that affect the recognition performances. The commonality of these approaches is to detect the face region and to extract features whether geometric or/and appearance-based.

Geometric features study the relationships between facial components such as eyes, nose, mouth, *etc.*, to construct a feature vector for training. The appearance features are extracted from the global face region or different facial regions by utilizing handcrafted filters such as Gabor filters [Janu et al. 2017] or descriptors such as the Local Binary Pattern (LBP) [Zhao and Pietikainen 2007] histograms, the Scale-Invariant Feature Transform (SIFT) [Lowe 2004], the Histogram of Oriented Gradient (HOG) [Dalal and Triggs 2005] or the Speeded-Up Robust Features (SURF) [Bay et al. 2008]. These descriptors are designed typically to encode the existence of various low-level features such as corners, color schemes, texture of facial image, *etc.*

Apart from spatial FE analysis of 2D images, dynamic FER utilizes image sequences for analysis of FEs where motion and spatial information are extracted to get the final representation. Geometric spatio-temporal features typically include tracking facial points while

appearance spatio-temporal features include spatio-temporal descriptors such as Local Binary Pattern Three Orthogonal Planes (LBP-TOP) [Wang et al. 2015] or 3 Dimensional Scale-Invariant Feature Transform (3D-SIFT) [Scovanner et al. 2007].

Finally, the extracted feature vectors, whether spatial or spatio-temporal, are fed to train traditional classifiers such as Support Vector Machines (SVM), Random Forest (RF) and k-Nearest Neighbours (k-NN), which are frequently reported in many studies [Jang et al. 2014, Kolodziej et al. 2018, Mostafa et al. 2018, Wang et al. 2018, Ramirez Cornejo and Pedrini 2018]. As it can be noticed, conventional FER approaches are characterized by processing first the facial features and then using them for inference. As those two stages are done separately, a sub-optimal performance is achieved especially on the most challenging datasets where lots of variability sources are mixed. It is more favourable to jointly learn those two stages for better optimal recognition performances.

**Deep Learning-Based FER Approaches.** Deep Neural Networks (DNN)s have been used for learning discriminative feature representations for automatic FER [Tran et al. 2017], by designing a hierarchical architecture composed of multiple nonlinear transformations based on Convolutional Neural Network (CNN) and Recurrent Neural Network (RNN). Because CNN-based methods cannot reflect temporal variations in the facial components, recent hybrid approach combined a CNN for the spatial features of individual frames and a RNN for finding temporal correlations and thus extending the facial representation into a spatio-temporal one. Typically for classification task, these networks are coupled with a logistic classifier layer for classification. Different from conventional approaches, the feature extraction stage and the classification stage are jointly optimized. This is done by enabling an “end-to-end” learning from raw images while coupling the learning parameters of a classifier jointly alongside feature representation parameters [Walecki et al. 2017]. By automating the process of feature extraction and classification directly from the raw data, the dependence on physics-based models and other pre-processing techniques is highly reduced. Li and Deng 2018 provided a survey on the effect of coupling or decoupling the process of feature extraction and classification for automatic FER. The reader is referred to [Ko 2018] for a review of FER based on visual information.

### 1.3.2 FER Approaches via Sign-Judgment

Emotion recognition methods using the sign-judgment approach are a two-phase based method. Since AUs are considered as building blocks of FEs, first the facial AUs are detected based on FACS and then the output emotion is inferred from the detected AUs based on EMFACS. There are few deterministic rules based techniques which map the computed facial AUs into emotion categories [Moon et al. 2013]. Valstar and Pantic 2006 investigates and compares between considering the underlying emotions from FEs using their feature deformation against a two-step rule-based approach using FACS and EMFACS. Valstar and Pantic 2006 use DNNs to find different possible combinations of AUs for predicting the emotions. Velusamy et al. 2011 derives a template string of AUs and develops an approximate string matching approach to map AUs onto emotions. Tong et al. 2010 models the relationships between AUs

and local facial component shapes using a unified probabilistic facial action model based on the Dynamic Bayesian Network (DBN), to simultaneously and coherently represent rigid and nonrigid facial motions, their spatiotemporal dependencies, and their image measurements. Kim and Kim 2019 presents a CNN model that classifies emotions based on image features, where the last layer is used to predict the possible activated AUs. Kim and Kim 2019 model generates the AUs with the assumption that features and emotion labels are the product of CNN's decision-making.

### 1.3.3 Discussion

Our work mainly deals with classification of FEs into a discrete set of emotional labels. We evaluate both conventional FER approach and deep learning based methods. The major work of this thesis is based on deriving a robust representation and on learning a classifier using recent advances in deep learning. In this thesis, we are not concerned in sign-judgment approach for classifying FEs into emotions. However, we use AUs as visual facial attributes for constructing a bridge for knowledge transfer and domain adaptation.

## 1.4 Thesis Contributions

In this dissertation, we study the problem of Facial Expression Analysis aiming at refining the classification and detection performances while dealing with less constrained environments *i.e.*, a set of non-basic spontaneous facial expressions, short and long videos sequences, portraits and profile views of faces, with facial hair or glasses, recorded with head motion, *etc.* and not just laboratory conditions. We put the main focus on studying spontaneous facial expressions rather than acted ones as they are closer to human natural facial behaviours. Nonetheless acted facial expression databases are used mainly for the experimental control and for comparative reason with state of the art methods. We design approaches that go beyond traditional facial expressions recognition methods to solve the problem of domain adaptation and to recognize new unseen emotions. We have divided our study into three major parts:

**Image and Video Based Macro-Facial Expression Recognition** using conventional based methods and deep learning based methods. We put the attention primarily on the influence of feature extraction and representation, including low-level, mid-level, and hierarchical representations and we study their impacts on the recognition rates. For low-level feature extraction we propose an improved version of the Bag of Visual Words model where we optimize the spatial arrangement and vector quantization process and where we speed up the learning process. For mid-level feature representation we propose a sparse representation method that utilizes dictionary learning for combining low-level features and for representing them in a higher sparse dimensional space where features among different classes might be linearly separable. We tackle the main problem of building discriminative dictionaries for facial expression recognition. Lastly, we propose a hierarchical representation (spatial and spatio-temporal) using a deep learning based method. We used Convolutional Neural Networks and

Convolutional Long Short Term Memory alongside different modules such as Spatial Pyramid Pooling Networks and 3D-Convolutional Neural Networks to extract local and global spatio-temporal features that are invariant to scale, rotation and occlusion and that increase the robustness against identity bias and various challenging factors while learning a multi-class logistic classifier.

**Adapting Facial Expression Models to New Domains or Tasks.** In a conventional Facial Expression Recognition framework, only the classes appearing in the training data can be recognized by the model during the inference phase. These approaches cannot tackle challenging scenarios such as when new emotional classes appear after the learning stage, resulting faulty categorization. Such scenarios happen when it is not feasible to obtain training examples for an emotional category to be distinguished such as for mental state classes because of its feature subtlety and fine granularity. Moreover, another scenario also happens when someone tries to generalize a trained algorithm over new subjects or contexts that might only work if and only if a very large database that covers all the intrinsic and extrinsic factors we discussed earlier are used to train that algorithm. Such a case is impossible to realize as it is very expensive and very hard to annotate. But, we could benefit from the available materials for adaptation to new domain or dataset without the need of retraining the model from scratch. By that, we might reduce the high cost of annotating new target domain data. Therefore, to cope with this task and solve the aforementioned scenarios, we investigate how to enforce domain-invariance to alleviate the domain shift problem and we propose a Domain Adaptation algorithm for emotion recognition. We also study how to transfer knowledge from a source domain data made of posed and spontaneous universal basic Ekman emotion classes to new target domain data corresponding to unseen mental state classes by proposing a Zero Shot Learning algorithm for emotion recognition. Our algorithms benefit from a common semantic space learned using natural language models. All visual features are mapped into this semantic space based on Euclidean learning. We propose a novel alignment method between the two spaces to guarantee a proper mapping.

**Micro-Facial Expression Detection.** We propose a spatio-temporal algorithm for spotting micro-expression segments including the onset and offset frames and to spatially localize in each frame the regions involved in the micro-facial muscle movements. Micro-expressions spotting is a challenging problem as a consequence of the locality and subtlety of facial muscle movements over a brief duration and due to their fast motion. Another challenge come from the fact that micro-expressions tend to be rare and since they are recorded with high-speed cameras, some parasitic movements such as eye gaze and blinking are captured and confused with micro-expressions. Therefore, to achieve our objective and to overcome the main challenges enforced by the nature of micro-expressions, we reformulate the problem into Anomaly Detection. We propose a deep Recurrent Convolutional Auto-Encoder to capture spatial and motion feature changes of natural facial behaviours such as neutral face, talking, moving, blinking, *etc.* Then, a statistical model based on a Gaussian mixture model is learned for estimating the probability density function of normal facial behaviours while associating a discriminating score to spot micro-expressions. Finally, an adaptive thresholding technique for identifying micro expressions from natural facial behaviours is proposed.

## 1.5 Thesis Outline

After this general introduction, the rest of the thesis consists of:

- (a) In Chapter 2 we discuss the state of the art methods which deals with Facial Expression Recognition in order to put our choices and their different components in context.
- (b) In Chapter 3 we describe our appearance-based methods for Macro-Facial Expression Recognition. They are based on different levels of feature representation and they work either with static images or dynamic image sequences.
- (c) In Chapter 4 we present a deep learning model for transferring knowledge where the aim is to alleviate the problem of domain shift and to promote the recognition of unseen classes at inference stage.
- (d) In Chapter 5 we develop a probabilistic model for Micro-Facial Expression Detection in space and time and a spatiotemporal feature extractor based on recurrent convolutional auto-encoder.
- (e) Chapter 6 summarizes the main contributions, states the main limitations of the proposed approaches and opens some perspectives and future directions.

# A Review of Facial Expressions: Registration, Representation, and Recognition

---

Analysis of facial behaviours has aroused considerable interest for building systems capable of automatically detecting and recognizing basic and non-basic emotions whether coming from macro or micro-facial expressions using images or videos as input signals. The line of research on facial expression analysis in AC community puts the focus on diverse questions and in particular on what are the important clues to describe facial expressions and how to represent them in an efficient and effective way to improve the success of the recognition or detection system. Computer vision and machine learning manipulate how to represent this information while psychologists focus on describing the important facial clues and their direct links with emotions. In this chapter, we shed light on how to represent facial expression information. First we impart FER systems into their fundamental components and we analyse the state of the art solutions to describe the role and the characteristics of each component. We base our investigation on the most relevant and recent surveys [Corneanu et al. 2016], [Sandbach et al. 2012], [Salah et al. 2011], [Pantic 2009], [Pantic and Rothkrantz 2000] and in particular on [Sariyanidi et al. 2015] who targets the transition from controlled to naturalistic ambiances.

## 2.1 Facial Expression Recognition Systems

An automatic facial analysis system from images or video can be decomposed into five parts: face detection, facial registration, face representation, facial feature extraction and FE recognition, as it is depicted in Figure 2.1. While the target of inference depends on the adopted facial expression theory (message-judgment or sign-judgment) as we discussed earlier in Chapter 1, the considerations regarding the FER pipeline are consistent to each of them, with only the inference layer being specific. FER systems proposed in the literature deal with each component differently in order to address the main challenges that are associated to head-pose variations, illumination variations, registration errors, occlusions and identity bias. Spontaneous behaviour generally comprises head-pose variations, which need to be taken into consideration before measuring FEs. Occluded faces due to head or camera movements, or accessories such as scarves or sunglasses or having large pose variations lead to failure in the face detection. Illumination contrasts can be tricky even under constant illumination due to



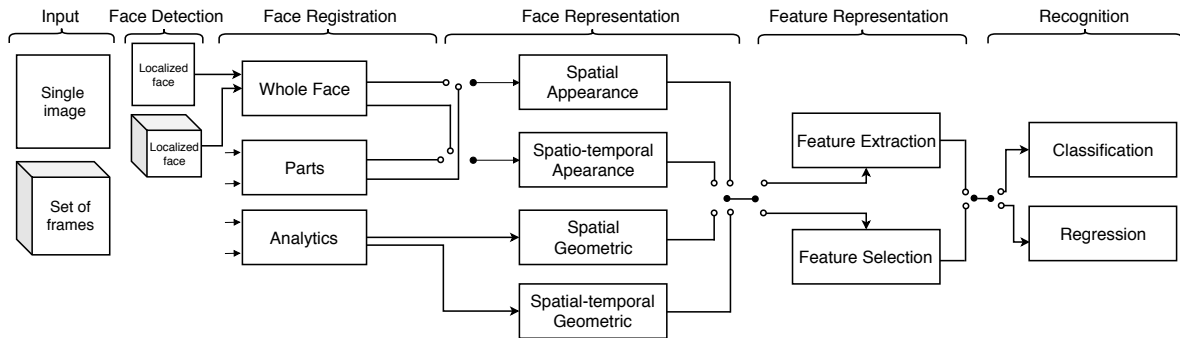


Figure 2.1: Generic pipeline for FER algorithms. The input is a peak facial expression image for spatial representations or a set of facial frames within a temporal window for spatio-temporal representations. The system outputs a discrete label if it is obtained through classification or a continuous signal if obtained through regression.

head movements. Registration techniques lead to registration errors, which demand to be taken into account to establish the applicability of the representation features. Dealing with identity bias requires the ability to tell identity-related texture and shape cues apart from expression-related cues for subject independent emotion recognition [Sariyanidi et al. 2015]. Furthermore, under the assumption that we are able to withstand all these challenges, the features of a representation shall also empower the detection of subtle facial displacements. FER components are responsible for targeting these roles such as aligning and normalising the visual information contained in the face after detecting it, eliminating irrelevant variability in the face which comes from misalignment, alleviating the effects of head pose variations and identity while capturing relevant features dedicated to emotion recognition task. In the following sections, we analyse and discuss how the functionality and of each component addresses the above-mentioned challenges.

## 2.2 Facial Detection

Face detection approaches aim at locating the faces and obtaining their bounding box or geometry. Current available facial expression databases are characterized by containing faces with near-frontal head poses, having a good image quality and resolution, and containing few partial occlusions. Viola and Jones 2001 algorithm in such conditions is sufficiently robust and accurate to detect faces. It is based on a cascade of weak classifiers and it is characterized by being fast and attaining high accuracy. Nonetheless, it suffers when the faces starts being occluded or having large pose variations. Some methods overcome these weaknesses by considering multiple pose-specific detectors. Jain et al. 2013 approaches the problem by proposing a robust CNN model. The reader is referred to [Zhang and Zhang 2010] for a complete review about face localization approaches. In our study, depending on the simplicity or the wildness of the database, we either use the Viola and Jones algorithm or the CNN-based method [Jain et al. 2013]. In some cases, we primarily use the Viola and Jones algorithm and back-up to

the CNN approach of Jain et al. in case of detection failure.

## 2.3 Face Registration

Face registration is an essential step to bring all faces to a common coordinate system, such as to rotate or frontalize sample faces to a reference face. The reference face is usually selected as a face with frontal pose and thus its selection becomes critical as it can make considerable changes in the performance of the registration algorithms. The reader is referred to [Wang et al. 2018] for a survey. Depending on the output of registration, studies [Pantic and Rothkrantz 2000, Sariyanidi et al. 2015] categorize face registration into three parts: holistic, part and analytic registration.

1. Holistic registration considers the global face coordinates as the Region of Interest (ROI) which can be rigid or non-rigid. Rigid holistic registration maps an input face to a prototypical face by detecting facial landmarks (visually fine-grained salient points in facial regions such as the end of the nose, ends of the eye brows, and the mouth) and uses their location to compute a global transformation (*e.g.*, Euclidean, Affine). The most used techniques are Active Appearance Model (AAM) [Cootes et al. 2001] and Lukas-Kanade approach [Baker and Matthews 2004]. Non-rigid holistic face registration has the capacity to map an expressive face into a neutral face by performing a piece-wise registration around each landmark point [Lucey et al. 2012].
2. Part registration considers prominent facial parts (*e.g.*, mouth, nose, eyes) localized as fixed-size patches, which may require the spatial consistency of each part and the number, size and location of the parts to be registered. Zhang and Tjondronegoro 2011 considers to perform part detection to localize each patch alone.
3. Analytic or Point registration models the face as a set of facial points. Such registration is needed for shape representations, for which registration involves the localization of landmark points. Valstar et al. 2010 proposes to register these landmarks points in a sequence using a point detector on the first frame and then tracked them.

In general, the type of facial features retrieved by a FER system is greatly dependent on the registration strategy employed. For instance, some representations are connected with a fixed type of registration, such as analytic face registration is defined in terms of the facial landmarks. Hence, the extracted features will represent the same part of the face for every example. On the one hand, such representations allow to build features robust to head pose rotations which can be approximated locally by affine transformations and illumination variations which are approximately locally homogeneous. On the other hand, they fail to capture the full face appearance as they neglect configural information completely, for example the cheeks do not contain landmarks. Differently, since holistic representations consider the full face, they grant to capture the full face appearance, but they lack some of the positive properties of local representations. Clearly, each representation type has different properties and levels of robustness against the different disturbance factors. Therefore, a crucial choice of the registration technique has to be made. Commonly, models validated on posed facial ex-

## **22A Review of Facial Expression: Registration, Representation, and Recognition**

pressions dataset employ holistic registration since the face is well aligned and the acquisition conditions are appropriately optimized for well controlled experiment. Nonetheless models validated on spontaneous facial expression dataset employ sophisticated holistic, parts or points registration techniques.

In this dissertation, our methods rely on considering the detected face as a whole entity. Our choice is motivated by taking their positive advantages in order to preserve the configural information. That is taking into account the subtle differences between facial components and their spatial and temporal links. But since we put the focus on improving the recognition performances on spontaneous facial expression datasets, we build sophisticated feature extraction methods to alleviate the sensitivity of the representation to intrinsic and extrinsic factors. To do so, we assist various feature representation levels mainly low-level, mid-level and high-level features. In each representation level, either salient points are located using salient detectors and described using descriptors or features are extracted in response to global or local learned filters.

### **2.4 Face Representation and Facial Feature Extraction**

Processing facial information and extracting relevant facial expression features entail choosing a facial representation, which is divided into two categories: spatial or spatio-temporal representations.

1. Spatial representations encode images at static level without involving dynamic morphological changes in a face and they consider only single images, usually the most expressive frame (the apex frame).
2. Spatio-temporal representations encode both spatial and temporal variations to describe how FE feature evolves in time. These representations helps to recognize subtle feature displacements, especially those coming from spontaneous facial behaviours.

Furthermore, depending on the type of patterns to be discovered in the image or sequence space, another classification scheme exists that classifies facial representation into appearance, shape or hybrid.

1. Appearance representations represent the texture of facial skin by considering the intensity values of the pixels, including wrinkles, bulges, and furrows. They include filters such as CNN, Independent Component Analysis (ICA), Principle Component Analysis (PCA), Gabor filters, *etc.*
2. Geometric representations are related to the shape of facial components as eyes, nose, mouth and to the locations of landmark points (corners of the eyes, the mouth, *etc.*). Geometric methods ignore texture information when computing the location of a landmark with respect to the neutral face or the distance between two landmarks.
3. Hybrid methods fuse both appearance and geometric representations.

Moreover, facial features based on facial representation, can be classified into engineered or

learned features.

1. Engineered features are hand-crafted. They are designed to represent the image by encoding minor details of the image, like lines, edges, or dots. They are the result of mathematical descriptors designed with certain properties in mind, for example the illumination invariance of LBP or the scale-invariance of SIFT.
2. Learned features are extracted by learning filters from a large dataset to address a specific task. DNNs and in particular CNNs are popular in this criteria.

In general, features can be extracted either locally or globally. Local features treat only specific parts of the face while global features consider the whole facial region. To this end, we can categorize the state of the art as shown in Table 2.1. A detailed explanation of these works is provided in Appendix 7.

| Engineered Features       |                      |                           |                         | Learned Features      |                        |
|---------------------------|----------------------|---------------------------|-------------------------|-----------------------|------------------------|
| Spatial                   |                      | Spatio-temporal           |                         | Spatial               | Spatio-Temporal        |
| Appearance                | Geometrical          | Appearance                | Geometrical             | Appearance            |                        |
| Iqbal et al. 2018         | Berretti et al. 2011 | Zhao and Pietikainen 2007 | Wu et al. 2013          | Yang et al. 2018      | Gu et al. 2017         |
| Sajjad et al. 2018        | Vretos et al. 2011   | Koelstra et al. 2010      | Kacem et al. 2017       | Zhang et al. 2018     | Liu et al. 2014d       |
| Littlewort et al. 2011    | Martinez et al. 2013 | Sun et al. 2014           | Dibeklioglu et al. 2013 | Zhao et al. 2016      | Wang et al. 2013       |
| Zafeiriou and Petrou 2009 | Kim et al. 2014      | Liu et al. 2014a          | Su et al. 2014          | Kaltwang et al. 2015  | Hasani and Mahoor 2017 |
| Munir and Arshid 2018     | Wang et al. 2014     | Shangfei WANG 2014        | Sandbach et al. 2011    | Zhao et al. 2015      | Demirkus et al. 2016   |
| Dhall et al. 2011         | Lu et al. 2017       | Liu and Yin 2015          | LE 2011                 | Liu et al. 2014e      | Yau et al. 2015        |
| Zhi et al. 2011           | Liliana et al. 2018  | Kamarol et al. 2016       | Ghimire and Lee 2013    | Zeng and Dobaie. 2018 | Kuo and Sarkis 2018    |

Table 2.1: Summary of facial representations and feature extraction for FER systems proposed in the literature.

## 2.4.1 Spatial Representations and Feature Encoding

There exist a sort of spatial representations related to shape and appearance as shown in Figure 2.1. In the following, we details those methods and define our choices.

### 2.4.1.1 Spatial Shape-Based Feature Representations

Facial point representation is among the most famous method among geometric shape representations. It expresses the face by concatenating the coordinates  $x$  and  $y$  of a pre-defined number of facial landmark points. Lucey et al. 2007 shows the capacity of reducing identity bias, which is achieved by subtracting the 74 facial points of expressive face from the neutral one. In this way, only the related facial expression remains and thus generalizing to any face. But a neutral face has to be available and registration errors are also plausible as it is based on coordinate values. As the intensity of the pixels is avoided, illumination variations are not an issue but they affect the registration certainty of the points. The dimensionality of this representation is relatively low. Alternative shape representations exist. For instance, Huang et al. 2010 uses the distances between facial landmarks as a substitute to raw coordinates. Tian et al. 2001 computes descriptors that describe the opening and closing of the eyes and mouth, and a collection of points that describe the state of the cheeks.

Recent studies tend to move from geometric based methods towards appearance based methods due to their robustness against various intrinsic and extrinsic factors. However, identity bias remains an outstanding issue for such low-level representation. Geometrical features tend to be useful only to describe obvious facial deformations. Yet, they fail to detect subtler characteristic of spontaneous facial expressions like texture changes.

### 2.4.1.2 Spatial Appearance-Based Feature Representations

In appearance-based criteria, low-level, mid-level and hierarchical features are among the most famous. Low-level information is typically encoded in a data-driven mode alongside a Bag-of-Visual-Words (BoVW) model by using descriptors. Mid-level information is encoded using sparse coding and dictionary learning. Hierarchical information consists of cascaded low and mid-level layers typically established using deep learning based methods.

These levels of information can also be applied on part-based faces which are looked at in term of independent parts, in the same way they are applied over the whole face. Such appearance-based part representation ignores the spatial relations among the registered parts and thus reducing the sensitivity to head-pose variations. For instance, Jeni et al. 2013 and Zhu et al. 2011 prove the usefulness of this representation for spontaneous emotion recognition where head-pose variations naturally occur.

I. **Encoding Low-Level Feature.** These approaches aim to explicitly exploit the visual information content of the FE image as far as possible. They mostly represent images by low-level feature primitives such as color, shape, and texture. For FER, color it is barely used. Shape is an important cue to identify and recognize objects, whose purpose is to encode simple geometrical forms such as straight lines in different directions. Texture is a very useful characterization for a wide range of image. Texture features can be extracted using various filters and descriptors such as Gabor filter [Turner 1986], SIFT [Lowe 1999], HOG [Dalal and Triggs 2005] or their combinations. Spatial textures are meaningful, easy to understand, and can be extracted from any shape without losing information but they are sensitive to noise and distortions. Histogram are simple to compute and intuitive but they have high dimension, and no spatial information is preserved. Moreover they are sensitive to noise. In the following, we explore the most frequent low-level feature representations.

(a) **Using Descriptors with BoVW.** This representation encodes an image as a set of features. Features consist of keypoints and descriptors. The keypoints are the salient points, no matter the image is rotated, shrunked, or expanded, its keypoints will always be the same. They can be located either densely over a grid or by sparsely locating salient point using detectors such as Difference of Gaussian (DoG). Descriptors tend to encode local features information rather than the face structure and provide features that are robust to affine transformations and invariant to global illumination changes to some degree. Local features can be described over full face regions or patch-based regions using common descriptors such as LBP [Ahonen et al. 2006], HOG [Dalal and Triggs 2005], SIFT [Lowe 1999] *etc.* The

keypoints and descriptors typically are used to construct visual vocabularies and represent each image as a frequency histogram of features that are in the image. To establish the final histogram, the obtained local features from the description of the keypoints or of each local patch region are firstly pooled to establish local histograms. Then, by concatenating all the local histograms, the image signature (final histogram) is obtained. Dalal and Triggs 2005 proposed to normalize the final histogram to unit-norm to improve the robustness of the overall representation. The histogram representations are tolerant against registration errors as they discard the location of adjacent features by performing the pooling operation. The pooling operation aims at reducing the dimensionality over local blocks by describing the features within the blocks jointly. As a drawback, Ahonen et al. 2006 shows that such representations are affected negatively by identity bias, as they favour identity-related cues rather than expressions. Shan et al. 2009 provided real-time operations using low-level histogram representations. This representation might reach very high dimensionality depending on the size of the visual vocabulary and therefore its generalization to spontaneous data requires further validation. The computation of visual words is based on a search over the visual vocabulary and depending on the vocabulary size and the search algorithm used, it can be computationally costly. In the following, we explore the most frequent low level feature descriptors.

- LBP descriptor describes local texture variations along a circular region and labels the pixels by thresholding the neighborhood of each pixel and considers the result as a binary number. It sets 1 for values equal or higher than the threshold and 0 for values lower than the threshold, where the threshold is the central value of the selected region. By that, the selected region matrix is converted from a gray scale level into a matrix containing binary values while ignoring the central value. Afterwards, each binary value from each position from the matrix is concatenated in clockwise or anti-clockwise direction leading into a binary string (*e.g.* 10001101). This binary string is then converted into an integer value and is set as central value of the matrix. The range of the most common LBP is set to gray scale  $[0, 255]$ . By that, a new image which represents the texture of the original image is obtained. To obtain the histogram from the new texture representation, a grid of  $x \times y$  parameters is defined to divide the image into multiple uniform regions. Then the histogram of each region is computed by representing the occurrences of each pixel intensity (for grayscale, each histogram will contain only 256 positions). The final histogram is a concatenation of each local histogram. Therefore the dimensionality of the representation depends on the range of integers. LBP uses three parameters. The *radius* which is used to build the circular local binary patterns and represents the radius around the central pixel. It is usually set to 1. The *neighbors* which is the number of sample points to build the circular local binary patterns. It is usually set to 8. The *grid*, in which the more cells, the finer the grid, the higher the dimensionality of the resulting feature vector. It is usually set to 8. LBP features are generally less sensitive to registration errors and have been used by the winner of the AVEC word-level challenge

[Savran et al. 2012].

- The HOG descriptor encodes images by the directions of the edges they contain. It extracts local features by applying gradient operators across the image and encoding their outputs in terms of gradient magnitude and angle. First, local magnitude-angle histograms are extracted from cells, and then these local histograms are combined across larger entities (blocks). The dimensionality increases when the blocks are overlapping. HOG was used by a prominent system in the FERA emotion challenge [Dahmane and Meunier 2011].
- A SIFT descriptor is computed from the gradient vector histograms of the pixels in the patch. For example, for a patch of size  $n \times n$ , with  $n = 4$ , there are eight possible gradient directions and therefore the total size of the SIFT descriptor is  $4 \times 4 \times 8 = 128$  elements. This descriptor is normalized to enhance invariance to changes in illumination and to ensure invariance to scale and rotation as well. These properties make the SIFT descriptor capable of providing a compact and powerful local representation of the range image and of the face surface. SIFT was used for emotion recognition in the wild [Liu et al. 2014b].

(b) **Using Gabor Filters.** Gabor filters can be applied both in holistic and part representations. A Gabor representation can be computed by convolving the input image with a set of Gabor filters of different scales and orientations [Vukadinovic and Pantic 2005]. Such filters have the capability to capture local structures, bulges and wrinkles which are typical of facial muscle activations. Kamarainen et al. 2006 and Wiskott et al. 1997 show that such representations are tolerant to illumination variations to a degree because such filters are localized in space. While Gritti et al. 2008 and Lades et al. 1993 show that the obtained representation is robust to registration errors because the filters are smooth and the magnitude of filtered images is robust to small translations and rotations which increase the robustness to misalignment. However, their major drawback lies in the high dimensionality of the convolutional output, in which a dimensionality reduction step such as using PCA is required. Moreover, another challenge is posed on finding the right parametrisation of the Gabor filters. And more importantly, Wiskott et al. 1997 shows that the outputs of this representation suffer from identity bias because it favours identity-related cues rather than expressions ones. Plus it is a computationally expensive method.

In this thesis, we propose to extract low level features by considering SIFT, HOG and we explore their combinations alongside of the standard BoVW [Sivic and Zisserman 2003] and the Spatial Pyramid Matching which performs histogram pooling and encodes componential information at various scales [Lazebnik et al. 2006a]. Upon that, we build an improved version of standard BoVW model for low level facial image description as described in Section 3.2. We accelerate its performances and increase the recognition rates while tackling the main weaknesses of such representation.

II. **Encoding Mid-level Feature.** Mid-level features can be seen as a weighted combinations of low-level features and they provide a bridge between low-level based information

and high level concepts, such as object and image level information. Beside carrying the same information as the initial low-level features, they carry additional contextual cues which make them visually discriminative and more representative, that is, they can be detected in a large number of images. Hence, mid-level representations abstract low-level feature information that is useful for later classification while being robust to irrelevant and noisy features. They basically encode features that are semantically interpretable from an affect recognition perspective rather than encoding the distribution of edges and so far describing local texture as low-level features. Many successful models for category-level image classification transform low-level feature representations into richer mid-level representations of intermediate complexity that are better adapted to the task at hand because they have desired discriminative properties to provide robustness against inter and intra-class variabilities and deformations to a certain degree. Among various methods, two methods that produce such representations are Non-negative Matrix Factorization (NMF) such as the one proposed by Nikitidis et al. 2012 and Zhi et al. 2011 and sparse coding such as the one proposed by Mahoor et al. 2011 and Zafeiriou and Petrou 2010.

- (a) **Non-Negative Matrix Factorization.** The NMF technique is a widely used tool for the analysis of high dimensional data as it automatically extracts sparse and meaningful features from a set of non negative data vectors. The NMF technique build a number of basis images and the coefficients of each basis image are the feature vectors. The method executes a minimization process to calculate the coefficients. Depending on the optimization algorithm and the number and size of the basis images, its computational complexity varies. NMF relies on training data and therefore it is tolerant to some illumination variations and registration errors as long as the data variability covers such changes [Tunç et al. 2012]. NMF-based representations have the capacity to deal with identity bias by learning identity-free basis images. This also depends on the number of identities provided during training as well as the capability of the technique to deal with the inter-personal variations. The dimensionality of this representation depends on the size of the basis images, for instance, less than 100 features are used in Zhi et al. 2011 while 200 in Nikitidis et al. 2012. Part-based NMF representation also exists, which describes facial parts by means of a sparsity-enforced NMF decomposition. This representation removes person-specific texture details from each patch before the computation of NMF to reduce identity bias and it places higher emphasis on facial activity. However, texture subtraction may be susceptible to illumination variation sand registration errors. As the representation is based on NMF, its sensitivity against these issues also depends on the training process. The NMF guarantees low dimensionality, which is one of the main motivation to use such methods.
- (b) **Using Sparse Representation.** Sparse coding considers that any image is sparse in some domain and provides a transformation where most coefficients are zero while very few entries are non-zeros [Candes and Wakin 2008]. The transformation can be adaptive using learned dictionary (*e.g.* data-driven) or non-adaptive using fixed dictionary (*e.g.* Fourier transform). The flexibility in building the dictionary provides the freedom to create dictionaries whose basis vectors are semantically



interpretable (*e.g.* pre-defined vector with label information). In FER, researchers defined dictionaries where each dictionary vector corresponds to an AUs [Mahoor et al. 2011] or to a basic emotion [Cotter 2010]. The final representation is formed by concatenating the coefficients of dictionary elements. In an AU dictionary, the coefficients with the maximal value would point to the AU displayed in the input facial image. This representation can be designed to be robust against partial occlusions as has been done by Cotter 2010 and Zafeiriou and Petrou 2010. In general, the sparse coefficients are computed by solving  $\ell_0$  or  $\ell_1$  minimization. Therefore, the computational complexity depends on the optimisation algorithm and the size of the dictionary and its basis vectors.

In this thesis, we propose a mid-level discriminative sparse features based on sparse coding and dictionary learning (Section 3.3). We define the mid-level features associated with a facial image as the sparse coefficients of its linear combination of low-level basis vectors in a learned dictionary. We tackle the main difficulties for constructing informative and discriminative dictionaries for FER. We evaluate the capacity of this learning scheme to overcome the identity bias, intra and inter personal variations and other factors. Afterwards, we seek to establish the relative importance of mid-level representation compared to low-level one by conducting a comprehensive experimental evaluation.

**III. Encoding Hierarchical Feature.** Hierarchical features are learned by an algorithm fed with raw data in order to automatically discover the right features needed for detection or classification. Deep learning based methods are hierarchical learning methods with multiple levels of representation. They are obtained by composing non-linear modules where each module transforms the representation at one level starting with the raw input data into a representation at a higher abstract level [LeCun et al. 2015]. Therefore, it first extracts local features that capture geometric structures from small neighborhoods and further grouped into larger units to produce higher level features. With enough composition of such transformations, complex functions can be learned thus permitting the network to handle complicated scenarios that may be difficult to capture through hand-crafted objective functions [Bengio 2009]. For instance, the learned features in the first layer typically capture the presence or absence of edges at particular orientations and locations in the image same as in low-level feature representations. The second layer typically detects patterns by spotting particular arrangements of edges, regardless of small variations in the edge positions same as in mid-level feature representations. The third layer may assemble patterns into larger combinations that correspond to parts of familiar objects, and subsequent layers would detect objects as combinations of these parts which leads to high-level features. Hence, features from higher levels of the hierarchy are usually formed by the composition of lower-level features, thus a reliable feature representation can be established in a feed-forward manner from the same low level features. Higher layers of representation amplify aspects of the input images that are important for discrimination and suppress irrelevant variations while being invariant to intra-class variations. For visual recognition problems, it is effective to use features from the last layer as they are most closely related to category-level semantics. Hierarchical representations can be designed for various purposes such as tackling partial occlusions

[Ranzato et al. 2011]. The filters in low-level layers are usually smooth filters that compute local differences, therefore they are robust against illumination and registration errors. Pooling operations (*e.g.* max-pooling [Rifai et al. 2012]) usually improve robustness to registration errors further. The computational bottleneck of the representation is the convolution whose overhead depends on the size and number of filters.

In this thesis, we introduce in Chapter 3 an approach that exploits hierarchies of features based on 2D and 3D CNN for Macro-Facial Expression Recognition. We devise an architecture that takes into consideration local and global features while enforcing invariant representation and handling less constrained environmental conditions efficiently. We also give insight on how to tune the architecture parameters effectively and how to analyze the model behaviour. In Chapter 4 and Chapter 5 are also build upon a deep learning based method in which we design models for Domain Adaptation and Zero Shot Learning for Emotion Recognition and Micro Expression Detection.

## 2.4.2 Spatio-Temporal Representations and Feature Encoding

Typically, spatio-temporal representations consider a range of frames within a temporal window as a single entity, and they enable modelling temporal variations in order to represent subtle expressions more efficiently. Time represents an important dimension as it provides discrimination to the similar looking expressions in space (*e.g.* closing eyes versus eye blinking) [Kaltwang et al. 2012, Koelstra et al. 2010].

Inference of expressions can be targeted at frame-level labelling or sequence-level labelling. Frame-level inference assigns a separate label prediction to every frame, whereas sequence-level prediction assigns one label to a number of frames that make up the sequence. Sequence labelling requires a mechanism for segmenting the input data into segments and it is a more realistic but challenging scenario than frame-based labeling.

So far most existing FER features are focused on a single frame process. However, this ignores the temporal information of emotion and is not sensitive to subtle facial movements [Pantic 2009]. Moreover, using single frame process might not be available all the time due to the fact that it is difficult to properly identify and label the emotions we experience on a regular basis [Gratz and Roemer 2004].

We broadly categorize the existing research in video representation into spatio-temporal shape-based representations and spatio-temporal appearance-based representations.

### 2.4.2.1 Spatio-Temporal Shape-Based Feature Representations

Spatio-temporal shape-based feature representations usually consider 3D facial shapes. A common theme in this direction is to represent facial surfaces by certain geometrical feature sets such as the convex parts, areas with high curvatures, saddle points as proposed in [Samir et al. 2006], [Tang and Huang 2008] and [Samir et al. 2009]. These approaches model the 3D face

to estimate the movements of the facial feature points. These features are related to the AUs and their movements control the emotional states of the subject. Although such geometrical feature definitions are intuitively meaningful, the computation of curvatures involves numerical approximation of second derivatives and is very sensitive to noise. Other approaches, such as those based on shape distribution [Osada et al. 2001] and conformal geometry [Wang et al. 2006], have also been proposed. Sandbach et al. 2011 exploits 3D motion-based features between frames of 3D facial geometry sequences, where an expressive sequence is modeled and feature selection methods are then applied to extract features for each of the onset and offset segments of the expression and finally these features are used to train a Hidden Markov Model (HMM) to model the full temporal dynamics of the expression.

Moreover, another line of research considers the 2D geometry of the facial shapes and then incorporates time dimension by tracking those features. For instance, Dibeklioglu et al. 2013 provided a representation that tracks the facial landmarks, and then computes the displacement signals of eyebrows, eyelids, cheeks, and lip corners. Ghimire and Lee 2013 automatically tracks landmarks in consecutive video frames, using the displacements based an elastic bunch graph matching displacement estimation. Feature vectors from individual landmarks as well as pairs of landmarks tracking results are extracted and normalized, with respect to the first frame in the sequence. Su et al. 2014 proposes a dynamic facial expression recognition method based on the auto-regressive model to model complicated facial motions. Kacem et al. 2017 proposes a geometric approach for modeling and classifying dynamic facial sequences based on Gramian matrices derived from the facial landmarks. Their representation consists of an affine-invariant shape representation and a spatial covariance of the landmarks.

### 2.4.2.2 Spatio-Temporal Appearance-Based Feature Representations

I. **Encoding Low-Level Spatio-Temporal Features.** Recently, approaches utilizing spatial features to form BoVW models have achieved success due to their simplicity and effectiveness, though they still have difficulties when distinguishing between emotions with high inter-variations. This is because they describe facial expressions by orderless bag of visual features and they ignore the spatial and temporal structure information of visual words. Scovanner et al. 2007 extended the standard BoVW representation into spatio-temporal one to capture temporal sequential structure for the application of human action recognition. Temporal BoVW starts by detecting spatio-temporal interest points or regions from the spatio-temporal video cube and then it describes them using 3D descriptors such as Space Time Interest Points (STIP) [Laptev 2005], 3 Dimensional Histogram of Oriented Gradient (3D-HOG) [Weinland et al. 2010], 3D-SIFT [Scovanner et al. 2007], *etc.*

Scovanner et al. 2007 proposes to carry out a random sampling of a video at different locations, times, and scales for selecting interest points. Once the points are sampled, Scovanner et al. described the spatio-temporal region around the points using 3D-SIFT. The length of the descriptor for 3D-SIFT is based on the number of sub-histograms, and the number of bins. For instance, a sub-histogram for a space-time patch of size  $n \times n \times t$ , with  $n = 2$  and  $t = 2$ , and with 32 histogram bins, 3D-SIFT yields to a feature vector

length  $2 \times 2 \times 2 \times 32 = 256$ . The descriptors gathered from all the interest points are then quantized by clustering them into a pre-specified number of clusters same as the spatial BoVW representation. The resultant cluster centers are now called temporal visual words and the collection of these cluster centers is referred as the spatio-temporal word vocabulary. The same representation could be established using 3D-HOG [Weinland et al. 2010] alongside part-based representation while the rest of the steps remains the same.

For facial expression analysis, temporal BoVW has been used by Simon et al. 2010 for identifying the boundaries of the existing AU events. For this purpose Simon et al. represent an arbitrary subset of a given facial image sequence with a single histogram. To do that, each frame in the subset is represented using the part-based SIFT representation and compressed with PCA to obtain a frame-wise vector. And then, each frame-wise vector is encoded using the BoVW paradigm that measures similarity by means of multiple vectors via soft clustering. Finally, all encoded frame-wise vectors are collected in a final histogram.

In this dissertation, we use 3D-SIFT and 3D-HOG with a temporal BoVW representation to model the spatio-temporal aspects in facial image sequences and we compare their efficiency against mid-level spatio-temporal features and deep hierarchical spatio-temporal features.

**II. Encoding Mid-Level Spatio-Temporal Features.** Deep neural networks are made of multiple layers that perform the task of feature extraction from images for detection or classification tasks. Recently, transfer learning allows to use pre-trained networks and to fine-tune the weights of these pre-trained networks by continuing the backpropagation. It is possible to fine-tune all the layers of the network, or it is possible to keep some of the earlier layers fixed and only fine-tune some higher-level layers of the network. Additional layers can be added, such as CNN or Fully Connected (FC) at the last layer. Depending on the level of the layer of termination, low level or mid level features can be extracted or learned. Other alternative approaches model the spatio-temporal mid-level features by introducing a hierarchical probabilistic framework using a DBN to select informative visual and temporal cues [Zhang and Ji 2005]. Gralewski et al. 2006 propose a linear tensor framework to encode the full image sequence into a tensor and then to extract facial motion signatures and to cluster these signatures by emotion.

In our work, in order to evaluate the power of learning mid-level spatio-temporal features, we benefit from pre-trained neural networks on very large databases, mainly AlexNet [Krizhevsky et al. 2012] and facial VGG-Net [Cimpoi et al. 2015]. First, image sequences are resized to match the requirements of these trained models and then, we add at the top of the feature extractor, a Convolutional Long Short Term Memory (ConvLSTM) cell to model the dynamic behaviour. Finally a logistic classifier for sequence labelling is used.

**III. Encoding Hierarchical Spatio-Temporal Features.** CNN and RNN based methods have achieved the state of the art performances on extracting hierarchical spatio-temporal information for increasing the prediction rates on emotion recognition task as

shown in recent studies [Mollahosseini et al. 2016b, Kahou et al. 2016 and Mohammad Mahoor 2017]. In this dissertation, we develop a novel end-to-end spatiotemporal deep learning based model for facial expression recognition. We take into account local and global spatiotemporal features, which allows the analysis of long and short video clips and gives the ability to capture spatial and temporal evolutions using 3D Convolutional Neural Network (3D-CNN) and ConvLSTM networks.

### 2.5 Feature Discrimination and Dimensionality Reduction

In general, features provide information about the characteristics of the objects of interest. They represent a certain visual property of an image either globally for the whole image, or locally for objects (eyes, mouth, nose, *etc.*) or specific regions. They are representative of the maximum relevant information that the FE image has to offer for a complete characterization of emotion. On contrary, feature discrimination methodologies analyze features to extract the most relevant ones in order to enhance the differences of similar-looking expressions of different emotions by discovering the spatial or spatio-temporal regions of interest. Feature discrimination methods can be divided into three classes: pooling, feature selection and feature extraction.

1. **Pooling Operation:** reduces the dimensionality over patches and describes features within the blocks jointly. Such description ignores the location of consecutive features and thereby increases the tolerance against registration errors. A sort of techniques exists, such as binning features over local histograms [Boureau et al. 2011], sampling the minimum or maximum value within a neighbourhood [Boureau et al. 2010] or computing the sum or average of the features across a neighbourhood [LeCun 2012]. Pooling is typically used to achieve:
  - (a) Invariance to image transformations.
  - (b) Better robustness to noise and clutter.
  - (c) Robustness to lighting conditions.
  - (d) Compactness of representation while discarding irrelevant details.

Even though pooling is widely applied on the spatial domain, a number of studies applies it on spatio-temporal neighbourhoods as in [Long et al. 2012]. In this thesis, we apply pooling at multiple scales to provide non-variant effective spatial and spatio-temporal features.

2. **Feature Selection:** aims at locating a subset of features from a facial representation and optionally to weight it to get a lower dimensional refined representation. Such process could be designed to have a semantic interpretation, such as discovering regions of interest in spatial [Shan et al. 2009], [Yang et al. 2009b] and spatio-temporal [Zhao and Pietikäinen 2009] domains. Feature selection may contribute at reducing identity bias, as they are expected to discover the regions that are informative in terms of expressions rather than identity. In this thesis, for low level representation, we use DoG, 2D-Harris detector [Harris and Stephens 1988] or dense points and we demonstrate their effect on recognition. In sparse representation model, we also assist the power of Random

Features Selection (RFS) for dimensionality reduction of the initial input image.

3. **Feature Extraction:** aims to extract the most relevant features from the initial face representation. This transformation can be non-adaptive using mathematical models or adaptive using a training data. The most popular non-adaptive transformation is the Discrete Cosine Transform (DCT) whereas the most popular adaptive transformation is PCA. PCA computes a linear transformation that aims at extracting decorrelated features out of possibly correlated features. Under controlled head-pose and imaging conditions, these features capture the statistical structure of expressions efficiently [Calder et al. 2001]. In this thesis, we develop a face feature description based on the Random Projection (RP) theory. We compare this method with PCA and RFS. However, for the deep learning model, the feature extraction is done via various modules such as  $1 \times 1$ -convolution with pooling or using a fully connected layer with less number of hidden units than the previous layer.

The above-listed linear transformations encode holistic information using the whole face [Kaltwang et al. 2012], [Nicolle et al. 2012], thus, it may render the overall pipeline affected by partial occlusions. However, deep learning solves this problem by adaptive selection of proper features at each iteration while being robust to occlusions, especially if it is trained with a proper parameter regularization such as the drop out technique.

## 2.6 Recognition

The final step of an emotion recognition system is to convert the captured information about facial motions and facial feature deformations into an abstract class. The literature puts the focus on message-judgment approaches due to their ease of interpretation for facial expressions. In this direction, multi-class approaches are frequent where SVM, Multilayer Perceptron (MLP), k-NN, Bagged Trees, RF and RNN are mainly used [Jang et al. 2014, Kolodziej et al. 2018, Mostafa et al. 2018, Ramirez Cornejo and Pedrini 2018]. Deep learning methods have been proven to be excellent for computer vision tasks such as classification, where multinomial logistic classifier is mainly used [Zhang et al. 2019, Devries et al. 2014].

On the other hand, few studies explore sign-judgment approaches, where emotions are described using visual attributes based on action units. Rule-based methods are mainly employed for categorization [Pantic 2009]. Due to the uncertainty in tracking and localizing of the facial features in AU detection, heuristic rules based techniques are too sensitive to noise. For instance, spontaneous facial expressions often co-occur with natural head movements and occlusions, which makes it challenging for accurately detecting AUs. However, by considering the whole image features, the loss of information due to occlusions might be compensated by other discriminative features. In addition, the complexity of facial movements allows thousands of distinct AU combinations [Scherer and Ekman 1982]. The lack of strong theoretical basis for combining facial actions to represent complex emotional states, makes the use of EMFACS less frequent [Kim and Kim 2019].

In this dissertation, we base our work on the message-judgment approach. We extract

spatial and temporal information from image sequences directly, and map it into a pre-defined basic and non basic emotion categories. Our choice is based on the fact that it is hard to establish a set of AUs for non-basic mental states as they are expensive and hard to annotate. We use SVM classifier alongside conventional FER methods (Sections 3.2 and 3.3) and multinomial logistic classifier with deep learning models (Sections 3.4). However, in Chapter 4, we evaluate our domain adaptation and zero shot emotion recognition approaches by applying it to a k-nearest neighbor classification, because we work with embedded spaces.

## 2.7 Conclusion

In Figure 1.1, we present a general scheme regarding facial expression sources of variabilities. In Figure 2.1 we present a generic pipeline for FER algorithms. We can see that those two Figures are complementary to each other as the first describes the possible input signals such as static images versus image sequences and what choices to consider regarding facial expression description. The later tells what kind of representation among many representations can be adapted. We shed the light on our choices in Figure 2.2.

In this thesis first we work with MaEs. The first direction we follow is the use as an input signal, the apex images. Within this direction, we first consider posed expressions with discrete labels to describe basic emotions and then we consider spontaneous expressions with discrete labels to describe basic and non-basic emotions. In order to proceed with the development of FER based on the defined choices, the FER pipeline has to be define. We consider the whole face registration method and spatial appearance based representation are adapted for encoding facial expression features. Discriminate features are then extracted and classified using classifiers.

The second direction we follow is the use as an input signal, the image sequences. Within this direction, we first consider posed expressions with discrete labels to describe basic emotion and then we consider spontaneous expressions with discrete labels to describe basic and non-basic emotion similar to the previous setting. Regarding the FER pipeline, we consider again the appearance based spatio-temporal feature representation and encoding scheme.

Obviously, as we put the focus mainly on the feature representation for building a successful FER, therefore we are going to search for the best feature encoding method that yields to a proper performance. Herein, we propose to encode features at multiple levels.

Afterwards, in this thesis, we kept the focus on MaEs, but beside focusing on building feature representation that alleviate the various extrinsic and intrinsic variabilities, we put the focus also on solving the mismatch between distributions coming from training and test sets (referred as the domain shift problem). As a matter of fact, feature representation and domain shift problems are both complementary to each other.

Finally, we focus on the problem of MiE detection. Similar to the way we deal with MaE, first the whole face is considered, and then the detected face is partitioned into patches to

locate the active facial regions. Spatio-temporal appearance based representation is adapted for feature representation. In this thesis, MiE recognition is out of our scope.

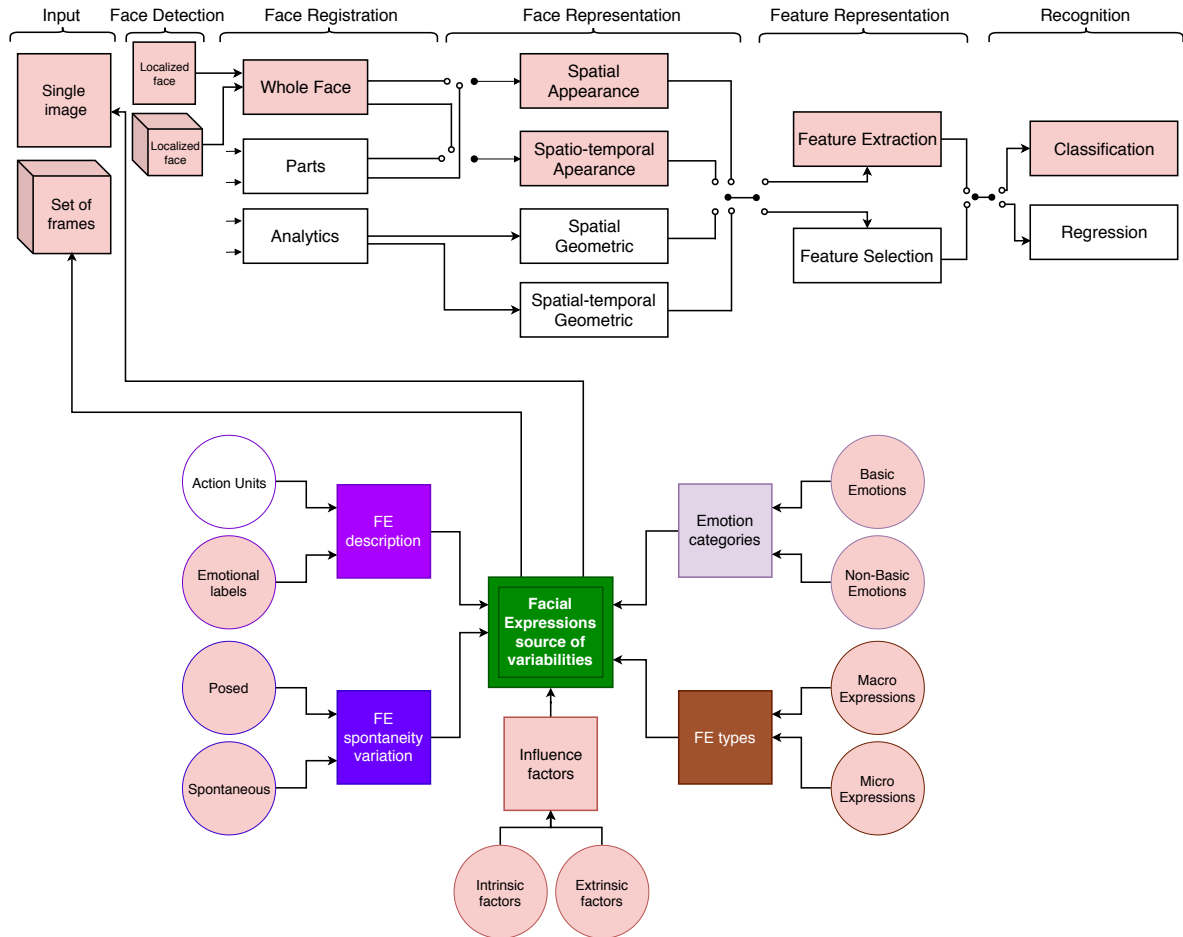


Figure 2.2: Our choices regarding facial expression recognition process and the input source of variabilities. Our choices are highlighted using colors.





# Macro Facial Expression Analysis

---

Facial expressions could appear different when presented with extrinsic and intrinsic variations even if performed by the same identity. Such changes within the same identity overwhelm the variations due to identity differences and make FER challenging, mainly in unconstrained conditions. Therefore, the complexity of discerning whether two FEs reveal similar or different emotions is shifted to the detection and description of a rich set of discriminative features. Tackling these issues requires extracting visual features that can reflect the essential visual content of the FE images while being robustly invariant to intrinsic and extrinsic factors.

In this chapter, our concern is MaE recognition and we put the focus on finding the best feature representation able to discriminate between FEs in an effective and efficient way, no matter the acquisition conditions or the expressions are. Therefore, we design different levels of facial feature appearance-based representations, mainly: low-level, mid-level and hierarchical features.

1. To extract low level features, we consider SIFT and HOG descriptors and we explore their combinations with standard BoVW and spatial pyramids for feature quantization. Upon that, we build an improved version of the standard BoVW model for low level image description as described in Section 3.2.
2. To extract mid-level discriminative features, we base our method on the sparse representation concept as described in Section 3.3. We propose a method for building discriminative dictionaries for MaE classification purpose.
3. To extract hierarchical features, we build neural networks using 2D and 3D CNNs and we devise an architecture that takes into consideration local and global features as presented in Section 3.4.

We analyze the effectiveness of those levels to handle complex situation when FEs are ambiguous, being occluded and under various environmental conditions for automatic FER. We also seek to establish the relative importance of each level of representation through a comprehensive evaluation. We believe a promising direction lies in between these three types of features.

We examine the performances of these feature representations under different sources of variabilities regarding MaEs analysis as presented in Figure 3.1. We decide to consider the description of FEs in term of discrete emotional labels. For recognizing emotions from images, we use the peak of an expression whose frame is associated with a single label. On the contrary, for recognizing emotions from image sequences, we use one label per sequence. We deal with

posed or spontaneous stimulus. Moreover, we consider Ekman’s basic emotions and mental states. The databases we use have various intrinsic and extrinsic variations.

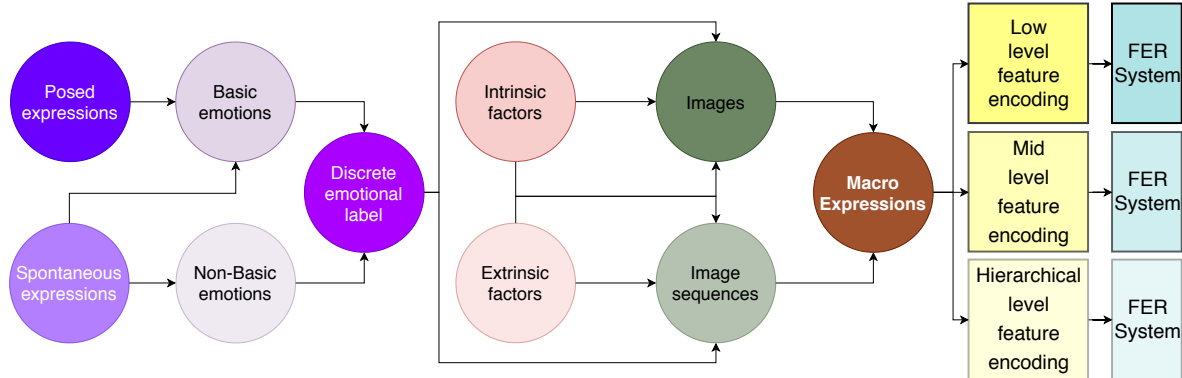


Figure 3.1: Challenges to face with MaE recognition.

In the following sections, first we describe the macro facial expression databases we use and the data protocol we propose for experimental evaluation in Section 3.1. Then, we describe the different levels of feature encoding and representations for MaE recognition.

### 3.1 Database Description and Training Protocol

Having sufficient labeled training data that include many variations about the population and various environmental conditions is an important factor for the design of an efficient FER system. In this section, we review the public available databases we use for evaluation setting. Some examples of each database are depicted in Figure 3.2. These databases contain visual recordings of subjects displaying Ekman’s basic and non-basic FEs, either acted or spontaneous, with frontal or near-frontal head poses and under different lighting conditions. We describe the dataset protocol setting used for splitting the data into training, validation and test sets, which follows the basic rule that identities which appear in the training set should not appear in the test set. The discussed databases represent either image-based or video-based data. Video databases are also used to evaluate image based approaches by extracting the most expressive facial images of the sequences.

**Setting up training, validation and test sets:** In general, they are defined as:

1. Training set: data used for algorithm learning..
2. Validation (development) set: is used to tune parameters, select features, and make other decisions regarding the learning algorithm performance.
3. Test set: is used to evaluate the generalization performance of an algorithm, but not to make any decision about what the algorithm is learning or how to tune parameters. Test data are unseen data.

**The Japanese Female Facial Expression (JAFFE)** database developed by Lyons et al. 1998, is a laboratory-controlled image database made of acted FEs related to Ekman’s



Figure 3.2: The databases represented from top to down correspond to: JAFFE, CK+, MMI, DISFA, DynEmo and MUG databases respectively.

emotions ((1) *anger*, (2) *disgust*, (3) *fear*, (4) *happiness*, (5) *sadness*, (6) *surprise*, plus (7) *neutral*). It contains 213 images of posed expressions from 10 Japanese females. The resolution of the original facial images is  $256 \times 256$  pixels. The number of images corresponding to each of the seven categories of expressions is 30 images per class. Each identity has roughly 3 images per class. The database is considered as challenging because it contains few examples per subject and expression. It has often been used in literature to evaluate the performance of FER algorithms. For devising the data protocol, we consider 147 images that belong to 7 identities as training set and the rest of 66 images that belong to 3 other identities are considered as test set. For validation set, we consider a leave-one-out cross validation technique over the training set to tune the algorithm hyper-parameters because the total number of data is limited.

**The Extended Cohn-Kanade Facial Expression (CK+)** database developed by Lucey et al. 2010, is composed of 123 identities. Each expression is represented as a sequence of images, starting from the neutral face up to the peak of expression, which means the offset segment is not included in this database. Each sequence is associated with one

label. It includes mainly posed Ekman expressions and spontaneous smiles. The length of the sequences varies from short to long. The minimal sequence length is 4 frames which is the case when acted expressions are performed with high intensity and thus reaching its peak of expressiveness very fast. The maximum sequence length is 71 frames. From CK+ database, 321 sequences are selected. These sequences are associated to 118 identities. For training phase, 291 sequences that belong to 107 identities are used, resulting in around 3500 expressive images. For validation phase, other 14 image sequences that belong to 6 other identities are used, resulting in 200 expressive images. For test phase, 16 sequences that belong to other 5 identities are used, resulting in 250 expressive images.

**The MMI Facial Expression (MMI)** database developed by Pantic et al. 2005, contains recordings of the full temporal patterns of a MaEs, from neutral, through a series of onset, apex, and offset phases and back again to a neutral face. From the MMI database, 203 videos are selected that are associated with an Ekman emotion label. The shortest sequence has a length of 33 images while the longest sequence has a length of 201 frames. The selected videos are performed by 30 identities ranging in age between 19 and 62, where 9 of them are females while 21 are males. They also have a different ethnic background. In our training protocol, 24 identities with 161 sequences are used for training resulting in around 1800 expressive images. While 3 other identities with 21 sequences are considered for validation phase yielding to 280 expressive images. Finally, the last 3 identities with 21 sequences are considered for testing phase yielding to 300 expressive images. Out of the 203 videos, 7 videos contain the frontal and the side views at the same time, among which 4 are performed by males and 3 by females. We include two male in the test set and one female in the validation set and the rest in the training set.

**The Denver Intensity of Spontaneous Facial Actions (DISFA)** database developed by Mavadati et al. 2013, is one of the few spontaneous facial expression databases. It contains 27 videos of subjects (12 females and 15 males) with different ethnicities. The facial behavior is recorded by using a high-resolution camera ( $1024 \times 768$  pixels) at 20 frames per second (fps) under uniform illumination. Each video is around 242 seconds, resulting in approximately 130,000 images ( $27 \times 242 \times 20$ ). Each frame is provided with an AUs annotation. In our study, each video is segmented into 9 segments and associated to one of the Ekman's basic emotion. To do that, we use Table 1.1 which is based on the EMFACS system [Ekman and Rosenberg 1997] to convert AU FACS codes to discrete emotional class labels. Out of 9 segments we extract 11 sequences. The minimal sequence length is 60 images while the longest sequence length is 1380 images. In total, 297 sequences are extracted from the 27 videos (11 sequences per subject). For training purpose, we use 19 subjects of 209 sequences, resulting in 2500 expressive images. For validation 4 subjects of 44 sequences are extracted yielding to 550 expressive images and for test another different 4 subjects of 44 sequences are used resulting in also 550 expressive images.

**The Video database of Natural Facial Expressions of Emotions (DynEmo)** [Tcherkassof et al. 2013] is a database containing elicited dynamic spontaneous facial expressions. It contains two sets of 233 and 125 recordings of facial expressions of ordinary Caucasian people (ages 25 to 65, 182 females and 176 males) filmed in natural but standardized condi-

tions. We only use Set 2, where emotional facial expression recordings are both associated with the emotional state of the expresser and with the time line (continuous annotations) of observers rating the recorded videos while displayed on a screen. Each video contains several emotions represented through a time line at rate 25 fps. Each subject may express different emotions in the video as well as the same emotion might be represented at different times. Out of the 125 recordings, 198 sequences corresponding to the six mental states: *Affected*, *Amusement*, *Curious*, *Disappointment*, *Fright*, and *Astonished* are extracted. The selected sequences are performed by around 21 identities per label and per different expressions (in total 125 different identities). For training, 138 sequences (23 sequences per expression) are used which result around 1500 expressive images in total. For validation and test, each set has around 30 sequences (5 sequences per expression), which results in around 400 expressive images in total for each set.

**The MUG Facial Expression (MUG)** [Aifanti and Delopoulos 2010] is a database containing image sequences of 86 subjects performing basic posed MaEs. The database contains 35 women and 51 men all of Caucasian origin between 20 and 35 years of age. Men are with or without beards. The subjects are not wearing glasses except for 7 subjects. There are no occlusions except for a few hair falling on the face. The image sequences of 77 subjects are available and each sequence contains 50 to 160 images. The number of the available sequences counts up to 1462 and they are categorically labeled. Out of 1462 sequences, we collected 70654 images. Those images are used in chapter 4 as a source domain database. However, this database is not used in Chapter 3 for different feature levels evaluation as we accessed to this database recently.

A summary that characterizes each of the macro facial expression database is given in Table 3.1.

## 3.2 A Low Level Feature Encoding Based on BoVW for FER

Bag of Visual Words models, which represent an image as a histogram of local features, have become the most popular method for image classification tasks. They have been first introduced by Sivic and Zisserman 2003 for object matching in videos. Sivic and Zisserman described the BoVW method as an analogy with text retrieval and analysis, in which a document is represented by word frequencies without regard to their order. The word frequencies are considered as the signature of the document and are then used to perform classification. This approach obtained the state of the art performance on several applications such as human action recognition [Peng et al. 2016], scene classification [Zhu et al. 2016] and face recognition [Hariri et al. 2017]. To the best of our knowledge, this approach has not been investigated for the task of facial expression recognition since Ionescu et al. 2013.

In this thesis, we aim at introducing some improvements over the standard BoVW method for FER. We also want to explore the generalization power of low-level features in case of strong deformations on faces as it is the case with acted FEs and in case of subtle deformations as it is the case with spontaneous FEs. Our choice for using the BoVW method as a technique

| Database | Subject | Samples                           | Elicitation method      | Expression distribution | Influencing factors   | Training set  | Validation set                              | Test set                                    |
|----------|---------|-----------------------------------|-------------------------|-------------------------|---|---|---|---|
| JAFFE    | 10      | 213 images                        | Posed expressions       | Ekman basic emotions    | 10 Japanese female<br>Frontal view<br>No occlusion  | 7 identities<br>143 images                                    | Leave-one-out cross validation              | 3 identities<br>66 images                   |
| CK+      | 118     | 321 image sequences               | Posed expressions       | Ekman basic emotions    | Ages 18 to 30<br>Multi ethnicity<br>Frontal view<br>AUs coded<br>No occlusion                 | 107 identities<br>291 sequences<br>3500 images                | 6 identities<br>14 sequences<br>200 images  | 5 identities<br>16 sequences<br>250 images  |
| MMI      | 30      | 203 image sequences               | Posed expressions       | Ekman basic emotions    | Ages 19 to 62<br>9 females and 21 males<br>Frontal and side view<br>AUs coded<br>No occlusion | 21 identities<br>161 sequences<br>1800 images                 | 3 identities<br>21 sequences<br>280 image   | 3 identities<br>21 sequences<br>300 images  |
| DISFA    | 27      | 27 sequences with 297 sequences   | Spontaneous expressions | Ekman basic emotions    | 12 females and 15 males<br>Multi ethnicity<br>AUs coded<br>Partial occlusion                  | 19 identities<br>209 sequences<br>2500 images                 | 4 identities<br>44 sequences<br>550 images  | 4 identities<br>44 sequences<br>550 images  |
| DynEmo   | 125     | 125 sequences with 198 sequences  | Spontaneous expressions | Non-basic emotions      | 65 females and 59 males<br>Multi ethnicity<br>AUs coded<br>Partial occlusion                  | 89 identities<br>138 sequences<br>1500 images                 | 18 identities<br>30 sequences<br>400 images | 18 identities<br>30 sequences<br>400 images |
| MUG      | 77      | 125 sequences with 1462 sequences | Posed expressions       | Basic emotions          | 30 females and 47 males<br>Caucasian<br>AUs coded<br>No occlusion                             | all 77 identities with 70654 images are used as source domain |   |   |

Table 3.1: A summary about the macro facial expression databases and their training protocol used in this thesis.

to tackle facial expression classification is motivated by the fact that differences between facial expressions are contained in the image changes of location, shape and texture of facial clues (eyes, nose, eyebrows, *etc*) and thus the results of classification could be provided based on the similarity of the contents of facial expression images.

The standard steps for deriving the signature for a facial expression image using BoVW are represented in Figure 3.3, (1): keypoints localization from the image, (2): keypoints description using local descriptors, (3): vector quantization of the descriptors by clustering them into  $k$ -clusters, resulting a visual word vocabulary which forms the codebook, (4): establishing the signature of each image by accumulating the visual words into a histogram, (5): normalizing the histogram by dividing the frequency of each visual word over the total number of visual words, and (6): training a classifier using the obtained image signature for classification task.

The BoVW model has been the most frequent and dominant used technique for visual content description. However this approach has some drawbacks that affect the performance. First, during the feature detection process, a large number of keypoints are extracted. This increases the computational process, in addition to the fact that most of these keypoints arise in the background regions. The second problem is the poor clustering, when the usual clustering method is the Lloyd's algorithm referred as k-means algorithm, due to the fact that several local features are encoded with the same visual word. The third limitation is that as standard BoVW represents an image with an unordered collection of local descriptors, thus the spatial organization of the data is lost. The final drawback is the weighting scheme, where standard BoVW considers all visual words equally while there might be some visual words that are of greater importance.

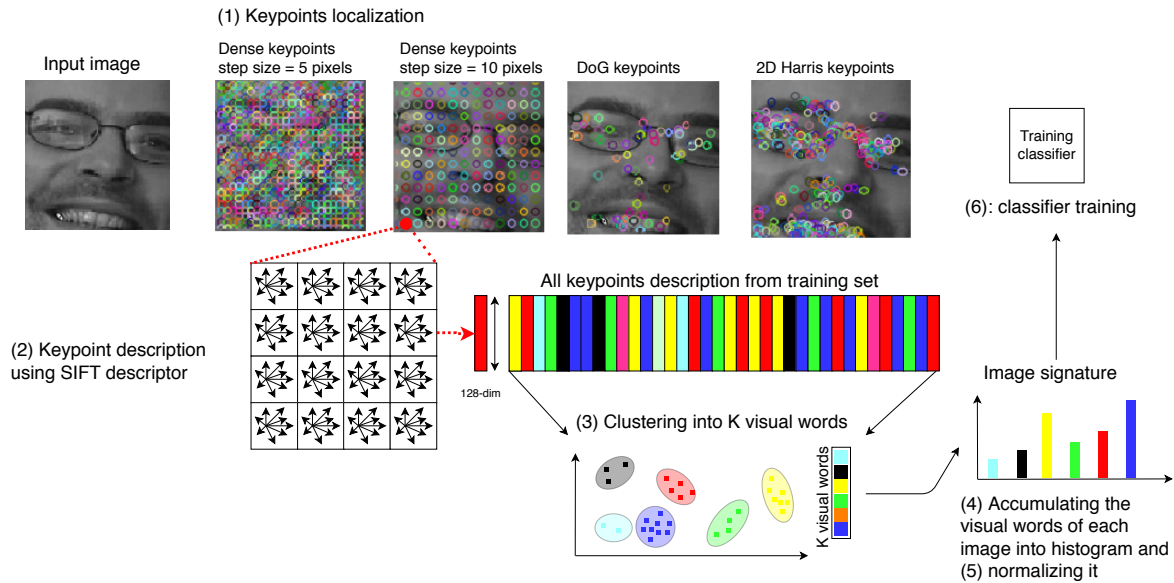


Figure 3.3: Standard BoVW representation for facial expressive image.

In the literature, many attempts have been conducted to improve the standard BoVW model. Lazebnik et al. 2006a proposed an extension of the BoVW model to exploit the spatial information based on spatial pyramid matching. This technique works by partitioning the image into increasingly fine sub-regions and computing histograms of local features found inside each sub-region. The technique shows significantly improved performances on scene categorization tasks. In Zhang et al. 2011 descriptive visual words and descriptive visual phrases are proposed as visual correspondences to text words and phrases, where visual phrases refer to the frequently co-occurring visual word pairs. In Xie et al. 2013 the descriptive ability of visual vocabulary has been investigated by proposing a weighting based method. In Altintakan and Yazici 2015 the k-means clustering method has been replaced by Self-Organizing Maps (SOM) in codebook generation as an alternative method. Obviously, state of the art methods which proposed to improve the standard BoVW method are dealing with one limitation at a time. In this thesis, we perform several improvements, almost at each step of the standard method.

We investigate the use of BoVW for acted and spontaneous macro facial expression recognition. More specifically, we search for the best feature detector that suits our application. We integrate the use of the k-means++ method [Arthur and Vassilvitskii 2007] with the BoVW method instead of k-means algorithm in an attempt to obtain a more distinctive codebook. We develop relative conjunction matrices in order to preserve the spatial organization of the data. We introduce an efficient weighting scheme based on Term-Frequency Inverse-Document-Frequency (TF.IDF), in order to scale up the rare visual words while damping the effect of the frequent visual words. Finally, for learning a distinctive SVM classifier, we use the histogram intersection kernel since we experience much faster training than with the Radial Basis Function (RBF) kernel.



### 3.2.1 Beyond the Standard Bag of Visual Words Model

#### 3.2.1.1 Feature Selection and Description

In order to recognize facial expressions, low level visual features could be extracted from the facial deformations of the appearance of the facial shape. For example, anger on a face can be characterized by: eyebrows pulled down, upper lids pulled up, lower lids pulled up, lips may be tightened. Thereby, facial visual ROIs such as: eyes, nose, mouth, cheeks, eyebrow, forehead, *etc.*, provide observable changes when this emotion occurs. Therefore, a feature detector that locates keypoints inside those ROIs would limit the risk of generating a huge number of redundant keypoints. Back to Figure 3.3, we can see that the 2D-Harris detector [Harris and Stephens 1988] is focused on locating keypoints over the ROIs. Although the DoG detector [Lowe 2004] has also focused on ROIs, the keypoints are not as numerous as required. Moreover, even though dense feature extraction is known to be good for many classification problems [Furuya and Ohbuchi 2009], for facial expression recognition, the located keypoints do not have to be huge and redundant as it is the case with dense features as shown in Figure 3.3.

Afterwards, the extracted keypoints are described using local descriptors. In our case SIFT descriptors are used because they are invariant towards extrinsic variations while being able to carry enough discriminative information. We also investigate the performance of other low level descriptors mainly HOG and the combination of SIFT-HOG features.

#### 3.2.1.2 k-means++ Clustering Algorithm

The next step is to perform vector quantization in order to quantize the space into a discrete number of visual words. This step is important to map the image from a set of high-dimensional descriptors to a reduced list of visual words and thus to provide a distinctive codebook. The usual method is to use the k-means algorithm which generates arbitrary bad clustering specially when it is unbounded between n-data points and k-integers (pre-defined number of clusters) [Arthur and Vassilvitskii 2007]. The simplicity of the k-means algorithm comes at the price of accuracy. To tackle this problem, we propose the use of the k-means++ algorithm which is the result of augmenting the k-means algorithm with a randomized seeding technique. The augmentation improves both the speed and the accuracy of the k-means method. The main steps of the k-means++ clustering are:

1. Choose an initial center uniformly at random from the data points.
2. For each data point  $x$ , compute  $D(x)$ , the distance between  $x$  and the nearest center that has already been chosen.
3. Choose one new data point at random as the new center, using a weighted probability distribution where a point  $x$  is chosen with a probability proportional to  $D(x)^2$ .
4. Repeat steps 2 and 3 until a total of  $k$  centers has been selected.
5. Proceed as with the standard k-means algorithm for clustering step.

## 3.2.1.3 Relative Conjunction Matrix

The BoVW approach describes an image as a bag of discrete visual words. The frequency distributions of these words are used for image categorization. The standard BoVW approach yields to a not complete representation of the data due to the fact that image features are modeled as independent and orderless visual words. Thus there is no explicit use of visual word positions within the image. Traditional visual words based methods suffer when facing with similar appearances but distinct semantic concepts [Aldavert et al. 2015]. In this study, we assume that establishing spatial dependencies might be useful for preserving the spatial organization of data. Thus we develop a novel facial image representation which uses the concept of the Relative Conjunction Matrix (RCM) to take into account links between the visual words. Thereby, a facial image is described by a histogram of pair-wise visual words to provide a more discriminative representation since it contains the spatial arrangement of the visual words.

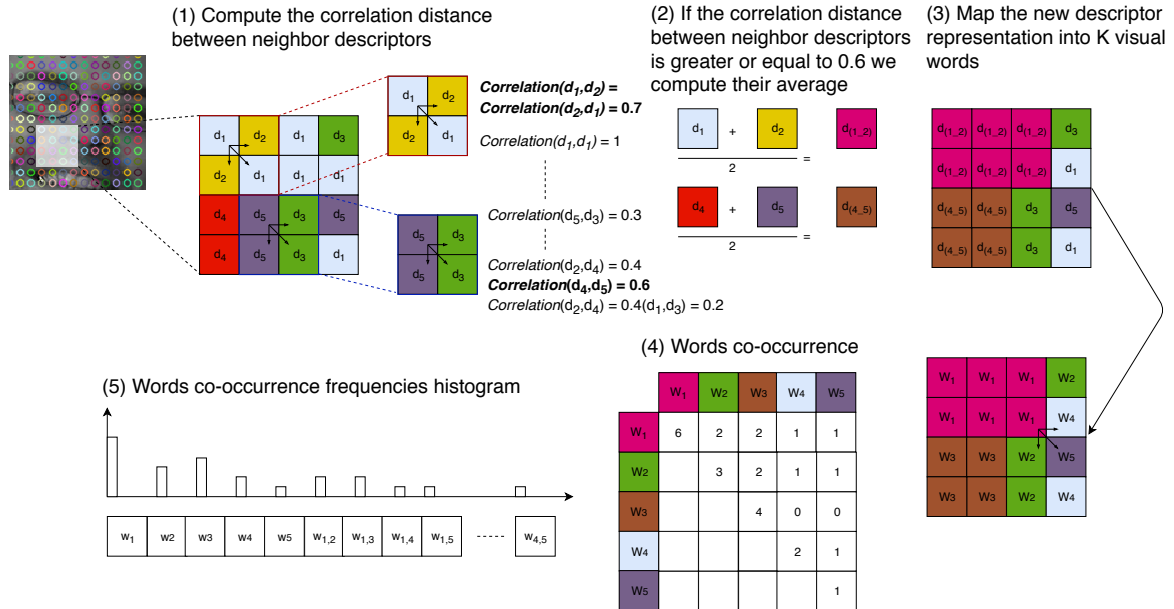


Figure 3.4: The steps of computing the relative conjunction matrix in order to take into account the spatial organization of the data.

A demonstration on how to compute the RCM is shown in Figure 3.4. An RCM of visual words defines the spatial order by first quantifying the relationships of the neighboring descriptors in which the distance correlation between the vectors of any two neighbour descriptors is computed (eq. (3.1)). If the correlation is above a certain threshold (experimentally we found that 0.6 is a good threshold), we join them together into a new grouping. Then, we map this new descriptor representation into K-visual words. Afterwards, we define an RCM to establish pairs of visual words by looking for all possible pairs of visual words. This can be considered as a representation of the contextual distribution of each visual word with respect to other visual words of the vocabulary. The relative conjunction matrix  $C$  has a size  $N \times N$ , where  $N$  is the vocabulary size. Each element  $C_{i,j}$  represents the pair of one independent feature to

another. The obtained  $C$  has all possible pairs. Each row vector of  $C$  stores how many times a particular visual word (for example  $W_1$ ) occurs with any other visual words (for example  $W_2, W_3, W_4, \dots, W_N$ ). For a particular facial expression, if any two visual words have similar contextual distributions, that means they are capturing something similar. Thus, they are related to each other. The diagonal and the upper part of  $C$  are considered for quantification.

$$\text{correlation}(u, v) = 1 - \frac{(u - \mu(u))(v - \mu(v))}{\|u - \mu(u)\|_2 \|v - \mu(v)\|_2} \quad (3.1)$$

### 3.2.1.4 TF.IDF weighting scheme

Weighting the visual words is crucial for classification performance but the standard BoVW just normalizes the visual words by dividing them with the total number of visual words in the image. Van Gemert et al. 2010 investigates several types of soft-assignments of visual words to image features and prove that choosing the right weight scheme can improve the recognition performance. Each weight has to take into account the importance of each visual word in the image. For FER, we are interested in scaling up the weights corresponding to the visual words extracted from the nose, the eyebrows, and the mouth etc. while damping the effect of frequent visual words that describe the hair and some non-deformable regions like the background. Therefore, it is possible to leverage the usage of the TF.IDF [Leskovec et al. 2014] weighting scheme to scale up the rare visual words while scaling down the frequent ones.

The standard weighting method used in the standard BoVW approach is equivalent to the Term Frequency (TF).  $\text{TF}(vw)$ , where  $vw$  is the visual word, measures how frequently a visual word occurs in an image. It is normalized by dividing it with the total number of visual words in the image. The utilization of  $\text{TF}(vw)$  in classification is rather straightforward and usually results in a decreased accuracy due to the fact that all visual words are considered equally important.

However, Inverse-Document-Frequency (IDF)( $vw$ ) of a visual word  $vw$  assigns different weights to features. It provides information about the general distribution of visual words  $vw$  amongst facial images of all classes. The utilization of  $\text{IDF}(vw)$  is based on its ability to distinguish between visual words with some semantical meanings and simple visual words. The  $\text{IDF}(vw)$  measures how unique a  $vw$  is and how infrequently it occurs across all training facial expression images.

$$\text{IDF}(vw) = \log\left(\frac{T}{n_{vw}}\right) \quad (3.2)$$

$T$ : total number of training images.

$n_{vw}$ : number of occurrences of  $vw$  in the whole training database  $T$ .

However, if we assume that certain visual words may appear a lot of times but have little importance, then we need to weight down the most frequent visual words while scaling up

the rare ones, by computing the Term-Frequency Inverse-Document-Frequency referred as TF.IDF. Where:

$$\text{TF.IDF}(vw) = \text{TF}(vw) \cdot \log\left(\frac{T}{n_{vw}}\right) \quad (3.3)$$

The  $\text{TF.IDF}_{vw,I}$  assigns to the visual word  $vw$  a weight in image  $I$ , where  $I \in T$ , such that: the weight is high when  $vw$  occurs frequently within a small number of images, thus lending high discriminating power to those images. On the contrary, the weight is low when the  $vw$  occurs less frequently in an image, or occurs in many images (for example,  $vw \in \text{background}$ ), thus offering a less pronounced relevant signal.

### 3.2.2 Facial Expression Classification Algorithm

The proposed Improved Bag of Visual Words (ImpBoVW) model for facial expression classification is summarized as follows:

1. Locate and extract keypoints from facial images either based on a feature detector (salient keypoints) such as: DoG, 2D-Harris detector, or by defining a grid with pre-specified spatial step (for example 5 pixels a priori keypoints).
2. Compute local features over the selected keypoints. SIFT, HOG, or SIFT-HOG are considered.
3. Quantize the descriptors gathered from all the keypoints by clustering them into  $k$ -clusters, using k-Mean++. This quantizes the space into a pre-specified number (vocabulary size) of visual words. The cluster centers represent the visual words which form the codebook.
4. Map a set of high dimensional descriptors into a list of visual words by assigning the nearest visual word to each of its component. This results the histogram of visual words. It summarizes the entire facial image and it is considered as the signature of the image.
5. Build feature grouping among the words. A co-occurrence based criterion is used for learning discriminative word groupings using the RCM.
6. Introduce the proper TF.IDF weighting scheme based on eq. (3.3).
7. Train a SVM classifier over the diagonal and the upper parts of the weighted conjunction matrix for facial expressions recognition. The histogram Intersection kernel (eq. (3.4)) is used by the SVM to learn a discriminative classifier.

### 3.2.3 State of the Art BoVW Representation Methods for Comparison Purpose

In order to evaluate the effectiveness of the proposed method and for fair comparison, we have implemented the standard BoVW method and the spatial pyramid BoVW model developed in [Lazebnik et al. 2006a].

The Spatial Pyramid Bag of Visual Words (SP BoVW) representation is an extension of an orderless BoVW image representation with a spatial pyramid, which has shown significantly improved performances on scene categorization tasks as shown in [Zhu et al. 2016]. The SP BoVW aims at subdividing the image into increasingly fine resolutions and at computing histograms of local features. Thus, it aggregates statistics of local features over fixed sub-regions.

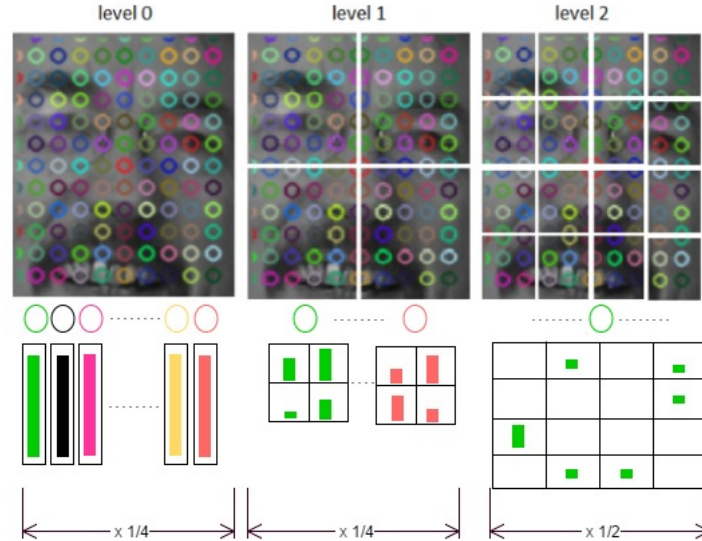


Figure 3.5: The construction of a three level spatial pyramid. The image has different feature types, indicated by different colors. At the top, the facial image is sliced at two different levels of resolution. Then, for each resolution and each channel, the features that fall in each spatial bin are counted. Finally, each spatial histogram is weighted according to eq. (3.5).

The histogram of visual words in this method represents images as the histogram of a series of visual key words, which are extracted from 128-SIFT features of training images via k-means. Then a series of key words of different resolution are extracted via a decomposition method to get the structural features of the images.

In spatial pyramid, as shown in Figure 3.5, an image is expressed in several layers. Each layer contains some feature blocks, in which the feature cell of the  $0_{th}$  layer is the image itself. From the  $0_{th}$  layer until the  $L_{th}$  layer, each cell of the previous one is divided into four non-overlapping parts. At last, the features of each cell are oined together as the final descriptor. In this scheme, the feature of the  $0_{th}$  layer is presented by a  $d$ -dimensional vector, corresponding to  $D$  blocks in the histogram, then that of the  $1_{th}$  layer is presented by a 4d-dimensional vector, while the  $2_{th}$  layer is presented by a 8d-dimensional vector, and so forth. Therefore, for the descriptor of the  $L_{th}$  layer, the dimension of feature vector is  $D = \sum_{i=0}^L 4^i$ . To better reveal the pyramid features, the sparse blocks at the bottom are assigned with larger weights in the Pyramid Matching Kernel (PMK) function, and the dense blocks at the top are assigned with smaller ones.

A match between two keypoints occurs if they fall into the same cell of the grid. Suppose

$X$  and  $Y$  are two sets of vectors in a  $d$ -dimensional feature space. Let us construct a sequence of grids at resolutions  $0, \dots, L$ , such that the grid at level  $l$  has  $2^l$  cells along each dimension, for a total of  $D = 2^{dl}$  cells. Let  $H_X^l$  and  $H_Y^l$  denote the histograms of  $X$  and  $Y$  at this resolution, such that  $H_X^l(i)$  and  $H_Y^l(i)$  are the number of points from  $X$  and  $Y$  that fall into the  $i$ th cell of the grid. Then the total number of the histogram orthogonal kernel at level  $l$  in  $D$  blocks is given by the histogram intersection function:

$$I(H_X^l, H_Y^l) = \sum_{i=1}^D \min(H_X^l(i), H_Y^l(i)) \quad (3.4)$$

The weight associated with level  $l$  is proportional to the cell width at that level:  $\frac{1}{2^{L-l}}$ . However, the PMK aims at penalizing matches found in larger cells since they involve increasing dissimilar features.

$$\begin{aligned} \text{PMK}^L(X, Y) &= I(H_X^l, H_Y^l)^L + \sum_{l=0}^{L-1} \frac{1}{2^{L-l}} (I^l - I^{l+1}) \\ &= \frac{1}{2^L} I^0 + \sum_{l=1}^L \frac{1}{2^{L-l+1}} I^l \end{aligned} \quad (3.5)$$

Equation (3.5) is known as Mercer kernel that combines both the histogram intersection and the PMK [Grauman and Darrell 2007].

For the spatial pyramid representation, the pyramid matching in the  $2D$ -image space is performed and the k-means clustering algorithm is used to quantize all feature vectors into  $M$  discrete channels. Each channel gives two  $2D$  vectors,  $X_m$  and  $Y_m$  corresponding to the coordinates of the features of channel  $m$  found in the respective images.

The final kernel (FK) represents the sum of the separate channel kernels:

$$\text{FK}^L(X, Y) = \sum_{m=1}^M \text{PMK}^L(X_m, Y_m) \quad (3.6)$$

### 3.2.4 Experimental Setup and Analysis

In this section, we present the experimental design used to evaluate the proposed algorithm and compare it to the Standard Bag of Visual Words (SBoVW) and SP BoVW approaches. First, we present the datasets used, then, we describe the evaluation procedure and finally we present the results.

**Data Exploration.** For effective and fair comparison, five different databases corresponding to MaEs are used with different intrinsic and extrinsic factors, three with Ekman's posed FEs,

namely: the JAFFE database, the CK+ and the MMI FE databases and one with spontaneous Ekman’s FEs, the DISFA. Moreover, we assess the methods performances over mental state non-basic spontaneous FE using the DynEmo database. The data protocol has been discussed earlier in Section 3.1 and summarized in Table 3.1.

**Evaluation Setting.** We focus the experimental evaluation of the proposed method on the following four questions: *What is the best point of interest detector that locates the most salient and reliable keypoints for FER? Does each of the proposed novelties improve the performance with respect to the SBoVW method? What is the influence of using k-means++? What is the influence of changing the keypoints descriptor from SIFT to HOG or considering their combination? Is the proposed approach efficient for posed and spontaneous, basic and non-basic FEs? Is it scalable for large databases?*

The proposed model has many parameters that influence the classification performance: the weighting scheme TF.IDF, the RCM, the combination of TF.IDF weighting scheme along with RCM, the k-mean and k-mean++ as the clustering method, the choice of the best keypoints detector and descriptor. Thereby, in order to answer the first three questions, we report performances of FER with the SBoVW model along with each novelty. The JAFFE (posed and basic FEs) and the DynEmo (spontaneous and non-basic) databases are used for the method validation and setting using static images. The final two questions are addressed using all the databases to establish the performance of the ImpBoVW model compared to SBoVW and SP BoVW using SIFT, HOG, and the combination of SIFT-HOG as descriptors for keypoints.

The multi-class classification is done using an SVM trained using the one-versus-all rule. The histogram intersection kernel presented in eq. (3.4) is used. Compared to RBF kernel, we experience faster computations while the accuracy rate has a smaller variance. The validation set is used in order to tune the algorithm hyper-parameters such as the regularization parameter  $C$  (the optimal value is 10.0). We fix the vocabulary size to 2000 visual words, since experimentally it shows the best classification performance. For spatial pyramid representation, we notice that at level-2 and level-3 the same performance is achieved. Thus to reduce the complexity of feature computation, we only report two levels.

Figures 3.6a and 3.6b present an ablation study where the performance of each novelty and its contribution over the JAFFE and DynEmo databases are reported respectively. The best model is referred to ImpBoVW which is a combination of the SBoVW method along with RCM and TF.IDF weighting scheme, its quantization is based on k-mean++ and its features are located using the 2D-Harris detector and described using the SIFT descriptors.

With our proposed ImpBoVW model, the average recognition rates obtained over the JAFFE and DynEmo databases are 92% and 64% respectively. If we evaluate the same model but with the DoG as keypoints detector, we got 84% accuracy rate for the JAFFE database and 49% for the DynEmo database. In the same manner, replacing 2D-Harris with dense keypoints results 89,5% accuracy rate over the JAFFE database and 55% accuracy rate over the DynEmo database. As a result, it appears that 2D-Harris detector is the most efficient. Moreover, it facilitates to reduce the complexity of the whole algorithm than dense points as it produces

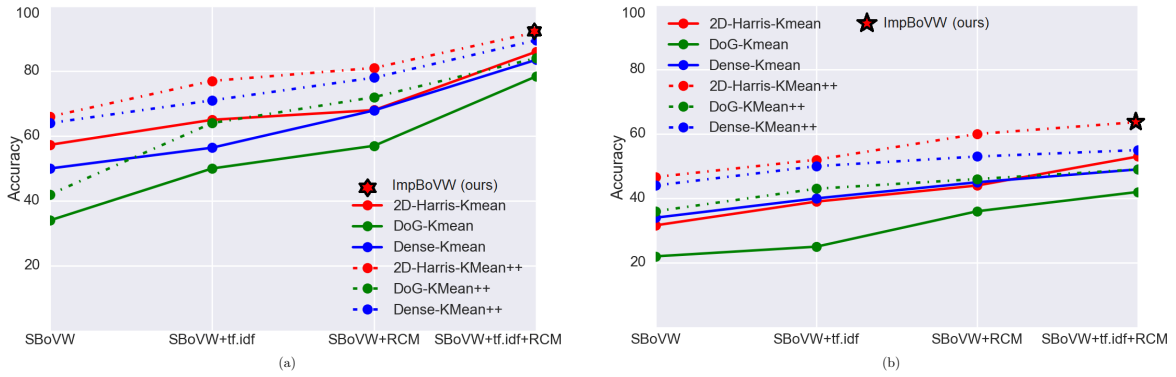


Figure 3.6: Classification accuracies of the FER system starting from the standard BoVW method successively improved either with TF.IDF or RCM to finish with the complete model ImpBoVW (SBoVW + TF.IDF + RCM). SIFT descriptors are used to extract low level features from keypoints. For each configuration of the algorithm, different keypoint detectors and different clustering methods are tested. (a): Over posed basic Ekman’s MaEs using the JAFFE database. (b): Over spontaneous non-basic MaEs using the DynEmo database.

less redundant keypoints. In addition, as shown in Figures 3.6a and 3.6b, the performances of each novelty along the SBoVW model gradually increase to reach its best performance over ImpBoVW, as our method enhances the global representational quality of the image signature. Those Figures also shows that the usage of k-means++ (represented with dotted lines) increases the classification rate significantly. Moreover, properly weighting the visual words with TF.IDF contributes in an increased performance and taking into consideration the spatial organization of the data using RCM even further improves the results.

Figure 3.7a addresses the generalization classification performances of the proposed ImpBoVW compared to the SBoVW and to the SP BoVW over acted and spontaneous databases with basic and non-basic expressions. The classification accuracies of the FER system starting from the SBoVW method is slightly improved with SP BoVW and significantly improved with ImpBoVW. We can observe that the performances of all methods over acted Ekman’s expressions databases achieves rate above 80%, with 91% for the JAFFE, 86% for the CK+ and 81% for the MMI databases. However, most methods struggle to achieve a good performance over spontaneous databases with basic and non-basic expressions, as it is the case with the DynEmo and the DISFA databases, for which the maximum classification rates are 62% and 64% respectively. Obviously, ImpBoVW achieves better performances than the SBoVW and the SP BoVW, whatever the database is considered.

Apparently, the power of the optimized low level features we propose to successfully overcome the controlled intrinsic and extrinsic factors over acted databases, on average, achieves  $\frac{91+86+81}{3} = 86\%$  recognition rate. Among those acted databases, the MMI database is the most challenging, as it includes frontal and profile views of subjects. Obviously, it is one of the main reason why ImpBoVW struggles to achieve a high rate as it is the case over the CK+ database or over the JAFFE database. Moreover, our feature encoding and representation method achieves much less performance when these intrinsic and extrinsic factors are more



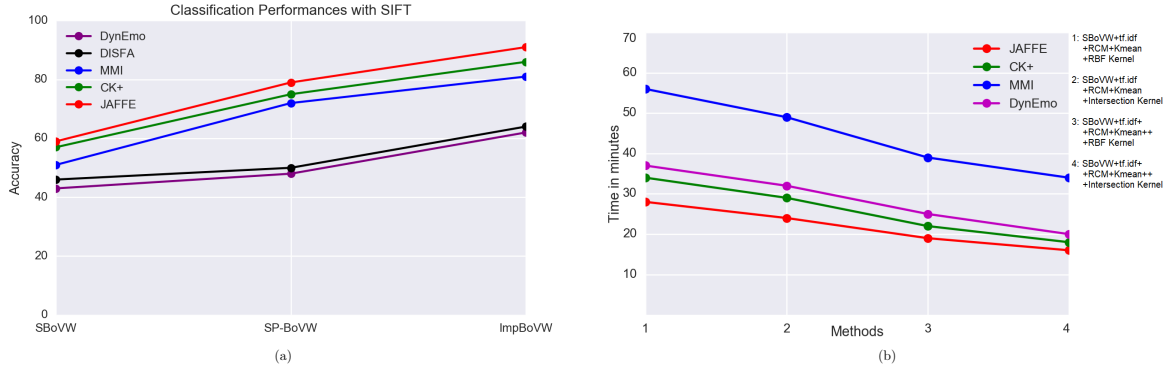


Figure 3.7: (a) Performances over five databases compared with SBoVW and SP BoVW while using SIFT descriptor. (b): Computational times for generating the BoVW features and classifying them into emotional categories.

sever and less controlled, as it is the case of the spontaneous databases, reporting on average around  $\frac{64+62}{2} = 63\%$  recognition rate. It sounds that low level features are not robust enough to build a reliable FER system.



Figure 3.8: The raw input image corresponds to a frightened spontaneous expression class along its corresponding HOG features that encode gradient orientations.

**Feature Descriptors:** We adapt the proposed method and we evaluate the performance again using different feature descriptors. We consider SIFT, HOG and their combination. Figure 3.8 presents an image along side its HOG feature representation. Figure 3.9 presents the boxplots for comparing the ImpBoVW alongside different feature descriptors. We notice a slight increase in the recognition rate while using the combination of SIFT-HOG. However, HOG alone slightly performs less than SIFT. Considering both descriptors would increase the time complexity, therefore, only SIFT is kept.

**Computational Time Performance:** We evaluate the time complexity of the proposed approach  $SBoVW + RCM + TF.IDF$  during vector quantization using either k-mean or k-mean++ method and also during training an SVM using either RBF kernel or histogram intersection kernel over acted and spontaneous databases. We present our results in Figure 3.7b in minutes. Method number 4 represents the ImpBoVW method we propose. The figure shows

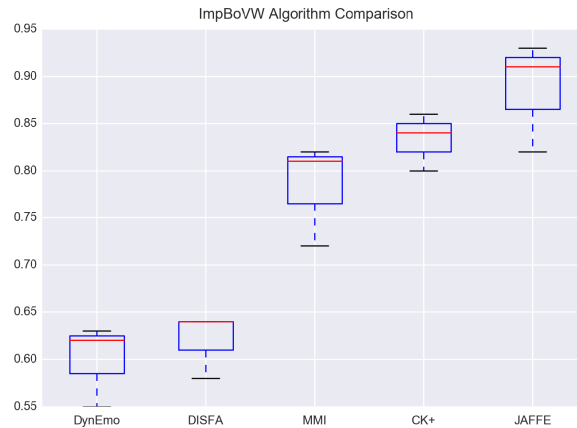


Figure 3.9: The recognition rates of the ImpBoVW on different databases. The *minimal values* correspond to *HOG*. The *red line* corresponds to *SIFT*. The *maximum values* correspond to their combination, *HOG+SIFT*.

the contribution of k-Mean++ in speeding up the process of vector quantization. In addition, the histogram intersection kernel has also contributed in decreasing the complexity of the learning part of the SVM. Figure 3.7b shows that our method achieves a good computational time compared to the original algorithm.

### 3.2.5 Conclusion

We introduced an improved BoVW approach for automatic FER. We examined several aspects of the SBoVW approach that are linked directly to gain classification performance, speed and scalability. It has been proved that the 2D Harris detector is suitable for emotion recognition. It selects adaptable and reliable salient keypoints. We improve the codebook generation process through employing k-means++ as a clustering method, wherein we gain speed and accuracy. The importance of spatial organization of the data has been taken into account in the feature representation by introducing an RCM to preserve the spatial order. We properly weighted the visual words after preserving the spatial order using TF.IDF based on their occurrences. Histogram intersection kernel has been used to decrease the complexity of the algorithm. We implemented SP BoVW for comparison purpose and different feature detection methods have been evaluated.

We noticed that extracting low level features around the eyes, nose, mouth and eyebrows plays a major role in achieving good FER performances only if strong deformations are present on those areas, which is the case with acted basic expressions presented in the JAFFE, CK+ and MMI databases. These particular regions of interest act as active areas changing along with the expressions. Within this setting of strong deformations, if the extrinsic and intrinsic factors are not fully controlled, as it is the case in MMI database where frontal and profile views are shown, low level features alone cannot provide distinctive features and thus leading to inferior recognition rate. We also noticed that for spontaneous expressions setting,

coming from basic and mental states presented in the DISFA and DynEmo databases respectively, where the facial deformations are subtle, low-level features alone are not so efficient to discriminate between different classes.

Therefore, a further investigation on the best way to encode facial expression features is needed. Hence, we propose to go beyond low level features and to inspect the power of mid level features for FER under different source of variabilities. Therefore, we propose to use the sparse representation concept to encode and describe facial expression features in sparse space. Our intuition is that sparse representation may enable us to deal with signal corruptions such as noise due occluded faces and to deal with subtle features it combines the discrimination power of reconstruction and discrimination power of classification.

### 3.3 A Mid Level Feature Encoding Based on Sparse Representation for FER

Sparse representation has been successfully applied to a variety of problems, including image denoising [Elad and Aharon 2006], image restoration [Mairal et al. 2008] and has been also introduced for image classification [Bradley and Bagnell 2008; Wright et al. 2008; Yang et al. 2009a]. This method has proven to be a good tool for representing and compressing high-dimensional signals [Yang et al. 2010] and offering a good classification performance if it is well designed [Huang and Aviyente 2007].

Sparse representation is a typical reconstructive method has not been widely used for FER. It aims at obtaining a representation that enables sufficient reconstruction, being able to deal with signal corruption like noise or missing data [Huang and Aviyente 2006]. It works well in applications such as image inpainting and coding. However, for image classification, discriminative methods such as linear discriminative analysis [Li and Yuan 2005] are often outperforming reconstructive methods. Basically, this is because discriminative methods look for generating a feature representation that maximizes the separation of distributions of images from different classes. Indeed, the success of the discriminative methods is conditioned under the assumption that the images to be classified are ideal, *i.e.*, being noiseless, complete and without extrinsic variations. However, the classification performances may degrade when the model is trained on a database that insufficiently encompasses various intrinsic and extrinsic variations. On the other hand, reconstructive methods have shown successful performance in addressing these problems. Combining the discrimination power with the reconstruction property and the sparsity of sparse representation in a unified framework would probably lead to better classification performances.

Our work is inspired by the good reputation of sparse representation in both theoretical researches and practical applications [Bradley and Bagnell 2008; Mairal et al. 2008; Wright et al. 2008]. Moreover, we are motivated for combining both reconstructive and discriminative methods to handle difficult spontaneous facial behaviour situations where intrinsic and extrinsic factor are challenging. Sparse representations have also the ability to provide sparse

### 3.3. A Mid Level Feature Encoding Based on Sparse Representation for FER 55

feature vectors that can share the same sparsity pattern at class level if it is correctly built, which is the main key to achieve an accurate recognition rate and gain generalization power.

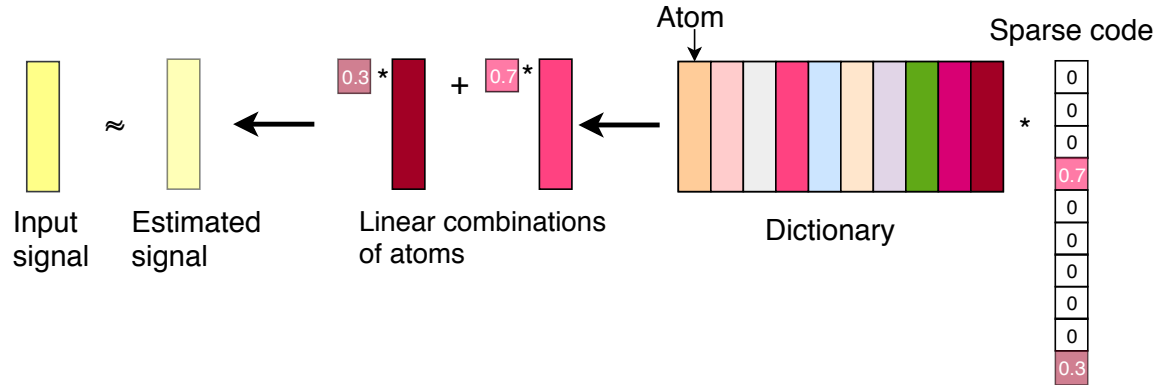


Figure 3.10: Graphical illustration of sparse representation concept. An input signal can be approximated by a linear combination of a set of atoms that compose a dictionary, such that the sparse code has as few as possible non-zero coefficients. For instance, the input signal here is reconstructed using two atoms out of ten. The sparse code is used as the feature vector for training the classifier.

The problem of sparse representation for FER can be solved by searching for the most compact representation of an image in terms of linear combinations of atoms (also referred to as basis vectors) using a pre-specified dictionary which is either learned or handcrafted. Then classifying a new facial expression image is achieved through finding its sparse coefficients with respect to the dictionary and comparing them with the combinations designed for each expression to be recognized or learned. A general demonstration is shown in Figure 3.10.

The success of the model relies on the quality of the dictionary that sparsifies the signals. The choice of a proper dictionary can be done using one of the two following ways [Rubinstein et al. 2010]: building a sparsifying dictionary by combining multiple standard transforms [Olshausen et al. 2001], including Curvelet Transform, Ridgelets Transform and DCT or by learning it from a training dataset. For instance, Wright et al. employ the entire set of training samples as the dictionary for discriminative sparse coding, and achieve impressive performances for face recognition. Many algorithms [Mairal et al. 2010 and Wang et al. 2010] have been proposed to efficiently learn an over-complete dictionary (when the number of atoms is much greater than the features size) that enforces some discriminative criteria. To successfully build a generalized framework for emotion classification, a dictionary learning based method is adopted.

In this thesis, we investigate the use of sparse representation for MaE recognition by considering the K-Singular Value Decomposition (K-SVD) [Aharon et al. 2006] and the Orthogonal Matching Pursuit (OMP) [Pati et al. 1993] for dictionary building and sparse coding respectively. The K-SVD algorithm is well suited for building dictionaries for image reconstruction but those dictionaries lack of discriminant capability for classification. Their objective function is designed to minimize the signal reconstruction error with as few basis vectors as possible.

Therefore searching for the sparse representation of a signal can be achieved by optimizing an objective function that includes two terms [Huang and Aviyente 2007], one measuring the signal reconstruction error and the other measuring the sparsity. Thereby, to induce discriminative capabilities while building a dictionary, that is how to effectively encourage the images from the same class to have similar sparse code patterns and those from different classes to have dissimilar sparse code patterns, we look into the global structure of the dictionary. Herein, several questions arise: given a dictionary  $D$  and an input signal  $y$ :

1. Is it possible to find a sparse code  $x$  that is *sparse enough* but still estimates the signal  $\hat{y} \approx y$  up to a certain error? If so, is there a *unique sparse representation matrix*  $X$  for different input samples  $Y$  coming from the same class label?
2. Is it possible to look for a representation of a test sample that uses a minimum number of atoms that correspond to the right class?
3. What is the optimal dictionary structure that satisfies both sparsity and reconstruction constraints while satisfying the first two questions? Mathematical models? Learnable? How to initialize it?
4. Can we build a compact dictionary with few basis vectors that leads to a good classification performances when the number of training data in each class is relatively small?

In order to answer the previous questions, let us introduce a simple example. Given a dictionary  $D$  with linearly dependent atoms having a strong correlation between each other, thus, the sparse code matrix  $X$  of an input sample matrix  $Y$  of a certain class label  $l_i$  using  $D$  will have many solutions while still achieving a low reconstruction error. Thus, as the first property is satisfied, *i.e.* a low reconstruction error, the obtained sparse code matrix  $X$  is not unique which might be a problem regarding classification purpose. Thereby, in order to obtain a unique sparse representation matrix for classification, the first condition is that its basis vectors of the dictionary to be *linearly independent* for those corresponding to different classes and linearly dependent if they belong to the same class. In other words, the dictionary atoms should be orthogonal.

Now, suppose we already satisfied this first condition, we would like to obtain a sparse code matrix  $X$  as sparse as possible, that is to encode any signal using few atoms. If the dictionary atoms are *informative* enough, the sparsity level could be dropped, since only few atoms would be capable of representing the input samples  $Y$ . By that, it would be possible to encode any test sample using a compact dictionary with few atoms per class without the need for a huge over-complete dictionary to achieve robust classification performances.

Obviously, the global structure of the dictionary is the main key to induce sparsity, discriminability and to achieve low reconstruction errors. However, different tasks have different dictionary learning rules. For image classification, the dictionary must contain discriminative and informative information in a way that the sparse representation code possesses the capability of distinctiveness between different classes. It must have linearly independent columns for those belonging to different classes while having columns that are highly informative and correlated for those belonging to the same class.

In order to achieve the best dictionary conditions for robust classification, we propose a

### 3.3. A Mid Level Feature Encoding Based on Sparse Representation for FER 57

pre-training stage for dictionary initialization based on the random projection concept that facilitates the discriminability of the sparse code. To do so, we develop a Random Face Feature Descriptor (RFFD) for dictionary pre-training that elegantly solves the problem of shared subspace distribution by decorrelating features vectors of different classes from each other. RFFD aims at projecting the raw data into a lower-dimensional space, while preserving their reconstructive and discriminative properties. Beside, it seeks for the best transformation matrix that maximizes the separation between the multiple classes which is the main key to induce sparsity and discrimination. Then, the pre-trained dictionary is refined via the K-SVD and OMP algorithms. By that, we compute sparse coefficients that address only a few atoms of the dictionary and more importantly those corresponding to the training data from a single class. Those sparse coefficients are used then to train an SVM classifier. We refer to our method as Spontaneous Facial Expression Recognition (SPFER) and its main stages are presented in Figure 3.11. Our method learns a structured and compact dictionary whose atoms have correspondences with class labels, so that the provided sparse code can be used to distinguish different classes as it is designed to have small within-class scatter and large between-class scatter.

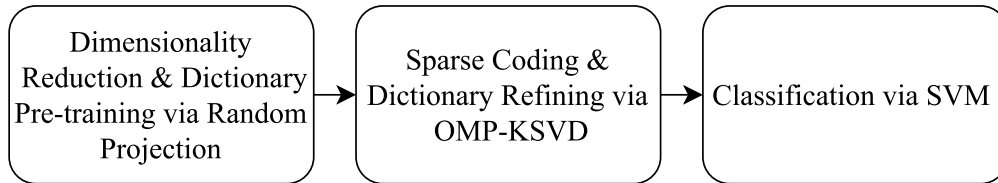


Figure 3.11: SPFER algorithm composed of three main stages, first the pre-training for dictionary initialization, then the sparse coding and the dictionary refining and finally the classification stage.

Next, we detail the formulation of sparse representation problem and how it can be solved using the K-SVD and OMP algorithms. Then, we present some of the state of the art methods that solve the sparsity problem for the classification task. Afterwards, we detail the stages of our SPFER method shown in Figure 3.11. Finally, we present the experimental results and analysis.

#### 3.3.1 Formulation of Sparse Representation and Dictionary Learning

Let us consider a fixed dictionary matrix  $D$  with  $m$  known atoms or basis vectors  $\{\mathbf{d}_1, \mathbf{d}_2, \dots, \mathbf{d}_m\} \in \mathbb{R}^d$ , with  $m > d$ . Each column of  $D$  represents one atom. Let  $\mathbf{y} \in \mathbb{R}^d$  be a column vector that represents an input signal. Suppose that all the atoms are used to approximately represent the input signal, thus it can be expressed as:

$$\mathbf{y} = \mathbf{d}_1 x_1 + \mathbf{d}_2 x_2 + \dots + \mathbf{d}_m x_m, \quad (3.7)$$

where  $x_i$ ,  $\{i\}_1^m$ , is the coefficient of  $\mathbf{d}_i$ . Equation (3.7) can be rewritten as:

$$\mathbf{y} = D\mathbf{x}, \quad (3.8)$$

where matrix  $D = [\mathbf{d}_1, \dots, \mathbf{d}_m]$  and  $\mathbf{x} = [x_1, \dots, x_m]^\top$ .

The stated problem in eq. (3.8) is an underdetermined linear system, ill-posed problem and it is incapable to uniquely represent the input signal  $\mathbf{y}$  using the dictionary matrix  $D$ . Nevertheless, it would be possible to impose an appropriate constraint on the representation solution  $\mathbf{x}$  to mitigate this difficulty. The sparse representation method conditions the linear combination of the dictionary matrix  $D$  to represent the input signal  $\mathbf{y}$  such that as many as possible of the coefficients  $\mathbf{x}$  are zero or close to zero and only few entries are differentially large. By that, the obtained representation is constraint to be sparse. This is formulated as a  $\ell_0$ -norm minimization constraint (eq. (3.9)) [Donoho and Elad 2003]

$$\hat{\mathbf{x}} = \operatorname{argmin} \|\mathbf{x}\|_0 \quad \text{such that} \quad \mathbf{y} = D\mathbf{x}, \quad (3.9)$$

where  $\|\cdot\|_0$  is the measure of sparsity referring to the number of nonzero entries in a vector.

**Computing the Sparse Code Using a Fixed Dictionary.** Greedy strategy approaches [Elad 2010, Tropp et al. 2006] address sparse representations with the  $\ell_0$ -norm minimization (eq. (3.9)) in a special way because they only seek an approximate sparse representation solution. The Matching Pursuit (MP) algorithm [Mallat and Zhang 1993] and the OMP algorithm [Pati et al. 1993, Tropp and Gilbert 2007] are among the most popular algorithms in this category to solve the problem stated in eq. (3.9). In this dissertation, to compute the sparse code  $\mathbf{x}$  of an input signal  $\mathbf{y}$  given a pre-specified dictionary  $D$ , we employ the OMP algorithm. The main steps for the OMP process are illustrated in Algorithm 1.

---

**Algorithm 1:** Orthogonal Matching Pursuit Algorithm

---

**task** : Approximate the constraint problem in eq. (3.9).  
**input** : Input signal  $\mathbf{y}$  and dictionary matrix  $D$ .  
**initialization:**  $t = 1$ ,  $\mathbf{r}_0 = \mathbf{y}$ ,  $\mathbf{x} = \mathbf{0}$ ,  $D_0 = \emptyset$ ,  
 $\emptyset$  denotes empty set,  $\Lambda_0 = \emptyset$  and  $\tau$  is a small constant threshold.  
**while**  $\|\mathbf{r}_t\| > \tau$  **do**  
    1. Find the best matching sample: the largest inner product between  
         $\mathbf{r}_{t-1}$  and  $\mathbf{d}_j$  ( $j \notin \Lambda_{t-1}$ ) exploiting  $\lambda_t = \operatorname{argmax}_{j \notin \Lambda_{t-1}} |\langle \mathbf{r}_{t-1}, \mathbf{d}_j \rangle|$ .  
    2. Update the index set  $\Lambda_t = \Lambda_{t-1} \cup \lambda_t$  and  
        reconstruct the dataset  $D_t = [D_{t-1}, \mathbf{d}_{\lambda_t}]$ .  
    3. Compute the sparse coefficient by using  
        the least square algorithm  $\hat{\mathbf{x}} = \operatorname{argmin} \|\mathbf{y} - D_t \hat{\mathbf{x}}\|$ .  
    4. Update the representation residual  $\mathbf{r}_t = \mathbf{y} - D_t \hat{\mathbf{x}}$   
    5. Iterate index  $t=t+1$   
**end**  
**output** : The dictionary  $D$  and the sparse code  $\mathbf{x}$

---

**Learning a Dictionary.** The literature has focused on building dictionaries that can provide sparse representations either by exploiting a pre-specified set of transformation functions like transform domain methods [Olshausen et al. 2001] to represent the image samples, or is devised

### 3.3. A Mid Level Feature Encoding Based on Sparse Representation for FER 59

based on learning methods. In the following, we formulate the problem of dictionary learning as an optimization problem based on the notions of the literature [Cheng et al. 2013, Aharon et al. 2006 and Zhang et al. 2015b]:

$$\operatorname{argmin}_{D \in \Omega, \mathbf{x}_i} \left\{ \frac{1}{N} \sum_{i=1}^N \frac{1}{2} \|\mathbf{y}_i - D\mathbf{x}_i\|_2^2 + \lambda P(\mathbf{x}_i) \right\}, \quad (3.10)$$

where  $\Omega = \{D = [\mathbf{d}_1, \dots, \mathbf{d}_M], \mathbf{d}_i^T \mathbf{d}_i = 1, \{i\}_1^M\}$ , with  $M$  atoms and  $N$  training samples.

$\mathbf{y}_i$  is the  $i$ -th sample vector from a known set.  $D$  is the learned dictionary and  $\mathbf{x}_i$  is the sparse code.  $P(\mathbf{x}_i)$  is the regularization term that controls the degree of sparsity and  $\lambda$  is a tuning parameter. The penalty term  $P(\mathbf{x}_i)$  can be defined by introducing the  $\ell_0$ -norm which leads to the sparsest solution of eq. (3.10).

Correspondingly, the theory of sparse representation can be applied to dictionary learning. The K-SVD is one of the most powerful dictionary learning algorithms using the  $\ell_0$ -norm penalty which is broadly adopted in image processing applications such as image compression, and feature coding of image representation [Bryt and Elad 2008, Wang et al. 2010]. The K-SVD algorithm is a generalization of k-means algorithm and it falls in the category of clustering based dictionary learning approaches. Thus, it is an unsupervised based method as the class label is not exploited in the process of learning the dictionary. The objective function of the K-SVD is formulated as:

$$\operatorname{argmin}_{D, X} \{ \|Y - DX\|_F^2 \} \quad \text{such that} \quad \|\mathbf{x}_i\|_0 \leq k, \{i\}_1^N, \quad (3.11)$$

where  $Y \in \mathbb{R}^{d \times N}$ ,  $D \in \mathbb{R}^{d \times M}$ , while  $M = N$ , which means we deploy all the training samples as atoms,  $X \in \mathbb{R}^{N \times N}$  is the sparse matrix of coefficients and  $k$  is the maximum number of allowed non-zero entries (sparsity limit).

Problem (3.11) is a *joint optimization problem* with respect to  $D$  and  $X$ , and therefore to solve it, an iterative and alternative optimization process between  $D$  and  $X$  is required. That is:

1. **First**, we consider the initial dictionary  $D$  as fixed and therefore Problem (3.11) is converted into sparse coding (eq. (3.12)):

$$\operatorname{argmin}_X \{ \|Y - DX\|_F^2 \} \quad \text{such that} \quad \|\mathbf{x}_i\|_0 \leq k, \{i\}_1^N. \quad (3.12)$$

Then its sub-problem is:

$$\operatorname{argmin}_{\mathbf{x}_i} \{ \|\mathbf{y}_i - D\mathbf{x}_i\|_2^2 \} \quad \text{such that} \quad \|\mathbf{x}_i\|_0 \leq k, \{i\}_1^N,$$

and can be solved by deploying the OMP Algorithm 1 for estimating  $\mathbf{x}_i$ .



2. **Second**, we consider the computed  $X$  as fixed and therefore Problem (3.11) is converted into a regression model (eq. (3.13)) for obtaining  $D$ , that is:

$$\hat{D} = \underset{D}{\operatorname{argmin}} \|Y - DX\|_F^2, \quad (3.13)$$

where  $\hat{D} = YX^\dagger$ ,  $\dagger$  means it requires inverse problem solving,  $\hat{D} = YX^\top(XX^\top)^{-1}$ . Considering the computational complexity of the inverse problem in solving eq. (3.13), the problem can be rewritten to update the dictionary  $D$  by fixing the other variables as well such as:

$$\begin{aligned} \hat{D} &= \underset{D}{\operatorname{argmin}} \|Y - DX\|_F^2 \\ &= \underset{D}{\operatorname{argmin}} \|Y - \sum_{j=1}^N \mathbf{d}_j \mathbf{x}_j^\top\|_F^2 \\ &= \underset{D}{\operatorname{argmin}} \|(Y - \sum_{j \neq l} \mathbf{d}_j \mathbf{x}_j^\top) - \mathbf{d}_l \mathbf{x}_l^\top\|_F^2, \end{aligned} \quad (3.14)$$

where  $\mathbf{x}_j$  is the  $j$ -th row vector of the matrix  $X$ . The residual representation  $E_l = (Y - \sum_{j \neq l} \mathbf{d}_j \mathbf{x}_j^\top) - \mathbf{d}_l \mathbf{x}_l^\top$  is computed first. Then  $\mathbf{d}_l$  and  $\mathbf{x}_l$  are updated. In order to maintain sparsity of the  $\mathbf{x}_l^\top$ , only the nonzero entries of  $E_l$  should be kept, *i.e.*,  $E_l^p$ , from  $\mathbf{d}_l \mathbf{x}_l^\top$ . Then, the SVD decomposes  $E_l^p$  into  $E_l^p = U\lambda V^\top$  followed by a dictionary update for  $\mathbf{d}_l$ .

The K-SVD steps for dictionary learning is summarized in Algorithm 2.

---

**Algorithm 2:** The K-SVD algorithm for dictionary learning and sparse coding.

---

**task** : Learning a dictionary  $D$  by solving the objective function (3.11).  
**input** : The matrix composed of given  $N$  samples  $Y = [\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_N]$ .  
**initialization:** Set the initial dictionary  $D$  to the  $\ell_2$ -norm unit matrix  
**while** *not converged* **do**  
    1. **for**  $i$  *in*  $N$  **do**  
        Given example  $\mathbf{y}_i$  employing OMP Algorithm (1) to solve the task problem for further estimating  $X_i$ , set  $l = 1$ ;  
        (a) **while**  $l$  *is not equal to*  $k$  **do**  
            i. Compute the overall representation residual  
                 $E_l = Y - \sum_{j \neq l} \mathbf{d}_j \mathbf{x}_j^\top$ ;  
            ii. Extract the column items of  $E_l$  which correspond to the nonzero elements of  $\mathbf{x}_l^\top$  and obtain  $E_l^p$ ;  
            iii. Perform SVD to decompose  $E_l^p$  into  $E_l^p = U\lambda V^\top$ ;  
            iv. Update  $\mathbf{d}_l$  to the first column of  $U$  and update the corresponding coefficients in  $\mathbf{x}_l^\top$  by  $\lambda(1, 1)$  times the first column of  $V$ ;  
            v.  $l = l + 1$ ;  
        **end**  
    **end**  
    2.  $i = i + 1$   
**end**  
**output** : The dictionary  $D$  and the sparse code matrix  $X$ .

---

### 3.3. A Mid Level Feature Encoding Based on Sparse Representation for FER 61

#### 3.3.2 Related Works Regarding Classification Using Sparse Representation

In the learning-based approach, machine learning methods are used to construct the dictionary from the training data. In this category, Wright et al. 2008 proposes a Sparse Representation Based Classification (SRC) approach for face recognition, to construct the dictionary by manually selecting training samples. SRC is a method that looks for the sparsest representation of a test example with the hope to select few training data from the correct class. It typically tackles the problem of sparsity using  $\ell_1$ -norm. In SRC, the reconstruction errors based on different classes are used to classify testing data. For each class  $i$ , a function is defined as  $\delta_i$ , which selects the sparse code associated to  $i$ -th class. The process of SRC is shown in Algorithm (3).

---

**Algorithm 3:** SRC algorithm

---

|               |   |
|---------------|---|
| <b>task</b>   | : Approximate the constraint problem $\hat{\alpha} = L(\alpha, \lambda) = \operatorname{argmin}_{\alpha} \frac{1}{2} \ \mathbf{y} - X\alpha\ _2^2 + \lambda \ \alpha\ _1$ |
| <b>input</b>  | : Input signal $\mathbf{y}$ and a dictionary $D$ with $c$ classes.  |
|               | 1. Solve $\ell_1$ -regularized least square equation:<br>$\operatorname{argmin}_x \ \mathbf{y} - D\mathbf{x}\ _2^2 + \lambda \ \mathbf{x}\ _1$                            |
|               | 2. Calculate the residuals based on categories:<br>$r_i(\mathbf{y}) = \ \mathbf{y} - D\delta_i(\mathbf{x})\ _2$ for $i = 1, \dots, c$                                     |
| <b>output</b> | : label( $\mathbf{y}$ ) = $\operatorname{argmin} r_i(\mathbf{y})$   |

---

Jiang et al. 2013 proposes the Label Consistent K-Singular Value Decomposition (LC-KSVD) approach as a supervised learning method for learning discriminative dictionary dedicated to image classification. Jiang et al. exploit the class label information to learn the dictionary and integrate the process of constructing the dictionary and an optimal linear classifier into a mixed reconstructive and discriminative objective function. LC-KSVD algorithm jointly obtains the learned dictionary and the classifier. Therefore, its objective function (eq. (3.15)) is formulated with three terms, where the first term is related to the reconstruction error, the second term is related to the discriminative sparse-code error, and the final term is related to the classification error.

$$\langle D, A, C, X \rangle = \operatorname{argmin}_{D, A, C, X} \|Y - DX\|_F^2 + \mu \|L - AX\|_F^2 + \eta \|H - CX\|_F^2 \text{ s.t. } \|\mathbf{x}_i\|_0 \leq k \quad (3.15)$$

where  $\mu$  and  $\eta$  are the weights of the corresponding contribution items,  $Y$  is the matrix composed of all the input data,  $D$  is the learned dictionary,  $X$  is the sparse code,  $A$  is a linear transformation matrix,  $H$  is the matrix composed of label information corresponding to  $Y$ ,  $C$  is the parameter term for the classifier and  $L$  is a joint label matrix for labels of  $Y$  and  $D$ .

While SRC and LC-KSVD have been shown to be effective for image classification, we compare our method to those studies.

### 3.3.3 The SPFER Algorithm

The detail of each stage of our method (SPFER) is presented. First, we state the concept of dimensionality reduction and in particular using RP. Then we present our face descriptor RFFD for dimensionality reduction and for satisfying the initial conditions of the dictionary, linearly independent and informative atoms for those belonging to different classes while linearly dependent and informative atoms for those belonging to the same class. Then, the dictionary refining and sparse coding stages is presented and followed by the classification stage.

#### 3.3.3.1 Dimensionality Reduction and Dictionary Pre-training

Dimensionality reduction has been the subject of keen studies for the past few decades, and rather than trying to outline this work we will focus upon three popular contemporary techniques: principal component analysis, random feature selection and random projection.

**Principal Component Analysis.** One popular method for feature extraction and dimensionality reduction is PCA [Goel et al. 2005]. PCA reduces multidimensional dataset into lower dimensions in a way that the low-dimensional subspace of the data describes as much of the variance in the data as possible. It attempts to find a set of orthonormal vectors that better represents the data points. These vectors are often called principal components. In particular, the first component  $\xi_1$  is found as the vector with the largest variance, *i.e.*

$$\xi_1 = \operatorname{argmax}_{\|u\|=1} \sum_{i=1}^n \langle x_i, u \rangle^2$$

and inductively,  $\xi_i$  is found as the vector with the largest variance among all the vectors that are orthogonal to  $\xi_1, \xi_2, \dots, \xi_{i-1}$ . In order to apply PCA for dimension reduction, we simply take the  $k$  first components out to obtain a matrix, and then form the new lower-dimensional data point.

**Random Features Selection.** Another technique explored in this work is RFS. The features are selected uniformly at random out of the original feature space resulting in a smaller feature space. Armano et al. 2011 builds an ensemble of classifiers which receive a random subset of the original features. In this thesis we aim to assess the method of RFS in a single classifier in order to compare it to the RP and PCA methods.

**Random Projection Theory and Concept.** A dimensionality reduction technique that is capable to preserve the reconstructive or discriminative properties of the original data can be marked as ideal [Sulic et al. 2010]. In this thesis, the random projection is used. With such a method, the original high-dimensional data are projected onto a low-dimensional subspace by using a random matrix  $R$ , where all the column vectors are normalized to be unit norm  $r_i \leftarrow \frac{r_i}{\|r_i\|_2}$ . This normalization makes sure that the dot product (correlation) between any

### 3.3. A Mid Level Feature Encoding Based on Sparse Representation for FER 63

two columns is within the range  $\{-1, +1\}$  and hence the absolute value of correlation between any two columns is bounded by 1, *i.e.*,  $0 \leq |r_i.r_j| \leq 1$ .

It is a computationally simple and efficient method that preserves the structure of the data without significant distortion [Goel et al. 2005, Sanghai et al. 2005, and Magen 2002]. As opposed to PCA, random projection is much cheaper to compute. Moreover, RPs are data-independent, they are constructed regardless how the point set is distributed.

The concept of RP is as follows: Given a data matrix  $X$  be a set of  $N$  points in  $\mathbb{R}^d$ , the dimensionality of data can be reduced by projecting it onto a lower-dimensional subspace formed by a set of random vectors  $R$  [Kaski 1998]:

$$A^{m \times N} = R^{m \times d} \cdot X^{d \times N}, \quad (3.16)$$

where  $d$  is the original dimension, and  $m$  is the desired lower dimension. The central idea of RP is based on the Johnson-Lindenstrauss lemma (JL lemma) [Johnson and Lindenstrauss 1984], which asserts that any subset of  $\mathbb{R}^d$  can be embedded into a low-dimension space  $\mathbb{R}^m$ , whilst keeping the Euclidean distance between any two points of the set almost the same. Formally, it is stated as follows: for any  $0 < \varepsilon < 1$  and any integer  $N$ , let  $m$  be a positive integer such that,

$$m \geq 4 \cdot \left( \frac{\varepsilon^2}{2} - \frac{\varepsilon^3}{3} \right)^{-1} \cdot \ln(N). \quad (3.17)$$

Then, for any set  $V$  of  $N$  points in  $R^d$ , there is a map  $f$  such as:  $R^d \rightarrow R^m$  subjected to:  $\forall \mu, \nu \in V$  [Dasgupta and Gupta 1999],

$$(1 - \varepsilon) \|\mu - \nu\|^2 \leq \|f(\mu) - f(\nu)\|^2 \leq (1 + \varepsilon) \|\mu - \nu\|^2, \quad (3.18)$$

where  $f(\mu)$  and  $f(\nu)$  are the projections of  $\mu$  and  $\nu$ .

Using the above lemma, [Dasgupta and Gupta 1999] shows that if we perform an orthogonal projection of  $N$  points in a vector space ( $R^f$ ) onto a selected lower-dimensional subspace, then distances between points are preserved (*i.e.*, not distorted more than a factor of  $(1 \pm \varepsilon)$ , for any  $0 < \varepsilon < 1$ ). For complete proofs of the lemma refer to [Dasgupta and Gupta 1999] and [Tsagkatakis and Savakis 2009].

The choice of the random matrix  $R$  is one of the crucial points of interest. Tsagkatakis and Savakis 2009 employs a random matrix  $R$  whose elements are drawn i.i.d from a zero mean, bounded variance distribution. There are many choices for the random matrix. A random matrix with elements generated by a normal distribution  $r_{i,j} \sim N(0, 1)$  is one of the simplest.

**Random Face Feature Descriptor.** A RFFD based on the RP concept is designed. RFFD firstly tackles the curse of dimensionality in which each image is projected onto a

$m$ -dimensional vector with a randomly generated projection matrix  $R$  from a zero-mean normal distribution. Each row of the transformation random matrix is  $l_2$  normalized. RFFD aims at minimizing the correlation between different classes while maximizing the correlation within-classes. It preserves the discriminative properties of the input data.

Algorithm (4) presents the RFFD. It looks for the best projection matrix  $R$  and the best dimension of projection  $m$  that preserve the structure and the reconstructive properties of the original data. The intuition behind this algorithm is as follows: since  $R$  is generated randomly, it is not guaranteed to preserve its relevant structure.

First of all, we have to ensure that the new low-dimensional vectors in  $A_i$ ,  $i = [1, \dots, m]$ , are not full of zeros otherwise bad quality features are generated. The quality of the projected matrix  $A^{m \times N}$  (Algorithm (4) step iii) is checked by thresholding every column vector in  $A_i$ . If the norm of  $A_i$  is smaller than a given threshold, it is considered as a bad feature vector and it has to be refined by generating another random matrix. Once a good data lower-dimensional data  $A^{(m,N)}$  is obtained,  $R$  is considered as a good random transformation matrix.

Moreover, the quality of the features vectors in  $A_i$  can vary from two different transformation matrices  $R$  and  $R'$ . We aim to pick out  $R$  or  $R'$  that induces the most discriminability between classes. To asset this property, selecting the best  $R$  among a set of  $R$ 's is then important and it is done as shown in the Algorithm (4) steps b and c.

In addition, selecting the appropriate dimension  $m$  that preserves the discriminative properties of the original data with a minimal distortion has a great effect on the final recognition rate. Therefore we deploy a line search strategy as shown in Algorithm (4) steps d and 2.

Finally, the projected data obtained as the output of the RFFD process are used to initialize the dictionary that is required for the sparse representation process. This step is important to induce sparsity during the learning process by initializing the dictionary with atoms that are highly informative and that have maximum separation between multiple classes.

### 3.3. A Mid Level Feature Encoding Based on Sparse Representation for FER 65

---

**Algorithm 4:** Random Face Feature Descriptor Algorithm.

---

**input :**

- $X \in R^{d \times N}$  : is the input matrix, where each column represents one sample.  $d$  is the original dimension of the image and  $N$  is the total number of samples.
- $M = [m_1, m_2, \dots, m_{dd}]$ : a list of possible desired lower dimensions.
- $rn$ : is the desired number for generated random matrices.

1. **for**  $m$  *in*  $M$  **do**

(a) **while**  $j$  *in*  $rn$  **do**

- i. Generate random matrix  $R_j^{[m \times d]} \in N(0, 1)$  and  $l_2$  normalized columns;
- ii. Compute  $A_{[m \times N]} = R_j^{[m \times d]} \cdot X_{[d \times N]}$ ;
- iii. Check the norm in  $A_{[m \times N]}$  based on a specific threshold;
- iv. If the norm of  $A_{[m \times N]}$  above the threshold, add  $A_{[m \times N]}$  to a list  $L_m$ ;

**end**

(b) **for**  $A$  *in*  $L_m$  **do**

- i. Apply Linear SVM over the obtained  $A$  using the cross validation set.
- ii. Store the recognition rate;

**end**

(c) Pick out the best  $A$  among  $A$ 's in  $L_m$  that achieves the highest classification accuracy rate;

(d) Add the best  $A$  and its classification accuracy rate to a list  $L_{mR}$ ;

**end**

2. Pick out from  $L_{mR}$  the  $A$  that reaches the highest accuracy rate and the corresponding best transformation matrix  $R_j$ ;

**output:**

- Projected data  $A_{[m \times N]}$  from the best  $R$  and  $m$ .
  - Best projection matrix  $R$  that generates the discriminative features.
  - Optimal lower dimension  $m$ .
- 

#### 3.3.3.2 Dictionary Refining and Sparse Coding

The second step of our framework as shown in Figure 3.11 is firstly to refine the pre-trained dictionary to sparsify the images via the K-SVD algorithm. Secondly it aims at deriving the sparse code associated to each signal by solving an  $l_0$ -norm regularization to enforce sparsity by using an approximate sparse reconstruction algorithm via OMP.

1. **Sparse Coding:** Keep the pre-trained dictionary  $D$  fixed and estimate  $X$  by solving (eq. 3.12):

$$\underset{X}{\operatorname{argmin}} \{ \|Y - DX\|_F^2 \} \quad \text{such that} \quad \|\mathbf{x}_i\|_0 \leq L, \{i\}_1^N.$$

The sparsity level  $L$  is a hyperparameter. The sparse representation  $X$  is optimized by using the OMP Algorithm (1). Compared with other alternative methods for sparse coding, a major advantage of the OMP is its simplicity and fast implementation.

2. **Dictionary Refining:** Keep the obtained sparse matrix  $X$  fixed and update the pre-trained dictionary  $D$  by solving (eq. 3.13):

$$\hat{D} = \underset{D}{\operatorname{argmin}} \|Y - DX\|_F^2$$

via K-SVD Algorithm (2) to better fit the data.

### 3.3.3.3 Classification Stage

In the last stage, the sparse matrix is used as feature vectors for classification. Our model trains a “Multinomial Linear Support Vector Machine” classifier [Vapnik 2013] for the purpose of facial expression recognition. We consider the linear SVM classifier among the others well-known classifiers, *i.e.* k-Means, Ada Boost and Decision Tree, since it shows the best results. In the training step, the training sparse matrix is used to learn a predictive model to recognize facial expressions. The test sparse matrix is used for generalization purpose to test the capability of the model to predict unseen facial expressions. A grid search is applied to find the best regularization parameter  $C$  to tune the classifier using the validation sparse matrix.

### 3.3.4 Experimental Setup and Analysis

A critical experimental evaluation of the proposed approach is presented. All the facial expression datasets presented in Section 3.1 that exhibit various emotions in different conditions starting from acted FEs to everyday natural and spontaneous FEs are used. Each database is considered alone and without any cross-database setting.

In this section, first we evaluate the performance of RFFD compared to PCA and RFS as an efficient dimensionality reduction technique. Then, we analyze the performance of RFFD under different target dimensions in terms of running time and classification accuracy. Second, we perform a control experiment to evaluate the effectiveness of the refining step of the dictionary learning, that is we first use the projected data as an initial dictionary and we perform sparse coding via OMP algorithm without any refining step and we check the accuracy rate. Then, we follow the same steps but with refining. Finally, we investigate the robustness of the proposed SPFER algorithm in dealing with posed and spontaneous facial expressions and we compare our model with state of the art algorithms mainly SRC and LC-KSVD algorithms.

**Dimensionality Reduction.** We run a primary study over all the databases in order to evaluate the performance of RFFD under different target dimensions  $m$  and along different random matrices. The reported classification results are the results obtained on the test sets while training a linear SVM classifier. We adopt the data protocol we discussed in Section (3.1). For the SVM classifier, there is an important hyper-parameter to set that is the weight of the regularization term. On the development dataset, we have noticed that using a fixed set of parameters does as well as cross-validating these parameters per class. Therefore, in all experiments, for simplicity, we have set the regularization weight of the SVM’s term to 0.001.

Figure 3.12 shows the performances of RFFD over acted databases under different target dimensions ranging between 600-feature points to 1000-feature points. For each target

### 3.3. A Mid Level Feature Encoding Based on Sparse Representation for FER 67

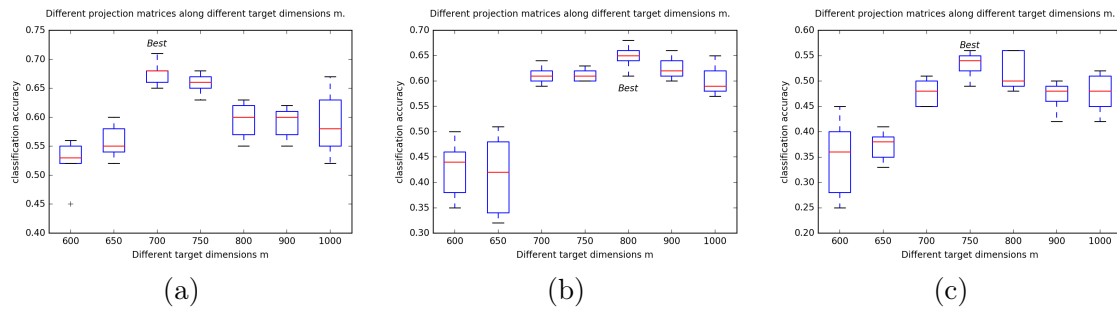


Figure 3.12: Performances of the linear SVM classifier for a variety of projected dimensions to evaluate RFFD over acted databases (a) JAFFE, (b) CK+ and (c) MMI.

dimension, five different random matrices are generated. Their performances are assisted by computing the classification accuracy rate. Figure 3.12(a) shows that the best model with the best projection matrix for the JAFFE database achieves an accuracy rate of 70% with a target dimension equal to 700-feature points. However for the CK+ database, the best model achieves a 68% accuracy rate under 800-feature points. As for the MMI database, the best model achieves a 56% accuracy rate under 750-feature points. We can see that different databases have different performances and different best target dimensionalities.

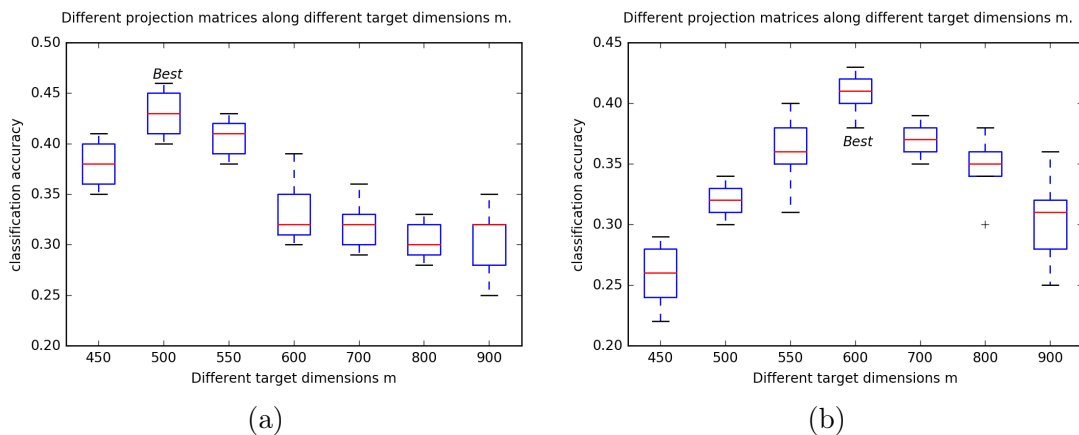


Figure 3.13: Performances of the linear SVM classifier for a variety of projected dimensions to evaluate RFFD over spontaneous databases (a) DISFA and (b) DynEvo.

Figure 3.13 shows the performances of RFFD over spontaneous databases under different target dimensions ranging between 450-feature points to 900-feature points. Figure 3.13(a) shows that the best model with the best projection matrix for the DISFA database achieves an accuracy rate of 45% with a target dimension equal to 500-feature points while for the DynEvo database with 600-feature points, the best model achieves 43% accuracy rate. The performances of RFFD dramatically decrease as spontaneous facial expressions are presented.

We have observed the performances of RFFD compared to different methods mainly PCA



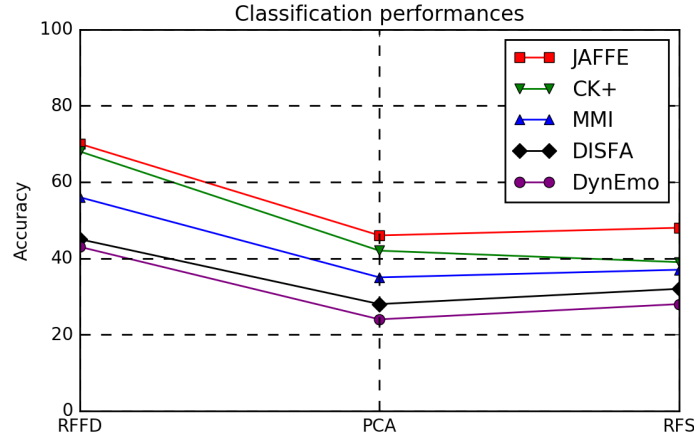


Figure 3.14: Different dimensionality reduction technique performances.

| Database        | JAFFE                         | CK+                             | MMI                             | DISFA                           | DynEmo                          |
|-----------------|-------------------------------|---------------------------------|---------------------------------|---------------------------------|---------------------------------|
| Dictionary size | $\mathbb{R}^{700 \times 143}$ | $\mathbb{R}^{800 \times 3500}$  | $\mathbb{R}^{750 \times 1800}$  | $\mathbb{R}^{500 \times 2500}$  | $\mathbb{R}^{600 \times 1500}$  |
| Description     | under-complete                | over-complete                   | over-complete                   | over-complete                   | over-complete                   |
| Sparse matrix   | $\mathbb{R}^{143 \times 143}$ | $\mathbb{R}^{3500 \times 3500}$ | $\mathbb{R}^{1800 \times 1800}$ | $\mathbb{R}^{2500 \times 2500}$ | $\mathbb{R}^{1500 \times 1500}$ |
| Sparsity level  | 50                            | 100                             | 150                             | 250                             | 200                             |

Table 3.2: Dictionary size for each of the facial expression database and the corresponding size of the sparse matrix.

and RFS as presented in Figure 3.14. For RFS, to choose the target dimensionality, we assign the one discovered by RFFD. The figure indicates that our initial results are promising as we achieve better separability between classes while having lower dimensional data.

**Dictionary Initialization.** Once the best projection matrix alongside the best target dimension  $m$  for each database is discovered using RFFD Algorithm (4), we use those transformation matrices to project any coming sample onto a lower dimensional subspace. The next step of our algorithm is to use the projected set of training examples to initialize the dictionary. The whole training set is used as dictionary atoms. Table 3.2 reports the type and the size of the dictionary  $\in \mathbb{R}^{m \times N}$  used for each database, where  $m$  is the projected data dimension and  $N$  is the number of training samples.

**Sparse Coding and Dictionary Refining.** We evaluate different versions of our SPFER algorithm to discover out the best sparsity level for each database and to highlight the importance of the dictionary refining step. First, in order to control the sparsity level as it is one of the important hyper-parameter for sparse coding, we conduct a study using the SPFER algorithm where we examine the recognition performance over a list having various sparsity levels. The best sparsity levels that yield to the best recognition rates are displayed in Table 3.2 and further analysis will be provided in the next paragraph. Second, in order to evaluate the importance of the dictionary refining step, we consider the initialized dictionary as final and we solve the problem of sparse coding via OMP Algorithm (1) without updating the

### 3.3. A Mid Level Feature Encoding Based on Sparse Representation for FER 69

dictionary. Then, the obtained sparse matrix  $X$  for each of the databases is used to train the linear SVM classifier.

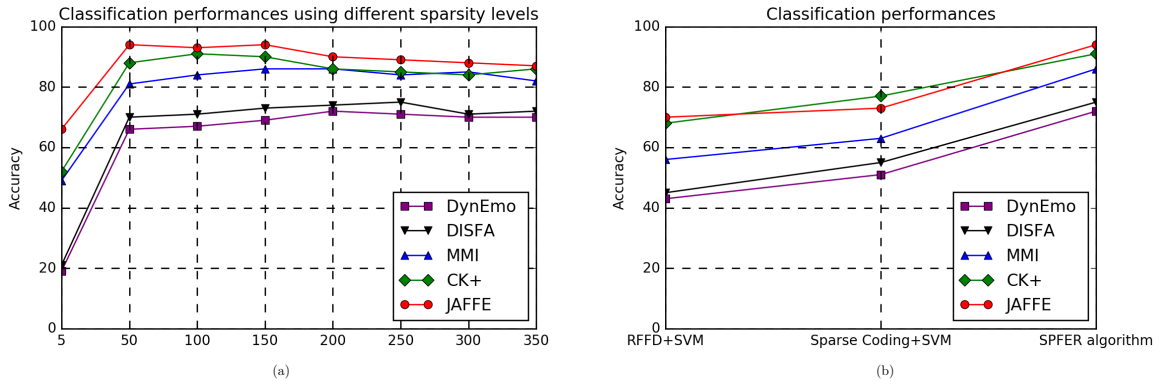


Figure 3.15: (a) Evaluation of the SPFER algorithm with different sparsity levels. The x-axis represents the sparsity level. (b): Evaluation of different variants of the SPFER algorithm.

**Experimental Evaluation Results.** Figure 3.15a displays the classification performances of the SPFER algorithm under different values of sparsity. We can notice that for the JAFFE database, a dictionary with a small number of atoms (sparsity level = 5) is capable to encode back the original signal while yielding to a recognition rate of 66%. The best sparsity level is achieved with  $L = 50$  and yields to a 94% accuracy rate. For the CK+ database, a 91% accuracy rate is achieved while being the maximum at sparsity level  $L = 100$ , however for the MMI database, with  $L = 150$ , a 86% is reported. Finally, for DISFA and DynEmo databases, 75% and 72% accuracy rates are reported at  $L = 250$  and  $L = 200$  respectively. Figure 3.15a shows that as the sparsity increases, a better performance is achieved with slight fluctuations. Considering the complexity of solving sparse coding, a compromise between setting  $L$  and the accuracy rate should be taken into consideration.

Figure 3.15b presents the classification performances of the SPFER algorithm with a dictionary initialized using the data projected via RFFD while solving the problem of sparse coding only without refining the dictionary and we refer to it as “Sparse Coding+SVM”. At this stage, we consider the best sparsity level reported in table 3.2. We compare its result to “RFFD+SVM”. Finally, we compare both results with the SPFER algorithm. For the JAFFE database, the initial classification performance using RFFD is 70%, the result improves slightly to 73% while considering the sparse codes as feature vectors. This slight improvement is due to the fact that sparse coding is done over an under-complete dictionary and thus it is harder to find a unique representation for samples coming from different classes. However, the classification performance did not drop as it suppose to behave because our dictionary atoms are still capable to approximately reconstruct back the original signal. For the CK+ database, sparse coding improves the classification performance over RFFD (68%) with 9%. However a further improvement is reported while including the dictionary refining step yielding to a 91% recognition performance. For the MMI database, using “RFFD+SVM”, a 56% accuracy rate is reported. An improvement to 63% accuracy rate is noticed using “Sparse Coding+SVM”. Beyond that, 86% recognition performance is reported for SPFER, thus having 30% improve-

ment over RFFD and 23% improvement over sparse coding with a fixed dictionary. Moreover, the recognition performance using the SPFER method yields to a notable improvement over spontaneous databases that include subtle facial expressions and being clutter to partial occlusion and various noise, having 75% and 72% accuracy rates over the DISFA and the DynEmo databases respectively. Our results show an enhancement for the discrimination of the sparse representation coefficients.

**Evaluation and Comparison to Existing Works.** In order to show the effectiveness of the proposed method, relative methods are utilized for comparisons mainly SRC and LC-KSVD as they are dedicated for image classification. The comparison results based on classification rates are presented. From Jiang et al. 2013 we adopt the default values  $\mu = 4.0$  and  $\eta = 2.0$  for the LC-KSVD, as they show good performances for all datasets. As a matter of fact, Jiang et al. 2013 shows that the effects of parameter selection of  $\mu$  and  $\eta$  induce very slight changes on the accuracy rate. We use the same data protocol as for the SPFER algorithm. Table 3.3 lists the average performances based on all datasets in our experiments. The results show the accuracies of the SPFER rank first among the SRC and the LC-KSVD whatever the database is.

| Method\Database | JAFFE | CK+ | MMI | DISFA | DynEmo |
|-----------------|-------|-----|-----|-------|--------|
| SPFER           | 94    | 91  | 86  | 75    | 72     |
| LC-KSVD         | 78    | 90  | 86  | 70    | 66     |
| SRC             | 81    | 88  | 84  | 71    | 65     |

Table 3.3: Recognition rates in % over all databases using state of the art methods.

### 3.3.5 Conclusion

As the main concern of the current chapter is the classification performances of different feature representation levels in predicting emotions via facial images, we gather in Table 3.4 the results of ImpBoVW and SPFER for comparison purposes. We can observe that mid-level features can achieve better classification performances in predicting emotions over both acted and spontaneous databases whether basic or non-basic expressions. Table 3.5 shows some failure and success cases associated with low level and mid level features over spontaneous databases mainly. We can see that any variation in a facial image regarding feature subtlety, hand gesture over face, or a subject wearing glasses, can affect the prediction quality negatively when features are represented using a low level representation. However, our results show that mid-level features are robust to such variations to a certain degree.

| Method\Database | JAFFE | CK+ | MMI | DISFA | DynEmo |
|-----------------|-------|-----|-----|-------|--------|
| SPFER           | 94    | 91  | 86  | 75    | 72     |
| ImpBoVW         | 91    | 86  | 81  | 64    | 62     |

Table 3.4: Recognition rates in % over all databases using low and mid level features.

### 3.3. A Mid Level Feature Encoding Based on Sparse Representation for FER 71





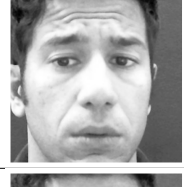


| Class        | Image   | ImpBoVW Prediction | SPFER Prediction |
|--------------|---|--------------------|------------------|
| amusement    |    | curious            | amusement        |
| curious      |    | fright             | curious          |
| disappointed |    | astonished         | disappointed     |
| disgust      |   | angry              | disgust          |
| fear         |  | sad                | fear             |
| sad          |  | fear               | sad              |
| happy        |  | neutral            | happy            |

Table 3.5: Low-level versus mid-level feature representation power over class prediction. The first three rows correspond to the DynEemo database. The last four rows correspond to the DISFA database.

### 3.4 Hierarchical Spatial and Spatio-Temporal Feature Encoding Based on Deep Neural Network for FER

In this dissertation, despite the efforts made in developing various methods for FER, further improvements are needed regarding the robustness of the feature encoding and representation. Especially, when it is applied to images or sequences that consider partial wild setting, *i.e.* spontaneous basic and non basic expressions with head occlusion and movements, in which the results are still not good enough. Deep neural network have shown to outperform traditional methods, such as those using hand-crafted low-level features or sparse mid-level features combined with decoupled classifiers. DNNs based methods [Mollahosseini et al. 2016, Mollahosseini et al. 2016a] reported promising results in FER and have shown the ability to extract more discriminative FEs features while training jointly a classifier and a feature extractor at the same time. Motivated by the success of deep learning based methods, we decided to build a neural network model for automatically extracting hierarchical features for FER. We want to study the power of the features constructed hierarchically over previous low-level and mid-level features for tackling challenging situations.

Moreover, a FER system can rely on using either spatial features from static images or by considering spatio-temporal information from an image sequence. In this section, we also want to study the impact of temporal information on the performance. We argue that considering temporal relations of consecutive frames would be essential for recognizing subtle changes in the appearance of FEs, to overcome some extrinsic factors such as head movements and occlusions. Therefore, we design a neural network that takes into consideration local and global spatio-temporal features in its complete version but that can also be reshape to consider spatial features only for comparison with ImpBoVW and SPFER.

Building a dynamic FER model is very challenging because most of the time the entire morphological facial event of emotion from the onset to the offset is very quick and includes noise [Mohammad Mahoor 2017]. Moreover, emotions as any action, can both speed-up or slow-down. Therefore it is important to incorporate information from close frames (short-term dependencies) and from far frames (long-term dependencies) to summarize the segment context despite of its time variations. In this thesis, we argue that there are two categories of spatio-temporal features present in emotional videos: 1) *Local spatio-temporal features* (short-term information) that stand out to the fine-grained motion information characterizing morphological changes occurring at small intervals throughout; 2) *Global spatio-temporal features* (long-term information) in which full semantic information about the emotional state along the entire sequence (global setting among the whole segments) is taken into account. The integration of local and global spatio-temporal information is crucial to summarize what can be a rather elaborate sequence of morphological segments.

We propose an approach to model local and global hierarchical spatio-temporal information behaviors based on 3D-CNN and ConvLSTM networks. We show that the task of dynamic emotional behaviour prediction benefits from a combination of global and local spatio-temporal learned features in order to effectively model the discriminative features of visual appearance

and its motion. The proposed model is capable of capturing subtle spatial and temporal variations of FEs and of learning in an end-to-end fashion. We give an intuition about tuning the filter hyper-parameters through visualizing the weight histograms and we give a deep insight into the network feature maps to provide better understanding.

First, since we put the focus on studying the power of spatio-temporal hierarchical features, we also investigate the power of spatio-temporal low- and mid-level features. For low spatio-temporal features, we use 3D descriptors alongside temporal BoVW and for mid spatio-temporal features, we use a pre-trained neural network to extract spatial features and we model the temporal links by employing a recurrent layer.

Second, in order to compare the power of the spatial features only, we derive a spatial neural network that shares a partially similar architecture as the temporal one. For clarity, we will present first the dynamic complete deep neural network architecture, followed by its 2D variant architecture and its results.

### 3.4.1 Hierarchical Spatio-Temporal FER Model

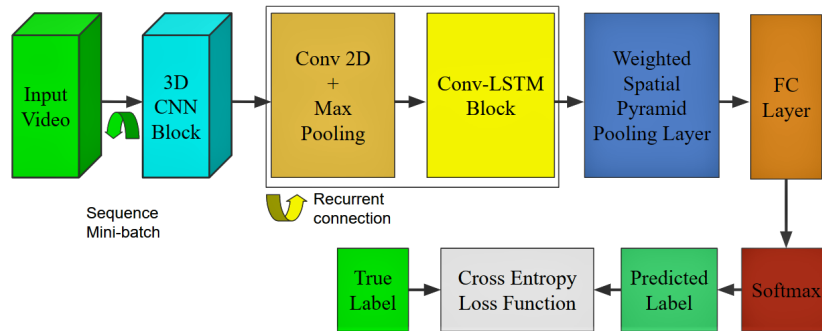


Figure 3.16: General model architecture for learning local and global hierarchical spatio-temporal features for FER.

An end-to-end deep learning architecture for FER in video domain is proposed. The general model architecture is presented in Figure 3.16. The proposed model firstly encodes local variations caused by expressions by learning local spatio-temporal features through a 3D-CNN, so that any subtle or fast change that the local neighboring frames may convey is captured. Local feature learning is a very important aspect since it helps to catch the transition of information in space and time and its speed. The purpose of our local spatio-temporal representation is to capture features that are hidden in the global representation of a long input image sequence.

However, in order to reveal patterns and features which provide a contextual and semantic representation for the entire sequence, a global analysis is required. Therefore, global spatio-temporal features are learned from these local features using a ConvLSTM in order to encode the global structure where we can derive a meaningful semantic representation that focuses only on the relevant FE features in the whole input sequence.

The layer between the 3D-CNN and the ConvLSTM is referred to as the transition layer. It is designed for a down-sampling purpose, in order to change the size of feature maps via a 2D Convolutional Neural Network (2D-CNN) and max pooling. Down-sampling at this level enhances translation invariant representation since the resolution of the feature maps is reduced by maxpooling over local neighborhood on the feature maps in the previous layer.

Yet, to facilitate a scale-invariant representation, a Spatial Pyramid Pooling Network (SPP-Net) [He et al. 2015a] follows the output of the ConvLSTM layer and is used to extract features at multi scales. It aggregates multi-scale local features over fixed sub-regions. Finally, an FC layer and a multinomial logistic regression classifier are used to estimate the class probabilities of the distinct FEs. In the following, we explain each of the aforementioned modules.

### 3.4.1.1 The 3D-CNN Block: Learning Local Spatio-Temporal Features

In general, local spatio-temporal features are extracted directly from image sequences to capture shape characteristics and to consider consecutiveness of motions. These features provide a relatively independent representation within emotions with respect to their spatio-temporal shifts and scales. The process of learning and extracting local spatio-temporal information includes the selection of spatio-temporal locations and scales in video sequence and the capture of shape and motion in the neighborhoods of the extracted points. To extract spatio-temporal features, the local cuboid is one of the common methods. Local features are known to provide robustness against extrinsic variations.

We assume that the global FE deformations can be characterized by local motions and therefore by local spatio-temporal features. Hence, it is important first to capture these local spatio-temporal features. We design a 3D-CNN architecture to extract discriminative features along both the spatial and the temporal dimensions by performing 3D convolutions, thereby capturing the motion information encoded in multiple adjacent frames. Thus, we identify repetitive but also sequential patterns with subtle and fast changes. The 3D-CNN layer generates multiple channels of information from the input facial frames.

**From 2D to 3D Convolutional Neural Networks.** In the following we provide an intuition about how the 3D-CNN works [Ji et al. 2013] starting from a 2D-CNN. Convolutional layers operate in a sliding-window manner on feature maps from the previous layer  $l - 1$  in order to extract spatial features from local neighborhoods using a 2D kernel convolution which represents the spatial arrangement of the activations. Then an additive bias is applied and the result is passed through an activation function usually the Rectified Linear Unit (ReLU). Formally, the value of an unit at position  $(x, y)$  in the  $j_{th}$  feature map in the  $i_{th}$  layer, denoted as  $v_{ij}^{xy}$ , is given by:

$$v_{ij}^{xy} = \sum_m \sum_{p=0}^{P_i-1} \sum_{q=0}^{Q_i-1} w_{ijm}^{pq} v_{(i-1)m}^{(x+p)(y+q)} + b_{ij}, \quad (3.19)$$

where  $P_i$  are  $Q_i$  are the sizes along the height and the width of the 2D kernel. The bias is  $b_{ij}$  for this feature map,  $m$  indexes over the set of feature maps in the  $(i - 1)$ th layer connected to the current feature map.  $w_{ijm}^{pq}$  is the  $(p, q)$ th value of the kernel connected to the  $m$ th feature map in the previous layer.

A 2D-CNN architecture is constructed by stacking multiple layers of convolution. The parameters of the CNN, such as the bias  $b_{ij}$  and the kernel weight  $w_{ijm}^{pq}$ , are usually learned.

The 3D convolution is achieved by convolving a 3D kernel to the cube formed by stacking multiple contiguous frames together. By this construction, the feature maps in the convolution layer are connected to multiple contiguous frames in the previous layer, thereby capturing motion information. Formally, the value at position  $(x, y, t)$  in the  $j$ th feature map in the  $i$ th layer, denoted as  $v_{ij}^{xyt}$  is given by:

$$v_{ij}^{xyt} = \sum_m \sum_{p=0}^{P_i-1} \sum_{q=0}^{Q_i-1} \sum_{r=0}^{R_i-1} w_{ijm}^{pqr} v_{(i-1)m}^{(x+p)(y+q)(t+r)} + b_{ij}, \quad (3.20)$$

where  $R_i$  is the temporal dimension of the 3D kernel.  $w_{ijm}^{pqr}$  is the  $(p, q, r)$ th value of the kernel connected to the  $m$ th feature map in the previous layer. A 3D convolutional kernel can extract features from the frame cube since the kernel weights are shared across the entire cube.

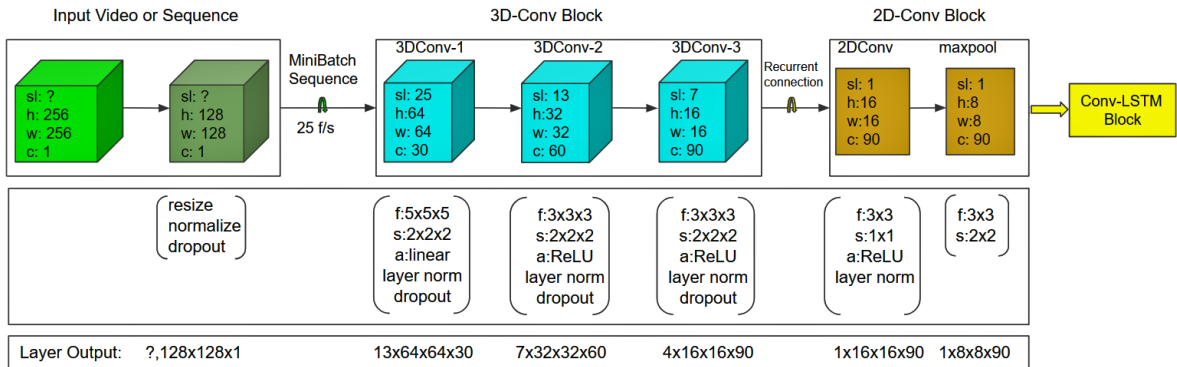


Figure 3.17: 3D Convolutional block architecture for encoding local variations. In this figure,  $sl$ : sequence length,  $f$ : feature maps,  $s$ : stride,  $a$ : activation,  $h$ : height,  $w$ : width,  $c$ : channels.

**3D-CNN Architecture for Extracting spatio-temporal Features.** Based on the 3D convolution described above, we devise the 3D-CNN architecture represented in Figure 3.17. The input video is re-sized to  $128 \times 128$  and only the gray channel is considered. The sequence length of the video is variable from one input to another (denoted as “?” in Figure 3.17). Thus, to overcome this problem, we simply fix all the inputs to have the same sequence length by concatenating at the beginning of each sequence the neutral face part. As we discussed earlier, we mostly care about the morphological segments and we hope that the neutral part no matter how much it lasts in a sequence will be ignored. The sequence length can be dramatically high (e.g., 1000 frames). Therefore, in general we use 25 frames as a mini-batch sequence to process out of the entire sequence length, thus we loop several times over the entire sequence



without overlapping. The weights are shared across the entire sequence, and thus loop, in order to model spatial and temporal dependencies. Otherwise we may lose the semantic and contextual information stored in the sequence.

The 3D convolutional block is designed with three cascade layers, the first layer is kept with a linear activation function and with a filter size  $5 \times 5 \times 5$ . The first dimension represents the temporal information while the last two dimensions represent the spatial information. This layer is associated with a stride by two in order to reduce the spatial and the temporal dimensions. The number of filters is 30. Each layer is followed by a layer normalization [Ba et al. 2016], which aims at normalizing the activities of the neurons. Then, a dropout [Srivastava et al. 2014] is used as the main regularization approach to reduce overfitting and to enhance generalization. Dropout essentially means randomly omitting the neurons of a layer by a certain probability. It works equivalently as adding random noise to the representation or performing model averaging. Later, more details about the choice of the filter hyperparameters will be given. Then the same topology as the one of the first layer is considered for the second and the third layers. The only difference is that the filter size is reduced to  $3 \times 3 \times 3$  and the number of feature maps is increased to 60 and 90. In addition, we use ReLU as the activation function.

Initially, if the input at a time-step  $k_1$  is  $25 \times 128 \times 128 \times 1$ , after the first pass over the 3D-Conv block, the output of the third 3D-Conv layer will be  $4 \times 16 \times 16 \times 90$ . The current output is then fed to the transition layer (2D-Conv) for down-sampling, resulting an output of size  $4 \times 8 \times 8 \times 90$ , followed by the ConvLSTM block. The ConvLSTM block accepts one time step at a time being sequential ( $1 \times 8 \times 8 \times 90$ ) and its parameters are shared across the entire new sequence time-steps  $k_f' = 4$ . Hence, the 3D-Conv Block allows to efficiently handle large input sequences and to produce a compact representation that encodes the local variations between consecutive frames.

#### 3.4.1.2 The ConvLSTM Block: Learning Global Spatio-Temporal Features

Facial expression video data contain complex emotional states that evolve over time. Thereby, it is most likely that these emotional states being potentially fused incoherently due to the vast quantity of information in the entire sequence. Working with such data requires to model their dynamic spatio-temporal structure. Therefore, we integrate a 3D-CNN with a ConvLSTM to avoid temporal collapse and to go beyond local spatio-temporal modeling while providing a contextual representation of the entire sequence. Herein, we detail the process of capturing global spatio-temporal features.

**From Multi-layer Perceptron to Convolutional Long Short Term Memory.** MLP is a computational model composed of a series of interconnected computational nodes (called neurons or units). These interconnections are weighted and arranged using a fully-connected topology, in which each unit in layer  $l+1$  is connected with every unit in layer  $l$ , forming layers of nodes in order to process information. The nodes of the layers are followed by a non-linear operation using ReLU, to create a decision boundary for the input data by projecting it into a

space where it becomes linearly separable. Units of MLP are defined in terms of the following function:

$$\mathbf{a}^{l+1} = \text{ReLU}(W^l \mathbf{a}^l + \mathbf{b}^l), \quad (3.21)$$

where:

- $\mathbf{a}^l$  denotes the level of response for the units in layer  $l$ , with  $a_i^l$  the activation of the unit  $i$  in layer  $l$ .
- $W$  is a weight matrix, where  $W_{ij}^l$  represents the parameter associated with the connection between unit  $j$  in layer  $l$ , and unit  $i$  in layer  $l + 1$ .
- $\mathbf{b}^l$  is the bias associated with units in layer  $l$ .

**Recurrent Neural Networks.** RNNs make use of a recurrent connection in every unit (or neuron). The activation is fed back to itself with a weight and a unit time delay. This recurrent connection provides each unit with a memory (hidden value) of past activations. This type of networks has Turing capabilities [Siegelmann and Sontag 1991] and, thus, is in principle suited for learning the temporal dynamics of sequential data.

Given a temporal input sequence  $\mathbf{a}^l = (a_1^l, a_2^l, a_3^l, \dots, a_T^l)$  of length  $T$ ,  $a_{t,i}^l$  being the activation of the unit  $i$  in hidden layer  $l$  at time  $t$ . An RNN maps it to a sequence of hidden values  $\mathbf{h}^l = (h_1^l, h_2^l, h_3^l, \dots, h_T^l)$  and outputs a sequence of activations  $\mathbf{a}^{l+1} = (a_1^{l+1}, a_2^{l+1}, a_3^{l+1}, \dots, a_T^{l+1})$  by iterating the following recursive equation:

$$\mathbf{h}_t^l = \tanh(W_{xh}^l \mathbf{a}_t^l + \mathbf{h}_{t-1}^l W_{hh}^l + \mathbf{b}_h^l), \quad (3.22)$$

where:

- $\mathbf{b}_h^l$  is the hidden bias vector.
- $\tanh$  is the hyperbolic tangent function.
- $W_{xh}^l$  is the input-hidden weight equivalent to  $W^l$  defined for the MLP,  $W_{hh}^l$  is the hidden-hidden weight matrix.

The activation for these recurrent units is defined by:

$$\mathbf{a}_t^{(l+1)} = \mathbf{h}_t^l W_{ha}^l + \mathbf{b}_a^l, \quad (3.23)$$

where:

- $W_{ha}^l$  denotes the hidden-activation weight matrix.
- $\mathbf{b}_a^l$  denotes the activation bias vector.

RNNs memory mechanism makes the learning challenging when dealing with real-world sequences [Gers et al. 2002] due to the problem of long term dependencies. RNNs are ca-

pable to extract semantic information only if the coming input has a short length sequence. Which means it is capable at correlating short term dependencies where the gap between the relevant information is small (*e.g.*, information between frames  $t_1$  and  $t_3$ ) while it will fail to extract salient information from long sequences (*e.g.*, information between frames  $t_1$  and  $t_{25}$ ). Unfortunately, as that gap grows (sequence length and its information dependencies), RNNs become unable to learn how to connect the information [Bengio et al. 1994], due to the problem of vanishing gradients.

**Long Short Term Memory.** Long Short Term Memory (LSTM) [Hochreiter and Schmidhuber 1997] is an extension of RNN, wherein it is associated with memory cells that store and output information, over time to explore long range dynamics, with non-linear gate units governing the information flow into and out of the cells. Therefore, that eases the learning of temporal relationships on long time scales and controls information flow so that the gradient is trapped into the cell. LSTM recursively maps the input representation at the current time step via a sequence of hidden states, and thus the learning process of LSTM is in a sequential manner the same as RNN. The input provided to an LSTM is fed into different gates that control which operation is performed on the cell memory. The memory cell  $c_t$  acts as an accumulator of the state information. The cell is accessed, written and cleared by several self-parameterized controlling gates. Every time a new input comes, its information is accumulated to the cell if the input gate  $i_t$  is activated. Also, the past cell status  $c_{t-1}$  could be forgotten in this process if the forget gate  $f_t$  is on. Whether the latest cell output  $c_t$  is propagated to the final state  $h_t$  is further controlled by the output gate  $o_t$ . Figure 3.18 shows the gating mechanism of write (input gate), read (output gate) or reset (forget gate). An LSTM cell is based on component-wise multiplications of the input, which define the behaviour of each individual memory cell. The activation of the LSTM units is calculated as for RNN (refer to eq. (3.22)). The major drawback of fully connected LSTM in handling spatio-temporal data is the usage of full connections in input-to-state and state-to-state transitions in which no spatial information is encoded.

**Convolutional Long Short Term Memory.** On the other hand, ConvLSTM [Xingjian et al. 2015] uses 2D-grid convolutions to leverage the spatial correlations of input data. Its convolutional structure in both the input-to-state and state-to-state transitions model the spatio-temporal links quite well. Formally, the inputs, the cell states, the hidden states and the gates of ConvLSTM are 4D tensors whose first dimension represents the time step, the second and the third are the spatial dimensions (height and width), and the last dimension is the feature map. The weight matrices here represent the 2d-convolutional kernels. The computation of the hidden value  $h_t$  of a ConvLSTM cell is updated at every time step  $t$ .

Figure 3.18 shows a diagram which explains how a ConvLSTM unit operates. To formally represent the ConvLSTM module, let “ $*$ ”, “ $\otimes$ ” and “ $\sigma$ ” represent respectively the convolutional operation, the Hadamard product and the sigmoid function. The ConvLSTM is formulated as:

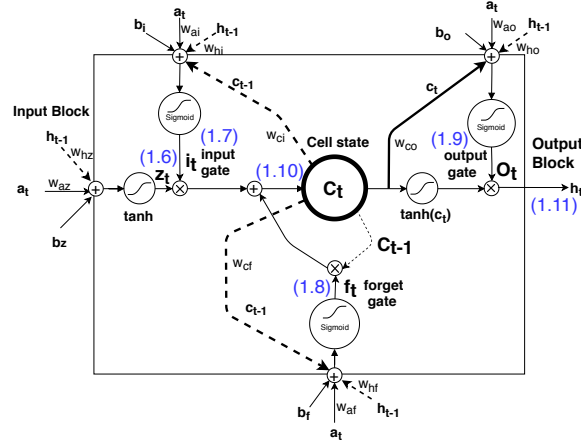


Figure 3.18: Convolutional Long Short-Term Memory Cell. The numbers refer to the corresponding equations. The inputs coming from different sources get convoluted with their filters, added up along with bias. The cell operates by learning gate functions that determine whether an input is significant enough to be memorized, to be forgotten or to be sent to the output. By using a gated way for sorting information over short or long time ranges, the discriminant spatio-temporal information is extracted.

$$z_t = \tanh(\mathbf{a}_t * W_{az} + \mathbf{h}_{t-1} * W_{hz} + \mathbf{b}_z) \quad (3.24)$$

$$\mathbf{i}_t = \sigma_i(W_{ai} * \mathbf{a}_t + W_{hi} * \mathbf{h}_{t-1} + W_{ci} \otimes \mathbf{c}_{t-1} + \mathbf{b}_i) \quad (3.25)$$

$$\mathbf{f}_t = \sigma_f(W_{af} * \mathbf{a}_t + W_{hf} * \mathbf{h}_{t-1} + W_{cf} \otimes \mathbf{c}_{t-1} + \mathbf{b}_f) \quad (3.26)$$

$$\mathbf{o}_t = \sigma_o(W_{ao} * \mathbf{a}_t + W_{ho} * \mathbf{h}_{t-1} + W_{co} \otimes \mathbf{c}_t + \mathbf{b}_o) \quad (3.27)$$

$$\mathbf{c}_t = z_t \otimes \mathbf{i}_t + \mathbf{c}_{t-1} \otimes \mathbf{f}_t \quad (3.28)$$

$$\mathbf{h}_t = \tanh(\mathbf{c}_t) \otimes \mathbf{o}_t, \quad (3.29)$$

where  $z$ ,  $i$ ,  $f$ ,  $o$  and  $c$  are respectively the input block, input gate, forget gate, output gate and cell activation 4D tensors, all having the same size as the tensor  $\mathbf{h}_t$  defining the hidden value. The term  $\mathbf{a}_t$  is the input of a memory cell layer at time  $t$ .  $W_{az}$ ,  $W_{ai}$ ,  $W_{hi}$ ,  $W_{af}$ ,  $W_{hf}$ ,  $W_{cf}$ ,  $W_{ac}$ ,  $W_{hc}$ ,  $W_{ao}$ ,  $W_{ho}$  and  $W_{co}$  are the weight matrices, with subscripts representing from-to relationships. For example,  $W_{ai}$  being the input-input gate matrix connecting  $a_t$  to  $i_t$  as shown in figure 3.18, while  $W_{hi}$  is the hidden-input gate matrix, and so on. The bias vectors are:  $\mathbf{b}_z$ ,  $\mathbf{b}_i$ ,  $\mathbf{b}_f$ ,  $\mathbf{b}_c$  and  $\mathbf{b}_o$ . The layers' notation has been omitted for clarity. The activation of the ConvLSTM units is calculated as for the RNN represented in eq. (3.22). By referring to eq. (3.29), the current cell state is updated based on the current filtered input added up with its previous filtered cell state. Therefore, the current cell state aggregates useful space-motion information sequentially along the time input space. In such a way, the last state is considered as the final representation that summarizes the contextual information of the entire sequence.

**ConvLSTM Architecture for Extracting Global Hierarchical Spatio-Temporal Features.** Based on the ConvLSTM cell described above, we devise a recurrent architecture, as

shown in Figure 3.19, composed of two ConvLSTM cells. The input for this block is one step at a time coming from the transition layer. Each input has a shape of  $1 \times 8 \times 8 \times 90$ , which denotes the time step, the spatial dimensions and the number of channels respectively. Each ConvLSTM cell has a filter size of  $3 \times 3$ , with 45 feature maps. The cell state  $c_t$  of the ConvLSTM-1 is considered as an input for the ConvLSTM-2. A layer normalization is done over the ConvLSTM-1 cell and dropouts with 65%. The output  $h_t$  of both ConvLSTM cells are concatenated, where each ConvLSTM outputs a shape of  $1 \times 8 \times 8 \times 45$ . The final output has a shape of  $1 \times 8 \times 8 \times 90$ . The output of the last step of each ConvLSTM cell is considered only at the end of the training, since it is an accumulation of information acquired from the entire sequence.

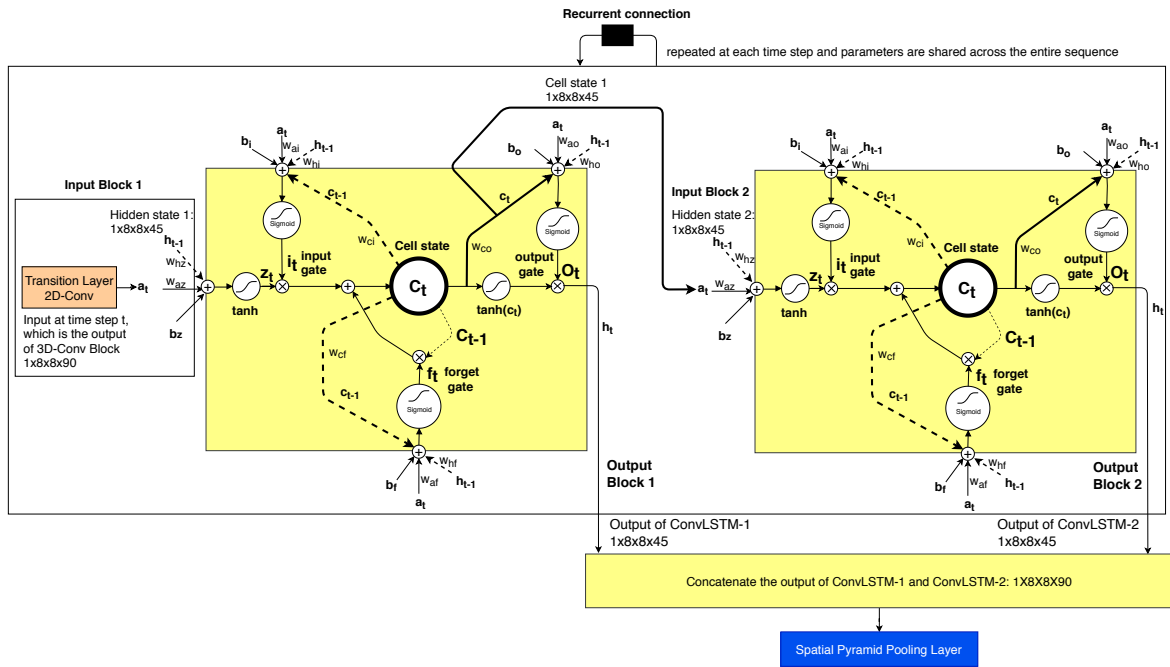


Figure 3.19: ConvLSTM architecture for extracting long term dependencies.

### 3.4.1.3 The Weighted Spatial Pyramid Pooling Layer (SPP-layer): Non-variant Feature Representation

While dealing with FEs, finding the appropriate representation requires presenting the network many instances in all its natural variations. Thereby the deep representation captures distinctive FE features in space and time. The spatial sources of natural variations in faces are: the location, the viewpoint (angles), and the size of the face.

The variations in location and angle are dealt straightforward by our algorithm due to the weight sharing employed in the 2D and 3D convolution layers as well in the ConvLSTM layers, in addition to creating filters that respond to angle-invariant features. However, the complication arises with face size variations along the sequence due to head gesture and varying distances from the camera. The latter may induce a blurring effect which leads to variations

in image resolution.

Fourier theory [Masters et al. 2009] has formalized the relations between image resolution, object size, and image scale: the fine details of an image are captured by high spatial frequencies, whereas the coarse visual structures are captured by low spatial frequencies. In order to encourage scale-invariant representation, spatial pyramid pooling layers on top of the output of the ConvLSTM layers are used to extract features at multi scales. SPP-Net is an extension of spatial pyramid matching [Lazebnik et al. 2006b] where the features in arbitrary regions (sub-images) are pooled. It basically partitions the image into divisions from finer to coarser levels, and aggregates local features in them. The SPP layer allows to increase scale-invariance and reduces the risk of overfitting. Typically, the SPP-Net treats the features found either at finer resolutions or at coarser resolutions in a same way. However, in this work, we adopt the original weighting scheme from the spatial pyramid kernel defined in [Lazebnik et al. 2006b] and we associate it to SPP-Net, wherein the features found at finer resolutions are more highly weighted than the features found at coarser resolutions. Formally, the weight for these multi-scale spatio-temporal features is defined as:

$$\frac{1}{2^L} M^0 + \sum_{l=1}^L \frac{1}{2^{L-l+1}} M^l, \quad (3.30)$$

where  $L$  is the total number of levels and  $l$  is the current level.  $M^L$  is the initial feature map at fine resolution, while  $M^0$  is the feature map at coarse resolution.

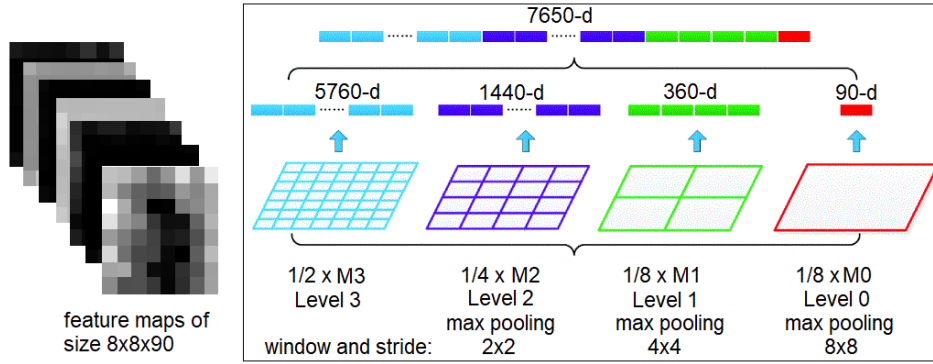


Figure 3.20: Weighted spatial pyramid pooling layer.

Figure 3.20 represents a sequence of grids at resolutions  $0, \dots, L$ , such that the grid at level  $l$  has  $2^l$  cells along each dimension. In our architecture, the output of the ConvLSTM Block has a shape of  $1 \times 8 \times 8 \times 90$ . Therefore we will obtain a weighted final spatio-temporal feature vector of size 7650-d, since by performing max-pooling with a window size  $8 \times 8$  and a stride by  $8 \times 8$ , we obtain a feature vector of size 90-d (as represented in Figure 3.20). Then, by performing max-pooling with a window size  $4 \times 4$  and a stride by  $4 \times 4$ , we obtain a feature vector of size 360-d. In the same way, by performing max-pooling with a window size  $2 \times 2$  and a stride by  $2 \times 2$ , we obtain a feature vector of size 1140-d. Finally, at level three, by considering the initial representation with size  $1 \times 8 \times 8 \times 90$ , we obtain 5760-d. By

concatenating all these feature vectors, we obtain a non-variant representation of size 7650-d.

### 3.4.1.4 Classification Layer: FC layer, Read Out Layer and Softmax Classifier

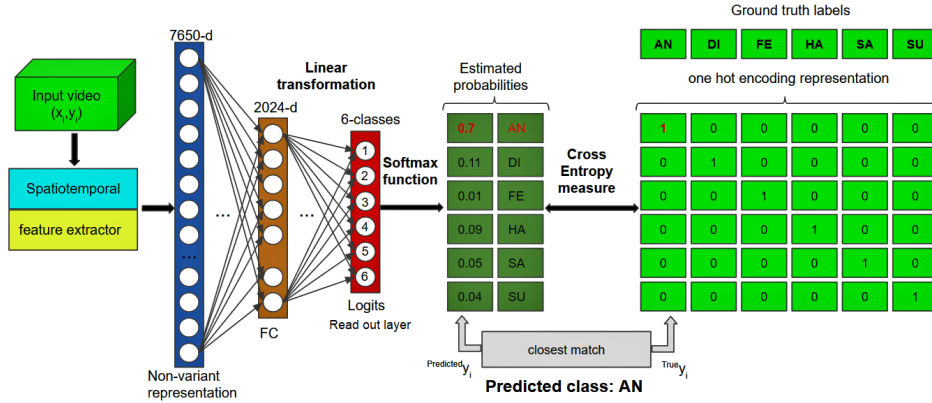


Figure 3.21: Classification stage: mapping the final representation into a probability distribution. The system outputs one label for every input sequence.

Once we obtain a non-variant representation (multiscale spatio-temporal feature vector of size 7650-d), it is fed to a FC layer of 2024 hidden units with a nonlinear activation function such as ReLU (refer to eq. (3.21)). In fully connected fashion, units between two adjacent layers are fully pairwise connected, while units within a single layer do not share connections.

An FC layer is known to be a universal function approximator with the ability of approximating more abstract representations of the non-variant representation. It can be a deep model itself, which is consistent with the spirit of feature re-use [Bengio et al. 2013]. It aims at allowing the flow of information between each feature point of the non-variant representation and the FC layer. Therefore, the final decision is based on every single entry of the final spatio-temporal feature vector. Moreover, by setting the FC layer to have 2024 units (less than its inputs), we learn and select discriminative spatio-temporal features. On the output of the FC layer, a dropout with probability of 0.5 is applied and then it is fed into a multinomial logistic classifier. An illustration is presented in Figure 3.21.

The classifier is composed of a read out layer and a softmax function. The read out layer has a FC topology. Its units are equal to the total number of classes  $C$ . Its activation is linear and it performs a linear transformation in order to obtain  $C$  distinct class logits or scores. These logits are then passed to a softmax function:

$$S(\text{logit}_i) = \frac{e^{\text{logit}_i}}{\sum_{i=1}^C e^{\text{logit}_i}},$$

in order to estimate a probability distribution by measuring the relationships between the categorical dependent variables. The index of the output units is denoted as  $i$ .

The output of the softmax function  $y_i^{Predicted}$  represents the predicted probability for the  $i_{th}$  class given an input video  $x_i$  with the true label  $y_i^{True}$ . Given the true probability distribution  $y_i^{True}$  of an input video  $x_i$  and the estimated probability distribution  $y_i^{Predicted}$  of the current model, a Cross Entropy measure is used to define the model loss function to minimize and it is represented as:

$$D(y_i^{Predicted}, y_i^{True}) = - \sum_i^C y_i^{True} \log(y_i^{Predicted}).$$

The loss function thus is:

$$loss = \frac{1}{N} \sum_{j=1}^N D_j,$$

where  $N$  represents a mini-batch of input video samples on which we average over. The final goal is to minimize the loss function by adjusting the parameters of the model (W and b) so that the output of the classifier is close to the label of each example.

### 3.4.2 Network Training and Initialization

The proposed model is implemented in Python using Tensorflow [Abadi et al. 2016]. The model training and classification are run over two GPUs Nvidia Titan X with 6 GB and 12 GB RAM respectively. Model parameters (weights and biases) are updated on CPU synchronously by waiting for all GPUs to finish processing a batch of data. The model is trained in a fully supervised way by backpropagating the gradients from the softmax layer through the convolutional layers. The network parameters are optimized by minimizing the cross-entropy loss function using the Adaptive Moment Estimation (ADAM) optimizer [Kingma and Ba 2014]. The advantages of the ADAM optimizer is that it allows setting the learning rate automatically based on the model weight update history.

For the sake of efficiency, when training and testing, data are segmented in mini-batches with a size of 128 emotional videos. Using this configuration, an accumulated gradient for the parameters is computed after every mini-batch. The model is trained with a learning rate of 0.0001 for the first 70000 iterations. After 70000 iterations, the learning rate is dropped to 0.00001 to insure the stabilization of weights update while convergence. The total number of epochs is set to 75000.

As we discussed earlier, a dropout operator is introduced on the initial input as a way to simulate noise and as a way of sampling from the original data. The dropout operator on the network layers is also considered as a form of regularization. During this training, the dropout operator sets the activation of randomly selected units to zero with a probability range between 0.5 and 1.0 (set randomly at every iteration). For the ConvLSTM-1 cell state, 65% of the neurons are omitted since we experience a better reduction of the variance (overfitting



reduction). For the 3D-Conv layers, since the filter is typically small (e.g,  $5 \times 5 \times 5$  filter resulting 125 units), it might hurt the learning procedure if the dropout is done with a high percentage, thus only 5% is applied. Finally, for the FC layer since the number of hidden units is relatively high, a dropout with 50% is applied.

Weights initialization is very important since if the weights in a network start too small, then the signal shrinks as it passes through each layer until it is too tiny to be useful. On the contrary, if the weights start too large, then the signal grows as it passes through each layer until it is too massive to be useful. Xavier initialization [Glorot and Bengio 2010] makes sure the weights are *just right*, keeping the signal in a reasonable range of values through many layers. In this work, we make the use of Xavier initializer.

The activation of the ConvLSTM cells is set to the Hyperbolic tangent (tanh) function, since it is bounded and its second derivative can sustain for a long range before going to zero. When using ReLU (unbounded) at this layer level, the model works only for the CK+ database while failing over the other databases. Obviously, our model could work over the CK+ database, because the sequences length is short and thus the gradient can sustain before going to zero, even if using ReLU as the non-linear function. However, using the tanh function for ConvLSTM layers, provides a good performance over all the databases with both short or long image sequences. The forget bias is set to 3.0. By default it is usually 1.0, but we experienced a better model behaviour with 3.0.

### 3.4.3 Data Augmentation for Facial Expressions Databases

In order to train a deep neural network without falling into the risk of overfitting and poor generalization problem, several approaches can be utilized during the training. The simplest is to add a regularization term on the norm of the weights. Another popular technique is to use dropout and normalization. Moreover, data augmentation is another technique to reduce overfitting on models where the amount of data can be increased using the information contained in training data. In this work, we perform data augmentation.

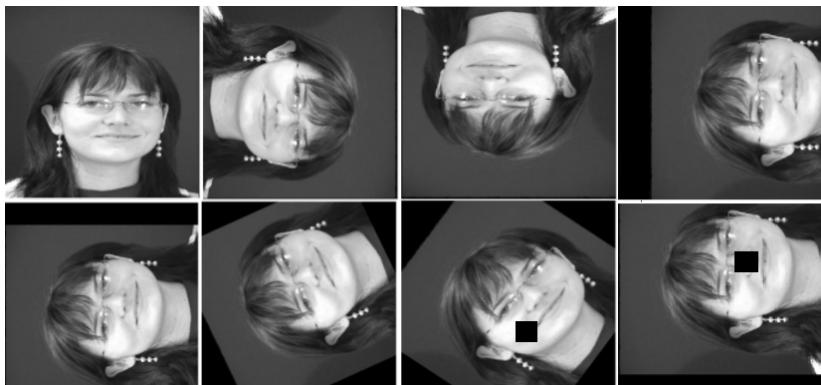


Figure 3.22: Augmenting image data by introducing geometric deformations.

**Data Augmentation.** Our inputs are sequences of frames that hold only one expression

label, thus the total number of training samples is considerably lower than with a frame-based approach. Therefore, in order to improve video classification performance, the number of data is increased by relying on augmentation techniques, such as: affine transformation, rotation, flipping and noise injection through introducing cropping with small windows. Alternatively, if we exclude data augmentation process from our method, we have to either collect bigger sequence labeled datasets or to reformulate the current problem and deal with it in an unsupervised manner.

In this thesis, for each input image sequence, we generate sequences that are shifted horizontally and vertically at different degrees, rotated at different degrees, flipped and mirrored. Moreover, cropping with a small widow size is introduced. Moreover, scaling images at different resolutions and varying the light intensities are also applied. For an image sequence dataset of size  $N$ , an image sequence dataset of size  $28 \times N$  is thus generated, where the factor 28 refer to the number of transformation operations we applied. Those transformations increase the severity of the extrinsic variations. The structure of each sequence should remain the same by applying the same transformation over the entire images of the same sequence. Figure 3.22 shows some images from different sequences that represent some of the transformations we applied. Table 3.6 shows the total number of image sequences for each of the databases we consider for training, validating and testing our model before and after data augmentation.

| Database   | CK+  | MMI  | DISFA | DynEmo |
|--|------|------|-------|--------|
| Number of initial image sequences                                | 321  | 203  | 297   | 198    |
| Number of image sequences after data augmentation                | 8988 | 5684 | 8316  | 5544   |
| Number of image sequences for training after data augmentation   | 8148 | 4508 | 5852  | 3864   |
| Number of image sequences for validation after data augmentation | 392  | 588  | 1232  | 840    |
| Number of image sequences for test after data augmentation       | 392  | 588  | 1232  | 840    |

Table 3.6: The total number of image sequences before and after data augmentation considered for training, validation and testing. The database description and its protocol are also provided in Table 3.1, Section 3.1.

### 3.4.4 Network Diagnosis: Visual Debugging

In the process of training deep neural networks, an important number of hyper-parameters have to be tuned. Often these hyper-parameters are brittle and produce poor performance when slightly off. In this work, we propose a method where we can benefit from visualization to assist choosing the appropriate values for these parameters and so that it is possible to early shutdown the network to re-adjust the hyper-parameters setting if necessary.

Figure 3.23 corresponds to an experimental setup where both databases MMI and CK+ are combined together. The purpose of this merge is to develop an initial model architecture where the morphological segments used for training are both long and short. Moreover, since these databases have observable facial deformations, the interpretability of the feature maps through visualization makes sense, so that we can interpret the model behaviour. This is one

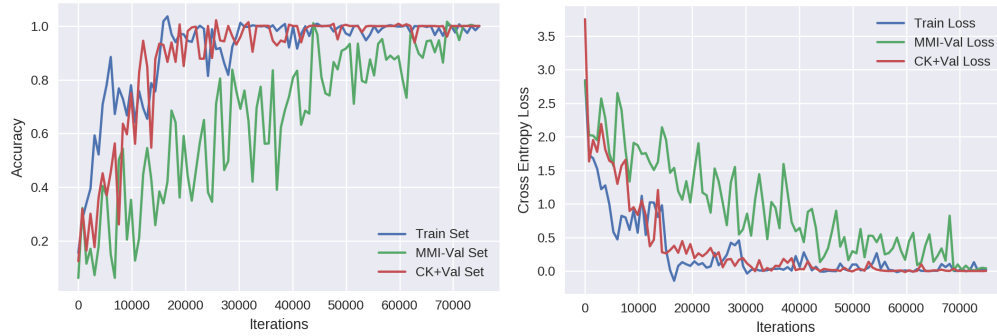


Figure 3.23: Learning curves and accuracy rates during the training and validation phases. The model is trained over a combination of MMI and CK+ databases. It is validated over the CK+ and the MMI validation sets.

of the reasons why we firstly evaluated our method on posed expressions, so that we get better insight on the model interpretability.

Firstly, we visualize the learning curves where both the cross entropy error and the recognition rate over the training and the validation sets are monitored. Figure 3.23 gives an insight about the model fitting and generalization. We can see that the model could fit the CK+ database easily since the training set is composed of very short sequences. The dynamic FEs presented in the CK+ database evolve from neutral to onset phase and there is not offset, which makes it easier to recognize. On the contrary, for the MMI database, fitting the model is harder due to the data variability: this database encompasses both frontal and profile views of faces and it includes the full morphological phase. We can see that the training process over the MMI database is accompanied with a lot of weight updates. The ADAM optimizer needs to run longer to tweak these weights and to converge. The fluctuations in the curves over iterations are due to the fact that we are using a mini-batch learning procedure, in addition that we are dropping out randomly some of the layer units at each iteration. Gal and Ghahramani 2016 proves that by following such a procedure at each iteration, the prediction confidence of the learned model is improved.

To show the effects of choosing appropriate and inappropriate hyper-parameters, we propose visualizing the weight histograms. For further analysis and better illustration, the weights of the 3DConv-1 layer are studied. Usually, for a 3D-CNN layer, we need to set spatial filter sizes (height and width) and temporal filter sizes (through time) and after, the filter is initialized using Xavier’s initializer. Under normal conditions, at the end of training, the weight histograms should form roughly a Gaussian distribution [Yosinski and Lipson 2012], which means the layers start representing and capturing useful patterns from the previous layers. Figure 3.24 shows that as the learning process goes on, the weight histogram starts taking the shape of a Gaussian distribution, that is the network starts capturing a meaningful representation. We set the weight appropriate hyper-parameters of the 3DConv-1 layer as: the kernel size =  $5 \times 5 \times 5$ , and the number of filters to 30. On the other hand, if we set the hyper-parameters of the same layer to a kernel size =  $3 \times 3 \times 3$ , and the number of filters to 30, we obtain the graph represented in Figure 3.25. It shows that even after 75000 iter-

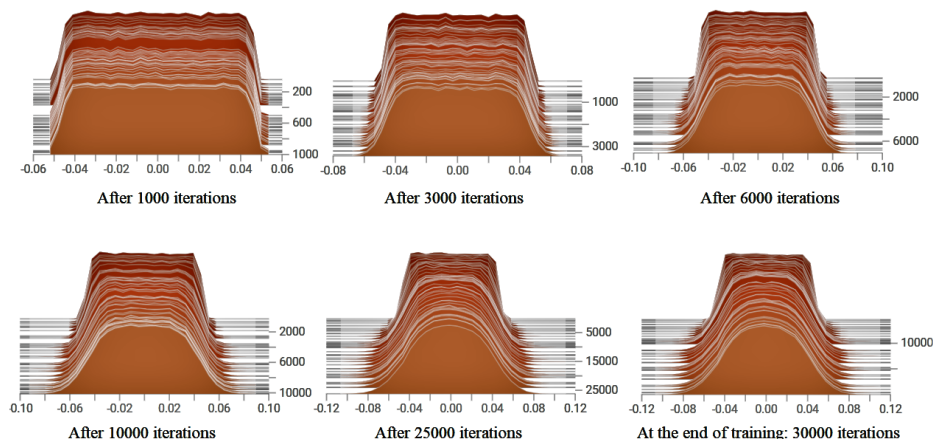


Figure 3.24: 3DConv-1 weight histograms through the learning procedure with an appropriate hyper-parameter choice.

ations, the network has difficulty in capturing meaningful patterns and thus provides a bad representation.

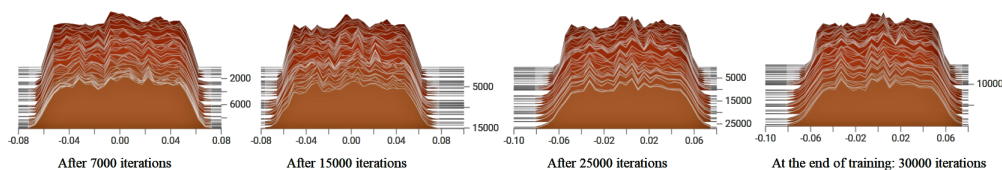


Figure 3.25: 3DConv-1 weight histograms through the learning procedure with an inappropriate hyper-parameter choice.

The visualization of the weight histogram is done in real time using TensorBoard [Abadi et al. 2016] while the network training is in progress. This is a very interesting property since it can provide a tool to directly shutdown the network after few iterations and to re-adjust the hyper-parameters of any layer if necessary (which is the case of Figure 3.25). The procedure allows to train efficiently a deep neural network architecture without waiting the full training to get insight about what is happening during the training. Therefore, we can interfere directly at any moment to regulate and optimize the network parameters and architecture.

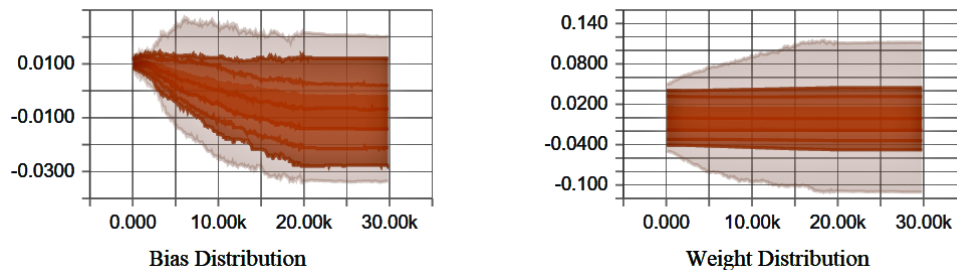


Figure 3.26: 3DConv-1 bias and weight distribution values over time.

Another important property we can exploit is the visualization of the weight and the bias distributions over time. During the training procedure the weight and the bias updates can be controlled to make sure they vary through the whole training procedure and did not get stuck at saddle points. If we face such a phenomenon, we can inject noise in the network or use a different regularization method that smoothes out the loss function variations. Figure 3.26 shows the distribution of the weights of the 3DConv-1 layer with the appropriate hyperparameter setting. It shows that the weights are updated at each iteration in order to minimize the final error. However, if the graph shows a constant line, it means that the weights are stabilized and that the optimizer could not find a way to proceed on.

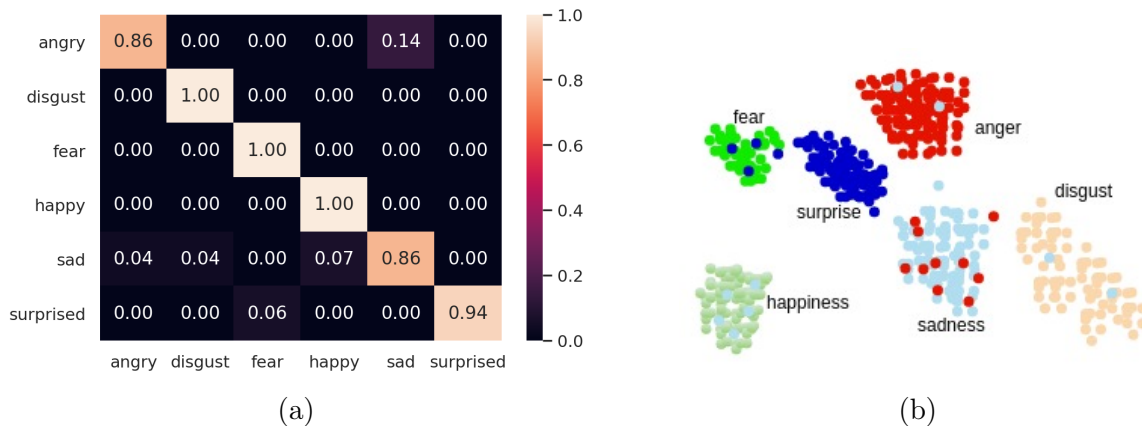


Figure 3.27: 2D visualization for the FC layer outputs over the test image sequences from the combination of MMI and CK+ databases using t-SNE projection and their corresponding confusion matrix.

In addition, in order to interpret the good generalization performance of a network on a classification task and to get insight about the separability of the data, we project the outputs of the FC layer, the final spatio-temporal feature vector representation, into a 2D space. The T-Distributed Stochastic Neighbor Embedding (t-SNE) [Maaten and Hinton 2008] is used for dimensionality reduction, because it is particularly well suited for the visualization of high-dimensional feature vectors. Figure 3.27 shows the output of each FC feature vector over the test image sequences of the MMI and CK+ databases, in which our network reported a 92% classification rate.

The resulting normalized confusion matrix is shown in Figure 3.27a. The *off diagonal* elements show the respective classification errors between FE classes, while the *diagonal* elements show the correct classification rates. It can be seen that most classes appear approximately separable, except for the *Angry* expression which has been confused with the *Sad* expression. In order to understand why, we intuitively look at their subspaces, as shown in Figure 3.27b. We found out that their subspaces are close to each other and that they overlap. Moreover, we can notice that *Fear* and *Surprised* are confused with each other. In addition, *Sad* has a subspace that is close to *Angry*, *Disgust* and *Happy* and thus it has been confused with these emotions.

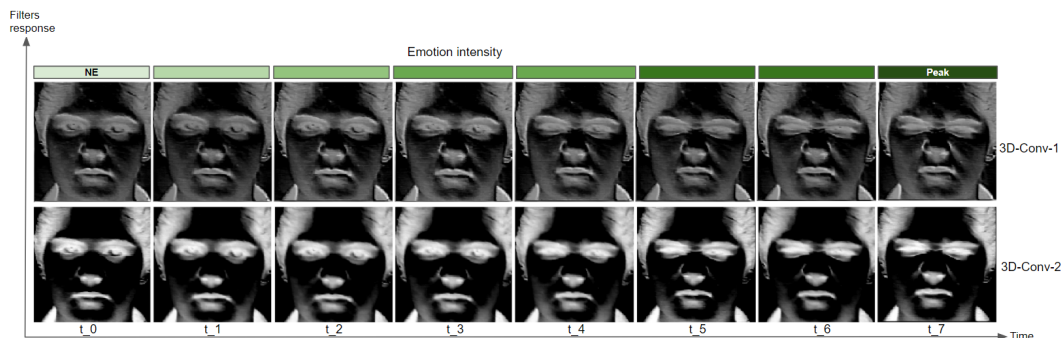


Figure 3.28: One of the feature map responses over time for a temporal sequence that corresponds to an *Angry* facial expression taken from the first and the second 3D-Conv block.

Finally, to better understand the features extracted by the learned model, we visualize some of the feature maps which are the response of the 3D convolutional filter banks over a test sample that corresponds to a very short sequence (for better visualization). The response is depicted in Figure 3.28 for the first and the second 3D-CNN layers. Figure 3.28 shows which parts of the facial image are strongly activated for the given FE, *Angry*. Our visualization shows the capability of the model to capture the variations in the intensity of the displayed expression through time and it shows how the model focuses from one layer to another only on the FE related features: nose, eyes, eyebrows, *etc.*

The feature maps could be seen as a representation that explicitly encodes what are the FE features involved for a certain emotion and implicitly encodes their spatial locations at each time step. 3D filters act as feature detectors from the original input facial image in space and time. By referring to Figure 3.28 and looking at *3DConv-1* layer, which is the output of one 3D-filter, we can see that a single 3D filter is capable at capturing features that evolve through time. For instance, it seems that the network is learning a 3D Gabor-like filter that responds to different edge texture features at the same time (horizontal, vertical and diagonal). In the next *3DConv-2* layer, as shown in the in Figure 3.28, mid level features start appearing which representing objects parts (eyes, nose, *etc.*). Those are the objects the network was trained to recognize automatically as patterns associated with a certain emotion through adjustment of the parameters of the model. As we keep visualizing deeper layers, we can see that the features become more abstract and strongly correspond to the patterns that contribute to the emotional class categorization. Other facial features as identity for instance, are going to be discarded by the model. This is possible through the design of an experimental setup that follows a *Subject-Independent* manner. Otherwise, the model most probably will learn both the identity and the emotion of the subject, which will result in poor generalization over unseen subject.

### 3.4.5 Hierarchical Spatio-Temporal to Hierarchical Spatial Feature Encoding Image-Based for FER

This section presents a 2D-CNN model for comparison purpose. The work focuses on the capacity of the spatial hierarchical feature representation for extracting discriminative and representative features for FER. We aim behind designing this model to compare it with the SPFER and the ImpBoVW models for assessing the relevance power of the different feature representations.

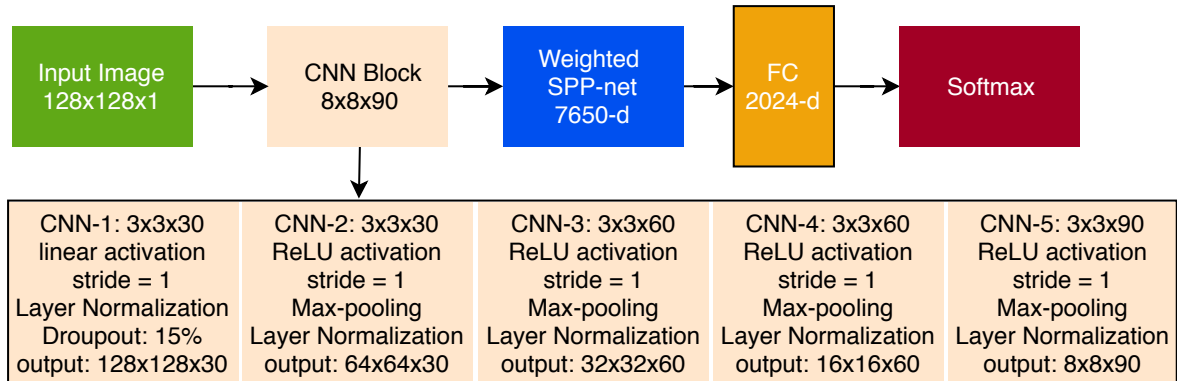


Figure 3.29: General model architecture for learning spatial hierarchical features for FER using static images.

**The 2D Spatial Network Architecture:** Our network is a variant of the spatio-temporal model we described in Section 3.4.1 in which we replace all the 3D-Conv layers and the ConvLSTM layers with a 2D-CNN followed by the same network configuration: normalization, stride and dropout values. The proposed network as shown in Figure 3.29 consists of five 2D-CNN layers, followed by a SPP-Net, a FC layer and a softmax layer. We follow the same initialization procedure we proposed in the previous section and the same way of controlling the hyper-parameters.

**Training Protocol:** In order to establish a reasonable number of images for training the 2D neural network, we use data augmentation with the same setting we proposed in Section 3.4.3. The number of images for each database is shown in Table 3.7. However, in order to compare our results with SPFER and ImpBoVW, we use the same test images we evaluated over those algorithms as shown in Table 3.7, the 6<sub>th</sub> row. The result of the algorithms using the same test protocol is shown in Table 3.8.

**Experimental Evaluation:** We evaluate the accuracy of the proposed 2D deep neural network architecture and compare it with the SPFER and ImpBoVW models we proposed previously. The results are reported in Table 3.8.

Herein, we aim at establishing the capacity power of the feature representation on overcoming the main challenges that affect FER. Our results are also plotted in Figure 3.30. We can see that our results confirm the superiority of hierarchical features over mid-level features and in particular over low-level features. The clear advantage of the proposed method is an

### 3.4. Hierarchical Spatial and Spatio-Temporal Feature Encoding Based on Deep Neural Network for FER 91

| Database:   | JAFFE            | CK+        | MMI        | DISFA      | DynEmo     |
|---|------------------|------------|------------|------------|------------|
| Number of initial image                                 | 213              | 321        | 203        | 297        | 198        |
| Number of images after data augmentation                | 5964             | 8988       | 5684       | 8316       | 5544       |
| Number of images for training after data augmentation   | 4004             | 8148       | 4508       | 5852       | 3864       |
| Number of images for validation after data augmentation | cross validation | 392        | 588        | 1232       | 840        |
| <i>Number of images tested over ImpBoVW and SPFER</i>   | <i>66</i>        | <i>250</i> | <i>300</i> | <i>550</i> | <i>400</i> |
| Number of images for test after data augmentation       | 1848             | 392        | 588        | 1232       | 840        |

Table 3.7: The total number of images before and after data augmentation considered for the training, validation and testing phases.

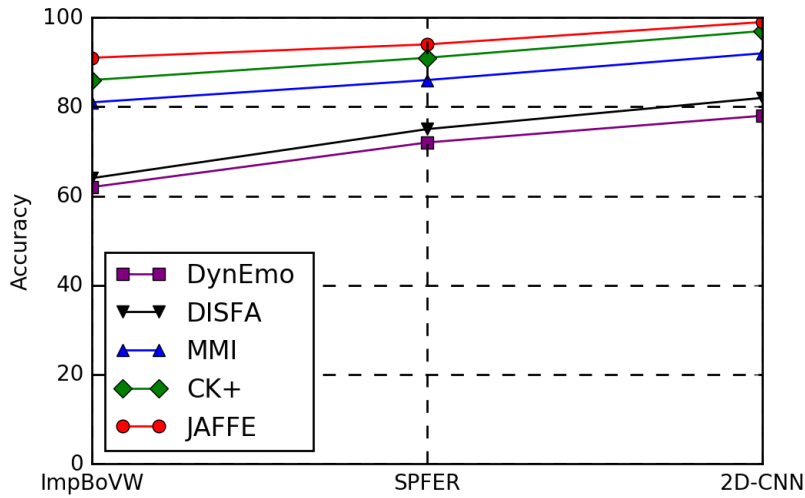


Figure 3.30: The 2D-CNN model results using the same test data protocol over the algorithms SPFER and ImpBoVW. The values are shown in Table 3.8.

| Method\Database | JAFFE | CK+ | MMI | DISFA | DynEmo |
|-----------------|-------|-----|-----|-------|--------|
| ImpBoVW         | 91    | 86  | 81  | 64    | 62     |
| SPFER           | 94    | 91  | 86  | 75    | 72     |
| 2D-CNN          | 99    | 96  | 92  | 81    | 77     |

Table 3.8: Recognition rates in % over all databases using low, mid level and hierarchical features using same data protocol.

increased classification accuracy over challenging spontaneous databases coming from basic and non-basic emotions and having various extrinsic and intrinsic variations, which is the case of the DISFA and the DynEmo databases. Moreover, the model achieves the best results over acted databases with basic expressions, such as the JAFFE, CK+ and MMI databases. We can see that for the MMI database, the recognition rate increases from 81% using ImpBoVW to 92% using 2D-CNN. Despite the fact that the MMI database is an acted database, it encompasses many challenging variations and with hierarchical features, we are able to overcome these challenges.



Table 3.9 presents the results of the 2D-CNN model over test sets that include data augmentation. In this setting, we can clearly see the robustness of the deep hierarchical features, even though the results drop approximately by 1%, it still competitive since the test sets here include a lot of extrinsic variations such as rotations at multiple degrees, occlusions at different locations, noise addition with different intensities, *etc.*

| Data Protocol                | JAFFE | CK+ | MMI | DISFA | DynEmo |
|------------------------------|-------|-----|-----|-------|--------|
| With Data Augmentation       | 99    | 96  | 90  | 81    | 77     |
| Similar to ImpBoVW and SPFER | 99    | 97  | 92  | 82    | 78     |

Table 3.9: The 2D-CNN model results over test data with data augmentation and over the same test data used for ImpBoVW and SPFER. The protocol is presented in Table 3.7, the last row.

The obtained results using 2D-image models we proposed: the 2D-CNN, the SPFER and the ImpBoVW, are not comparable with the spatio-temporal model we will present in the next Section 3.4.6, because the temporal links are not taken into account while only the expressive frame is considered. In this research, regarding 2D FER systems, we present a comprehensive evaluation and solutions that can tackle challenging FE variations in less controlled environment.

### 3.4.6 Experimental Setup and Results Analysis Over the Spatio-Temporal Model

In this section, we focus on the importance of temporal information. We design our experimental setup in a way that permits us to:

1. Analyse the performances of the spatio-temporal model for FER over acted and spontaneous databases.
2. Compare our hierarchical spatio-temporal model with state of the art methods.
3. Study the influence of the feature representation and encoding whether based on high level features extracted in a unique process and integrated at classification level (our model), or based on features extracted using low or mid level and then followed by classification (classical approaches).

**Remark.** The obtained results using 2D-image models we proposed: the 2D-CNN, the SPFER and the ImpBoVW, are not comparable with the spatio-temporal model because the temporal links are not taken into account while only the most expressive frame is considered.

**Model Analysis.** We previously analyse our model performances over the CK+ and the MMI databases, which consist of acted basic expressions and includes short and long image sequences respectively. Herein, alongside these databases, we test our model behaviour over spontaneous databases with basic and non-basic expressions, namely the DISFA and the DynEmo. The result are shown in Table 3.10.

| Database         | CK+          | MMI            | CK+ & MMI    | DISFA        | DynEmo       |
|------------------|--------------|----------------|--------------|--------------|--------------|
| Recognition rate | $97 \pm 0.5$ | $88.9 \pm 1.1$ | $92 \pm 0.8$ | $71 \pm 1.4$ | $67 \pm 1.5$ |

Table 3.10: Hierarchical spatio-temporal model classification performances over acted and spontaneous databases having short and long image sequences and encompassing various intrinsic and extrinsic variations. The sign  $\pm$  indicates the average variance of our model when we repeat the same experiment three times. The variance is directly correlated to the parameters initialization.

Table 3.10 presents the recognition rates on different databases. We can see that among posed dynamic databases, recognizing sequences evolving from onset to apex, which is the case of the CK+ database, a high recognition rate is achieved, reaching 97%. However, for recognizing sequences evolving from onset to apex and returning back to offset, which is the case of the MMI database, the classification performance is 89%. We inspect some failure cases on the MMI database, to check out why the model performance decreases while we are still dealing with posed database. We find out that most of the falsely classified samples belong to profile views that include corruption which we added during data augmentation. Obviously, adding noise to profile faces, mainly over prominent ROIs, such as the mouth or the eyes, leads to miss-classification because the symmetry of those prominent ROIs does not exist in the profile view. Therefore, there is not enough distinctive features for proper classification.

On the contrary, recognizing spontaneous FEs is a much harder problem than recognizing posed ones, in which the extrinsic variations are not very well controlled. As shown in Table 3.10, the recognition rate over basic spontaneous FEs using the DISFA database is 71%. While the recognition rate over mental state expressions using the DynEmo database is 67%. Figure 3.32a and 3.32b show the normalized confusion matrices over the DynEmo and the DISFA databases.

Figure 3.32a shows that most of the Ekman’s spontaneous FEs appear approximately separable with an average recognition rate per class around 71%. The main confusion is associated with *Disgust* expression which is predicted with 20% as *Angry*. In addition, being *Sad* is estimated with 29% as *Fear*. Finally, being *Surprised* has been overlapped mainly with feeling *Happy* with 25% and 13% with *Fear*. Figure 3.32b shows that most of the spontaneous mental states appear to overlap with each other. For instance, being *Astonished* is overlapping with *Curious*. This is may be due to the fact that being *Astonished* or impressed can lead to *Curious* or being amazed. The highest recognition rate is related to the *Astonished* emotion while the lowest one is related to *Curious*. The emotion *Curious* has made significant confusion in all of the categories, it is mainly overlapped with *Amused*, *Astonished*, *Disappointed* and *Affected*. It sounds out that *Curious* is a more complicated feeling where emotions appear to blend together. Being *Amused* has overlapped with *Astonished* and *Curious*. Moreover, we can notice that being *Affected* is overlapped with being *Disappointed* and contrariwise. However, being *Fright* has been affected by being *Disappointed*.

By comparing the confusion matrix of the spontaneous DISFA database (Figure 3.32a) with the confusion matrix obtained by the combination of the posed MMI and CK+ databases

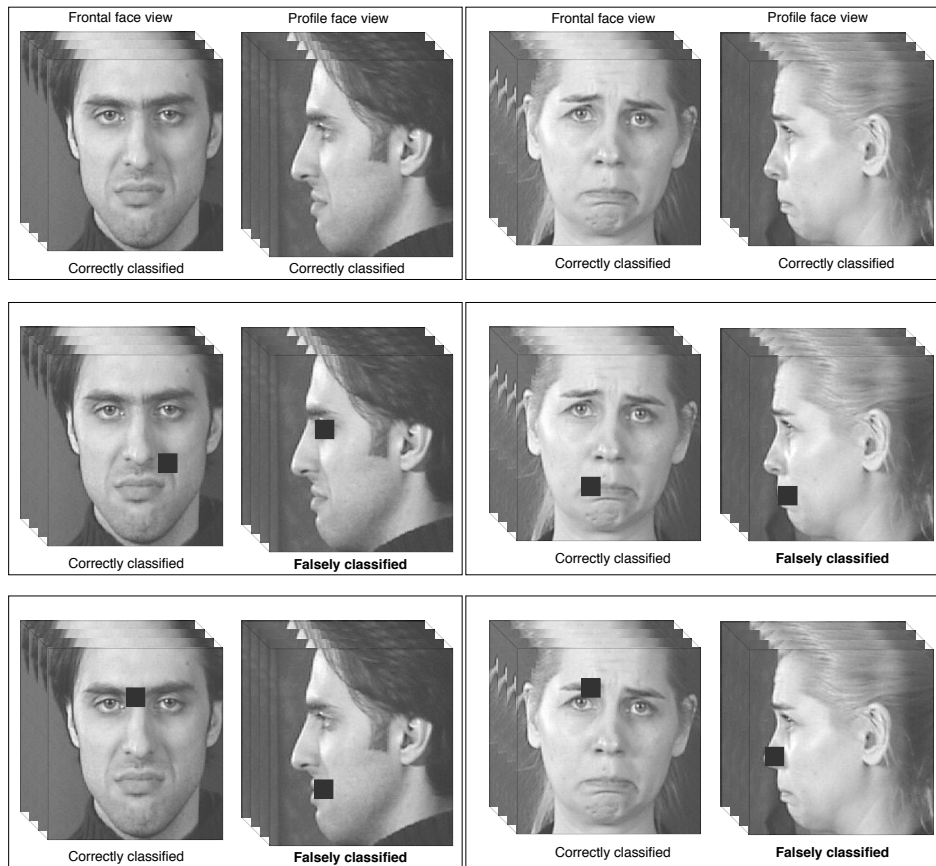


Figure 3.31: Frontal and profile face image sequences for Sad expression. In this example, all the frontal face image sequences are correctly classified. For the profile faces, some examples are falsely classified and confused with the Disgust expression.

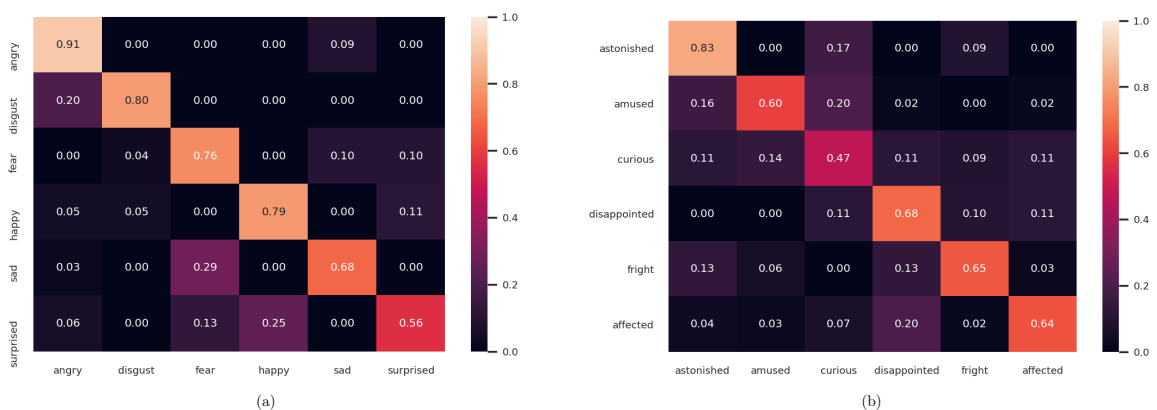


Figure 3.32: Normalized confusion matrices over the (a) DISFA database, (b): DynEmo database.

(Figure 3.27), we observe that the emotions generated from posed stimuli yield to high recognition rates while the same emotions induced lower recognition rates when generated from spontaneous stimuli. The algorithm has to deal with stronger variabilities with spontaneous expressions.

In order to reduce the overlap between emotions, the usual scenario is either to collect more reliable labeled datasets or to improve the performance of the classifier and feature extraction methods. In addition, we usually look at the class labels as independent, that is, no relationship among the class labels exists. Alternatively, we may be able to improve the overall performance by considering such dependencies between classes. A recent computer vision work [Silva-Palacios et al. 2017] has considered such dependencies and shown improvement on the accuracy with respect to the basic flat approach.

**Model Comparison with State of the Art.** For fair comparison, we compare our results with a recent deep learning work [Mohammad Mahoor 2017] that models the spatio-temporal relationships in FER and we report the recognition rates of some recent state of the art methods. By referring to Table 3.11, first we can observe that our model achieves a better recognition accuracy over basic expression databases whether posed or spontaneous than other approaches. Second, compared to similar approaches such as [Mohammad Mahoor 2017], we achieve a better performance on the challenging spontaneous DISFA database having with 13% additional classification rate enhancement over the best reported result by Mohammad Mahoor, 58%. Our method differs by using a ConvLSTM module to maintain the spatial and temporal correlations between the features in space and time. And afterwards, we learn a non-variant representation to aggregate multi-scale features that provide robustness against scaling or low resolution frames. However, the method proposed by Mohammad Mahoor is based on transforming the 2D Inception-ResNet architecture into 3D using a 3D-CNN instead of a 2D-CNN and then flattening the final feature vectors and passing them sequentially to an LSTM unit. By doing that, Mohammad Mahoor lost the spatial correlations along time. We also improve the recognition results over the CK+ database by around 4% and around 2% for the MMI database from the best reported results.

| Databases | <i>Our model</i>  | state of the art  |
|-----------|-------------------|---|
| CK+       | <b>97</b> ± 0.5   | 93.6 Zhang et al. 2015a,<br>93.2 Mollahosseini et al. 2016b,<br>92 Liu et al. 2014c,<br>93.2 Mohammad Mahoor 2017 |
| MMI       | <b>88.9</b> ± 1.1 | 86.7 Shan et al. 2009,<br>79.8 Taheri et al. 2014,<br>78.51 Mohammadi et al. 2014,<br>77.5 Mohammad Mahoor 2017   |
| DISFA     | <b>71</b> ± 1.4   | 55.0 Mollahosseini et al. 2016b,<br>58.0 Mohammad Mahoor 2017   |

Table 3.11: State of the art recognition rates in %.

**Studying the Influence of the Feature Representation and Encoding.** We study the impact of various feature representation while using spatio-temporal representation. First,

|           |                   | Models      |            |                 |                |
|-----------|-------------------|-------------|------------|-----------------|----------------|
| Databases | <i>Our model</i>  | Alex<br>Net | VGG<br>Net | 3D-SIFT<br>BoVW | 3D-HOG<br>BoVW |
| CK+       | <b>97</b> ± 0.5   | 92.2        | 91         | 83.9            | 89.4           |
| MMI       | <b>88.9</b> ± 1.1 | 57          | 62         | 63.72           | 61.09          |
| CK+ & MMI | <b>92</b> ± 0.8   | 60.4        | 65         | 59.4            | 62.8           |

Table 3.12: Recognition rates in % over dynamic acted expressions.

|           |                  | Models      |            |                 |                |
|-----------|------------------|-------------|------------|-----------------|----------------|
| Databases | <i>Our model</i> | Alex<br>Net | VGG<br>Net | 3D-SIFT<br>BoVW | 3D-HOG<br>BoVW |
| DynEmo    | <b>67</b> ± 1.5  | 53.6        | 55         | 45.8            | 52.2           |
| DISFA     | <b>71</b> ± 1.4  | 56.1        | 58         | 48              | 53.4           |

Table 3.13: Recognition rates over dynamic spontaneous expressions.

we rely on a transfer learning method in order to benefit from pre-trained neural networks on very large databases that could work as a mid-level feature extractor. We mainly use AlexNet [Krizhevsky et al. 2012] and facial VGG-Net [Cimpoi et al. 2015]. We choose the last convolutional layer activations as features and we add on the top of it a ConvLSTM cell to model the dynamic behaviour. In this setting, the image sequences are resized to match the requirements of those trained models. Then, in order to evaluate the power of low level spatio-temporal features, we use spatio-temporal hand crafted descriptors, mainly 3D-SIFT [Scovanner et al. 2007] and 3D-HOG [Scherer et al. 2010], accompanied with BoVW for video representations and followed by SVM for classification.

Table 3.12 shows the recognition rates over dynamic acted expressions achieved on each of the databases: MMI, CK+ and their combination (MMI and CK+). Table 3.13 shows the recognition rates over dynamic spontaneous expressions achieved on each of the databases DISFA and DynEmo. Tables 3.12 and 3.13 show the capability of our model at recognizing acted and spontaneous dynamic FEs with a recognition rate that outperforms other methods whether using features extracted from transfer learning or using classical spatio-temporal descriptors.

By comparing the results obtained in Table 3.12, we notice that among posed dynamic databases, recognizing FEs with ideal extrinsic and intrinsic variations, such as the CK+ database, achieves better recognition than recognizing posed expression with less controlled conditions, as it is the case of the MMI database. For instance, with low level spatio-temporal features along BoVW, the recognition rate over the CK+ is above 89% and it is also competitive with the results obtained from pre-trained networks. Once profile faces are introduced to the training as in the MMI, the highest performance using these low level spatio-temporal methods is around 63%. Moreover, once spontaneous expressions are introduced, the capability to recognize them decreases to reach its maximum at 53%. Also, we can conclude from Tables 3.12 and 3.13 that mid-level features extracted from trained networks, slightly did better than the low level features. However, the best performance yet is still achieved by learning

hierarchical spatio-temporal features, which are very important especially when dealing with spontaneous facial behaviours.

Beyond that, Tables 3.12 and 3.13 underline important facts. Essentially, they emphasize the performances of different feature extraction and representation, which range from low level features (descriptors), mid level features (hierarchical features extracted from pre-trained network), to high hierarchical level features extracted and integrated at classification level (our approach and the work done in [Mohammad Mahoor 2017]). Clearly, an improved performance is achieved by learning a more sophisticated representation for modelling the local and the global variations in the input FE video through combining 3D-CNN and ConvLSTM networks.

Our representation offers several advantages over those obtained through hand-crafted descriptors, wherein we capture higher-order statistics such as angles, edges, object parts and complete objects which give the flexibility to be tuned to the statistics of the specific object classes being considered (FEs). More importantly, the end-to-end approach can be adapted to new domains where the hand-crafted descriptors may not be appropriate. For instance, we were able to adapt other pre-trained networks trained for different tasks to extract features for the emotion recognition task. The results show that those features slightly perform better than hand-crafted features. The advantages of pre-trained networks lie in the computational time complexity, where spatio-temporal descriptors take a long time to be computed. Nonetheless, designing the right architecture is still the best solution to achieve a better recognition rate.

### 3.4.7 Conclusion

In this chapter, we focused our study on macro facial expressions coming from posed and spontaneous stimuli that represent universal basic behaviours and mental states. Different intrinsic and extrinsic factors have been considered. We developed models that consider only the spatial dimension, where the input is the most expressive image and is associated with a discrete class label. A general representation that summarizes the main results of this chapter is shown in Figure 3.33. Among spatial models, we test the power of various facial feature representations and encoding. In particular, first, we encode low level features using handcrafted descriptors and we improve the BoVW representation. This model achieved around 86% recognition rate over posed facial stimuli and around 63% over spontaneous stimuli. Afterwards, we encode mid-level features using sparse representation concept, in which we built a discriminative dictionary specified for facial expression classification. The results of SPFER shown a significant improvement over spontaneous stimuli, in which on average it achieved around 73%. Moreover, an improvement over posed stimuli is also reported, with a 90% average classification rate. Finally, we encode hierarchical features using convolutional neural networks and we achieved notable improvements over both posed and spontaneous stimuli, with 95% and 79% average classification rates respectively. Obviously, hierarchical features are powerful for improving the FER performances and can manage to overcome various challenges.

Spatial models do not consider motion features and are not capable of solving ambiguous

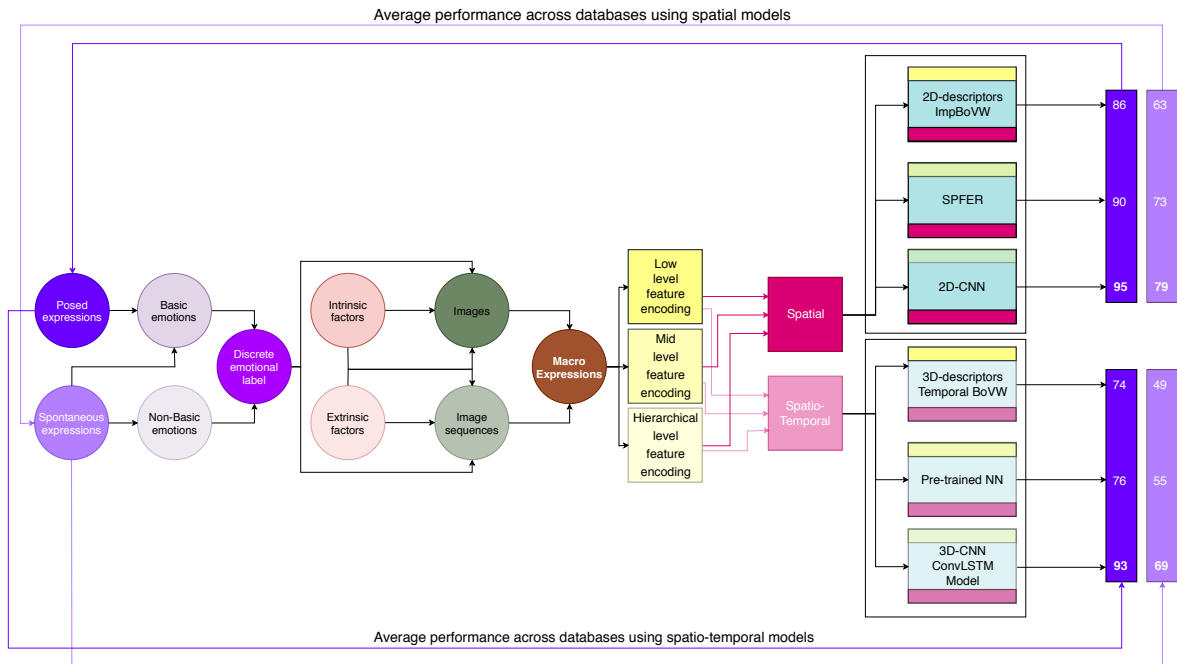


Figure 3.33: Facial expression input source variabilities with respect to the average performances over spatial and spatio-temporal models. The reported numbers are computed by averaging the classification rates of each model over either acted or spontaneous databases.

FEs when emotions are different but have similar facial expression deformations and only subtle differences. Very subtle facial deformations cannot be encoded with such spatial models. Therefore, we consider developing a hierarchical spatio-temporal representation that takes into account local and global image sequence information, in order to model subtle feature deformations and to overcome various extrinsic and intrinsic variations. Our model reported the best performances among the state of the art and among different levels of spatio-temporal feature encoding. Especially over posed stimulus where we achieved 93%. However, on spontaneous stimulus, on average we achieved around 69%. Those results are very encouraging.

In this study, we cannot directly compare the spatial and spatio-temporal results. As a matter of fact, those two representations have different settings and model designs. For instance, the reported high performances on the 2D models are not an indication that the 3D models should exceed those results. However, we can still get insight on the importance of the feature encoding level, where it appears that high level feature encoding is preferred to solve complex tasks as FER. Similar to spatial models experimental setup, we built a low level feature representation using on the shelf 3D-descriptors along a temporal BoVW model. Moreover, we rely on recent pre-trained models to extract mid-level deep features and we process them temporally.

Obviously, jointly designing an optimal hierarchical feature extractor and classifier are not enough to absolutely improve the performances of FER. Further investigation is required at the level of feature adaptation, data collection, knowledge transfer. For instance, it is

### **3.4. Hierarchical Spatial and Spatio-Temporal Feature Encoding Based on Deep Neural Network for FER**

---

**99**

not practical to collect huge databases that encompass all extrinsic and intrinsic variabilities. Moreover, it is very hard to label and induce spontaneous facial deformations. We propose to profit from the current available datasets, whether posed or spontaneous, and to direct our line of research into building a model based on hierarchical feature encoding, as our results shown their robustness, toward adapting their source domain where they are trained over to new target domains with new data distributions or new unseen classes.





# Adapting Facial Expression Models to New Domains or Tasks

---

Training a model for FER requires large amounts of labelled data which is difficult to acquire. And even if protocols for eliciting spontaneous reactions have been developed, they are not able to tackle with all the possible emotions and all the possible sources of variabilities. In this chapter, we introduce a method that adapts facial expression models trained for a particular visual domain that is posed expression datasets and to a new domain that is spontaneous expression datasets by learning a transformation that minimizes the effect of domain-induced changes in the feature distribution. This is called Domain Adaptation (DA) process. Moreover, we study the problem of Zero Shot Learning (ZSL) for FER recognition, where emotions in the test set are unseen during the training step. The proposed DA and ZSL methods for FER are solved jointly using a deep neural network model and are realized in an end-to-end learning fashion by learning a mapping across visual and semantic representations, so that one can apply the learned model to recognize cross-domain data and/or unseen image categories. In the following, we present an introduction to domain adaptation and zero shot learning methods, followed by the related work and the proposed model. Finally, we present our model validation through extensive experimental evaluations.

## 4.1 Introduction to DA and ZSL for FER

**Adapting to Feature Distributions.** Most of recent existing FER methods are based on supervised discriminative learning [Eleftheriadis et al. 2015; Happy and Routray 2015; Jun et al. 2015; Kim et al. 2014] and some rely on deep CNN [Chanti and Caplier 2018; Chen et al. 2015; Yang et al. 2018], including our contributions in Chapter 3. These models are mainly focused on building a feature representation and encoding for robust classification that alleviate from FER challenges coming from different extrinsic and intrinsic factors variability. These methods achieve impressive accuracies when trained and tested on the same posed FE database while achieving good accuracies to a certain degree when trained and tested on the same spontaneous FE database. Therefore, these supervised learning methods behave well when training data (*source domain*) and test data (*target domain*) are drawn from the same feature distribution.

In real-world applications, FE databases are inconsistent between each other. For instance,

face images of the same expression may have different appearances in the face images within the same database. On the contrary, different expressions may have a similar appearance in the face images of different subjects from different databases. Such inconsistency is due to varying domains because of different extrinsic and intrinsic factors variability, such as different cameras, illuminations, populations, acquisition setup and participants' culture background or personality, *etc.* As a consequence of the foregoing inconsistency between different databases, the performance degrades when we train a FER system on one source domain and we test it on another target domain [Zhu et al. 2016, Parthasarathy and Busso 2017]. The mismatch distribution problem is often referred as domain-shift [Torralba and Efros 2011].

In this context, a robust FER model must take special care during the learning process to infer models that adapt well to the test data they are deployed on. Yet, many critical issues associated to the target domain induce the domain-shift problem. Mainly three: 1) the inter-subject-expression variations such as the way to produce an expression are inconsistent across different people and some might be different, 2) the large variance in face pose, illumination, occlusions, changes in the camera and image resolution, and finally 3) the issues with spontaneous expressions with various intensities.

Table 4.1 represents classification rates when shifting from a posed emotion category dataset to a spontaneous dataset. As Table 4.1 shows, it is inadequate to straightly use an emotion classifier trained on the posed emotion source domain to classify spontaneous emotions, as the recognition rate diminishes significantly on the new target domain. Even when the same features are extracted in both domains, and a fundamental normalization is performed on the image and the feature vectors, the fundamental cause of the domain shift can strongly alter the feature distribution.

Domain Adaptation (DA) [Ben-David and Schuller 2003; Crammer et al. 2006] aims to learn a representation that optimizes performance on the target domain based on knowledge learning from the source domain coming from different distributions. In this chapter, *our first task is to investigate how to build an embedding space that enforces domain-invariance*, that is minimizing the mismatches between the feature distributions among different domains. Our interest is to train a FE classifier from a large *posed* FEs collection performed by a subset of subjects (the source domain) and to use it for new subjects performing the same FEs but in *spontaneous* way (the target domain).

We contend that addressing the problem of DA for FER is essential because: 1) while labeled posed facial expression datasets are becoming larger, more available and easier to annotate, several applications such as drowsiness or pain detection require realistic facial expressions that are difficult to annotate. 2) It is unrealistic to collect many labels in each new domain, especially if one considers the large number of possible extrinsic and intrinsic factor variabilities.

Therefore, we propose a Domain Adaptation for Facial Expression Recognition method (DA-FER) that transfers emotion category knowledge from large posed labeled datasets to new spontaneous target domains with similar labels but with different data distributions. The key idea is to establish a non-linear transformation function that maps both domains closer to




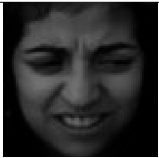






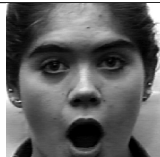

| Class     | Source: posed<br>(MMI set)  | MaE | Target: spontaneous<br>MaE (DISFA set)  | Recognition rate: source domain | Recognition rate: target domain |
|-----------|---|-----|---|---------------------------------|---------------------------------|
| anger     |    |     |    | 88                              | 24                              |
| disgust   |    |     |    | 91                              | 31                              |
| fear      |    |     |    | 82                              | 25                              |
| happiness |   |     |   | 95                              | 36                              |
| sadness   |  |     |  | 84                              | 27                              |
| surprise  |  |     |  | 99                              | 32                              |

Table 4.1: Examples of visual domain shift. Performance degradation of the classification method based on 2D-CNN model (Section 3.4.5) when trained and tested on image domains coming from two different distributions: the MMI dataset as the source domain and the DISFA dataset as the target domain.

each other in an embedding space. The embedding space acts as an intermediate layer which alleviates from the domain shift problem. The learning process relies on the available source domain dataset and during inference the learned mapping function is applied on the target domain.

**Adapting to Unseen Categories.** Likewise, one of the main challenges for scaling up FER systems for real-world non-basic emotional categories, is the lack of annotated data. For instance, most affective behaviors that occur in real interpersonal interactions or in HCI are

not basic emotions but pervasive ones (*scared, anxious, hesitant, thinking, suspicious*). Basic emotions are less frequent (*anger, disgust, fear, happiness, sadness, surprise*), while other are quite rare and complex (*depression*), or extremely rare. State of the art FER systems achieved tremendous progresses on solving the task of recognizing universal basic emotions using sufficient labeled posed or spontaneous facial expression samples for the model training. However, this is affordable due to the ease of interpretation of the related facial expressions, the ease of acquisition of posed facial expressions and recently spontaneous ones. Nevertheless, the emotional classes follow a long-tailed distribution which basically means that emotions that do not occur frequently in real life have a large number of images (as the case of Ekman's universal emotions) while new complex emotions occurring more often have much less data (as the case of pervasive ones). The point is that it is difficult and expensive to collect adequate annotated samples for each new emotional category. How to train efficient classification models that allow non-basic emotional class data to take advantage from the statistical knowledge learned from the numerous samples of basic emotions databases is our one of our goals.

In the conventional FER framework only emotional classes present in the training data can be recognized by the model during the test phase. These approaches have difficulties to tackle the challenging scenarios in which new emotional classes appear after the learning stage. Such scenarios happen due to the innumerable complex and nuance emotions which makes it unfeasible to assemble a complete list of all emotion classes and collect large training samples for each category. The problem becomes even more serious when we target fine-grained emotional categories such as mental states (*e.g. irritated, anxious, ashamed, curious, disappointed, affected, and astonished*), where only subtle distinctions exist.

In the absence of emotional category annotations, the Zero Shot Learning (ZSL) method [Farhadi et al. 2009; Lampert et al. 2009] has been designed to transfer knowledge from observed classes to unseen classes using a common embedding space that acts as a sort of bridge. The embedding space can be given in the form of a semantic representation using high-level visual attributes [Akata et al. 2013; Jayaraman and Grauman 2014; Mensink et al. 2014; Wang and Ji 2013] annotated by human experts or algorithms, or in the form of textual descriptions of classes using Distributed Word Embeddings (DWE) [Mikolov et al. 2013; Pennington et al. 2014] or Natural Language Processing (NLP) [Kiperwasser and Goldberg 2016]. The advantage of ZSL methods is to reduce the high cost of annotating new target domain data.

In this chapter, *our second task is to study how to transfer knowledge from a source domain data of posed and spontaneous universal basic Ekman's classes to a target domain data corresponding to unseen mental state classes*. Therefore, we propose a Zero Shot (ZS) Facial Expression Recognition (ZS-FER) based on a semantic space embedding. ZS-FER is built on a source domain dataset containing posed and spontaneous basic emotions, in order to provide guidance towards knowledge transfer to recognize a novel emotional target domain containing mental states.

Both DA-FER and ZS-FER models are based on learning a non-linear transformation function that is responsible for establishing an embedding space, where domain shift can be alleviated or unseen categories embedded. Therefore, both models can be solved jointly.

## 4.2 Joint Formulation of DA and ZSL and Challenges

**Problem Formulation.** Suppose we have access to a source domain  $\mathcal{S}$  with  $n_{\mathcal{S}}$  instances  $\mathcal{S} = \{(I_i^{\mathcal{S}}, \mathbf{a}_k^{\mathcal{S}}, z_k^{\mathcal{S}})\}_{i=1, k=1}^{n_{\mathcal{S}}, K}$  of source classes  $\mathcal{Z}^{\mathcal{S}}$ , and to a target domain  $\mathcal{T}$  with  $n_{\mathcal{T}}$  instances  $\mathcal{T} = \{(I_i^{\mathcal{T}}, \mathbf{a}_k^{\mathcal{T}}, z_k^{\mathcal{T}})\}_{i=1, k=1}^{n_{\mathcal{T}}, K}$  of target classes  $\mathcal{Z}^{\mathcal{T}}$ . Each instance  $I_i \in \mathbb{R}^{h \times w \times c}$  represents an image, where  $h, w$  and  $c$  are the image dimensions and the number of channels, respectively.  $K$  is the total number of classes in each of the source or target domain and it can have different values for each domain. The class label vector for the  $\mathcal{S}$  and  $\mathcal{T}$  data are represented as  $\mathbf{z}^{\mathcal{S}}$  and  $\mathbf{z}^{\mathcal{T}}$  respectively.  $\mathbf{a}_k$  represents either the class name (*e.g.* Surprised) or the textual description vector of visual attributes for a given class name  $z_k$  (*e.g.* Surprised is the class name: {inner brow raiser, outer brow raiser, upper lid raiser, lips part, and jaw drop} is the textual description vector).

A general representation for such a model and its expected embedding space is demonstrated in Figure 4.1. Given an instance  $I_i$ , we compute its *visual signature*  $\mathbf{x}_i \in \mathbb{R}^d$  in the visual embedding space. Likewise, given the class name  $z_k$  and its textual visual attribute description vector  $\mathbf{a}_k$  that describes  $z_k$ , we compute its semantic representation  $\mathbf{y}_k \in \mathbb{R}^m$  in a semantic embedding space referred as the *semantic class prototype*. One can think of  $\mathbf{x}_i$  as a deep feature vector learned from a CNN model ( $f(\cdot)$ ) and  $\mathbf{y}_k$  can be interpreted as semantic knowledge learned in the text domain ( $g(\cdot)$ ) via Distributed Word Embeddings (DWE), such as Word to Vector (word2vec) [Mikolov et al. 2013], Global Vectors for Word Representation (GloVe) [Pennington et al. 2014]) or through Natural Language Processing (NLP) models [Kiperwasser and Goldberg 2016]. NLP or DWE aim at providing semantically-meaningful representations that capture the semantic similarity between different words (class name) or sentences (textual visual attributes).

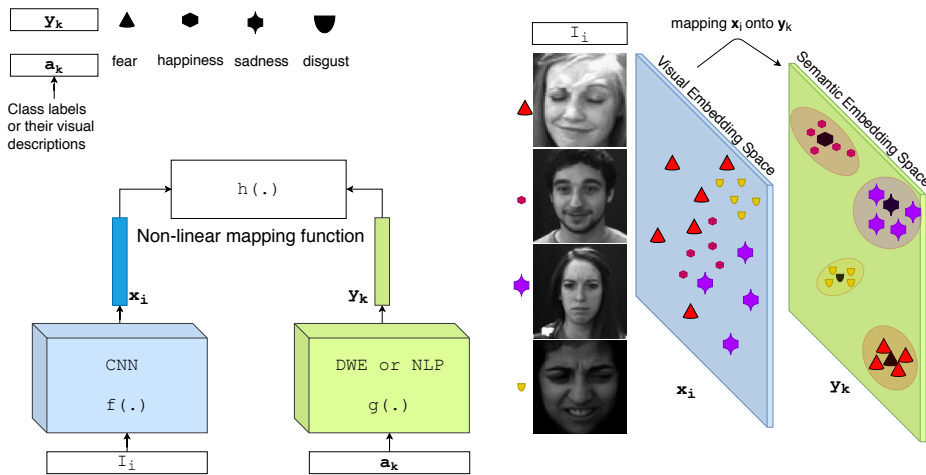


Figure 4.1: Regression model for establishing an embedding space that maps the visual image signatures  $\mathbf{x}_i$  onto the semantic prototypes  $\mathbf{y}_k$ .

To this end, the main objective is to learn a proper non-linear mapping function ( $h(\cdot)$ ) that maps the visual signature  $\mathbf{x}_i$  onto the semantic embedding space, such that all the feature

points  $\mathbf{x}_i$  of a category  $z_k$  cluster around their semantic class prototype  $\mathbf{y}_k$ , independently of its domain or task.

To better understand the problem, let us introduce an example as shown in Figure 4.2. Figure 4.2a shows that the source domain visual signatures are shifted from their center which is the semantic class prototype. However, this shift is not as severe as for the visual signatures coming from the target domain, which are totally confused. Moreover, if a new target example arrives, as this class has no center to be associated to, it spanned randomly over this space. Therefore, to solve the domain adaptation and to categorize new unseen examples as shown in Figure 4.2b, we learn a transformation that can jointly deal with 1) the domain-induced changes and 2) the capacity to recognize a new class never seen during the training phase.

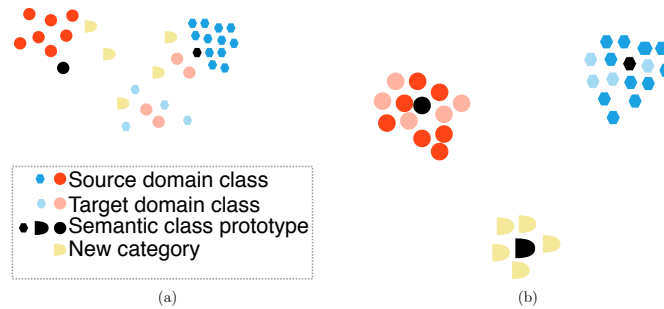


Figure 4.2: (a) Illustration of the domain shift problem where the feature points of the source domain and target domain are both shifted from each other and from their class prototype. (b) We aim to reunify samples from source and target domains in a common invariant space around their semantic prototype and to apply the learned model to new categories never seen before. Different classes are represented by different colors and shapes.

In this thesis, for the DA setting, the source domain data  $\mathcal{Z}^S$  is made of posed Ekman’s FEs, while for the target domain data  $\mathcal{Z}^T$  is made of Ekman’s with spontaneous FEs. The main problem is that the classes come from two different distributions (posed and spontaneous), resulting a domain-shift problem. For the ZSL setting, the source domain classes  $\mathcal{Z}^S$  are Ekman’s emotions with posed and spontaneous FEs, while the target the target domain classes  $\mathcal{Z}^T$  are mental state FEs. In the ZSL setting, the source and the target domains are disjoint,  $\mathcal{Z}^S \cap \mathcal{Z}^T = \emptyset$  and they could be coming either from the same distribution or from a different one. Therefore, ZSL could also suffer from the domain-shift problem.

**Challenges.** As we rely on designing a semantic embedding space to induce domain-invariance and to transfer knowledge from seen to unseen classes, it is important to impose some constraints on the embedding space structure as there is no guarantee that visual and semantic information could conform well. The fact is that when encoding the attribute vector  $\mathbf{a}_k$  into a semantic vector  $\mathbf{y}_k$  through the NLP or the DWE models,  $\mathbf{y}_k$  can capture semantic relationships among classes but they are totally blind to the visual domain. That is, they do not have to follow the same distribution as their visual features. For instance, Qiao et al. 2017 proves that the semantic similarity does not necessarily correspond to the visual similarity (semantic-visual gap). The *visual-semantic discrepancy* is an issue because it leads to inferior

classification or clustering performances and to domain-shift [Qiao et al. 2017, Roy et al. 2018, Gune et al. 2018].

Moreover, due to the within-class variations in visual samples, different visual signatures of the same class are obtained and thus together with the domain-shift problem, the mapping from the visual to the semantic space becomes non-trivial in which a single class prototype (center) could be shifted from many of those visual signatures. In addition, a simple direct mapping from the visual to the semantic space using regressors suffers from the hubness problem [Lazaridou et al. 2015b; Shigeto et al. 2015]. Hubs are vectors that tend to be near a high proportion of samples, pushing their correct labels away, so that, a few unseen or seen class prototypes become the nearest neighbours of many data points (an illustration is presented on Figure 4.12).

As a consequence, a compatibility constraint between the image feature and the class embeddings has to be imposed. In this thesis, we tackle this challenge by designing an embedding space that stands on re-aligning the semantic representation using the visual feature probability distribution. In this way we guarantee the fitting of the visual and semantic information in the new aligned embedding space. The aim is to enrich the link between visual features and their corresponding semantic information while preserving the intrinsic geometrical Euclidean structure to induce the discriminative power. Indeed, the aligned semantic representation becomes aware of the visual feature structure and therefore the knowledge of the semantic space becomes consistent with that of the visual feature space.

## 4.3 Related Works

### 4.3.1 Domain Adaptation

The main objective of domain adaptation is to leverage features from a labeled source domain and learn a classifier or embedding space for a target domain, with a similar but different data distribution. Recent domain adaptation methods utilize CNN approaches trained to minimize a classification loss and to maximize the domain-invariance. The domain-invariance is achieved either through a discrepancy loss [Ajakan et al. 2014] that reduces the shift between the two domains, or through an adversarial loss [Pineiro 2018] which encourages a common feature space with respect to a discriminator. Many previous works focus on learning a metric measure ([Bellet et al. 2013] for a survey) to reduce the shift. Herein, in order to put our work in context, we begin by describing the general domain adaptation model associated with the linear metric setting and then related to a metric learning setting based on deep learning.

**Domain Adaptation Method using Linear Setting (DA-LS)** [Saenko et al. 2010]. The goal is to learn a linear transformation using a labeled training data and then to utilize the learned similarity function in a classification or clustering algorithm.

Assume there are two domains  $\mathcal{S}$  (source) and  $\mathcal{T}$  (target). Given the feature vectors  $\mathbf{x} \in \mathcal{S}$  of size  $d_{\mathcal{S}}$  and  $\mathbf{y} \in \mathcal{T}$  of size  $d_{\mathcal{T}}$ , a linear transformation  $W$  of size  $d_{\mathcal{S}} \times d_{\mathcal{T}}$  from  $\mathcal{S}$  to  $\mathcal{T}$  can



be established. For instance it can be achieved using the inner product similarity function between  $\mathbf{x}$ ,  $W$  and  $\mathbf{y}$  which is defined as:

$$\underset{W}{\text{Sim}}(\mathbf{x}, \mathbf{y}) = \mathbf{x}^\top W \mathbf{y}. \quad (4.1)$$

The idea is that the shift can be approximated with an arbitrary linear scaling and rotation of the feature space. The aim is to recover this transformation by leveraging labeled data consisting of similarity and dissimilarity constraints between points in the two domains. Since the matrix  $W$  corresponding to the metric is symmetric positive semi-definite, thus it can be seen as mapping samples coming from two different domains into a common invariant space, in order to learn and classify instances more effectively across domains.

The transformation learning typically used is as:

$$d_W(\mathbf{x}, \mathbf{y}) = (\mathbf{x} - \mathbf{y})^\top W (\mathbf{x} - \mathbf{y}). \quad (4.2)$$

Now by factorizing  $W$  as  $W = G^\top G$  we can equivalently view the distance  $d_W$  between points  $x$  and  $y$  as  $(G\mathbf{x} - G\mathbf{y})^\top (G\mathbf{x} - G\mathbf{y})$ , that is,  $d_W$  is the squared Euclidean distance after applying the linear transformation specified by  $G$ . The transformation  $G$  maps data points from both domains into an invariant space.  $W$  and  $G$  are learned using a metric learning algorithm [Davis et al. 2007]. For comparison purpose, we use the Linear Setting as a way for solving domain adaptation.

**Similarity-Based Classifier Network via Deep Learning for DA** [Pinheiro 2018]. The model is referred as Similarity-Based Classifier Network (SimNet), and it is composed of two different components: (i) the domain-invariance component based on an adversarial loss, which forces the features of both domains,  $f(\mathbf{x}_S)$  and  $f(\mathbf{y}_T)$ , to be as indistinguishable as possible and (ii) a classifier based on a set of prototypes,  $\mu_c$  (one for each category  $c \in \{1, 2, \dots, C\}$ ). The prototypes are vector representations that are representative of each category that appears in the training dataset. For example, Pinheiro first builds synthetic images that correspond to object categories. Then the encoding of each of those synthetic images for each category is considered as class prototypes (centers in the embedding space). Different from them, we consider as prototype  $\mathbf{y}$  the encoding via NLP of the attribute vector  $\mathbf{a}$  of each class.

SimNet is based on the assumption that there exists an embedding for each category such that all the points of the category aggregate around it, independently of its domain. An inference is then performed on a test image by simply finding the most semantically similar prototype. The classifier is composed of  $C$  different prototypes, one per category. Each prototype represents a general embedding for a category, incorporating all its variations. Each prototype is represented by a  $m$ -dimensional vector  $\mu_c \in \mathbb{R}^m$ , which is encoded from the synthetic images using a CNN  $g(\cdot)$  with trainable parameters  $\theta_g$ . The prototype of each class is computed by the average representation of all source samples belonging to the category  $c$  as:

$$\boldsymbol{\mu}_c = \frac{1}{|X^c|} \sum_{\mathbf{x}_i^S \in X^c} g(\mathbf{x}_i^S), \quad (4.3)$$

where  $X^c$  is the set of all images in the source domain labeled with category  $c$ . Similarly, the input images (from either domain) are represented as an  $n$ -dimensional vector.  $f(\cdot)$  is based on a CNN which encodes the image features into an  $n$ -dimensional vector, parameterized by  $\theta_f$ . Then a similarity metric between images and prototypes is learned to predict which of the prototypes (and therefore categories) best describes a given input. The similarity between an input image  $\mathbf{x}_i$  and a prototype  $\boldsymbol{\mu}_c$  is defined as a bilinear operation:

$$h(\mathbf{x}_i, \boldsymbol{\mu}_c) = f_i^T W \boldsymbol{\mu}_c, \quad (4.4)$$

with  $W \in \mathbb{R}^{d \times m}$  being the trainable parameters.  $W$  is an unconstrained bilinear similarity operator, and it is not required to be positive or symmetric.

The model is trained to discriminate the target prototype  $\boldsymbol{\mu}_c$  from all other prototypes  $\boldsymbol{\mu}_k$  (with  $k \neq c$ ), given a labeled image. The outputs of the network are interpreted as class conditional probabilities by applying a softmax operator over the bilinear operator.  $\theta = \{\theta_f, \theta_g, W\}$  represents the set of all trainable parameters of the model. Learning is achieved by minimizing the negative log-likelihood, over all labeled samples as:

$$\mathcal{L}_{class}(\theta) = - \sum_{(\mathbf{x}_i, l_i)} \left[ h(\mathbf{x}_i, \boldsymbol{\mu}_{l_i}) - \log \sum_k e^{h(\mathbf{x}_i, \boldsymbol{\mu}_k)} \right]. \quad (4.5)$$

Domain confusion is achieved with a domain discriminator  $D$ , parameterized by  $\theta_d$ . The discriminator classifies whether a data point is drawn from the source or the target domain, and it is optimized following a standard classification loss:

$$\mathcal{L}_{disc}(\theta, \theta_d) = - \sum_i^{n_S} \log D(f(\mathbf{x}_i^S)) - \sum_i^{n_T} \log(1 - D(f(\mathbf{x}_i^T))). \quad (4.6)$$

The model is trained to jointly maximize the domain confusion and infer the correct category on the source samples, through the similarity-based classifier. Therefore, the final goal is to optimize the following mini-max objective:

$$\min_{\theta_f, \theta_g, W} \max_{\theta_d} \mathcal{L}_{class}(\theta_f, \theta_g, W) - \lambda \mathcal{L}_{disc}(\theta_f, \theta_d), \quad (4.7)$$

where  $\lambda$  is a balance parameter between the two losses.

At inference time the prototypes are computed a priori following eq. (4.3). Then we compute the similarity between a target-domain test image and each prototype. The label that best matches the query is output.

**Fine-tuning Based Method for DA.** Recently, studies [Jung et al. 2015, Ding et al. 2017; Kaya et al. 2017; Xu et al. 2018; Zhang et al. 2018] have investigated deep learning approaches for training facial expression classifiers for knowledge transfer via fine-tuning a pre-trained neural network model. The main concern studied so far is how to relatively train a deep network for expression recognition using a target domain by fine-tuning a CNN model that has been already trained using a source domain with a large labeled dataset designed for a same/different task. Despite the fact that the fine-tuning can partially mitigate the small dataset issue, the performance is still not high since the deep features probably contain irrelevant information from the source pre-trained domain.

### 4.3.2 Zero Shot Learning

Diverse techniques for reducing the number of necessary labelled training images for object detectors have been developed in the recent years. Some of them still require at least some labeled training examples to detect future object instances such as the one shot learning [Lake et al. 2011, Cho et al. 2014] which learns new objects from only a few examples. Nonetheless, zero shot learning is among the recent methods that require no training examples about the target classes. Existing ZSL methods differ according to the used semantic spaces: either visual attributes, word embedding, or image sentence descriptions. ZSL was firstly developed by using an intermediate layer which acts as an embedding space using visual attributes [Farhadi et al. 2009, Lampert et al. 2009, Parikh and Grauman 2011], which describe the visual appearance of the concept or instance by assigning labelled visual properties easily transferable from seen to unseen classes. An example is shown in Figure 4.3. Recently, the visual attributes have been replaced by word embedding [Lampert et al. 2013, Akata et al. 2015, Qiao et al. 2017] and later on by image sentence descriptions [Mao et al. 2015, Reed et al. 2016].

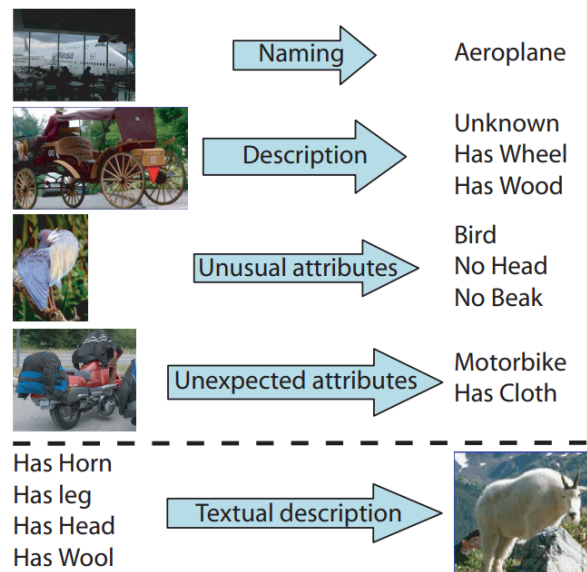


Figure 4.3: An example of visual attribute description. Adapted from [Farhadi et al. 2009]

**Information Transfer by Attribute Sharing for ZSL.** Lampert et al. 2009 provided one of the first studies on the problem of object classification when the training and test classes are disjoint. The authors tackle the problem by introducing an attribute-based classification. Their method performs object detection based on a human-specified high-level description of the target classes instead being based of training images. The description consists of semantic attributes, like shape, color, texture or even geographic clues. For example, as shown in Figure 4.4, an animal “Otter” can be described using the attribute vector “black, not white, brown, no stripes, live in water, eat fish” while “Zebra” can be described as “black, white, not brown, has stripes, does not live in water, does not eat fish”. The attribute vector can be either binary or continuous. The attributes deliver an intermediate layer in the classifier cascade. They empower the system to detect classes, for which it has not seen a single training example. Such properties can be pre-learned, *e.g.* from image datasets unrelated to the current task. Eventually, new classes can be detected based on their attribute representation, without the need for re-training the classifier.

|                   |     |   |  |
|-------------------|-----|---|--|
| <u>otter</u>      |     |    |    |
| black:            | yes |   |  |
| white:            | no  |   |  |
| brown:            | yes |   |  |
| stripes:          | no  |   |  |
| water:            | yes |   |  |
| eats fish:        | yes |   |  |
| <u>polar bear</u> |     |  |  |
| black:            | no  |   |  |
| white:            | yes |   |  |
| brown:            | no  |   |  |
| stripes:          | no  |   |  |
| water:            | yes |   |  |
| eats fish:        | yes |   |  |
| <u>zebra</u>      |     |  |  |
| black:            | yes |   |  |
| white:            | yes |   |  |
| brown:            | no  |   |  |
| stripes:          | yes |   |  |
| water:            | no  |   |  |
| eats fish:        | no  |   |  |

Figure 4.4: A description by high-level attributes allows the transfer of knowledge between object categories: after learning the visual appearance of attributes from any class with training examples, one can detect also object classes that do not have any training images, based on which attribute description a test image fits best. Adapted from [Lampert et al. 2009]

Next, we describe how conventional classifiers are not capable to categorize a new instance, and how the use of attributes allows to transfer information between object classes either using *Direct Attribute Prediction (DAP)* or *Indirect Attribute Prediction (IAP)* based on the study of [Lampert et al. 2009].

Typically, conventional classifiers learn one parameter or representation vector  $\theta_k^S$  for each training class  $z_k^S$  as shown in Table 4.2. Since the classes  $z_1^T, z_2^T, \dots, z_K^T$  are not present during the training phase, thus no parameter vector  $\theta_k^T$  can be derived for them. Therefore, it is implausible to make predictions about these classes. To make predictions about novel classes, a pairing between classes in  $\mathcal{Z}^S$  and  $\mathcal{Z}^T$  needs to be introduced. Since no training data for the unobserved classes is available, this pairing cannot be learned from samples, but has to be inserted into the system by human effort. To this end, serious constraints arise: 1) the

amount of human effort to define new classes should be minimal, otherwise collecting and labeling new training samples would be still a cheaper solution; 2) pairing data that demands common knowledge only is desirable over specialized expert knowledge, as the latter is often tough and costly to obtain. As humans are capable at supporting good prior knowledge about attributes that describe objects and therefore allow to discriminate between them, it is possible to collect the necessary information without a lot of expense. Moreover, as the attributes are assigned on a per-class basis instead of a per-image basis, the manual effort to add a new class is kept minimal.

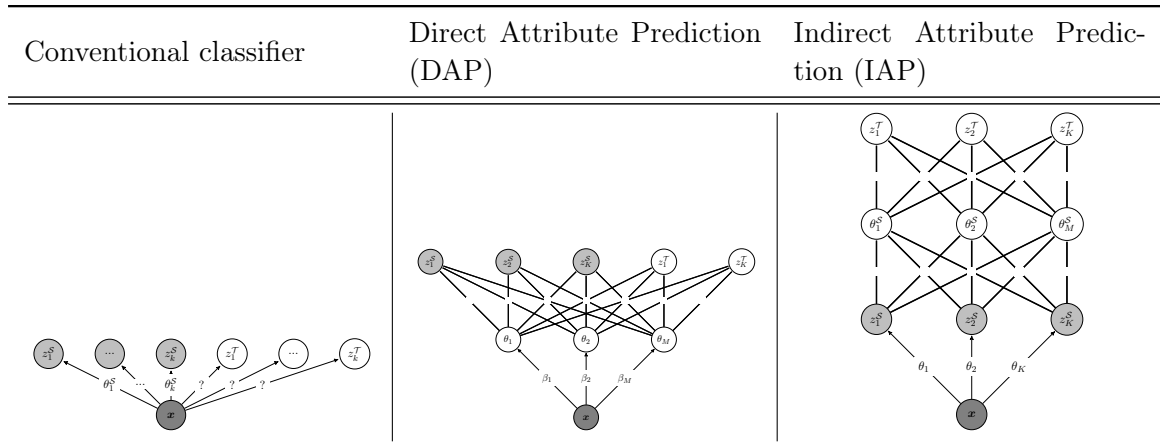


Table 4.2: Graphical representation of the across-class learning task: dark gray nodes are always observed, light gray nodes are observed only during training, while white nodes are never observed but must be inferred. An ordinary classifier learns one parameter  $\theta_k$  for each training class. It cannot generalize to classes  $z_k^T$  that are not part of the training set. In an attribute-based classifier (DAP) with fixed class–attribute relations (thick lines), training labels  $z_k^S$  imply training values for the attributes  $\theta_m$ , from which parameters  $\beta_m$  are learned. At test time, attribute values can directly be inferred, and this implies output class labels even for unseen classes. A multi-class based attribute classifier (IAP) combines both ideas: multi-class parameters  $\theta_k$  are learned for each training class. At test time, the posterior distribution of the training class labels induces a distribution over the labels of unseen classes by means of the class–attribute relationship.

The main principle of **DAP** and **IAP** is to learn a non-trivial classifier  $\theta : \mathcal{I} \mapsto \mathcal{Z}^T$  by transferring knowledge between  $\mathcal{Z}^S$  and  $\mathcal{Z}^T$  through  $\mathcal{Y}$ .  $\mathcal{I}$  is the image feature space,  $\mathcal{Z}^S$  and  $\mathcal{Z}^T$  are the domain and target label spaces and  $\mathcal{Y}$  is the attribute space.

**DAP**, illustrated in Table 4.2, uses an in between layer of attributes to decouple the images from the layer of labels. During the training, the output class label of each training sample induces a deterministic labeling of the attribute layer. Consequently, any supervised learning method can be used to learn per-attribute parameters  $\beta_m$ . At test time, this allows the prediction of attribute values for each test sample, from which the test class label is inferred. Note that the classes during testing can differ from the classes used for training, as long as the coupling attribute layer is determined in a way that does not require a training phase.

**IAP**, depicted in Table 4.2, uses the visual attributes to transfer knowledge between classes, but the attributes form a connecting layer between the two layers of labels, one for classes that are known at training time and one for classes that are not. As a consequence, the training phase of IAP is an ordinary multi-class classification while at test time, the predictions for all training classes induce a labeling of the attribute layer, from which a labeling over the test classes can be inferred.

The conflict between both approaches lies in the relationship between the training and the test classes. Directly learning the attributes ends up in a network where all classes are treated equally. When the class labels are inferred at test time, the decision for all classes are based on the attribute layer only. The training classes are potential output classes during testing and as a result this can introduce a bias. However, deriving the attribute layer from the label layer instead of from the samples can be considered as a regularization step that creates only sensible attribute combinations.

The main drawbacks of these methods is the need to visually describe the target domain classes (such as using AU). In some scenarios, it is very difficult to describe fine feature displacement among different classes, which is the case with mental state classes we are dealing with and no AUs are available. As a consequence, we cannot validate these approaches. Thus, we may seek for another bridge for transferring information such as using the class name and thus establishing class embedding.

**Information Transfer via Word Embedding for ZSL.** Recently, unsupervised training processes based on DWE in particular word2vec and GloVe have been studied [Changpinyo et al. 2016; Frome et al. 2013; M. Norouzi and Dean. 2013; Norouzi et al. 2014; Socher and Ng. 2013; Xian et al. 2016; Yang and Hospedales 2015; Zhang et al. 2017], as an encouraging alternative for visual-attributes vectors towards fully automatic ZSL. It has been shown that a visual attribute space is often more qualified than a word vector space because the annotators implicitly infuse the visual attribute vectors with visual information based on their knowledge and experience of the concepts. A good performance is obtained at the price of more manual attribute annotations for each class. However, the word vector semantic space using the class name is cheaper to obtain with minimal efforts. Therefore it is the right choice to do for recognition of unseen classes especially when it is not possible to annotate those unseen classes with a visual attribute vector due their fine-grained feature displacements.

In our work, for ZSL setting, we rely on the name of the classes and we use it to encode the similarity between them. However, the main problem with DWE is that they do not cover visual information and they are limited to pure textual representations, because they are trained to maintain the semantic relation of concepts from large text corpora. They only capture the semantic link between different classes. For instance, the concepts “violin” and “piano” are firmly linked in the semantic sense, though their visual appearances are absolutely distinct. Therefore, the semantic correlation does not necessarily conform to the visual similarity which in turn leads to visual-semantic discrepancy. Using the class names requires to solve this problem.

Few works have proposed to improve ZSL bases on DWE by associating it with visual

cues. For instance a visual word2vec [Kottur et al. 2016] is trained by adding abstract scenes to context. Other works propose language models to predict the visual representations jointly with the linguistic features [Lazaridou et al. 2015a]. A recent class of methods attempt to impose a congruity constraint between visual and semantic embeddings by aligning geometrical properties of the latter onto the former or by using the visual space as the semantic space or by embedding a shared joint space that allows knowledge transfer. For instance, seminal works [Mensink et al. 2014, Kodirov et al. 2015, Long et al. 2016 and Qiao et al. 2017] attempt to impose a constraint between visual and semantic embeddings by aligning their geometrical properties. Zhang et al. 2017 proposes to learn a deep embedding model for zero-shot learning by using the visual space as the embedding space instead of embedding it into a semantic space or an intermediate space. Qiao et al. 2017 proposes a Visually Aligned Word Embedding (VAWE) algorithm that uses the triplet hinge loss function to re-align the word embeddings with visual information and to alleviate the visual-semantic discrepancy. Changpinyo et al. 2016 proposes Synthesized Classifiers (SynC) for zero-shot learning that align the semantic space that is derived from external information to the model space that concerns itself with recognizing visual features via forming a weighted graph. Frome et al. 2013 proposes a Deep Visual Semantic Embedding (DeViSE) trained to identify visual objects using both labeled image data as well as semantic information gleaned from unannotated text. Norouzi et al. 2014 proposes the Convex Combination of Semantic Embeddings (ConSE) approach for constructing an image embedding system from any existing n-way image classifier and a semantic word embedding model, which contains the n-class labels in its vocabulary. Yang and Hospedales 2015 proposes Multi-Domain and Multi-Task Learning (MTL) framework where the core concept is a semantic descriptor for tasks or domains. Xian et al. 2016 proposes a Latent Embedding (LatEm) model for learning a compatibility function between image and class embeddings. It augments a bilinear compatibility model by incorporating latent variables. Roy et al. 2018 proposes a Visually-Driven Semantic Augmentation (VdSA), a ZSL approach that augments the semantic information coming from attributes such as DWEs with that of the visual patterns which are distilled from data by means of soft labels

In contrast, we aim at re-aligning the semantic space using the visual feature distribution so that the semantic structure becomes similar to that in the visual domain rather than applying a context prediction objective across visual and semantic domains.

#### 4.4 The Proposed Model for DA-FER and ZS-FER

The general structure of our unified model for DA-FER and ZS-FER is shown in Figure 4.5. It contains one sub-network (the visual model via a CNN) to obtain the visual signature  $\mathbf{x}$  and its feature probability distribution  $\boldsymbol{\alpha}$  by learning  $f_{\theta_f}(\cdot)$ . It contains a second sub-network (the semantic model via a NLP) to obtain the semantic embedding vector  $\mathbf{y}$  by learning  $g_{\theta_g}(\cdot)$ . We formulate another sub-network  $h(\cdot)$  (the re-alignment model) that takes as inputs  $(\boldsymbol{\alpha}, \mathbf{y})$  and outputs a richer re-aligned semantic embedding vector  $\mathbf{sa}$ . An Euclidean embedding learning is done between  $\mathbf{x}$  and  $\mathbf{sa}$ . We choose the Euclidean distance as the proximity measure because the representations  $\mathbf{sa}_i$  and  $\mathbf{x}_i$  are continuous and dense vectors.

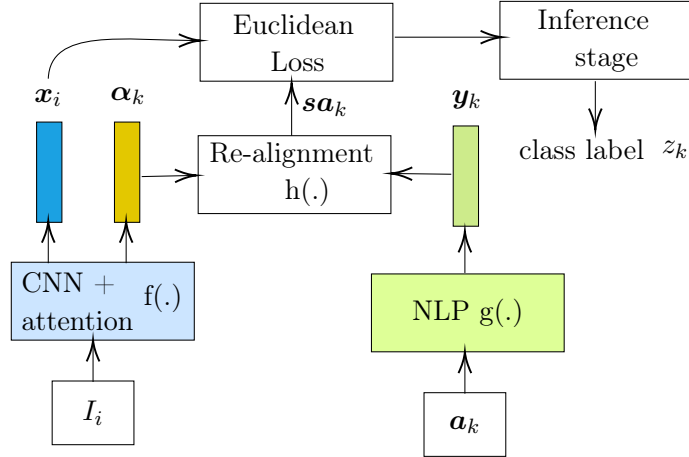


Figure 4.5: Global overview of the DA-FER and ZS-FER methods.  $\mathbf{x}$  is the encoded visual signature obtained via a deep CNN model.  $\boldsymbol{\alpha}$  is the visual feature distribution obtained via the Soft Attention Model.  $\mathbf{y}$  is the semantic class prototype that describes the class label or its visual attribute description obtained via an NLP model.  $\mathbf{sa}$  is the re-aligned semantic vectors. The whole system is trained in an end-to-end fashion using the source domain information and produces the estimated class label  $z$ .

The model relies on computing the feature probability distribution  $\boldsymbol{\alpha}_i$  for an image  $I_i$  in order to re-align the semantic vector  $\mathbf{y}_k$ . To do that, we propose a Soft Attention (SFA) model which is composed of two soft attention modules. The first one focuses on capturing the feature probability distribution while the second one focuses on the location probability distribution with respect to a certain class.

The intuition behind this approach is to take into account the difference of the two distributions in the visual and semantic spaces and to align one with respect to the other. For instance, suppose the visual feature probability distribution of a certain class (*e.g.* Surprised) follows a normal distribution while its semantic class prototype follows a beta distribution. Then, by mapping the normal distribution over the beta distribution, we lost the visual feature distribution. Now suppose that for two different classes (*e.g.* Surprised and Happy), they have exactly the same semantic distribution (*e.g.* beta), then by mapping their visual features over their semantic prototypes, all the feature points are going to collapse over a certain class prototype. Therefore, we assume that by mapping the image visual signature onto the semantic space, the overall distribution differences among different visual signatures belonging to different classes is lost in the new semantic space.

In order to deal with that, we capture for each image of a certain class, its feature probability distribution and we use it to align its semantic vector which supposed to be blind to any visual feature differences. By doing that, we are orienting the new semantic class prototypes of different categories to adapt themselves according to the feature probability distribution of their visual feature counterpart and therefore we reduce the visual-semantic gap. Another advantage of the aligning process through the feature probability distribution is that it allows the semantic space to keep the discriminability and separability properties of the visual signa-



tures. For instance, suppose that our visual model  $f(\cdot)$  is capable at discriminating between two classes composed of very subtle facial features, then using their feature probability distribution, we can tell the semantic space to respect their distribution in the new semantic space as well.

We demonstrate the advantage of using Soft Attention SFA for aligning those spaces and we validate our assumption using ablation studies. We use the probability distributions to describe the feature distributions, however other statistics are also exist such as median or the first- and second-order statistics. In the following sections we detail the model and the role of each of its components.

#### 4.4.1 The Visual Model $f(\cdot)$

The visual model computation process  $f_{\theta_f}(\cdot)$  establishes the visual signature  $\mathbf{x}_i$  of an image  $I_i$  and its feature probability distribution  $\alpha_i$  as shown in Figure 4.6. The designed visual model relies on two important modules: 1) a deep CNN and 2) double soft attentions modules.

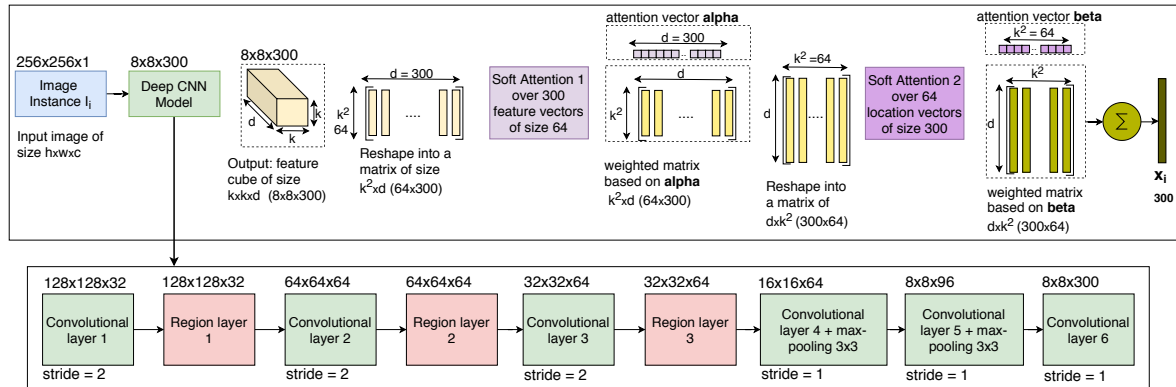


Figure 4.6: Zoom on the visual model modules. Our model is composed of a deep CNN module and another two sequential Soft Attention modules, one for discovering the feature probability distribution and another for discovering the location probability distribution. Our architecture can learn which information to emphasize or suppress and provides a compact visual signature vector.

#### Reason Behind the Visual Model Architecture and Role of the its Sub-Modules:

Spontaneous emotions can have very similar facial appearances and can be characterized with only some local facial muscle activations on sparse facial regions with subtle spatial deformations. In particular, emotional categories such as mental states are very challenging due to the subtle inter-class variances and large intra-class differences between different categories. Therefore, the main challenge of such subtle and local fine-grained emotional categories is *how to locate discriminative regions and how to select the most discriminant features from those regions*. Herein, we aim at designing a deep CNN architecture that identifies and focuses with higher importance on critical active facial regions and model subtle FE differences contained in those regions. We want to establish image signatures that contain unique cues so that their

visual embedding space is well designed to distinguish between fine-grained emotions. Therefore, to capture subtle differences in local regions, a Region Layer (RL) [Zhao and Zhang. 2016] is incorporated. To select the most relevant feature map among the extracted feature maps, a feature soft attention module (SFA-1,  $\alpha$  vector) is designed and finally to locate the most active facial regions among the most discriminant feature maps, a location soft attention module (SFA-2,  $\beta$  vector) [Sharma and Salakhutdinov. 2015] is employed.

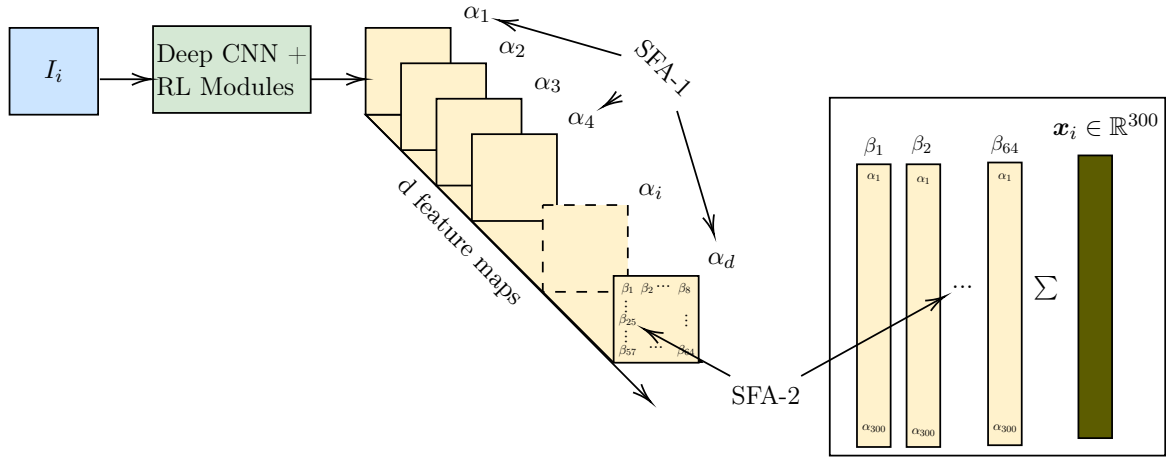


Figure 4.7: Module arrangements: a deep CNN integrated with region layers (RL), followed by a feature soft attention (SFA-1) and a location soft attention (SFA-2). The whole model produces  $x_i$  for an image  $I_i$ .

**Module Arrangement:** We show in Figure 4.7 an intuitive way to arrange our models fitting our purposes. First, we would like to extract representative features from the input image, therefore, a deep CNN module is designed first. Second, within this module, RL modules are included to extract subtle features from local regions. Third, in order to focus on the most discriminant feature maps (the outputs of the deep CNN and RL modules) and to weight their importance, a feature soft attention module (SFA-1) is added just after. Finally, to select the most active regions among the most active feature maps and to pool them into a single feature vector, a location soft attention module (SFA-2) is added.

**Model Architecture:** An outline of the proposed architecture that integrates a deep CNN model alongside the SFA modules is shown in Figure 4.6. The deep CNN model is composed of, from left to right, a linear convolutional layer-1 filtering on a face image, followed by an RL [Zhao and Zhang. 2016] with a patch size half of the input size. All filter sizes are  $3 \times 3$  except for the last convolutional layer-6 of size  $1 \times 1$ . Each layer is followed by an Instance Normalization (IN) [Huang and Belongie 2017] for addressing internal covariate shift problem and by a Parametric Rectified Linear Unit (PReLU) activation [He et al. 2015b] and a stride by 2 for down sampling except for the convolutional layers-4, 5, and 6. Convolutional layers 4 and 5 are followed by a max pooling with  $3 \times 3$  window. The output of convolutional layer-6 of shape  $8 \times 8 \times 300$  is passed to the attention module SFA-1.

The deep CNN model takes as input an image  $I_i$  of size  $h \times w \times c$  and produces a feature cube  $\mathcal{C}$  of size  $k \times k \times d$ . The feature cube is reshaped into  $d$  vectors of  $k^2$ -dimension and it

is passed through the first soft attention module SFA-1, which produces a feature probability distribution  $\alpha_i \in \mathbb{R}^d$ . The feature softmax  $\alpha_i$  weights each feature vector of  $k^2$  spatial location dimensions in  $d$  feature maps such that it focuses its attention on the corresponding  $d$  vectors that the model discovers as important. The weighted feature vectors based on  $\alpha_i$  are referred as weighted feature slices in a feature cube ( $C_i^\alpha$ ) and are passed to the second soft attention module SFA-2. It takes as an input the  $k^2$  weighted vertical feature slices  $\in \mathbb{R}^d$  and computes the expected value  $x_i \in \mathbb{R}^{d=m}$  (eq. (4.14)), referred as visual signature vector, according to the location softmax  $\beta_i \in \mathbb{R}^{k^2}$ . The visual signature  $x_i$  is mapped later over an aligned semantic class prototype  $sa_i \in \mathbb{R}^m$ .

**Region Layer Module:** To capture details of local regions Taigman et al. 2015 proposed several Locally Connected Layers (LCN)s that confine different filters to each pixel location without weight sharing. However the main disadvantage of LCNs is the need of a large number of parameters. In conventional CNNs, the kernels are shared across the entire image and for facial expression recognition this spatial stationarity assumption would not hold because different regions have different local statistics. Zhao and Zhang. 2016 proposed Region Layer (RL) as an alternative design between LCN and CNN. RL confines the same filter for each local region and produces an individually updating of the weights on each region. In this study, we integrate RL within the deep CNN architecture. In order to illustrate how RL works, we present a demonstration in Figure 4.8.

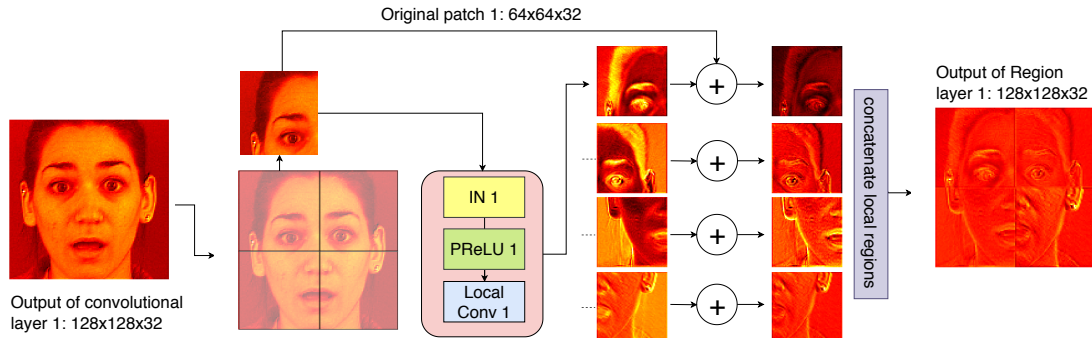


Figure 4.8: A feature map is output from Conv1, and uniformly divided into  $4 \times 4$  patches. Each  $60 \times 60$  pixel patch is processed with Instance Normalization (IN), followed by Parametric Rectified Linear Unit (PReLU) activation and then locally convolved with  $3 \times 3$  filters with the same number of Conv1 channels. Afterwards, each original patch is re-weighted by adding it to the convolved one. The output of the RL is the concatenation of all re-weighted patches.

Figure 4.8 details the RL architecture and demonstrates an example by considering the first RL. After the standard convolutional layer 1, the feature maps are partitioned into equal size patches and followed by the IN-1, the PReLU-1 and a regional local convolution (Conv1). The new output is added over the original patches and the final output is the concatenation of all the refined patches.

In our model, we take advantages of PReLU (eq. (4.8)) over ReLU, as its coefficient regarding the negative part is not constant and is adaptively learned. The motivation behind using PReLU is to avoid zero gradients.

$$\text{PReLU}(m_i) = \begin{cases} m_i & \text{if } m > 0 \\ \gamma_i m_i & \text{if } m \leq 0, \end{cases} \quad (4.8)$$

where the  $\gamma$  parameter in PReLU is learnable and  $m_i$  represents the feature map.

**Feature Soft Attention Module (SFA-1):** The significance of attention mechanism has been studied extensively in the previous literature [Mnih et al. 2014, Jaderberg et al. 2015]. It is basically inspired from the human visual system on exploiting a sequence of partial regions and selectively focusing on salient parts in order to capture the visual structure better [Larochelle and Hinton 2010]. In this thesis, we design a feature soft attention module that emphasizes on most the informative facial features while suppressing less useful facial features. It works by selectively focusing on some parts of facial features extracted from the convolutional layer 6 in order to find where are the most discriminant features with respect to a certain class and to what degree each of these features contribute to the discriminability criterion. Feature SFA-1 outputs the feature probability distributions  $\alpha$ . Those probabilities weight the features based on their relevance importance. They provide the ability to capture the visual structure better and thus we can use  $\alpha$  to align the semantic vector  $y$ , resulting  $sa$ .

**Location Soft Attention Module (SFA-2):** Its role is to search for the most active facial regions for each class and to improve the representation of interests. It works by systematically focusing on small parts of the image in order to find what is the most important region with respect to a certain class. This location can change during the training process, as the model learns more about the relevant regions of the image with respect to its class label. Location SFA-2 outputs the location probability distribution  $\beta$  in which the resolution of a specific region centered at a particular location is high if it is relevant region and low if it is not a relevant region. Therefore, SFA-2 adds an additional dimension of interpretability, by giving us the ability to visualize (as the case shown in Figure 4.9) where network attends to extract its features from. An additional advantage of attention mechanism in general, is its ability of pooling features dynamically in order to obtain a compact feature vector, thus it is kind of advanced fusing mechanism for dimensional reduction.

Figure 4.9 visualizes the first feature map from each layer of the deep CNN architecture and shows the final attended regions where the most discriminative features are extracted.

The model for a surprised face  $I_i$  gives its attention using the softmax location  $\beta_i$  only over the mouth and the eyes, which are the most prominent regions of interest. This implies that the embedded visual signature  $x_i \in \mathbb{R}^{d=m=300}$  is informative and compact. This example shows the effectiveness of the RL and SFA modules to permit precise identification and to select of the regions of interest and their most discriminant features.

**Soft Attention Modules (SFA-1 and SFA-2) Computations:** Given the convolutional layer 6 feature maps and by considering them as a feature cube  $\mathcal{C}$  of size  $k \times k \times d$ . Let us reshape  $\mathcal{C}_i$  into a matrix  $\in \mathbb{R}^{d \times k^2}$ , with vector  $\mathbf{c}_j \in \mathbb{R}^{k^2}$ . In order to achieve the first goal which consists in computing the feature softmax  $\alpha_i$ , we design a function that computes the unnormalized relevance scores  $\mathbf{rs}_j$  based on eq. (4.9) using  $d$  vectors. Once all relevance

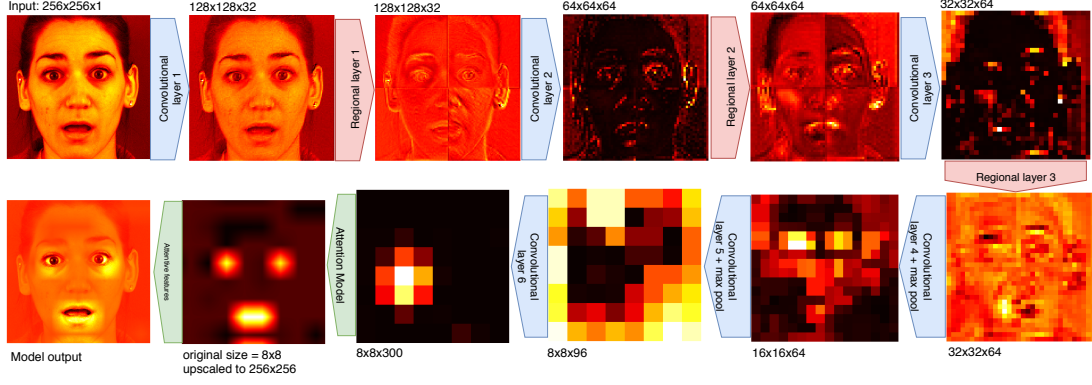


Figure 4.9: A demonstration for the attended locations for a surprised emotion.

scores are computed, they are normalized to obtain a probability distribution  $\alpha_i$  based on eq. (4.10). Then the feature cube  $\mathcal{C}_i$  is weighted using eq. (4.11), which results in  $\mathcal{C}_i^\alpha$ .

$$\mathbf{r}\mathbf{s}_j^\alpha = \tanh(W_j^\alpha \odot \mathbf{c}_j), \text{ where } j \in \{1, \dots, d\}, \quad (4.9)$$

$$\alpha_j = \frac{\exp(\mathbf{r}\mathbf{s}_j^\alpha)}{\sum_{j=1}^d \exp(\mathbf{r}\mathbf{s}_j^\alpha)}, \text{ such that } \sum_{j=1}^d \alpha_j = 1, \quad (4.10)$$

$$\mathcal{C}_i^\alpha = \alpha_i \odot \mathcal{C}_i, \text{ where } \mathcal{C}_i^\alpha \in \mathbb{R}^{d \times k^2} \text{ and } \alpha_i \in \mathbb{R}^{d=m}, \quad (4.11)$$

where the symbol  $\odot$  refers to the Hadamard product and  $W_j^\alpha$  corresponds to the weight mapping of the  $j$ th vector to  $\mathcal{C}_i$ .

In order to achieve the second goal, that is computing the location softmax  $\beta_i$ , we compute the relevance score based on eq. (4.12) using the  $k^2$  weighted feature slices  $\mathcal{C}_i^\alpha$ .  $\beta_i$  weights the regions of the input image that the model finds important. Afterwards, the relevance scores are normalized to obtain the probability distribution  $\beta$  based on eq. (4.13). Finally, we compute the expected value  $\mathbf{x}_i$  of the input  $(\mathcal{C}_i^\alpha)^\top \in \mathbb{R}^{k^2 \times d}$ , by taking the expectation over the weighted feature slices  $(\mathbf{c}_j^\alpha)^\top \in \mathbb{R}^d$  in different regions using eq. (4.14). The final visual feature vector representation for each input image  $I_i$  is  $\mathbf{x}_i \in \mathbb{R}^{d=m}$ .

$$\mathbf{r}\mathbf{s}_j^\beta = \tanh(W_j^\beta \odot (\mathbf{c}_j^\alpha)^\top), \text{ where } j \in \{1, \dots, k^2\}, \quad (4.12)$$

$$\beta_j = \frac{\exp(\mathbf{r}\mathbf{s}_j^\beta)}{\sum_{j=1}^{k^2} \exp(\mathbf{r}\mathbf{s}_j^\beta)}, \text{ such that } \sum_{j=1}^{k^2} \beta_j = 1, \quad (4.13)$$

$$\mathbf{x}_i = \sum_j^{k^2} \beta_j (\mathbf{c}_j^\alpha)^\top, \text{ where } \mathbf{x}_i \in \mathbb{R}^{d=m}. \quad (4.14)$$

4.4.2 The Semantic Model  $g(\cdot)$ 

A general scheme for the semantic model is shown in Figure 4.10. This model produces the semantic vector  $\mathbf{y}_k$  for each of the DA-FER and ZS-FER methods. Its core is the Natural Language Processing (NLP) module that captures the semantic similarities among word(s) and sentences.

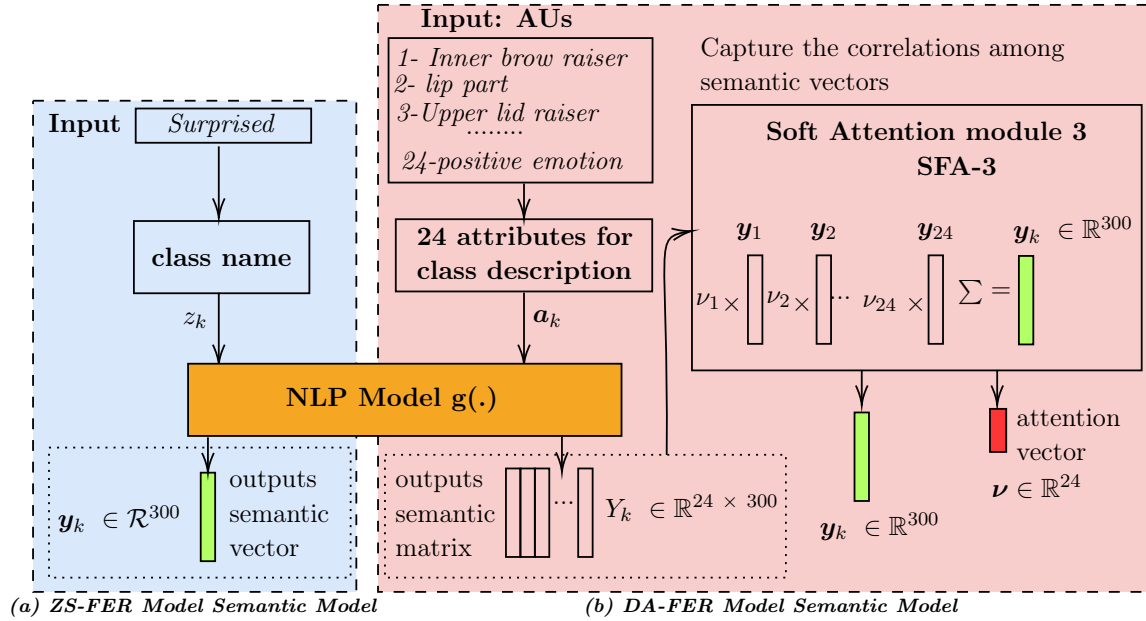


Figure 4.10: An outline of the proposed semantic representation for the DA-FER and ZS-FER methods respectively. An NLP model encodes the class name or its description into a continuous feature vector(s). For ZS-FER, the class name is used and thus a semantic vector  $\mathbf{y}_k \in \mathbb{R}^{300}$  is produced. For DA-FER, 24 visual attributes that describe each class are provided as an input and the model produces for each attribute entry a vector  $\mathbf{y} \in \mathbb{R}^{300}$ . A set of 24 semantic vectors are then fed to a soft attention model SFA-3 to encode the relationships between those vectors and to provides a compact semantic vector  $\mathbf{y}_k \in \mathbb{R}^{300}$ .

**Visual Attributes and Word Embedding for DA-FER and ZS-FER.** To represent any facial expression class in an attribute space, an attribute ontology has to be defined and each facial expression class needs to be related to an attribute vector  $\mathbf{a}_k$  that describe using linguistic sentences or words its visual spatial deformations.

For the DA-FER, since we deal with Ekman's basic emotions to transfer knowledge learned from posed expressions to spontaneous expressions, it is possible to use the linguistic description of Action Units (AUs) of a certain emotion as entry for  $\mathbf{a}_k$ . However, to avoid any extra effort while encoding AU occurrences for each class, we consider that each Ekman emotional class has a set of pre-defined AUs as shown in Table 4.3, even if some of these AUs for a certain class are actually not performed by a specific subject in some FE images. Afterwards, we establish the amount of presence of each semantic attribute entry for a given class by building a soft attention module (SFA-3) that captures the correlations among the  $\mathbf{a}_k$  entries.

| Emotion   | Involved AUs       |
|-----------|--------------------|
| Angry     | 4,7,10,17,23,24    |
| Disgust   | 4,9,10,17,24       |
| Fear      | 1,2,4,5,7,20,25,26 |
| Happiness | 6,12,25            |
| Sadness   | 1,4,6,15,17        |
| Surprised | 1,2,5,25,26        |

Table 4.3: Emotion in terms of prototypical facial AUs. Adapted from [Du et al. 2014a].

But AUs labelling for non-basic spontaneous expressions such as mental state expressions is very time consuming and difficult to obtain, since AU labels should be reviewed by trained experts. The requirement of the corresponding it prevents leveraging available databases for the ZS-FER. To overcome this aspect, most recent works explore the semantic word vector space [Akata et al. 2015; M. Norouzi and Dean. 2013; Y. Fu and Gong 2014] as an alternative to visual attributes in which only the class name is required. It is learned using unannotated textual large corpus for Natural Language Processing (NLP) tasks to capture the semantic relationships between different class names [Elhoseiny and Elgammal. 2013]. In this thesis, the textual description of class names is adopted for the *ZS-FER* model, due to the difficulty of coding the involved facial muscles for unseen mental state expressions.

**Language Model  $g(\cdot)$ .** The objective is to leverage semantic knowledge learned in the text domain to provide a semantically-meaningful vector representation of words and sentences. We adopt a pre-trained NLP model ( $g(\cdot)$ ) [Kiperwasser and Goldberg 2016] trained on the OntoNotes corpus, with the pre-trained GloVe word embeddings [Pennington et al. 2014] trained on the Common Crawl corpus. The input for  $g(\cdot)$  is a word (*e.g.* “angry”) or a sentence (*e.g.* inner brow raiser) and the output of  $g(\cdot)$  is a real-valued feature vector  $\mathbf{y}_k \in \mathbb{R}^{m=300}$  which can be used as a feature vector in many applications such as information retrieval or document classification. In this thesis,  $\mathbf{y}_k$  is used as a bridge for knowledge transfer. It is introduced to our system to permit external knowledge interrogation when no information about unseen classes are presented to the model during training.

Most word vector methods rely on the distance or angle between pairs of word vectors as the primary method for evaluating the intrinsic quality of such a set of word representations and thus to produce  $\mathbf{y}_k$ . However, recent methods rely on first encoding various dimensions of difference such as GloVe vectors [Pennington et al. 2014] and then using those vectors to capture further semantic and contextual relationships using bidirectional LSTM [Kiperwasser and Goldberg 2016].

**DA-FER Method: Designing the Semantic Space using Visual Attributes.** Each facial expression of a certain class is represented with a set of attributes using a linguistic description of AUs with some words. The total number of AUs we use is 16. We also add additional semantic attributes that describe if an emotion is “positive” (as the case of happiness) or “negative” (rest of the emotions) or “both” (surprised and neutral face may represent a positive or a negative expression). In addition, we describe a neutral face with the following

six semantic attributes : relaxed brows, relaxed lids, relaxed nose, relaxed lips, relaxed cheeks and relaxed jaw. Therefore, in total, we have 24 attributes that can be used to describe an emotion. The emotion class  $z_k$  of an image  $I_i$  is initially encoded with an attribute vector  $\mathbf{a}_k$  as shown in Table 4.4. For instance the  $\mathbf{a}_k$  vector for the *angry* expression is encoded as “brow lowerer, lid tightener, upper lid raiser, chin raiser, lip tightener, lip pressor and negative emotion”, while the rest of the vector is assigned to zero. Afterwards, each component  $\mathbf{a}_{k,j}$  in  $\mathbf{a}_k$ , where  $j \in \{1, \dots, 24\}$  is encoded via  $g(\cdot)$  into a real-valued feature vector  $\in \mathbb{R}^{300}$ . As a result, for each attribute vector  $\mathbf{a}_k$ , we compute a sparse matrix  $Y$  of dimensionality  $24 \times 300$ , where  $Y = [\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_j, \dots, \mathbf{y}_{24}]$ , where  $\mathbf{y}_k \in \mathbb{R}^{300}$ .

| Index | Description          | Action Unit | Angry | Disgust | Fear | Happiness | Sadness | Surprised | Neutral |
|-------|----------------------|-------------|-------|---------|------|-----------|---------|-----------|---------|
| 1     | Inner Brow Raiser    | AU1         |       |         | ✓    |           | ✓       | ✓         |         |
| 2     | Outer Brow Raiser    | AU2         |       |         | ✓    |           |         | ✓         |         |
| 3     | Brow Lowerer         | AU4         | ✓     | ✓       | ✓    |           | ✓       |           |         |
| 4     | Upper Lid Raiser     | AU5         |       |         | ✓    |           |         | ✓         |         |
| 5     | Cheek Raiser         | AU6         |       |         |      | ✓         | ✓       |           |         |
| 6     | Lid Tightener        | AU7         | ✓     |         | ✓    |           |         |           |         |
| 7     | Nose Wrinkler        | AU9         |       | ✓       |      |           |         |           |         |
| 8     | Upper Lid Raiser     | AU10        | ✓     | ✓       |      |           |         |           |         |
| 9     | Lip Corner Puller    | AU12        |       |         |      | ✓         |         |           |         |
| 10    | Lip Corner Depressor | AU15        |       |         |      |           | ✓       |           |         |
| 11    | Chin Raiser          | AU17        | ✓     | ✓       |      |           |         |           |         |
| 12    | Lip Stretcher        | AU20        |       |         | ✓    |           |         |           |         |
| 13    | Lip Tightener        | AU23        | ✓     |         |      |           |         |           |         |
| 14    | Lip Pressor          | AU24        | ✓     | ✓       |      |           |         |           |         |
| 15    | Lip Part             | AU25        |       |         | ✓    | ✓         | ✓       | ✓         |         |
| 16    | Jaw Drop             | AU26        |       |         | ✓    |           |         | ✓         |         |
| 17    | Relaxed Brows        | —           |       |         |      |           |         |           | ✓       |
| 18    | Relaxed Lids         | —           |       |         |      |           |         |           | ✓       |
| 19    | Relaxed Nose         | —           |       |         |      |           |         |           | ✓       |
| 20    | Relaxed Lips         | —           |       |         |      |           |         |           | ✓       |
| 21    | Relaxed Cheeks       | —           |       |         |      |           |         |           | ✓       |
| 22    | Relaxed Jaw          | —           |       |         |      |           |         |           | ✓       |
| 23    | Negative Emotion     | —           | ✓     | ✓       | ✓    |           | ✓       | ✓         | ✓       |
| 24    | Positive Emotion     | —           |       |         |      | ✓         |         | ✓         | ✓       |

Table 4.4: Basic emotional categories with their visual attribute vector.

**Dependencies Among Attribute Entries.** We consider the task of capturing the global semantic relationships among facial AUs that are structurally dependent. Initially, we consider that all the semantics description entries for a certain attribute vector  $\mathbf{a}_k$  have equal importance. Then, we estimate via integrating an attention mechanism module their contributions by learning a distributional vector  $\nu_k$  that focuses on how important each entry  $\mathbf{y}_{k,j}$ , where  $j \in \{1, \dots, j, \dots, 24\}$ , is. Therefore  $\nu_k$  is composed of a probability distribution vector of 24 entries, each one weighting a specific semantic attribute entry  $\mathbf{y}_{k,j}$ .

Encoding semantic description entry dependencies is critical for improving the semantic embedding manifold. To this end, we propose the SFA-3 module, similar to the SFA-2 module, that establishes the dependencies among the semantic attribute entries  $\mathbf{y}_{k,j}$ . The SFA-3 module, as shown in Figure 4.10, takes as an input the  $Y$  matrix and outputs a probability distribution  $\nu_k \in \mathbb{R}^{24}$ , that captures the contribution of each attribute. Then, a compact feature vector  $\mathbf{y}_k \in \mathbb{R}^{m=300}$  is computed which represents the expected value as demonstrated in eq. (4.17). Thus, our SFA-3 module provides a natural way to fuse information of the semantic description.



$$\mathbf{r} \mathbf{s}_j^\nu = \tanh(W_j^\nu \odot (\mathbf{y}_j)), \text{ where } j \in \{1, \dots, 24\}, \quad (4.15)$$

$$\nu_j = \frac{\exp(\mathbf{r} \mathbf{s}_j^\nu)}{\sum_{j=1}^{24} \exp(\mathbf{r} \mathbf{s}_j^\nu)}, \text{ such that } \sum_{j=1}^{24} \nu_j = 1, \quad (4.16)$$

$$\mathbf{y}_k = \sum_j^{24} \nu_j \cdot \mathbf{y}_j, \text{ where } \mathbf{y}_k \in \mathbb{R}^{d=m}. \quad (4.17)$$

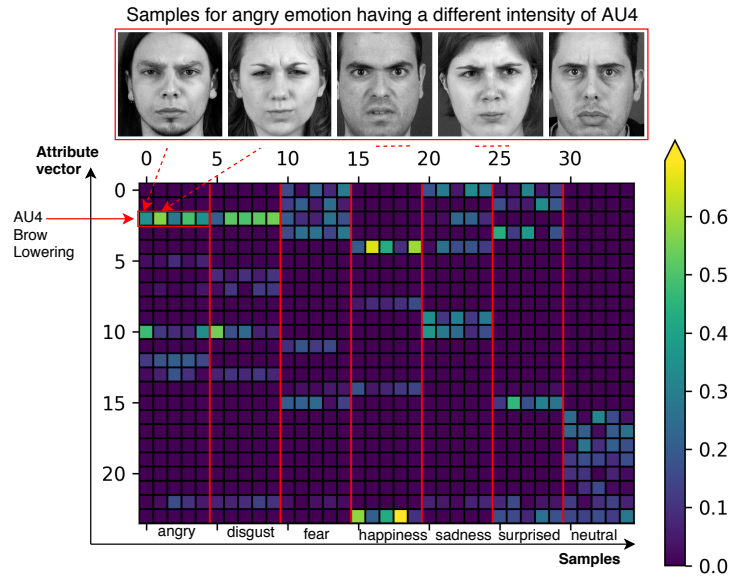


Figure 4.11: Attribute entry dependencies among samples either from the same or from different classes. The vertical red lines represent the limit among the different classes: angry, disgust, fear, happiness, sadness, surprised and neutral respectively. Five images per class are shown, *e.g.* the first five columns correspond to the angry class. Each column represents the probability distribution  $\nu_k$  that captures the percentage of importance of each semantic entry.

Figure 4.11 shows the probability distributions of the attention vectors  $\nu$  for samples taken from the same class or from different ones. The samples come from the DISFA database. For instance, the first five columns represent  $\nu_1, \nu_2, \nu_3, \nu_4, \nu_5$  for the angry emotional class. The third entry in each of these vectors  $\nu_{1,3}, \nu_{2,3}, \nu_{3,3}, \nu_{4,3}$ , and  $\nu_{5,3}$  corresponds to AU4 (shown in Table 4.4). We can see that the contribution of AU4 for different images of the same “angry” class varies. For instance our model demonstrates that  $\nu_{1,3}$  and  $\nu_{5,3}$  share the same probability value, around 0.3, while for  $\nu_{3,3}$  its value is around 0.2. Moreover, in our model  $\nu_{2,3}$  and  $\nu_{4,3}$  as they demonstrate obvious feature deformations corresponding to AU4, are represented with an intensity probability of 0.55 and 0.43 respectively. This proves the effectiveness of our method for capturing semantic relationships and weighting the entries of a specific attribute vector while jointly considering their image visual signature.

**ZS-FER: Designing the Semantic Space via Class Embedding.** When a specific class

description is very hard to obtain or annotate, specially for unseen mental state expressions, the class name can be used for exploring the semantic relationships among categories. Therefore, we can directly use  $g(\cdot)$  using the class name as an input to produce  $\mathbf{y}_k \in \mathbb{R}^{300}$ .

### 4.4.3 Semantic Space Re-alignment Process

We re-visit how the Euclidean distances between categories in the semantic space are related to those in the visual space. We aim at maximizing the correlations between the visual feature representation  $\mathbf{x}_i \in \mathbb{R}^{300}$  and its corresponding semantic representation  $\mathbf{y}_k \in \mathbb{R}^{300}$ . To overcome the visual-semantic discrepancy, we propose a non-linear function  $h(\cdot)$  eq. (4.18) to re-align  $\mathbf{y}_k$  using the probability distribution  $\alpha_i$  obtained from the visual information and we obtain  $\mathbf{sa}_i$ . Then, the projection of  $\mathbf{x}_i$  on  $\mathbf{sa}_k$  is learned using an Euclidean learning.

$$\mathbf{sa}_k = \tanh(\alpha_i \times \mathbf{y}_k) \quad (4.18)$$

### 4.4.4 Inference Stage

Let us consider an image  $I_i^{\mathcal{T}}$  coming from the target domain  $\mathcal{T}$ . The image feature signature  $\mathbf{x}_i^{\mathcal{T}}$  is computed via  $f(\cdot)$  and we obtain its feature softmax probability distribution  $\alpha_i^{\mathcal{T}}$ . Each semantic embedding prototype  $\mathbf{y}_k$  for each class  $z_k^{\mathcal{T}} \in \mathcal{Z}^{\mathcal{T}}$ ,  $k \in \{\text{angry, sadness, ..., affected, astonished, ...}\}$  is computed using  $g(\cdot)$ . Afterwards, it is aligned using  $h(\cdot)$  based on  $\alpha_i^{\mathcal{T}}$  to obtain  $\mathbf{sa}_k^{\mathcal{T}}$ . Using a k-NN algorithm, the nearest neighbour between the vectors  $\mathbf{sa}_k^{\mathcal{T}}$  and the vector  $\mathbf{x}_i^{\mathcal{T}}$  is picked up and assigned to its label class  $z_k^{\mathcal{T}}$ . By that, we condition the semantic representation for any coming instance based on its feature softmax distribution to refine its geometric space in order to alleviate the visual-semantic discrepancy.

## 4.5 Experimental Setup and Analysis Results

In this section, we conduct a set of experiments to validate our models for ZS recognition and DA applied to FER. First, we summarize the content of the databases used for the evaluation setting in Table 4.5 and then we compare our approaches with the most relevant works discussed in related work (Section 4.3). We split the evaluation into two sections, one for DA-FER and one for ZS-FER. In each section, quantitative, qualitative and ablation studies are performed to analyse which module plays the most important role.

### 4.5.1 DA-FER performance evaluation and analysis

**DA Data Protocol.** In order to evaluate and compare solutions to the domain shift problem, we propose a protocol using different available facial expression databases. In this way, we

| Database                          | Database Information  | Expressions   | # of images |
|-----------------------------------|---|---|-------------|
| CK+ [Lucey et al. 2010]           | 118 Subjects, multi-ethnicity,<br>81 females, 36 males,<br>No Occlusion,    | <b>Posed</b> ,<br><i>Basic expressions</i> ,<br>AUs coded         | 100,000     |
| MMI [Pantic et al. 2005]          | 30 Subjects, multi-ethnicity,<br>9 females, 21 males,<br>No Occlusion       | <b>Posed</b> ,<br><i>Basic expressions</i><br>AUs coded           | 18,637      |
| MUG [Aifanti and Delopoulos 2010] | 77 Subjects, Caucasian,<br>30 females, 47 males,<br>No Occlusion            | <b>Posed</b> ,<br><i>Basic expressions</i><br>AUs coded           | 70,654      |
| DISFA [Mavadati et al. 2013]      | 27 Subjects, multi-ethnicity,<br>12 females, 15 males,<br>Partial Occlusion | <b>Spontaneous</b> ,<br><i>Basic expressions</i><br>AUs coded     | 130,000     |
| DynEmo [Tcherkassof et al. 2013]  | 358 Subjects, Caucasian,<br>182 females, 176 males,<br>Partial Occlusion    | <b>Spontaneous</b> ,<br><i>Mental expressions</i><br>No AUs coded | 75,180      |

Table 4.5: Facial expression databases used for Da-FER and ZS-FER validations

establish a multiple-domain labeled dataset which provides a challenging environment category learning task and reflects partially the difficulty of real-world facial expression recognition. We build the source domain distribution with **Ekman’s posed facial expressions** by merging the following databases: CK+, MMI, and MUG. The total number of images used for training is 189,291. We build the target domain with **Ekman’s spontaneous facial expressions** using the DISFA database. We split it into 7 independent subjects with 38,690 images in total for the validation phase and 20 other independent subjects with around 91,310 images in total for the test phase. The total number of images per class in each domain is approximately balanced.

**Comparative algorithms.** We evaluate our domain adaptation approach DA-FER by applying it to a k-NN classification of emotion categories. We study domain shifts and compare our approach to several baseline methods: DA-LS and SimNet based methods (discussed in Section 4.3.1). In order to evaluate the performances of a pre-trained model, we use Face VGG-Net [Chatfield et al. 2014] and we refer to it as VGG-Net. We choose the 1000-dim last fully connected layer activations as features and we add on top of it a softmax layer for classification. The network is re-trained with a training set coming from the target domain itself.

**Quantitative Metrics.** To consider how the visual-semantic consistency affects the performances of the DA-FER, we measure the average neighbourhood overlap [Romera-Paredes and Torr 2015, Qiao et al. 2017]. That is the visual feature vector distance between two classes as the average distance between all pairs of visual features within those two classes. Also, this is equivalent of calculating the distance between their mean feature vectors. Expressly, the visual feature vector distance between two classes  $i$  and  $j$  is:

$$D_{i,j} = \|\mu_i - \mu_j\|_2, \quad (4.19)$$

where  $\mu_i$  is the mean feature vector for each class and  $\|\cdot\|_2$  is the  $L_2$ -norm. Similarly, the semantic distance between two classes can be calculated in the same way by replacing  $\mu_i$  and  $\mu_j$  with the semantic embeddings of classes  $i$  and  $j$ . The visual feature mean summaries the visual appearance of each class.

Now, let us define  $N_{vis}(i, K)$  and  $N_{sem}(i, K)$  as the sets including the  $K$  most similar classes to class  $i$  in the visual and semantic domains respectively. Afterwards each class  $i$ , its top- $K$  nearest classes in the visual domain are calculated using eq. (4.19). Similarly, we calculate the top- $K$  nearest classes of  $i$  in the semantic domain. The average neighbourhood overlap can be defined using eq. (4.20) as the average number of shared neighbours (out of the  $K$  nearest neighbours) for all  $C$  classes in semantic and visual domains. A value closer to  $K$  indicates that the embedding is more consistent with the visual domain.

$$\text{Consistency} = \sum_{i=1, \dots, C} |N_{vis}(i, K) \cap N_{sem}(i, K)| / C. \quad (4.20)$$

**Special Case: DA-FER model without the alignment step.** We consider the DA-FER method without considering the re-alignment step done via  $h(\cdot)$  and we keep the whole architecture the same. By that, we map the visual feature representation  $\mathbf{x}_i$  over their semantic class prototype  $\mathbf{y}_k$  using an Euclidean learning.

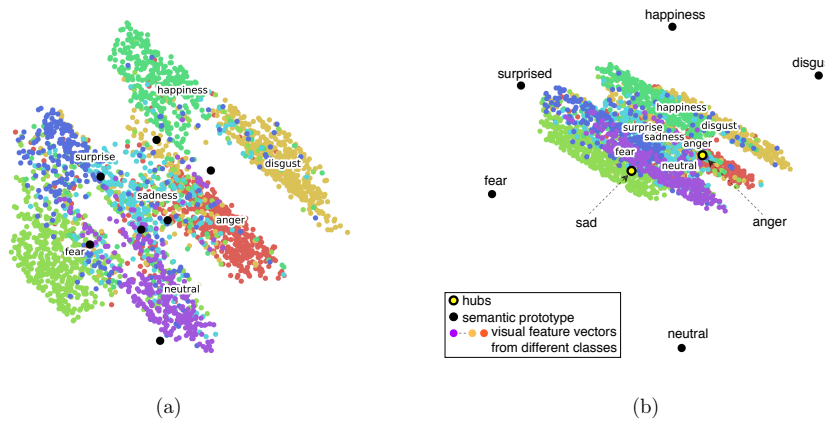


Figure 4.12: Feature representation for image visual signatures coming from different classes and projected over their semantic class prototype (black dot) using data coming from: *Source Domain* (a) and *Target Domain* (b).

In Figure 4.12a we plot  $\mathbf{x}_i$  for the source domain validation data and the mean vector of the semantic class prototype per class, which is represented as a black dot. In the same way, we plot in Figure 4.12b,  $\mathbf{x}_i$  for the validation data coming from the target domain and the mean vector of the semantic class prototype per class. Our representation effectively shows that the method without considering any re-alignment between the visual and the semantic representations suffers from domain shift even over the validation set coming from the source domain. For instance, the semantic class prototype in Figure 4.12a is shifted from their feature distribution. The domain shift problem is even more severe in the target domain (as shown

in Figure 4.12b), where the visual signatures are mapped very close to each other and apart from their mean semantic prototype, thus suffering from inter-class separability. Above all, the projected visual feature representation vectors of the target domain data as shown in Figure 4.12b shrunk towards two semantic prototypes related to classes “sad” and “anger”. Then there is an adverse effect: the semantic vectors which are related to classes “sad” and “anger” are more likely to become hubs, which means they become nearest neighbours of many projected visual feature representation vectors. Therefore, our idea for considering a semantic re-alignment is to reunite samples from two different domains (Figures 4.12a & 4.12b) in an invariant semantic space (by adapting the semantic class prototype to the feature probability distribution) so that we can learn and classify new samples more effectively across domains.

To quantitatively measure the overlap between the source domain visual feature signatures and the target domain visual feature signatures, we follow the same concept of measuring the consistency (eq. (4.20)) but we replace the  $N_{\text{vis}}(i, K)$  with  $N_{\text{vis-source}}(i, K)$  and  $N_{\text{sem}}(i, K)$  with  $N_{\text{vis-target}}(i, K)$ . We found out that the consistency measure is 4.93 out of 7, which is relatively low, leading an inferior classification performance of 51.5% accuracy over the target data. The classification performance over the source data is 89%.

**Ablation study.** We present an ablation study to evaluate the effect of using the proposed architecture under various conditions. To do so, in the following, we change some modules of our original framework DA-FER which is considered as reference point. As presented previously in Figure 4.6, in order to obtain the visual signature representation  $\mathbf{x}_i$  for an image instance  $I_i$ , our model relies on the following consecutive modules:  $I_i \rightarrow \text{CNN} + \text{RL} + \text{SFA-1} + \text{SFA-2} \rightarrow \mathbf{x}_i$ . While for obtaining the semantic class prototype  $\mathbf{y}_k$ , we take an attribute vector  $\mathbf{a}_k$  to compute it using NLP+SFA-3. We present in Table 4.6 various models based on modifying the original framework DA-FER.

For **Model-1**, we replace the SFA-3 module by a simple averaging (eq. (4.21)) across the entire semantic vectors  $\mathbf{y}_j$  to get  $\mathbf{y}_k$  without capturing the contribution of AUs. For **Model-3**, SFA-1 + SFA-2 are replaced by a fully connected (FC) layer that maps the last convolutional layer into a vector  $\mathbf{y}_k \in \mathbb{R}^{300}$ . **Model-1** and **Model-2** still consider the re-alignment step while other variant models **Model-3**, 4 and 5 do not. **Model-3** is the simplest among all the other variants and can be seen as a simple regressor based on a neural network, that is no domain adaptation is taken into account. **Model-4** and **Model-5** are slightly modified versions of **Model-3** where the region layer and soft attention modules are introduced.

$$\mathbf{y}_k = \frac{1}{|Y|} \sum_{\mathbf{y}_j \in Y} \mathbf{y}_j, \quad (4.21)$$

where  $Y \in \mathbb{R}^{24 \times 300}$  is the set of all semantic representations for an attribute vector  $\mathbf{a}_k$ .

By choosing  $K = 7$ , Table 4.6 presents on the first line some preliminary performances that reveal that the semantic embeddings with a persistent visual-semantic neighbourhood structure clearly produce good recognition rates. Then, the ablation studies results are reported on the other lines of Table 4.6.

| Model Variant  |   | Consistency Target | Accuracy Target % | Accuracy Source % |
|----------------|---|--------------------|-------------------|-------------------|
| <b>DA-FER</b>  | $I_i \rightarrow \text{CNN} + \text{RL} + \text{SFA-1} + \text{SFA-2} \rightarrow \mathbf{x}_i$<br>$\mathbf{a}_i \rightarrow \text{g}(\cdot) + \text{SFA-3} \rightarrow \mathbf{y}_k$     | <b>7.0</b>         | 79                | <b>93</b>         |
| <b>Model-1</b> | $I_i \rightarrow \text{CNN} + \text{RL} + \text{SFA-1} + \text{SFA-2} \rightarrow \mathbf{x}_i$<br>$\mathbf{a}_i \rightarrow \text{g}(\cdot) + \text{averaging} \rightarrow \mathbf{y}_k$ | 6.4                | 73                | 85                |
| <b>Model-2</b> | $I_i \rightarrow \text{CNN} + \text{SFA-1} + \text{SFA-2} \rightarrow \mathbf{x}_i$<br>$\mathbf{a}_i \rightarrow \text{g}(\cdot) + \text{averaging} \rightarrow \mathbf{y}_k$             | 5.26               | 62                | 76                |
| <b>Model-3</b> | $I_i \rightarrow \text{CNN} + \text{FC} \rightarrow \mathbf{x}_i$<br>$\mathbf{a}_i \rightarrow \text{g}(\cdot) + \text{averaging} \rightarrow \mathbf{y}_k$                               | 2.19               | 30                | 39                |
| <b>Model-4</b> | $I_i \rightarrow \text{CNN} + \text{RL} + \text{FC} \rightarrow \mathbf{x}_i$<br>$\mathbf{a}_i \rightarrow \text{g}(\cdot) + \text{averaging} \rightarrow \mathbf{y}_k$                   | 2.95               | 35                | 43                |
| <b>Model-5</b> | $I_i \rightarrow \text{CNN} + \text{RL} + \text{FC} \rightarrow \mathbf{x}_i$<br>$\mathbf{a}_i \rightarrow \text{g}(\cdot) + \text{SFA-3} \rightarrow \mathbf{y}_k$                       | 4.3                | 48                | 59                |

Table 4.6: Ablation study: accuracies % on DA target and source data with different models which are variant of the DA-FER approach.

Table 4.6 shows that a simple straightforward model (Model-3) achieves only a 30% accuracy rate over the target domain and 39% accuracy rate over the source domain. This model suffers from the domain-shift problem. Once this model is modified into Model-4, in which RL is integrated at the level of visual representation, a noticeable improvement in both consistency 2.59 and accuracy 35% is achieved. If we add on the top of Model-4 the SFA-3 module which is responsible for capturing the dependencies across AUs, and we keep RL, which leads to Model-5, a better feature and semantic representation is achieved yielding to a 48% accuracy rate and to a slightly better consistency 4.3. However, by integrating the feature and location attention models (SFA-1, SFA-2), which permit for semantic re-alignment as it is the case of Model-1 and Model-2, the performance enhances dramatically leading to a better consistency rate: 6.4 and 5.26 respectively. The main difference between Model-1 and Model-2 concerns the semantic representation, where SFA-3 is kept in Model-1 improving its achievement while for Model-2, SFA-3 is replaced by a simple averaging using eq. (4.20). The above results clearly show a motive for proposing to re-align the neighbourhood structure of the semantic embeddings so that they become consistent with their visual domain counterparts.

Table 4.6 demonstrates the importance of designing a good feature representation for domain adaptation in both embedding spaces (visual and semantic). A good feature representation enables achieving a high classification rate on the source domain while a good semantic aligned representation allows minimizing the overlap between the two domains yielding to better classification performances over the target domain.

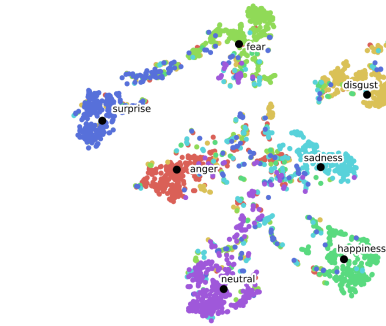
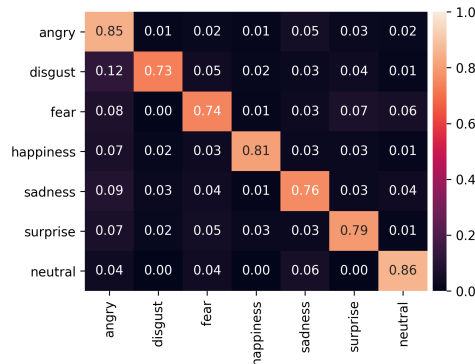
| Method        | Consistency | Accuracy % |
|---------------|-------------|------------|
| <b>DA-FER</b> | <b>7</b>    | <b>79</b>  |
| SimNet        | 5.8         | 65         |
| DA-LS         | 4.9         | 52         |
| VGG-Net       | 3.1         | 34         |

Table 4.7: Benchmarking our proposed DA-FER with the state of the art in DA.

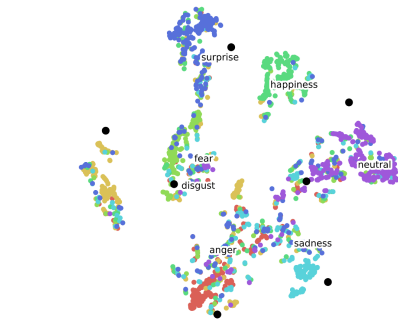
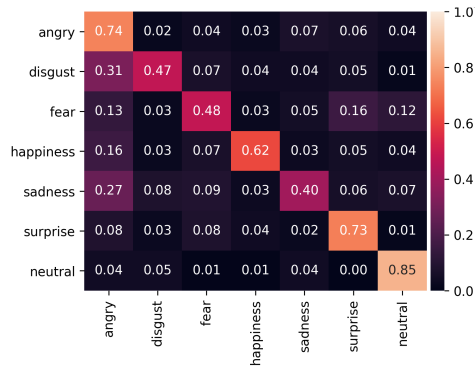
**Comparison with DA state of the art methods.** We follow the same setting and we compare the proposed DA-FER to different domain adaptation algorithms mainly: SimNet, DA-LS and VGG-Net models over the data protocol we devised previously. We compare the classification performances and we investigate the embedding spaces and confusion matrices. Also we report the consistency measures to report the overlap rate.

Figure 4.13 presents the confusion matrices and the corresponding embedding Euclidean space. Figure 4.13(a) shows that our model DA-FER establishes an embedding space in which each semantic class prototype (represented as a black dot) is surrounded by its corresponding target visual signature, yielding to intra-class compactness and resulting in a promising classification performance of 79% using a k-NN with best  $k = 7$  as shown in Table 4.7. The discriminative power of the DA-FER embedding space with both an intra-class compactness and separable inter-class differences, leads to less confusion between classes. For SimNet (Figure 4.13(b)), we observe that the target visual signatures are partially drifted from their semantic class prototype, resulting in less competitive performance with a 65% average accuracy rate as shown in Table 4.7. We can see that SimNet has a good intra-class compactness and separable inter-class differences. But, the main problem is that it still suffers from the domain shift problem. For DA-LS (Figure 4.13(c)), we can observe that the target visual signatures for different classes are partially separable from each other. Their semantic class prototypes are not aligned with their visual signatures and the model only captures the semantic relations among attributes while being blind to any visual feature correlations that might exist. The semantic-visual discrepancy is obvious in this case and leads to an inferior classification performance of 52% as shown in Table 4.7. Finally, with VGG-Net transfer learning, as shown in Figure 4.13(d), the embedding space is suffering from a proper separability between classes resulting in very poor classification performance of 34% accuracy rate while having a higher rate of data overlap.

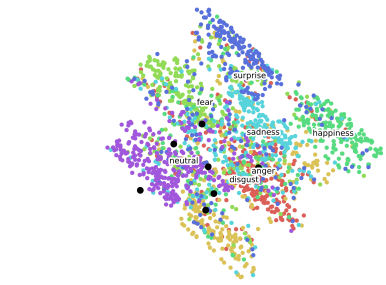
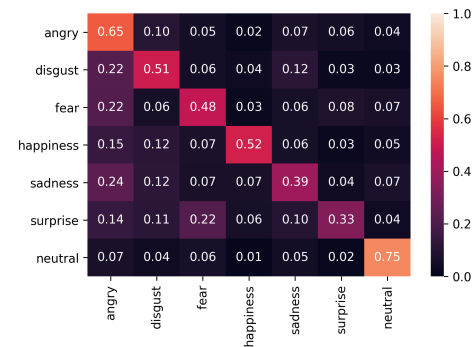
For all models, we plot the confusion matrices (Figure 4.13) in order to describe the adaptation performances over each class. Figure 4.13 indicates that the DA-FER method improves the precision and sensitivity rate while categorizing most of the classes correctly. Some samples coming from the “disgust” class are mapped over the “angry” class and thus resulting an average sensitivity measure around 0.64. The Figure shows that the SimNet method offers a good compromise of sensitivity and precision. The DA-LS method is slightly worse than SimNet. Finally it appears that VGG-Net exhibits low sensitivity and low precision.



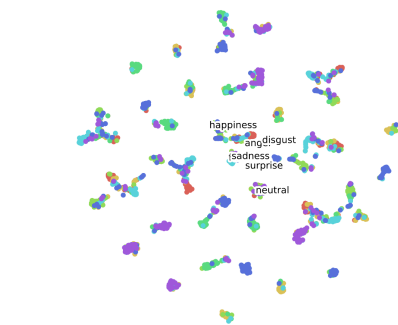
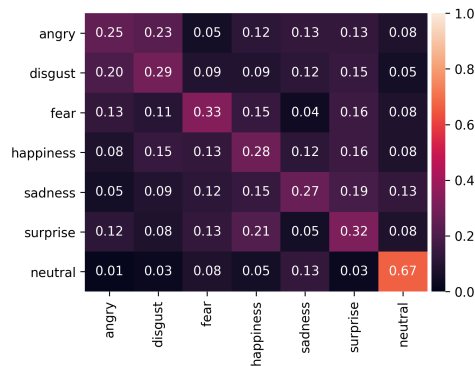
(a) DA-FER



(b) SimNet



(c) DA-LS



(d) VGG-Net

Figure 4.13: t-SNE projection of visual and semantic signatures on the first two dimensions for target domain DISFA dataset and their confusion matrices.



### 4.5.2 ZS-FER Performance Evaluation and Analysis Results

**ZS Data Protocol.** To test our ZS-FER method, we need a large-scale dataset that contains facial images with their corresponding labels. First, we build the source domain with Ekman’s posed and spontaneous basic expressions (angry, disgust, fear, happiness, sadness, surprised and neutral), by merging the following databases: CK+, MMI, MUG and DISFA. The total number of facial images used for training is around 319,291. We build the target domain data using the DynEmo database. This dataset contains around 75,180 facial images corresponding to 358 independent subjects while using Set-1 and Set-2 of this database. We split it into 58 independent subjects for validation with around 12000 facial images and 300 other subjects for the testing phase with around 63,180 facial images. The total number of images per class for each domain is approximately balanced.

**Metric evaluation.** Same as for the Domain Adaptation evaluation setting, we measure the accuracy and the average neighbourhood overlap in order to evaluate the performance. To do so, we use the validation set of the target domain and we define  $N_{vis}(i, K)$  and  $N_{sem}(i, K)$  as the sets that include the K most similar classes to class  $i$  in visual and semantic domains respectively. K in our case is 7 which is the total number of classes in the target domain. Afterwards each class  $i$ , its top-K nearest classes in the visual domain is calculated using eq. (4.19) and put it in  $N_{vis}(i, K)$ . Similarly, we calculate the top-K nearest classes of  $i$  in the semantic domain and put it in  $N_{sem}(i, K)$ . A value closer to K=7 indicates that the embedding is more consistent with the visual domain.

**Ablation study and analysis.** We present an ablation study to evaluate the effect of our re-alignment step and to highlight the effect of other modules on zero shot facial expression recognition. To do so, we compare our approach ZS-FER with the baseline (same architecture as ZS-FER but without considering any alignment) and with other variants: **Model-1**, **Model-2**, **Model-3**, **Model-4** and **Model-5** as shown in Table 4.6 first column and we report the classification accuracy and consistency rates in Table 4.8.

| Method        | Consistency | Accuracy %  |
|---------------|-------------|-------------|
| Baseline      | 3.87        | 39.5        |
| Model-1       | 5.23        | 54.4        |
| Model-2       | 4.84        | 51.2        |
| Model-3       | 1.23        | 19.7        |
| Model-4       | 1.53        | 21.9        |
| Model-5       | 1.92        | 26.3        |
| <b>ZS-FER</b> | <b>6.18</b> | <b>61.2</b> |

Table 4.8: Zero-shot performances over variant models of the ZS-FER approach using ZS data protocol.

| Method            | Consistency | Accuracy %  |
|-------------------|-------------|-------------|
| Zhang et al. 2017 | 4.39        | 43.5        |
| ConSE             | 4.58        | 45.8        |
| DeViSE            | 4.61        | 46.4        |
| MTL               | 4.66        | 47.2        |
| SynC              | 4.68        | 47.7        |
| LatEm             | 4.62        | 46.8        |
| VdSA              | 4.70        | 48.6        |
| VAVE              | 4.78        | 49.4        |
| <b>ZS-FER</b>     | <b>6.18</b> | <b>61.2</b> |

Table 4.9: Zero-shot performances in comparison with the state of the art on ZS data protocol.

**Ablation discussion.** Table 4.8 reveals the importance of optimizing the semantic manifold in the development of new ZS recognition algorithms. For instance when the visual and

semantic spaces are aligned, the overall similarity between the visual features and the class prototypes is maximized which is reflected by observing the consistency measure. Moreover, Table 4.8 notifies the importance of designing a good feature representation for zero shot recognition in both embedding spaces (visual and semantic). For instance, ZS-FER achieves 61.2% classification accuracy rate with 6.18 consistency measure (7 being the maximum) while the baseline achieves only 39.5% classification accuracy rate with 3.87 consistency measure. Thus it is important to infer that some associations between the visual space and the semantic space are crucial to be built which is the key to boost zero shot recognition performance. Eventually, once the re-alignment step is introduced back again to the system, as it is the case of Model-1 and 2, a boost in classification and consistency performances is recorded (54.4% and 51.2% respectively). The simplest model version Model-3 only achieves 19.7% classification rate as the two spaces are inconsistent. From Table 4.8, we deduce that the ZS recognition performances are positively correlated to the changes with the semantic manifold.

**Comparative with ZSL state of the art methods.** To evaluate the proposed ZS-FER model against the state of the art approaches we discussed in Section 4.3.2: ConSE, DeVISE, MTL, SynC, LatEm, VdSA, and VAWE, we use the same ZS data protocol and we report the classification accuracy and consistency rates in Table 4.9.

**Comparative discussion.** We provide a direct comparison between our ZS-FER method and other ZS recognition methods on fine-grained unseen mental state classes. All these methods use deep features to represent images in the visual space and use word embedding to represent the semantic space. It is clear that the ZS recognition performance is better when the semantic structures are enforced to be consistent with their visual domain counterparts as it is the case with ZS-FER. Compared to VAWE and VdSA, as they consider semantic space alignment, they slightly perform better than the other methods. On average, all reported methods have an average accuracy rate around 41.1%, thus with our method, we boost the classification performances over unseen mental state classes by 20.1%. This boost in performances is due to the fact that our method can compensate for the lack of transfer ability of the learned mapping function by refining the manifold structure in the semantic space, especially when two manifolds in visual and semantic are seriously inconsistent. Our algorithm can learn an optimized visual-semantic mapping function and also a new aligned semantic space which is correlated to its visual counterpart.

**Further analysis.** To better understand the performances of the two best zero-shot models (ZS-FER and VAWE), we visualize the confusion matrices per class as shown in Figures 4.14a and 4.14b. Our method ZS-FER is capable at achieving 80% recognition rate over the class *curious* while VAWE achieves only 64%. Similarly, ZS-FER achieves 100% recognition rate over the class *astonished* while VAWE achieves also a high rate with 92%. For the class *irritated*, ZS-FER achieves 68% while VAWE achieves 28%. This class using our method, has been mainly confused with the class *anxious*. While the other method, confuse this class with the classes *anxious* and *ashamed*. ZS-FER method confuses the class *anxious* with the class *irritated*, however VAWE confuses the same class with the class *disappointed*. We can observe in Figure 4.14a, that most mental state classes, mainly *anxious*, *ashamed* and *affected* are confused with the class *irritated*. As a matter of fact, these mental states are

psychologically correlated. Both models appear to have good knowledge transfer ability for the class *astonished*. On contrary, both models appear to have weak knowledge transfer ability for the class *affected*. This might be due to fact that the semantic representation obtained via the class embedding is not well semantically connected with the other available seen classes in the source domain. Visual attributes such as AUs, might provide a better bridge of connectivity. However, it is hard to obtain and annotate, contrary to the basic expressions.

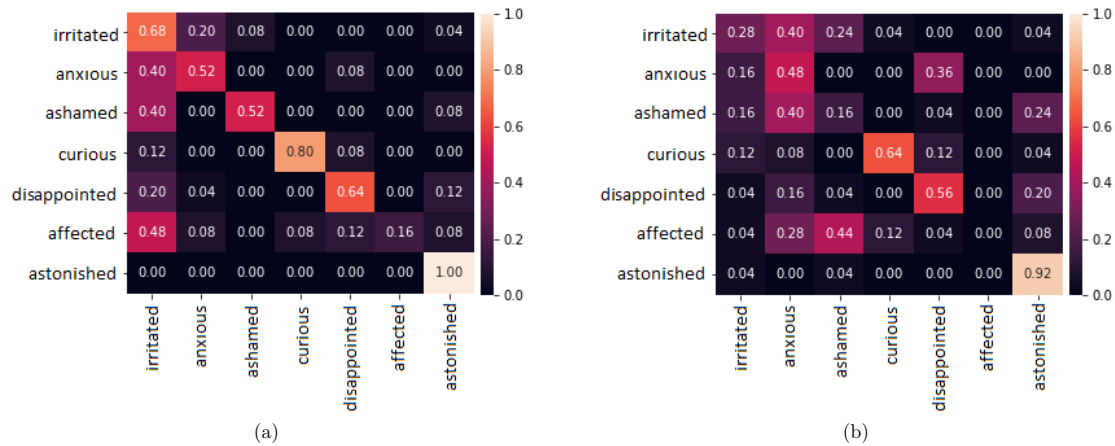


Figure 4.14: (a) Confusion Matrix over the DynEmo database using our model ZS-FER. (b) Confusion Matrix over the DynEmo database using the VAWE model.

## 4.6 Conclusion

In this chapter we proposed a unified framework of domain adaptation and zero shot learning for facial expression recognition. The core concept is an aligned semantic space where we induce structural correspondences among features in domains or tasks by conditioning the semantic space over the visual feature distribution. Our method is highly effective and flexible as it can be integrated with other existing algorithms to enhance the embedding structure while reducing the visual-semantic discrepancy.

The first goal of our proposal was to alleviate the problem of domain shift induced by various intrinsic and extrinsic factors. As a matter of fact, we realized in the study proposed in Chapter 3 the crucial need of considering the domain shift problem, as we discovered that even with a good hierarchical representation, it is still challenging to achieve a good recognition rate over Ekman’s spontaneous facial expressions.

For instance, the 2D-CNN model proposed in Chapter 3 achieved 81% recognition rate over the test set of the DISFA. However, using the proposed model DA-FER, we achieved 79%. It worths to note that the model DA-FER has been never exposed to any spontaneous FE images during training, which is not the case for the 2D-CNN model. DA-FER model has been only trained on Ekman’s posed FE images and the test set used in DA data protocol includes most of the identities of the DISFA database, as we do not need to split any part

of it for training. Despite this fact, to get an insight on the real recognition rates that could be achieved while using the same data protocol in both algorithms, we test our model using the data protocol proposed in Chapter 3 for the DISFA database and we achieved on 83%. Obviously an improvement is recorded.

The method DA-FER can be considered as a weakly supervised method, as it does not need fully annotated data regarding the target domain which has the advantage of reducing the time and complexity of human efforts for annotating large-scale database.

Beyond data adaptation, in this chapter our second goal was to study the ability to transfer knowledge to classes that have never been seen before during the training phase. Our proposal for ZS-FER achieved promising results (61.2% recognition rate) over unseen non-basic spontaneous facial expressions, which is the case of the DynEmo database. Transferring knowledge to unseen classes is still not a mature field and further improvements are needed. One of the advantage we showed of using such a model is their capacity to exploit the knowledge learned from similar or different tasks in a semi-supervised manner. We believe that further improvement could be established by enhancing the bridge used for transferring the knowledge. For instance, instead of using the class names and their semantic encodings, we could use physiological signals or the facial description using action units, only if they are available.



# Micro Facial Expression Analysis

---

Micro-expressions (MiE) are the cause of either conscious suppression or unconscious repression of expressions when a person experiences an emotion but attempts to mask over the facial deformations [Warren et al. 2009, Shen et al. 2012]. Understanding MiEs helps to identify the deception and the true mental condition of a person. Unlike macro-facial expressions (MaE), which typically last for 0.5-4 seconds [Matsumoto and Hwang 2011] and thus can be immediately recognized by humans, MiEs generally remain less than 0.2 seconds, as well they are very subtle [Ekman 2009, Warren et al. 2009] which makes them difficult to spot and recognize them. In order to improve the capacity of people to identify and recognize MiEs, researchers in psychology made improvements to train specialists using the Micro Expression Training Tools [Ekman 2002]. However, even with these training tools, visual reading of MiEs by experts is only around 45% [Endres and Laidlaw 2009, Frank et al. 2009]. Obviously, spotting and recognizing MiEs with a human eye is an extremely difficult task, as there is a need for more descriptive facial feature displacements and motion information. The objective of this chapter is to propose a process for MiEs spotting, that is identifying their temporal and spatial locations in a video sequence while effectively dealing with parasitic movements. The classification of MiEs is out of scope of this work.

## 5.1 Introduction

Automated facial MiE analysis is a relatively new research area [Polikovskiy et al. 2009] when compared with MaE analysis. Although both are related to FE, these two topics should be considered as different research problems. A micro-expression is a rapid and brief facial deformation provoked unintentionally, and it reveals genuine emotions even when people try to mask them [Ekman 1992]. It tends to be more probable in a high-stakes situations in which showing emotions is risky. Compared with a MaE, a MiE has three main characteristics [Ekman 2009, Warren et al. 2009]: (1) micro (brief duration), (2) subtle (low intensity) and (3) local (involves local movements of the facial region).

The phenomenon was first discovered by Haggard and Isaacs 1966, who called them micromomentary facial expressions. Following, Ekman and Friesen 1969, reported finding MiEs while checking a video tape for a psychiatric patient to figure out possible traits of suicide tendency. Admitting the patient seems happy throughout the video, a transient look of grief in the lasting two frames (  $1/12$  s) was found when the tape was inspected in slow motion. This transient of grief was soon confirmed through a confession from the patient in another

counseling session: she lied to conceal her plan to commit suicide. In the following decades, research on MiEs has continued. Ekman 1992, Porter and Ten Brinke 2008, found that facial MiEs are the most important behavioural source for lie indication and can be used for attitude detection as well. Recent studies report many potential applications while using MiE cues, such as affect monitoring [Porter and Ten Brinke 2008], lie detection [Bernstein and Loftus 2009], and clinical diagnosis [Russell et al. 2006].

Researches on MiEs analysis proceed mainly along two dimensions: (1) *MiE Spotting* for automatically localizing the temporal occurrence of MiEs from onset to offset and (2) *MiEs Recognition* for determining the emotional label from well-segmented video containing MiE from onset to offset. MiEs recognition has received more attention since 2011 while assuming that MiEs segments have already been localized [Pfister et al. 2011, Duan et al. 2016, Xu et al. 2017 and Huang and Zhao 2017]. Conversely, few studies since 2014 have been reported regarding the problem of MiEs spotting [Moilanen et al. 2014, Patel et al. 2015, Xia et al. 2016, Li et al. 2017, Tran et al. 2017 and Borza et al. 2017b], though being the primary step for MiEs recognition.

In authentic circumstances, highly trained experts find it difficult to detect a MiE due to its subtle movement nature and characteristic. Endres and Laidlaw 2009 reports only 40% of success, while Frank et al. 2009 reports 50% of success when spotting a MiE. The main goal of this chapter is to propose an automated process for MiEs spotting.

As the duration of an MiE is very short, to capture its speed and subtlety, a high-speed camera is a must during data acquisition. However, the usage of high-speed camera tends to produce parasitic motions and deformations, such as those related to head movements, eye blinks, gaze direction, and mouth opening or closing movements. Those parasitic movements along other facial muscle activations are usually reinforced, in return resulting a confusion with MiEs. As a result, it is essential to eliminate the interferences from unrelated facial MiE information and to emphasize in the meantime on important characteristics of MiEs. On the top of that, to detect a MiE, a method that captures subtle facial motions and subtle local spatial deformations effectively is required.

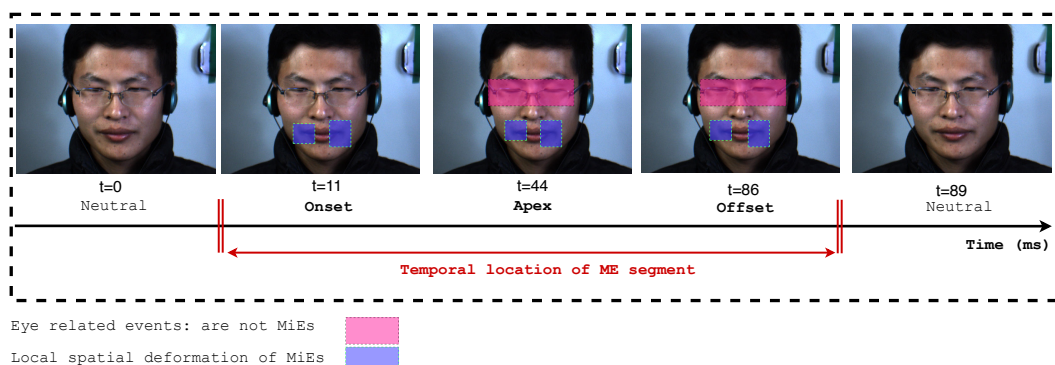


Figure 5.1: Spatio-temporal MiEs detection. A face with a micro-expression that appears around the lip corners associated with a fast eye blinking at the same time.

In this work, we have three main goals summarized in Figure 5.1:

1. To spot MiE segments (onset-offset frames).
2. To pinpoint their subtle local spatial deformations over facial regions.
3. To effectively deal with parasitic movements by distinguishing motions related to MiEs from other facial events.

To achieve our goals, first we need to consider the nature of how MiEs are produced. For instance, MiEs tend to be naturally infrequent since people try not to produce them and very specific conditions are required to evoke MiEs. Therefore a small number of data can be collected about MiEs accompanied with parasitic movements and deformations even though the acquisition process is done in a strictly controlled environment. Having at disposal few data regarding single MiE segments, typically about 2 to 20 MiE frames from onset to offset if recorded with a high-speed camera at 60 fps, it is difficult to utilize supervised learning methods toward automating the process of MiE segment detection.

As a consequence, a weakly supervised method is proposed in which we reformulate the problem of MiEs spotting into a problem of *Anomaly Detection*. All facial motion and deformations but those caused by MiEs are considered as Natural Facial Behaviour (NFB) events. NFB motions and spatial deformations are learned so that we can detect MiEs motions and deformations in the frame as they are different from the one the model has learned.

To this end, by reformulating the problem into anomaly detection, we alleviate the main challenge of dealing with small amount of MiE segments as we deal mainly with NFB events, which are frequent. More importantly, it is more efficient to deal with NFB segments as it is possible to extract for them discriminant spatio-temporal information because their motions and deformations are bigger.

## 5.2 Related Work

Eliciting micro-expression is as difficult as recognizing them because it only appears at certain situations when an individual tries to suppress felt emotions but fails. Therefore, we explain first some of the recent spontaneous MiE databases that are utilized in state of the art methods and which we also use to evaluate our proposal, and we highlight on the way of inducing them. Then, we present the related work regarding micro expression spotting and finally we explore some recent anomaly detection methods based on recent surveys [Chandola et al. 2009, Sodemann et al. 2012 and Kiran et al. 2018].

### 5.2.1 Micro Expressions Databases Elicitation and Training Protocol

Developing a MiE detection system requires databases recorded with high temporal and spatial resolutions and having sufficient training samples. We review here the most recent developed micro-expression databases and also the methods of MiE elicitation.



**Elicitation Method.** According to Ekman and Friesen, when a person tries to conceal his or her feelings, the true emotions would leak quickly and may be manifested as micro-expressions. It was found that watching emotional video episodes while neutralizing faces is an effective method to elicit spontaneous micro-expressions without many irrelevant facial movements [Yan et al. 2013b, Li et al. 2013]. Therefore, to elicit MiEs, studies [Yan et al. 2013a, Yan et al. 2014a and Li et al. 2013] induced participants to experience a high arousal and promoted a motivation to disguise. The participants were asked to watch the video clips in front of a screen and avoid any body movement. Some participants were asked to keep neutralized faces when watching video clips while some other participants only tried to suppress the facial movements when they realized there was a facial expression. This process is followed by micro-expression sample selection and category labeling by an experimenter who kept the emotional clips that only last less than one second or those with onset duration less than 250 milliseconds (ms).

**Micro Expressions Databases.** We summarize in Table 5.1 the basic information for each of the database and present the data protocol we devise. Then, each database is detailed. Figure 5.2 shows frame sequences for a MiE event.

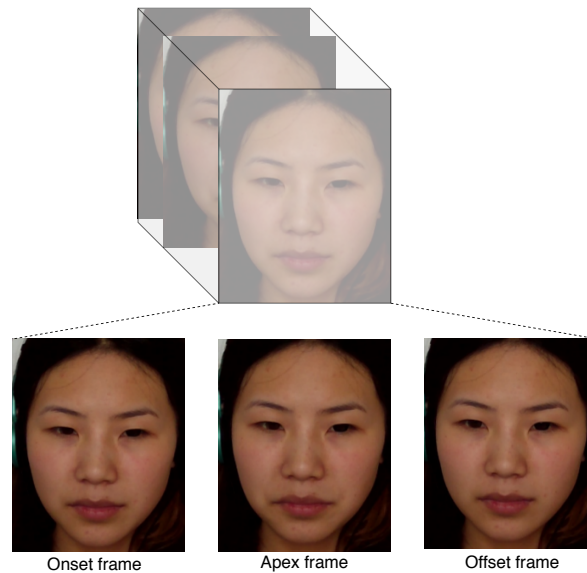


Figure 5.2: An example of frame sequence from CASME-i, including onset frame, apex frame and offset frame. The facial MiE movement associated to this sequence is lip corner depressor.

| Database | Number of subjects | Spatial resolution around the face | Temporal resolution | Number of valid sessions | Validation set                    | Test set                              | Training set                   |
|----------|--------------------|------------------------------------|---------------------|--------------------------|-----------------------------------|---------------------------------------|--------------------------------|
| CASME-i  | 19                 | 200x340 pixels                     | 60 fps              | 172                      | SU:{2,9,13}<br>with 18 sessions   | SU:{6,12,18}<br>with 28 sessions      | SU:{rest}<br>with 126 sessions |
| CASME-ii | 26                 | 200x340 pixels                     | 200 fps             | 255                      | SU:{23,24,26}<br>with 39 sessions | SU:{1,7,21,22,25}<br>with 29 sessions | SU:{rest}<br>with 187 sessions |
| SMIC-HS  | 16                 | 150x160 pixels                     | 100 fps             | 164                      | SU:{1,4,11}<br>with 29 sessions   | SU:{2,8,112}<br>with 30 sessions      | SU:{rest}<br>with 105 sessions |

Table 5.1: A summary of the current spontaneous micro-expression databases with the devised data protocol for model validation.

**The Chinese Academy of Sciences Micro-expression (CASME)-i** database [Yan et al. 2013a] contains 195 spontaneous MiE sequences that correspond to 19 subjects. It is recorded at 60 fps. It is divided into two sets CASME-A (subjects 1 up to 7) and CASME-B (subjects 8 up to 19). CASME-A is recorded under natural light conditions and with a high spatial resolution ( $1280 \times 720$ ) resulting about  $280 \times 340$  pixels on the facial area. While CASME-B is recorded in a room under 2 led lights with a lower resolution camera ( $640 \times 480$ ) resulting about  $150 \times 190$  pixels on facial area. Out of 195 sessions around 172 sessions are valid for spotting MiEs, with the ground truth available for the onset and the offset frames.

**The CASME-ii** database [Yan et al. 2014a] is recorded under a *high temporal resolution* (200 fps ) and a *spatial resolution* of  $280 \times 340$  pixels on facial area. The total number of sessions is 255 that correspond to 26 subjects. Each session is a short video clip up to few seconds.

**The Spontaneous Micro-expression High Speed (SMIC-HS)** database [Li et al. 2013] contains 160 spontaneous MiE sequences that correspond to 16 subjects. Clips from all participants were recorded with a high speed camera at 100 fps with standard spatial resolution  $640 \times 480$ , resulting about  $150 \times 190$  pixels on the facial area.

Among those databases, CASME-i is the most challenging because it is recorded with a camera having temporal resolution 60 fps only, while other recent database have higher resolution. Low temporal resolution makes it challenging to capture motion deformation. As a matter of fact, the CASME-i database has not been reported frequently in state of the art as the CASME-ii and the SMIC-HS databases.

**Training Protocol Setting.** The precision of the proposed detection method is evaluated in a *Subject-Independent* manner.

- CASME-i: for the *validation set*: subjects  $\{2, 9, 13\}$  are considered with 18 sessions while for the *test set*: subjects  $\{6, 12, 18\}$  are considered with 28 sessions. The rest of the subjects are considered as the *training set* with 126 sessions.
- CASME-ii: for the *validation set*: subjects  $\{23, 24, 26\}$  are considered with 39 sessions while for the *test set*: subjects  $\{1, 7, 21, 22, 25\}$  are considered with 29 sessions. The rest of the subjects are used as the *training set* with 187 sessions.
- SMIC-HS: for the *validation set*: subjects  $\{1, 4, 11\}$  are considered with 29 sessions while for the *test set*: subjects  $\{2, 8, 112\}$  are considered with 30 sessions. For the *training set*, 101 sessions are considered.

The choice for the validation and the test sets are based on providing a variety of subject facial deliberate actions such as head movements and eye blinks, wherein the ability of our proposal can be evaluated and test to what extent NFBs could be confused with MiEs.

### 5.2.2 Micro Expression Detection Methods

Recent studies towards enhancing MiEs detection have been reported. Pfister et al. 2011, proposes an innovative framework, where a Temporal Interpolation Model (TIM) alongside Multiple Kernel Learning (MKL) to recognize short expressions is developed. The authors showed that TIM is beneficial while using a standard camera of 25 fps so that it can help matching the detection accuracy as that of 100 fps. To address large variations in the spatial appearances of MiEs, the face geometry is cropped and normalized according to the eye positions from a Haar eye detector and according to the feature points from an AAM. The sixty-eight AAM feature points are transformed using a Local Weighted Mean transformation to model the face and this process is followed by a spatio-temporal feature extraction using a hand-crafted descriptor, mainly the LBP-TOP. Finally, TIM and random forest are utilized to classify MiEs from non-MiEs.

In 2014, a number of algorithms start to appear. A weighted feature extraction scheme has been proposed by Liong et al. 2014 to capture subtle MiEs movements based on Optical Strain. It is defined as the relative amount of deformation of an object. Its ability to capture muscular movements on faces within a time interval makes it suitable for MiEs research contrary to the Optical Flow which is highly sensitive to any change in brightness. The motion information is derived from optical strain magnitudes and is used as a weighting function for the LBP-TOP feature extractor. The last step directly uses the motion information in order to avoid the loss of essential information from the original image intensity values. Then, an SVM is used for classification.

A training free based method for automatically spotting rapid facial movements is proposed by Moilanen et al. 2014. The method relies on analyzing differences in appearance-based features of sequential frames. It aims at finding the temporal locations and to provide the spatial information about facial movements. It is mainly composed of five steps: (1) tracking stable facial points followed by image alignment; (2) dividing frames into blocks; (3) extracting local features using LBP descriptor; (4) calculating the  $\chi^2$  distance within a defined time interval for each block of sequential frames; (5) handling the difference matrix by: (i) obtaining difference values for each frame by averaging the highest block difference values, (ii) contrasting relevant peaks by subtracting the average of the surrounding frames' difference values from each peak, and (iii) using thresholding and peak detection to spot rapid facial movements in the video. The authors showed through their experimental analysis that the proposed method is sensitive to detect other facial events such as eye blinks, global head movements or brightness variations that are not produced by MiEs.

To analyze MiEs, Yan et al. 2014b assesses the spatio-temporal representation. The authors define a Constraint Local Model algorithm to detect faces and track feature points. Based on these points, the ROIs on the face are drawn. Then, the LBP descriptor is used for feature extraction from the defined ROIs and mainly for texture description. Finally, the rate of texture change is obtained by computing the difference between the first frame and the other frames.

The first algorithm that has spotted the onset and the offset frames of MiEs was proposed by Patel et al. 2015. The authors compute the optical flow vector around facial landmarks and integrate them in local spatio-temporal regions. A heuristics to filter non-micro expressions is introduced to find the appropriate onset and offset times. Finally, false detections as head movements, eye blinks and eye gaze changes are reduced by thresholding.

Xia et al. 2016 highlighted the main problems of detecting micro-expressions such as subtle head movements in unconstrained lighting conditions. To face these challenges, a random walk model is introduced to calculate the probability of individual frames being MiEs. Then an Adaboost model is utilized to estimate the initial probability for each frame and the correlation between frames is considered into the random walk model. The AAM and the Procrustes analysis are used to describe the geometric shape of a human face. Finally, the geometric deformation is modeled and used for Ada-boost training.

The detection precision in the latest works since 2017-up-to-date increased. Borza et al. 2017a proposed a framework which can detect the frames in which MiEs occur as well as determine the type of the emerged expressions. Their method uses motion descriptors based on absolute image differences. Ada-boost is used to differentiate MiEs from non-MiEs. The facial ROI is restricted to ten facial regions in which the 68 facial landmarks reside. Li et al. 2017 proposed to spot MiEs using feature difference contrast and peak detection. Their method starts by dividing the facial region into equal size blocks and then tracks each block along the sequence. Spatio-temporal feature extraction using the LBP-TOP descriptor over each block is utilized and then followed by feature difference analysis and thresholding. Borza et al. 2017b captures the movements in facial regions based on an absolute difference technique with a random forest as classifier. Duque et al. 2018 spotted MiEs in a video by analyzing the phase variations between frames obtained from a Riesz Pyramid. This method is capable at differentiating MiEs from eye movements. Lastly, Li et al. 2018 proposed to improve the detection accuracy by recognizing local and temporal patterns of facial movements. The method consists of three parts, a pre-processing step to detect facial landmarks and extract the ROIs, then the extraction of local temporal patterns from a projection in PCA space and eventually the detection of MiEs using an SVM classification.

Our approach shares some similarities with previous works regarding the way to pre-process the facial image. The face is firstly detected, cropped and followed by block division for partitioning the image space. Our proposal does not require any face alignment since our algorithm does not depend on feature difference analysis, and therefore no further tracking algorithm is needed. Regarding spatio-temporal feature extraction, among many local spatio-temporal descriptors, the literature has focused on using LBP-TOP and 3D Histogram of Oriented Gradient (3D-HOG). Here, we decide to take advantage of deep learning to provide a spatio-temporal representation that is robust against background changes, illumination and various environmental changes. Finally, the literature utilizes the extracted features to capture the dissimilarities between MiEs and non-MiEs frames based on either a classification method or a training free based method. Here, we propose a probabilistic framework based on Gaussian Mixture Models (GMM) and adaptive thresholding to identify MiEs from non-MiEs.

### 5.2.3 Anomaly Detection Based-Methods

Anomaly detection is the process of identifying abnormal patterns that correspond to changes in appearance and motion. Typically abnormal events occur rarely and are hardly to annotate or not sufficiently represented. In such a case, class imbalance problems can occur because of the divergence between normal and abnormal sample ratios. Consequently, when only normal behavior samples are easily accessible, it is possible to utilize a semi-supervised method, which only uses normal data to build the model. Existing approaches for semi-supervised methods in the literature can be roughly placed into two categories: (1) Lossless compression/Reconstruction based methods and (2) statistical models.

*Lossless compression/Reconstruction based methods:* the main principle of these methods comes from information theory perspectives [Wang et al. 2012] in measuring the information quantity, and in detecting anomalies according to compression result instead of statistics. These methods assume that anomalies cannot be effectively reconstructed from low-dimensional projections and therefore anomalies will result from higher reconstruction errors. However, due to the complex structures of the normal class (images and videos) and in some domains due to the fine-granularity of spatial and motion information, obtaining an accurate normal class data without containing few abnormal data is not an easy task. The lurk of abnormal data into normal class data might generate indistinguishable reconstruction error and thus limit the performance of such methods. Recent anomaly detection models have utilized deep learning methods using AutoEncoder topology [Marchi et al. 2015a, Yang et al. 2015, Marchi et al. 2015b] where the reconstruction error is used as an activation signal to detect anomalies. The performances of these models are promising but also report significant false positive rates.

*Statistical models:* these models rely on data being generated from a particular distribution. Statistical anomaly detection techniques assume that normal data instances occur in the high probability regions of a stochastic model, while anomalies occur in the low probability regions of the stochastic model. These methods tend to fit a model to the given normal behavior data and then apply a statistical inference test to determine if an unseen instance belongs to this model or not. Instances that have a low probability of being generated from the learned model are declared as anomalies. Gaussian Mixture Models for anomaly detection are frequently used [Basharat et al. 2008, Laxhammar et al. 2009, Nikisins et al. 2018 and Lim et al. 2019] and show good performances. Such techniques assume that the data is generated from a Gaussian distribution. A threshold is applied to the anomaly scores to determine the anomalies. The main limitation of these methods lies down in its difficulty to directly address the curse of dimensionality problem due to multi- or high-dimensional data [Chandola et al. 2009].

To this end, we firstly utilize a reconstruction-based method and propose a recurrent convolutional autoencoder that is capable at preserving essential spatio-temporal information of Natural Facial Behaviour (NFB) samples while generating a low-dimensional representation. Then, a statistical model is designed, by feeding the extracted spatio-temporal features into a Gaussian Mixture Model.

### 5.3 Micro Expression Detection Algorithm

To achieve our objective and to overcome the main challenges enforced by the nature of micro-expressions, we propose an *Anomaly Detection System for Micro-Expression Spotting (ADS-MiE)*. In this thesis, we refer as anomaly the presence of unusual patterns occurring irregularly or being different from other usual normal patterns. Herein, we assign MiEs to the anomaly class which represents abnormal facial behaviours and as the normal class we consider everyday life events referred as NFBs. NFB events consist of fast blinking, eye-gaze changes, facial action unit activations, global head and mouth movements (opening/closing). The anomaly class (MiEs) is often absent during training, poorly sampled or not well defined while the normal class (NFBs) is well characterized and has a lot of normal samples in the training data. In this study, anomalous behaviors are appointed to MiEs due to three main reasons:

1. MiE events occur infrequently in comparison to NFB events.
2. They are not well represented within the image sequences available for modeling.
3. MiE events exhibit significantly distinctive spatio-temporal information with respect to NFB events.

Due to the lack of data of the anomaly class, designing a statistical approach for modeling the normal class (NFBs) distribution and rejecting samples (MiEs) not following this distribution is not straightforward. Therefore, we propose a decoupled process. First, to accurately chart the intrinsic spatio-temporal links of the normal class, a Recurrent Convolutional AutoEncoder (RCAE) is built and learned to entangle the different explanatory factors of spatio-temporal variations in the normal class. Then, the parameters density of the normal class is estimated in the new subspace (latent features from RCAE) using a GMM. Finally, the weighted log-likelihood is computed for ranking the output at each time instance. Typically, a low probability score is expected for the anomaly class as they do not belong to the modeled normal behaviour distribution. Distinguishing MiEs from NFBs based on the obtained likelihood requires thresholding, and to do so we propose an adaptive thresholding technique.

Since subtle expressions occur in highly localized facial regions and across time, we consider a spatio-temporal anomalous sample to be a region, wherein the data values within the spatio-temporal region are different from the ones in its neighborhoods. To provide spatial localization, an image is partitioned into equal regions while preserving their temporal links and a probability distribution that represents each region is estimated. Determining the spatial location has the advantage of masking unrelated facial movements for further analysis and to enhance MiEs classification. Partitioning the image into blocks has the advantage of sampling a large number of individual blocks which provides a greater statistical power while estimating the Probability Density Function (PDF). Moreover processing small spatial regions is computationally less expensive.

Our approach has the benefit of requiring only training events from the normal class and not requiring any data from anomalous events. Our method considers the parasitic motions and deformations as NFB events. As a consequence, those events have to be frequent enough to

be properly modeled but in practice parasitic motions and deformations are not well sampled in the training data. As a result these parasitic data will be badly modelled and probably confused with MiEs. To overcome this challenge, we propose a temporal sampling method to generate several multiscale temporal deformations that induce robustness to the uncertainty of motions of the events, *i.e.* parasitic motions from MiEs motions. Therefore, each block sequence is divided into multiple temporal segments at varying temporal resolutions thus defining instances. A collection of instances is represented as a bag. The whole bags that correspond to NFBs are used to train the RCAE. Then, a probability density function for each bag is separately estimated. At inference time, the full video clips including normal and abnormal patterns are presented. Then for each block at a time instance, the weighted log-likelihood is computed and followed by adaptive thresholding for MiEs spotting.

The proposed ADS-MiE is demonstrated in Figure 5.3 which explains how it works for spotting and localizing MiEs. Let us consider an anomaly video composed of  $k$  frames that include NFBs and MiEs. After pre-processing each frame, a block division (step 1) is applied. The temporal link of each block is kept and a bag of block sequences is established with length  $T_k$ . During the *Learning Stage*, the blocks related to MiE sequences are removed from each bag and only NFB block sequences are considered. Within this stage, a temporal multiscaling (step 2) is applied and its output is what we refer to instances (a sequence of blocks of fixed length  $T_{20}$  with various motion speeds). A collection of instances is used to train an RCAE for dynamic appearance modeling (step 3). The obtained spatio-temporal vectors of each bag are used to train a separate GMM model (step 4) that establishes the regional distribution of each bag. During the *Inference Stage*, the full video clip is considered and no temporal multiscaling is taken into account. However, a sliding window that covers consecutive 20 time steps is used to generate a fixed length sequence so that its spatio-temporal features are encoded and then fed to the modeled regional distribution (step 4) to rank its output (step 5). An adaptive threshold is computed based on the input video. Afterwards deviated patterns (MiEs) are spotted in time and space (step 6) by applying the computed threshold over the weighted log-likelihood scores.

### 5.3.1 The Learning Stage

Two processing steps are carried out, one to partition the face based on a spatial grid (Section 5.3.1.1) and the other to sub-sample at various temporal speeds local NFBs samples via temporal multiscaling (Section 5.3.1.2). Then two learning steps are executed separately: the spatio-temporal feature learning (Section 5.3.1.3) and the PDF estimation (Section 5.3.1.4). The aim is to model the NFBs distribution for each bag and to maximize its likelihood (Section 5.3.2.1). Finally, the output is ranked via adaptive thresholding (Section 5.3.2.2) to identify MiE segments.

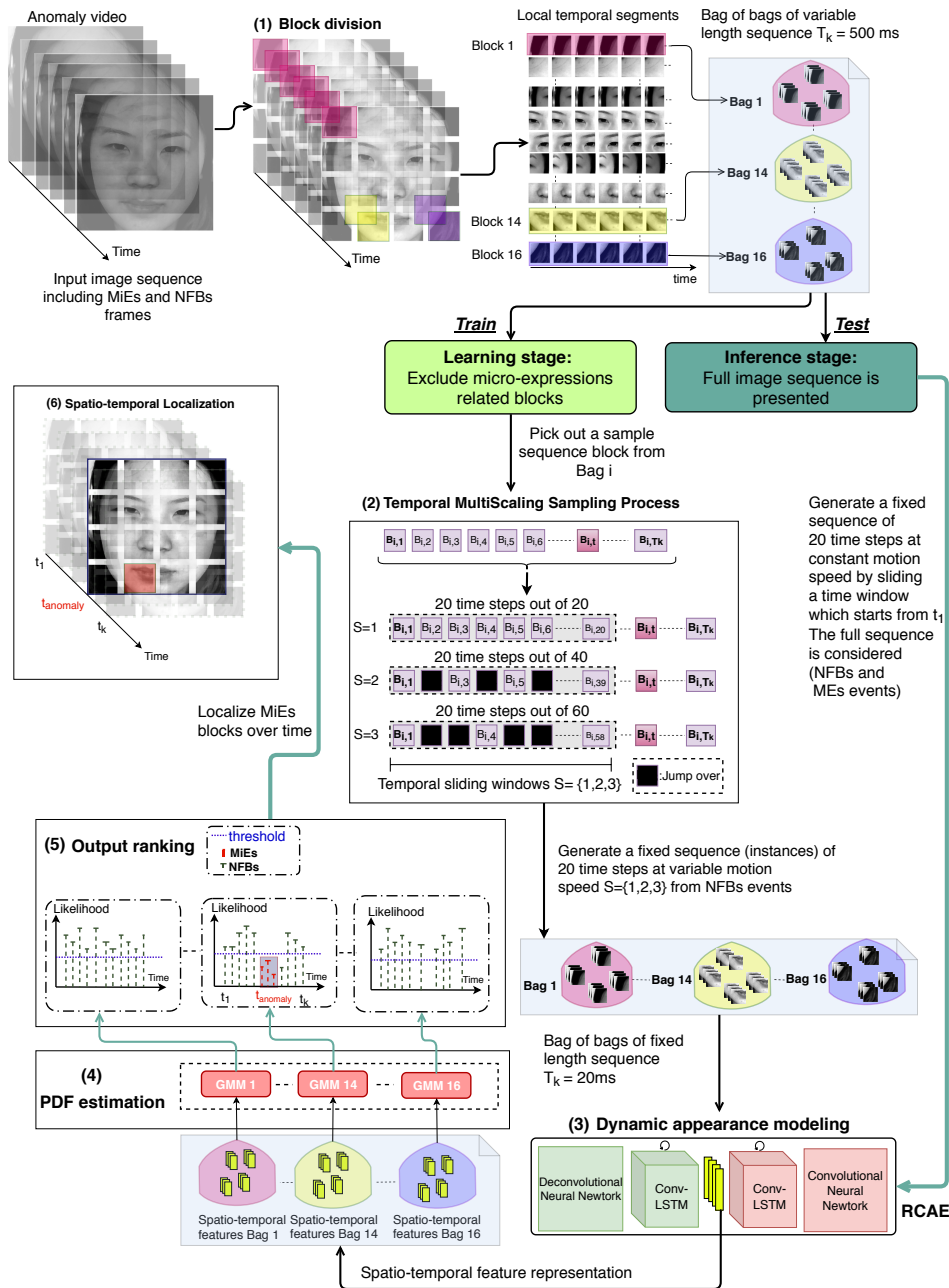


Figure 5.3: General block diagram of the ADS-MiE.

5.3.1.1 Grid Pattern Based Method for Block Division

The intuition behind a grid pattern based method is to divide every frame into blocks. The spatial connection between blocks is not considered but the temporal connection is kept. Then, for each block separately, a model is built to look for abnormal patterns that deviate from normal ones. By evaluating each block alone, the localization is possible. In this work, first the facial region is extracted using the Viola and Jones algorithm, then the face is re-sized to have



a spatial resolution of  $360 \times 360$  pixels. Afterwards, each frame is divided into  $4 \times 4$  blocks. Hence, 16 local regions are obtained where individual blocks have a spatial resolution of  $90 \times 90$  pixels. Blocks are separately tracked through time to keep their temporal information and they are considered as observed data. By that, a large number of local spatial deformations are reachable.

### 5.3.1.2 Temporal Multiscaling for Sub-sampling

Analysing subtle changes in facial behavior is a challenging problem because of its fine-granularity. MiEs in high spatio-temporal resolution video may simultaneously co-occur with some NFBs events specifically eye-related events. Obtaining sufficient information to distinguish NFBs from MiEs is thus critical. Therefore, in order to have a more significant database for NFB training, we propose a temporal multiscaling method that generates different dynamic motion information at various speeds along the temporal axis. It works by skipping some blocks frames at different time scales. Obtaining fine motion information for the normal data class helps to empower the feature extraction process while learning RCAE. Another advantage of our method is that it generates a fixed length sequence from a variable one.

Our proposal for Temporal Multiscaling Sampling (TMS) works by sliding a Temporal Window (TW) that covers 20 time steps out of a variable length video  $T_k$  as shown in Figure 5.3. The main characteristic of the TW is that it can jump between frames according to a parameter  $S = \{1, 2, 3\}$ . With  $S = \{1\}$ , consecutive block frames are considered that are every 20 ms. With  $S = \{2\}$ , 20 ms out of 40 ms are covered with a jump of one between blocks. And with  $S = \{3\}$ , 20 ms out of 60 ms are covered with a jump of two between blocks. TW does not need to start from  $t_1$ . It starts randomly at any time instance such that it satisfies the condition of covering 20 ms (being the minimal duration of a MiE). The optimal length of the TW is a hyper-parameter that is tuned via experimental validation. The output sequence after the TMS process has a shape of  $20 \times 90 \times 90 \times 1$ , 20 being the number of images in a sequence, 90 being the height and width of each block and 1 referring to the gray channel. The minimal number of data samples being generated after the TMS process is  $3 \times 16 \times n$ , where 3 represents the number of sliding windows performed over each sequence, 16 represents the number of blocks generated from each static image and  $n$  is the original number of training sequences. To this end, another advantage of the TMS process is its ability to generate a huge number of samples with various deformations within the image space and along the temporal axis which empowers the learning process of spatio-temporal feature extraction and the statistical modeling.

### 5.3.1.3 Spatio-Temporal Feature Learning

NFBs and MiEs are characterized by some distinctive spatial structures and temporal links between prominent facial regions over time. The temporal evolutions and spatial displacements allow to analyze the current situation relatively to the past, which is critical for describing the entire content of the input sequence. In this study, we decide to leverage a learning method

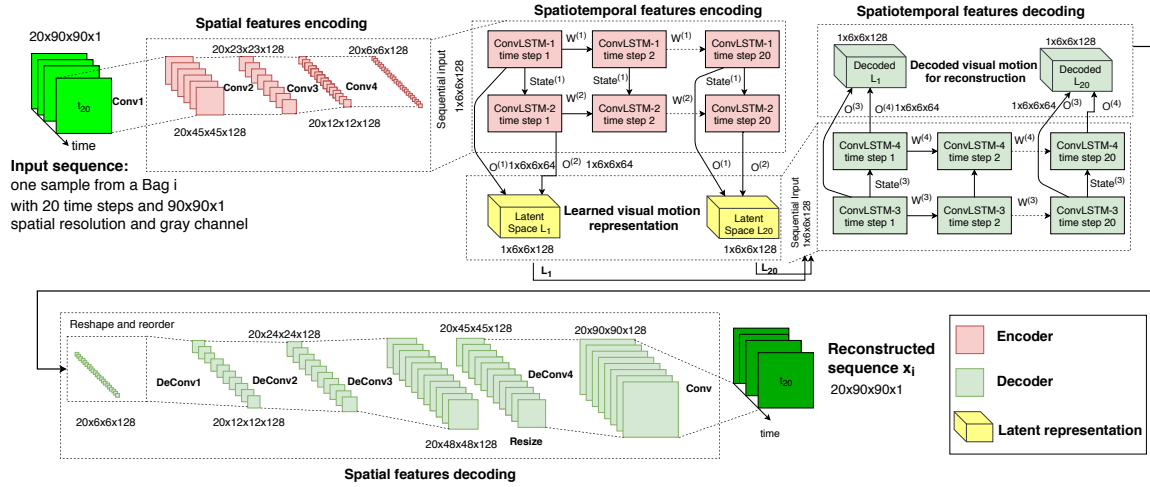


Figure 5.4: Spatio-temporal feature learning for NFBs. Both the Encoder and the Decoder are made up of multilayered CNNs and ConvLSTMs.

to learn visual motion features related to NFBs over short-term and long-term temporal horizons without any annotation constraints. For this task, an RCAE is developed as shown in Figure 5.4, wherein a convolutional network followed by a multilayer Convolutional Long Short-Term Memory (ConvLSTM) cell [Xingjian et al. 2015] is designed. It aims firstly at encoding the input sequence into a fixed length effective representation using a non-linear transformation. Secondly, it aims at extracting the spatial deformations and motion information. And more importantly, at memorizing the past states of spatial block motions by continuous updates of the cell states of the encoder. Afterwards, a decoder that has approximately the mirror architecture than the encoder is designed to map back the extracted spatio-temporal information into its original input space. The mean squared error is used as the objective function.

Given a set of training samples  $X^{train}$  with NFB blocks only, the main goal is to learn a feature representation (referred as *latent representation*  $L$ ) that captures normal behavior spatio-temporal patterns. Let  $x_i \in \mathbb{R}^{20 \times 90 \times 90 \times 1}$  be a sample coming from the facial sub-region block  $i$  of 20 ms. The learned distribution  $\mathcal{D}$  is estimated by building a representation  $f_\theta : X^{train} \rightarrow \mathbb{R}$  that minimizes the mean square error cost function  $C_{\mathcal{D}}(\theta; x_i)$  parameterized by  $\theta$ :

$$\theta^* = \operatorname{argmin}_{\theta} \sum_{x_i \in X^{train}} C_{\mathcal{D}}(\theta; x_i) = \operatorname{argmin}_{\theta} \sum_{x_i \in X^{train}} \|f_\theta(x_i) - x_i\|^2. \quad (5.1)$$

Let assume the input space of  $X$  is  $\mathcal{X}$  and the latent space of  $L$  is  $\mathcal{L}$ . Learning a representation using an *Autoencoder* topology consists of mapping the input space  $\mathcal{X}$  into  $\mathcal{L}$  using the **Encoder** while mapping it back from  $\mathcal{L}$  to  $\mathcal{X}$  using the **Decoder** as shown in Figure 5.4.

**Recurrent Convolutional AutoEncoder.** Figure 5.4 illustrates the architecture for cap-

turing visual motion features. The **Encoder** is designed by stacking four multiple layers of convolution, where each layer is followed by a stride of 2 for down sampling. During convolutional layers, multiple input activations within a filter window are fused to output a single activation. Those CNNs normally extract from each block spatial features related to the spatial patterns. They learn special filters for capturing angles, deformations, edges and other types of appearance features and textures. They encode the primary components of the facial blocks. Some of the feature map activations are represented in Figure 5.5, which shows the presence of specific visual features or patterns that are informative and less redundant compared to the original image patch. For example, the top row of Figure 5.5 represents the response to the eyebrows and closed eye patterns, same for the lips corner and the nose in the last two rows.



Figure 5.5: Filters responses at convolutional layer number 2. The first column on the left represents the original patch.

The feature maps of the final layer Conv-4 are fed to a ConvLSTM module. The ConvLSTM module aims at modeling the spatio-temporal information of short and long motions and at providing a feature vector that encodes both spatial appearance and motion patterns within a sequence. This module is designed by stacking two layers of ConvLSTM, wherein, the cell state of ConvLSTM-1 is fed to the ConvLSTM-2, to preserve the previous spatio-temporal information. Once the encoder reads all the input sequences, it produces the latent representation that encodes the spatio-temporal information of the block sequence. The latent representation  $L$  for each time step, is formed by concatenating the outputs (hidden states) of ConvLSTM-1 and ConvLSTM-2.

A **Decoder** is designed to map back the latent representation  $L$  onto its original space. First it starts by reading  $L$ , where  $L \in \{L_1, \dots, L_{20}\}$ , then it decodes the visual motion vector within  $L$  through ConvLSTM-3 and ConvLSTM-4 and finally it outputs a sequence of vectors. The output is followed by four deconvolutional layers with stride by 2 for up sampling. Then, a resizing layer is designed using the nearest neighbours method to preserve the spatial structure of the original input. During deconvolutional layers, a single input activation with multiple outputs is obtained. Deconvolutional layers decode the full spatial features using the feature maps of the previous layer in order to recover the details of the facial block components and usually they are considered as learnable up-sampling layers. The feature maps of Deconv-4 are fed into a final convolutional layer that maps them into a single input channel with a linear

activation function, in order to obtain the logits or the reconstruction of the image patches. The network parameters are optimized by minimizing the mean square error function between the reconstructed features and the input features of the entire input sequence (eq. (5.1)) using the ADAM optimizer [Kingma and Ba 2014].

Due to the complexity of the deep recurrent autoencoder, training and scalability become an issue, bringing poor generalization. Hinton et al. 2006 proved that a gradient-based optimization starting with a random initialization appears to often get stuck in poor solutions specially for deep architecture. Therefore Hinton et al. 2006 proposed a greedy layer-wise training strategy, bringing better generalization and helping to mitigate the difficult optimization problem of deep networks by better initializing the weights of all layers. The layer-wise training works by training one layer at a time. The subsequent layer is then stacked at the top of the features produced by the previous layer and the whole model is retrained again. In our proposal, we use a layer-wise training strategy by first training the convolutional autoencoder. Then we modify the convolutional autoencoder architecture to include the recurrent layers (ConvLSTM) while loading back the trained weights as a way for re-initializing the network parameters.

The number of filters is set to 128 and the filter size is set to  $3 \times 3$  in all convolutional and deconvolutional layers. The considered activation function is the rectified linear unit function. Each layer is followed by a layer normalization that normalizes the activity of the neurons. For the recurrent part of the encoder and the decoder, 64 filters are considered with size  $3 \times 3$  and the hyperbolic tangent function is used as an activation function. A dropout [Srivastava et al. 2014] with a probability of 65% is applied on the cell states as a regularization technique to reduce overfitting and to enhance generalization.

**Formal Representation.** Let us first simplify the process of building the **convolutional autoencoder** then the process of building the **recurrent convolutional autoencoder**. Given an input sequence (instance from bag  $i$ )  $x_i = (x_1, \dots, x_t, \dots, x_{20})$  such that,  $x_t \in \mathbb{R}^{1 \times 90 \times 90 \times 1}$ . Let  $x_t \equiv h_0$  be a single channel input of spatial size  $90 \times 90$ . Each convolutional layer  $l$  maps the previous input at layer  $l-1$ , into a set of feature map  $h_l$ . The latent map  $h_l$  obtained by the  $k$ th filter of layer  $l$  after the convolutional layer  $l$  is:

$$h_l^k = \text{ReLU}(h_{l-1}^k * W_l^k + b_{l-1}^k), \quad (5.2)$$

where ReLU is the rectified linear unit function, which is mostly used with the convolutional operation (\*).  $W^k$  are the weights of the  $k$ th filter, and  $b^k$  is the bias of the  $k$ -th feature map of the current layer. The latent map  $h_l$  is mapped back into its original space using a deconvolutional operation, resulting  $\hat{x}$ . Equation (5.1) is the objective function that is used to minimize the reconstruction error and to update the parameters  $\theta = \{W, b\}$ .

To extend the convolutional autoencoder into a **recurrent convolutional autoencoder**, the latent space  $h_l$ , which is the latent map obtained at the last convolutional layer  $l$  is fed into a ConvLSTM module.

Let  $h_{l=4}$  represented as  $\mathbf{a}$ , be the feature map obtained by the convolutional layer number 4 from the encoder layer, where  $\mathbf{a} \in \mathbb{R}^{20 \times 6 \times 6 \times 128}$  as shown in Figure 5.4. The recurrent neural network needs first to compute the hidden vector sequence  $h^{rnn} = \{h_1^{rnn}, \dots, h_t^{rnn}, \dots, h_{20}^{rnn}\}$  which is solved through an iterative process:

$$h_t^{rnn} = \tanh(W_{ih}a_t + W_{hh}h_{t-1}^{rnn} + b_h), \quad (5.3)$$

where  $W_{ih}$  and  $W_{hh}$  denote the input-hidden and hidden-hidden weighting matrices while  $b_h$  is the bias vector. The hyperbolic tangent function is denoted as  $\tanh$ . Usually it is preferable over ReLU in recurrent layer since it is bounded and prevents gradient descent vanishing or exploding phenomena, because its second derivative can sustain for a long range before going to zero.

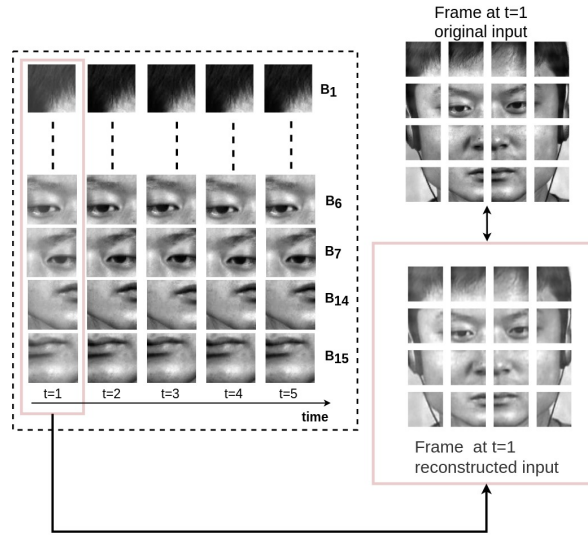


Figure 5.6: The reconstructed blocks over the first 5 time instances.

**Qualitative Analysis.** In order to assist the performance of the recurrent convolutional auto-encoder on its ability for learning good representative features, the reconstructed blocks over time are drawn for unseen test samples from the CASME-ii database. Figure 5.6 demonstrates the effectiveness of the learned model at capturing effective spatio-temporal patterns retaining information about appearance and motion. This information allows a good reconstruction of the input sequence. Our demonstration shows that the learned model is not learning the identity map. Instead, it is capable at extracting motion and visual information present in unseen samples. In addition, it is able to decode back to the original form, which requires decoding the spatio-temporal information stored in the latent representation.

#### 5.3.1.4 Modeling the Normal Facial Behavior Distribution

Having at disposal efficient spatio-temporal features that correspond to the latent space encoded from RCAE for each bag of NFB events, it is possible to model the distribution of these

bags separately to establish bags' PDFs. To model the distribution of each bag, a GMM is utilized. The choice for GMM is due to its ability at learning hidden structures within the latent manifold and for its high detection accuracy.

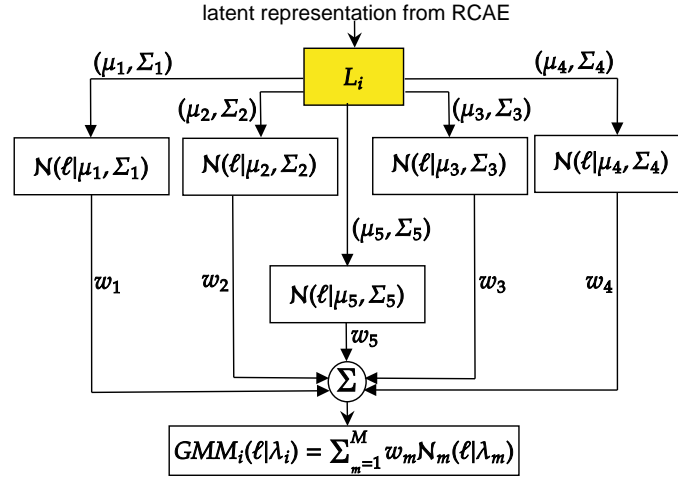


Figure 5.7: Weighted sum of 5 component densities.

**Gaussian Mixture Model.** Let us consider a set of training samples  $X_i^{train}$ , where  $x_i \in \mathbb{R}^{20 \times 90 \times 90 \times 1}$  is the sample coming from the facial sub-region bag  $i$  of 20 time steps. Let us assume  $L_i^{train}$  is the latent space represented by RCAE for a bag  $i$ , whose instances at time  $t$  are denoted by  $\ell_{i,t} \in \mathbb{R}^{1 \times 6 \times 6 \times 128}$ , resulting a feature vector of dimension 4608. To simplify the notation, the dependence on  $i, t$  will be omitted in the rest of the chapter, turning it into  $\ell$ . In order to allow a good spatial localization with a low number of mixture components, the  $GMM_i$  models,  $i \in \{1, \dots, 16\}$ , are learned separately from  $L_i^{train}$ . Otherwise, it is possible to learn a single GMM over the entire blocks with a high number of mixture components. However with a larger number of mixture components, the complexity and the model convergence become an issue.

An overview of the  $GMM_i$  structure is represented in Figure 5.7. The weighted sum of the  $m$  component densities,  $m = \{1, \dots, 5\}$ , is  $GMM_i$ . Each  $GMM_i$  model is parameterized by  $\lambda_i = \{W_i, \mu_i, \Sigma_i\}$  from all component densities, where  $\mu_i = \{\mu_1, \dots, \mu_m\}$  is the mean vector, and  $\Sigma_i = \{\Sigma_1, \dots, \Sigma_m\}$  is the covariance matrix and  $W_i = \{W_1, \dots, W_m\}$  is the weight matrix.

$$GMM_i(\ell|\lambda_i) = \sum_{m=1}^M W_m \mathcal{N}_m(\ell|\lambda_m), \quad (5.4)$$

such that  $\sum_{m=1}^M W_m = 1$ . The Gaussian distribution of one component density  $m$  is given by:

$$\mathcal{N}_m(\ell|\lambda_m) = \frac{1}{(2\pi)^{d/2}} \frac{1}{|\Sigma_m|^{1/2}} \exp\left\{-\frac{1}{2}(\ell - \mu_m)^T \Sigma_m^{-1} (\ell - \mu_m)\right\}. \quad (5.5)$$

One covariance matrix for each Gaussian component is defined and the full rank covariance

matrices for the models are estimated. The main goal is to estimate the parameters  $\lambda_i$  of each GMM $_i$ , which matches the best distribution of training feature vectors  $\ell$ . Given a parameter set initialized by the k-Means algorithm, the Expectation-Maximization algorithm [Dempster et al. 1977] estimates the optimal parameter set  $\lambda_i$  that maximizes the average likelihood over the training set. In order to determine the suitable number of components  $M$ , mixture models with different numbers of components are tested, mainly  $M=\{2, 5, 10\}$  over the validation set.

### 5.3.2 The Inference Stage

At inference stage, as shown in Figure 5.3, the full video clip including NFBs and MiEs frames is presented. The spatio-temporal features for each block are encoded and fed to the learned GMM to rank the output over a grid of spatial blocks. By adaptively thresholding the output, the temporal regions and the spatial blocks related to MiEs are spotted.

#### 5.3.2.1 Output Ranking

NFB observations are statistically modeled by a joint probability. Such a PDF captures the correlations between the features and produces the data likelihood for a particular NFB observation, assuming that it is normal and independent of previous observations (independent and identically distributed). The distribution is estimated using a large dataset that has been generated from the PDF we seek for, for instance for NFB events. Once the PDF for NFBs is modeled, it is possible to compute the Bayesian posterior probability that a facial behavior observation is normal. But, the PDF for abnormal facial events such as MiEs is unavailable, since we exclude them from the entire process. In the presence of the NFB distribution only, the weighted log likelihood (eq. (5.6)) of any observation is an indication of the degree to which the corresponding observation is normal. More precisely, if the weighted log likelihood is below a particular threshold, it is very unlikely that it has been generated from the normal facial behavior PDF and thus it is most probably caused by an abnormal facial behavior event. Therefore, the log likelihood score is a discriminant measure that can be used.

#### 5.3.2.2 Adaptive Thresholding for Decision Making

Decision making is the process of identifying temporal bounds and local spatial locations of MiEs in a facial video clip. The process is based on the weighted log likelihood as a score which is thresholded in an adaptive way. A general representation of the thresholding process is demonstrated in Figure 5.8. Let  $P_{block}(\ell)$  be the weighted log likelihood at time  $t$  of block  $B_i$  as represented in Figure 5.8(a),  $P_{block}(\ell)$  is computed using eq. (5.6).

$$P_{block}(\ell) = \log\left(\sum_{m=1}^M W_m \times \mathcal{N}_m(\ell|\lambda_m)\right). \quad (5.6)$$

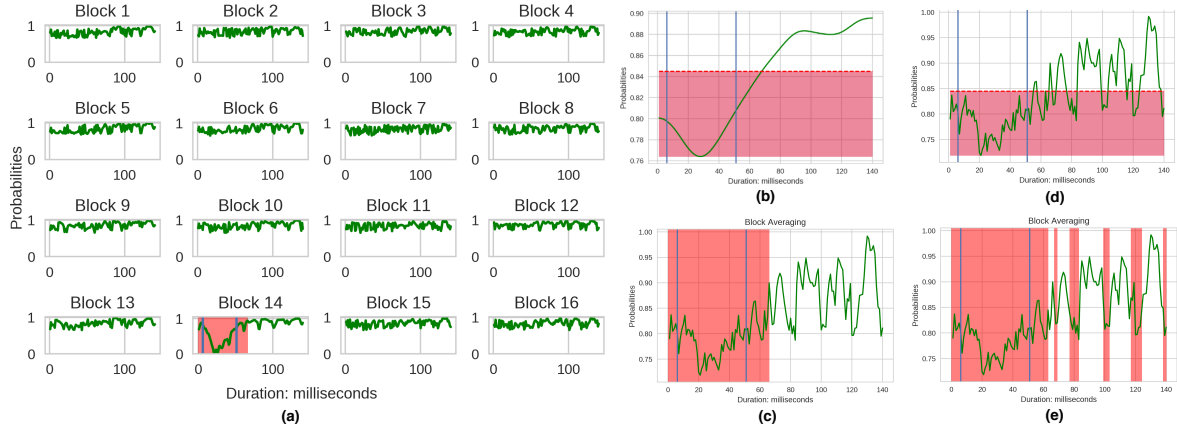


Figure 5.8: Decision making process via adaptive thresholding. The two vertical blue lines are the temporal ground truth while the vertical shaded pink areas are our prediction. The horizontal shaded pink areas correspond to the threshold.

In order to describe the spatio-temporal changes of the video in a compact and consistent representation, we perform a mean pooling across time over  $P_{block}(\ell)$  to obtain a composite curve as shown in Figure 5.8(c) and it is referred to as  $P_{video}$ .  $P_{video}$  is computed using eq. (5.7) and it represents the average weighted log likelihood of the whole video. The intuition behind this pooling step is to help in accentuating the appearance-motion changes by aggregating the local weighted log-likelihoods.

$$P_{video} = \frac{1}{16} \sum_{i=1}^{16} P_{block,i}. \quad (5.7)$$

By applying a 1D-Gaussian kernel over  $P_{video}$  with a standard deviation  $\sigma = 10$ , a smoother curve is obtained as shown in Figure 5.8(b), and we refer to as  $P_{sv}$ . Smoothing is required in order to distinguish relevant peaks from local magnitude variations and noise. For instance, if a threshold  $\mathcal{T}$  is set manually around 0.85 and applied over  $P_{sv}$  (Figure 5.8(b)), only relevant peaks are picked out as shown in Figure 5.8(c), which correspond to the fastest facial movements. In contrast, if the same threshold is applied over an unsmoothed graph that corresponds to  $P_{video}$  as shown in Figure 5.8(d), many false positive peaks are detected as shown in Figure 5.8(e).

For adaptive thresholding, a threshold  $\mathcal{T}$  is defined over the smooth graph  $P_{sv}$  in order to locate the temporal bounds where MiEs occur, as:

$$\mathcal{T} = \max(P_{sv}) - \mu(P_{sv}) + \min(P_{sv}) - 0.5 \times \sigma(P_{sv}). \quad (5.8)$$

where  $\mu$  and  $\sigma$  are the mean and the standard deviation of  $P_{sv}$ . Once  $\mathcal{T}$  is computed, it is applied over  $P_{sv}$  (Figure 5.8(b)), all likelihoods lower than  $\mathcal{T}$  are considered as MiEs frames. By that, the temporal bounds are obtained. To specify the spatial location of MiEs occurrences



within the image,  $\mathcal{T}$  is applied separately over each block  $P_{block}$  (Figure 5.8(a)). As a result, it appears that only  $B_{14}$  corresponding to the left lip corner is the sub-region associated with a fast motion and facial appearance changes.

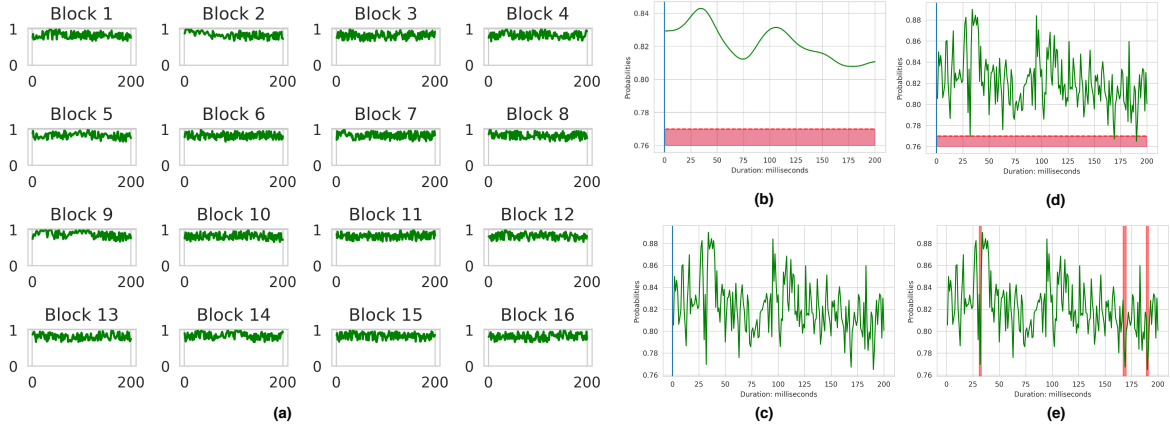


Figure 5.9: Decision making process over a video without any MiE. (a): weighted log-likelihood score over time for each image patch; (c), (d) & (e): Mean pooling over all blocks in (a); (b): smoothed curve with 1-D Gaussian filtering (c); The horizontal shaded areas represents the threshold. The vertical shaded areas represents the prediction of MiE segments.

Let us now consider the case where a video without any MiEs is presented to the ADS-MiE system. We compute the probability score over each block as shown in Figure 5.9(a). Afterwards, we perform mean pooling and we obtain the curve shown in Figure 5.9(c), which is then followed by an 1D-Gaussian smoothing filter and we got the curve shown in Figure 5.9(b). Threshold  $\mathcal{T} = 0.77$  is computed via eq. (5.8) over the smoothed curve. Then,  $\mathcal{T}$  is applied over the smoothed curve and we observe that no temporal or spatial bound is found as  $\mathcal{T}$  is always lower than the minimal probability values, as shown in Figure 5.9(c). However, if  $\mathcal{T}$  is applied directly over the average pooled curve of Figure 5.9(d), three false positive movements are obtained and considered as MiEs as shown in Figure 5.9(e). Herein, we proof the validity of our method to eliminate noisy movements and to differentiate between NFB and MiE events even in the absence of the later.

## 5.4 Experimental Setup & Analysis

To validate and select the best performance of the proposed algorithm among a set of hyper-parameters to control, the following experiments are conducted: regarding the *Feature Learning Stage*, firstly, the effect of  $TW = \{10, 20, 30\}$  ms is evaluated. Secondly, the influence of the temporal multiscaling while modeling NFBs by setting  $S = \{1, 2, 3\}$  and  $S = \{1\}$  in two distinct experiments is tested. Thirdly, in order to validate the robustness of the hierarchical spatio-temporal representation at capturing motions and spatial changes via RCAE, a comparison with other existing spatio-temporal features like LBP-TOP and 3D-HOG is performed. Fourthly, in order to determine the appropriate number of mixture components for the GMM

model, the validation set is used to select the best model by testing  $M = \{2, 5, 10\}$ . Regarding the *Inference Stage*, the effectiveness of the adaptive thresholding technique for decision making is compared with the obtained results when using cross validation for determining the best threshold. To evaluate and compare the performance of the ADS-MiE algorithm with state of the art methods, experiments are conducted on the three benchmarks: CASME-i [Yan et al. 2013a], CASME-ii [Yan et al. 2014a] and SMIC-HS [Li et al. 2013].

### 5.4.1 Evaluation Metrics

The Precision, Recall and Area Under the ROC Curve (AUC) are reported. Moreover, in order to evaluate the temporal boundary precision with respect to the ground truth, the Mean Average Duration (MAD) of MiE segments, which is the minimal average length of MiE segments for a specific database, is estimated and compared with the Mean Average Shift (MAS), which is an indicator of how much the prediction of the temporal location is expanded or collapsed compared to MAD. It is defined as:

$$MAS(q) = \frac{1}{n} \sum_{k=1}^{k=n} |q_p - q_g| \quad (5.9)$$

$$MAS(u) = \frac{1}{n} \sum_{k=1}^{k=n} |u_p - u_g| \quad (5.10)$$

$$MAS = \frac{1}{2} (MAS(q) + MAS(u)), \quad (5.11)$$

where  $q_p$  and  $q_g$  indicate the predicted and the ground truth of the onset frames respectively. Same for the offset frames represented as  $u_p$  and  $u_g$  respectively.

### 5.4.2 Parameters Evaluation and Discussion

For a fair evaluation of the proposed techniques, a control experiment is performed over the CASME-i database. It aims at: figuring out the best value for  $TW = \{10 \text{ ms}, 20 \text{ ms}, 30 \text{ ms}\}$ , evaluating the effect of temporal multiscaling (with and without) and comparing the thresholding techniques: adaptive versus using a fixed threshold obtained by cross validation. CASME-i is considered since it is more challenging than SMIC-HS and CASME-ii due to its low spatial and temporal resolutions. The best values obtained using the control experiment are then also used for SMIC-HS and CASME-ii processing.

Nonetheless, to pick out the best number of mixture components  $M = \{2, 5, 10\}$ , we use the validation set. We find out that when  $M = \{10\}$  and  $M = \{5\}$ , better results are obtained than with  $M = \{2\}$ . Although the absolute best results are obtained with  $M = \{10\}$ , in order to reduce the complexity of the algorithm, we chose  $M = \{5\}$ , since to some degrees it is

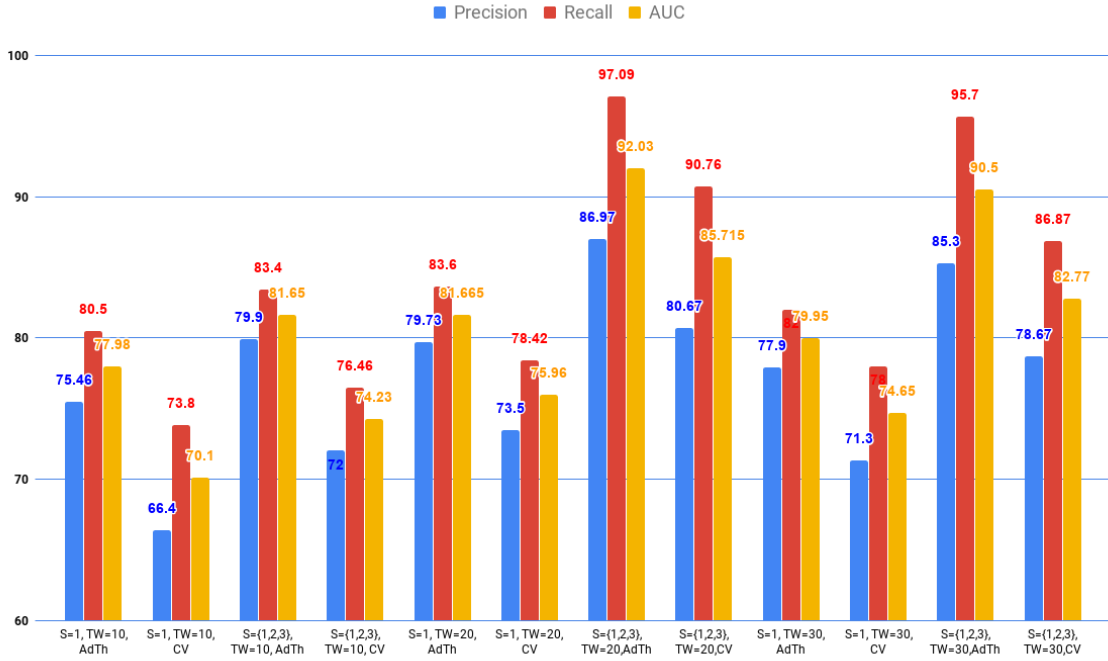


Figure 5.10: Studying the effects of different time windows  $TW = \{10, 20, 30\}$ , temporal Multiscaling  $S = \{1, 2, 3\}$ , and thresholding techniques (Adaptive: *AdTh* or Cross Validation: *CV*) over CASME-i database with  $M = \{5\}$ .

still near the best results. In the following we only report the evaluation of the results with  $M = \{5\}$ . The experimental setup for evaluation is performed as demonstrated in Figure 5.10.

For each temporal window  $TW = \{10, 20, 30\}$  ms, four experimental results are reported as shown in Figure 5.10 where each experiment reports Precision, Recall and AUC values. In the first two experiments with  $S = \{1\}$ , no temporal multiscaling is considered, contrary to the third and the fourth experiments with  $S = \{1, 2, 3\}$ . For each experiment either adaptive threshold (**AdTh**) or fixed threshold obtained via typical cross validation (**CV**) evaluation is done. The best obtained results are those under the following experimental setup:  $TW = \{20\}$ ,  $S = \{1, 2, 3\}$ , **AdTh**, reporting a Precision = 86.97%, a Recall = 97.09% and an AUC value = 92.03%.

For the same parameters setting:  $S = \{1, 2, 3\}$ ,  $TW = \{20\}$ , but with the cross validation method to pick out the best threshold  $\mathcal{T}$ , found at a probability = 0.72, a Precision = 80.67%, a Recall = 90.76% and an AUC value = 85.71% are reported. Obviously, the adaptive thresholding provides better precision and results than with cross validation. Moreover, by comparing the performance with parameters:  $TW = \{20\}$  with adaptive thresholding (**AdTh**) while  $S = \{1\}$ , a Precision = 79.73%, a Recall = 83.6% and an AUC value = 81.66% are reported. As a result, the temporal multiscaling improves the precision rate from 79.73% to 86.97%. This confirms the assumption that the temporal multiscaling enriches the modelling of NFBs and reduces the false positive rate.

### 5.4.3 Detection Results over MiE Databases

The performances of our algorithm using the best parameters setting are reported with time window  $TW=\{20\}$ , temporal multiscaling  $S=\{1, 2, 3\}$ , adaptive threshold and  $M=\{5\}$  over the CASME-i, CASME-ii and SMIC-HS databases. Moreover, we report the mean average shift duration in milliseconds to estimate to what extent the estimation of the onset-offset segment frames expand or collapse from the ground truth. For this purpose, the MAD of MiEs over each database is computed and compared with MAS. The  $\pm$  sign corresponds to the standard deviations of the sample mean distribution.

Table 5.2 shows that spotting MiEs on high speed camera and high spatial resolution video sequences improves the precision rate, as demonstrated when using the CASME-ii database. However, the lower the spatio-temporal resolution, the more challenging MiEs spotting is (as the case of CASME-i database) since the motion and the feature displacements become very similar to those with stationary motions. Developing an accurate MiE detection system requires high quality data. In addition, Table 5.2 shows that the detection algorithm has an acceptable mean average shift duration compared to the mean average duration of the MiEs. This is an important aspect when considering using the detected frames for MiEs recognition in a second step.

The proposed method is then compared with some state of the art methods. The results are presented in Table 5.3. The performances of our algorithm are higher than those obtained in [Li et al. 2017] and [Borza et al. 2017b]. However, compared to [Duque et al. 2018], the same performance is achieved over the CASME-ii database, while having a better score over the SMIC-HS database. It is worthy to note that the state of the art methods focus only on high quality databases (SMIC-HS and CASME-ii) while neglecting CASME-i due to its low temporal resolution.

| Database            | Precision | Recall | AUC    | MAS               | MAD   |
|---------------------|-----------|--------|--------|-------------------|-------|
| CASME-i (60 fps )   | 86.97%    | 97.09% | 92.03% | $5.1 \pm 0.62$ ms | 20 ms |
| SMIC-HS (100 fps )  | 88.87%    | 97.82% | 93.76% | $5.9 \pm 0.82$ ms | 34 ms |
| CASME-ii (200 fps ) | 91.6%     | 98.9%  | 95.17% | $9 \pm 0.95$ ms   | 66 ms |

Table 5.2: Performance evaluation over the three micro facial expressions databases.

| State-of-the-art   | SMIC-HS       | CASME-ii      |
|--------------------|---------------|---------------|
| Li et al. 2013     | 65.55%        | NR            |
| Liong et al. 2014  | 72.87%        | NR            |
| Li et al. 2017     | 83.32%        | 92.98%        |
| Borza et al. 2017b | NR            | 93.4%         |
| Duque et al. 2018  | 89.80%        | 95.13%        |
| <b>ADS-MiE</b>     | <b>93.76%</b> | <b>95.17%</b> |

Table 5.3: Reported AUC values in the state of the art.

#### 5.4.4 Feature Learning Strategies and Evaluation

For evaluating the efficiency of the proposed feature learning method using RCAE, we compare our method with usual handcrafted spatio-temporal features, mainly LBP-TOP and 3D-HOG. The algorithm architecture remains the same, but in the feature learning stage, the spatio-temporal features over each sample block are extracted using either LBP-TOP or 3D-HOG. We only report the results using the best sets of parameters: For LBP-TOP :  $S = \{1, 2, 3\}$ ,  $TW = \{20\}$ , adaptive thresholding and  $M = \{7\}$ , while for 3D-HOG.:  $S = \{1, 2, 3\}$ ,  $TW = \{20\}$ , adaptive thresholding and  $M = \{4\}$ . RCAE has a superior performance over handcrafted features as shown in Table 5.4. Obviously, RCAE is capable at representing and extracting a meaningful spatial and temporal information. Handcrafted descriptors obviously are not able to represent various speeds of motions and spatial displacements.

| Feature extraction | CASME-i       | SMIC-HS       | CASME-ii      |
|--------------------|---------------|---------------|---------------|
| LBP-TOP            | 66.16%        | 68.3%         | 78.2%         |
| 3D-HOG             | 61.43%        | 63.87%        | 71.9%         |
| RCAE               | <b>92.03%</b> | <b>93.76%</b> | <b>95.17%</b> |

Table 5.4: Evaluation using different feature representations. AUC values are reported.

## 5.5 Conclusion

A novel algorithm to spot MiEs spatially and temporally is proposed. The problem of MiE detection is reformulated as an anomaly detection problem. Frequent normal facial behaviors are considered as regularities while infrequent facial behaviors such as micro expressions are considered as irregularities. The main strengths of our algorithm are its simplicity and accuracy in detecting MiEs with subjects having the freedom to perform some deliberate facial actions alongside neutral faces. Moreover, it has reasonable temporal detection deviation and reaches a good detection rate over various spatial and temporal resolution databases. On the contrary, the decoupled learning process between the feature learning and the PDF estimation could be one of the limitations we encounter, which may reduce its efficiency from being end-to-end. Mixture Density Network (MDN) could be a solution, where the RCAE latent space could be fed to MDN to model its distribution alongside the spatio-temporal feature extraction.

# Thesis Summary, Future Work and Publications

---

## 6.1 Summary

Facial expression provides a good protocol to evaluate the emotional state of human beings. This thesis dealt with the problem of macro facial expression recognition and the detection of micro expressions.

Building an automated system for macro facial expression recognition is a very challenging task, especially when dealing with spontaneous expressions. Such expressions include a lot of factors that cause changes in the appearance of face and that affect the process of extracting discriminative features do differentiate different expressions. Such conditions are very hard to control in spontaneous mode and therefore building a reliable system that works in real time is not straightforward. In addition, dealing with spontaneous facial expressions, includes expressiveness variabilities which make the visual and temporal deformations for a certain expression inconsistent among different people.

This thesis dealt with those challenging factors at the level of feature representation and encoding. We started our work by searching for the best spatial feature representation that yields to a better classification rate and then we extended the best representation into spatio-temporal one. We decided to evaluate the power of low level features, mid level features and hierarchical features. To extract low level features, we built an improved version of bag of visual words to improve its performance. We found out that low level features are not able to deal with subtle spatial deformations coming from spontaneous expressions of basic and non basic categories while they are capable at dealing with strong spatial deformations coming from posed basic expressions. Their performances is around 63% over spontaneous databases. To extract mid level features, we extracted sparse codes by learning a discriminative dictionary that sparifies the input facial expression images. Mid level features turned out to be more discriminating than low level features as they provide robustness against some intra and inter class variabilities and deformations to a certain degree. Their sensitivity to these factors depends on the learning process. Their performances is around 73% over spontaneous databases. Finally, we extracted hierarchical features using convolutional neural networks. Those features led to a higher classification rate, around 79%, over spontaneous databases. Apparently, learning complex features helps to tackle the foregoing factors.

Dealing with static images, motion features are neglected. Solving ambiguous facial expressions and extracting very subtle deformations require incorporating the temporal information. Since the hierarchical features showed its efficiency for extracting proper facial expression features, this representation was extended into a temporal one. Herein, we also compare three different levels of spatio-temporal features for comparison reason, because we cannot compare 2D models with 3D ones. We found out that hierarchical spatio-temporal features allowed us to achieve around 69% classification rate as an average over spontaneous emotions of basic and non basic categories when mid-level and low-level spatio-temporal features achieved only 55% and 49% classification rates.

Those results could be furthered improved by training our algorithms over a larger spontaneous database that encompasses most of the extrinsic and intrinsic factors. But, collecting and annotating such a database is very hard and is not practical. In this thesis, we argued that it would be possible to benefit from the available databases for transferring knowledge learned from one domain to another. By considering domain adaptation, results over basic spontaneous database (the DISFA) is improved from 81%, while using a 2D-CNN model trained over the training set coming from the DISFA, to 83%, while training a DA-FER model on posed databases in which the DISFA database have never been seen even in the training set. Obviously, FER models can benefit from both domain adaptation and hierarchical feature encoding to tackle challenging scenarios. Beyond data adaptation, we also studied the ability to transfer knowledge to classes that have never been seen during the training phase. Therefore, we proposed the ZS-FER model and we achieved promising results (61.2% recognition rate) over unseen non-basic spontaneous facial expressions. We believe that further improvements could be established by enhancing the bridge used for transferring the knowledge.

In this dissertation, we also dealt with micro facial expression detection. We proposed an anomaly detection algorithm to spot micro-expression segments. Our method relies on extracting hierarchical spatio-temporal features and on using a Gaussian mixture model for estimating the probability density function. Our results achieved promising performance compare to some state of the art techniques. The main advantage of our method is that it gives the subject the freedom to perform other facial activities rather than just keeping a neutral face. By formulating the problem into anomaly detection, we opened a new view to the problem.

The results and the methods of our studies regarding macro and micro facial expression recognition and detection are an important contribution to the field of affective computing and biometric research. Our findings contribute to the development of a robust facial expression representation at multiple levels and also to incorporate knowledge transfer for domain adaptation and zero shot learning. We have deployed a visualization method to highlight the salient regions of face images. We gave an intuition on tuning a deep neural network. Our work also contributed to the field of computer vision and machine learning, in which our methods can be applied to other domains.

## 6.2 Future Work

Because FER is an important way to infuse emotions into machines, it is advantageous that various studies on its future application are being conducted. Future works may also address the usage of deep learning algorithms for emotion recognition using multi channels. For instance, when analysing facial expressions, we always consider context information: the situation, knowledge about the observed person, speech, voice, hand and body gestures. Likewise, an automatic FER system would need to obtain and combine information from different cues, as it is expected that FER can improve its current recognition rate, including even micro-expressions.

Moreover, many more information could be extracted from a face. For instance, eye movements and head pose are important means to communicate emotional states like boredom or interest and to emphasize mental state expressions. By including these in the analysis, probably a wider range of emotions can be read from the face and distinguished. By comparing human and automatic facial expression recognition, we may be able to advance our understanding of the process and discover new ways of improving automatic facial expression recognition.

We also see a need for research to collect and define large databases of classes that include non-basic affective state which depend on the situation. For example for user interface devices, puzzlement and impatience might be more relevant emotional categories than fear.

Finally, it would be interesting to put more effort on on studying and representing the subtlety of facial expression deformations and to build a unified framework for macro- and micro-facial expressions detection and recognition.



## 6.3 Publications

### 1. Journal Articles

- (a) D. Al Chanti and A. Caplier, (2018). “Deep Learning for Spatio-Temporal Modeling of Dynamic Spontaneous Emotions,” in IEEE Transactions on Affective Computing. DOI: 10.1109/TAFFC.2018.2873600 <https://ieeexplore.ieee.org/document/8481451>

### 2. Journal Papers Under Preparation

- (a) “Domain Adaptation and Zero-Shot Facial Expression Recognition.” To be submitted soon for <https://www.journals.elsevier.com/neural-networks>
- (b) D. Al Chanti and A. Caplier, (2019). “ADS-ME: Anomaly Detection System for Micro-expression Spotting.” Elsevier-Pattern Recognition. To be submitted soon for <https://www.journals.elsevier.com/pattern-recognition>

### 3. International Conference Articles

- (a) D. Al Chanti and A. Caplier, (2017). “Spontaneous Facial Expression Recognition using Sparse Representation.” In Proceedings of the 12th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications - Volume 5: VISAPP, (VISIGRAPP 2017) ISBN 978-989-758-226-4, pages 64-74. <http://www.scitepress.org/DigitalLibrary/Link.aspx?doi=10.5220/0006118000640074>
- (b) D. Al Chanti and A. Caplier, (2018). “Improving Bag-of-Visual-Words Towards Effective Facial Expressive Image Classification.” In Proceedings of the 13th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications - Volume 5: VISAPP, ISBN 978-989-758-290-5, pages 145-152. <http://www.scitepress.org/DigitalLibrary/Link.aspx?doi=10.5220/0006537601450152>

# Brief Review of Table 2.1

---

Summary of the facial representations and feature extractions for FER system proposed in the literature which are presented in Table 2.1, Chapter 2.

## 7.1 Engineered Spatial Appearance Based Features

Buciu and Pitas 2004 uses Eigenfaces, Fisherfaces, and Laplacianfaces on full face images. Gabor filter banks have been successfully used as efficient facial features [Candes and Romberg 2005 and Littlewort et al. 2011] because they are locally concentrated and have been shown to be robust to block occlusions [Donoho 2006]. Dhall et al. 2011 extracts multiples 2D information using Discrete Cosine Transform and Gray-Level Co-Occurrence Matrix for texture analysis. Zhi et al. 2011 proposes Graph-Preserving Sparse Non-negative Matrix Factorization which aims at finding the closest match to a set of base images and at assigning its associated primary emotion. Zafeiriou and Petrou 2009 improves this approach by proposing a new technique called Projected Gradient Kernel Non-negative Matrix Factorization. Geetha and Palaniappan. 2009 describes the appearance of gray scale images by spreading an array of cells across the mouth and extracting the mean intensity from each. Sajjad et al. 2018, propose facial appearance and texture feature-based facial expression recognition framework for sentiment knowledge discovery. Features are extracted from the HOG descriptor with the uniform local ternary pattern descriptor and fused. Those features are extracted from the entire face image rather than from facial components. Munir and Arshid 2018, propose to apply the Fast Fourier Transform and Contrast Limited Adaptive Histogram Equalization method to compensate the poor illumination. Then, for every pixel, a binary pattern code is generated. Iqbal et al. 2018 proposes a Neighborhood-aware Edge Directional Pattern descriptor to extract stable feature descriptions, especially in the presence of weak and distorted edges due to noise.

## 7.2 Engineered Spatial Geometric Based Features

Berretti et al. 2011 and Vretos et al. 2011 generates geometric representations such as depth maps. Kim et al. 2014 uses Active Appearance Model (ASM)-based face normalization and embedded HMM using the two-dimensional discrete cosine transform. Martinez et al. 2013 proposes a method that combines a regression-based approach with a probabilistic graphical

model-based face shape model. Liliana et al. 2018 compute geometric facial components feature directly on pixels basis. Wang et al. 2014 uses the ASM algorithm to align the faces, then extract LBP features. Lu et al. 2017 proposes discrete shearlet transform which is a multiscale geometric analysis method.

### 7.3 Engineered Spatio-Temporal Appearance Based Features

Koelstra et al. 2010 proposes a dynamic texture-based approach using motion history images and nonrigid registration using Free-Form Deformations to derive motion orientation histogram descriptors in both the spatial and temporal domains. Liu and Yin 2015 considers dynamic appearance features by processing sequences with SIFT flow and chunking them into clips. Contiguous clip frames are wrapped and subtracted, spatially dividing the clip with a grid. The resulting cuboids with higher inter-frame variability for either radiance or flow are selected, extracting a Bag of Words histogram from each as signature. Zhao and Pietikainen 2007 uses LBP-TOP for spatio-temporal feature extraction. LBP-TOP is an extension of LBP computed over three orthogonal planes at each bin of a 3D volume formed by stacking the frames. Sun et al. 2014 uses a combination of LBP-TOP and Local Phase Quantization from Three Orthogonal Planes, a descriptor similar to LBP-TOP but more robust to blur. Liu et al. 2014a proposes a combination of low level features using various descriptors extracted from an overlapping grid at each frame then the extracted features are embedded into Riemannian manifolds. Kamarol et al. 2016 propose a spatiotemporal texture map for capturing subtle spatial and temporal variations of facial expressions with a low computational complexity.

### 7.4 Engineered Spatio-Temporal Geometric Based Features

F Dibeqlioglu et al. 2013 proposes to track the facial landmarks, and then to compute the displacement signals of eyebrows, eyelids, cheeks, and lip corners. Afterwards, the mean displacement signal of the lip corners is analyzed and the three main temporal phases of the expressions are estimated. Then, the facial expression dynamics on eyebrows, eyelids, cheeks, and lip corners are extracted from each phase separately. Wu et al. 2013 introduces methods to construct a face shape prior model based on Restricted Boltzmann Machines and Deep Belief Networks. Ghimire and Lee 2013 automatically track landmarks in consecutive video frames, using the displacements based on elastic bunch graph matching displacement estimation. Feature vectors from individual landmarks as well as pairs of landmarks tracking results are extracted and normalized, with respect to the first frame in the sequence. Su et al. 2014 proposes a dynamic facial expression recognition method based on the auto-regressive model to model complicated facial motions. LE 2011 proposes the use of level curve deformations for comparing facial shapes. This method uses pair-wise and segment-wise distances between the level curves comprising the spatiotemporal features for expression recognition from 3D dynamic faces. Sandbach et al. 2011 exploits 3D motion-based features between frames of 3D facial geometry sequences, where an expressive sequence is modeled to contain an onset

followed by an apex and an offset. Feature selection methods are then applied to extract features for each of the onset and offset segments of the expression and finally these features are used to train a HMM to model the full temporal dynamics of the expression. Kacem et al. 2017 proposes a geometric approach for modeling and classifying dynamic facial sequences based on Gramian matrices derived from the facial landmarks. Their representation consists of an affine-invariant shape representation and a spatial covariance of the landmarks.

## 7.5 Learned Spatial Appearance Features

Yang et al. 2018 proposes to recognize facial expressions by extracting information of the expressive components through a de-expression learning procedure based on residue learning. Zhang et al. 2018 propose an end-to-end deep learning model based on a generative adversarial network to learn a generative and discriminative identity representation for face images. Zhao et al. 2016 formalizes a regression problem for frame-level expression estimation and devote an optimization algorithm based on Alternating Direction Method of Multipliers with parameter learning. Kaltwang et al. 2015 proposes to estimate the intensity levels of facial AUs in videos toward interpreting facial expressions. The authors formulate a generative latent tree model and optimize its parameters. Zhao et al. 2015 introduces a joint-patch and multi-label learning to address the problem of facial expression which leverages group sparsity by selecting a sparse subset of facial patches. Liu et al. 2014e presents a Boosted Deep Belief Network framework where a set of features are learned and selected to form a boosted strong classifier in a statistical way. Zeng and Dobaie. 2018 proposes facial expression recognition via learning deep sparse autoencoders for learning discriminative features and training a classifier from the data.

## 7.6 Learned Spatio-Temporal Appearance Features

Gu et al. 2017 proposes a RNN-based method for dynamic facial analysis and compare their method efficiency with conventional Bayesian filters. Liu et al. 2014d proposes spatio-temporal manifold modeling of videos based on a mid-level representation. Wang et al. 2013 model and capture a facial expression as a complex activity using Interval Temporal Bayesian Network. Hasani and Mahoor 2017 proposes a two-part network consisting of a DNN-based architecture followed by a Conditional Random Field module for facial expression recognition in videos. Demirkus et al. 2016 develops a fully automatic hierarchical and probabilistic framework that models the collective set of frame class distributions and feature spatial information over a video sequence. Yau et al. 2015 propose an extreme sparse learning approach to jointly learn a dictionary and a nonlinear classification model. Kuo and Sarkis 2018 proposes a DNN-based model using CNN and RNN for exploiting temporal information.

## 7.7 Copy Rights



### Deep Learning for Spatio-Temporal Modeling of Dynamic Spontaneous Emotions

Author: Dawood Adel AL CHANTI

Publication: Affective Computing, IEEE Transactions on

Publisher: IEEE

Date: Dec 31, 1969

*Copyright © 1969, IEEE*

# Bibliography

- Abadi, Martin et al. (2016). “TensorFlow: A system for large-scale machine learning”. In: *12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16)*, pp. 265–283 (cit. on pp. 83, 87).
- Adolphs, Ralph (2002). “Recognizing emotion from facial expressions: psychological and neurological mechanisms”. In: *Behavioral and cognitive neuroscience reviews* 1.1, pp. 21–62 (cit. on p. 7).
- Aharon, Michal, Michael Elad, and Alfred Bruckstein (2006). “K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation”. In: *IEEE Transactions on signal processing* 54.11, pp. 4311–4322 (cit. on pp. 55, 59).
- Ahonen, Timo, Abdenour Hadid, and Matti Pietikainen (2006). “Face description with local binary patterns: Application to face recognition”. In: *IEEE Transactions on Pattern Analysis & Machine Intelligence* 12, pp. 2037–2041 (cit. on pp. 24, 25).
- Aifanti N., Papachristou C. and A. Delopoulos (2010). “The MUG facial expression database.” In: *In 11th International Workshop on Image Analysis for Multimedia Interactive Services WIAMIS*. IEEE, pp. 1–4 (cit. on pp. 41, 126).
- Ajakan, Hana, Pascal Germain, Hugo Larochelle, François Laviolette, and Mario Marchand (2014). “Domain-adversarial neural networks”. In: *arXiv preprint arXiv:1412.4446* (cit. on p. 107).
- Akata, Zeynep, Florent Perronnin, Zaid Harchaoui, and Cordelia Schmid (2013). “Label-embedding for attribute-based classification”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 819–826 (cit. on p. 104).
- Akata, Zeynep, Scott Reed, Daniel Walter, Honglak Lee, and Bernt Schiele (2015). “Evaluation of output embeddings for fine-grained image classification”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2927–2936 (cit. on pp. 110, 122).
- Aldavert, David, Marçal Rusiñol, Ricardo Toledo, and Josep Lladós (2015). “A study of Bag-of-Visual-Words representations for handwritten keyword spotting”. In: *International Journal on Document Analysis and Recognition (IJ DAR)* 18.3, pp. 223–234 (cit. on p. 45).
- Altıntakan, Umit Lutfu and Adnan Yazici (2015). “Towards effective image classification using class-specific codebooks and distinctive local features”. In: *IEEE Transactions on Multimedia* 17.3, pp. 323–332 (cit. on p. 43).
- Armano, Giuliano, Camelia Chira, and Nima Hatami (2011). “A new gene selection method based on random subspace ensemble for microarray cancer classification”. In: *IAPR International Conference on Pattern Recognition in Bioinformatics*. Springer, pp. 191–201 (cit. on p. 62).
- Arthur, David and Sergei Vassilvitskii (2007). “k-means++: The advantages of careful seeding”. In: *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*. Society for Industrial and Applied Mathematics, pp. 1027–1035 (cit. on pp. 43, 44).

- Assari, Mohammad Amin and Mohammad Rahmati (2011). “Driver drowsiness detection using face expression recognition”. In: *2011 IEEE International Conference on Signal and Image Processing Applications (ICSIPA)*, pp. 337–341 (cit. on p. 3).
- Ba, Lei Jimmy, Ryan Kiros, and Geoffrey E. Hinton (2016). “Layer Normalization”. In: *Neural Information Processing Systems*, p. 12. arXiv: 1607.06450 (cit. on p. 76).
- Baker, Simon and Iain Matthews (2004). “Lucas-kanade 20 years on: A unifying framework”. In: *International journal of computer vision* 3, pp. 221–255 (cit. on p. 21).
- Barros, P. and S. Wermter (2015). “Recognizing complex mental states with deep hierarchical features for Human-Robot Interaction”. In: *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 4065–4070 (cit. on p. 8).
- Basharat, Arslan, Alexei Gritai, and Mubarak Shah (2008). “Learning object motion patterns for anomaly detection and improved object detection”. In: *Proc CVPR IEEE*, pp. 1–8 (cit. on p. 144).
- Bay, Herbert, Andreas Ess, Tinne Tuytelaars, and Luc Van Gool (2008). “Speeded-up robust features”. In: *Computer vision and image understanding* 110.3, pp. 346–359 (cit. on p. 14).
- Bellet, Aurélien, Amaury Habrard, and Marc Sebban (2013). “A survey on metric learning for feature vectors and structured data”. In: *arXiv preprint arXiv:1306.6709* (cit. on p. 107).
- Ben-David, Shai and Reba Schuller (2003). “Exploiting task relatedness for multiple task learning”. In: *Learning Theory and Kernel Machines*. Springer, pp. 567–580 (cit. on p. 102).
- Bengio, Yoshua, Patrice Simard, and Paolo Frasconi (1994). “Learning long-term dependencies with gradient descent is difficult”. In: *IEEE transactions on neural networks* 5.2, pp. 157–166 (cit. on p. 78).
- Bengio, Yoshua et al. (2009). “Learning deep architectures for AI”. In: *Foundations and trends in Machine Learning* 2.1, pp. 1–127 (cit. on p. 28).
- Bengio, Yoshua, Aaron Courville, and Pascal Vincent (2013). “Representation learning: A review and new perspectives”. In: *IEEE transactions on pattern analysis and machine intelligence* 35.8, pp. 1798–1828 (cit. on p. 82).
- Bernstein, Daniel M and Elizabeth F Loftus (2009). “How to tell if a particular memory is true or false”. In: *Perspect Psychol Sci* 4.4, pp. 370–374 (cit. on p. 138).
- Berretti, Stefano, Boulbaba Ben Amor, Mohamed Daoudi, and Alberto Del Bimbo (2011). “3D facial expression recognition using SIFT descriptors of automatically detected keypoints”. In: *The Visual Computer* 27.11, p. 1021 (cit. on pp. 23, 165).
- Borza, Diana, Radu Danescu, Razvan Itu, and Adrian Darabant (2017a). “High-speed video system for micro-expression detection and recognition”. In: *Sensors* 17.12, p. 2913 (cit. on p. 143).
- Borza, Diana, Razvan Itu, and Radu Danescu (2017b). “Real-time micro-expression detection from high speed cameras”. In: *Int C Intell Comp Co*, pp. 357–361 (cit. on pp. 138, 143, 159).
- Boulogne, Guillaume-Benjamin Duchenne de (1862). *Mécanisme de la physionomie humaine: Texte*. Renouard (cit. on p. 2).
- Boureau, Y-Lan, Jean Ponce, and Yann LeCun (2010). “A theoretical analysis of feature pooling in visual recognition”. In: *Proceedings of the 27th international conference on machine learning (ICML-10)*, pp. 111–118 (cit. on p. 32).

- Boureau, Y-Lan, Nicolas Le Roux, Francis Bach, Jean Ponce, and Yann LeCun (2011). “Ask the locals: multi-way local pooling for image recognition”. In: (cit. on p. 32).
- Bradley, David M and J Andrew Bagnell (2008). “Differential sparse coding”. In: (cit. on p. 54).
- Bruce, Vicki (1993). “What the human face tells the human mind: Some challenges for the robot-human interface”. In: *Advanced Robotics* 8.4, pp. 341–355 (cit. on p. 1).
- Bryt, Ori and Michael Elad (2008). “Compression of facial images using the K-SVD algorithm”. In: *Journal of Visual Communication and Image Representation* 19.4, pp. 270–282 (cit. on p. 59).
- Buciu, Ioan and Ioannis Pitas (2004). “Application of non-negative and local non negative matrix factorization to facial expression recognition”. In: *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on*. Vol. 1. IEEE, pp. 288–291 (cit. on p. 165).
- Byrnes, Alyssa and Cynthia Sturton (2018). “On Using Drivers’ Eyes to Predict Accident-Causing Drowsiness Levels”. In: *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, pp. 2092–2097 (cit. on p. 8).
- Calder, Andrew J, A Mike Burton, Paul Miller, Andrew W Young, and Shigeru Akamatsu (2001). “A principal component analysis of facial expressions”. In: *Vision research* 41.9, pp. 1179–1208 (cit. on p. 33).
- Candes, Emmanuel and Justin Romberg (2005). “11-magic: Recovery of sparse signals via convex programming”. In: 4, p. 46 (cit. on p. 165).
- Candes, Emmanuel J and Michael B Wakin (2008). “An introduction to compressive sampling [a sensing/sampling paradigm that goes against the common knowledge in data acquisition]”. In: *IEEE signal processing magazine* 25.2, pp. 21–30 (cit. on p. 27).
- Chandola, Varun, Arindam Banerjee, and Vipin Kumar (2009). “Anomaly detection: A survey”. In: *ACM computing surveys (CSUR)* 41.3, p. 15 (cit. on pp. 139, 144).
- Changpinyo, Soravit, Wei-Lun Chao, Boqing Gong, and Fei Sha (2016). “Synthesized classifiers for zero-shot learning”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5327–5336 (cit. on pp. 113, 114).
- Chanti, D. A. Al and A. Caplier (2018). “Deep Learning for Spatio-Temporal Modeling of Dynamic Spontaneous Emotions”. In: *IEEE Transactions on Affective Computing*, pp. 1–1 (cit. on p. 101).
- Chao, C., H. K. Lin, J. Lin, and Y. Tseng (2012). “An Affective Learning Interface with an Interactive Animated Agent”. In: *2012 IEEE Fourth International Conference On Digital Game And Intelligent Toy Enhanced Learning*, pp. 221–225 (cit. on p. 3).
- Chatfield, Ken, Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman (2014). “Return of the Devil in the Details: Delving Deep into Convolutional Nets”. In: *Proceedings of the British Machine Vision Conference*. BMVA Press (cit. on p. 126).
- Chen, M., L. Zhang, and J. P. Allebach (2015). “Learning deep features for image emotion classification”. In: *2015 IEEE International Conference on Image Processing (ICIP)*, pp. 4491–4495 (cit. on p. 101).
- Cheng, Hong, Zicheng Liu, Lu Yang, and Xuewen Chen (2013). “Sparse representation and learning in visual recognition: Theory and applications”. In: *Signal Processing* 93.6, pp. 1408–1425 (cit. on p. 59).



- Cho, Kyunghyun, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio (2014). “Learning phrase representations using RNN encoder-decoder for statistical machine translation”. In: *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (cit. on p. 110).
- Cimpoi, Mircea, Subhansu Maji, and Andrea Vedaldi (2015). “Deep filter banks for texture recognition and segmentation”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3828–3836 (cit. on pp. 31, 96).
- Cohn, Jeffrey F. (2007). “Foundations of Human Computing: Facial Expression and Emotion”. In: *Artificial Intelligence for Human Computing*. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 1–16 (cit. on p. 4).
- Cootes, T. F., G. J. Edwards, and C. J. Taylor (2001). “Active appearance models”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 23.6, pp. 681–685 (cit. on p. 21).
- Corneanu, Ciprian Adrian, Marc Oliu Simón, Jeffrey F Cohn, and Sergio Escalera Guerrero (2016). “Survey on rgb, 3d, thermal, and multimodal approaches for facial expression recognition: History, trends, and affect-related applications”. In: *IEEE transactions on pattern analysis and machine intelligence* 38.8, pp. 1548–1568 (cit. on p. 19).
- Cotter, Shane F (2010). “Sparse representation for accurate classification of corrupted and occluded facial expressions”. In: *2010 IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, pp. 838–841 (cit. on p. 28).
- Cowie, Roddy, Ellen Douglas-Cowie, Nicolas Tsapatsoulis, George Votsis, Stefanos Kollias, Winfried Fellenz, and John G Taylor (2001). “Emotion recognition in human-computer interaction”. In: *IEEE Signal processing magazine* 18.1, pp. 32–80 (cit. on p. 11).
- Craig, Kenneth D, Susan A Hyde, and Christopher J Patrick (1991). “Genuine, suppressed and faked facial behavior during exacerbation of chronic low back pain”. In: *Pain* 46.2, pp. 161–171 (cit. on p. 9).
- Crammer, Koby, Michael Kearns, and Jennifer Wortman (2006). “Learning from data of variable quality”. In: *Advances in Neural Information Processing Systems*, pp. 219–226 (cit. on p. 102).
- Dahmane, Mohamed and Jean Meunier (2011). “Continuous emotion recognition using Gabor energy filters”. In: *international conference on Affective computing and intelligent interaction*. Springer, pp. 351–358 (cit. on p. 26).
- Dalal, Navneet and Bill Triggs (2005). “Histograms of oriented gradients for human detection”. In: (cit. on pp. 14, 24, 25).
- Darwin, Charles and Phillip Prodger (1872). *The expression of the emotions in man and animals*. Oxford University Press, USA (cit. on p. 2).
- Dasgupta, Sanjoy and Anupam Gupta (1999). “An elementary proof of the Johnson-Lindenstrauss lemma”. In: *International Computer Science Institute, Technical Report*, pp. 99–006 (cit. on p. 63).
- Davis, Jason V, Brian Kulis, Prateek Jain, Suvrit Sra, and Inderjit S Dhillon (2007). “Information-theoretic metric learning”. In: *Proceedings of the 24th international conference on Machine learning*. ACM, pp. 209–216 (cit. on p. 108).
- Demirkus, M., D. Precup, J. J. Clark, and T. Arbel (2016). “Hierarchical Spatio-Temporal Probabilistic Graphical Model with Multiple Feature Fusion for Binary Facial Attribute

- Classification in Real-World Face Videos”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 38.6, pp. 1185–1203 (cit. on pp. 12, 23, 167).
- Dempster, Arthur P, Nan M Laird, and Donald B Rubin (1977). “Maximum likelihood from incomplete data via the EM algorithm”. In: *Journal of the royal statistical society. Series B (methodological)*, pp. 1–38 (cit. on p. 154).
- Devries, T., K. Biswaranjan, and G. W. Taylor (2014). “Multi-task Learning of Facial Landmarks and Expression”. In: *2014 Canadian Conference on Computer and Robot Vision*, pp. 98–103 (cit. on p. 33).
- Dhall, Abhinav, Akshay Asthana, Roland Goecke, and Tom Gedeon (2011). “Emotion recognition using PHOG and LPQ features”. In: *Face and Gesture 2011*. IEEE, pp. 878–883 (cit. on pp. 11, 23, 165).
- Dibeklioglu, Hamdi, Albert Ali Salah, and Theo Gevers (2013). “Like Father, Like Son: Facial Expression Dynamics for Kinship Verification”. In: *The IEEE International Conference on Computer Vision (ICCV)* (cit. on pp. 23, 30, 166).
- Ding, Hui, Shaohua Kevin Zhou, and Rama Chellappa (2017). “Facenet2expnet: Regularizing a deep face recognition net for expression recognition”. In: *2017 12th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2017)*. IEEE, pp. 118–126 (cit. on p. 110).
- Donoho, David L (2006). “Compressed sensing”. In: *Information Theory, IEEE Transactions on* 52.4, pp. 1289–1306 (cit. on p. 165).
- Donoho, David L and Michael Elad (2003). “Optimally sparse representation in general (nonorthogonal) dictionaries via  $l_1$  minimization”. In: *Proceedings of the National Academy of Sciences* 100.5, pp. 2197–2202 (cit. on p. 58).
- Du, Shichuan, Yong Tao, and Aleix M Martinez (2014a). “Compound facial expressions of emotion”. In: *Proceedings of the National Academy of Sciences* 111.15, E1454–E1462 (cit. on pp. 5, 8, 122).
- Du, Shichuan, Yong Tao, and Aleix M. Martinez (2014b). “Compound facial expressions of emotion”. In: *Proceedings of the National Academy of Sciences* 111.15, E1454–E1462 (cit. on p. 8).
- Duan, Xiaodong, Qiguo Dai, Xinhan Wang, Yuangang Wang, and Zhichao Hua (2016). “Recognizing spontaneous micro-expression from eye region”. In: *Neurocomputing* 217, pp. 27–36 (cit. on p. 138).
- Duque, Carlos, Olivier Alata, Rémi Emonet, Anne-Claire Legrand, and Hubert Konik (2018). “Micro-Expression Spotting using the Riesz Pyramid”. In: *WACV 2018* (cit. on pp. 143, 159).
- Ekman, P (2002). *MicroExpression Training Tool (METT)*. University of California, San Francisco (cit. on p. 137).
- Ekman, P and W Friesen (1969). “Nonverbal leakage and clues to deception in Psychiatry”. In: (cit. on pp. 137, 140).
- Ekman, Paul (1992). “Facial expressions of emotion: an old controversy and new findings”. In: *Philosophical Transactions of the Royal Society of London. Series B: Biological Sciences* 335.1273, pp. 63–69 (cit. on pp. 137, 138).
- (2009). “Lie catching and microexpressions”. In: *The philosophy of deception*, pp. 118–133 (cit. on pp. 10, 137).

- Ekman, Paul and Daniel Cordaro (2011). “What is meant by calling emotions basic”. In: *Emotion review* 3.4, pp. 364–370 (cit. on p. 7).
- Ekman, Paul and Wallace V Friesen (1978). *Manual for the facial action coding system*. Consulting Psychologists Press (cit. on pp. 2, 4).
- Ekman, Paul and Erika L Rosenberg (1997). *What the face reveals: Basic and applied studies of spontaneous expression using the Facial Action Coding System (FACS)*. Oxford University Press, USA (cit. on pp. 5, 6, 40).
- Ekman, Rosenberg (1997). *What the face reveals: Basic and applied studies of spontaneous expression using the Facial Action Coding System (FACS)*. Oxford University Press, USA (cit. on pp. 9, 10).
- Elad, Michael (2010). *Sparse and redundant representations: from theory to applications in signal and image processing*. Springer Science & Business Media (cit. on p. 58).
- Elad, Michael and Michal Aharon (2006). “Image denoising via sparse and redundant representations over learned dictionaries”. In: *Image Processing, IEEE Transactions on* 15.12, pp. 3736–3745 (cit. on p. 54).
- Eleftheriadis, S., O. Rudovic, and M. Pantic (2015). “Discriminative Shared Gaussian Processes for Multiview and View-Invariant Facial Expression Recognition”. In: *IEEE Transactions on Image Processing* 24.1, pp. 189–204 (cit. on p. 101).
- Elhoseiny Mohamed, Babak Saleh and Ahmed Elgammal. (2013). “Write a classifier: Zero-shot learning using purely textual descriptions.” In: *In Proceedings of the IEEE International Conference on Computer Vision*, pp. 2584–2591 (cit. on p. 122).
- Endres, Jennifer and Anita Laidlaw (2009). “Micro-expression recognition training in medical students: a pilot study”. In: *BMC Med Educ* 9.1, p. 47 (cit. on pp. 137, 138).
- Essa and Pentland (1994). “A vision system for observing and extracting facial action parameters”. In: *1994 Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 76–83 (cit. on p. 2).
- Farhadi, Ali, Ian Endres, Derek Hoiem, and David Forsyth (2009). “Describing objects by their attributes”. In: *2009 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, pp. 1778–1785 (cit. on pp. 104, 110).
- Fasel, Beat and Juergen Luetttin (2003). “Automatic facial expression analysis: a survey”. In: *Pattern recognition* 36.1, pp. 259–275 (cit. on p. 4).
- Frank, MG, Malgorzata Herbasz, Kang Sinuk, A Keller, and Courtney Nolan (2009). “I see how you feel: Training laypeople and professionals to recognize fleeting emotions”. In: *The Annual Meeting of the International Communication Association. Sheraton New York, New York City* (cit. on pp. 137, 138).
- Frome, Andrea, Greg S Corrado, Jon Shlens, Samy Bengio, Jeff Dean, Tomas Mikolov, et al. (2013). “Devise: A deep visual-semantic embedding model”. In: *Advances in neural information processing systems*, pp. 2121–2129 (cit. on pp. 113, 114).
- Furuya, Takahiko and Ryutarou Ohbuchi (2009). “Dense sampling and fast encoding for 3D model retrieval using bag-of-visual features”. In: *Proceedings of the ACM international conference on image and video retrieval*. ACM, p. 26 (cit. on p. 44).
- Gal, Yarín and Zoubin Ghahramani (2016). “Dropout as a bayesian approximation: Representing model uncertainty in deep learning”. In: *international conference on machine learning*, pp. 1050–1059 (cit. on p. 86).

- Geetha A., Vennila Ramalingam S. Palanivel and B. Palaniappan. (2009). “Facial expression recognition—A real time approach”. In: *Expert Systems with Applications* 36.1, pp. 303–308 (cit. on p. 165).
- Gers, Felix A, Nicol N Schraudolph, and Jürgen Schmidhuber (2002). “Learning precise timing with LSTM recurrent networks”. In: *Journal of machine learning research* 3.Aug, pp. 115–143 (cit. on p. 77).
- Ghimire, D. and J. Lee (2013). “Geometric feature-based facial expression recognition in image sequences using multi-class AdaBoost and support vector machines”. In: *Sensors* 13.6, pp. 7714–7734 (cit. on pp. 23, 30, 166).
- Glorot, Xavier and Yoshua Bengio (2010). “Understanding the difficulty of training deep feed-forward neural networks”. In: *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pp. 249–256 (cit. on p. 84).
- Goel, Navin, George Bebis, and Ara Nefian (2005). “Face recognition experiments with random projection”. In: *Defense and Security*. International Society for Optics and Photonics, pp. 426–437 (cit. on pp. 62, 63).
- Gralewski, Lisa, Neill Campbell, and Ian Penton-Voak (2006). “Using a tensor framework for the analysis of facial dynamics”. In: *Automatic Face and Gesture Recognition, 2006. FGR 2006. 7th International Conference on*. IEEE, pp. 217–222 (cit. on p. 31).
- Gratz, Kim L and Lizabeth Roemer (2004). “Multidimensional assessment of emotion regulation and dysregulation: Development, factor structure, and initial validation of the difficulties in emotion regulation scale”. In: *Journal of psychopathology and behavioral assessment* 26.1, pp. 41–54 (cit. on p. 29).
- Grauman, Kristen and Trevor Darrell (2007). “The pyramid match kernel: Efficient learning with sets of features”. In: *Journal of Machine Learning Research* 8.Apr, pp. 725–760 (cit. on p. 49).
- Gritti, Tommaso, Caifeng Shan, Vincent Jeanne, and Ralph Braspenning (2008). “Local features based facial expression recognition with face registration errors”. In: *2008 8th IEEE International Conference on Automatic Face & Gesture Recognition*. IEEE, pp. 1–8 (cit. on p. 26).
- Gu, Jinwei, Xiaodong Yang, Shalini De Mello, and Jan Kautz (2017). “Dynamic Facial Analysis: From Bayesian Filtering to Recurrent Neural Network”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (cit. on pp. 12, 23, 167).
- Gune, Omkar, Biplab Banerjee, and Subhasis Chaudhuri (2018). “Structure Aligning Discriminative Latent Embedding for Zero-Shot Learning.” In: *BMVC*, p. 218 (cit. on p. 107).
- Haggard, Ernest A and Kenneth S Isaacs (1966). “Micromomentary facial expressions as indicators of ego mechanisms in psychotherapy”. In: *Methods of research in psychotherapy*. Springer, pp. 154–165 (cit. on p. 137).
- Han, H., C. Otto, X. Liu, and A. K. Jain (2015). “Demographic Estimation from Face Images: Human vs. Machine Performance”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 37.6, pp. 1148–1161 (cit. on p. 1).
- Happy, S. L. and A. Routray (2015). “Automatic facial expression recognition using features of salient facial patches”. In: *IEEE Transactions on Affective Computing* 6.1, pp. 1–12 (cit. on p. 101).

- Hariri, Walid, Hedi Tabia, Nadir Farah, David Declercq, and Abdallah Benouareth (2017). “Geometrical and visual feature quantization for 3d face recognition”. In: *VISAPP 2017 12th Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications* (cit. on p. 41).
- Harris, Chris and Mike Stephens (1988). “A combined corner and edge detector.” In: *Alvey vision conference*. Vol. 15. Citeseer, pp. 10–5244 (cit. on pp. 32, 44).
- Hasani, B. and M. H. Mahoor (2017). “Spatio-Temporal Facial Expression Recognition Using Convolutional Neural Networks and Conditional Random Fields”. In: *2017 12th IEEE International Conference on Automatic Face Gesture Recognition (FG 2017)*, pp. 790–795 (cit. on pp. 12, 23, 167).
- He, K., X. Zhang, S. Ren, and J. Sun (2015a). “Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 37.9, pp. 1904–1916 (cit. on p. 74).
- He, Kaiming, Xiangyu Zhang, Shaoqing Ren, and Jian Sun (2015b). “Delving deep into rectifiers: Surpassing human-level performance on imagenet classification”. In: *Proceedings of the IEEE international conference on computer vision*, pp. 1026–1034 (cit. on p. 117).
- Hess, Ursula and Robert E Kleck (1990). “Differentiating emotion elicited and deliberate emotional facial expressions”. In: *European Journal of Social Psychology* 20.5, pp. 369–385 (cit. on p. 9).
- Hinton, Geoffrey E, Simon Osindero, and Yee-Whye Teh (2006). “A fast learning algorithm for deep belief nets”. In: *Neural Comput* 18.7, pp. 1527–1554 (cit. on p. 151).
- Hochreiter, Sepp and Jürgen Schmidhuber (1997). “Long short-term memory”. In: *Neural computation* 9.8, pp. 1735–1780 (cit. on p. 78).
- Hu, Guosheng et al. (2018). “Deep Multi-Task Learning to Recognise Subtle Facial Expressions of Mental States”. In: *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 103–119 (cit. on p. 8).
- Huang, Hung-Fu and Shen-Chuan Tai (2012). “Facial expression recognition using new feature extraction algorithm”. In: *ELCVIA: electronic letters on computer vision and image analysis* 11.1, pp. 41–54 (cit. on p. 8).
- Huang, Ke and Selin Aviyente (2006). “Sparse representation for signal classification”. In: *NIPS*. Vol. 19, pp. 609–616 (cit. on p. 54).
- (2007). “Sparse representation for signal classification”. In: *Advances in neural information processing systems*, pp. 609–616 (cit. on pp. 54, 56).
- Huang, Kuan-Chieh, Sheng-Yu Huang, and Yau-Hwang Kuo (2010). “Emotion recognition based on a novel triangular facial feature extraction method”. In: *The 2010 International Joint Conference on Neural Networks (IJCNN)*. IEEE, pp. 1–6 (cit. on p. 23).
- Huang, Xiaohua and Guoying Zhao (2017). “Spontaneous facial micro-expression analysis using spatiotemporal local radon-based binary pattern”. In: *the Frontiers and Advances in Data Science (FADS), 2017 International Conference on*. IEEE, pp. 159–164 (cit. on p. 138).
- Huang, Xun and Serge Belongie (2017). “Arbitrary style transfer in real-time with adaptive instance normalization”. In: *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1501–1510 (cit. on p. 117).

- Ionescu, Radu Tudor, Marius Popescu, and Cristian Grozea (2013). “Local learning to improve bag of visual words model for facial expression recognition”. In: *Workshop on challenges in representation learning, ICML* (cit. on p. 41).
- Iqbal, M. T. B., M. A. Wadud, B. Ryu, F. Makhmudkhujaev, and O. Chae (2018). “Facial Expression Recognition with Neighborhood-aware Edge Directional Pattern (NEDP)”. In: *IEEE Transactions on Affective Computing*, pp. 1–1 (cit. on pp. 11, 23, 165).
- Iwano, Y., M. Yoneyama, and K. Shirai (1996). “Recognition of facial expressions using associative memory”. In: *1996 IEEE Digital Signal Processing Workshop Proceedings*, pp. 243–246 (cit. on p. 2).
- Jaderberg, Max, Karen Simonyan, Andrew Zisserman, et al. (2015). “Spatial transformer networks”. In: *Advances in neural information processing systems*, pp. 2017–2025 (cit. on p. 119).
- Jain, Arjun, Jonathan Tompson, Mykhaylo Andriluka, Graham W Taylor, and Christoph Bregler (2013). “Learning human pose estimation features with convolutional networks”. In: *arXiv preprint arXiv:1312.7302* (cit. on pp. 20, 21).
- Jang, G., J. Park, A. Jo, and J. Kim (2014). “Facial Emotion Recognition Using Active Shape Models and Statistical Pattern Recognizers”. In: *2014 Ninth International Conference on Broadband and Wireless Computing, Communication and Applications*, pp. 514–517 (cit. on pp. 15, 33).
- Janu, Neha, Pratistha Mathur, Sandeep Kumar Gupta, and Shubh Lakshmi Agrwal (2017). “Performance analysis of frequency domain based feature extraction techniques for facial expression recognition”. In: *Cloud Computing, Data Science & Engineering-Confluence, 2017 7th International Conference on*. IEEE, pp. 591–594 (cit. on p. 14).
- Jayaraman, Dinesh and Kristen Grauman (2014). “Zero-shot recognition with unreliable attributes”. In: *Advances in neural information processing systems*, pp. 3464–3472 (cit. on p. 104).
- Jeni, László A, Jeffrey M Girard, Jeffrey F Cohn, and Fernando De La Torre (2013). “Continuous au intensity estimation using localized, sparse facial feature space”. In: *2013 10th IEEE international conference and workshops on automatic face and gesture recognition (FG)*. IEEE, pp. 1–7 (cit. on p. 24).
- Ji, Shuiwang, Wei Xu, Ming Yang, and Kai Yu (2013). “3D convolutional neural networks for human action recognition”. In: *IEEE transactions on pattern analysis and machine intelligence* 35.1, pp. 221–231 (cit. on p. 74).
- Jiang, Zhuolin, Zhe Lin, and Larry S Davis (2013). “Label consistent K-SVD: Learning a discriminative dictionary for recognition”. In: *IEEE transactions on pattern analysis and machine intelligence* 35.11, pp. 2651–2664 (cit. on pp. 61, 70).
- Jing, H., X. Lun, L. Dan, H. Zhijie, and W. Zhiliang (2015). “Cognitive emotion model for eldercare robot in smart home”. In: *China Communications* 12.4, pp. 32–41 (cit. on p. 3).
- Johnson, William B and Joram Lindenstrauss (1984). “Extensions of Lipschitz mappings into a Hilbert space”. In: *Contemporary mathematics* 26.189-206, p. 1 (cit. on p. 63).
- Jun, H., C. Jian-feng, F. Ling-zhi, and H. Zhong-wen (2015). “A method of facial expression recognition based on LBP fusion of key expressions areas”. In: *The 27th Chinese Control and Decision Conference (2015 CCDC)*, pp. 4200–4204 (cit. on p. 101).

- Jung, Heechul, Sihaeng Lee, Junho Yim, Sunjeong Park, and Junmo Kim (2015). “Joint fine-tuning in deep neural networks for facial expression recognition”. In: *Proceedings of the IEEE international conference on computer vision*, pp. 2983–2991 (cit. on p. 110).
- Kacem, Anis, Mohamed Daoudi, Boulbaba Ben Amor, and Juan Carlos Alvarez-Paiva (2017). “A Novel Space-Time Representation on the Positive Semidefinite Cone for Facial Expression Recognition”. In: *The IEEE International Conference on Computer Vision* (cit. on pp. 23, 30, 167).
- Kahou, Samira Ebrahimi et al. (2016). “Emonets: Multimodal deep learning approaches for emotion recognition in video”. In: *Journal on Multimodal User Interfaces* 10.2, pp. 99–111 (cit. on p. 32).
- Kaltwang, Sebastian, Ognjen Rudovic, and Maja Pantic (2012). “Continuous pain intensity estimation from facial expressions”. In: *International Symposium on Visual Computing*. Springer, pp. 368–377 (cit. on pp. 29, 33).
- Kaltwang, Sebastian, Sinisa Todorovic, and Maja Pantic (2015). “Latent Trees for Estimating Intensity of Facial Action Units”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (cit. on pp. 23, 167).
- Kamarainen, J-K, Ville Kyrki, and Heikki Kalviainen (2006). “Invariance properties of Gabor filter-based features-overview and applications”. In: *IEEE Transactions on image processing* 15.5, pp. 1088–1099 (cit. on p. 26).
- Kamarol, S. K. A., M. H. Jaward, J. Parkkinen, and R. Parthiban (2016). “Spatiotemporal feature extraction for facial expression recognition”. In: *IET Image Processing* 10.7, pp. 534–541 (cit. on pp. 23, 166).
- Kaski, Samuel (1998). “Dimensionality reduction by random mapping: Fast similarity computation for clustering”. In: *Neural Networks Proceedings, 1998. IEEE World Congress on Computational Intelligence. The 1998 IEEE International Joint Conference on*. Vol. 1. IEEE, pp. 413–418 (cit. on p. 63).
- Kaya, Heysem, Furkan Gürpınar, and Albert Ali Salah (2017). “Video-based emotion recognition in the wild using deep transfer learning and score fusion”. In: *Image and Vision Computing* 65, pp. 66–75 (cit. on p. 110).
- Kim, D., M. Sohn, H. Kim, and N. Ryu (2014). “Geometric Feature-Based Face Normalization for Facial Expression Recognition”. In: *2014 2nd International Conference on Artificial Intelligence, Modelling and Simulation*, pp. 172–175 (cit. on pp. 23, 101, 165).
- Kim, S. and H. Kim (2019). “Deep Explanation Model for Facial Expression Recognition Through Facial Action Coding Unit”. In: *2019 IEEE International Conference on Big Data and Smart Computing (BigComp)*, pp. 1–4 (cit. on pp. 16, 33).
- Kingma, Diederik and Jimmy Ba (2014). “Adam: A method for stochastic optimization”. In: *preprint arXiv:1412.6980* (cit. on pp. 83, 151).
- Kiperwasser, Eliyahu and Yoav Goldberg (2016). “Simple and Accurate Dependency Parsing Using Bidirectional LSTM Feature Representations”. In: *Transactions of the Association for Computational Linguistics* 4, pp. 313–327 (cit. on pp. 104, 105, 122).
- Kiran, B Ravi, Dilip Mathew Thomas, and Ranjith Parakkal (2018). “An overview of deep learning based methods for unsupervised and semi-supervised anomaly detection in videos”. In: *Journal of Imaging* 4.2, p. 36 (cit. on p. 139).

- Ko, Byoung (2018). “A brief review of facial emotion recognition based on visual information”. In: *sensors* 18.2, p. 401 (cit. on p. 15).
- Kodirov, Elyor, Tao Xiang, Zhenyong Fu, and Shaogang Gong (2015). “Unsupervised domain adaptation for zero-shot learning”. In: *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2452–2460 (cit. on p. 114).
- Koelstra, S., M. Pantic, and I. Patras (2010). “A Dynamic Texture-Based Approach to Recognition of Facial Actions and Their Temporal Models”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 32.11, pp. 1940–1954 (cit. on pp. 1, 23, 29, 166).
- Kolodziej, M., A. Majkowski, R. J. Rak, P. Tarnowski, and T. Pielaszekiewicz (2018). “Analysis of Facial Features for the Use of Emotion Recognition”. In: *19th International Conference Computational Problems of Electrical Engineering*, pp. 1–4 (cit. on pp. 15, 33).
- Kosti, Ronak, Jose M. Alvarez, Adria Recasens, and Agata Lapedriza (2017). “Emotion Recognition in Context”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (cit. on p. 8).
- Kottur, Satwik, Ramakrishna Vedantam, José MF Moura, and Devi Parikh (2016). “Visual word2vec (vis-w2v): Learning visually grounded word embeddings using abstract scenes”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4985–4994 (cit. on p. 114).
- Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E Hinton (2012). “Imagenet classification with deep convolutional neural networks”. In: *Advances in neural information processing systems*, pp. 1097–1105 (cit. on pp. 31, 96).
- Kuo Chieh-Ming, Shang-Hong Lai and Michel Sarkis (2018). “A compact deep learning model for robust facial expression recognition”. In: *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 2121–2129 (cit. on pp. 12, 23, 167).
- Lades, Martin, Jan C Vorbruggen, Joachim Buhmann, Jörg Lange, Christoph Von Der Malsburg, Rolf P Wurtz, and Wolfgang Konen (1993). “Distortion invariant object recognition in the dynamic link architecture”. In: *IEEE Transactions on computers* 3, pp. 300–311 (cit. on p. 26).
- Lake, Brenden, Ruslan Salakhutdinov, Jason Gross, and Joshua Tenenbaum (2011). “One shot learning of simple visual concepts”. In: *Proceedings of the Annual Meeting of the Cognitive Science Society*. Vol. 33. 33 (cit. on p. 110).
- Lampert, Christoph H, Hannes Nickisch, and Stefan Harmeling (2009). “Learning to detect unseen object classes by between-class attribute transfer”. In: *2009 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, pp. 951–958 (cit. on pp. 104, 110, 111).
- (2013). “Attribute-based classification for zero-shot visual object categorization”. In: *IEEE transactions on pattern analysis and machine intelligence* 36.3, pp. 453–465 (cit. on p. 110).
- Laptev, Ivan (2005). “On space-time interest points”. In: *International journal of computer vision* 64.2-3, pp. 107–123 (cit. on p. 30).
- Larochelle, Hugo and Geoffrey E Hinton (2010). “Learning to combine foveal glimpses with a third-order Boltzmann machine”. In: *Advances in neural information processing systems*, pp. 1243–1251 (cit. on p. 119).
- Laxhammar, Rikard, Goran Falkman, and Egils Sviestins (2009). “Anomaly detection in sea traffic—a comparison of the gaussian mixture model and the kernel density estimator”. In:



- Information Fusion, 2009. FUSION'09. 12th International Conference on.* IEEE, pp. 756–763 (cit. on p. 144).
- Lazaridou, Angeliki, Nghia The Pham, and Marco Baroni (2015a). “Combining language and vision with a multimodal skip-gram model”. In: *In Annual Conference of the North American Chapter of the Association for Computational Linguistics* (cit. on p. 114).
- Lazaridou, Angeliki, Georgiana Dinu, and Marco Baroni (2015b). “Hubness and pollution: Delving into cross-space mapping for zero-shot learning”. In: *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. Vol. 1, pp. 270–280 (cit. on p. 107).
- Lazebnik, S., C. Schmid, and J. Ponce (2006a). “Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories”. In: *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*. Vol. 2, pp. 2169–2178 (cit. on pp. 26, 43, 47).
- Lazebnik, Svetlana, Cordelia Schmid, and Jean Ponce (2006b). “Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories”. In: *Computer vision and pattern recognition, 2006 IEEE computer society conference on.* Vol. 2. IEEE, pp. 2169–2178 (cit. on p. 81).
- LE Vuong, TANG Hao-et HUANG Thomas S. (2011). “Expression recognition from 3D dynamic faces using robust spatio-temporal shape features.” In: *Face and Gesture 2011. IEEE*, pp. 414–421 (cit. on pp. 23, 166).
- LeCun, Yann (2012). “Learning invariant feature hierarchies”. In: *European conference on computer vision*. Springer, pp. 496–505 (cit. on p. 32).
- LeCun, Yann, Yoshua Bengio, and Geoffrey Hinton (2015). “Deep learning”. In: *nature* 521.7553, p. 436 (cit. on p. 28).
- Leskovec, Jure, Anand Rajaraman, and Jeffrey David Ullman (2014). *Mining of massive datasets*. Cambridge University Press (cit. on p. 46).
- Li, J., C. Soladie, and R. Segquier (2018). “LTP-ML: Micro-Expression Detection by Recognition of Local Temporal Pattern of Facial Movements”. In: *Proc Int Conf Autom Face Gesture Recognit*, pp. 634–641 (cit. on p. 143).
- Li, Ming and Baozong Yuan (2005). “2D-LDA: A statistical linear discriminant analysis for image matrix”. In: *Pattern Recognition Letters* 26.5, pp. 527–532 (cit. on p. 54).
- Li, Shan and Weihong Deng (2018). “Deep facial expression recognition: A survey”. In: *arXiv preprint arXiv:1804.08348* (cit. on p. 15).
- Li, Xiaobai, Tomas Pfister, Xiaohua Huang, Guoying Zhao, and Matti Pietikäinen (2013). “A spontaneous micro-expression database: Inducement, collection and baseline”. In: *Proc Int Conf Autom Face Gesture Recognit*. IEEE, pp. 1–6 (cit. on pp. 10, 140, 141, 157, 159).
- Li, Xiaobai, HONG Xiaopeng, Antti Moilanen, Xiaohua Huang, Tomas Pfister, Guoying Zhao, and Matti Pietikainen (2017). “Towards reading hidden emotions: A comparative study of spontaneous micro-expression spotting and recognition methods”. In: *IEEE T Affect Comput* (cit. on pp. 138, 143, 159).
- Liliana, D. Y., M. R. Widyanto, and T. Basaruddin (2018). “Geometric Facial Components Feature Extraction for Facial Expression Recognition”. In: *2018 International Conference*

- on *Advanced Computer Science and Information Systems (ICACSIS)*, pp. 391–396 (cit. on pp. 8, 23, 166).
- Lim, Ching Leng Peter, Wai Lok Woo, Satnam S Dlay, and Bin Gao (2019). “Heart-rate-Dependent Heartwave Biometric Identification With Thresholding-Based GMM–HMM Methodology”. In: *IEEE Intl Conf Ind I* 15.1, pp. 45–53 (cit. on p. 144).
- Liong, Sze-Teng, John See, Raphael C-W Phan, Anh Cat Le Ngo, Yee-Hui Oh, and KokSheik Wong (2014). “Subtle expression recognition using optical strain weighted features”. In: *Asian Conference on Computer Vision*. Springer, pp. 644–657 (cit. on pp. 142, 159).
- Littlewort, G., J. Whitehill, T. Wu, N. Butko, P. Ruvolo, J. Movellan, and M. Bartlett (2011). “The motion in emotion — A CERT based approach to the FERa emotion challenge”. In: *Face and Gesture 2011*, pp. 897–902 (cit. on pp. 11, 23, 165).
- Liu, Mengyi, Ruiping Wang, Shaoxin Li, Shiguang Shan, Zhiwu Huang, and Xilin Chen (2014a). “Combining Multiple Kernel Methods on Riemannian Manifold for Emotion Recognition in the Wild”. In: *Proceedings of the 16th International Conference on Multimodal Interaction*. ICMI ’14. Istanbul, Turkey: ACM, pp. 494–501 (cit. on pp. 23, 166).
- (2014b). “Combining multiple kernel methods on riemannian manifold for emotion recognition in the wild”. In: *Proceedings of the 16th International Conference on multimodal interaction*. ACM, pp. 494–501 (cit. on p. 26).
- Liu, Mengyi, Shaoxin Li, Shiguang Shan, Ruiping Wang, and Xilin Chen (2014c). “Deeply learning deformable facial action parts model for dynamic expression analysis”. In: *Asian Conference on Computer Vision*. Springer, pp. 143–157 (cit. on p. 95).
- Liu, Mengyi, Shiguang Shan, Ruiping Wang, and Xilin Chen (2014d). “Learning Expressionlets on Spatio-Temporal Manifold for Dynamic Facial Expression Recognition”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (cit. on pp. 12, 23, 167).
- Liu, P. and L. Yin (2015). “Spontaneous facial expression analysis based on temperature changes and head motions”. In: *2015 11th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG)*. Vol. 1, pp. 1–6 (cit. on pp. 23, 166).
- Liu, Ping, Shizhong Han, Zibo Meng, and Yan Tong (2014e). “Facial Expression Recognition via a Boosted Deep Belief Network”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (cit. on pp. 23, 167).
- Long, Fei, Tingfan Wu, Javier R Movellan, Marian S Bartlett, and Gwen Littlewort (2012). “Learning spatiotemporal features by using independent component analysis with application to facial expression recognition”. In: *Neurocomputing* 93, pp. 126–132 (cit. on p. 32).
- Long, Yang, Li Liu, and Ling Shao (2016). “Attribute embedding with visual semantic ambiguity removal for zero shot learning”. In: (cit. on p. 114).
- Lowe, David (1999). “Object recognition from local scale-invariant features.” In: *iccv*. Vol. 99. 2, pp. 1150–1157 (cit. on p. 24).
- (2004). “Distinctive image features from scale-invariant keypoints”. In: *International journal of computer vision* 60.2, pp. 91–110 (cit. on pp. 14, 44).
- Lu, Y., S. Wang, W. Zhao, Y. Zhao, and J. Wei (2017). “A novel approach of facial expression recognition based on shearlet transform”. In: *2017 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, pp. 398–402 (cit. on pp. 23, 166).
- Lucey, P., J. F. Cohn, I. Matthews, S. Lucey, S. Sridharan, J. Howlett, and K. M. Prkachin (2011). “Automatically Detecting Pain in Video Through Facial Action Units”. In: *IEEE*

- Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* 41.3, pp. 664–674 (cit. on p. 3).
- Lucey, Patrick, Jeffrey F Cohn, Takeo Kanade, Jason Saragih, Zara Ambadar, and Iain Matthews (2010). “The extended cohn-kanade dataset (ck+): A complete dataset for action unit and emotion-specified expression”. In: *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition-Workshops*. IEEE, pp. 94–101 (cit. on pp. 11, 39, 126).
- Lucey, Patrick, Jeffrey F. Cohn, Kenneth M. Prkachin, Patricia E. Solomon, Sien Chew, and Iain Matthews (2012). “Painful monitoring: Automatic pain monitoring using the UNBC-McMaster shoulder pain expression archive database”. In: *Image and Vision Computing* 30.3. Best of Automatic Face and Gesture Recognition 2011, pp. 197–205 (cit. on p. 21).
- Lucey, Simon, Ahmed Bilal Ashraf, and Jeffrey F Cohn (2007). “Investigating spontaneous facial action recognition through aam representations of the face”. In: *Face recognition*. IntechOpen (cit. on p. 23).
- Lyons, Michael, Shota Akamatsu, Miyuki Kamachi, and Jiro Gyoba (1998). “Coding facial expressions with gabor wavelets”. In: *Automatic Face and Gesture Recognition, 1998. Proceedings. Third IEEE International Conference on*. IEEE, pp. 200–205 (cit. on p. 38).
- M. Norouzi T. Mikolov, S. Bengio Y. Singer J. Shlens A. Frome G. S. Corrado and J. Dean. (2013). “Devise: A deep visual-semantic embedding model”. In: *In NIPS*, pp. 2121–2129 (cit. on pp. 113, 122).
- Maaten, Laurens van der and Geoffrey Hinton (2008). “Visualizing data using t-SNE”. In: *Journal of Machine Learning Research* 9.Nov, pp. 2579–2605 (cit. on p. 88).
- Magen, Avner (2002). “Dimensionality reductions that preserve volumes and distance to affine spaces, and their algorithmic applications”. In: *Randomization and approximation techniques in computer science*. Springer, pp. 239–253 (cit. on p. 63).
- Mahoor, Mohammad H, Mu Zhou, Kevin L Veon, S Mohammad Mavadati, and Jeffrey F Cohn (2011). “Facial action unit recognition with sparse representation”. In: *Face and Gesture 2011*. IEEE, pp. 336–342 (cit. on pp. 27, 28).
- Mairal, Julien, Michael Elad, and Guillermo Sapiro (2008). “Sparse representation for color image restoration”. In: *Image Processing, IEEE Transactions on* 17.1, pp. 53–69 (cit. on p. 54).
- Mairal, Julien, Francis Bach, Jean Ponce, and Guillermo Sapiro (2010). “Online learning for matrix factorization and sparse coding”. In: *The Journal of Machine Learning Research* 11, pp. 19–60 (cit. on p. 55).
- Majumder, Anima, Laxmidhar Behera, and Venkatesh K Subramanian (2014). “Local binary pattern based facial expression recognition using Self-organizing Map”. In: *2014 International Joint Conference on Neural Networks (IJCNN)*. IEEE, pp. 2375–2382 (cit. on p. 8).
- Mallat, Stéphane G and Zhifeng Zhang (1993). “Matching pursuits with time-frequency dictionaries”. In: *IEEE Transactions on signal processing* 41.12, pp. 3397–3415 (cit. on p. 58).
- Mao, Junhua, Xu Wei, Yi Yang, Jiang Wang, Zhiheng Huang, and Alan L Yuille (2015). “Learning like a child: Fast novel visual concept learning from sentence descriptions of images”. In: *Proceedings of the IEEE international conference on computer vision*, pp. 2533–2541 (cit. on p. 110).

- Marchi, Erik, Fabio Vesperini, Florian Eyben, Stefano Squartini, and Björn Schuller (2015a). “A novel approach for automatic acoustic novelty detection using a denoising autoencoder with bidirectional LSTM neural networks”. In: *Int Conf Acoust Spee*. IEEE, pp. 1996–2000 (cit. on p. 144).
- Marchi, Erik, Fabio Vesperini, Felix Weninger, Florian Eyben, Stefano Squartini, and Björn Schuller (2015b). “Non-linear prediction with LSTM recurrent neural networks for acoustic novelty detection”. In: *IEEE IJCNN*. IEEE, pp. 1–7 (cit. on p. 144).
- Martinez, B., M. F. Valstar, X. Binefa, and M. Pantic (2013). “Local Evidence Aggregation for Regression-Based Facial Point Detection”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35.5, pp. 1149–1163 (cit. on pp. 23, 165).
- Mase, Kenji (1991). “Recognition of facial expression from optical flow”. In: *IEICE TRANSACTIONS on Information and Systems* 74.10, pp. 3474–3483 (cit. on p. 2).
- Masters, Barry R, Rafael C Gonzalez, and Richard Woods (2009). “Digital image processing”. In: *Journal of biomedical optics* 14.2, p. 029901 (cit. on p. 81).
- Matsumoto, David and Hyi Sung Hwang (2011). “Evidence for training the ability to read microexpressions of emotion”. In: *Motivation and Emotion* 35.2, pp. 181–191 (cit. on p. 137).
- Mavadati, S Mohammad, Mohammad H Mahoor, Kevin Bartlett, Philip Trinh, and Jeffrey F Cohn (2013). “Disfa: A spontaneous facial action intensity database”. In: *IEEE Transactions on Affective Computing* 4.2, pp. 151–160 (cit. on pp. 6, 40, 126).
- Mensink, Thomas, Efstratios Gavves, and Cees GM Snoek (2014). “Costa: Co-occurrence statistics for zero-shot classification”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2441–2448 (cit. on pp. 104, 114).
- Mikolov, Tomas, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean (2013). “Distributed representations of words and phrases and their compositionality”. In: *Advances in neural information processing systems*, pp. 3111–3119 (cit. on pp. 104, 105).
- Mnih, Volodymyr, Nicolas Heess, Alex Graves, et al. (2014). “Recurrent models of visual attention”. In: *Advances in neural information processing systems*, pp. 2204–2212 (cit. on p. 119).
- Mohammad Mahoor, Behzad H et al. (2017). “Facial Expression Recognition Using Enhanced Deep 3D Convolutional Neural Networks”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 30–40 (cit. on pp. 32, 72, 95, 97).
- Mohammadi, MR, Emad Fatemizadeh, and Mohammad H Mahoor (2014). “PCA-based dictionary building for accurate facial expression recognition via sparse representation”. In: *Journal of Visual Communication and Image Representation* 25.5, pp. 1082–1092 (cit. on p. 95).
- Moilanen, Antti, Guoying Zhao, and Matti Pietikäinen (2014). “Spotting rapid facial movements from videos using appearance-based feature difference analysis”. In: *Int C Patt Recog*. IEEE, pp. 1722–1727 (cit. on pp. 138, 142).
- Mollahosseini, A., D. Chan, and M. H. Mahoor (2016). “Going deeper in facial expression recognition using deep neural networks”. In: *2016 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 1–10 (cit. on p. 72).
- Mollahosseini, Ali, Behzad Hasani, Michelle J Salvador, Hojjat Abdollahi, David Chan, and Mohammad H Mahoor (2016a). “Facial expression recognition from world wild web”. In:

- Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 58–65 (cit. on p. 72).
- Mollahosseini, Ali, David Chan, and Mohammad H Mahoor (2016b). “Going deeper in facial expression recognition using deep neural networks”. In: *Applications of Computer Vision (WACV), 2016 IEEE Winter Conference on*. IEEE, pp. 1–10 (cit. on pp. 32, 95).
- Moon, Hankyu, Rajeev Sharma, and Namsoon Jung (2013). *Method and system for measuring human response to visual stimulus based on changes in facial expression*. US Patent 8,462,996 (cit. on p. 15).
- Mostafa, A., M. I. Khalil, and H. Abbas (2018). “Emotion Recognition by Facial Features using Recurrent Neural Networks”. In: *2018 13th International Conference on Computer Engineering and Systems (ICCES)*, pp. 417–422 (cit. on pp. 15, 33).
- Munir Asim, Ayyaz Hussain Sajid Ali Khan Muhammad Nadeem and Sadia Arshid (2018). “Illumination invariant facial expression recognition using selected merged binary patterns for real world images”. In: *Optik-Elsevier*. Vol. 158, pp. 1016–1025 (cit. on pp. 11, 23, 165).
- Nicolle, Jérémie, Vincent Rapp, Kévin Bailly, Lionel Prevost, and Mohamed Chetouani (2012). “Robust continuous prediction of human emotions using multiscale dynamic cues”. In: *Proceedings of the 14th ACM international conference on Multimodal interaction*. ACM, pp. 501–508 (cit. on p. 33).
- Nikisins, Olegs, Amir Mohammadi, André Anjos, and Sébastien Marcel (2018). “On Effectiveness of Anomaly Detection Approaches against Unseen Presentation Attacks in Face Anti-Spoofing”. In: *The 11th IAPR International Conference on Biometrics (ICB 2018)*. EPFL-CONF-233583 (cit. on p. 144).
- Nikitidis, Symeon, Anastasios Tefas, Nikos Nikolaidis, and Ioannis Pitas (2012). “Subclass discriminant nonnegative matrix factorization for facial image analysis”. In: *Pattern Recognition* 45.12, pp. 4080–4091 (cit. on p. 27).
- Norouzi, Mohammad, Tomas Mikolov, Samy Bengio, Yoram Singer, Jonathon Shlens, Andrea Frome, Greg S Corrado, and Jeffrey Dean (2014). “Zero-shot learning by convex combination of semantic embeddings”. In: (cit. on pp. 113, 114).
- Olshausen, Bruno, Phil Sallee, and Michael Lewicki (2001). “Learning sparse image codes using a wavelet pyramid architecture”. In: *Advances in neural information processing systems*, pp. 887–893 (cit. on pp. 55, 58).
- Osada, Robert, Thomas Funkhouser, Bernard Chazelle, and David Dobkin (2001). “Matching 3D models with shape distributions”. In: *Proceedings International Conference on Shape Modeling and Applications*. IEEE, pp. 154–166 (cit. on p. 30).
- Pantic, Maja (2009). “Machine analysis of facial behaviour: Naturalistic and dynamic behaviour”. In: *Philosophical Transactions of the Royal Society B: Biological Sciences* 364.1535, pp. 3505–3513 (cit. on pp. 19, 29, 33).
- Pantic, Maja and Leon JM Rothkrantz (2000). “Automatic analysis of facial expressions: The state of the art”. In: *IEEE Transactions on Pattern Analysis & Machine Intelligence* 12, pp. 1424–1445 (cit. on pp. 1, 4, 19, 21).
- Pantic, Maja, Michel Valstar, Ron Rademaker, and Ludo Maat (2005). “Web-based database for facial expression analysis”. In: *Multimedia and Expo, 2005. ICME 2005. IEEE International Conference on*. IEEE, 5–pp (cit. on pp. 6, 40, 126).

- Parikh, Devi and Kristen Grauman (2011). “Relative attributes”. In: *2011 International Conference on Computer Vision*. IEEE, pp. 503–510 (cit. on p. 110).
- Parthasarathy, Srinivas and Carlos Busso (2017). “Jointly Predicting Arousal, Valence and Dominance with Multi-Task Learning.” In: *INTERSPEECH*, pp. 1103–1107 (cit. on p. 102).
- Pashov Alex, Alex (2019). “Muscles of facial expressions and how they work”. In: (cit. on p. 5).
- Patel, Devangini, Guoying Zhao, and Matti Pietikäinen (2015). “Spatiotemporal integration of optical flow vectors for micro-expression detection”. In: *Int C ACIVS*. Springer, pp. 369–380 (cit. on pp. 138, 143).
- Pati, Yagyensh Chandra, Ramin Rezaifar, and PS Krishnaprasad (1993). “Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition”. In: *Signals, Systems and Computers, 1993. 1993 Conference Record of The Twenty-Seventh Asilomar Conference on*. IEEE, pp. 40–44 (cit. on pp. 55, 58).
- Peng, Xiaojiang, Limin Wang, Xingxing Wang, and Yu Qiao (2016). “Bag of visual words and fusion methods for action recognition: Comprehensive study and good practice”. In: *Computer Vision and Image Understanding* 150, pp. 109–125 (cit. on p. 41).
- Pennington, Jeffrey, Richard Socher, and Christopher Manning (2014). “Glove: Global vectors for word representation”. In: *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pp. 1532–1543 (cit. on pp. 104, 105, 122).
- Pfister, Tomas, Xiaobai Li, Guoying Zhao, and Matti Pietikäinen (2011). “Recognising spontaneous facial micro-expressions”. In: *Proc ICCV IEEE*, pp. 1449–1456 (cit. on pp. 138, 142).
- Phillips, P Jonathon and Alice J O’toole (2014). “Comparison of human and computer performance across face recognition experiments”. In: *Image and Vision Computing* 32.1, pp. 74–85 (cit. on p. 1).
- Picard, Rosalind W (1997). “Affective computing MIT press”. In: *Cambridge, Massachusetts* (cit. on p. 2).
- Pinheiro, Pedro O (2018). “Unsupervised domain adaptation with similarity learning”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 8004–8013 (cit. on pp. 107, 108).
- Polikovskiy, Senya, Yoshinari Kameda, and Yuichi Ohta (2009). “Facial micro-expressions recognition using high speed camera and 3D-gradient descriptor”. In: (cit. on p. 137).
- Porter, Stephen and Leanne Ten Brinke (2008). “Reading between the lies: Identifying concealed and falsified emotions in universal facial expressions”. In: *Psychological science* 19.5, pp. 508–514 (cit. on pp. 10, 138).
- Qiao, Ruizhi, Lingqiao Liu, Chunhua Shen, and Anton van den Hengel (2017). “Visually aligned word embeddings for improving zero-shot learning”. In: *Proceedings of the British Machine Vision Conference* (cit. on pp. 106, 107, 110, 114, 126).
- Ramírez Cornejo, J. Y. and H. Pedrini (2018). “Emotion Recognition from Occluded Facial Expressions Using Weber Local Descriptor”. In: *2018 25th International Conference on Systems, Signals and Image Processing (IWSSIP)*, pp. 1–5 (cit. on pp. 15, 33).
- Ranzato, M, Joshua Susskind, Volodymyr Mnih, and Geoffrey Hinton (2011). “On deep generative models with applications to recognition”. In: (cit. on p. 29).

- Reed, Scott, Zeynep Akata, Honglak Lee, and Bernt Schiele (2016). “Learning deep representations of fine-grained visual descriptions”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 49–58 (cit. on p. 110).
- Rifai, Salah, Yoshua Bengio, Aaron Courville, Pascal Vincent, and Mehdi Mirza (2012). “Disentangling factors of variation for facial expression recognition”. In: *European Conference on Computer Vision*. Springer, pp. 808–822 (cit. on p. 29).
- Rinn, William E (1984). “The neuropsychology of facial expression: A review of the neurological and psychological mechanisms for producing facial expressions”. In: *Psychological bulletin* 95.1, pp. 52–77 (cit. on pp. 8, 12).
- Romera-Paredes, Bernardino and Philip Torr (2015). “An embarrassingly simple approach to zero-shot learning”. In: *International Conference on Machine Learning*, pp. 2152–2161 (cit. on p. 126).
- Roy, Abhinaba, Jacopo Cavazza, and Vittorio Murino (2018). “Visually-Driven Semantic Augmentation for Zero-Shot Learning.” In: *BMVC*, p. 85 (cit. on pp. 107, 114).
- Rozin, Paul and Adam B Cohen (2003). “High frequency of facial expressions corresponding to confusion, concentration, and worry in an analysis of naturally occurring facial expressions of Americans.” In: *Emotion* 3.1, p. 68 (cit. on p. 8).
- Rubinstein, Ron, Alfred M Bruckstein, and Michael Elad (2010). “Dictionaries for sparse representation modeling”. In: *Proceedings of the IEEE* 98.6, pp. 1045–1057 (cit. on p. 55).
- Russell, Tamara A, Elvina Chu, and Mary L Phillips (2006). “A pilot study to investigate the effectiveness of emotion recognition remediation in schizophrenia using the micro-expression training tool”. In: *Brit J Clin Psychol* 45.4, pp. 579–583 (cit. on p. 138).
- Saenko, Kate, Brian Kulis, Mario Fritz, and Trevor Darrell (2010). “Adapting visual category models to new domains”. In: *European conference on computer vision*. Springer, pp. 213–226 (cit. on p. 107).
- Sajjad, Muhammad, Adnan Shah, Zahoor Jan, Syed Inayat Shah, Sung Wook Baik, and Irfan Mehmood (2018). “Facial appearance and texture feature-based robust facial expression recognition framework for sentiment knowledge discovery”. In: *Cluster Computing* 21.1, pp. 549–567 (cit. on pp. 11, 23, 165).
- Salah, Albert Ali, Nicu Sebe, and Theo Gevers (2011). “Communication and automatic interpretation of affect from facial expressions”. In: *Affective Computing and Interaction: Psychological, Cognitive and Neuroscientific Perspectives*. IGI Global, pp. 157–183 (cit. on p. 19).
- Samir, Chafik, Anuj Srivastava, and Mohamed Daoudi (2006). “Three-dimensional face recognition using shapes of facial curves”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 28.11, pp. 1858–1863 (cit. on p. 29).
- Samir, Chafik, Anuj Srivastava, Mohamed Daoudi, and Eric Klassen (2009). “An intrinsic framework for analysis of facial surfaces”. In: *International Journal of Computer Vision* 82.1, pp. 80–95 (cit. on p. 29).
- Sandbach, G., S. Zafeiriou, M. Pantic, and D. Rueckert (2011). “A dynamic approach to the recognition of 3D facial expressions and their temporal models”. In: *Face and Gesture 2011*, pp. 406–413 (cit. on pp. 23, 30, 166).

- Sandbach, Georgia, Stefanos Zafeiriou, Maja Pantic, and Lijun Yin (2012). “Static and dynamic 3D facial expression recognition: A comprehensive survey”. In: *Image and Vision Computing* 30.10, pp. 683–697 (cit. on p. 19).
- Sanghai, Kaushal, Ting Su, Jennifer Dy, and David Kaeli (2005). “A multinomial clustering model for fast simulation of computer architecture designs”. In: *Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*. ACM, pp. 808–813 (cit. on p. 63).
- Sariyanidi, Evangelos, Hatice Gunes, and Andrea Cavallaro (2015). “Automatic analysis of facial affect: A survey of registration, representation, and recognition”. In: *IEEE transactions on pattern analysis and machine intelligence* 37.6, pp. 1113–1133 (cit. on pp. 19–21).
- Sauter, Disa A and Agneta H Fischer (2018). “Can perceivers recognise emotions from spontaneous expressions?” In: *Cognition and Emotion* 32.3, pp. 504–515 (cit. on p. 9).
- Savran, Arman, Houwei Cao, Miraj Shah, Ani Nenkova, and Ragini Verma (2012). “Combining video, audio and lexical indicators of affect in spontaneous conversation via particle filtering”. In: *Proceedings of the 14th ACM international conference on Multimodal interaction*. ACM, pp. 485–492 (cit. on p. 26).
- Scherer, Klaus R and Paul Ekman (1982). *Handbook of methods in nonverbal behavior research*. Vol. 2. Cambridge University Press Cambridge (cit. on p. 33).
- Scherer, Maximilian, Michael Walter, and Tobias Schreck (2010). “Histograms of oriented gradients for 3D object retrieval”. In: *In Europe on Computer Graphics, Visualization and Computer Vision*, p. 8 (cit. on p. 96).
- Scovanner, Paul, Saad Ali, and Mubarak Shah (2007). “A 3-dimensional sift descriptor and its application to action recognition”. In: *Proceedings of the 15th ACM international conference on Multimedia*. ACM, pp. 357–360 (cit. on pp. 15, 30, 96).
- Sebe, Nicu, Michael S Lew, Yafei Sun, Ira Cohen, Theo Gevers, and Thomas S Huang (2007). “Authentic facial expression analysis”. In: *Image and Vision Computing* 25.12, pp. 1856–1863 (cit. on p. 2).
- Shan, Caifeng, Shaogang Gong, and Peter W McOwan (2009). “Facial expression recognition based on local binary patterns: A comprehensive study”. In: *Image and vision Computing* 27.6, pp. 803–816 (cit. on pp. 8, 25, 32, 95).
- Shangfei WANG Menghua HE, Zhen GAO Shan HE Qiang JI (2014). “Emotion recognition from thermal infrared images using deep Boltzmann machine”. In: *Frontiers of Computer Science* 8.4, 609, p. 609 (cit. on p. 23).
- Sharma Shikhar, Ryan Kiros and Ruslan Salakhutdinov. (2015). “Action recognition using visual attention.” In: *ICLR* (cit. on p. 117).
- Shen, Xun-bing, Qi Wu, and Xiao-lan Fu (2012). “Effects of the duration of expressions on the recognition of microexpressions”. In: *Journal of Zhejiang University Science B* 13.3, pp. 221–230 (cit. on p. 137).
- Shigeto, Yutaro, Ikumi Suzuki, Kazuo Hara, Masashi Shimbo, and Yuji Matsumoto (2015). “Ridge regression, hubness, and zero-shot learning”. In: *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, pp. 135–151 (cit. on p. 107).



- Shreve, M., S. Godavarthy, V. Manohar, D. Goldgof, and S. Sarkar (2009). “Towards macro- and micro-expression spotting in video using strain patterns”. In: *2009 Workshop on Applications of Computer Vision (WACV)*, pp. 1–6 (cit. on p. 10).
- Siegelmann, Hava T and Eduardo D Sontag (1991). “Turing computability with neural nets”. In: *Applied Mathematics Letters* 4.6, pp. 77–80 (cit. on p. 77).
- Silva-Palacios, Daniel, Cèsar Ferri, and María José Ramírez-Quintana (2017). “Improving Performance of Multiclass Classification by Inducing Class Hierarchies”. In: *Procedia Computer Science* 108, pp. 1692–1701 (cit. on p. 95).
- Simon, Tomas, Minh Hoai Nguyen, Fernando De La Torre, and Jeffrey F Cohn (2010). “Action unit detection with segment-based svms”. In: *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. IEEE, pp. 2737–2744 (cit. on p. 31).
- Sivic, Josef and Andrew Zisserman (2003). “Video Google: A text retrieval approach to object matching in videos”. In: *null*. IEEE, p. 1470 (cit. on pp. 26, 41).
- Socher Richard, Milind Ganjoo Christopher D. Manning and Andrew Ng. (2013). “Zero-shot learning through cross-modal transfer.” In: *NIPS*, pp. 935–943 (cit. on p. 113).
- Sodemann, Angela A, Matthew P Ross, and Brett J Borghetti (2012). “A review of anomaly detection in automated surveillance”. In: *IEEE T Syst Man Cyb* 42.6, pp. 1257–1272 (cit. on p. 139).
- Srivastava, Nitish, Geoffrey E Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov (2014). “Dropout: a simple way to prevent neural networks from overfitting.” In: *J Mach Learn Res* 15.1, pp. 1929–1958 (cit. on pp. 76, 151).
- Su, Z., J. Chen, and H. Chen (2014). “Dynamic facial expression recognition using autoregressive models”. In: *2014 7th International Congress on Image and Signal Processing*, pp. 475–479 (cit. on pp. 23, 30, 166).
- Sulic, Vildana, Janez Perš, Matej Kristan, and Stanislav Kovacic (2010). “Efficient dimensionality reduction using random projection”. In: *15th Computer Vision Winter Workshop*, pp. 29–36 (cit. on p. 62).
- Sun, Bo, Liandong Li, Tian Zuo, Ying Chen, Guoyan Zhou, and Xuewen Wu (2014). “Combining Multimodal Features with Hierarchical Classifier Fusion for Emotion Recognition in the Wild”. In: *Proceedings of the 16th International Conference on Multimodal Interaction*. ICMI ’14. Istanbul, Turkey: ACM, pp. 481–486 (cit. on pp. 23, 166).
- Suwa, Motoi (1978). “A preliminary note on pattern recognition of human emotional expression”. In: *Proc. of The 4th International Joint Conference on Pattern Recognition*, pp. 408–410 (cit. on p. 2).
- Taheri, Sima, Qiang Qiu, and Rama Chellappa (2014). “Structure-preserving sparse decomposition for facial expression analysis”. In: *IEEE Transactions on Image Processing* 23.8, pp. 3590–3603 (cit. on p. 95).
- Taigman, Yaniv, Ming Yang, Marc’Aurelio Ranzato, and Lior Wolf (2015). “Web-scale training for face identification”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2746–2754 (cit. on p. 118).
- Takeuchi, Akikazu and Katashi Nagao (1993). “Communicative facial displays as a new conversational modality”. In: *Proceedings of the INTERACT’93 and CHI’93 Conference on Human Factors in Computing Systems*. ACM, pp. 187–193 (cit. on p. 1).

- Tang, Hao and Thomas S Huang (2008). “3D facial expression recognition based on automatically selected features”. In: *2008 IEEE computer society conference on computer vision and pattern recognition workshops*. IEEE, pp. 1–8 (cit. on p. 29).
- Tcherkassof, Anna, Damien Dupré, Brigitte Meillon, Nadine Mandran, Michel Dubois, and Jean-Michel Adam (2013). “DynEmo: A video database of natural facial expressions of emotions.” In: *The International Journal of Multimedia & Its Applications* 5.5, pp. 61–80 (cit. on pp. 7, 40, 126).
- Tian, Y-I, Takeo Kanade, and Jeffrey F Cohn (2001). “Recognizing action units for facial expression analysis”. In: *IEEE Transactions on pattern analysis and machine intelligence* 23.2, pp. 97–115 (cit. on p. 23).
- Tian, Ying-Li, Takeo Kanade, and Jeffrey F Cohn (2005). “Facial expression analysis”. In: *Handbook of face recognition*. Springer, pp. 247–275 (cit. on pp. 8, 12, 13).
- Tong, Y., J. Chen, and Q. Ji (2010). “A Unified Probabilistic Framework for Spontaneous Facial Action Modeling and Understanding”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 32.2, pp. 258–273 (cit. on p. 15).
- Torralba, Antonio and Alexei A Efros (2011). “Unbiased look at dataset bias”. In: pp. 1521–1528 (cit. on p. 102).
- Tran, L., X. Yin, and X. Liu (2017). “Disentangled Representation Learning GAN for Pose-Invariant Face Recognition”. In: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1283–1292 (cit. on p. 15).
- Tran, Thuong-Khanh, Xiaopeng Hong, and Guoying Zhao (2017). “Sliding Window Based Micro-expression Spotting: A Benchmark”. In: *International Conference on Advanced Concepts for Intelligent Vision Systems*. Springer, pp. 542–553 (cit. on p. 138).
- Tropp, Joel A and Anna C Gilbert (2007). “Signal recovery from random measurements via orthogonal matching pursuit”. In: *IEEE Transactions on information theory* 53.12, pp. 4655–4666 (cit. on p. 58).
- Tropp, Joel A, Anna C Gilbert, and Martin J Strauss (2006). “Algorithms for simultaneous sparse approximation. Part I: Greedy pursuit”. In: *Signal processing* 86.3, pp. 572–588 (cit. on p. 58).
- Tsagakatakis, Grigorios and Andreas Savakis (2009). “A random projections model for object tracking under variable pose and multi-camera views”. In: *Distributed Smart Cameras, 2009. ICDS-C 2009. Third ACM/IEEE International Conference on*. IEEE, pp. 1–7 (cit. on p. 63).
- Tunç, Birkan, Volkan Dağlı, and Muhittin Gökmen (2012). “Class dependent factor analysis and its application to face recognition”. In: *Pattern Recognition* 45.12, pp. 4092–4102 (cit. on p. 27).
- Turner, Mark (1986). “Texture discrimination by Gabor functions”. In: *Biological cybernetics* 55.2-3, pp. 71–82 (cit. on p. 24).
- Valstar, M. F. and M. Pantic (2006). “Biologically vs. Logic Inspired Encoding of Facial Actions and Emotions in Video”. In: *2006 IEEE International Conference on Multimedia and Expo*, pp. 325–328 (cit. on p. 15).
- Valstar, Michel, Brais Martinez, Xavier Binefa, and Maja Pantic (2010). “Facial point detection using boosted regression and graph models”. In: *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. IEEE, pp. 2729–2736 (cit. on p. 21).

- Van Gemert, Jan C, Cor J Veenman, Arnold WM Smeulders, and Jan-Mark Geusebroek (2010). “Visual word ambiguity”. In: *IEEE transactions on pattern analysis and machine intelligence* 32.7, pp. 1271–1283 (cit. on p. 46).
- Vapnik, Vladimir (2013). *The nature of statistical learning theory*. Springer Science & Business Media (cit. on p. 66).
- Velusamy, S., H. Kannan, B. Anand, A. Sharma, and B. Navathe (2011). “A method to infer emotions from facial Action Units”. In: *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 2028–2031 (cit. on p. 15).
- Viola, Paul, Michael Jones, et al. (2001). “Rapid object detection using a boosted cascade of simple features”. In: *CVPR (1)* 1, pp. 511–518 (cit. on p. 20).
- Vretos, N., N. Nikolaidis, and I. Pitas (2011). “3D facial expression recognition using Zernike moments on depth images”. In: *2011 18th IEEE International Conference on Image Processing*, pp. 773–776 (cit. on pp. 23, 165).
- Vukadinovic, Danijela and Maja Pantic (2005). “Fully automatic facial feature point detection using Gabor feature based boosted classifiers”. In: *2005 IEEE International Conference on Systems, Man and Cybernetics*. Vol. 2. IEEE, pp. 1692–1698 (cit. on p. 26).
- Walecki, Robert, Vladimir Pavlovic, Björn Schuller, Maja Pantic, et al. (2017). “Deep structured learning for facial action unit intensity estimation”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3405–3414 (cit. on p. 15).
- Wang, J., Z. Zhao, J. Liang, and C. Li (2018). “Video-Based Emotion Recognition using Face Frontalization and Deep Spatiotemporal Feature”. In: *2018 First Asian Conference on Affective Computing and Intelligent Interaction (ACII Asia)*, pp. 1–6 (cit. on p. 15).
- Wang, Jinjun, Jianchao Yang, Kai Yu, Fengjun Lv, Thomas Huang, and Yihong Gong (2010). “Locality-constrained linear coding for image classification”. In: *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*. IEEE, pp. 3360–3367 (cit. on pp. 55, 59).
- Wang, N., J. Han, and J. Fang (2012). “An Anomaly Detection Algorithm Based on Lossless Compression”. In: *Int Conf Netw Distr*, pp. 31–38 (cit. on p. 144).
- Wang, Nannan, Xinbo Gao, Dacheng Tao, Heng Yang, and Xuelong Li (2018). “Facial feature point detection: A comprehensive survey”. In: *Neurocomputing* 275, pp. 50–65 (cit. on p. 21).
- Wang, Sen, Yang Wang, Miao Jin, Xianfeng Gu, and Dimitris Samaras (2006). “3D surface matching and recognition using conformal geometry”. In: *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR’06)*. Vol. 2. IEEE, pp. 2453–2460 (cit. on p. 30).
- Wang, X., X. Liu, L. Lu, and Z. Shen (2014). “A New Facial Expression Recognition Method Based on Geometric Alignment and LBP Features”. In: *2014 IEEE 17th International Conference on Computational Science and Engineering*, pp. 1734–1737 (cit. on pp. 23, 166).
- Wang, Xiaoyang and Qiang Ji (2013). “A unified probabilistic approach modeling relationships between attributes and objects”. In: *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2120–2127 (cit. on p. 104).

- Wang, Y., Hui Yu, B. Stevens, and Honghai Liu (2015). “Dynamic facial expression recognition using local patch and LBP-TOP”. In: *2015 8th International Conference on Human System Interaction (HSI)*, pp. 362–367 (cit. on pp. 8, 15).
- Wang, Ziheng, Shangfei Wang, and Qiang Ji (2013). “Capturing Complex Spatio-temporal Relations among Facial Muscles for Facial Expression Recognition”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (cit. on pp. 12, 23, 167).
- Warren, Gemma, Elizabeth Schertler, and Peter Bull (2009). “Detecting deception from emotional and unemotional cues”. In: *Journal of Nonverbal Behavior* 33.1, pp. 59–69 (cit. on pp. 9, 10, 137).
- Weinland, Daniel, Mustafa Özuysal, and Pascal Fua (2010). “Making action recognition robust to occlusions and viewpoint changes”. In: *European Conference on Computer Vision*. Springer, pp. 635–648 (cit. on pp. 30, 31).
- Wiskott, Laurenz, Jean-Marc Fellous, Norbert Krüger, and Christoph Von Der Malsburg (1997). “Face recognition by elastic bunch graph matching”. In: *International Conference on Computer Analysis of Images and Patterns*. Springer, pp. 456–463 (cit. on p. 26).
- Wright, John, Allen Y Yang, Arvind Ganesh, S Shankar Sastry, and Yi Ma (2008). “Robust face recognition via sparse representation”. In: *IEEE transactions on pattern analysis and machine intelligence* 31.2, pp. 210–227 (cit. on pp. 54, 55, 61).
- Wu, Yue, Zuoguan Wang, and Qiang Ji (2013). “Facial Feature Tracking Under Varying Facial Expressions and Face Poses Based on Restricted Boltzmann Machines”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (cit. on pp. 23, 166).
- Xia, Zhaoqiang, Xiaoyi Feng, Jinye Peng, Xianlin Peng, and Guoying Zhao (2016). “Spontaneous micro-expression spotting via geometric deformation modeling”. In: *Comput Vis Image Und* 147, pp. 87–94 (cit. on pp. 138, 143).
- Xian, Y., Z. Akata, G. Sharma, Q. Nguyen, M. Hein, and B. Schiele (2016). “Latent Embeddings for Zero-Shot Classification”. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 69–77 (cit. on pp. 113, 114).
- Xie, Yi, Shuqiang Jiang, and Qingming Huang (2013). “Weighted visual vocabulary to balance the descriptive ability on general dataset”. In: *Neurocomputing* 119, pp. 478–488 (cit. on p. 43).
- Xingjian, SHI, Zhourong Chen, Hao Wang, Dit-Yan Yeung, Wai-Kin Wong, and Wang-chun Woo (2015). “Convolutional LSTM network: A machine learning approach for precipitation nowcasting”. In: *Adv Neur In*, pp. 802–810 (cit. on pp. 78, 149).
- Xu, Baohan, Yanwei Fu, Yu-Gang Jiang, Boyang Li, and Leonid Sigal (2018). “Heterogeneous knowledge transfer in video emotion recognition, attribution and summarization”. In: *IEEE Transactions on Affective Computing* 9.2, pp. 255–270 (cit. on p. 110).
- Xu, Feng, Junping Zhang, and James Z Wang (2017). “Microexpression identification and categorization using a facial dynamics map”. In: *IEEE T Affect Comput* 8.2, pp. 254–267 (cit. on p. 138).
- Y. Fu T. Hospedales, T. Xiang Z. Fu and S. Gong (2014). “Transductive multi-view embedding for zero-shot recognition and annotation”. In: *In ECCV*, pp. 584–599 (cit. on p. 122).
- Yan, Wen-Jing, Qi Wu, Yong-Jin Liu, Su-Jing Wang, and Xiaolan Fu (2013a). “CASME database: a dataset of spontaneous micro-expressions collected from neutralized faces”. In: *Proc Int Conf Autom Face Gesture Recognit*. IEEE, pp. 1–7 (cit. on pp. 140, 141, 157).

- Yan, Wen-Jing, Qi Wu, Jing Liang, Yu-Hsin Chen, and Xiaolan Fu (2013b). “How fast are the leaked facial expressions: The duration of micro-expressions”. In: *Journal of Nonverbal Behavior* 37.4, pp. 217–230 (cit. on p. 140).
- Yan, Wen-Jing, Xiaobai Li, Su-Jing Wang, Guoying Zhao, Yong-Jin Liu, Yu-Hsin Chen, and Xiaolan Fu (2014a). “CASME II: An improved spontaneous micro-expression database and the baseline evaluation”. In: *PloS one* 9.1, e86041 (cit. on pp. 140, 141, 157).
- Yan, Wen-Jing, Su-Jing Wang, Yu-Hsin Chen, Guoying Zhao, and Xiaolan Fu (2014b). “Quantifying micro-expressions with constraint local model and local binary pattern”. In: *Workshop at the ECCV*. Springer, pp. 296–305 (cit. on p. 142).
- Yang, B., J. Cao, R. Ni, and Y. Zhang (2018). “Facial Expression Recognition Using Weighted Mixture Deep Neural Network Based on Double-Channel Facial Images”. In: *IEEE Access* 6, pp. 4630–4640 (cit. on p. 101).
- Yang, Huan, Baoyuan Wang, Stephen Lin, David Wipf, Minyi Guo, and Baining Guo (2015). “Unsupervised extraction of video highlights via robust recurrent auto-encoders”. In: *preprint arXiv:1510.01442* (cit. on p. 144).
- Yang, Huiyuan, Umur Ciftci, and Lijun Yin (2018). “Facial Expression Recognition by De-Expression Residue Learning”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (cit. on pp. 23, 167).
- Yang, Jianchao, Kai Yu, Yihong Gong, and Tingwen Huang (2009a). “Linear spatial pyramid matching using sparse coding for image classification”. In: *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. IEEE, pp. 1794–1801 (cit. on p. 54).
- Yang, Jianchao, John Wright, Thomas S Huang, and Yi Ma (2010). “Image super-resolution via sparse representation”. In: *IEEE transactions on image processing* 19.11, pp. 2861–2873 (cit. on p. 54).
- Yang, Peng, Qingshan Liu, and Dimitris N Metaxas (2009b). “Boosting encoded dynamic features for facial expression recognition”. In: *Pattern Recognition Letters* 30.2, pp. 132–139 (cit. on p. 32).
- Yang, Yongxin and Timothy M Hospedales (2015). “A unified perspective on multi-domain and multi-task learning”. In: (cit. on pp. 113, 114).
- Yau, W., S. Shojaeilangari, K. Nandakumar, J. Li, and E. K. Teoh (2015). “Robust Representation and Recognition of Facial Emotions Using Extreme Sparse Learning”. In: *IEEE Transactions on Image Processing* 24.7, pp. 2140–2152 (cit. on pp. 12, 23, 167).
- Yi, Jizheng, Xia Mao, Lijiang Chen, Yuli Xue, and Angelo Compere (2014). “Facial expression recognition considering individual differences in facial structure and texture”. In: *IET Computer Vision* 8.5, pp. 429–440 (cit. on p. 8).
- Yosinski, Jason and Hod Lipson (2012). “Visually debugging restricted boltzmann machine training with a 3d example”. In: *In Representation Learning Workshop, 29th International Conference on Machine Learning*, p. 6 (cit. on p. 86).
- Zafeiriou, S. and M. Petrou (2009). “Nonlinear Nonnegative Component Analysis”. In: *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2860–2865 (cit. on pp. 11, 23, 165).
- Zafeiriou, Stefanos and Maria Petrou (2010). “Sparse representations for facial expressions recognition via  $l_1$  optimization”. In: *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition-Workshops*. IEEE, pp. 32–39 (cit. on pp. 27, 28).

- Zeng Nianyin, Hong Zhang Baoye Song Weibo Liu Yurong Li and Abdullah M. Dobaie. (2018). “Facial expression recognition via learning deep sparse autoencoders”. In: *Neurocomputing*, pp. 643–649 (cit. on pp. 23, 167).
- Zhang, Cha and Zhengyou Zhang (2010). “A survey of recent advances in face detection”. In: (cit. on p. 20).
- Zhang, Feifei, Tianzhu Zhang, Qirong Mao, and Changsheng Xu (2018). “Joint Pose and Expression Modeling for Facial Expression Recognition”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (cit. on pp. 23, 167).
- Zhang, L. and D. Tjondronegoro (2011). “Facial Expression Recognition Using Facial Movement Features”. In: *IEEE Transactions on Affective Computing* 2.4, pp. 219–229 (cit. on pp. 8, 21).
- Zhang, Li, Tao Xiang, and Shaogang Gong (2017). “Learning a deep embedding model for zero-shot learning”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2021–2030 (cit. on pp. 113, 114, 132).
- Zhang, S., S. Zhang, T. Huang, W. Gao, and Q. Tian (2018). “Learning Affective Features With a Hybrid Deep Model for Audio Visual Emotion Recognition”. In: *IEEE Transactions on Circuits and Systems for Video Technology* 28.10, pp. 3030–3043 (cit. on p. 110).
- Zhang, S., X. Pan, Y. Cui, X. Zhao, and L. Liu (2019). “Learning Affective Video Features for Facial Expression Recognition via Hybrid Deep Learning”. In: *IEEE Access* 7, pp. 32297–32304 (cit. on p. 33).
- Zhang, Shiliang, Qi Tian, Gang Hua, Qingming Huang, and Wen Gao (2011). “Generating descriptive visual words and visual phrases for large-scale image applications”. In: *IEEE Transactions on Image Processing* 20.9, pp. 2664–2677 (cit. on p. 43).
- Zhang, Xiao, Mohammad H Mahoor, and S Mohammad Mavadati (2015a). “Facial expression recognition using  $\{l\}$  -  $\{p\}$ -norm MKL multiclass-SVM”. In: *Machine Vision and Applications* 26.4, pp. 467–483 (cit. on p. 95).
- Zhang, Yongmian and Qiang Ji (2005). “Active and dynamic information fusion for facial expression understanding from image sequences”. In: *IEEE Transactions on pattern analysis and machine intelligence* 27.5, pp. 699–714 (cit. on p. 31).
- Zhang, Zheng, Yong Xu, Jian Yang, Xuelong Li, and David Zhang (2015b). “A survey of sparse representation: algorithms and applications”. In: *IEEE access* 3, pp. 490–530 (cit. on p. 59).
- Zhao, G. and M. Pietikainen (2007). “Dynamic Texture Recognition Using Local Binary Patterns with an Application to Facial Expressions”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 29.6, pp. 915–928 (cit. on pp. 14, 23, 166).
- Zhao, Guoying and Matti Pietikäinen (2009). “Boosted multi-resolution spatiotemporal descriptors for facial expression recognition”. In: *Pattern recognition letters* 30.12, pp. 1117–1127 (cit. on p. 32).
- Zhao, Kaili, Wen-Sheng Chu, Fernando De la Torre, Jeffrey F. Cohn, and Honggang Zhang (2015). “Joint Patch and Multi-Label Learning for Facial Action Unit Detection”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (cit. on pp. 23, 167).
- Zhao, Rui, Quan Gan, Shangfei Wang, and Qiang Ji (2016). “Facial Expression Intensity Estimation Using Ordinal Information”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (cit. on pp. 23, 167).

- Zhao Kaili, Wen-Sheng Chu and Honggang Zhang. (2016). “Deep region and multi-label learning for facial action unit detection.” In: *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3391–3399 (cit. on pp. 117, 118).
- Zhi, R., M. Flierl, Q. Ruan, and W. B. Kleijn (2011). “Graph-Preserving Sparse Nonnegative Matrix Factorization With Application to Facial Expression Recognition”. In: *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* 41.1, pp. 38–52 (cit. on pp. 8, 11, 23, 27, 165).
- Zhou Jianzhong, Jia Shan, Cao Peipei, Yang Yuankui, and Wang Xunheng (2005). “Modeling and application of multimodal affective user interface with multimedia computer sensing”. In: *Proceedings. 2005 First International Conference on Neural Interface and Control, 2005*. Pp. 28–31 (cit. on p. 3).
- Zhu, Qiqi, Yanfei Zhong, Bei Zhao, Gui-Song Xia, and Liangpei Zhang (2016). “Bag-of-visual-words scene classifier with local and global features for high spatial resolution remote sensing imagery”. In: *IEEE Geoscience and Remote Sensing Letters* 13.6, pp. 747–751 (cit. on pp. 41, 48).
- Zhu, R., Gaoli Sang, and Q. Zhao (2016). “Discriminative Feature Adaptation for cross-domain facial expression recognition”. In: *2016 International Conference on Biometrics (ICB)*, pp. 1–7 (cit. on p. 102).
- Zhu, Yunfeng, Fernando De la Torre, Jeffrey F Cohn, and Yu-Jin Zhang (2011). “Dynamic cascades with bidirectional bootstrapping for action unit detection in spontaneous facial behavior”. In: *IEEE transactions on affective computing* 2.2, pp. 79–91 (cit. on p. 24).

