



HAL
open science

Analysis of Micro-Expressions based on the Riesz Pyramid: Application to Spotting and Recognition

Carlos Arango Duque

► **To cite this version:**

Carlos Arango Duque. Analysis of Micro-Expressions based on the Riesz Pyramid: Application to Spotting and Recognition. Signal and Image Processing. Université de Lyon, 2018. English. NNT: 2018LYSES062 . tel-02335434

HAL Id: tel-02335434

<https://theses.hal.science/tel-02335434>

Submitted on 28 Oct 2019

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.



N°d'ordre NNT : 2018LYSES062

THESE de DOCTORAT DE L'UNIVERSITE DE LYON

opérée au sein de
Université Jean Monnet

Ecole Doctorale N°488
Sciences Ingénierie Santé

Spécialité : STIC - Image, Son, Signalement

Soutenue publiquement le 06/12/2018, par :
Carlos Arango Duque

Analysis of Micro-Expressions based on the Riesz Pyramid: Application to Spotting and Recognition

Devant le jury composé de :

Kpalma, Kidiyo	Professeur, INSA Rennes	Président
Yang-Song, Fan	Professeure, Université de Bourgogne	Rapporteuse
Séguier, Renaud	Professeur, Centrale Supélec	Rapporteur
Achard, Catherine	Maître de Conférences HDR, Sorbonne Université	Membre du jury
Emonet, Rémi	Maître de Conférences, Université Jean Monnet	Membre du jury
Legrand, Anne-Claire	Maître de Conférences, Université Jean Monnet	Membre du jury
Konik, Hubert	Maître de Conférences, Université Jean Monnet	Membre du jury
Alata, Olivier	Professeur, Université Jean Monnet	Directeur de thèse

Carlos Andrés Arango Duque

Analysis of Micro-Expressions based on the Riesz Pyramid: Application to Spotting and Recognition

Rapporteurs: Kidiyo Kpalma et Catherine Achard

Examineurs : Fan Yang-Song et Renaud Séguier

Superviseurs: Olivier Alata

Co-encadrants : Rémi Emonet, Anne-Claire Legrand et Hubert Konik

Université Jean Monnet

Laboratoire Hubert Curien

Image Science & Computer Vision

Spatio-temporal Data Analysis

18 Rue du Professeur Benoît Lauras

42023 Saint-Étienne

Acknowledgements

First and foremost, I want to thank my Thesis director Olivier Alata and co-directors, Rémi Emonet, Anne-Claire Legrand and Hubert Konik for their guidance and counseling during these three years and for teaching me to be a better researcher. I will also like to thank Dr. Pascal Giroux and Alexandre Bertholon for their work in the DAME project and all the staff of the rehabilitation center at CHU Saint Étienne that let us record them for the project.

Furthermore, I will also like to thank Thomas Gautrais for taking the time to patiently teach me to optimize my experiments using the lab's computer cluster. I want to also thank my colleagues and friends Brisbane, Dennis, Jose and Omar for countless coffee breaks, lunches, discussions and overall support during my thesis.

Finalmente me gustaría agradecer a mi familia por su constante apoyo y por otorgarme los medios para realizar mis sueños. Ustedes son mi mas grande inspiración y tesoro.

Résumé français

Introduction

La communication entre deux personnes implique des signaux verbaux (un message véhiculé dans la langue parlée) et des signaux non verbaux (langage corporel, tonus, expressions du visage) pour parvenir à un point de compréhension. Cependant, il existe certains contextes pour lesquels la plupart des signaux de communication verbaux et non verbaux ne sont pas disponibles. Dans ces situations, il est presque impossible d’engager une conversation ou d’essayer de traduire les émotions du visage ou le langage corporel d’une personne. Une analyse et une exploitation appropriée des micro-expressions pourraient venir palier ce manque de communication. Les micro-expressions sont des expressions faciales brèves et subtiles qui apparaissent et disparaissent en une fraction de seconde. Ce type d’expression faciale se produit généralement dans des situations à fort enjeu et est considéré comme reflétant “l’intention réelle” de l’être humain [64]. Cependant, détecter et reconnaître les micro-expressions est une tâche difficile pour l’homme. Il peut donc être pertinent de développer des systèmes d’aide à la communication exploitant les micro-expressions.

De nombreux travaux ont été réalisés dans les domaines de l’informatique affective et de la vision par ordinateur pour analyser les expressions faciales [19, 38, 69, 174, 198, 238]. Certains sont dédiés à l’analyse des micro-expressions mais dans une moindre mesure. De plus, une grande majorité de ces méthodes reposent essentiellement sur des méthodes de vision par ordinateur classiques telles que les motifs binaires locaux, les histogrammes de gradients orientés et le flux optique. Étant donné que ce domaine de recherche est relativement nouveau, des perspectives nouvelles restent à explorer.

Dans cette thèse, nous présentons une nouvelle méthodologie pour l’analyse des petits mouvements (que nous appellerons par la suite mouvements subtils) et des micro-expressions. Nous proposons d’utiliser la pyramide de Riesz, une approximation multi-échelle et directionnelle de la transformation de Riesz qui a été utilisée pour l’amplification du mouvement dans les vidéos à l’aide de l’estimation de la phase 2D locale, comme base de notre méthodologie. À partir de cette représentation, nous extrayons les variations de phase

de l'image afin de développer une méthode capable de détecter des mouvements subtils tels que les micro-expressions d'une séquence vidéo. Enfin, nous exploitons l'orientation dominante de la pyramide de Riesz pour formuler un nouveau descripteur pour classer les micro-expressions.

Chapitre 1 : Expressions Faciales

Les expressions faciales ou les macro-expressions sont des vecteurs puissants pour la communication humaine. Ils peuvent transmettre l'émotion, l'humeur, l'intention, la personnalité et enrichir la communication en général. Par définition, le moment où le premier mouvement de visage perceptible se produit est appelé *onset*, tandis que le moment où l'expression faciale disparaît (lorsque le visage revient à un état neutre) est appelé *offset*. Le moment où l'expression faciale atteint le pic d'intensité maximale s'appelle l'*apex* [233]. Certains chercheurs ont émis l'hypothèse que certaines expressions du visage sont universelles (spécifiquement pour sept émotions de base [58, 66]). Cependant, d'autres auteurs ont proposé que les humains sont trop complexes et que les expressions faciales ne représentent pas des émotions mais ce seraient plutôt des outils flexibles pour les interactions sociales.

Les micro-expressions, en revanche, sont des expressions faciales courtes et involontaires qui exposeraient les vraies émotions. Les deux principales caractéristiques des micro-expressions sont leur faible intensité spatiale et leur courte durée. Elles ont été initialement découvertes par Ekman et Friesen lorsqu'ils étudiaient des gestes qui pourraient être utilisés comme indices de déception [62]. Il existe deux principaux défis pour établir une base de référence formelle pour l'analyse de la micro-expression. Premièrement, les micro-expressions sont difficiles à identifier dans des catégories des émotions claires en raison de leur faible intensité spatiale. Deuxièmement, il n'y a pas de réel consensus au sein de la communauté scientifique sur la durée des micro-expressions [232].

Certaines tentatives ont été faites pour normaliser l'analyse des expressions faciales, comme le "Facial Action Coding System" ou FACS (un outil pour mesurer les expressions faciales en tant que combinaisons de mouvements faciaux localisés) [60]. Un autre moyen a été de créer des bases de données de micro-expressions pouvant être partagées par la communauté scientifique. Ces bases de données peuvent être divisées en 2 groupes : bases de données contenant des poses, dans lesquelles les personnes ont été enregistrées en imitant des micro-expressions et des bases de données contenant des mouvements spontanés, dans lesquelles des personnes ont été enregistrées pendant que leurs émotions sont déclenchées et réfrénées.

Finalement, pour proposer un cadre d'analyse approprié, nous devons prendre en compte les caractéristiques clés des micro-expressions, telles que leur nature spontanée ou leur courte

durée, et sélectionner les bases de données appropriées pour les tester (base de données spontanée enregistrée avec une caméra haute vitesse comme SMIC [113] et CASME II [230]).

Chapitre 2 : État de l'art

L'analyse des macro-expressions est un domaine d'intérêt pour la vision par ordinateur depuis de nombreuses années. L'analyse de micro-expressions, par contre, est un domaine de recherche relativement nouveau (les premiers travaux datent de 2009), où de multiples contributions restent à apporter. Considérant que les méthodes d'analyse des expressions micro et macro partagent des composants similaires dans la conception et la construction, ce chapitre se concentre sur la présentation des progrès de l'état de l'art dans les deux domaines.

Nous proposons une taxonomie pour la reconnaissance automatique des expressions faciales composée de 4 parties principales :

- Analyse du visage composée de deux étapes :
 - Localisation du visage : tout d'abord, les visages doivent être localisés dans l'image. Il existe deux approches principales de localisation de visage. Les approches de détection localisent l'emplacement d'un visage présent dans l'image à l'aide d'une boîte englobante tandis que les approches par segmentation "pixel" affectent une valeur binaire à chaque pixel.
 - Suivi des repères faciaux : ensuite, l'emplacement de traits spécifiques du visage ou de repères (par exemple, le coin des yeux, les sourcils, la bouche, le bout du nez, etc.) sont localisés. Ces points appelés "facial landmarks" deviennent très importants car ils sont utilisés pour spécifier des régions d'intérêt où les caractéristiques peuvent être extraites, pour la normalisation des visages et pour l'extraction de caractéristiques géométriques.
- Extraction de caractéristiques d'image : les caractéristiques sont extraites du visage avec des techniques dépendantes du type de données. Ces caractéristiques (ou descripteurs) peuvent exploiter l'apparence visuelle ou la géométrie des images (ou alors une combinaison des deux), être locales ou globales, ou encore être statiques ou dynamiques [38].

- Spotting : il s'agit du processus permet de détecter le moment où une expression faciale a lieu. Dans cette partie, les caractéristiques extraites à l'étape précédente sont analysées temporellement. Il existe deux types de méthodes de spotting :
 - Méthodes heuristiques : un ensemble de règles est conçu afin de calculer la relation temporelle des entités extraites entre les images et, enfin, de créer une méthode de recherche qui localise l'apex.
 - Méthodes par apprentissage : les micro-expressions sont repérées en formant un modèle qui détecte la micro-expression dans une séquence vidéo. Le spotting devient un problème de classification pour déterminer s'il s'est produit une micro-expression ou pas.
- Classification ou reconnaissance: Les caractéristiques extraites sont utilisées pour entraîner une méthode à classer des micro-expressions. Les méthodes d'apprentissage automatique peuvent être divisées en méthodes statiques ou dynamiques.

Finalement, nous résumons les méthodes et discutons des lignes directrices pour construire un système d'analyse de micro-expression approprié qui sera présenté chapitre 4 pour la détection et chapitre 5 pour la classification (ou la reconnaissance).

Chapitre 3 : Analyse de mouvements subtils à l'aide de la pyramide de Riesz

L'analyse du mouvement subtil permet d'aborder différentes applications et doit devenir un domaine à part entière. Par exemple, des schémas respiratoires anormaux peuvent être détectés en analysant les mouvements du corps pendant l'inspiration et l'expiration [210]. Bien que notre premier réflexe soit d'analyser ce type de mouvement à l'aide des outils classiques d'estimation de mouvement (méthodes directes et indirectes), certains défis pourraient nous amener à repenser cette approche. Premièrement, les méthodes classiques pourraient favoriser la mesure de mouvements plus importants et considérer les mouvements subtils comme un simple bruit. Deuxièmement, les méthodes classiques peuvent soit ajuster leurs paramètres pour détecter les mouvements subtils, mais devenir sensibles au bruit, soit devenir robustes au bruit mais ignorer les mouvements subtils.

Ces dernières années, certains auteurs ont proposé une méthode appelée amplification de mouvement dans laquelle les variations d'amplitude ou de phase de mouvements subtils sont amplifiées. Ces méthodes utilisent une représentation multi-échelle appelée représentation pyramidale dans laquelle une image est filtrée et sous-échantillonnée à plusieurs reprises (la pyramide générée est une séquence d'images dans laquelle la densité et la résolution

de l'échantillon sont réduites de manière régulière [6]). Bien que ces méthodes rendent le mouvement subtil plus évident pour l'œil humain, certaines mises en garde sont nécessaires. Si la méthode d'amplification est basée sur l'amplitude, elle peut également amplifier le bruit de l'image [222]. Par contre, [207] a proposé une méthode basée sur la phase qui n'amplifie pas le bruit. Cependant, comme elle exploite des pyramides orientables complexes, ces systèmes sont très complets et coûteux à construire. Une méthode plus récente, basée sur la pyramide de Riesz, a été proposée. Elle convient au traitement vidéo en temps réel exploitant les phases [209].

Néanmoins, le principal problème de ces techniques reste que ces méthodes exagèrent le mouvement plutôt que de l'estimer explicitement. Cependant, notre étude montre que les représentations intermédiaires produites par ces méthodes, en particulier leurs variations de phase, peuvent être utilisées comme des substituts du mouvement. Plus précisément, la représentation basée sur la pyramide de Riesz s'est révélée être une représentation simple, adaptable et à traitement rapide de mouvements subtils.

La pyramide de Riesz est une approximation multi-échelle du signal monogénique (qui est une extension 2D du signal analytique) obtenu à partir de la transformée de Riesz. La transformée de Riesz peut être considérée comme une transformation de Hilbert 2D directionnelle. Elle permet de calculer dans différentes sous-bandes une paire d'images en quadrature, c'est à dire déphasées de 90 degrés par rapport à l'orientation dominante en chaque pixel. Afin de rendre la pyramide de Riesz plus rapide, l'image est décomposée en sous-bandes non orientées utilisant une pyramide d'images inversible. Une pyramide laplacienne est utilisée au cours de notre étude. Elle s'obtient à l'aide de deux filtres de réponses impulsionnelles finies ce qui constitue une approximation efficace en termes calculatoires de la transformée de Riesz. Au niveau de chaque sous-bande, un triplet contenant l'image I et la paire d'images filtrées (R_1, R_2) constitue alors l'approximation du signal monogénique. (I, R_1, R_2) en chaque pixel peut être converti en coordonnées sphériques pour donner l'amplitude locale A , l'orientation locale θ et la phase locale ϕ en utilisant les équations

$$\begin{aligned} I &= A \cos(\phi) \\ R_1 &= A \sin(\phi) \cos(\theta) \\ R_2 &= A \sin(\phi) \sin(\theta) \end{aligned} \tag{0.1}$$

Le triplet de coefficients de la pyramide de Riesz $(I; R_1; R_2)$ peut aussi être représenté sous la forme d'un quaternion, la sous-bande d'origine I étant la partie réelle et les deux composants de l'approximation de la transformée de Riesz $(R_1; R_2)$ étant les composants imaginaires i et j du quaternion. Cependant, la solution pour l'équation précédente n'est

pas unique. (A, ϕ, θ) et $(A, -\phi, \theta + \pi)$ sont des solutions possibles. Cela peut être résolu en considérant

$$\phi \cos(\theta), \phi \sin(\theta) \quad (0.2)$$

C'est ce qu'on appelle la phase quaternionique et elle est invariante à l'indétermination expliquée précédemment. La séquence des phases quaternioniques peut être obtenue en "déroulant" d'abord les phases quaternioniques dans le temps (ce qui signifie obtenir des valeurs successives entre $-\pi$ et π), puis en utilisant un filtre linéaire invariant dans le temps. Cependant, étant donné que nous devons déterminer le moment exact où un mouvement subtil est détecté, nous ne pouvons utiliser les filtres causaux traditionnels susceptibles de retarder la réponse du signal. Par conséquent, nous proposons d'utiliser un filtre à réponse impulsionnelle finie (RIF) non-causal. En outre, le rapport signal sur bruit du signal de phase peut aussi être amélioré en atténuant spatialement le bruit dans chaque image avec un filtre spatial passe-bas gaussien.

Remarquons que, dans les régions de faible amplitude, la phase quaternionique est plus sensible au bruit dans l'image et que le signal de phase de mouvement lié au mouvement peut ne pas avoir du sens. Ainsi, nous proposons une méthode pour isoler les régions d'intérêt où un mouvement subtil pourrait avoir lieu en masquant les zones de bruit par un seuillage sur les amplitudes locales. Enfin, nous transformons la phase quaternionique en un signal 1-D, en calculant son énergie sur les zones sélectionnées image par image. Ce signal est alors utilisé pour l'analyse temporelle et fréquentielle de mouvements subtils.

Nous évaluons notre méthode de détection de mouvements subtils avec une base de données que nous avons réalisée (voir Annexe C) car nous n'avons pas trouvé de bases de données existantes appropriées. En raison de l'inexistence d'une base de données de mouvements subtils étiquetée et publique, nous avons dû tester nos expériences sur un ensemble de données plutôt limité. Après avoir testé notre méthode en faisant varier les niveaux de bruit gaussien et de bruit "poivre et sel", il a été possible de conclure que notre méthode dépasse les autres méthodes similaires mais exploitant le flot optique. Il faudrait compléter ces premiers résultats sur une base de données plus complète. Nous avons également illustré l'intérêt de notre méthode d'analyse de mouvements subtils en présentant brièvement quelques applications potentielles dans la vie réelle. De plus, les performances en termes de temps de calcul de notre méthode semblent indiquer qu'elle pourrait potentiellement être utilisée à l'avenir pour les applications en ligne.

Chapitre 4 : Détection de micro-expressions exploitant la pyramide de Riesz

Comme démontré dans le chapitre précédent, la représentation quaternionique de la phase et de l'orientation du signal monogénique de Riesz s'est révélée être un outil apte à l'analyse de mouvements subtils qui pourrait potentiellement être exploité pour la détection de micro-expressions. Ainsi, dans ce chapitre, nous proposons une méthode capable de repérer des micro-expressions dans une vidéo en analysant les variations de phase entre les images obtenues à partir de la pyramide de Riesz.

Nous utilisons le détecteur en cascade proposé par Viola et Jones [206] pour détecter la zone du visage dans l'image. Ensuite, nous divisons la zone du visage en zones d'intérêt spécifiques afin de limiter la recherche de repères faciaux d'intérêt (yeux, nez et bouche). Nous utilisons le modèle actif d'apparence proposé par [204] pour localiser un ensemble de points de repère. Nous suivons ces points au fil du temps en utilisant l'algorithme de Kanade-Lucas-Tomasi (KLT). Nous sélectionnons et définissons alors 5 régions d'intérêt : sourcil gauche, sourcil droit, œil gauche, œil droit et bouche.

Nous traitons la séquence d'images recadrées en utilisant la pyramide de Riesz, comme décrit dans le chapitre précédent. Nous obtenons alors à la fois une amplitude locale A et une phase quaternionique $(\phi \cos(\theta), \phi \sin(\theta))$ d'une sous-bande image donnée. Afin d'optimiser le processus de repérage, nous avons décidé de combiner la méthode de masquage d'amplitude du chapitre précédent avec un masque créé à partir des régions faciales d'intérêt dans lequel, selon le FACS, des micro-expressions pourraient apparaître.

L'étape suivante consisterait à calculer les variations de phase dans le temps et à détecter les mouvements subtils sous forme de micro-expressions. Cependant, suivant cette logique, les clignements des yeux et les changements du regard peuvent être considérés à tort comme des micro-expressions. Au lieu de simplement ignorer les informations fournies par les zones oculaires, nous pouvons les utiliser pour aider notre système à éliminer les éventuels faux positifs. Ainsi, nous concevons une méthode de détection (ou "spotting") heuristique de micro-expressions qui analyse le mouvement des régions d'intérêt locales et sépare les micro-expressions réelles des mouvements oculaires, réduisant ainsi la quantité de faux positifs possibles.

Nous évaluons notre méthode à l'aide de deux bases de données différentes (SMIC [113] et CASME II [230]). Les expériences ont montré que notre méthode de "spotting" surpasse les autres méthodes de l'état de l'art. De plus, les résultats de notre expérience d'analyse des paramètres ont montré que cette méthode est robuste aux changements de paramètres.

Chapitre 5 : Classification de micro-expressions

Jusqu'à maintenant, nous avons extrait les éléments du signal monogénique pour détecter lorsqu'un mouvement subtil ou une micro-expression a lieu. Cependant, le signal monogénique a également été utilisé dans la littérature pour la reconnaissance de micro-expressions [119, 148]. Ainsi, sur la base de nos travaux précédents, nous proposons un cadre qui utilise la pyramide de Riesz pour extraire des caractéristiques de phase orientées multi-échelles, ce qui nous permet de modéliser et de classer les micro-expressions.

Dans les chapitres précédents, nous avons utilisé la pyramide de Riesz pour extraire l'amplitude locale A et la phase quaternionique filtrée $(\phi \cos(\theta), \phi \sin(\theta))$ pour le "spotting" de mouvements subtils et les micro-expressions. Cependant, dans les deux cas, nous n'avons pas exploité l'orientation locale θ . Selon la formulation du signal monogénique, l'orientation locale θ représente la direction dominante dans l'image à un point donné. Cependant, il n'est pas possible de trouver une estimation absolue pour l'orientation locale et elle est également affectée par le problème d'ouverture. Néanmoins, en utilisant la représentation de phase quaternionique, nous avons pu représenter le mouvement dans différentes directions.

Après avoir analysé différentes caractéristiques des micro-expressions, nous proposons de modéliser l'évolution temporelle d'une micro-expression dans une seule image appariée appelée la paire d'images "Mean Oriented Riesz" ou MOR. Nous calculons simplement la phase quaternionique filtrée d'une séquence d'une micro-expression de son onset jusqu'à son apex, puis les images des moyennes temporelles sont calculées. L'intuition principale est qu'en effectuant une moyenne temporelle de la phase quaternionique filtrée, le mouvement réel de chaque pixel est modélisé dans une seule orientation et une seule amplitude, tout en réduisant l'effet du mouvement détecté de manière incorrecte compte-tenu du bruit.

Nous proposons un nouveau descripteur pondéré qui extrait des caractéristiques d'information d'orientation et de phase appelé "Mean Oriented Riesz Features" (MORF). La paire d'images MOR est divisée en une grille de blocs rectangulaires. Ensuite, un histogramme de la phase orientée est créé pour chaque bloc. Pour chaque pixel d'un bloc, un intervalle angulaire est sélectionné en fonction de l'orientation θ et un vote pondéré est exprimé en fonction de la valeur de la phase ϕ . L'histogramme résultant des caractéristiques est utilisé pour classer les différentes micro-expressions.

Une série d'expériences a montré que notre méthode de classification est capable de concurrencer d'autres méthodes bien connues et plus développées de l'état de l'art. De plus, les résultats de notre expérience d'analyse des paramètres ont montré que cette méthode est robuste aux changements de paramètres. Les résultats également peuvent être améliorés en combinant et en amplifiant simplement la phase orientée à partir de différents niveaux de la

pyramide de Riesz, ce qui suggère qu'un taux de reconnaissance plus élevé peut être atteint en effectuant des développements et des expérimentations supplémentaires.

Chapter 6 : Conclusion et Perspectives

Au cours de ce travail de thèse de Doctorat, de nouvelles méthodes pour l'analyse du mouvement subtile et de la micro-expression ont été proposées. Les principales contributions exploitant toutes la pyramide de Riesz sont :

- Une méthode pour l'analyse des mouvements subtils.
- Une méthode pour la détection (ou spotting) de micro-expressions.
- Une méthode pour la classification de micro-expressions.

Quelques perspectives intéressantes peuvent être envisagées à partir des travaux exposés dans cette thèse de Doctorat :

- Un meilleur système de suivi du visage devrait être intégré à notre cadre afin de travailler dans des environnements moins contraints comme c'est le cas pour le projet DAME (voir Annexe A).
- Notre méthode de micro-expression devrait être testée dans une base de données de vidéos plus longues (comme CAS(ME)² [168]).
- Utiliser les résultats de la méthode de détection de micro-expression pour améliorer la méthode de reconnaissance de la micro-expression.
- Créer une nouvelle technique d'estimation de mouvements à l'aide de la pyramide de Riesz.
- D'autres alternatives à la pyramide de Riesz devraient également être envisagées pour l'analyse des mouvements subtils.

Contents

Introduction	3
1 Facial Expressions	5
1.1 Macro-Expressions	5
1.2 Micro-Expressions	7
1.3 Facial Action Coding System	9
1.4 Micro-expressions Datasets	11
1.4.1 Posed Datasets	11
1.4.2 Spontaneous Datasets	12
1.4.3 DAME Dataset	14
1.5 Challenges and Considerations	14
1.6 Chapter Conclusions	15
2 State of the Art	17
2.1 A Taxonomy for AFER	17
2.2 Face Registration	18
2.2.1 Face Localization	18
2.2.2 Facial Landmarks Tracking	19
2.2.3 Regions of Interest	21
2.3 Feature Extraction Methods	23
2.3.1 Geometrical Features	23
2.3.2 Appearance Features	24
2.3.3 Hybrid Features	27
2.4 Facial Expression Spotting	27
2.4.1 Heuristic methods	28
2.4.2 Trained Methods	29
2.5 Classification Methods	31
2.5.1 Static Models	31
2.5.2 Dynamic Models	32
2.5.3 Deep Learning	32
2.6 Multimodal Fusion	33
2.7 State of the art summary	35
2.8 Discussion	36
2.9 Chapter Conclusions	37

3	Subtle Motion Analysis using the Riesz Pyramid	41
3.1	Background	42
3.1.1	Motion Estimation	42
3.1.2	Multi-scale Representation	47
3.1.3	Motion Amplification	48
3.1.4	Linear Video Magnification	49
3.1.5	Phase-based Magnification	50
3.2	Introduction to the Riesz Pyramid	51
3.2.1	The Analytical Signal	52
3.2.2	Local Amplitude and Local Phase	52
3.2.3	The Monogenic Signal and the Riesz transform	53
3.2.4	Implementing the Riesz Pyramid	54
3.3	Riesz Pyramid Motion Magnification	55
3.3.1	Riesz Pyramid coefficients	55
3.3.2	Quaternion representation of the Riesz Pyramid	56
3.3.3	Filtering of Quaternionic Phase	56
3.3.4	Amplification	58
3.4	Subtle Motion Analysis	58
3.4.1	Temporal Filtering Considerations	59
3.4.2	Amplitude Masking	61
3.4.3	Motion Spotting	62
3.5	Results	63
3.5.1	Preliminary Evaluation	63
3.5.2	Spotting Experiment	64
3.5.3	Running time comparison	69
3.6	Chapter Conclusions	70
4	Micro-Expression Spotting using the Riesz Pyramid	73
4.1	Face Registration	74
4.1.1	Face Detection	74
4.1.2	Facial Landmarks Fitting	75
4.1.3	Face Tracking	75
4.1.4	Regions of Interest	77
4.2	Riesz Transform and Filtering	77
4.2.1	Masking regions of interest	78
4.3	Micro-Expression Spotting	78
4.4	Experimental Results and Discussions	83
4.4.1	Evaluation Procedure	83
4.4.2	Results	84
4.4.3	Parameter Analysis	86
4.4.4	Discussion	89
4.5	Chapter Conclusions	92

5	Micro-Expression Classification	93
5.1	Oriented Phase Motion Representation	94
5.1.1	Local Orientation of the monogenic signal	94
5.1.2	Mean Oriented Riesz Image Pair	96
5.2	Mean Oriented Riesz Features	98
5.2.1	Implementation Details	98
5.2.2	Modifications on MORF	100
5.3	Experimental Results and Discussions	102
5.3.1	Experiment 1: Partial Macro-Expressions classification	103
5.3.2	Experiment 2: MORF vs MOOF for ME classification	108
5.3.3	Experiment 3: MORF variations	110
5.3.4	Discussion	115
5.4	Chapter Conclusions	117
6	Conclusions and Perspectives	119
	Bibliography	123
A	Project DAME	145
A.1	Introduction	145
A.2	Micro Expression Elicitation Experiment	147
A.2.1	Participants	147
A.2.2	List of Materials	147
A.2.3	Stimulation paradigm design	148
A.3	Video Processing	150
A.4	Micro-Expression Analysis	151
A.5	Preliminary Results	152
A.6	Discussion and Future Perspectives	152
B	Quaternions	155
B.1	Complex Exponential and Logarithms	156
C	Subtle Motion Dataset	157
D	Support Vector Machine	161
D.1	Classical Formulation	161
D.1.1	Hard-margin	161
D.1.2	Soft-Margin	162
D.2	Non-linear Extension	163
D.2.1	Dual Problem	163
D.2.2	Kernel trick	164
D.3	Hyperparameter Tuning	165
E	Additional Classification Experiments	167

E.1	Lucas-Kanade vs. TV-L1 optical flow	167
E.2	TV-L1 Optical Flow Parameter Evaluation	168
E.3	Amplitude based MORF	168
E.4	Masked Eyes MORF	170
List of Figures		173
List of Tables		177
List of publications		179

Introduction

Communication between two people involves verbal (a prompt that is conveyed in spoken language) and nonverbal cues (body language, tone, facial expressions) to reach a point of sharing understanding. When communicating nonverbally with others, we often use subtle signals, which are part of a larger communication process [45]. Furthermore, facial expressions might convey emotions about what is being communicated. A simple smile can indicate our approval of a message, while a scowl might signal displeasure or disagreement. Thus, recognizing emotions in facial expressions becomes vital for improving human information exchange. However, there are certain contexts in which most verbal and nonverbal communication cues won't be available. In these situations, it would be nearly impossible to either engage in a conversation or try to read the facial emotions or body language of a given person. For example, it is very complex for a doctor to communicate with autistic patients in order to assess their affective state [33], or do a correct pain assessment for unconscious and semi-conscious patients [9]. Hence, new communication cues have to be researched and developed in order to create communication channels between doctors and patients.

Fortunately, there is a possible solution brought from study of emotions and facial expressions called micro-expressions. Micro-expressions are brief and subtle facial expressions that go on and off the face in a fraction of a second. This kind of facial expression usually occurs in high stake situations, where people have something valuable to gain or lose and is considered to reflect a human's real intent [64].

However, detecting and recognizing micro-expressions is a challenging task for humans. There are some commercial tools and training methods [139, 155] which had proven to improve the ability to recognize micro-expressions [132]. However some studies seem to suggest that only people with good communications abilities really benefit from this kind of training [67]. Other studies suggest that people who have more problems focusing on a task, such as schizophrenic patients, will have more problems detecting micro-expressions [173]. Even more, some authors have suggested that some micro-expressions can successfully be concealed from a human observer if they are followed by some mouth movements like a smile [96]. Thus, systems that are able to assist people in this task are greatly needed.

There has been a great lot of work regarding facial expression analysis in affective computing and computer vision [19, 38, 69, 174, 198, 238]. There has been also some interest works in micro-expression analysis, albeit not as many. Furthermore, a great majority of these methods are based at their core on classically established computer vision methods such as local binary patterns, histogram of gradients and optical flow. Considering the fact that this area of research is relatively new, much contributions remains to be made. What's more, since a majority of the current works are focused on recognizing micro-expressions rather than detecting them, proposals that take into account both of these aspects might carry greater impact on this field of research.

In this thesis, we present a novel methodology for subtle motion and micro-expression analysis. We propose to use the Riesz pyramid, a multi-scale steerable Hilbert transformer which has been used for 2-D phase representation and video amplification, as the basis for our methodology. From this representation, we extract the image phase variations and use them to develop a method that is able to spot subtle motions such as micro-expressions from a video sequence. Furthermore, we exploit the dominant orientation from the Riesz transform to formulate a new descriptor for micro-expression classification.

The main structure of the document is outlined as follows:

- **Chapter 1** introduces the background and the problem at hand. We start with an explanation about what are facial expressions and most precisely what are micro-expressions. We also talk about the facial action coding system, the public available micro-expressions datasets and discuss the different challenges and considerations to take into account before developing a micro-expression analysis system.
- **Chapter 2** is a state of the art review in automatic facial expression analysis systems. In this chapter, we introduce a taxonomy to frame the different steps of a facial expression system: face localization, feature extraction, facial expression spotting and classification. We explore the different methods and considerations for each step. Finally we summarize the methods and discuss the guidelines to construct a proper micro-expression analysis system.
- **Chapter 3** serves as an introduction to the Riesz pyramid and its applicability for subtle motion analysis. After a brief introduction to motion estimation, pyramidal representation and video magnification methods, the reader is introduced to the concepts of Riesz transform, the monogenic signal and the Riesz pyramid. Then, the formulation for Riesz pyramid motion amplification is presented. Afterward, we present our method for subtle motion analysis based on the Riesz pyramid representation including some experiments in image sequences with subtle motions.

- **Chapter 4** presents our proposed micro-expression spotting method. An implementation for face detection and facial landmark localization is presented. We describe our method to isolate the areas of potential subtle motion using the facial landmarks and the local amplitude extracted from the monogenic signal. Then, we describe our heuristic algorithm for subtle facial motion spotting that is able to separate real micro-expressions from subtle eye movements and blinks. Finally, we present our experiments, discussion and conclusion.
- **Chapter 5** presents our proposed micro-expression classification method. After analysing the orientation component of the Riesz pyramid, we propose a way to model a micro-expression motion sequence as an oriented phase image pair. Then, we show the implementation of this model as a set of novel feature descriptors for micro-expression recognition. We propose to evaluate our proposed method by doing three sets of classification experiments. In these experiments, we test the effect of spatial intensity for facial expression classification, compare our oriented phase image pair model with one constructed using optical flow, and compare our methodology with the state of the art.
- **Chapter 6** presents the conclusions of the thesis, perspectives and future work.

Facial Expressions

Facial expressions in general are powerful tools for human communication. They can convey emotion, mood, intention, personality and enrich communication in general. However, before we start talking about our proposed method, we need to set some basic notions about the task at hand. Some questions that need to be asked are: what are facial expressions? Are they universal or cultural dependant? Do facial expressions convey true emotions? What is the difference between macro and micro-expressions? How fast are micro-expressions? Is there any standard for measuring facial expressions? is there any available standard database that we can use to compare our work with the state of the art?

In this chapter, we intend to answer these questions. This chapter is organized as follows: Sec. 1.1 introduces the concept of facial macro-expression and its relation with emotion; Sec. 1.2 introduces the concept of micro-expressions and its characteristics; Sec. 1.3 describes the Facial Action Coding System and how it divides and categorizes facial movements; Sec.1.4 presents the current available micro-expression datasets; Sec. 1.5 highlights the challenges and considerations to take into account before developing a micro-expression analysis system; Finally, Sec. 1.6 presents the chapter's conclusions.

1.1 Macro-Expressions

Facial expression (FE), from a non-verbal communication point of view, is the observable result of some facial movements. From a temporal point of view, the instant when the first noticeable face movement takes place is called onset, while the moment when the facial expression disappears (when the face returns to a neutral state) is called offset. The moment when the facial expression reaches the peak of highest intensity is called apex [233]. Facial expressions which last more than half of a second (from onset to offset) and that are easy to detect by the naked eye are called **macro-expressions**. Most of the work of facial expression analysis is focused on these types of expressions¹.

For many years it was firmly believed that facial expressions, much like other aspects of verbal (language, colloquial expressions) or non-verbal communication (hand and body gestures) were heavily influenced by the communicator's culture, geography and even personal

¹Macro-expression is a term that appeared at the same time as micro-expression in order to differentiate one from each other. It is very common to use "facial expressions" when referring to "macro-expressions"



Figure 1.1.: 7 basic emotions [54]

experience. However, Charles Darwin was one of the first researchers who suggested that some facial expressions are actually universal, that is, are expressed by everybody regardless of culture. In his book, “The expression of emotions in man and animals”, he states: “The young and the old of widely different races, both with man and animals, express the same state of mind by the same movements” [44]. This claims were brought back again by Silvan Tomkins, who suggested that emotion was the basis of human motivation and that the seat of emotion was in the face [66]. Later, Paul Ekman performed different cross-cultural studies on the recognition of facial expressions of emotion in literate and preliterate cultures which provided evidence to the hypothesis of the universality of facial expressions [61]. Furthermore, Ekman [58] states that there is strong evidence for 7 basic emotions: happiness, anger, disgust, contempt, sadness, fear and surprise (Fig. 1.1). This categorical description of emotions is very widely used in affective computing due to its simplicity and its claim of universality [38].

However, humans are complex creatures who feel and emote different types of emotions in different scenarios. For example, the expression of someone who just received an unexpected birthday gift (happily surprised) would be completely different from one who just received terrible news (sadly surprised). Some authors have proposed that there are actually 21 types of emotions [52]. A popular approach is to place the emotions into a space having a limited set of dimensions. These dimensions include valence (how pleasant or unpleasant a feeling is), arousal or activation (how likely is the person to take action under the emotional state) and control (the sense of control over the emotion) [38] (an example can be seen in Fig. 1.2). The problem with these models is that it is more challenging to link these emotions descriptions to a facial expression.

Although, the universality theory of facial expression is widely accepted by many researchers there are some arguments against this theory. Some authors have proposed that facial expressions are not fixed, semantic read-outs of internal states such as emotions or intentions, but flexible tools for social influence. According to the behavioral ecology view of facial displays, our facial expressions do not represent emotions but rather they are “social tools” that are used as lead signs of contingent action in social negotiation [40]. Even Ekman admits that, not only can there be emotion without expression, there can be what appears to

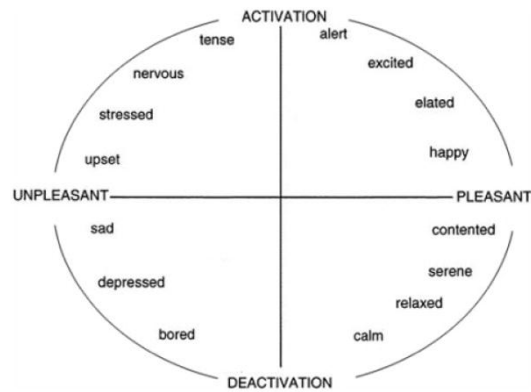


Figure 1.2.: A graphical representation of the circumplex model of affect with the horizontal axis representing the valence dimension and the vertical axis representing the arousal or activation dimension. [167]

be expression without emotion [56]. Thus, we should look spontaneous facial expressions as the representations of true emotion.

1.2 Micro-Expressions

Micro-expressions (we will refer them from now on as ME) are facial expressions which have a shorter duration than macro-expressions. They are involuntary and expose a person's true emotions. They can happen as a result of conscious suppression or unconscious repression [155]. The two main characteristics of MEs are their low spatial intensity and short duration.

They were initially discovered by Ekman and Friesen when they were studying “leaked” gestures which could be used as deception clues [62]. They were examining a filmed interview with a psychiatric patient, who was concealing her plan to commit suicide. In the film, taken after she had been in the hospital for a few weeks, the patient tells the doctor she no longer feels depressed and asks for a weekend pass to spend time at home with her family. She later confesses that she had been lying so that she would be able to kill herself when freed from the hospital's supervision. Using slow-motion replay they were able to see a complete sadness facial expression, but it was there only for an instant, quickly followed by a smiling appearance [57].

Although, MEs have been used as a way to detect concealed emotion, they have also been promoted as tools for lie detection. However there are two problems with the latter assumption. Firstly, not every individual who is concealing an emotion will show a ME so their absence is not evidence of truth. Secondly, even when someone shows a micro-expression, that is not sufficient to be certain the person is lying [57].

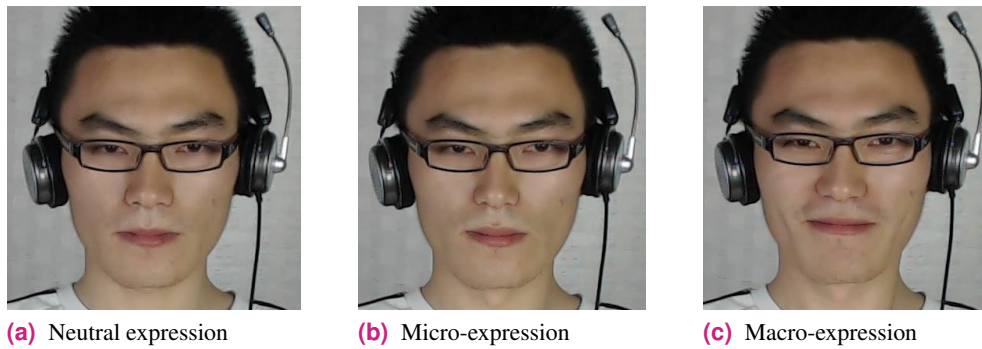


Figure 1.3.: Intensity of facial expressions [168]

One problem that comes from ME analysis is the difficulty of emotion labelling. Unlike macro-expressions, where facial expressions are more defined, MEs have a low spatial intensity which makes the process of inferring their corresponding emotion harder. Fig. 1.3 shows an example of the different spatial intensities from a neutral facial expression, the apex of a micro-expression and the apex of a macro-expression of a smile. As we can see, it is easier to deduce the type of emotion from the macro-expression image compared to the ME image. As consequence, different authors propose to divide MEs into classes that comprise more than one emotion (for instance: Positive/Negative/Surprise). This will be further evidenced in Sec. 1.4, in which different databases propose different divisions.

Even though the duration of a facial expression is considered one of the main features that separates micro-expressions from macro-expressions, there is not a real consensus in the scientific community about the duration of micro-expressions. Different authors have made different estimates about it:

- “A micro-expression flashes on and off the face in less than one-quarter of a second” [59]
- “Micro-expressions (that last 1/3 sec. or less)” [66]
- “Micro-expressions are extremely quick facial expressions of emotion that appear on the face for less than half a second” [75, 133]
- “Perhaps expressions last well under a second-perhaps 1/5 to 1/25 of a second” [63, 165]
- “The duration of a micro-expression is from 1/3 to 1/25 seconds” [163, 164]
- “Participants were shown images for durations in the range of microexpressions (15 ms and 30 ms)” [33]

	Duration	Number of Frames				
		25 fps	30 fps	60 fps	100 fps	200 fps
Lower Limit ME	170 ms	4.25	5.1	10.2	17	34
Upper Limit ME	500 ms	12.5	15	30	50	100
Lower Limit Onset Phase	65 ms	1.625	1.95	3.9	6.5	13
Upper Limit Onset Phase	260 ms	6.5	7.8	15.6	26	52

Table 1.1.: Estimated micro-expression duration under different frame rates

However, none of the previously mentioned authors provided any evidence to validate their estimates. Authors Yan et al. [232] attempt to define the duration of micro-expressions by collecting, coding and analysing the duration between the onset and offset frames of image sequences with fast leaked expressions. The distribution of the duration of expressions was described and then the duration boundaries of the micro-expression are estimated using a fitting curve of the distribution. Their studies showed that ME duration goes from 170 to 500 ms (Table. 1.1). Furthermore, they also noted that some MEs have fast onset phase but very slow offset phase (some even remain in apex for seconds, thus much longer than 500 ms). Thus, they calculated the onset duration in order to see whether it might be a good indicator for ME. Their studies showed that ME onset phase duration goes from 65 to 260 ms. The duration of micro-expressions becomes critical in the image acquisition step. A conventional camera (RGB or thermal imaging) can capture video between 25 to 30 fps which might not be sufficient for capturing fully detailed ME image sequences (for example a 25 fps camera would only be able to capture between 1 to 6 or 7 frames for the onset phase). On the other hand a high speed camera can capture from 100 to 500 fps with the caveat that they are pricey and the captured data will inevitably become computationally more expensive to process. Thus it becomes critical to select adequate capturing devices for the image acquisition step or select a method that is able to interpolate or synthesize missing data [138, 241].

1.3 Facial Action Coding System

Facial expressions (either macro or micro) involve different small motions and position changes of the muscles beneath the face skin. Thus, some authors have proposed to study these motions separately and compose facial expression as a combination of these motions. In 1978, Paul Ekman and Friesen published a tool for measuring these facial motions called the “Facial Action Coding System”.

The Facial Action Coding System (FACS) is an anatomically based system for measuring all visually discernible facial movement [60]. It describes all visually distinguishable facial

Upper Face Action Units					
AU 1	AU 2	AU 4	AU 5	AU 6	AU 7
					
Inner Brow Raiser *AU 41	Outer Brow Raiser *AU 42	Brow Lowerer *AU 43	Upper Lid Raiser AU 44	Cheek Raiser AU 45	Lid Tightener AU 46
					
Lid Droop	Slit	Eyes Closed	Squint	Blink	Wink
Lower Face Action Units					
AU 9	AU 10	AU 11	AU 12	AU 13	AU 14
					
Nose Wrinkler AU 15	Upper Lip Raiser AU 16	Nasolabial Deepener AU 17	Lip Corner Puller AU 18	Cheek Puffer AU 20	Dimpler AU 22
					
Lip Corner Depressor AU 23	Lower Lip Depressor AU 24	Chin Raiser *AU 25	Lip Puckerer *AU 26	Lip Stretcher *AU 27	Lip Funneler AU 28
					
Lip Tightener	Lip Pressor	Lips Part	Jaw Drop	Mouth Stretch	Lip Suck

Figure 1.4.: Facial Action Units (AUs) of upper and lower face [60]

activity on the basis of 46 unique action units (AUs), as well as several categories of head and eye positions and movements [66] (some examples can be seen in Fig. 1.4). Each AU has a numeric code (the designation of which is fairly arbitrary) as well as the muscle groups involved in each action². Furthermore, FACS also codes the intensity of each facial action on a five point intensity scale (from A, the weakest trace of a movement, to E, the maximum possible intensity). Also, there is a code when action occurs to right side (R) or the left side (L) of the face.

Facial expressions can be coded as a combination of AUs. For instance, a genuine smile can be coded as a combination of AU6 (cheek raiser) and AU12 (lip corner puller). However, this type of coding should not be taking at heart since different people might show different levels of facial expressivity. For example, patients of Parkinson’s disease can exhibit a reduction of spontaneous facial expression, designated as “facial masking”, a symptom in which facial muscles become rigid, therefore their expressions are less intense. Furthermore, when working with micro-expressions where facial movements are so subtle, certain AUs will appear depending on people’s face expressivity (whereas a macro-expression of anger could be easily coded as AU4+AU5+AU7+AU23, an ME of anger can be coded as AU4+AU7 or just as AU4 depending of the subject [46]). Thus, it would be unwise to use AUs as a definitive code for emotions but rather as an indication of emotion.

²For clarification, FACS does not provide any information of the degree of muscle activation. This is due to the fact that a given muscle may act in different ways to produce visibly different actions

1.4 Micro-expressions Datasets

One of the greatest challenges of ME analysis research is the lack of ME standard datasets. In contrast with the large quantity of publicly available FE datasets, there are only a few available for MEs. The MEs datasets can be divided into 2 groups: Posed and Spontaneous datasets.

1.4.1 Posed Datasets

These datasets were obtained by recording people who were asked to mimic MEs. These databases are designed under the assumption that an ME is a subtle macro-expression, thus it could be simulated by the subject:

- **USF-HD dataset** [184, 185]: This database consists of 47 sequences and contains 181 macro-expressions (smile, surprise, anger, sadness) and 100 micro-expressions. Videos were collected either by a JVC-HD100 or a Panasonic AG-HMC40 camcorder at a resolution of 1280×720 and frame-rate of 29.7 fps. The length of each video is on average approximately 1 minute in length. For micro-expressions, subjects were shown some example videos containing micro-expressions prior to being recorded. The subject was then asked to mimic them. The authors of this database don't disclose the number of subjects, neither their age, nor their gender, nor their ethnicity. This database is not publicly available.
- **Polikovsky** [163, 164]: This database consists of 42 ME samples (containing the 7 basic emotions). Videos were collected by a Point Grey Grasshopper camera at a resolution of 640×480 and frame-rate of 200 fps. 10 university students of different ethnicities were used as subjects³. The authors of this dataset don't disclose the gender of the participants. The participants were instructed to perform 7 basic emotions with low facial muscle intensity and to go back to the neutral face expression as fast as possible, simulating the ME motion. This database is not publicly available.
- **SFED2007** [153]: The Subtle Facial Expression Database 2007 has 20 subjects performing 4 facial expressions (neutral, subtle smile, subtle surprise and subtle angry). All the subjects are Asians (the authors of this dataset don't disclose neither the gender, nor the age of the participants). The image size is 640×480 . There isn't any information available about the devices or the procedure to capture the data. This database is available in the [intelligent media laboratory](#) web-page from the Pohang University Science and Technology (POSTECH).

³Although, the authors don't explicitly disclose the subject's age, one can broadly assume the age range to be between 18-30 years

Database	USF-HD	Polikowsky	SFED2007
Subjects	-	10	20
Samples	181 macro, 100 micro	42	80
Image Size	1280 × 720	640 × 480	640 × 480
Camera Speed	29.7 fps	200 fps	-
Age Range	-	University Students	-
Men/Women	-	-	-
Diversity	-	5 Asian, 4 Caucasian and 1 Indian	All Asian
Estimated Emotions	4 (smile, surprise, anger and sad)	7 (basic emotions)	4 (smile, surprise, anger and neutral)
AU labels	No	Yes	-

Table 1.2.: ME posed databases

1.4.2 Spontaneous Datasets

These datasets were obtained by recording people whose emotions are elicited. These databases are designed under the assumption that, because micro-expressions are involuntary, the participant needs to genuinely experience the emotion, thus, the facial expressions must be provoked.

- **SMIC** [113]: The Spontaneous Micro-expression database, consists of 3 datasets of ME samples containing 164 ME samples labelled as 3 types of emotion (positive, negative and surprise). Videos were collected by a PixelINK PL-B774U High speed (HS) camera at a resolution of 640 × 480 and a frame-rate of 100 fps, a normal visual camera (VIS) and a near-infrared (NIR) camera, both with 25 fps and resolution of 640 × 480. 16 valid subjects of different age, gender and ethnicity were used as participants. The subjects were instructed to watch some emotional videos while hiding their true emotions (under penalty of having to fill a long and boring questionnaire). This database is available in the Center for Machine Vision and Signal Analysis (**CMVS**) research center web-page from the University of Oulu.
- **CASME** [233]: The Chinese Academy of Sciences Micro-expression database, consists of 195 ME samples labelled as the 7 basic emotions. Videos were collected with 2 different cameras: A BenQ M31 camera at a resolution of 1280 × 720 and a frame-rate of 60 fps; and a Point Grey GRAS-03K2C camera at a resolution of 640 × 480 and a frame-rate of 60 fps. 19 valid subjects of different age and gender (all subjects have the same ethnicity) were used as participants. The subjects were instructed to watch some emotional videos while hiding their true emotions (under penalty of lowering their payment for participating in the experiment). This database is available in the **Prof Xiaolan Fu's group** web-page of the Chinese Academy of Sciences.

Database	SMIC	CASME	CASME II	CAS(ME) ²	SAMM
Subjects	16	19	26	22	30
Samples	164	195	247	250 macro, 53 micro	159
Image Size	640 × 480	1280 × 720 640 × 480	640 × 480	640 × 480	2040 × 1088
Camera Speed	HS 100 fps NIR 30 fps VIS	60 fps	200 fps	25 fps	200 fps
Age Range	22 – 34	$\mu = 22 \sigma = 1.6$	$\mu = 22 \sigma = 1.6$	$\mu = 22.59$ $\sigma = 2.2$	$\mu = 34.48$ $\sigma = 13.73$
Gender (M/F)	10/6	7/12	10/16	6/16	14/16
Diversity	8 Asian, 8 Caucasian	All Asian	All Asian	All Asian	18 Caucasian, 12 others
Emotions	3 (Pos, Neg and Sur)	7 (basic emotions)	5 (smile, disgust, surprise, repression and others)	4 (Pos, Neg, Sur and Others)	7 (basic emotions)
Au labels	No	Yes	Yes	Yes	Yes

Table 1.3.: ME spontaneous datasets

- **CASME II** [230]: It is an improved version of the CASME database. It consists of 247 ME samples labelled as 5 types of emotions (happiness, disgust, surprise, repression and others). Videos were collected by a Point Grey GRAS-03K2C camera at a resolution of 640 × 480 and a frame-rate of 200 fps. 26 valid subjects of different age and gender (all subjects have the same ethnicity) were used as participants. Some participants were asked to keep neutralized faces when watching video clips while other participants only tried to suppress the facial movements when they realized there was a facial expression. This database is available in the [Prof Xiaolan Fu's group](#) web-page of the Chinese Academy of Sciences.
- **CAS(ME)²** [168]: This database consists of long videos containing 250 macro-expression samples and 53 ME samples labelled as 4 types of emotions (positive, negative, surprise, and others). Videos were collected by a Logitech Pro C920 camera at a resolution of 640 × 480 and a frame-rate of 30 fps. 22 valid subjects of different ages and genders (all subjects have the same ethnicity) were recruited. The participants were asked to suppress their expressions to the best of their ability (they were informed that their monetary rewards would be reduced if they produced any noticeable expression). This database is available in the [Prof Xiaolan Fu's group](#) web-page of the Chinese Academy of Sciences.
- **SAMM** [46]: The Spontaneous Micro-Facial Movement dataset contains 159 ME samples labelled as the 7 basic emotions. Videos were collected by a LBasler Ace

acA2000-340km camera at a resolution of 2040×1088 and a frame-rate of 200 fps. The authors aimed to create a very diverse dataset, thus, 30 participants of different ages (from 19 to 57 years old), genders and ethnicities (13) were recruited. The participants were asked to suppress their expressions to the best of their ability (they were informed that their monetary rewards would be reduced if they produced any noticeable expression). This database can be downloaded from a link found in the paper [46] or by contacting the authors of it.

1.4.3 DAME Dataset

During the development of this thesis, we worked along the “Centre Hospitalier Universitaire” (CHU) de Saint-Etienne in a project called DAME (Detection of Awareness with Micro-Expression) to analyse micro-expressions and facial movements from patients with disorders of consciousness in an intensive care unit of a hospital. In this project, we aimed to create an ME elicitation protocol, a multi-modal video recording and stimuli delivery system, and an ME analysing and annotation software. One of the goals of this project is to create a spontaneous annotated ME database using healthy subjects and patients suffering of disorders of consciousness (DAME database). For a more detailed explanation about the DAME project and its initial results, we refer the reader to Sec. A.

1.5 Challenges and Considerations

We would like to highlight a series of challenges and considerations that need to be addressed before embarking in the development of an ME analysis system:

- **Spatial Subtlety:** MEs are very subtle and almost invisible motion that is difficult to identify by the naked eye. Thus, a good ME system needs to use a method which is sensitive enough to detect and recognize this type of motion.
- **ME Duration:** Considering that MEs has a short variable duration (as discussed in Sec. 1.2), a good ME system needs to be able to detect and analyse quick temporal events.
- **Frame rate:** Either if we are selecting a publicly available database or implementing an acquisition system, the captured data’s frame-rate needs to be carefully selected. A system with a small frame-rate might loose some important details, but an overly high frame-rate might end up providing redundant data which would be computationally expensive to process.

1.6 Chapter Conclusions

In this chapter, we were able to recognize the study of facial micro-expressions as a more reliable approach to detect human true emotion compared to classical facial macro-expressions. However, the low spatial amplitude and short duration of MEs makes them challenging to analyse. Nevertheless, the growing interest in researching and developing ME analysis systems can be reflected in the ever-growing quantity of methods proposed every year (as we will see in Sec.2) and in the emergence of different ME databases.

It is worth noting that the most recent and more used databases available consider key features of MEs such as their spontaneous nature or their short duration (by capturing them using high-speed cameras). In a similar fashion, we must also take into account the key features, concepts and challenges regarding MEs before embarking in the development of a micro-expression analysis system.

State of the Art

Macro expression analysis has been an area of interest in computer vision for many years. Micro expression analysis, on the other hand, is a relatively new field of research (the first works date the year 2009) where multiple contributions remain to be made. Regardless of that, there has been an interesting number of proposed approaches in the state of the art. Considering that both micro and macro expressions analysis methods share some similar components in conception and construction, this chapter will focus on present advances in the state of the art in these areas. This chapter is structured as follows: Sec. 2.1 introduces a taxonomy to classify the different steps and methods for facial expression analysis systems. Sec.2.2 presents face registration methods. Sec. 2.3 classifies and displays different methods for feature extraction. Sec. 2.4 shows current techniques for ME spotting. Sec. 2.5 explores different methods for classification. Sec. 2.6 presents AFER methods that integrate multimodal data. In Sec. 2.7 we make a summary on the state of the art of micro expressions analysis. In Sec. 2.8 we discuss the guidelines to construct a proper ME analysis framework. Finally, Sec. 2.9 presents chapter's conclusions.

2.1 A Taxonomy for AFER

Automatic facial emotion recognition (AFER) systems have been researched for quite sometime. However, there is still not a definitive taxonomy to classify all the methods. Different surveys have proposed their own structures or frameworks for AFER systems [38, 69, 193]. In Fig. 2.1 we propose a taxonomy for AFER (as well as a guideline for an AFER framework) which consists of 4 main parts:

- **Face Registration:** First the faces must be localized in the image and then, during registration, facial landmarks are localized (Sec. 2.2).
- **Feature Extraction:** Features are extracted from the face with techniques dependent on the type of data. The approaches are divided into pre-designed or learned, appearance based or geometric based (or a fusion of both), local or global and static or dynamic (Sec. 2.3).
- **Spotting:** Techniques to detect the moment when a facial expression takes place (Sec. 2.4).

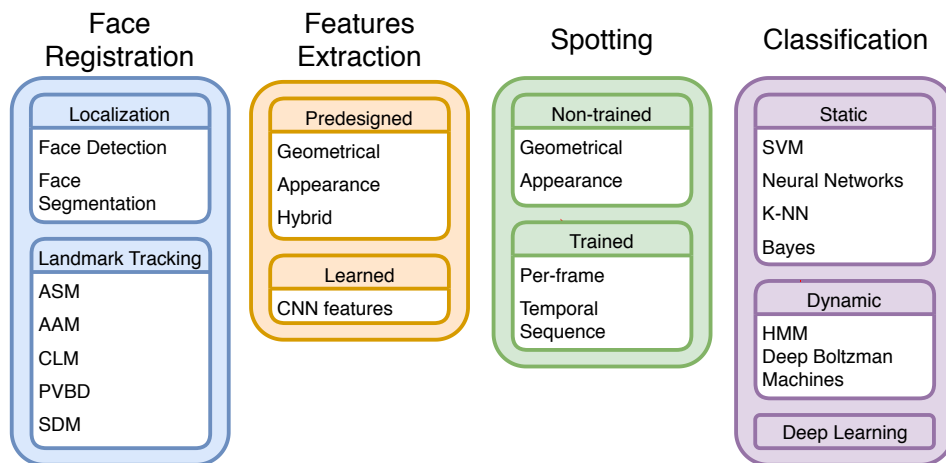


Figure 2.1.: Taxonomy for AFER in computer vision

- Classification: Machine learning techniques are used to discriminate between different facial expressions (Sec. 2.5).

2.2 Face Registration

2.2.1 Face Localization

Although there are plenty face localization methods in the computer vision community (specially in Biometry) the localization step is overlooked in many AFER papers as they focus more on the feature extraction and classification steps¹. Nonetheless this step is required in order to have a fully automatic FE analysis system. There are two main face localization approaches. Detection approaches locate the position of a face present in the image as a bounding box while segmentation approaches simply assigns a binary value to each pixel.

For RGB images the Viola&Jones [206] is still one of the most used algorithms for face detection [137, 161, 184]. It is based on a cascade of weak classifiers which can quickly detect frontal faces. Due to its speed and accuracy, this is the de facto face detection method when working with images with frontal faces captured in a controlled environment. However, it is weak to heavy occlusion and large pose variations. In recent years, some methods have been proposed which are able to detect faces in a wide range of orientations and poses using a Width-First-Search (WFS) tree [88] or using a single model based on deep convolutional neural networks (CNN) [68, 109] (see Fig. 2.2). Other authors have proposed to solve the problem of occlusion by multiview bootstrapping: an initial keypoint detector is used to produce noisy labels in multiple views of the face which are triangulated in 3D

¹While some papers either ignore or briefly comment on the face localization step in their framework others use databases in which their faces are already cropped

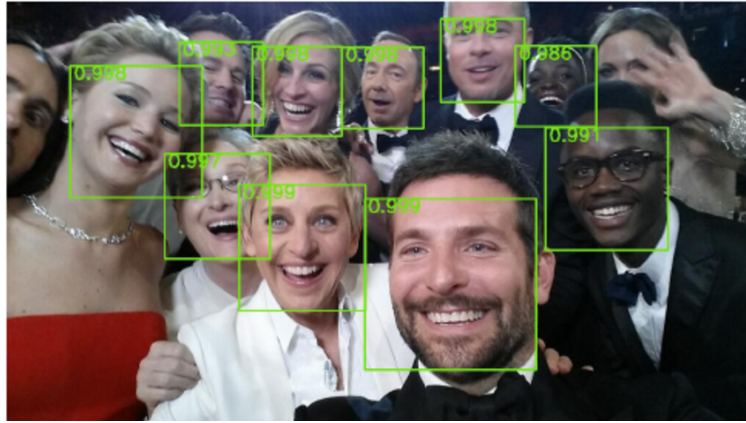


Figure 2.2.: An example of user generated photos on social networks that contains faces in various poses, illuminations and occlusions [68]

using multiview geometry. Then, the reprojected triangulations are used as new labeled training data to improve the detector [187]².

Face segmentation techniques normally exploit color and texture information (such as skin tone) and then refine it using some kind of correction [1, 28, 197].

Although in theory RGB techniques are applicable to thermal images there is a simpler alternative. In most cases the thermal signature of the face is distinct from that of the environment [103] and segmenting the image according to the radiant emittance of each pixel is enough [101, 203]. One alternative proposed by [220] is based on the head curve geometry in which the face is extracted after five points are located on the head boundary.

2.2.2 Facial Landmarks Tracking

Once the face is detected we need to track the location and movement of specific facial features (e.g corner of the eyes, eyebrows, mouth, tip of the nose etc.). When we are analysing facial expression we need to accurately describe the location, shape and deformation of these facial features. For RGB images, this is done by characterizing these areas with a series of points that surrounds them called facial feature points also known as fiducial points or facial landmarks. These points become very important for the feature extraction step since they are used to specify the regions of interest (ROI) where the features are extracted, for face normalization (in which faces are set to a specific size and rotation using different geometric transformations) and even their temporal displacement can be extracted as a dynamic feature.

There are some methods in which facial feature points are independently tracked without using prior knowledge about the face. As a result, they usually are susceptible to facial

²This method has been used to detect face and hand keypoints and are part of the [OpenPose](#) project

expression change, face poses change, occlusion etc. On the other hand, methods with shape prior model capture the dependence between facial feature points by explicitly modeling the general properties as well as the variations of facial shape or appearance [225]. We will focus on the latter type of methods.

Most facial feature tracking methods are based on the point distribution model (PDM) which represents the mean and variance of a shape based on a training set of similar shapes. The most classical method based on PDM is called active shape models (ASM) in which a linear generative model captures shape variations based on principal component analysis (PCA) [37]. Some ASM variations focus on modelling only shape variations using restricted boltzman machines [225] and using a hierarchical shape model [200]. However, ASM was later extended by also including the face appearance variation from a holistic perspective in active appearance models (AAM) [36, 134] which was also optimized for faster facial feature fitting [204] (see Fig. 2.3a). Another modelling scheme called constrained local model (CLM) describes the appearance variation of each facial feature point independently [39]. This was later optimized by simplifying the distribution of landmark locations obtained from each local detector using a non parametric representation [179]. A regression based approach of CLM called Discriminative response map fitting (DRMF) was proposed by [10] in which its response map can be represented by a small set of parameters. Supervised Descent Method (SDM) proposes to detect facial features by learning a sequence of descent directions which minimizes the mean of Non-linear Least Squares (NLS) functions at different points in the face [228]. The reader is referred to [212] for a more comprehensive review of other PDM tracking techniques.

An alternative to PDM is the 3D morphable face model which represents the 3D geometry of the individual facial expressions of a person but also constrains motion due to local deformations [55]. This type of model has been used to analyze patients during epileptic seizures [135] and for facial animation [23]. Some alternative approaches like the Piecewise Bézier Volume Deformation tracker (PBVD) don't model the whole face but instead takes a series of patches of the 3D morphable model which are embedded in Bézier volumes to guarantee the continuity and smoothness of the model [80, 195] (see Fig. 2.3b).

Although the tasks of face detection and landmark localization have traditionally been approached as separate problems, [242] proposes an unified approach in which they use a mixture of trees with a shared pool of parts to detect the face and landmarks from different head poses. [187]

Normally for thermal images it is difficult to track fiducial points so regions of interest are mostly selected manually based on the face morphology. However some authors have proposed some alternatives to automatically detect ROIs. For instance, [203] proposes to detect a series of interest points using Harry's operator and then cluster these points into ROIs using k-means. [85] proposes to uses a genetic algorithm (GA) in order to both

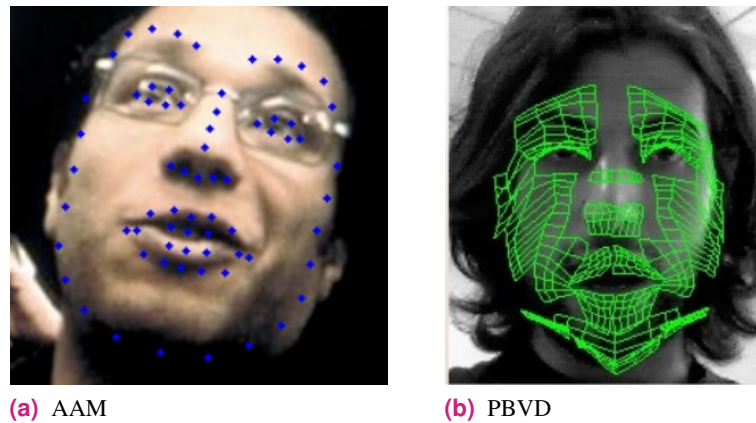


Figure 2.3.: Illustration of 2 faces registered using (a) an optimized active appearance model [204] and (b) Piecewise Bézier Volume Deformation tracker [11]

select the optimal ROIs and tune in the parameters of its feature extraction method. [143] automatically selected regions in which temperature increases or decreases significantly when human emotions change and called them t-ROIs.

2.2.3 Regions of Interest

The regions of interest (ROI) are regions of an image enclosed within boundaries from which we will extract some features. In our case, the most simple ROI would be defined by the location of the face and its features given by a face registration method. Although, for some holistic methods, this general ROI would suffice, most featured methods prefer to divide the face into regions in order to analyse them separately. In general, there are two ways to define the face ROIs:

- **Face grid:** Some authors propose to separate the face into a grid of equally divided rectangles [117, 231] (Fig. 2.4a). The main advantage of this approach is that, due to the rectangular nature of digital images, the ROIs are very easy to divide and the areas do not require additional alignment. Furthermore, they cover the whole face area. On the other hand, not all the given ROIs might provide relevant information (for instance, the lower corners of Fig. 2.4a) or provide redundant information (it is very likely that the right and left nostril areas might provide the same information). Furthermore, the size of the extracted data is normally higher compared to local ROIs. Normally a face grid will provide $m \times n$ ROIs which will be a higher quantity than s manually selected number of local ROIs ($m \times n \gg s$). This disadvantages can be mitigated by manually discarding certain ROIs.
- **Local ROIs:** Some authors propose to select and crop some specific areas of the face using facial landmarks as reference [22, 47] (Fig. 2.4b). The main intuition for

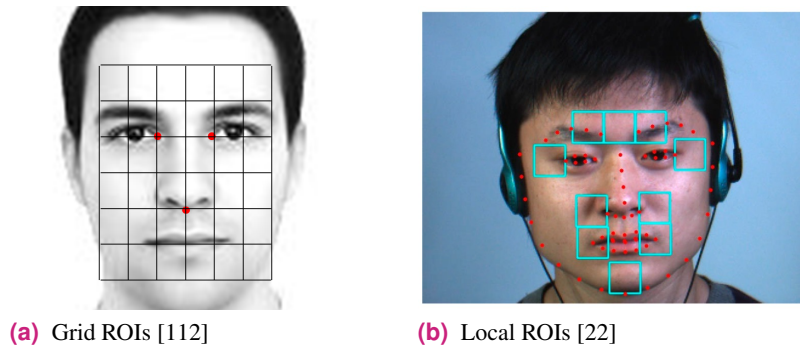


Figure 2.4.: Facial ROIs

this type of methods is that it is better to focus on certain face areas which might produce more pronounced movements than others when an ME takes place. The main advantages of this type of method is that it avoids extracting redundant or unnecessary data. Furthermore, the user can manipulate the location and size of the ROIs individually and adapt to each person's face. However, the main disadvantage is that the alignment ROIs are dependant of the facial landmark tracking method robustness (if the tracking is jittery or is failing it will directly affect the ROIs). What's more, the location of ROIs are limited to the facial landmarks (for instance, the cheeks are not normally considered a facial landmark, even though some motion could be potentially detected during MEs).

There are a couple of further considerations that must be taken when selecting ROIs. The first one is regarding the eye's ROIs. Considering that people spontaneously generates an average of 15-20 blinks per minute [141], this areas might become the source of false positives. However discarding the eye areas would potentially dismiss true emotional face movements. Certain movements that occur in the eye area during facial expressions (like AU5 upper lid raiser and AU7 lid tightener) might also occur during an ME. Thus, one must make a choice in whether selecting or dismissing the eye ROIs. The second consideration comes from everyday face occlusion. For instance, frown lines (wrinkles that appear in the forehead while frowning) might provide some interesting emotional information. Unfortunately this type of analysis cannot be made with people whose haircuts cover their forehead (like in Fig. 2.4b). Another source of everyday face occlusion are glasses. The glasses frame partially occlude the area between eyes and eyebrows and the nose bridge between eyes (where the AU9, nose wrinkler, takes place)³. Thus, ROIs must be selected for both glass users and non-users.

³Dismissing faces with glasses would be counter-productive because nearly half of the subjects in ME databases wear glasses

2.3 Feature Extraction Methods

Feature extraction methods can be divided into different categories according to three criteria:

- **Pre-designed or learned features:** Pre-designed features are hand-crafted to extract relevant information. Learned features are automatically learned from a set of training data (this is such the case of deep learning approaches) [38]. Pre-designed features can be further divided into **geometrical** and **appearance** features.
- **Global or Local features:** Global features or holistic features extract information from the whole face and local features from specific regions of interest where the most relevant information might be found.
- **Static or Dynamic features:** Static features describes the information of a single frame or image while dynamic ones analyse image sequences and extract the temporal information.

2.3.1 Geometrical Features

Geometrical features aim to describe the shape of facial features and their deformation caused by facial expressions. Geometric features for RGB modality usually describe the face deformation based on the position of specific facial landmarks.

Static Features Although methods like [27] measure the distance between facial landmarks by tracking markers placed on a person face, the usual methodology is to use markerless face registration methods like ASM [87], CLM [32], Bézier volume deformation tracker [11, 34, 35] or SDM [125] for the same purpose. On the other hand, [106] proposes to use variable-intensity templates to describe the facial deformation as a change in the intensities of multiple points fixed on a rigid shape model. Some authors propose to use the measured distance from tracking methods and compute a series of time-domain statistics (such as velocity, average, standard deviation) on some short time intervals as additional input [2].

Dynamic Features The idea is to describe how the face geometry changes over time. Classically for RGB data, the facial motion is estimated by the difference of color or intensity between the pixels of consecutive frames, usually through Optical Flow (OF) [86]⁴. Inspired in the success of histogram features success in the object recognition community, [29] proposed the Histogram of Oriented OF (HOOF) descriptor to model the distribution of OF

⁴The description of how optical flow can be found in Sec. 3.1.1

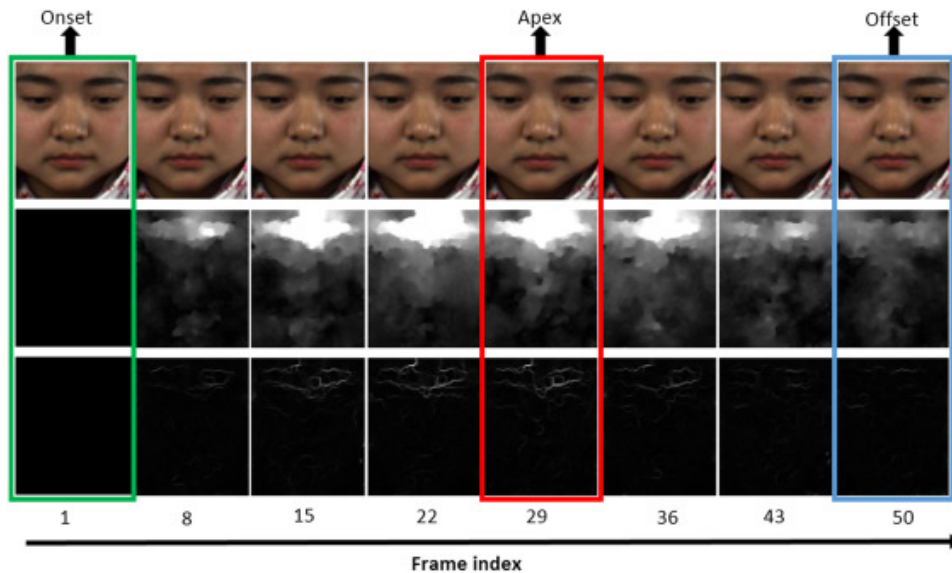


Figure 2.5.: Illustration of (top row) original images; (middle row) optical flow magnitude computed between the onset and subsequent frames; and (bottom row) optical strain computed between the onset and subsequent frames [118].

during a video sequence. Another descriptor derives the OF vectors to calculate the strain or non-rigid deformation [183]. This technique has been used for the analysis of micro expressions [115, 116] and to evaluate patients with facial palsy [186]. Methods like Main Directional Mean Optical flow (MDMO) [123] and Facial Dynamic Maps (FDM) [229] extract OF motion vectors from previously divided ROIs. [118] proposes to calculate the OF between the onset and apex frames and calculate a Bi-Weighted oriented OF descriptor (BI-WOOF) which is a variation of HOOOF in which the OF strain is used as a weighting coefficient (see Fig. 2.5). Other authors have proposed to calculate which facial areas have consistent and relevant facial motion and use it to filter the OF [5]. Another descriptor called Fusion Motion Boundary Histograms (FMBH) creates a series of weighted histograms from the horizontal and vertical derivatives of OF [125].

Although geometrical features are effective to describe facial macro expressions they fail to detect subtler changes like wrinkles and skin texture changes. Furthermore geometric features cannot be extracted from thermal images since facial features are dull and plain, thus it becomes very difficult to accurately localize any fiducial point [38].

2.3.2 Appearance Features

Appearance features use the intensity information of the image with the intention of describing facial features that appear temporally during any kind of facial expressions. They are also more stable to noise allowing for the detection of a more complete set of facial expressions, being particularly important for detecting micro expressions [38]. For RGB

data, usually these descriptors are applied either over the whole facial patch or at each cell of a grid.

Static Features are mostly based on standard feature descriptors and can either be applied to the whole facial area or at each cell of the face divided by a grid or applied to specific regions of interest. Some examples are:

- **Gabor Filters:** it extracts features at different scales and orientations by convolving the image with a bank of oriented bandpass filters. It has been used in [223, 240]
- **Local Binary Patterns (LBP):** this tool is a texture operator which labels the pixels of an image by thresholding the neighborhood of each pixel and considers the result as a binary number [162]. It has been used by [180, 231].
- **Histogram of Oriented Gradients (HOG):** The basic idea is that local object appearance and shape can often be characterized by the distribution of local intensity gradients or edge directions. This is implemented by dividing the image into small spatial regions (“cells”), and for each cell, a local 1-D histogram of gradient directions or edge orientations is accumulated over the pixels of the cell [43]. It has been used by [50, 112].
- **Integral Projection (IP):** is a one-dimensional pattern, or signal, obtained through the sum of a given set of pixels along a given direction. Horizontal and vertical integral projections are most commonly used, although they can be applied on any direction [131]. It has been used by [126].

Some other static holistic methods aim to reduce the dimensionality of the facial data using principal component analysis (PCA) [110] and Discriminant Tensor Subspace Analysis (DTSA) [215]. Some authors propose to create a set of Gabor magnitude response images and then apply the LBP operator in each of them (LGBP) in order to reduce the data dimensionality [227].

In the thermal imaging case, the descriptors exploit the difference of temperature between regions. 2D Discrete Cosine Transform (2D-DCT) is used in [101, 234] to decompose the face into cosine waves and the feature vector is generated using a heuristic rule. Gray Level Co-Occurrence Matrices (GLCM) are used in [85] to encode texture information by representing the occurrence frequencies of pairs of pixels intensities at a given distance. Dimension reduction techniques have also been used for extracting features in thermal images. PCA was used in [203] and also Eigenspace Method based on Class Features (EMC) [144, 145]. The difference between PCA and EMC is that PCA finds the eigenvector to maximize the total variance of the projection to line, while EMC maximizes the difference

between the within-class and between-class variance [143]. Another method proposed by [3] called Kernel PCA projects the data into non-linear high dimensional space using the kernel method before applying PCA in order to capture the non-linear relationships among the pixels.

Dynamic Features A very basic example of dynamic appearance descriptor is proposed by [22] by calculating the absolute difference between a current and previous frame. However, most dynamic global appearance descriptors are 3D extensions of already established appearance descriptors. For instance, Local Binary Patterns from Three Orthogonal Planes (LBP-TOP) is an extension of LBP which takes a stack of consecutive frames as 3D volume and compute it over three orthogonal planes [97, 112, 142, 152, 161, 239]. [214] projects the images into a tensor independent color space (TICS) before applying LBP-TOP. Due to the popularity of LBP-TOP, a plethora of variations have emerged for feature extraction including:

- LGCP-TOP: A descriptor similar to LBP-TOP but more robust to lighting changes [137].
- LBP-SIP: LBP with Six Intersection Points reduces the redundancy in LBP-TOP patterns [217].
- MOP-LBP: Mean Orthogonal Plane LBP takes the average plane from each stack first, and then compute the LBP on the three average planes [216] (see Fig. 2.6).
- CBP-TOP: Centralized Binary Pattern compares the center pixel point with pairs of neighbor points to obtain CBP codes which are insensitive to white noise and decrease the histogram length considerably [79].
- STCLQP: Spatiotemporal Completed Local Quantized Pattern also extracts sign, magnitude and orientation information while obtaining a compact and discriminative codebook [91].
- STLBP-IIP: Spatio-Temporal LBP with Improved Integral Projection preserves the shape property of micro-expressions and then enhance discrimination of the features for micro-expression recognition [89, 90].

Other extensions of standard descriptors are 3D Histogram of Oriented Gradients (3D-HOG) [163, 164] and 3D Histogram of Image Oriented Gradients (HIGO) which is similar to HOG which is robust against changes in illumination and contrast [111]. [148] uses the Riesz wavelet to transform the input images into multi-scale monogenic wavelets, then it extract features from the magnitude, orientation and phase from each scale.

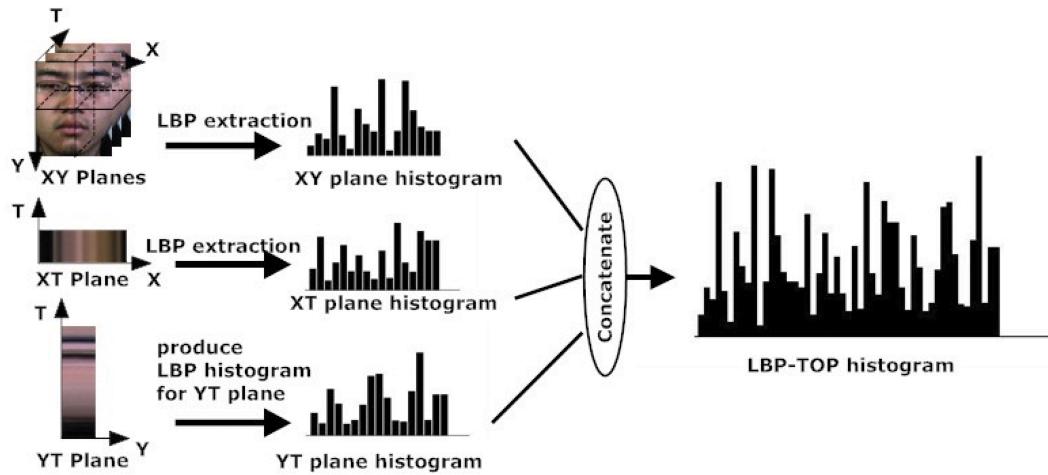


Figure 2.6.: LBP-TOP is a 3D variant of LBP which considers the co-occurrence statistic on three orthogonal planes: XY, XT and YT. [216]

2.3.3 Hybrid Features

Since geometrical features gives a different information than appearance features some works have considered using a combination of both. In the static case, [153] combines the geometrical features tracked by AAM with an extracted appearance vector and [119] concatenates orientation, magnitude and optical strain extracted from OF and the phase components from the monogenic signal obtained by the Riesz Pyramid. In the dynamic case some works have propose to use optical strain as a weight to for LBP-TOP (OSW-LBP-TOP) [116]. Other authors propose a 2-step approach in which first they detect spatio-temporal interest point (STIP) using Harris3D point detector followed by HOG descriptor to extract local space-time features [190]. Another 2-step approach first extracts the sparse part of Robust PCA (RPCA) of an image and then apply a Local Spatiotemporal Directional Features (LSTD) descriptor such as LBP-TOP [213].

2.4 Facial Expression Spotting

Macro or micro-expression spotting refers to the process to detect the moment when a facial expression takes place. Although, many authors prefer to use temporally and manually segmented videos for facial expression analysis, there are a lot of potential real-life applications where such information might not be available. Thus, spotting facial expressions becomes a primary step for a fully automated facial expression recognition system. The first step in ME spotting methods generally is to locate and detect the face and its main landmarks (Sec. 2.2). The second step is to extract some features from the face. The third step, which is to temporally analyse the extracted features, can be divided into 2 types: heuristic and trained methods.

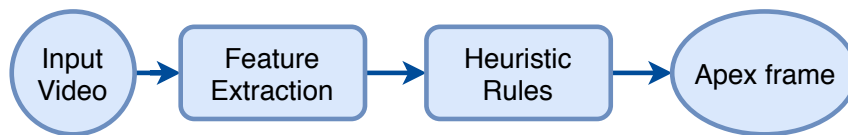


Figure 2.7.: Heuristic ME spotting general framework

2.4.1 Heuristic methods

Generally speaking, these methods work by extracting features from all the video frames, designing a set of rules that computes the temporal relationship of the extracted features between frames, and finally creating a searching method that locates the apex frame (Fig. 2.7). In heuristic spotting methods, MEs are spotted without training a model a priori. A summary of these methods can be found in Table 2.1.

Similarly to discussed in Sec. 2.3, feature extraction methods for ME spotting can be divided into appearance and geometrical features methods. In the case of appearance features, some methods choose to process each frame using LBP [112, 117, 231] and IP [126], or a sequence of frames using 3D-HOG [47]. In the case of geometrical features, some authors propose to use the displacement between frames of facial landmarks tracked using CLM [117, 231]. Some other authors propose to exploit optical flow between different frames and further process its vectors to analyze subtle motion. For instance, [154] propose to extract the OF vector from small local spatial regions (based on facial landmarks) and integrate them into motion features. Other authors propose to model MEs as face deformations by calculating its Optical Strain (which is the 2D-derivative of OF) and extract its magnitude [117, 185, 184, 182]. Another approach is to simply extract the OF from different gridded ROIs and integrate them into a series of histograms (HOOF) [112].

In the case of apex detection, the apex is generally obtained by comparing the features from different frames and create a temporal 1D signal from which the highest peak will correspond to the apex. For example, [117, 184, 185, 182] sums the magnitudes of the optical strain magnitude to create the signal. Some authors propose to obtain the apex by comparing the histogram difference [47] or the correlation [117, 231] between the features extracted from the first frame and the ones from the rest of the frames. One proposed method, called feature difference analysis, compares the differences of the features of sequential video frames within a specified interval in order to create the temporal signal [112, 140, 126]. In [154], the signal is calculated by adding the vectors of OF motion features from consecutive frames starting from the first frame of the video.

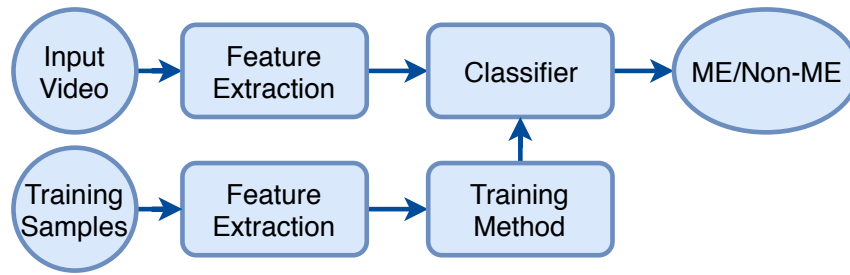


Figure 2.8.: Trained ME spotting general framework

2.4.2 Trained Methods

In trained spotting methods, MEs are spotted by training a model that detects ME in a video sequence. The detection becomes a classification problem of MEs vs Non-MEs. Consequently, the set of rules that determine whether an ME takes place is depends on the training method and the training samples. Generally speaking, these methods work by first extracting features from the video sequence, and then scanning the processed video with a trained system to classify frames as MEs or Non-MEs (Fig. 2.8). A summary of these methods can be found in Table 2.2.

In the case of appearance features extraction methods, some authors choose to process the motion magnitude variation between a past and current frame [22] while others use dynamic appearance features such as LBP-TOP, HOG-TOP or HIGO-TOP [202]. In the case of geometrical features, [125] proposes to measure the geometrical distance or displacement between key facial landmarks along the video sequence, while [226] proposes to measure the geometrical deformation, that is, the displacement of the facial landmarks of an ASM model.

From the features extracted, a series of ME and non-ME samples are taken in order to train a classifier. The detection steps come by training a classifier using ME and non-ME samples and then scanning the video sequence. Two things must be considered while designing a trained spotting method: how to deal with the scale of the MEs (different MEs have different lengths) and how to integrate the spotting results (to avoid predicting multiple MEs around the real apex). Some authors propose to use either nearest-neighbor interpolation [125] or Temporal Interpolation Model (TIM) [202] to the extracted feature vector in order to keep the same scale before classifying using a SVM. On the other hand, [22] proposes to train a series of weak classifiers and merge the spotted results that are close to each other. A more complex method is proposed by [226], in which, an Adaboost model estimates the initial probability that a frame contains an ME and then uses a random walk model to compute the probability for a sequence of frames contains an ME.

Category	Feature Extraction	Detection Method	Dataset	Results	Evaluation Metric
Appearance features	LBP [117]	Correlation Coefficient	CASME II	MAE = 13.55% SE = 0.79	Mean Absolute & Standard Error
	LBP [112]	Feature difference	SMIC	SMIC-HS = 0.8332 SMIC-VIS = 0.8453 SMIC-NIR = 0.8060	Area under ROC curve
	LBP [231]	Correlation Coefficient	CASME II (50 samples)	ME = 0.31 frames SE = 1.56	Mean Error & Standard Error
	Integral Projection [126]	Feature difference	CASME	AUC-A = 0.82 AUC-B = 0.90	Area under ROC curve
	3D-HOG [47]	Histogram Distance	SAMM	AUC = 0.7513	Area under ROC curve
Geometrical features	CLM [231]	Correlation Coefficient	CASME II (50 samples)	ME = 1.02 frames SE = 1.88	Mean Error & Standard Error
	CLM [117]	Subtraction	CASME II	MAE = 17.21% SE = 0.89	Mean Absolute & Standard Error
	Optical Strain [117]	Strain Magnitude		MAE = 14.43% SE = 0.83	Standard Error
	OF Motion features [154]	Motion Features	SMIC	AUC = 0.95	Area under TP vs FP curve
	Optical Strain [182]	Strain Magnitude	USF-HD	AUC macro = 0.85 AUC micro = 0.62	Area under ROC curve
			SMIC	AUC = 0.94	
	HOOF [112]	Feature difference	SMIC	SMIC-HS = 0.6941 SMIC-VIS = 0.7490 SMIC-NIR = 0.7323	Area under ROC curve
			CASME II	AUC = 0.6499	

Table 2.1.: Heuristic ME spotting methods

Category	Feature Extraction	Classifier	Dataset	Results	Evaluation Metric
Appearance features	Motion Magnitude [22]	Adaboost	CASME II	Acc = 81.75%	Accuracy
	LBP-TOP [202]		SMIC	Acc = 76.71%	
	HOG-TOP [202]	SVM	CASME II	27.19%	Miss rate
	HIGO-TOP [202]			25.39%	
Geometrical features	Facial Geometric [125]	SVM	SMIC	SMIC-HS = 84.15% SMIC-VIS = 74.90% SMIC-NIR = 73.23%	ME vs Non-ME
	Geometrical Deformation [226]	RW-adaboost	SMIC	AUC = 0.8693	Area under ROC curve
			CASME	AUC = 0.9208	

Table 2.2.: Trained ME spotting methods

2.5 Classification Methods

2.5.1 Static Models

Static models evaluate each frame independently, using classification techniques such as:

- **Artificial Neural Networks (ANN):** It is a computational model based on the structure and functions of biological neural networks [196]. It is used in [14, 235]. Some variations have been proposed such as neurofuzzy networks [92] and a single layered network called Extreme Learning Machine (ELM) [215].
- **k-Nearest Neighbor:** it is a simple algorithm that stores all available cases and classifies new cases based on a similarity measure [176]. it is used in [101, 110, 227].
- **Boosting:** it is an ensemble technique that attempts to create a strong classifier from a number of weak classifiers [25]. AdaBoost (Adaptive Boosting) is used in [15].
- **Support Vector Machine (SVM):** it is a discriminative classifier formally defined by a separating hyperplane. In other words, given labeled training data (supervised learning), the algorithm outputs an optimal hyperplane which categorizes new examples [150]⁵. it is used in [50, 90, 115, 116, 152, 214, 217], among others. [161] proposed a SVM variation called Multiple Kernel Learning (MKL) in which a combination of multiple kernels are selected instead of just one.
- **Naive Bayes Classifier:** it is a classifier based on applying Bayes' theorem with strong (naive) independence assumptions between the features. [11] and [35] proposed to classify FEs using both Naive Bayes Classifier and a variation called Tree-Augmented Naive Bayes (TAN).
- **K-Means:** it is a clustering technique in which k-centroids are considered to be the mean of a group of points and is applicable to objects in a continuous n-dimensional space [205]. It has been used to classify the motion in specific ROIs during any FE [164, 163].
- **Sparse Representation Classifier:** it is based on the discriminative nature of sparse representation, where test samples can be represented as a linear combination of basis vectors selected from a dictionary [190, 221, 240].

⁵A more complete definition and formulation can be found in Sec. D

- **Ensemble Learning:** They use multiple learning algorithms to obtain better predictive performance. For example, GentleSVM uses Gentleboost, a variation of Adaboost that converges faster, as a feature selector for a SVM classifier [223].
- **Gaussian Fitting:** A 2D gaussian model is fit to the training data for each class. A new input is tested by computing its probability to belong to any given class [22].

2.5.2 Dynamic Models

Dynamic models take into account features extracted independently from each frame to model the evolution of the expression over time [38]:

- **Hidden Markov Models (HMM):** It is a type of probabilistic model often used to model temporal data. A Markov model is a system that produces a Markov chain (an observed sequence), and a hidden Markov model is one where the rules for producing the chain are unknown or “hidden”. The rules include two probabilities: (i) that there will be a certain observation and (ii) that there will be a certain state transition, given the state of the model at a certain time [21]. In [124], HMM are used to recognize emotion states. [114] uses a combination of discriminant analysis and HMM. In [34, 35], the authors propose to use a multilevel HMM both for FE spotting and recognition.
- **Restricted Boltzman Machines (RBM):** They are models that permits tractable inference but allows much more complicated structure to be extracted from time series data than HMMs. A variation called Temporal RBM is used in [237].

2.5.3 Deep Learning

Learned features are usually trained through a joint feature learning and classification pipeline [38]. For instance, [83] proposes a multi-task mid-level feature learning method to enhance the discrimination ability of extracted low-level features by learning a set of class-specific feature mappings, which would be used for generating a mid-level feature representation.

More recently, deep learning architectures have been used to jointly perform feature extraction and recognition. In [122], a two-step iterative process is used to train Boosted Deep Belief Networks (BDBN) where each DBN learns a non-linear feature from a face patch, jointly performing feature learning, selection and classifier training. [84] uses a Deep Boltzmann Machine (DBM) to detect FEs from thermal images. A hybrid method used a Convolutional Neural Networks (CNN) to predict emotions in still frames and the

extracted probability vectors are set to a fixed length and then used to train an SVM [98]. In [100], the spatial features of micro-expression at different expression-states are encoded using a CNN and then the temporal characteristics of the different expression-states of the micro-expression are encoded using long short-term memory (LSTM) recurrent neural networks. In order to avoid to train a CNN that only learns low level features, some authors have proposed to estimate the OF of a video sequence and then use these high level features to feed a Dual Temporal Scale Convolutional Neural Network (DTSCNN) [158].

2.6 Multimodal Fusion

Many works have considered multimodality for recognizing emotions, either by considering different visual modalities describing the face or, more commonly, by using other sources of information (e.g. audio or physiological data). Fusing multiple modalities has the advantage of increased robustness and conveying complementary information [38]. For instance RGB data conveys information about the face texture while thermal images can detect changes in the blood flow produced by emotions.

The fusion approaches of different multimodal approaches can be classified in 4 main categories: direct data, feature, decision and sequential fusion (see Figure 2.9).

- **Direct data fusion** or **Input fusion** directly merges the input data from different modalities. This approach has the advantage of allowing the extraction of features from a richer data source, but is limited to input data correlated for both spatial and temporal domains.
- **Feature Fusion** merges the modalities at the feature level. Feature fusion directly exploits correlations between features from different modalities, and is specially useful when sources are synchronous in time. However, it forces the classifier to work with a higher-dimensional feature space, increasing the likelihood of over-fitting. The most straightforward approach is plain feature fusion which consists in concatenating the feature vector from both modalities. Such is the case of [143] and [144] to fuse RGB and thermal imaging. This can be used even when the data comes from non spatially correlated modalities like the case of [218] which fuses video data with biomedical signals (galvanic skin response, electromyography and electrocardiogram). Some works use feature selection techniques like [16] which uses genetic algorithms in order to select the best features to merge RGB and thermal imaging.
- **Decision Fusion** merges the data after applying expression recognition, at the decision level. Decision fusion is usually considered for asynchronous data sources, and can be trained on modality-specific datasets, increasing the amount of available

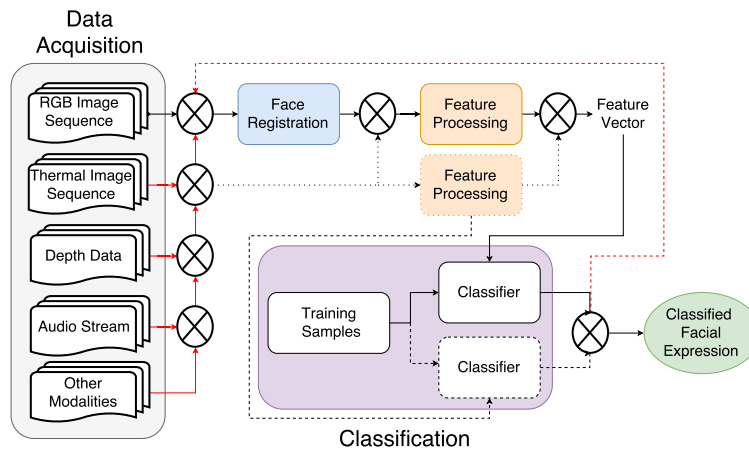


Figure 2.9.: General Framework for different modality fusion approaches. The product signs represent the modality fusion strategy. The specific components of different strategies are represented with different types of lines. The red line corresponds to input fusion, the black dotted line to feature fusion, the black dashed line to decision fusion and the red dashed line to sequential fusion

data. Decision fusion merges the results of multiple classifiers/regressors into a final prediction. The goal is either to obtain a final class prediction, a continuous output specifying the intensity/confidence for each expression or a continuous value for each dimension in the case of continuous representations. The simplest approach is the maximum rule, which selects the maximum of all posterior probabilities. This technique is sensible to high-confidence errors. A classifier incorrectly predicting a class with high confidence would be frequently selected [38]. This can be improved if one considers the strengths of each individual modality. The weight criterion solves this by assigning a confidence to each classifier output. For instance, [143] and [144] manually set the weights of the classified facial expressions using RGB and thermal data. [98] needed to combine the predictions of different modality-specific models, including: a CNN trained to recognize facial expressions in single frames, a DBN that is trained on audio information, a relational autoencoder that learns spatio-temporal features in order to capture human actions; and shallow network that is trained on visual features extracted around the mouth of the human subject in video. Thus they develop a technique of aggregating the per model and per class predictions via random search over simple weighted averages.

- **Sequential Fusion** is a technique that applies the different modality predictions in sequential order. It uses the results of one modality to disambiguate those of another when needed. For instance [234] combines thermal and acoustic data. The method uses a speech recognition system to detect the three timing positions of just before speaking, and just when speaking the phonemes of the first and last vowels and select their corresponding thermal images in order to extract their features.

2.7 State of the art summary

In this section we present a small review of the methods for ME spotting and recognition systems. A summary of the spotting methods presented in Sec. 2.4 are presented in Table 2.1 for non-trained methods and Table 2.2 for trained methods. We divide the methods using our established taxonomy into appearance and geometrical features groups and highlight both their specific feature extraction method, detection method, as well their reported results in different datasets with their respective evaluation metric. As we can see from these tables, there has been a similar quantity of method that extract either appearance or geometrical features. Also, most of the work use either SMIC or CASME II datasets to test their work. The reason might be because both datasets are captured with high speed rate giving enough frames to analyse the motion of an ME. Furthermore, both datasets contain short videos of 3-8 seconds with one or two MEs which, considering the duration of MEs (Sec. 1.2), are long enough for a spotting experiment. We can also observe that different authors use different metrics (such as mean error, standard error, ROC curves, TP vs FP, miss-rate, etc.)⁶.

A summary of the ME recognition methods is presented in Table 2.3. We divide the methods using our established taxonomy into appearance, geometrical, hybrid and learned features groups and highlight both their specific feature extraction method. We also divide the classification families between SVM which cover a majority of the approaches (we specify the chosen kernel), deep neural network and others (classification methods that were used just once). We also show the reported results in different datasets with their respective evaluation metric (we added “-DET” to a database when the experiment was recognizing between MEs and not MEs). Most reported results were tested under a Leave-One-Subject-out (LOSO) cross-validation (CV), although there are some results tested with k-fold validation, or data split into training/validation sets (there were a couple of works that didn’t disclose their testing methodology). Most results measured the accuracy percentage of their methods and some others measured their F-score. As we can see, a majority of the works used appearance based descriptors for their feature extraction step. Furthermore a majority of the works choose to use support vector machines as their classifiers.

We also decided to track the time-line of ME analysis research using computer vision (Fig. 2.10) from the first research works until the most recent ones⁷. As we can see, since 2014 there has been an increment in the research of both ME classification and spotting. However, there is much more focus on the classification task compared to the spotting task (more than twice the number of papers focused in classification).

⁶We will discuss further about this in Sec. 4.4.4

⁷This chart does not include the works published in 2018

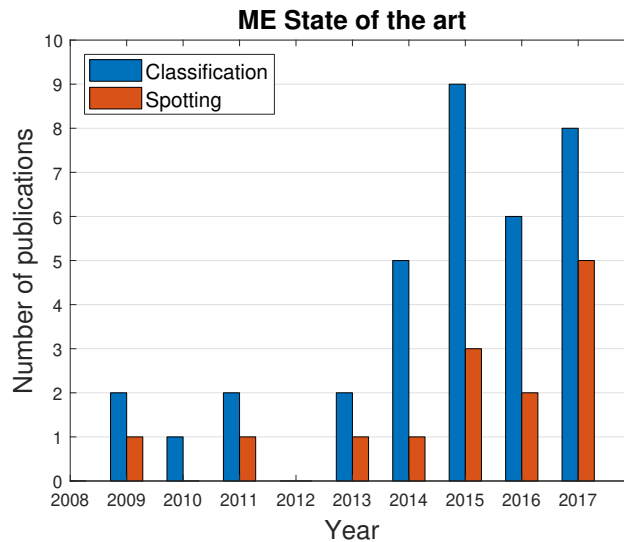


Figure 2.10.: ME research over the years

2.8 Discussion

From the state of the art review, we can draw some initial points for a possible ME analysis framework.

1. A good framework should not depend on cropped face data, but rather implement a face registration which would be robust enough to get precise facial ROI but adaptable enough to adapt to present and future ME databases. Fortunately, since the current available databases have been recorded in a controlled environments, where there are no radical changes in illumination and pose nor heavy occlusion, a simple frontal face registration scheme would suffice.
2. A careful selection of the facial regions of interest is essential before the feature extraction step. Taking in considerations what was discussed in Sec. 2.2.3, it seems that both ways of selecting ROIs have advantages and disadvantages. Thus, the selection of ROIs should depend on the specific application.
3. A good feature extraction method for ME analysis would be one that is able to characterize motion, thus, one should choose a dynamic method, or at least complement a static method with a feature difference method. Furthermore, in the case of ME spotting, one should choose a method that is sensible enough to detect subtle motions but robust enough to ignore image noise.
4. In the case of ME classification, a good feature extraction method would be one who would be able to clearly characterize the difference between MEs. For example, when

analysing the eyebrows movement during an ME, a good geometrical dynamic feature descriptor would exploit the difference in direction when a person is surprised (the eyebrows raise) compared when is angry (the eyebrows lower).

5. A good analysis framework would consider to both spot and classify MEs since great contributions remain to be made in both areas. Considering that the feature extractions used for ME spotting are similar to the ones used for recognition (See Sec. 2.4), it is possible to propose an unified feature extraction methodology that is adapted for both cases.
6. It seems that a great majority of the feature extraction methods used for ME analysis are appearance based (most of them variations of LBP-TOP) or if they are geometrical based they are based on Optical flow. A good contribution would be to propose a method that goes outside these paradigms and offers a fresh perspective on the ME analysis problem.
7. A good classification method, would be one that effectively learns the differences between features, given the limited data. Although, Deep learning methods have become popular in recent years, they require a great amount of data, which current ME datasets do not possess.
8. Only one of the given ME databases provide any type of multimodal information (SMIC database has both color and near-infrared images), thus only unimodal approaches can be proposed for the moment⁸.

2.9 Chapter Conclusions

In this chapter we introduced a taxonomy to divide the different steps of automatic facial macro and micro expressions analysis systems. Furthermore, we did a review of the state of the art in face localization, feature extraction, facial expression spotting, classification methods and multi-modal fusion used for both macro and micro-expression analysis.

After summarizing the state of the art in micro-expression analysis, we were able to remark certain current tendencies which will allow us to formulate the guidelines to construct a proper ME analysis framework.

⁸Some other multimodal ME databases might appear in the future like the one proposed in Sec. A

Micro Expression Classification Methods							
Feature Extraction		Classifier		Dataset	Results	Evaluation Metric	
Category	Approach	Category	Approach				
Appearance features	HOG [50]	Support Vector Machine	Linear Kernel	CK+	80.00%	Not Provided	
	STCLQP [91]		Linear Kernel	SMIC	64.02%	LOSO	
				SMIC-Det	75.31%		
				CASME II	58.39%		
				CASME	57.31%		
	STLB-IP [89]		Chi-square kernel	SMIC	57.93%	LOSO	
				CASME II	59.51%		
	STLB-IIP [90]		Chi-square kernel	SMIC	63.41%	LOSO	
				CASME II	64.78%		
	LBP [112]		LSVM	SMIC	HS = 60.37% VIS = 78.87% NIR = 67.61%	LOSO	
				CASME II	64.78%		
	SMIC			HS = 61.59% VIS = 77.46% NIR = 64.79%			
				CASME II	63.97%		
	SMIC	HS = 68.29% VIS = 81.69% NIR = 67.61%					
		CASME II		67.21%			
	LBP-TOP [137]	RBF kernel		SMIC-Det	68.9%		75/25 data split
				SMIC	48.6%		
	LGCP-TOP [137]	RBF kernel		SMIC-Det	61.2%		75/25 data split
				SMIC	48.1%		
	LBP-TOP [142]	RBF kernel	CASME II	51.00%	Acc LOSO		
				0.47%	F-score		
	Riesz Wavelet [148]	RBF kernel	CASME II	46.15%	Acc LOSO		
				0.4307%	F-score		
	LBP-TOP & Intensity variation [153]	RBF kernel	CASME II	51.91%	LOSO		
	MOP-LBP [216]	Polynomial kernel	SMIC	50.00%	LOSO		
		RBF kernel	CASME II	45.75%			
LBP-SIP [217]	Linear kernel	SMIC	64.02%	LOSO			
	RBF kernel	CASME II	66.40%				
LBP-TOP & TIC [214]	Linear kernel	CASME II (4 classes)	62.30%	LOSO			
Gabor Filters [223]	GentleSVM	CK+	92.66%	10-fold CV			
Gabor Filters [223]	Others	Gentle Adaboost	CK+	88.61%	10-fold CV		
Motion Magnitude[22]		2D Gaussian Modelling	SMIC	79.48%	Not Provided		
	CASME		85.71%				

Micro Expression Classification Methods							
Feature Extraction		Classifier		Dataset	Results	Evaluation Metric	
Category	Approach	Category	Approach				
Appearance features	3D-HOG [163, 164]	Others	K-means	Polikowsky	-	AU recognition	
	CBP-TOP [79]		Extreme Learning Machine	CASME	82.07%	60/40 data split	
	LBP-TOP [79]		Nearest Neighbour	CASME II & CK+	65.55%	Random sub-sampling validation	
	LBP-TOP [161]		Random Forest	SMIC-Det	74.3%	LOSO	
			Multiple Kernel Learning	SMIC (Neg/Pos)	71.4%		
	2D Gabor Filter [240]		SRC	CASME II	64.88%	LOSO	
		CASME	71.19%				
Geometrical features	RHPM [5]	Support Vector Machine	LibSVM	CASME II	65.35%	LOSO	
	BI-WOOF [118]		Non specified	SMIC	HS = 0.62 VIS = 0.58 NIR = 0.58	F-score LOSO	
					CASME II		0.61
	BI-WOOF + Riesz phase [119]		Linear kernel	SMIC	68.29%	Acc LOSO	
					0.67	F-score	
					CASME II	62.55%	Acc LOSO
	Optical Strain & Wiener Filter [115]		Linear Kernel	SMIC	0.65	F-score	
					53.56%	LOSO	
	MDMO [123]		Polynomial kernel	SMIC	58.97%	LOSO	
					CASME II		51.69%
					CASME		56.29%
	Facial Dynamics Map [229]		RBF kernel	SMIC-Det	75.30%	LOSO	
					SMIC		54.88%
					CASME II		41.96%
CASME		42.02%					
FMBH [125]	RBF kernel	SMIC	71.95%	LOSO			
			CASME II		69.11%		
			CASME		61.33%		
			CAS(ME) ²		73.67%		
Hybrid features	DLSTD [213]	Support Vector Machine	Non Specified	SMIC	67.68%	LOSO	
				CASME II	65.44%		
	AAM shape & appearance parameter [153]		Non specified	SFED2007	88.125%	50/50 data split	
	OSW-LBP-TOP [116]		Polynomial Kernel	SMIC	63.95%	LOSO	
	RBF Kernel	CASME II	66.40%	LOVO			

Micro Expression Classification Methods						
Feature Extraction		Classifier		Dataset	Results	Evaluation Metric
Category	Approach	Category	Approach			
Learned features	Feature Mapping [83]	Support Vector Machine	Non specified	SMIC	63.95%	LOSO
				CASME	59.81%	
Learned features	Learned [100]	Deep Neural Network	CNN & LSTM	CASME II augmented	60.98%	LOSO
	OF [158]		Dual Temporal Scale CNN	CASME + CASME II augmented	66.67%	3-fold CV

Table 2.3.: ME classifier methods

Subtle Motion Analysis using the Riesz Pyramid

Although the human visual system is capable of detecting a plethora of objects, actions and phenomena in our surrounding environment, it has limited spatio-temporal sensitivity. Consequently, there are some movements of low-spatial amplitude that are difficult to detect by the human eye. Processing and analysing these subtle motions might help to reveal interesting information about the world [120]. For instance, it has been reported that the cyclical movement of blood in the human body causes the head to move in a subtle periodic motion which has been used to calculate the heart rate in a non invasive manner [12, 95]. Furthermore, abnormal breathing patterns can be detected by analysing the body motion during inspiration and expiration [210]. Another example comes from modal analysis (the measuring of the dynamic response of structures, fluids or other system during excitation). The vibration of a running engine could be measured using a non-contact sensor such as a camera [31]. And as we have already previously mentioned, micro-expressions, which can be difficult to identify even for trained people [67, 96, 173], could be analysed using video processing.

As previously stated, the analysis of subtle motion has an interesting range of applications and the potential to become a whole field of its own in the area of motion analysis. Although our first instinct would be to analyse this type of motion using the classical tools for motion estimation, there are certain challenges that might make us rethink of this approach. First of all, classical methods might favour the measurement of larger motions and dismiss subtle motions as mere noise. Secondly, classical methods might face the challenge of either adjust their parameters to detect subtle motion but become sensible to noise, or they become strong to image noise but ignore subtle motion.

In recent years some authors have proposed a method called motion magnification in which either the magnitude or phase variations of subtle motions are amplified. Although these methods do make the subtle motion more evident for the human eye, there are some caveats. If the magnification method is amplitude based, it can amplify the image noise as well [222]. On the other hand, [207] have proposed a phase based method which does not amplify the noise. However since it is based on complex steerable pyramids, these systems are very overcomplete and costly to construct. A more recent method, based on the Riesz pyramid was proposed which is suitable for real-time phase-based video processing [209].

Nevertheless, the main problem with these techniques is that these methods exaggerate the motion rather than explicitly estimate it. However, our careful examination showed that intermediate representations produced by these methods, particularly their phase variations, can be used as proxies for motion. Specifically, the Riesz pyramid based representation has shown to be a simple, adaptable and fast-processing representation of subtle motions. Not only that, but considering that a lot of real-life applications require to detect when an event takes place, these representations could potentially be used to temporally spot subtle events (finding the temporal locations of subtle movements from a video sequence) and even, detect micro-expressions.

In this section, we propose a subtle motion analysis method, using the Riesz pyramid as its core. This chapter is organized as follows. Sec. 3.1 gives an introduction to the classical motion estimation techniques, motion amplification and phase-based motion analysis methods. Sec. 3.2 serves as an introduction to the Riesz transform and the monogenic signal. Sec. 3.3 presents the application of the monogenic signal for video sequences, its quaternionic representation and filtering. Sec. 3.4 describes a methodology for subtle motion analysis using the processed monogenic signal. Sec. 3.5 describes some potential applications, our experiments, results and discussion. Finally, Sec. 3.6 presents our conclusions for this chapter.

3.1 Background

3.1.1 Motion Estimation

Motion estimation is the process of examining the movement of objects in a video sequence by determining the motion vectors that describe the transformation between two consecutive frames. This information is fundamental for video understanding and object tracking. Relevant applications include video surveillance, robotics, autonomous vehicles navigation, human motion analysis, quality control in manufacturing, video search and retrieval, and video restoration [160]. The motion estimation methods can be divided into two groups: indirect or feature based methods and direct or pixel based methods.

Indirect Methods

This family of methods extracts a sparse set of distinct features from each image separately, and then recover and analyse their correspondences in order to determine the motion. The features are image edges, corners, and other structures well localized in two dimensions. These type of methods involves two stages. Firstly, the features are found in two or more consecutive images. Secondly, these features are matched between the frames (see Fig. 3.1).

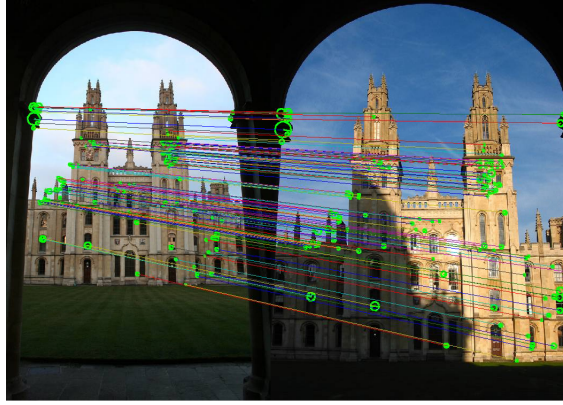


Figure 3.1.: Feature matching

In the simplest and commonest case, two frames are used and two sets of features are matched to give a single set of motion vectors.

Each of the two stages of feature-based flow estimation has its own problems. The feature detection stage requires features to be located accurately and reliably (which is not trivial task). The feature matching stage has the correspondence problem of ambiguous potential matches occurring (which might cause feature flicker).

Direct Methods

This family of methods extracts image measurable information at each pixel in the image in order to estimate the motion. The starting point for most direct methods is the “brightness constant constraint” (the assumption that the points of an object keep their gray value during motion), namely, given two images $J(x, y)$ and $I(x, y)$,

$$J(x, y) = I(x + u(x, y), y + v(x, y)) \quad (3.1)$$

where (x, y) are pixel coordinates, and (u, v) denotes the displacement of pixel (x, y) between two images. Assuming that the motion is small, and linearising I around (x, y) , we can obtain the following constraint:

$$I_x u + I_y v = -I_t \quad (3.2)$$

where (I_x, I_y) are the spatial derivatives of the image brightness and $I_t = I - J$. However, since the displacement of each pixel is defined by two quantities, u and v , the brightness constraint can determine only the “normal motion”, i.e., the motion along the direction of image gradient (this is called the “aperture problem”) [224]. Thus, the brightness constraint alone is insufficient to determine the displacement of a pixel [94]. To find the motion vectors

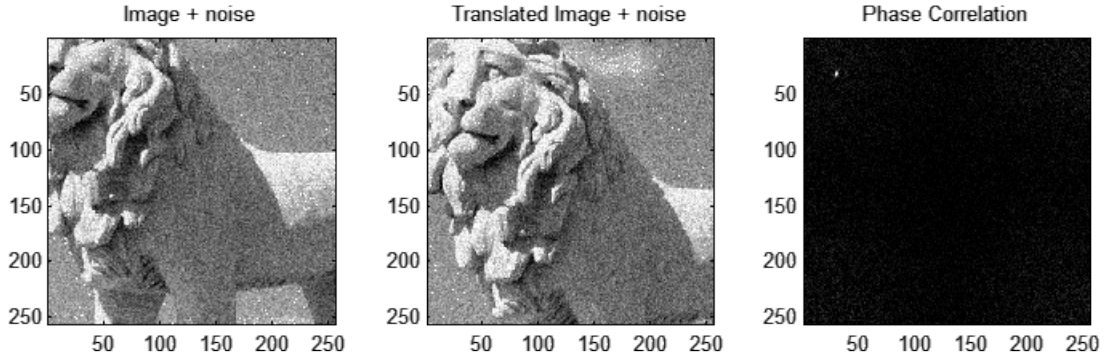


Figure 3.2.: Phase Correlation

another set of equations is needed, given by some additional constraints. Different direct methods introduce additional conditions for estimating the actual motion.

Phase Correlation: This method uses a frequency-domain approach (usually the fast Fourier transform) to estimate the relative translation offset between two similar images. The result of correlation between two images is an image which has peak intensities at locations where the two images match the best. The intuition behind correlation is that the resulting image will have the maximum value when the hills and troughs of the images match up. This is a global approach and the resulting motion vector represents the translation of the whole image. Given two input images I and J , calculate the 2-D Fourier transform in both images:

$$F_I = \mathcal{F}\{I\} \quad F_J = \mathcal{F}\{J\} \quad (3.3)$$

Calculate the cross-power spectrum by

$$R = \frac{F_I \circ F_J^*}{|F_I \circ F_J^*|} \quad (3.4)$$

where \circ is the entry-wise product and F_J^* is the complex conjugate of F_J . Then, obtain the normalized cross-correlation by applying the inverse Fourier transform.

$$r = \mathcal{F}^{-1}\{R\} \quad (3.5)$$

Determine the location of the peak in r .

$$(\Delta x, \Delta y) = \arg \max_{(x,y)} \{r\} \quad (3.6)$$

Block Matching Algorithm (BMA): In this method, the video frames are partitioned into non-overlapping blocks of pixels. Each block is predicted from a block of equal size in the

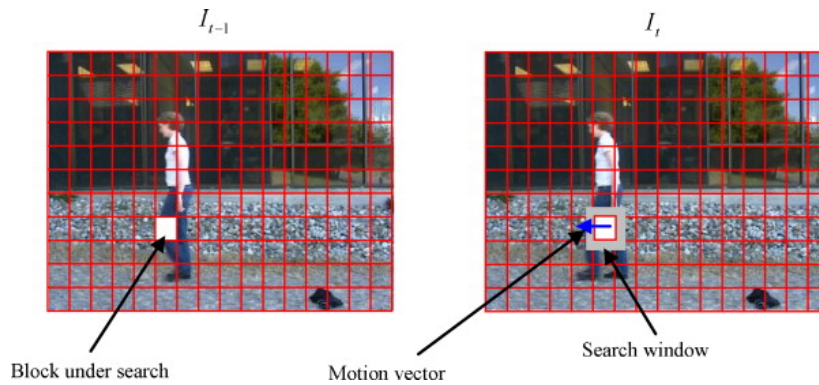


Figure 3.3.: Block Matching Algorithm [41]

previous frame. In particular, for each block at the current frame, the algorithm aims for the best matching block within a search window from the previous frame, while minimizing a certain matching metric [41]. The core assumption of all the BMAs is that the inter-frame changes are small. Some of the matching metrics used are mean absolute difference (MAD), mean squared error (MSE) or sum of absolute differences (SAD), cross-correlation coefficient (CCC) among others. Some of the algorithms used for block matching are exhaustive search, three step search, two dimensional logarithmic search, diamond search, among others.

Optical Flow (OF): It is one of the most known and used family of motion estimation techniques. This family uses a variety of differential methods to solve Eq. 3.2.

- **Lucas-Kanade method:** It assumes that the flow is essentially constant in a local neighbourhood of the pixel under consideration, and solves the equation by the least squares criterion [127].
- **Horn-Schunck:** It assumes smoothness in the flow over the whole image. Thus, it tries to minimize distortions in flow by introducing a global smoothness constraint [86]. The flow is formulated as a global energy functional which is then sought to be minimized. This technique has managed to stay reliable and accurate after 30 years without drastic changes on its formulation [194]. The main disadvantage of this method is that high gradients are penalized and it effectively disallows discontinuities [159].
- **Phase-based method:** It proposes to track constant phase contours as an approximation of the motion field by computing the phase gradient of a spatio-temporally bandpassed video [74]. [77] proposed a similar method but computing the phase gradient of images filtered with a bank of quadrature pair filters.
- **TV-L1:** It is based on the Horn-Schunck formulation but it allows discontinuities in the flow field. The flow can be understood as the minimization of a global energy



Figure 3.4.: Lucas-Kanade Dense Optical Flow [175]

functional by using the L^1 norm and a regularization term using the total variation of the flow [159, 236].

- **SIFT flow:** this method consists of matching densely sampled, pixel-wise SIFT features between two images, while preserving spatial discontinuities. The SIFT features allow robust matching across different scene/object appearances, whereas a discontinuity-preserving spatial model allows matching of objects located at different parts of the scene [121].
- **FlowNet:** In this method, a convolutional neural network is trained to learn strong features which are used to find correspondences between two frames [51].

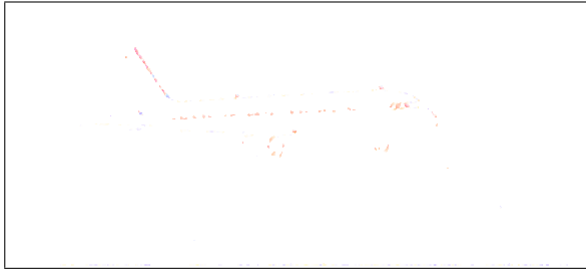
In some of these methods, a motion vector is assigned to each pixel of the image, resulting in a dense motion field (see Fig. 3.4). It has the advantage to provide a precise description of the motion. However it entails a costly motion representation.

Dealing with subtle motion

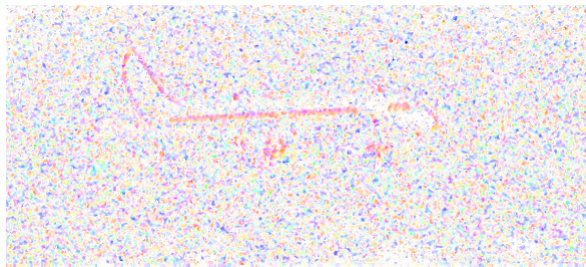
Although the previously mentioned methods have been successfully used in the past to analyse different kinds of motion, they were mostly designed to estimate evident and visible motion. Subtle motions, such as vibration and micro-expressions, are generally very fast and have low amplitude. In theory, using synthetic images, we could adjust the parameters of the motion estimation method until it is able to detect the desired motion. However, real images that were captured with any kind of sensor, will inevitably add some level of noise. For indirect estimation methods, this would mean an increase in the feature matching correspondence error (the matched features would flicker over time) making their variation difficult to distinguish from actual detected subtle motions. For direct methods, the presence of noise (which by definition is the random variation of brightness or color information in images) would go directly in contradiction of the constant brightness assumption. Thus,



(a) Input image with Gaussian noise



(b) Noise reduction is too high



(c) Noise reduction is too low

Figure 3.5.: Noise reduction effect in subtle motion estimation

these methods deal with noise by adding new parameters and constraints (like the smoothness constraint in Horn-Schunck OF method or the total variation of flow in TV-L1 OF method). However, if the magnitude of the image noise is comparable to the magnitude of the subtle motion, we face an implementation dilemma: we either adjust the parameters to ignore the noise, risking missing subtle motion information, or we adjust them to be sensible to subtle motion but risking falsely modelling noise as motion. Let's take Fig. 3.5a in which an airplane image is translated one pixel to the right in the presence of Gaussian noise. We apply the Lucas-Kanade optical flow method and manipulate the parameter that controls the noise reduction. Fig. 3.5b shows a false-color representation of the case in which the noise is ignored but also a lot of the motion information is ignored as well and Fig. 3.5b shows the case in which the subtle motion is modelled but also the noise is falsely modelled.

3.1.2 Multi-scale Representation

Another challenge that comes from analysing motion comes from the scale of the images and motion itself. Scenes in the world contain objects of many sizes moving at different

velocities. Moreover, two objects moving at the same speed that are at different distances from a viewer will be perceived as having different speeds. As a result, any analysis procedure that is applied only at a single scale may miss information at other scales. The solution is to carry out analyses at all scales simultaneously [6].

One way to do this is through the pyramid representation in which an image is repeatedly filtered and subsampled (usually by a scale of two). The generated pyramid is a sequence of images in which both sample density and resolution are decreased in regular steps [6]. If this sequence of images were stacked one atop the other they would form a pyramid (see Fig. 3.6) where the original image would be at the bottom layer (level 1), the first processed smaller image would be the second layer (level 2) and so on. Some examples of pyramid representation are:

- **Gaussian Pyramid:** It is the most basic of pyramid representation. It is constructed by repeatedly blurring (with a Gaussian averaging function) and subsampling an image.
- **Laplacian Pyramid:** First the Gaussian pyramid is calculated. Then, at each level, the Gaussian image is subtracted by the upsampled image of the following level to obtain the band-pass “Laplacian” image [26]. In order to enable the image reconstruction the most superior level of the pyramid has a downsampled image (not a “Laplacian” image).
- **Steerable pyramid:** The steerable pyramid was proposed by Simoncelli and Freeman [188, 189] and it is an over-complete transform that decomposes an image according to spatial scale, orientation, and position. The steerable pyramid is implemented by recursively splitting an image into a set of oriented sub-bands (by applying a bank of “steerable” (oriented) filters at each scale), a high-pass residual band and a low-pass residual band [166]. The main characteristic of the “steerable” filters is that their orientation can be arbitrary selected using a linear combination of “basis” filters.

Motion estimation techniques like optical flow also can benefit from pyramid representation. Some implementations called coarse-to-fine methods, build image pyramid for an image pair, estimate the motion at each level, and then, all the detected displacements are scaled and fused at the original level¹.

3.1.3 Motion Amplification

As discussed in Sec.3.1.1, subtle motion analysis is a non trivial task. Nevertheless, some authors have proposed to, instead of trying to analyse subtle motion as it was originally

¹This last step is also known as “collapsing the pyramid”.

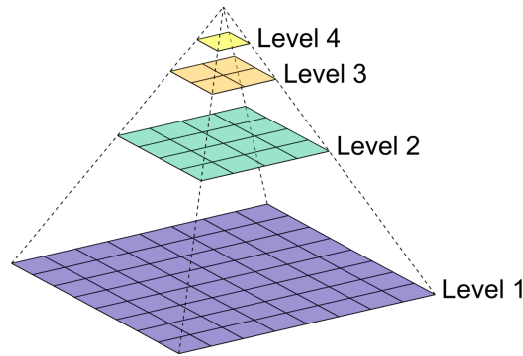


Figure 3.6.: Visual representation of image pyramid with 4 levels

captured, to apply some sort of preprocessing to an input video in order to amplify or reveal small motions. This family of techniques is called motion amplification or magnification.

There are two different perspectives on how to analyze the motion to be amplified. The first one is called *Lagrangian* perspective, in reference to fluid dynamics where the trajectory of particles is tracked over time [222]. For instance in [120] motion is computed explicitly (using optical flow) and the frames of the video are warped according to the magnified velocity vectors. [211] proposes a cartoon animation filter that exaggerates the motion in order to make the output more “alive” or “animated”.

The second one is called *Eulerian* perspective where properties of a voxel of fluid, such as pressure and velocity, evolve over time. These methods study and amplify the variation of pixel values over time, in a spatially-multiscale manner. These techniques have been used for modal analysis (the study of the dynamic properties of structures under vibrational excitation) [31], for enhancing the motion of the blood flow in the finger veins for liveness detection [169], to enhance the movement of facial expressions for anti-spoofing in a face biometric system [20], for amplifying the motion of pulsating arteries during an endoscopic surgery [136] and for micro-expression recognition [153].

3.1.4 Linear Video Magnification

The basic approach is to consider the time series of color values at any pixel and amplify variation in a given temporal frequency band of interest. This is done by first decomposing the video sequence into different spatial frequency bands (using a Laplacian pyramid). Then each spatial band is temporally processed. For each time series corresponding to the value of a pixel in a frequency band, a bandpass filter is applied to extract the frequency bands of interest. For example, if we want to magnify the human pulse (24–240 beats per minute), the filter should bandpass frequencies between 0.4 – 4 Hz. The temporal processing is uniform for all spatial levels, and for all pixels within each level. Then, the extracted filtered signal is multiplied by a magnification factor α (this factor can be specified by the user). Next,

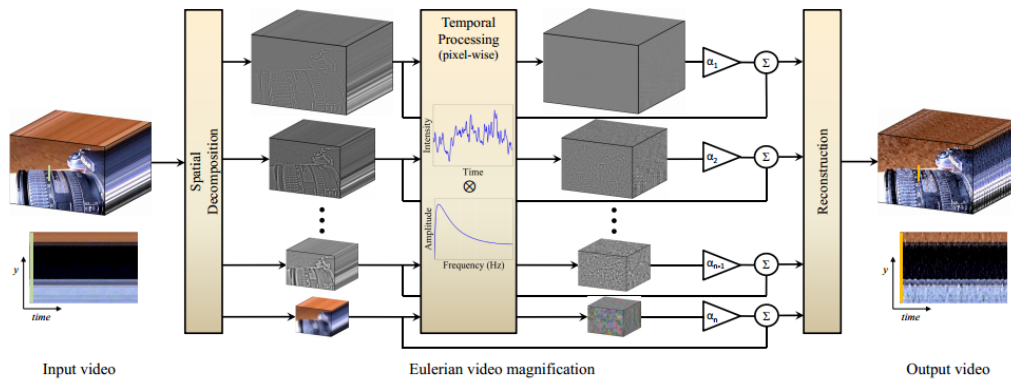


Figure 3.7.: Eulerian motion amplification framework [222]

the magnified signal is added to the original and the resulted spatial pyramid is collapsed to obtain the final output [222]. The complete framework is shown in Fig. 3.7. The main drawback of this method, is that it can significantly amplify noise when the magnification factor is increased.

3.1.5 Phase-based Magnification

The first attempts to analyse motion based on local phase can be tracked back to Fleet and Jepson who showed that the temporal evolution of contours of constant phase provides a better approximation to the local velocity than do contours of constant amplitude [74]. They demonstrated that phase contours are more robust with respect to smooth shading and lighting variations, and more stable with respect to small deviations from image translations. Gautama and Van Hulle improved upon this concept and proposed a method to estimate the motion field by tracking the constant phase contours. This was done by computing the phase gradient of images filtered with a bank of quadrature pair of Gabor filters [77]. A quadrature filter is a complex filter pair whose real part is related to its imaginary part via a Hilbert transform [201]².

Based on these attempts at phase-based motion estimation, Wadhwa et al. [207] propose a method that amplifies the phase variations of the coefficients of a complex-valued steerable pyramids over time. They do that by measuring the phase within each sub-band using the pairs of even and odd-phase oriented spatial filters whose outputs are the complex-valued coefficients in the steerable pyramid.

Specifically their method works as follows: They compute the local phase over time at every spatial scale and orientation of a steerable pyramid. Then, they temporally bandpass these phases to isolate specific temporal frequencies relevant to a given application and remove

²This will be expanded in Sec. 3.2.1

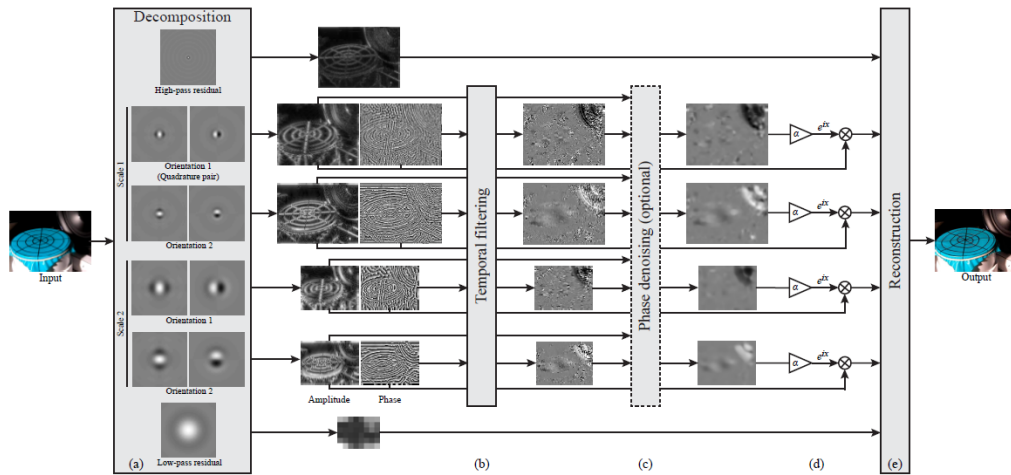


Figure 3.8.: Phase-based motion amplification framework [207]

any temporal DC component. These temporally bandpassed phases correspond to motion in different spatial scales and orientations. To synthesize magnified motion, they multiply the bandpassed phases by an amplification factor α [207]. The framework can be seen in Fig. 3.8.

The main advantages of this method is that it is able to amplify the subtle motion without amplifying the image noise (compare the results of the Eulerian magnification method in Fig. 3.9c with the ones of the phase-based method in Fig. 3.9d). However, the main disadvantage of this method comes from the complex steerable pyramids which are very overcomplete (21 times) and costly to construct, requiring either a large number of filter taps or a frequency domain construction where care must be taken to avoid spatial wrap-around artifacts. The overcompleteness and high cost of implementing the complex steerable pyramid make this method slow to compute [209].

3.2 Introduction to the Riesz Pyramid

Although Eulerian motion amplification methods seemed like promising approaches to deal with subtle motion, their tendencies to either amplify the noise or computational cost render them impractical. Fortunately, [209] implemented a phase-based video magnification method based on the Riesz pyramid that is much less over-complete than the previous method and that does not amplify the noise. However, there are some basic concepts that the reader need to know before we can breakdown this method. In this section, we will introduce the concept of analytical signal, local phase, local amplitude, the monogenic signal, the Riesz transform and its pyramid representation.

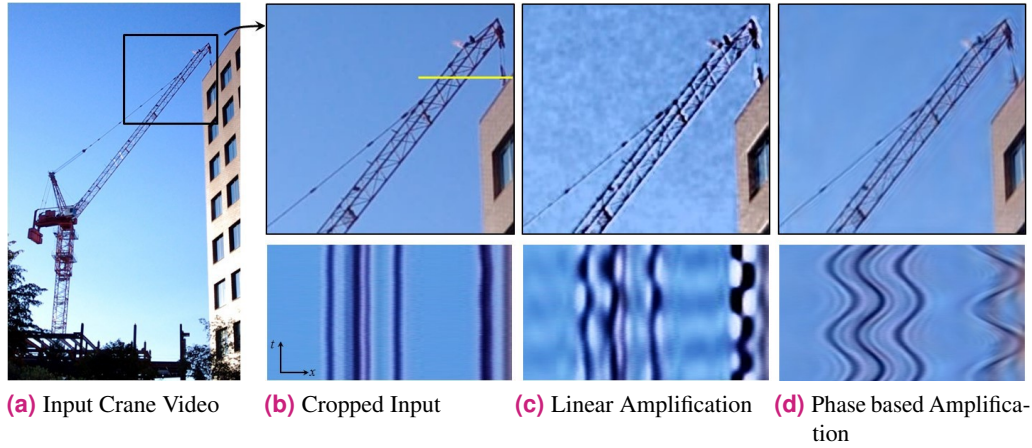


Figure 3.9.: Motion magnification results. Our input is a construction crane which is apparently still (a). The upper row corresponds to the resulting image and the lower row corresponds to the temporal evolution of a line of pixels (yellow line). We can compare the original input video (b) with the results from Linear Amplification method (c) [222] and Phase based amplification (d) [207]

3.2.1 The Analytical Signal

From a mathematical point of view, the analytical signal is a complex-valued one-dimensional (1-D) signal that has no negative frequency components. The analytic representation of a real-valued function is comprised of the original function and its Hilbert transform. That is, for a signal $f(t)$, given its Hilbert transform $\hat{f}(t)$, the analytical signal is:

$$f_A(t) = f(t) + i\hat{f}(t) \quad (3.7)$$

where

$$\hat{f}(t) = \frac{1}{\pi} \int_{-\infty}^{\infty} \frac{f(\tau)}{t - \tau} d\tau \quad (3.8)$$

Computationally one can write the Hilbert transform as the convolution of $f(t)$ with $\frac{1}{(\pi t)}$. The Hilbert transform can be considered to be a filter which simply shifts phases of all frequency components of its input by $\frac{\pi}{2}$. For example, the Hilbert transform of $\cos(\omega t)$, where $\omega > 0$, is $\cos(\omega t - \pi/2)$ or $\sin(\omega t)$. The analytic signal is important in one-dimensional (1-D) signal processing, and has been used in various applications like coding information (phase and frequency modulation), radar-based object detection, processing of seismic data, speech recognition, airfoil design, etc [72].

3.2.2 Local Amplitude and Local Phase

Let us consider the analytical signal of a simple sinusoid of some fixed amplitude A and frequency ω_0 :

$$f(t) = A \cos(\omega_0 t) \quad (3.9)$$

Applying Eq. 3.8 to Eq. 3.9 we find that the Hilbert transform of $f(t)$ is a sine wave:

$$\hat{f}(t) = A \sin(\omega_0 t) \quad (3.10)$$

Thus, the analytical signal is a complex exponential:

$$f_A(t) = A \cos(\omega_0 t) + iA \sin(\omega_0 t) \quad (3.11)$$

$$= A e^{i\omega_0 t} \quad (3.12)$$

The analytic version of our sinusoidal signal is useful to us as it allows us to extract values for the amplitude, A , and phase, $\phi = \omega_0 t$, of the signal by expressing the complex analytic signal in polar form:

$$A(t) = \sqrt{f(t)^2 + \hat{f}(t)^2} \quad (3.13)$$

$$\phi(t) = \arctan\left(\frac{\hat{f}(t)}{f(t)}\right) \quad (3.14)$$

However this result is not limited to simple sinusoids of fixed amplitude and constant frequency, but can in fact be applied to general signals. We can consider the analytic form of a signal to be produced by a complex exponential with time-varying amplitude and frequency:

$$f_A(t) = A(t) e^{i\phi(t)} \quad (3.15)$$

This gives rise to the concepts of local phase and local amplitude, which are the phase and amplitude of this polar representation. The **local amplitude** is a measure of the envelope of the signal at that point, and the **local phase** is a measure of the shape of the signal at that point [24]. Thus, a separation of these two important characteristics is achieved (some authors refer to this as the “split of identity” [72]).

3.2.3 The Monogenic Signal and the Riesz transform

Given the usefulness of the analytic signal in signal processing and interpretation, it is natural to try to extend the approach to two dimensions. Ideally we would like a way to extract a phase quadrature image from an original image, such that we can extract local phase and amplitude information. Unfortunately, this process is not straightforward in two dimensions because now different directions must be considered. Some authors have tried to generate representations that consider the Hilbert transform along a preferred or pre-selected direction in the image. However, a better representation would be isotropic, i.e. would treat all directions in the image in the same way [24].

Felsberg and Sommer proposed a generalization of the analytical signal called the **monogenic signal**, which is truly rotation-invariant [72]. Just like we need two components to

express the gradient in an image, it turns out that in the 2-D analytical signal case we need two imaginary parts (one for each axis direction) and therefore we cannot represent the monogenic signal using a complex number. Instead, the monogenic signal is represented using **quaternions**, which in simple terms, can be generalised as complex numbers with one real part and three imaginary parts (more details about quaternions and its operations can be found in Sec. B).

This new 2-D analytical signal is based on a 2-D generalization of the Hilbert transform known as the **Riesz transform**. The combination of the 2-D signal and the Riesz-transformed one forms our new 2-D analytic signal (the monogenic signal)³. The Riesz transform can be viewed as a steerable Hilbert transformer that gives a way to compute a quadrature pair of a non-oriented image sub-band that is 90 degrees phase-shifted with respect to the **dominant orientation** at every pixel. This allows us to phase-shift and translate image features only in the direction of the dominant orientation at every pixel (this is the reason why this representation can be used for motion analysis).

Let $I(\mathbf{x})$ be a 2D gray scale image of a spatial variable $\mathbf{x} = (x, y)^\top$, and let $F(\boldsymbol{\omega})$ be its frequency-domain representation found using the 2D Fourier transform, where $\boldsymbol{\omega} = (\omega_x, \omega_y)^\top$ is a two-dimensional frequency [24]. The two odd parts of the monogenic signal are:

$$F_{R_1}(\boldsymbol{\omega}) = \begin{cases} i \frac{\omega_x}{\|\boldsymbol{\omega}\|} F(\boldsymbol{\omega}), & \boldsymbol{\omega} \neq 0 \\ 0, & \boldsymbol{\omega} = 0 \end{cases} \quad (3.16)$$

$$F_{R_2}(\boldsymbol{\omega}) = \begin{cases} i \frac{\omega_y}{\|\boldsymbol{\omega}\|} F(\boldsymbol{\omega}), & \boldsymbol{\omega} \neq 0 \\ 0, & \boldsymbol{\omega} = 0 \end{cases} \quad (3.17)$$

where $R_1(\mathbf{x})$ and $R_2(\mathbf{x})$ correspond to the image domain representation of F_{R_1} and F_{R_2} . In contrast to the 1-D analytical signal case, the local phase now includes additional geometric information (orientation).

3.2.4 Implementing the Riesz Pyramid

The Riesz pyramid is the pyramid representation of the Riesz transform. The image is decomposed into multiple sub-bands, each of which corresponds to a different spatial scale, and then taking the Riesz transform of each sub-band. An ideal version of the Riesz pyramid could be built in the frequency domain using octave (or sub-octave) filters similar to the ones proposed in [207] and the frequency domain Riesz transform (Eq. 3.17). However, it requires the use of costly Fourier transforms to construct.

³This is analogous to the way the analytical signal is defined in Sec. 3.2.1

In order to make the Riesz pyramid faster, even suitable for online processing, the following steps are implemented:

- Instead of using the Fourier transform, the image is decomposed into non-oriented sub-bands using an invertible image pyramid such as the Laplacian pyramid.
- Instead of using the Riesz transform, an approximate Riesz transform is defined by two finite difference filters, which is significantly more efficient to compute. Since most of the energy from the previously processed sub-bands are concentrated in a frequency band around $\|\omega\| = \frac{\pi}{2}$, the Riesz transform can be approximated with the three tap finite difference filters $[0.5, 0, -0.5]$ and $[0.5, 0, -0.5]^T$ [209].

3.3 Riesz Pyramid Motion Magnification

In this section, we will determine how to obtain the local amplitude and the **quaternionic phase** to process subtle motions. We will describe how to obtain the Riesz coefficients, their quaternionic representation and filtering as showed in the work of the Riesz pyramid motion amplification method proposed by [209].

3.3.1 Riesz Pyramid coefficients

The Riesz pyramid is constructed by first breaking the input image into non-oriented subbands using an efficient, invertible Laplacian pyramid, and then taking an approximate Riesz transform of each band (as mentioned in Sec. 3.2.4). If a given image subband I is filtered using this method, the result is the pair of filter responses, $(R_1; R_2)$. The input I and Riesz transform $(R_1; R_2)$ together form a triplet (the monogenic signal). This can be converted to spherical coordinates to yield the local amplitude A , local orientation θ and local phase ϕ using the equations

$$\begin{aligned} I &= A \cos(\phi) \\ R_1 &= A \sin(\phi) \cos(\theta) \\ R_2 &= A \sin(\phi) \sin(\theta) \end{aligned} \tag{3.18}$$

That is, the local phase ϕ and orientation θ are angles in a spherical representation of the value (I, R_1, R_2) . The local orientation θ represent the dominant direction in the image at that point. However, the solution for Eq. 3.18 is not unique. That means that both (A, ϕ, θ) and $(A, -\phi, \theta + \pi)$ are possible solutions. This can be solved by considering

$$\phi \cos(\theta), \phi \sin(\theta) \tag{3.19}$$

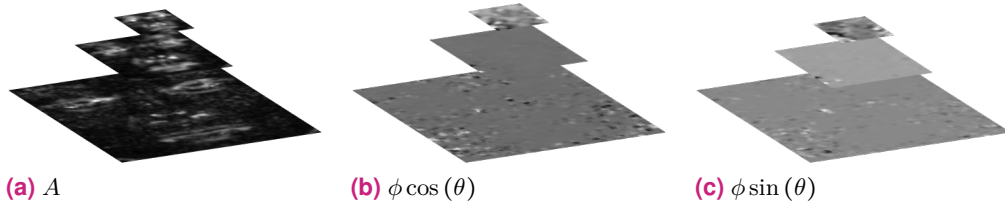


Figure 3.10.: Local amplitude and quaternionic phase of different levels of the Riesz pyramid

which are invariant to this sign ambiguity. If the Riesz pyramid coefficients are viewed as a quaternion, then Eq. 3.19 could be obtained as the quaternion logarithm of the normalized coefficient.

3.3.2 Quaternion representation of the Riesz Pyramid

The Riesz pyramid coefficient triplet $(I; R_1; R_2)$ can be represented as a quaternion \mathbf{r} with the original subband I being the real part and the two Riesz transform components $(R_1; R_2)$ being the imaginary i and j components of the quaternion⁴.

$$\mathbf{r} = I + iR_1 + jR_2 \quad (3.20)$$

The previous equation can be rewritten using Eq. 3.18 as:

$$\mathbf{r} = A \cos(\phi) + iA \sin(\phi) \cos(\theta) + jA \sin(\phi) \sin(\theta) \quad (3.21)$$

Thus, the local amplitude A and quaternionic phase defined in Eq. 3.19 are computed as:

$$\begin{aligned} A &= \|\mathbf{r}\| \\ i\phi \cos(\theta) + j\phi \sin(\theta) &= \log(\mathbf{r}/\|\mathbf{r}\|) \end{aligned} \quad (3.22)$$

The quaternionic phase is computed by applying Eq. B.13 to the specific case of the normalized Riesz pyramid coefficients. Fig. 3.10 shows the local amplitude and quaternionic phase from different levels of the Riesz pyramid applied to an image from the CASME II database [230].

3.3.3 Filtering of Quaternionic Phase

In previous *Eulerian* motion amplification papers, motions of interest were extracted and denoised by applying temporal bandpass filters [207, 222]. However, the quaternionic

⁴The reader is referred to Sec. B for more information about quaternions and their operations

phase cannot be naively filtered since it is a wrapped quantity [208]. Therefore a technique developed in [107] is used to filter a sequence of unit quaternions (by first unwrapping the quaternionic phases in time and then using a linear time invariant (LTI) filter). This technique is used to filter the Riesz pyramid coefficients at each pixel in each scale in time. It is also assumed that the local orientation is roughly constant in time and space.

Suppose at a single frame n , a single pixel (x, y) in a single scale ω_r the normalized Riesz pyramid coefficients are:

$$\mathbf{r}_n = \cos(\phi_n) + i \sin(\phi_n) \cos(\theta_n) + j \sin(\phi_n) \sin(\theta_n) \quad (3.23)$$

In the case of ordinary complex phase unwrapping, we would take the principal value of the difference between successive terms and then do a cumulative sum to give an unwrapped sequence in which the difference between two successive terms is always in the interval $(-\pi, \pi]$. We compute the principal value of the difference between two successive coefficients by dividing them and then taking the logarithm:

$$\log(\mathbf{r}_1), \log(\mathbf{r}_2 \mathbf{r}_1^{-1}), \dots, \log(\mathbf{r}_n \mathbf{r}_{n-1}^{-1}) \quad (3.24)$$

If we assume that $\theta_n = \theta + \epsilon$, i.e. the local orientation is roughly constant over time at every pixel, the k term will be close to zero. More specifically,

$$\begin{aligned} \mathbf{r}_n \mathbf{r}_{n-1}^{-1} &= \cos(\phi_n - \phi_{n-1}) \\ &+ i \sin(\phi_n - \phi_{n-1}) \cos(\theta) \\ &+ j \sin(\phi_n - \phi_{n-1}) \sin(\theta) + O(\epsilon) \end{aligned} \quad (3.25)$$

by ignoring the $O(\epsilon)$ term, the logarithm is

$$i([\phi_n - \phi_{n-1}]) \cos(\theta) + j([\phi_n - \phi_{n-1}]) \sin(\theta) \quad (3.26)$$

The second step is to perform a cumulative sum of (Eq. 3.24)

$$\phi_1 \mathbf{u}, (\phi_1 + [\phi_2 - \phi_1]) \mathbf{u}, \dots, \left(\phi_1 + \sum_{l=2}^n [\phi_l - \phi_{l-1}] \right) \mathbf{u} \quad (3.27)$$

where $\mathbf{u} = i \cos \theta + j \sin \theta$. If we let $\phi'_n = \phi_1 + \sum_{l=2}^n [\phi_l - \phi_{l-1}]$ the series can be written as:

$$i \phi'_n \cos(\theta) + j \phi'_n \sin(\theta) \quad (3.28)$$

Afterwards we can isolate motions of interest in the quaternionic phase signal using an LTI filter. Furthermore, the signal-to-noise ratio (SNR) of the phase signal can be increased by spatially denoising each frame with an amplitude-weighted spatial blur with Gaussian

Kernel K_ρ with standard deviation ρ on the i and j components of the temporally filtered signal.

$$i \frac{A\phi' \cos(\theta) * K_\rho}{A * K_\rho} + j \frac{A\phi' \sin(\theta) * K_\rho}{A * K_\rho} \quad (3.29)$$

Assuming that the orientation does not change substantially in the support of K_ρ , then $\cos(\theta)$ and $\sin(\theta)$ can be moved outside of the convolution in Eq. 3.29 to get:

$$i \cos(\theta)\phi'' + j \sin(\theta)\phi'' \quad (3.30)$$

where

$$\phi'' = \frac{A\phi' * K_\rho}{A * K_\rho} \quad (3.31)$$

where (3.30) is the filtered quaternionic phase obtained for each pixel of each subband in each frame.

3.3.4 Amplification

First, the filtered quaternionic phase (Eq. 3.30) is multiplied by a magnification factor (α), then we perform a quaternion exponentiation on it to produce a unit quaternion:

$$\cos(\alpha\phi'') + i \sin(\alpha\phi'') \cos(\theta) + j \sin(\alpha\phi'') \sin(\theta) \quad (3.32)$$

We then multiply this unit quaternion by the original coefficient $I + iR_1 + jR_2$ in the Riesz pyramid. We only need the real part of the result, which by Eq. B.4 is equal to:

$$I \cos(\alpha\phi'') - R_1 \sin(\alpha\phi'') \cos(\theta) - R_2 \sin(\alpha\phi'') \sin(\theta) \quad (3.33)$$

This gives the coefficients of a real Laplacian-like pyramid for every frame, in which the motions have been magnified, which can then be collapsed to produce a motion magnified video [209].

3.4 Subtle Motion Analysis

In the previous section, we were able to explore the method to process motion which was used for [209] for motion magnification. Although it is a powerful method, it limits itself to process the video without considering when a subtle motion is taking place. Nevertheless, our examination of the resulting quaternionic phase and local amplitude shows potential for motion detection and estimation. Thus, we propose a method that is able to analyse subtle motion and spot the exact moment when it appears on a video using the quaternionic representation of the Riesz pyramid. Our proposed algorithm goes as follows: first we use the Riesz Pyramid to calculate the amplitude and quaternionic phase of the images (Fig. 3.11b).

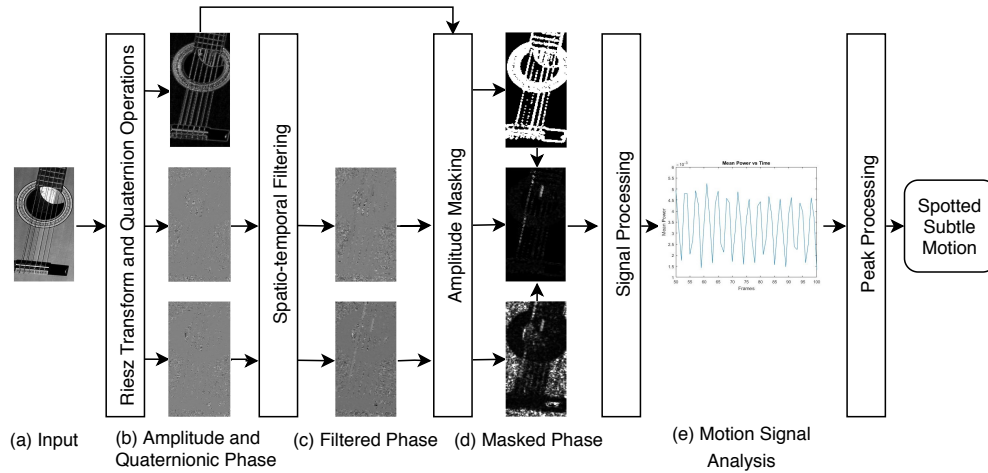


Figure 3.11.: Subtle motion analysis framework.

Secondly, we implement a proper spatio-temporal filtering scheme which can enhance motions of interest without producing delays or undesired artifacts (Fig. 3.11c). Thirdly, we isolate areas of potential subtle motion based on the computed amplitude (Fig. 3.11d). Finally, we measure the dissimilarities of quaternionic phases over time, average them, and transform them into a 1-D signal (Fig. 3.11e), which is used to estimate the moment when the subtle motion is taking place.

3.4.1 Temporal Filtering Considerations

For an image sequence of N frames we perform the process described in Sec.3.2 and Sec.3.3.2 for each frame $n \in N$. However, not all levels of the pyramid are able to provide useful information about the subtle motion. Thus, after processing our video using different pyramid levels, we select the one that shows more subtle changes. We then obtain both local amplitude A_n and quaternionic phase $(\phi_n \cos(\theta), \phi_n \sin(\theta))$. We apply the process described in Sec. 3.3.3 to obtain the unwrapped quaternionic phase (Eq. 3.28). However the temporal filtering scheme proposed by [209] might not be appropriate for the problem at hand and we should consider a different approach.

The previous work in eulerian motion magnification have given their users freedom to choose any temporal filtering method available. However, since we require to pinpoint the exact moment when subtle motion is detected we cannot use traditional causal filters which may delay the signal response (Fig. 3.13c). Therefore, we propose to use a digital

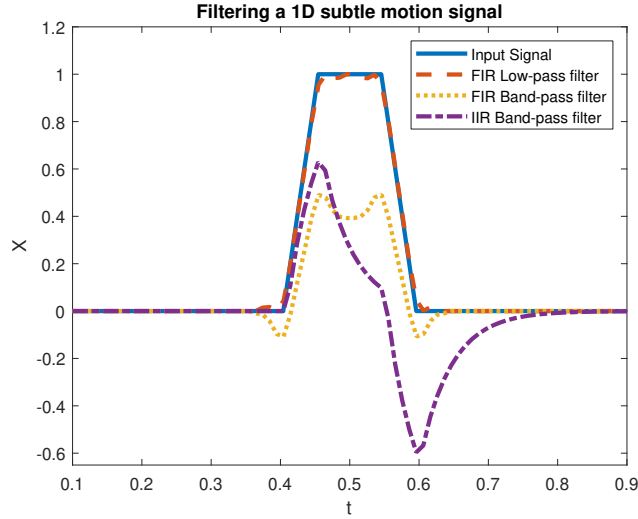


Figure 3.12.: Non-periodic signal filtering representation

non-causal symmetrical zero-phase finite impulse response (FIR) filter⁵. Thus, we filter the unwrapped quaternionic phase (Eq. 3.28):

$$\phi'_n \mathbf{u} = b_0 \phi'_n \mathbf{u} + \sum_{k=1}^p b_k (\phi'_{n+k} \mathbf{u} + \phi'_{n-k} \mathbf{u}) \quad (3.34)$$

where \mathbf{u} represents $(\cos(\theta), \sin(\theta))$, p is an even number (to maintain the symmetry) and b_k is a coefficient of a FIR filter of length $2p + 1$. One limitation of this method is that non-causal filters requires to use the previous and following p frames from the current frame (therefore for online applications there must be a delay of at least p frames).

Another element to consider is that *Eulerian* amplification methods are tailored for a particular task. These methods aim to amplify subtle periodical movements (such as human breathing, the vibration of an engine, the oscillations of a guitar string, etc) by temporally band-passing some potential movements and amplifying them. However, these methods do not consider subtle non periodical movements (such as blinking or facial MEs) and whether the filtering scheme delays the signal.

Let's consider the case of non-periodic subtle motion as a 1D discrete signal sampled at 100 Hz, which has a small displacement during 60 ms and then after some time comes back to the original state (the blue line in Fig. 3.12). If we apply a IIR Butterworth band-pass causal filter to reduce its noise as in [209] it will delay the signal (the purple line in Fig. 3.12). Furthermore, if we use a non-causal band-pass filter, it won't delay the signal but some large oscillations (compared to the gain of the signal) will appear near the beginning and the end of the subtle motion (yellow line in Fig. 3.12) as stated by Gibbs phenomenon but not if we apply a low pass filter (the orange line in Fig. 3.12).

⁵It is possible to use a causal filter instead, as long as the delay added by this type of filter is compensated

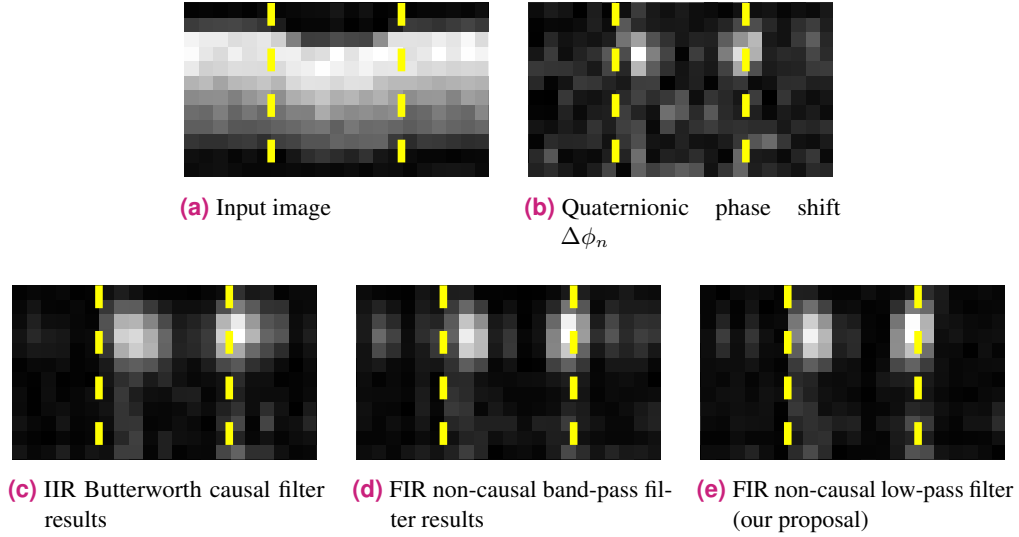


Figure 3.13.: A comparison of different filter responses for subtle motion detection.

After applying the new temporal filter, we apply the same spatial filtering function (Eq. 3.29) to obtain ϕ'' (Eq. 3.30 and Eq. 3.31). Finally, since we are aiming to detect any significant quaternionic phase shifts between frames and to compensate for the cumulative sum made in (Eq. 3.27), we calculate the difference of two consecutive filtered quaternionic phases:

$$\Delta\phi_n \mathbf{u} = \phi''_n \mathbf{u} - \phi''_{n-1} \mathbf{u} \quad (3.35)$$

where $\mathbf{u} = i \cos \theta + j \sin \theta$.

3.4.2 Amplitude Masking

The first step is to simplify the quaternionic phase shift by discarding the orientation and calculate the euclidean norm of the phase thus:

$$|\Delta\phi_n| = \sqrt{(\Delta\phi_n \sin \theta)^2 + (\Delta\phi_n \cos \theta)^2} \quad (3.36)$$

One thing to consider before trying to detect subtle motions is the problem of image noise. Assuming the general case of two static images corrupted with some level of additive Gaussian noise, their quaternionic phase difference would be non-zero ($|\Delta\phi_n| > 0$) even after the phase SNR is improved by ways of spatial and temporal filtering (Sec. 3.3.3). We have observed that the $\Delta\phi_n$ values could have a high variance in areas where local amplitude A has a relative low value regardless of the presence of motion (use Fig. 3.14b and Fig. 3.14c for comparison). Considering that the motion phase-signal in regions of low amplitude is not meaningful [207] we decide to isolate these areas using a threshold of validation computed from the local amplitude. However, since the scale of local amplitude

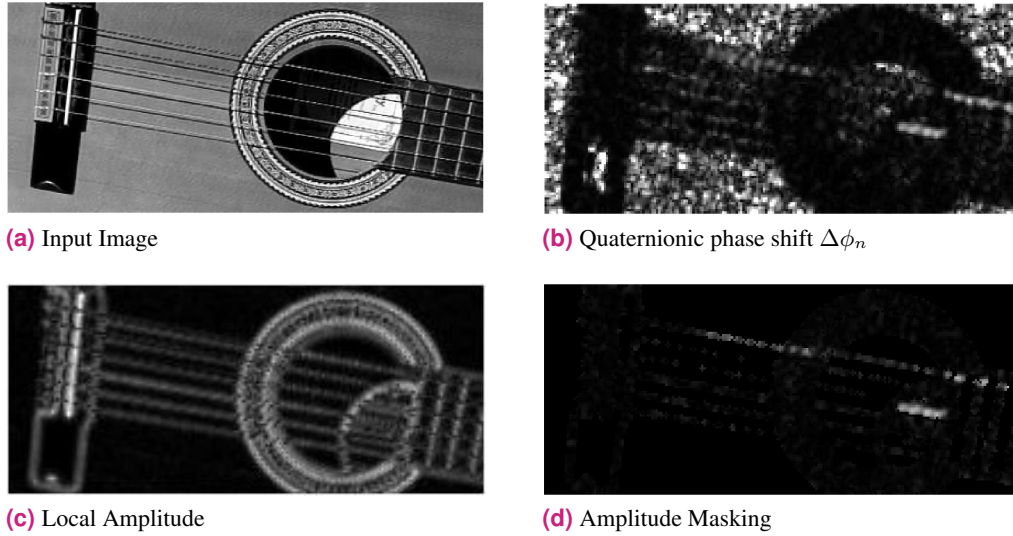


Figure 3.14.: Isolating relevant quaternionic phase using the amplitude mask.

might vary from subject to subject (some videos might have objects with stronger edges compared to others) we need to normalize the local amplitude before we can threshold it:

$$M = \begin{cases} 1 & \text{if } \beta \leq \frac{A_n}{A_q} \\ 0 & \text{if } \beta > \frac{A_n}{A_q} \end{cases} \quad (3.37)$$

where A_n is the calculated local amplitude of the image at frame n , A_q is the 95-percentile of the empirical distribution of the amplitudes along the video and β is a threshold selected by the user (see Sec. 3.5.2). The mask can be further refined using morphological opening. Finally, we mask the phase norm ($\Delta\phi_n$) with M (as seen in Fig. 3.14d). By masking the areas of low amplitude we have effectively selected the regions in which subtle motion can be detected.

3.4.3 Motion Spotting

Although the amplitude masking step aims to discard any area that could contribute with noisy data, some spurious pixels might get through this step. Thus, for each masked frame $\Delta\phi_n$, we select the values smaller than the 90-percentile. This is done to avoid outliers caused by noisy pixels which do not represent the subtle motion. From the selected pixels we calculate the average power :

$$P_n = \frac{1}{L} \sum_{l=0}^{L-1} |\Delta\phi_{n,l}|^2 \quad (3.38)$$

where l is the index of the selected pixels and L is the total number of selected pixels. P_n is a one-dimensional signal which peaks or local maxima represents changes in the image sequence.

In order to distinguish relevant peaks (subtle motions) from local magnitude variations and background noise, we use a method to contrast the differences of P_n proposed by [112]. This method compares the differences of P_n within a specified interval. Since subtle motions might take more than 2 consecutive frames, we analyse micro-intervals of K frames (an odd number bigger than 2). Then, for each current frame value, we subtract the average of the k -th frame value before the current frame and the k -th frame value after the current frame, where

$$k = \frac{1}{2}(K - 1) \quad (3.39)$$

Thus, for the n -th value in the contrasted difference vector $C(\phi)$ is calculated by:

$$C_n = P_n - \frac{1}{2}(P_{n-k} + P_{n+k}) \quad (3.40)$$

Finally, we select from C_n the peaks or local maxima that go over a threshold T and that are separated by at least K frames. The threshold is calculated as:

$$T = C_{median} + p \times (C_{max} - C_{median}) \quad (3.41)$$

where C_{median} and C_{max} are the median and maximum value of C_n for the whole video and p is a percentage parameter in the range $[0, 1]$.

3.5 Results

Our proposed method allows a user to analyze subtle motions in videos. Although, the Eulerian amplification methods are able to reveal imperceptible phenomena not previously visualized on video, our method goes further and is able to quantify this motion. Thus, in the following section, we decide to show some potential applications using the videos provided in the supplemental material of [209, 207].

3.5.1 Preliminary Evaluation

We select a video of a baby sleeping under a blanket in a cradle that, when it is magnified, amplifies the subtle movements of the baby breathing (see Fig. 3.15a). We decided to test our method using the same filtering parameters suggested by the supplemental material of [209, 207]. For the Riesz transform step we select the second level of the pyramid (see Fig. 3.6). We design a non-causal FIR bandpass temporal filter of order 10 with passband between 0.04 and 0.4 Hz. We use a Gaussian Kernel K_ρ with standard deviation $\rho = 2$ for spatial

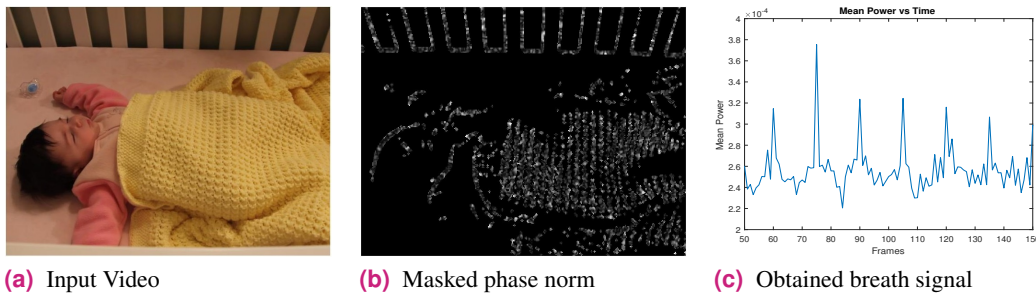


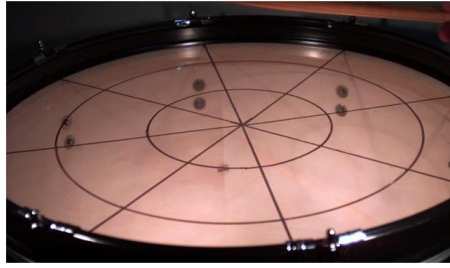
Figure 3.15.: Subtle motion detection for breathing measurement for a baby monitoring system.

filtering. For the amplitude masking step we select a threshold $\beta = 0.3$ (see Fig. 3.15b). The segmented areas are transformed into a 1-D signal using mean power (Eq. 3.38). As we can see in Fig. 3.15c, the local maxima in the signal correspond to the moment when the breathing motion is at its peak, thus we can estimate the breathing patten of the baby.

We also select a video of a drum, for which the motion amplification method, magnifies its vibration (see Fig. 3.16a). For this example we decided to design a filtering using a passband between 60 and 90 Hz instead of the narrow passband between 74 and 78 suggested by [207]. This video was recorded using a high speed camera at 1900 fps. For the Riesz transform step we select the third level of the pyramid. The following parameters are the ones suggested in the supplemental material of [209, 207]. We design a non-causal FIR bandpass temporal filter of order 44 with passband between 60 and 90 Hz. We use a Gaussian Kernel K_ρ with standard deviation $\rho = 2$ for spatial filtering. For the amplitude masking step we select a threshold $\beta = 0.15$ (see Fig. 3.16b). The segmented areas are transformed into a 1-D signal using mean power (Eq. 3.38). As we can see in Fig. 3.16c, the detected vibration behaves like a combination of sinusoidal waves (which is expected since the drum is emitting acoustic waves). We can further analyze the signal spectrum using the fast Fourier transform. As we can see from Fig. 3.16d, there are some frequency peaks outside the narrow spectrum considered by [207] which could better characterize the frequency of the drum vibration.

3.5.2 Spotting Experiment

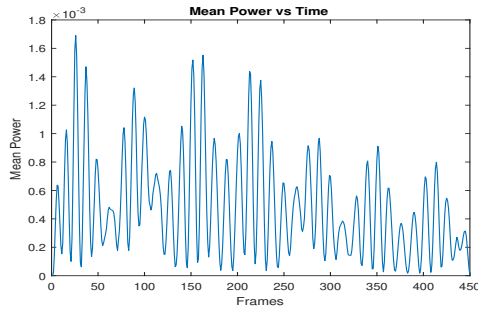
Although, the examples presented in the previous section show the potential of our method for subtle motion analysis, we would also like to measure its spotting accuracy and robustness. Thus, we have designed an experiment to compare the performance of our method compared with methods in the state of the art under different levels of noise.



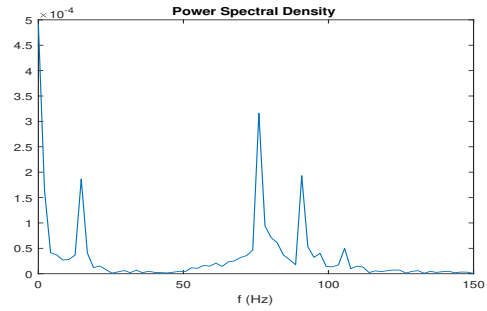
(a) Input Video



(b) Masked phase norm



(c) Obtained sinusoidal signal



(d) Magnitude spectrum

Figure 3.16.: Subtle motion spectral analysis of a vibrating drum.

Database

To the extent of our knowledge, there is not a public labelled subtle motion database available. Thus we decided to create our own database consisting of image sequences with subtle motions. We create 18 image sequences using real and artificially generated images which elements are for the most part static with the exception of one or two instances of subtle motion. The subtle motion has been simulated by translating either an object in the image or the whole image by one pixel per frame during two or three frames. We also label the time when the subtle motion starts (onset) and ends (offset) (For more details, we refer the reader to Sec. C).

Method Comparison

We decide to compare our proposed method against other classical approaches for motion detection. First, the image is divided into a grid of equal-sized blocks. Then, we extract information from each block using three different feature descriptors: local binary pattern (LBP) [149], histogram of oriented gradients (HOG) [43] and optical flow (OF) using the Lucas-Kanade method [13]. Then we use the feature difference analysis method proposed by [112] to compare the differences of the appearance-based features for each block within a specified interval. The difference between HOG and LBP histograms is calculated using the Chi-Squared (χ^2) distance. Then we sort the feature difference values from the blocks

and calculate the mean of the greatest values that surpass the 80–percentile. Finally, we use the method to contrast the differences and the peak detection discussed in Sec. 3.4.3 (Eq. 3.40 and 3.41 respectively).

Parameter Selection

We select the parameters for the spotting methods which will work in all videos without added noise. For our spotting method we select the second level of the Riesz pyramid. We design a FIR non-causal low-pass temporal filter with cut-off frequency of 30 Hz, corresponding to a filter of order 10. We use a Gaussian Kernel K_ρ with standard deviation $\rho = 2$ for spatial filtering. For the amplitude masking step we select a threshold $\beta = 0.25$.

For the LBP method we divide the image into a grid of 6×6 equal-sized blocks. The LBP descriptor has a radius of 3 pixels with 16 neighbours. For the HOG method we use the function *extractHOGFeatures* from Matlab. For each block produced by the function there are 2×2 cells of $[8, 8]$ pixels. For the OF method, we divide the image into a grid of 6×6 and compute the flow’s amplitude⁶.

Image Noise Measurement

Considering that subtle motions have low amplitude and, in some cases, they could be mistaken for noise, we decide to test the robustness of the motion detection methods in the presence of different levels of noise. We choose to test the videos under Gaussian additive noise and Salt and Pepper noise. In order to do a standard measurement of the noise among the different videos, we measure it using the peak signal-to-noise ratio (PSNR). PSNR is defined by the mean square error (MSE). Given a noise-free $u \times v$ monochrome image I and the image with added noise K , MSE is defined as:

$$MSE = \frac{1}{uv} \sum_{i=0}^{u-1} \sum_{j=0}^{v-1} [I(i, j) - K(i, j)]^2 \quad (3.42)$$

The PSNR (in dB) is defined as:

$$PSNR = 10 \cdot \log_{10} \left(\frac{MAX_I}{\sqrt{MSE}} \right) \quad (3.43)$$

where MAX_I is the maximum possible pixel value of the image.

⁶We don’t extract the orientation since it doesn’t provide any important information about the instant when a subtle motion takes place.

Evaluation Methodology

The first step is, for each video, to add a specific level of noise. Then we evaluate the methods accuracy by comparing all the detected peak frames in each method with the ground truth labels from each video in order to tell whether they are true or false positive subtle motions. Within a certain threshold level ($p = 0.75$ see Eq. 3.41), if one spotted peak is located within the frame range of $[\text{ONSET}, \text{OFFSET}]$ of a labelled subtle motion video, the detected sequence is considered as one true positive subtle motion. Otherwise a penalization of a possible detected subtle motion (ψ frames) is counted as false positive. Since the noise is added randomly we repeat the test and measure the methods' accuracy 20 times.

We define the true positive rate (TPR) as the percentage of frames of correctly spotted subtle motion divided by the total number of ground truth subtle motion frames in the database. The false positive rate (FPR) is calculated as the percentage of incorrectly spotted frames divided by the total number of non-subtle motion frames from all the image sequences. We evaluate the performance of the subtle motion detection methods by tracing curves with TPR and FPR as the y axis and PSNR as the x axis.

Experimental Results

The spotting results under different levels of Gaussian noise are presented in Fig. 3.17. Our method has shown to have an equal or higher TPR in the presence of most levels of Gaussian noise compared to the other methods except when the PSNR is between 20 and 23 dB in which the OF method has a higher TPR. Similarly, the FPR of our method is equal or lower in most levels of Gaussian Noise except when the PSNR is between 20 and 23 dB. The LBP and HOG method had a lower TPR and a higher FPR compared to our method and the OF method.

The spotting results under different levels of density of Salt and Pepper noise are presented in Fig. 3.18. Our method has shown to have an equal or higher TPR in the presence of most levels of Salt and Pepper noise compared to the other methods except when the PSNR is between 19 and 26 dB in which the OF method has a higher TPR. Similarly, the FPR of our method is equal or lower in most levels of Salt and Pepper Noise except when the PSNR is between 19 and 26 dB. The LBP and HOG method had a lower TPR and a higher FPR compared to our method and the OF method. However, the LBP method seems to have a better performance between 18 and 23 dB than HOG.

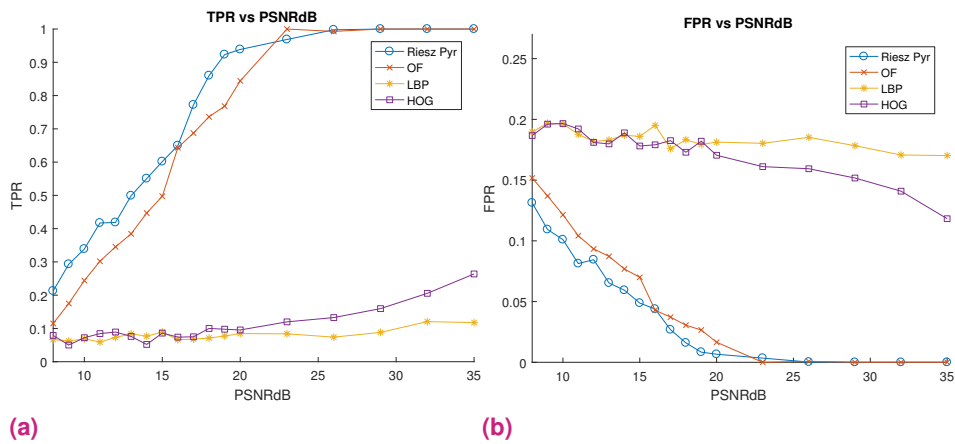


Figure 3.17.: Performance curves of different subtle motion spotting techniques in presence of different levels of Gaussian noise

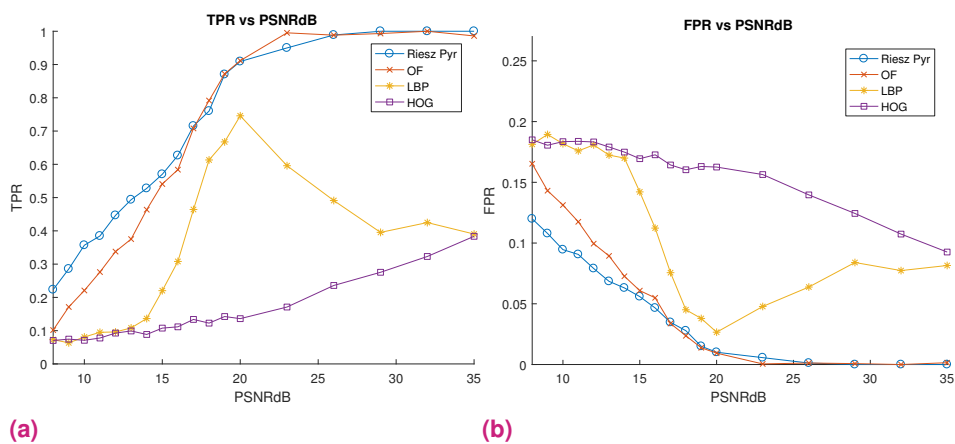


Figure 3.18.: Performance curves of different subtle motion spotting techniques in presence of different levels of Salt and Pepper noise

Discussion

The results in the previous section show that our method has, in general, a better spotting accuracy compared to the other methods tested. However, a closer examination of our database has shown that our method performs better in images with overall strong edges and its performance decreased in images with blurry edges. A possible explanation for this behaviour is that our amplitude masking method (see Sec. 3.4.2) aims to bypass areas of higher amplitude in which the phase noise is lower and discard areas of low amplitude in which the phase noise is higher. However, in images with blurry edges, the computed amplitude will be low all over the image and our system will end up bypassing areas of higher noise, thus, compromising our method's accuracy.

The performance of the OF method is comparable to our method. Since the Lucas-Kanade optical flow method was formulated under the temporal persistence assumption (motion remains small from frame to frame), it seems like an appropriate method for describing subtle motion [99]. However, in videos with color gradients and certain levels of noise, the optical flow accuracy is decreased. During the development of this experiment, it was initially suggested to use a more accurate OF method such as TV-L1, however the required processing time increased dramatically with these methods (specially since each sequence was analysed 20 times) making them impractical for our experimental design (see more details in Sec. 3.5.3).

The accuracy of the HOG method was low because image gradients are sensible to noise. The accuracy of the LBP method was also low because LBP is sensitive to noise and sometimes may classify two or more different patterns falsely to the same class [170]. However, the overall results suggest that our method better describes subtle motion than LBP and HOG descriptors.

3.5.3 Running time comparison

We compare the running time of the different methods used in the previous section. We select some image sequences from our database, and add enough Gaussian noise so their PSNR is 20 dB. We run all the implementations in Matlab using the same cpu. Furthermore, we include an implementation of TV-L1 optical flow for comparison purposes. The Lucas-Kanade optical flow and HOG method were implemented using built-in functions from the Matlab library. The LBP method was implemented from [147]. The TV-L1 optical flow method was implemented from [42, 236]. We measure the time to run a function by calling the specified function multiple times, and returns the median of the measurements (to avoid measuring first-time cost)⁷.

⁷We use the Matlab function *timeit*

Image Sequence	Resolution	Riesz Pyramid	Lucas-Kanade OF	TV-L1 OF	HOG	LBP
Airplane	$314 \times 626 \times 50$	7.01	0.77	294.67	0.69	100.77
Checkerboard	$198 \times 198 \times 50$	1.08	0.09	46.63	0.15	34.92
Cube	$200 \times 200 \times 50$	1.08	0.09	45.45	0.15	35.11
Kanagawa	$370 \times 550 \times 50$	5.72	0.69	319.36	0.73	100.13
Sweets	$300 \times 400 \times 50$	3.21	0.43	170.86	0.53	93.46

Table 3.1.: Running times (in seconds) of comparable MATLAB implementations of subtle motion spotting methods

Table. 3.1 shows the running times for the different implementations. We can see that the fastest methods, HOG and Lucas-Kanade optical flow, are the ones using built-in functions which speed-up their performances. Comparing the other implementations we can see that our method is 14 times faster than the LBP method and 44 times faster than the TV-L1 optical flow method. The results show that by decomposing the image using a Laplacian pyramid representation and using an approximate Riesz transform in our method, as proposed in Sec. 3.2.4, have yielded into a fast method.

3.6 Chapter Conclusions

In this chapter, we presented a subtle motion analysis and spotting method based on the quaternionic representation of the monogenic signal. Our main contributions are:

- A new temporal filtering scheme which can enhance motions of interest without producing delays or undesired artifacts.
- A method to isolate regions of interest where subtle motion might take place and mask noisy areas using the image amplitude.
- A method to transform the quaternionic phase into a 1-D signal which is used for temporal and frequency analysis of subtle motions.

After testing our method using our own database under different levels of Gaussian additive noise and salt and pepper noise, we can conclude that our method surpasses other state of the art methods. Due to the unavailability of a public labelled subtle motion database we had to test our experiments in a rather limited dataset. Further tests will require us to create or find a more complete database in order to obtain more statistically significant results. We also illustrated the power of our subtle motion analysis method by briefly presenting a couple of potential real-life applications. What's more, the fast performance of our method seem to suggest that it could potentially be used in the future for online applications. Although, in

this chapter we only tested one level of the Riesz pyramid, in the following chapter we will test and compare the results from different levels.

The quaternionic representation of phase and orientation from the Riesz monogenic signal has shown to be a powerful tool that could potentially be exploited in the future for more focused applications in the fields of modal analysis, biomedical signals processing, among other areas in which subtle motion analysis, spotting and modelling are required. Thus, we propose to use this tool to implement both facial micro-expression spotting and recognition methods.

Micro-Expression Spotting using the Riesz Pyramid

Although the study of automatic emotion recognition based on facial MEs have gained momentum in the last couple of years, much of this work has focused on classifying emotions. However, most of the research work use temporally and manually segmented videos (that are known to contain MEs).

Considering that a lot of real-life applications require to detect when an ME takes place, spotting MEs becomes a primary step for a fully automated facial expression recognition (AFER) system. We have already presented some of the current methods for ME spotting in Sec. 2.4. However, this remains to be a challenging task because MEs are quick and subtle facial movements of low-spatial amplitude which are very difficult to detect by the human eye.

Another possible approach would be to use *Eulerian* motion amplification techniques [207, 222] in order to enhance the spotting process. These techniques have already been used in the past, as a pre-processing step, to enhance the ME recognition [152, 153]. However, it has also shown to degrade the ME spotting accuracy [112].

As demonstrated in Sec. 3, the quaternionic representation of phase and orientation from the Riesz monogenic signal has shown to be a powerful tool for subtle motion analysis that could potentially be exploited for ME spotting. Thus, in this chapter, we propose a method which is able to spot MEs in a video by analysing the phase variations between frames obtained from the Riesz Pyramid. The proposed method does not require training or pre-labelling of the videos.

This chapter is organized as follows: Sec. 4.1 describes the face registration step, that is, how from an input video the face is detected, tracked and cropped. Sec. 4.2 explains how we adapt Riesz pyramid motion analysis (Sec. 3.2) and amplitude masking (Sec. 3.4.2) for ME spotting. Sec. 4.3 illustrates how the quaternionic phase is transformed into a series of 1-D signals that are used to spot the MEs. Sec. 4.4 shows our experiments, results and discussion. Finally, Sec. 4.5 presents our conclusions. Our proposed framework is summarized in Fig. 4.1.

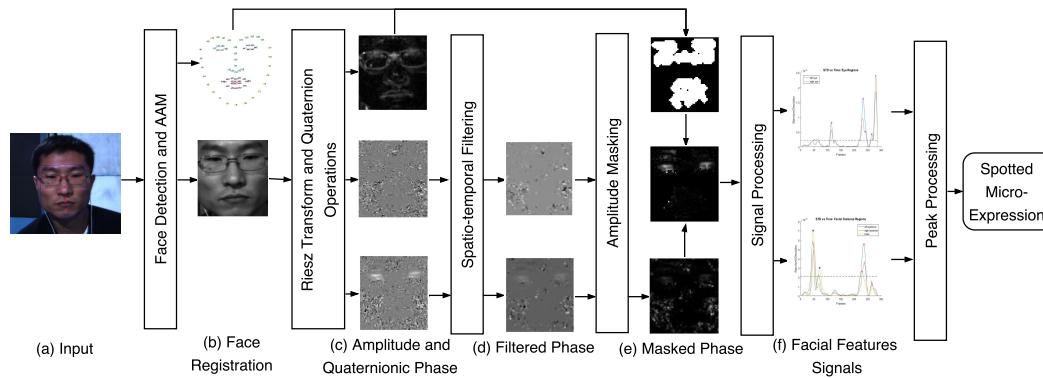


Figure 4.1.: ME spotting framework.

4.1 Face Registration

In this section, we talk in detail how we detect a face in a video, how we localize a series of landmark points, how we track these points overtime and how we select certain areas of interest for feature extraction.

4.1.1 Face Detection

We used the cascade detector proposed by Viola and Jones [206] to detect the face area in the image. Then we divide the face area into specific regions of interest in order to constraint the search of important facial features (eyes, nose and mouth). In order to avoid having multiple detected objects the algorithm merges the detected objects that meet a certain “detection threshold” to produce a singular bounding box around the target object. The detection threshold refers to how many times the candidate object was detected during the multi-scale detection phase. This threshold is made tunable in order to detect the facial features no matter what are the illumination conditions.

The Viola and Jones algorithm provides us with the bounding box of each detected object. In theory we could assume that the centroid of the bounding box corresponds to the approximate center of the detected facial feature. However, as we can see in Fig. 4.2 (the yellow marks represent the centroids), this is not really the case for the mouth area (Sometimes the detected area centroid is lower than the actual centroid). We propose to correct the centroid by detecting the line between the closed lips¹ (the red mark is the correction we implement for the mouth).

Mouth Detection Correction : First we calculate the laplacian of the mouth image (Fig. 4.3a) and detect which pixels have the strongest values (Fig. 4.3b). Then we discard the pixels with a low outliers by calculating the inter-quantile range of the candidate values

¹We assume that the subjects in the video start with a neutral expression and mouth closed

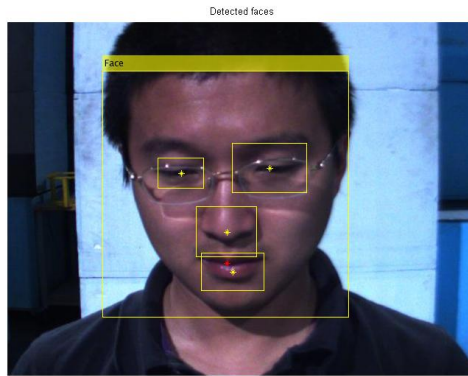


Figure 4.2.: Facial features bounding boxes and their centroids.

(in the example in Fig. 4.3d the outlier is the yellow mark). The corrected centroid becomes the mean vertical position of the accepted line pixels (red mark in Fig. 4.2).

4.1.2 Facial Landmarks Fitting

We used the active appearance model proposed by [204] to locate a series of fiducial points. The model consists of 68 fiducial points that delineate the edge of the face, eyebrows, eyes, nose and mouth (Fig. 4.4a). Since a good initialization of the AAM becomes critical for an effective tracking, we used the corrected centroids of the detected facial features bounding boxes described in the previous section to translate, scale and rotate the model to fit the detected face (see Fig. 4.4b). In order to make the model fitting faster the face image is initially re-sized to half of its original size and then we start to update the model until it achieves some convergence criteria (the average difference between the previous and currently updated model becomes minimal or/and after a maximal number of iterations have taken place). Then, the image and model are transformed to their original size in order to fit the model in a more precise way (until it reaches the level of convergence previously discussed).

4.1.3 Face Tracking

Once we fit the facial landmarks we must track the facial features along the video sequence². Initially, we selected certain facial landmarks which won't move during facial expressions (we selected the inner corner of the eyes and the lower point of the nose between the nostrils). Then, we tracked these points using the Kanade-Lucas-Tomasi (KLT) algorithm [199] and used them to realign the face over time.

²For our approach we assume that the head pose of the subject recorded in the video will not drastically change from a frontal pose and that any change would be either slight rotations or translations in the X or Y plane but not in the Z plane (we assume that the subject won't go closer or farther away from the camera during the video recording).

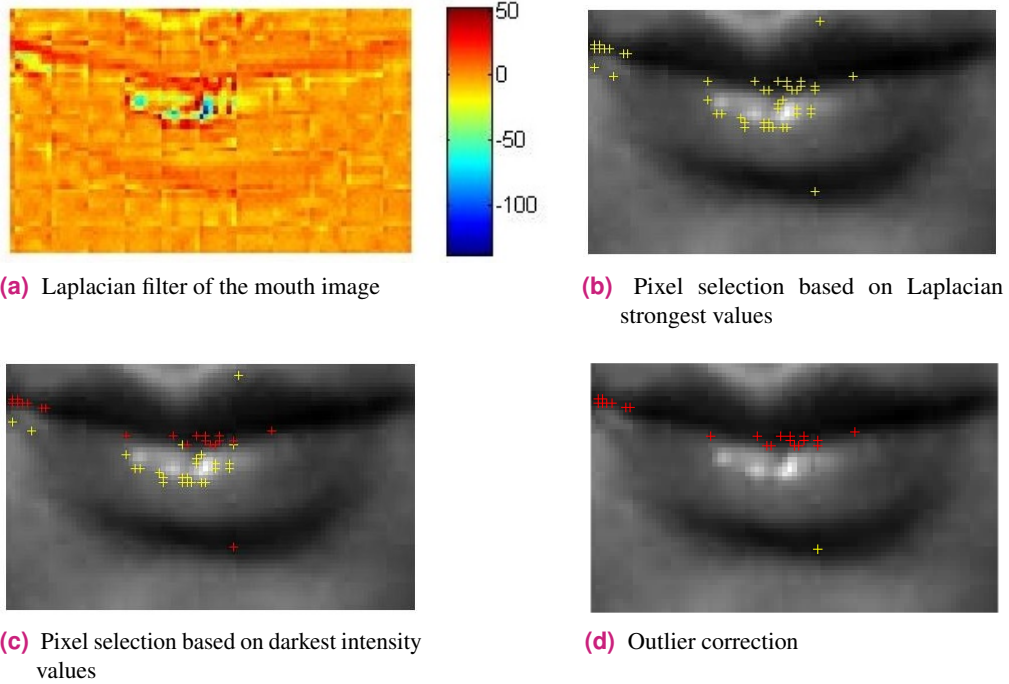


Figure 4.3.: Detecting the edge between lips.

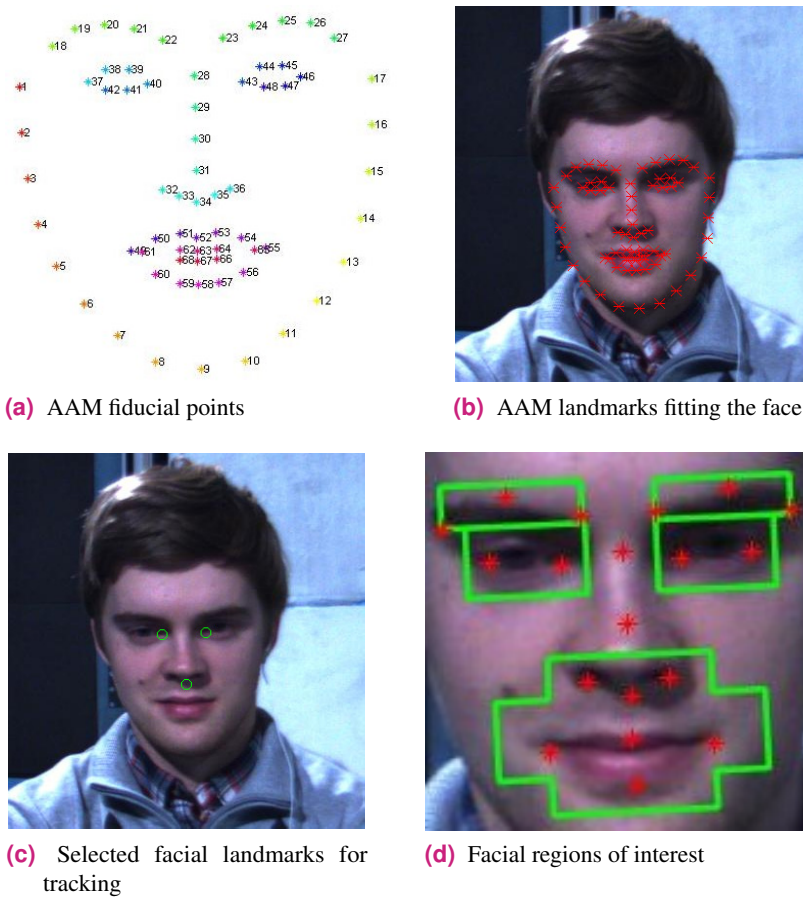


Figure 4.4.: Face ROI localization and tracking

4.1.4 Regions of Interest

We opted to create a series of local ROIs (see Sec. 2.2.3) based on the FACS and the fiducial points tracked in the AAM. We selected 5 ROIs (delimited within green lines in Fig. 4.4d): left eyebrow, right eyebrow, left eye, right eye and mouth (this region also includes part of the cheeks near the mouth corners and the lower nose area). The size and orientation of the ROIs depend on the orientation and distance between the facial fiducial points (as shown by the red marks in Fig. 4.4d). Since the recorded faces might vary in orientation, we create a series of binary mask that only crops the image inside the ROIs. Finally, we use these ROIs to crop the aligned facial region of interest for each frame in the video.

4.2 Riesz Transform and Filtering

For the sequence of cropped images obtained in 4.1.4 of N frames, we transform the images into a series of Riesz pyramids, extract the local amplitude and quaternionic phase and filter them (see the details in Sec. 3.3.1, Sec. 3.3.2 and Sec. 3.3.3) for each frame $n \in [1, \dots, N]$. However, depending on the spatial resolution of the cropped faces and the speed of the image acquisition system, the local phase computed from certain frequency sub-bands will contribute more to the construction of a certain motion compared to the ones from other levels. In other words, not all the levels of the pyramid are able to provide useful information about the subtle motion. For instance, this is evidenced in [152], which propose to train an automatic frequency band predictor to adaptively select the most discriminative frequency band to amplify as ME recognition preprocessing step. Thus, we must test the quaternionic phase from different sub-bands (pyramid level) and select the level which better represents the subtle motion we want to detect (see Sec. 4.4.3 for more details).

We then obtain both local amplitude A_n and quaternionic phase $(\phi_n \cos(\theta), \phi_n \sin(\theta))$ from the selected sub-band. We apply the process described in Sec. 3.3.2 to obtain the filtered quaternionic phase ϕ . However, we need to adapt our filtering scheme to isolate motions of interest (MEs). Based on the work of [232], we know that an ME can last from 170 to 500 ms. However, some MEs have a fast onset and a slow offset, so our filtering scheme should be designed to consider certain events as MEs even if only the onset is detectable. Taking that in mind, we consider the cutoff frequency for a low-pass filter should be 10 Hz (we assume that the shortest ME will last more than 100 ms). Furthermore, considering that we know the maximum duration of a ME, we explore the possibility to use a band-pass filter (from 2 to 10 Hz) to filter the quaternionic phase.

From the previous step we obtain the filtered quaternionic phase. However, since we are aiming to detect any significant quaternionic phase shifts between frames and to compensate

for the cumulative sum done in the quaternionic phase unwrapping [208], we calculate the difference of two consecutive filtered quaternionic phases:

$$\Delta\phi_n \mathbf{u} = \phi_n \mathbf{u} - \phi_{n-1} \mathbf{u} \quad (4.1)$$

where $\mathbf{u} = i \cos \theta + j \sin \theta$. We also must consider what kind of temporal filter we must implement for our application. We decided to use the same digital non-causal zero-phase finite impulse response (FIR) filter described in Sec. 3.4.1.

4.2.1 Masking regions of interest

In order to optimize the spotting process, we decided to mask facial regions of interest (ROIs) in which, according to the Facial Action Coding System (FACS) [65], MEs might appear. For that purpose, we create a mask M_1 using the facial landmarks localized in Sec. 4.1.4. These areas are further isolated using an adaptive threshold of validation computed from the local amplitude (similar to the one discussed in Sec. 3.4.2).

However, since the scale of local amplitude might vary from subject to subject (for example, in videos with subjects wearing glasses the local amplitude in the border of the glass frames was very high compared to the rest of the face), we need to normalize the local amplitude before we can threshold it.

$$M_2 = \begin{cases} 1 & \text{if } \beta \leq \frac{A_n}{A_q} \\ 0 & \text{if } \beta > \frac{A_n}{A_q} \end{cases} \quad (4.2)$$

where A_n is the calculated local amplitude of the image at frame n , A_q is the 95-percentile of the empirical distribution of the amplitudes along the video and β is a threshold selected by the user (see Sec. 4.4.1). By masking the low amplitudes areas we have effectively selected the regions in which MEs can be detected. We combine both masks ($M = M_1 \& M_2$) and refine the result further using morphological opening (Fig. 4.5d). Finally, we mask the quaternionic phase $\Phi_n \mathbf{u}$ (Fig 4.5b) with M (as seen in Fig. 4.5e).

4.3 Micro-Expression Spotting

The next step would be to calculate the phase variations over time and spot any subtle movements as MEs. However, by this logic, eye blinks and eye gaze changes could be wrongfully considered as MEs. Instead of just ignoring the information given by the eye areas, we can use it to help our system discard the possible false-positives. Thus, we divide our masked data in five different areas: the eye regions (left and right eye), and the facial features regions (left and right eyebrow and mouth area).

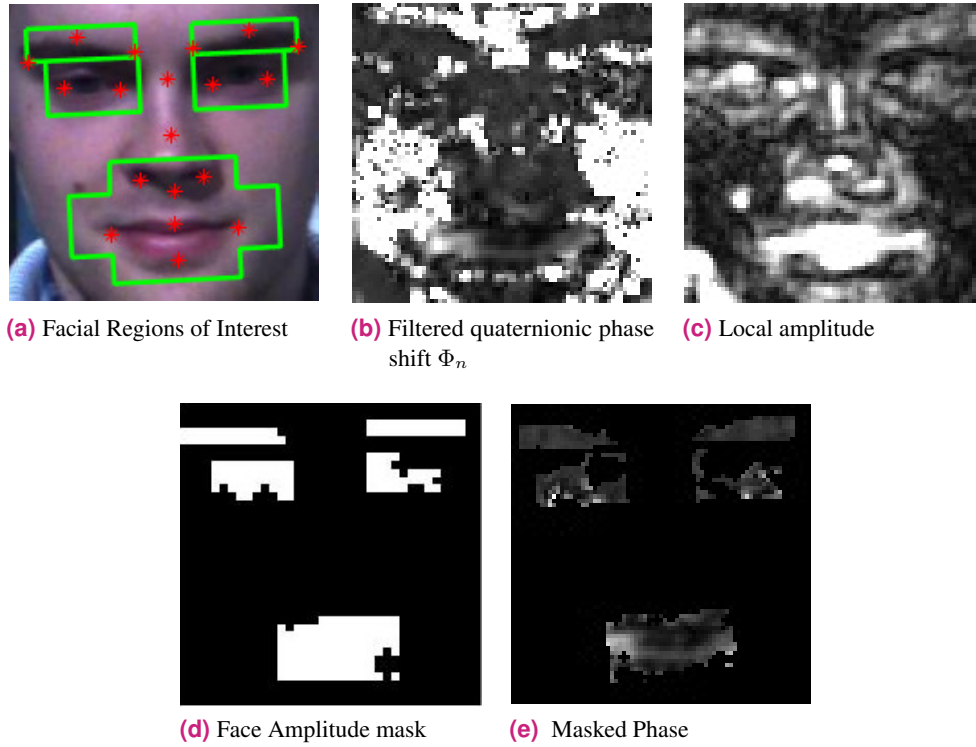


Figure 4.5.: Masking facial regions of interest.

Our proposed method goes as follows: First, for each ROI, we transform the quaternionic phase variance overtime into 1-D signals and separate them into 2 groups: eye signals (Fig. 4.6a) and facial feature signals (Fig. 4.6a). Secondly, we detect peaks on the signal that go over a certain threshold (the horizontal dashed line) which represent subtle changes in the video. For Fig. 4.6a these peaks (magenta dots) represent eye blinks or movements while in Fig.4.6b represent ME candidates. Finally, we select the peaks (or pair of consecutive peaks)³ in Fig.4.6b that does not coincide in time with the ones in Fig. 4.6a as a true MEs (cyan dots).

Preprocessing

Our first step is to minimize the effect of potential global movements from the spotting process. For this, we will consider any head pose change or translation that might occur during the video as rigid motion. With this in mind, we subtract the average of the masked quaternionic phase:

$$\Phi'_{n,s} \mathbf{u}'_s = \Phi_{n,s} \mathbf{u}_s - \frac{1}{\text{card}(M)} \sum_{r \in M} \Phi_{n,r} \mathbf{u}_r \quad (4.3)$$

³Some MEs have a fast onset and offset, thus, they would be represented by two peaks. However, some other MEs have a slow offset, so they would be represented by only one peak

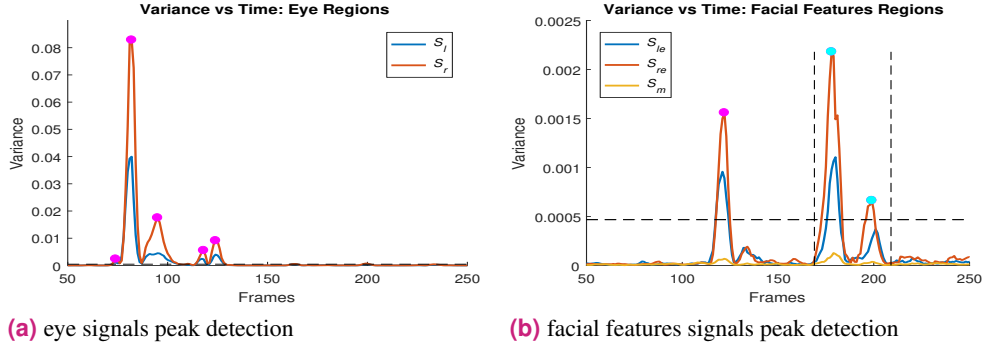


Figure 4.6.: Micro-expression spotting process.

where s is a masked pixel, $\mathbf{u}'_s = i \cos \theta'_s + j \sin \theta'_s$ and $\text{card}(M)$ is the cardinality of the mask M . When an object follows a rigid motion such as translation, all its elements follow a motion with the same orientation and magnitude, thus Eq. 4.3 reduces the effect of this kind of movements for the spotting step while the elements in non-rigid movements such as facial expressions follow different orientations and magnitudes and are not heavily affected by this step. Facial expressions are non-rigid movements they won't be heavily affected by the previous process. Next, we compute the magnitude of the phase by calculating the euclidean norm of the phase:

$$|\Phi'_n| = \sqrt{(\Phi'_n \sin \theta')^2 + (\Phi'_n \cos \theta')^2} \quad (4.4)$$

Feature Signal Generation

For each area and each frame, we calculate the variance:

$$S(n) = \frac{1}{L} \sum_{l=0}^{L-1} |\Phi'_{n,l} - \mu_n|^2 \quad (4.5)$$

where S is a 1-D signal which peaks represent subtle changes in the video, μ_n is the mean value of $|\Phi'_n|$ in a given area, l is the index of the pixels in a selected area and L is the total number of pixels in that area.

Peak Analysis

From the previous step, we obtain 5 signals: S_l and S_r which correspond to the left and right eye areas and S_{le} , S_{re} and S_m which correspond to the left eyebrow, right eyebrow and mouth areas respectively. The next step is to calculate an adaptive threshold for the

eye signals and one for the facial feature signals⁴. We start by computing the median and maximum values of the calculated signals:

$$max_E = \max_{\forall n \in N} (S_l(n), S_r(n)) \quad (4.6)$$

$$max_S = \max_{\forall n \in N} (S_{le}(n), S_{re}(n), S_m(n)) \quad (4.7)$$

$$med_S = \text{median}_{\forall n \in N} (S_{le}(n), S_{re}(n), S_m(n)) \quad (4.8)$$

and then we create the thresholds:

$$T_E = \frac{max_E}{2} \quad (4.9)$$

$$T_F = med_S + (max_S - med_S) \times \alpha \quad (4.10)$$

The next step is to localize any peak or local maxima in the signals that surpasses the previously computed thresholds (using T_E as threshold for S_l and S_r and T_F as threshold for S_{le} , S_{re} , S_m). For each signal S , we obtain a matrix:

$$\mathbf{P} = \begin{bmatrix} x_1 & n_1 \\ \vdots & \vdots \\ x_k & n_k \end{bmatrix} \quad (4.11)$$

where k is the total number of detected peaks, x_i and n_i are the magnitude and time (frame number) of a detected peak.

We perform a procedure to refine \mathbf{P} by choosing the tallest peak and discard all peaks which are closer than a minimal peak-to-peak separation (ψ frames). Then, the procedure is repeated for the tallest remaining peak and iterates until it runs out of peaks to consider. The next step is to discard redundant information by combining different peak matrices into one without duplicates (see Algorithm 1). We obtain $\mathbf{P}_E = \text{FUSEPEAKS}(\mathbf{P}_l, \mathbf{P}_r)$ for the eyes areas and $\mathbf{P}_F = \text{FUSEPEAKS}(\mathbf{P}_{le}, \mathbf{P}_{re}, \mathbf{P}_m)$ for the facial features areas.

Then, we compare each peak obtained by the facial features (P_F) with each one obtained by the eyes (P_E) using a set of rules to identify MEs and discard eye blinking (see Algorithm 2). Finally, we discard as an eye blink any peak from \mathbf{P}_F that has a corresponding peak in \mathbf{P}_E (see Algorithm 2).

⁴The thresholds are different because one is meant to be used for blink spotting and the other for ME spotting.

Input : A set of input peak matrices $\{\mathbf{P}_1, \dots, \mathbf{P}_m\}$

Output : Fused peak matrix \mathbf{P}_u

```

1 Concatenate the elements of the input matrices:  $\mathbf{P}_f = \begin{bmatrix} \mathbf{P}_1 \\ \vdots \\ \mathbf{P}_m \end{bmatrix}$  where  $k_f =$  number of rows
   in  $\mathbf{P}_f$ 
2 Sort  $\mathbf{P}_f$  using  $n_{f,i}, i = 1, \dots, k_f$ 
3  $s = 1$  and  $\mathbf{P}_{u,s} = [x_{u,s}, n_{u,s}] = \mathbf{P}_{f,1} = [x_{f,1}, n_{f,1}]$ 
4 for  $i \leftarrow 2$  to  $k_f$  do
5   if  $n_{u,s} = n_{f,i}$  then
6     if  $x_{u,s} \leq x_{f,i}$  then
7        $x_{u,s} = x_{f,i}$ 
8     end
9   else
10     $s = s + 1$ 
11     $\mathbf{P}_{u,s} = [x_{u,s}, n_{u,s}] = \mathbf{P}_{f,i} = [x_{f,i}, n_{f,i}]$ 
12  end
13 end

```

Algorithm 1: FusePeak: Peak fusion algorithm

Input : Peak matrices $\mathbf{P}_E, \mathbf{P}_F$

Output : ME peak matrix \mathbf{P}_m

```

1  $k_F =$  number of rows in  $\mathbf{P}_F$ ;
2  $k_E =$  number of rows in  $\mathbf{P}_E$ ;
3  $s = 1$ ;
4 for  $i \leftarrow 1$  to  $k_F$  do
5    $flag = 0$ 
6   for  $j \leftarrow 1$  to  $k_E$  do
7     if  $n_{F,i} = n_{E,j}$  AND  $x_{F,i} \leq x_{E,j}$  then
8        $flag = 1$ 
9     end
10  end
11  if  $flag = 0$  then
12     $\mathbf{P}_{m,s} = [x_{m,s}, n_{m,s}] = \mathbf{P}_{F,i} = [x_{F,i}, n_{F,i}]$ 
13     $s = s + 1$ 
14  end
15 end

```

Algorithm 2: ME spotting

One thing to take into consideration is the nature of MEs. During most MEs, the face goes from a neutral state (onset) to a moment when the ME is at its peak (apex) and then, after a short period of time, it goes back to a neutral state (offset). However, there are some micro-expressions that have fast onset phase but very slow offset phase (some even remain in apex for seconds) which some methods would fail to detect. That means that an ME is comprised of either one or two subtle motions. Therefore, our method has been adapted to detect either one or two peaks per ME (a pair of peaks would be identified as one ME as seen in Fig. 4.6b) depending on the case. One advantage of this approach is that, if an eye movement happens during the onset phase or the offset phase, our method might discard only one peak and the ME would still be spotted.

4.4 Experimental Results and Discussions

In this section, we describe the experimental procedures. Firstly, we talk about the parameters selection and evaluation scheme for our proposed method. Secondly, we present our results and compare them with the state of the art. Thirdly, we study the impact of the different parameters on our system. Finally, we discuss the relevance of our results and what challenges we faced in the implementation of our method.

4.4.1 Evaluation Procedure

For our experimentation, we selected two spontaneously elicited ME databases: the Spontaneous Micro-Expression Database (SMIC-HS) [113]⁵ and the improved Chinese Academy of Sciences Micro-expression (CASME II) [230]. For both datasets, we can calculate up to 4 levels of the Riesz pyramid (see Fig. 3.6). The first two levels (which have the information of the high frequency sub-bands) seem to carry an important amount of undesired noise and the third level of the pyramid seems more sensible to detect MEs compared to the fourth level (see Sec. 4.4.3). Thus, we choose to use only the third level of the pyramid.

Based on Sec.4.2, we design a FIR non-causal low-pass temporal filter with cut-off frequency of 10 Hz and a band-pass filter with cut-off frequencies from 2 to 10 Hz. The filter order was selected according to the database sampling rate (filter of order 18 for the SMIC-HS database and 36 for the CASME II database). After analysing the spectral response of both filters in both databases (Fig. 4.7a and Fig. 4.7b), we can observe that the low-pass filter is more effective at attenuating high frequencies, thus, we choose it for temporal filtering. For spatial filtering, we used a Gaussian Kernel K_ρ with standard deviation $\rho = 2$ in both databases.

⁵We decided to use only the videos captured using the high speed camera

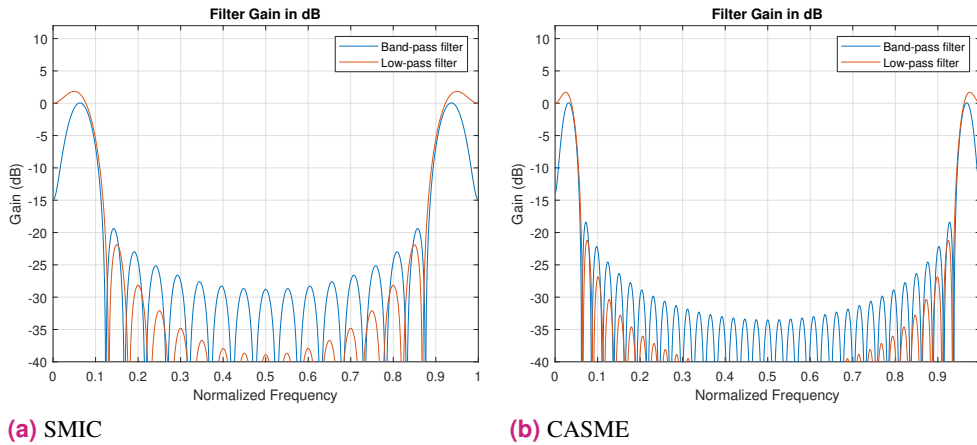


Figure 4.7.: Filter gains for FIR filters

One thing to consider in the parameter selection is the impact of the duration of MEs in the spotting process. Since MEs can last from 170 to 500 ms [232], we selected a conservative lower bound for the peak-to-peak minimal separation ψ by taking half of the minimal expected duration (that would be 85 milliseconds which will correspond to 9 frames for SMIC-HS and 18 frames for CASME II). For the β parameter, we perform a leave-one-subject-out cross validation which shows that the best values for β are close to 0.1 for the SMIC-HS dataset and between 0.2 and 0.22 for the CASME II dataset. We report the aggregate results of the evaluation measure in Sec. 4.4.2.

After the peak detection step, all the spotted peak frames are compared with ground truth labels to tell whether they are true or false positive spots. With a certain threshold level, if one spotted peak is located within the frame range (between the onset and offset) of a labelled ME video, the spotted sequence will be considered as one true positive ME. Otherwise, we will count a the duration of the labelled ME (offset–onset+1 frames) as a false positive. We define the true positive rate (TPR) as the percentage of frames of correctly spotted MEs divided by the total number of ground truth ME frames in the database. The false positive rate (FPR) is calculated as the percentage of incorrectly spotted frames divided by the total number of non-ME frames from all the image sequences. We evaluate the performance of our ME spotting method using receiver operating characteristics (ROC) curves with TPR as the y axis and FPR as the x axis.

4.4.2 Results

We performed the spotting experiment on CASME II and the high speed camera dataset SMIC-HS (described in Sec. 1.4.2). The spotting results on each dataset are presented in Fig. 4.8. The ROC curve is drawn by varying the parameter α in Eq. 4.10 (from 0 to 1 with a step size of 0.05). We evaluate the accuracy of our method by calculating the area under the ROC curve (AUC) for each dataset. The AUC percentage for the SMIC-HS database is

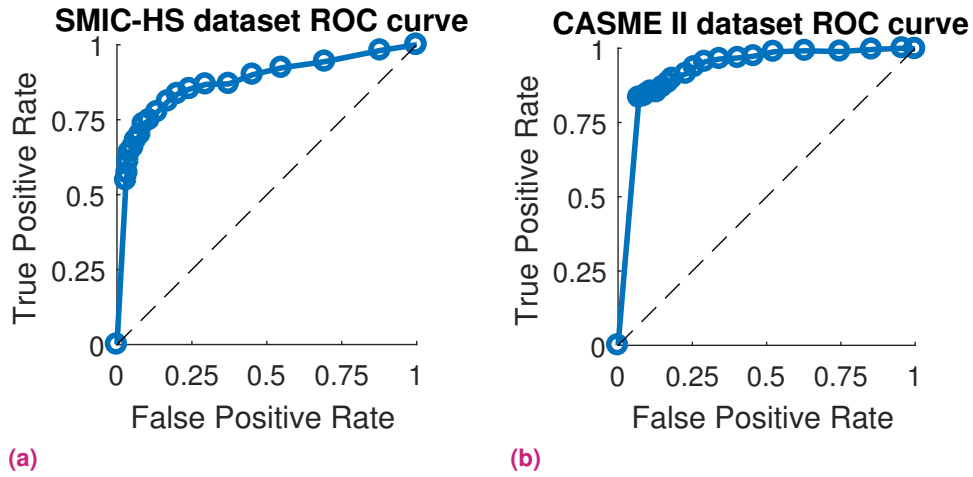


Figure 4.8.: ROC curves for ME spotting on SMIC-HS and CASME II datasets

Database	SMIC-HS	CASME II
LBP [112]	83.32%	92.98%
HOOF [112]	69.41%	64.99%
IP [126]	-	92.89%
Ours	88.26%	95.37%

Table 4.1.: AUC values of the ME spotting experiments using different methods on SMIC-HS and CASME II datasets using the measure metrics from [112].

88.61%, and for CASME II it is **90.93%**. However, most of the operating points in the ROC curve are not reasonable due to high number of false positives. Instead, some examples can be given with a reasonable ratio between true and false positives. For SMIC-HS dataset a spotting accuracy of 75.20% was achieved with only 10.89% FPR using $\alpha = 0.4$, and for CASME II dataset a spotting accuracy of 83.16% was achieved with only 10.52% of FPR using $\alpha = 0.15$.

We wanted to compare our results with the work made by [112] and [126] since we use a similar method of evaluation (ROC curves). However, this might become challenging since their method and parameters are different from ours. In order to compare results we changed some parameters that we used in Sec. 4.4.1. Specifically, the spotting range was changed to $[\text{ONSET} - (L - 1)/4, \text{OFFSET} + (L - 1)/4]$ and the false positive penalty was changed to L frames, being L a time window of about 0.32 seconds according to their work ($L = 33$ for SMIC-HS and $L = 65$ for CASME II). As it can be observed in table 4.1, the results of our method outperforms the results reported in [112] and [126].

4.4.3 Parameter Analysis

To evaluate the impact of the different parameters on our system, we test our proposed framework while varying its parameter values. The parameters we decided to evaluate are:

- The pyramid level from the Riesz transformation step (See Section 4.2). We evaluate the 4 levels obtained from the Riesz pyramid decomposition.
- β : the threshold in the amplitude masking step (See Section 4.2.1). We vary it from 0 to 0.9 using different scales (from 0 to 0.1 we use a scale of 0.01, from 0.1 to 0.3 we use a scale of 0.02 and finally from 0.3 to 0.9 we use a scale of 0.1).
- ψ : the minimal peak-to-peak separation (See Section 4.3). We vary it from 3 to 25 frames for the SMIC-HS dataset and from 3 to 40 frames in the CASME II dataset.

We perform our spotting experiments similarly as in Sec. 4.4.1 for both datasets and for each set of parameters we obtain the AUC. We decided to represent the result for each dataset as four result surfaces (one for each Riesz pyramid level) depicting the AUC as a function of ψ and β . The result surfaces and their contour representation can be seen in Fig. 4.9 and 4.10 for the SMIC-HS dataset and in Fig. 4.11 and 4.12 for the CASME II dataset respectively.

We also show the impact of the Riesz pyramid level by calculating the mean AUC as a function of β (the results are shown in Fig. 4.13a and 4.13b). We observe that we obtain the best performance values using the third level of the pyramid for the SMIC-HS dataset and the third and fourth levels for the CASME II dataset. However, a further inspection of the result surfaces in CASME II show that the results in the third level of the pyramid are slightly better. Thus, we decide to further inspect the result surface of the third pyramid level for both datasets (Fig. 4.13c and 4.13d). We observe that stable results are obtained when β ranges between 0.05 and 0.25 in the SMIC-HS dataset and between 0.05 to 0.3 in the CASME II dataset. An even closer inspection shows that we obtain the best results when β varies between 0.07 and 0.13 and ψ varies between 5 to 15 frames (at a 100 fps it corresponds to 50 to 150 milliseconds) for the SMIC-HS dataset (Fig. 4.13c) and when β varies between 0.12 and 0.15 and between 0.18 and 0.22 and ψ is bigger than 10 frames (at a 200 fps it corresponds to 50 milliseconds) for CASME II dataset (Fig. 4.13d).

One thing to take in consideration is that, by further augmenting the value of ψ , we might unknowingly discard MEs that occur within a small window of time. Considering that some videos in the SMIC-HS dataset contain multiple MEs but all videos in the CASME II dataset

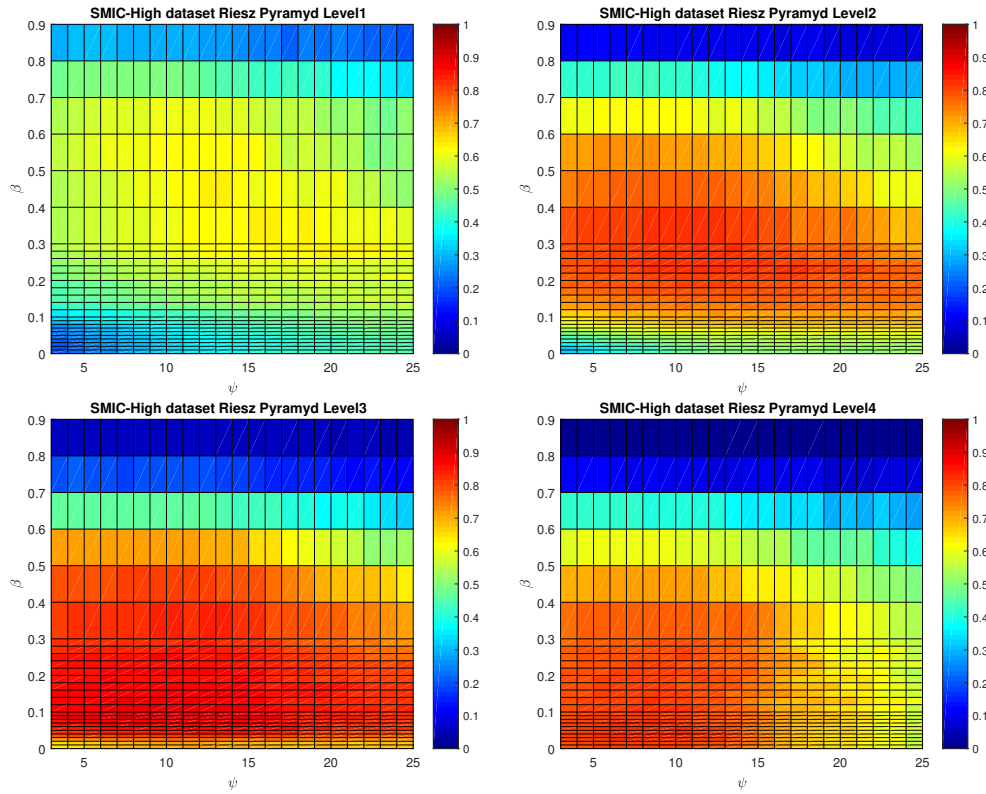


Figure 4.9.: Result surfaces for the different levels of the Riesz pyramid for the SMIC-HS dataset

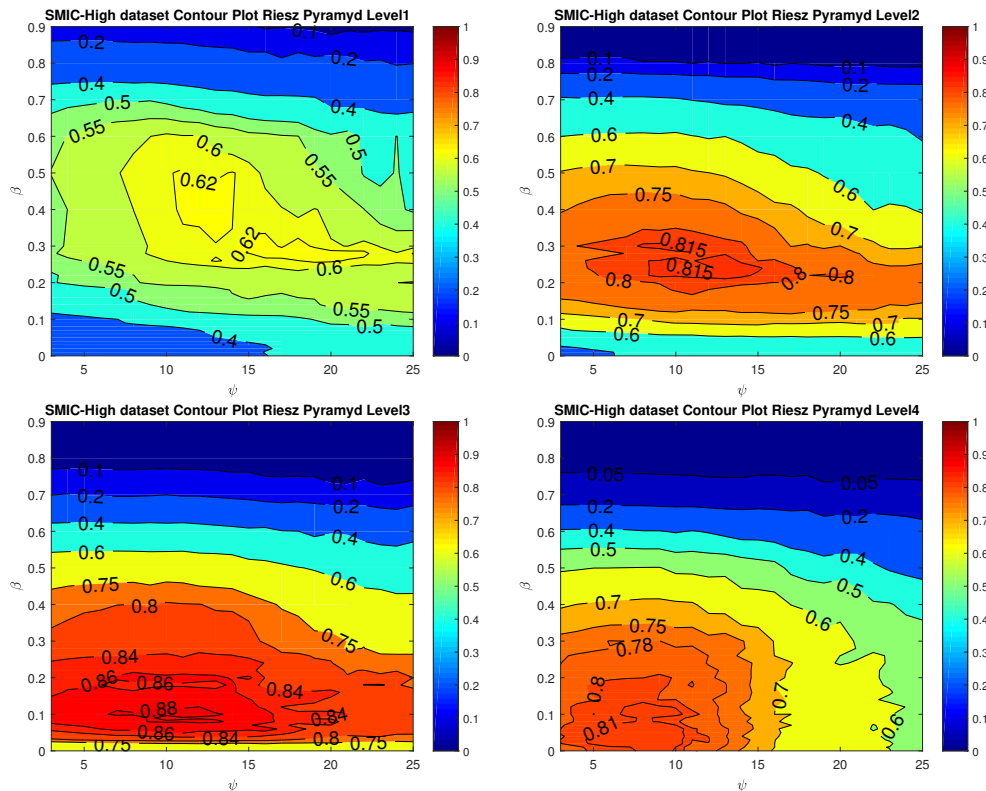


Figure 4.10.: Contour plots for the results of the different levels of the Riesz pyramid for the SMIC-HS dataset

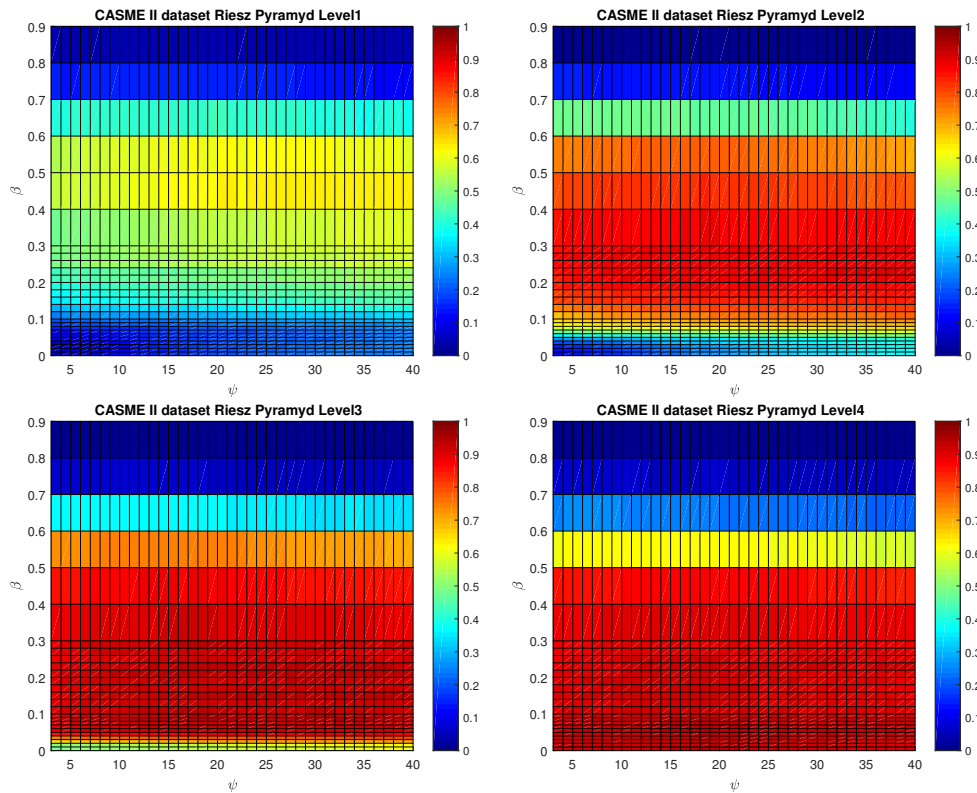


Figure 4.11.: Result surfaces for the different levels of the Riesz pyramid for the CASME II dataset

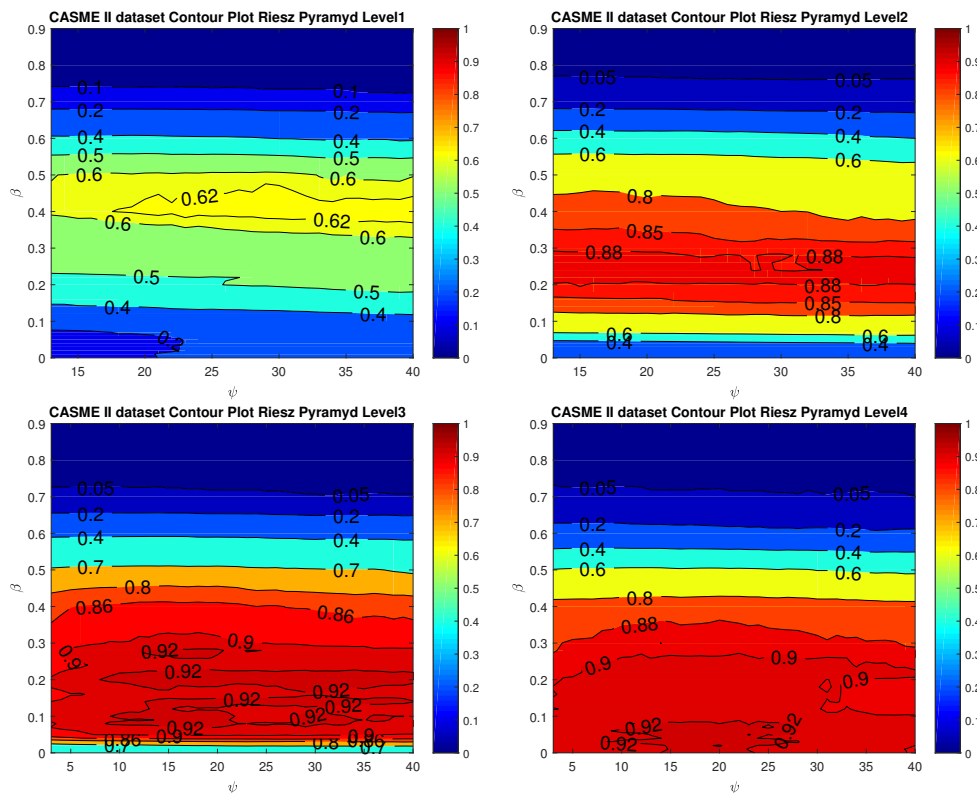


Figure 4.12.: Contour plots for the results of the different levels of the Riesz pyramid for the CASME II dataset

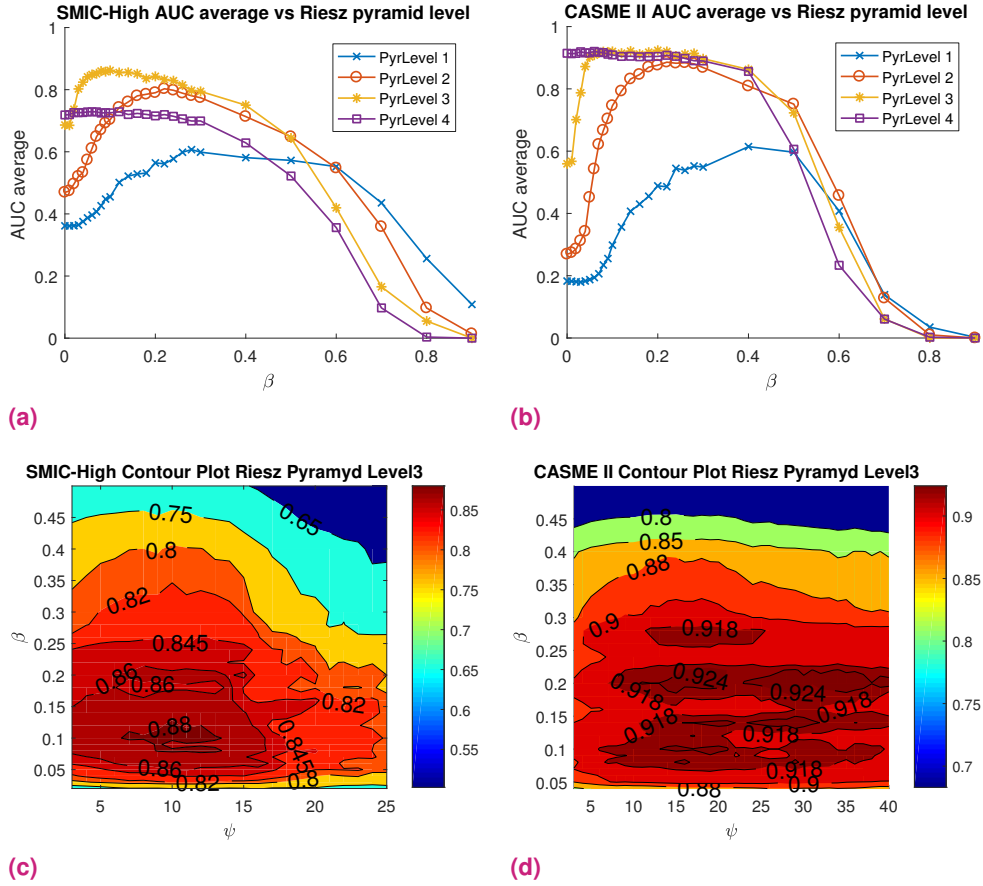


Figure 4.13.: Parameter Evaluation results in SMIC-HS and CASME II datasets

contain only one ME, it would explain why augmenting the value of ψ affects the results for SMIC-HS but not the results for CASME II.

After analysing our results, we estimate that, since both of the evaluated datasets have the same spatial resolution (640×480 pixels), we could use the same level of the pyramid for both and we could fairly establish a robust value range for β (between 0.05 and 0.25). Furthermore, this range value for β is coherent with our values obtained by our cross validation scheme. We can also estimate that a robust value for ψ for both datasets should correspond to at least 50 milliseconds. Moreover, we can conclude that the level of the pyramid and β are the parameters which have more impact on the performance of our system.

4.4.4 Discussion

Although both databases are similar, the SMIC-HS contain longer video-clips making the spotting task more difficult and more prone to false positive. We suspect this might be the reason why a higher AUC is achieved on CASME II database. Furthermore, the subjects

in the CASME II dataset were at a closer distance to the camera during video recording, thus the captured faces had a bigger resolution which result in a shift of the ME motion to low frequencies. This might explain why we could obtain good results using the fourth level of the Riesz pyramid in the CASME II dataset but not in the SMIC-HS dataset. A scale normalization of the captured faces could allow us to fix the pyramid level selection, regardless of the database.

Upon detailed examination of the spotting results, we found that a large portion of false negatives were caused by our algorithm dismissing true MEs as eye movements. This might happen because our system does not differentiate eye blinks from eye-gaze changes, thus discarding MEs that happen simultaneously with eye-gaze changes.

One of the main challenges of comparing our ME spotting method with the state of the art comes from the fact that there is not a single standard performance metric. [202] have proposed an evaluation standard for ME spotting but it only applies for sliding-window based methods. For example, [117] evaluate their work using mean absolute error and standard error and [48] evaluate their work using recall, precision and F-measure. But even in the case where different methods use the same metric, the results might not be comparable. For instance, [48] also uses ROC curves to evaluate their work but fails to disclose how they compute their false positives. In addition, comparing our work to the ones by [112] and [126] becomes complex because the false positive penalties used in both evaluations were different. That is the reason why the initial AUCs obtained in the beginning of Sec. 4.4.2 are different from the ones in our comparative Table 4.1, because the first tests had higher false positive penalties. This happens because these penalties were based on an assumed duration of MEs. However, since micro expressions have different duration times (as discussed in Sec. 4.3), a different approach should be considered for computing false positives.

Failure Cases

There are some cases in which our algorithm does not perform correctly. For instance, when our algorithm fails to detect the correct location of a subject's face and its landmarks, our method would not be able to extract the phase signal from relevant image areas (See Fig. 4.14).

Another instance of failure comes when our method is unable to remove the effect of macro-movements. Although, we have proposed a face alignment method using the KLT tracking algorithm (Sec. 4.1.3) and a rigid-motion minimization step (Eg. 4.3), in some cases it is not enough to suppress the effect of global movements. This macro-movements, when transformed to a feature signal, produces peaks that can wrongfully be spotted as MEs. Let's take as an example, a video which has a small face translation which produces

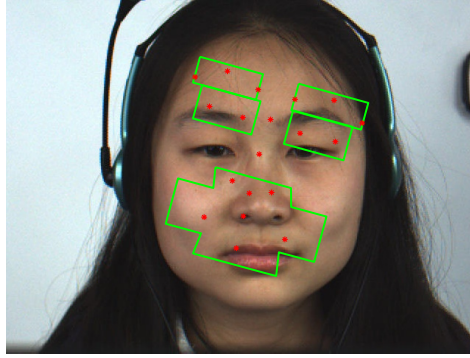


Figure 4.14.: Face registration failure case.

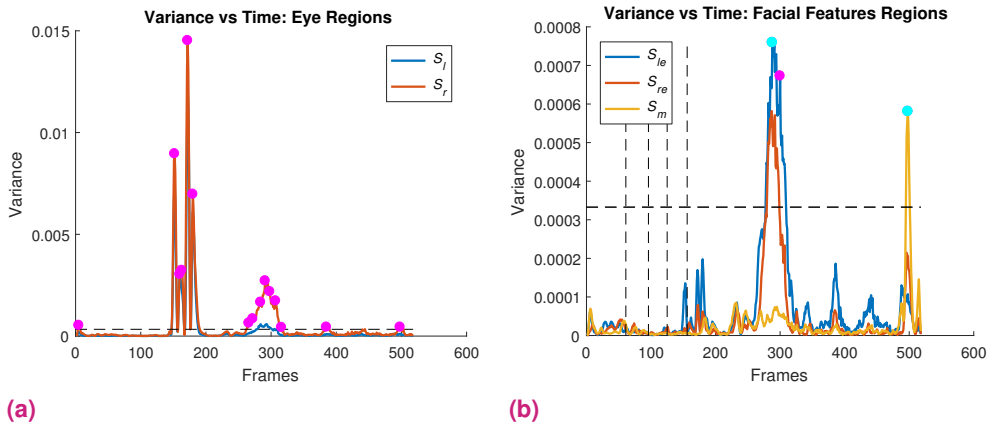


Figure 4.15.: Failure case when our system has wrongfully detected the peaks produced by macro-movements as MEs

a peak at frames 290 and a facial macro-expression that produces a second peak at frame 500. This video is represented by Fig. 4.15b, where the vertical dashed lines represent the time period between the true onset and true offset, the horizontal dashed line represents the minimum peak height threshold, the magenta dots represent the detected peaks and the cyan dots represent peaks identified as MEs. As we can see in Fig. 4.15b, our system wrongfully detects the peaks produced by these macro-movements as MEs. However, if we compare the signals produced by macro-movements with true MEs, we can see that the former produces wider peaks than the latter. Thus, we could discard these types of peaks in the future by analysing their width.

Another cause of error comes from the scale of the feature signals. The adaptive thresholds used for peak analysis (Eq. 4.9 and 4.10) depend on the maximum values of the feature signals. Consequently, the magnitude of the highest peak in a feature signal affects the accuracy of our method. Let's take as an example, a video which has a small face translation which produces a peak at frame 100 (Fig. 4.16b). Although, Algorithm 2 does not spot this peak as an ME, its magnitude creates a high threshold, thus our system ignores the peak that represents the true ME.

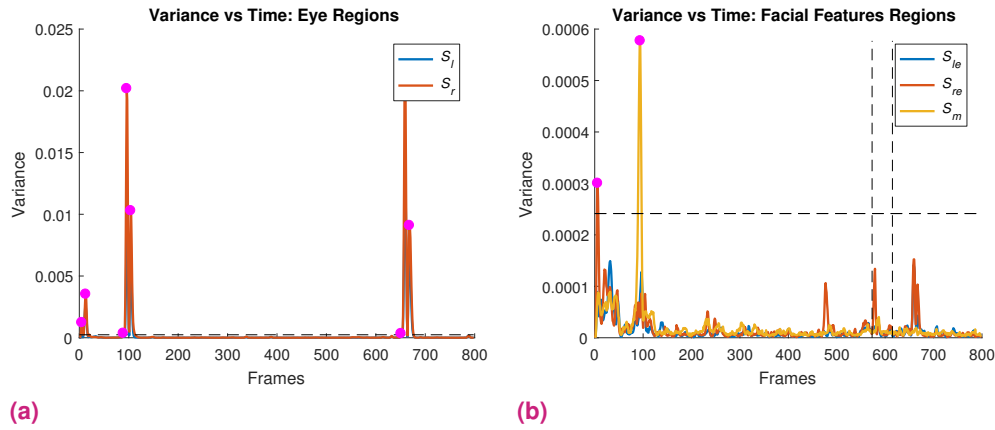


Figure 4.16.: Failure case when our system have created a tall threshold which ignores the peak produced by the true ME

4.5 Chapter Conclusions

In this chapter, we presented a facial micro-expression spotting method based on the quaternionic representation of the Riesz monogenic signal. Our main contributions are:

- We adapted the filtering scheme and amplitude masking method proposed in the previous chapter to analyse MEs.
- We designed a heuristic ME spotting method which analyses the motion of local face ROIs. What's more we proposed a methodology that separates real micro-expressions from subtle eye movements decreasing the quantity of possible false positives.

Experiments on two different databases showed that our spotting method surpasses other methods from the state of the art. Furthermore, the results of our parameter analysis experiment showed that this method is robust to changes in parameters. The results of our parameter analysis experiment also suggests that, even if one level of the Riesz pyramid might contain the most relevant information regarding facial micro-movements, other pyramid levels might contain complementary information about MEs. Thus, we should consider to find a way to combine the different levels of the pyramid for ME spotting in the future. Furthermore, we propose to use the quaternionic representation of phase from the Riesz monogenic signal for facial micro-expression recognition.

Micro-Expression Classification

Recognizing different micro-expressions has been a challenging problem ever since they were discovered by Ekman and Friesen [62] in the seventies. Although, there were some initial modest proposals for ME classification, it wasn't until the release of public available spontaneous ME datasets that a great number of proposals started to appear. A typical ME framework goes as follows: A face localization and cropping step, followed by a feature extraction method and a learning method.

Some feature extraction methods used in ME recognition extract the image intensity information to describe temporal facial features that appear during a facial expression (appearance features). A very popular family of appearance features methods used for ME recognition are LBP-TOP [97, 112, 142, 152, 161, 239] and its extensions [79, 89, 90, 91, 137, 217, 216]. Other methods extract features from the shape and deformation of facial features caused by facial expressions (geometrical methods). Most of these methods extract features from the optical flow of the ME sequence [5, 118, 123, 229]. Other methods use pre-trained systems that have learned mid-level [83] or high-level features to extract from images (learned methods) [100, 158]¹.

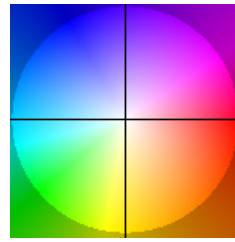
So far, we have extracted the elements of the monogenic signal to detect when a subtle motion or an ME takes place. However, the monogenic signal has also been used in the literature for ME recognition. [148], uses the Riesz wavelet to transform the input images into multi-scale monogenic wavelets, then it extracts features from the magnitude, orientation and phase from each scale. In another approach, [119] proposes to combine the orientation, magnitude and optical strain extracted from OF with the phase component from the monogenic signal. In both cases, the authors extracted information about the orientation and the phase of the motion (either from the OF or the monogenic signal) in order to model MEs.

Thus, based on our previous work, we propose a framework that uses the Riesz pyramid to extract multi-scale oriented phase features which allow us to model and classify MEs. This chapter is organized as follows: Sec. 5.1 presents a novel way to represent motion using the orientation and phase from the monogenic signal. Sec. 5.2 introduces a novel oriented phase features extraction method to model MEs. In Sec. 5.3, we present a series of experiments

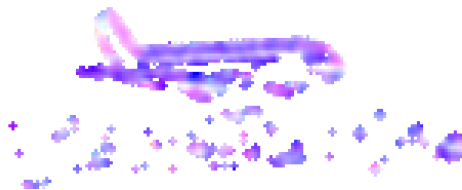
¹A complete list of all these mentioned can be found in Sec. 2.3



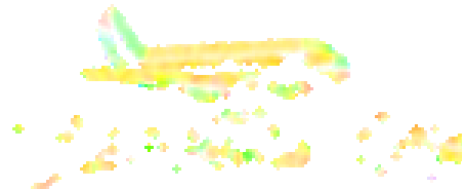
(a) Image Sequence



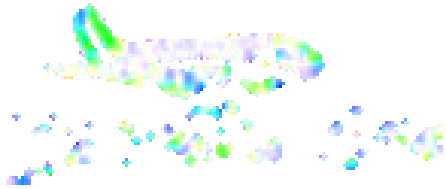
(b) False-color representation



(c) Up Shift



(d) Down Shift



(e) Left Shift



(f) Right Shift

Figure 5.1.: False-Color representation of oriented quaternionic phase response to different motions

for partial macro and micro-expression recognition. Finally, Sec. 5.4 presents the chapter's conclusions.

5.1 Oriented Phase Motion Representation

In this section, we propose an image pair model for ME representation that uses both the orientation and phase components of the Riesz pyramid as proxies for motion. This section goes as follows: Sec. 5.1.1 talks about the orientation component of the Riesz pyramid including its advantages and limitations; Sec. 5.1.2 proposes a way to model and normalize the temporal evolution of ME motion by means of a mean oriented image pair;

5.1.1 Local Orientation of the monogenic signal

In chapter 3 and chapter 4 we used the Riesz pyramid to extract the local amplitude A and the filtered quaternionic phase $(\phi \cos(\theta), \phi \sin(\theta))$ for spotting subtle motions and even MEs. However in both cases we calculate the euclidean norm of the phase and discard the

local orientation θ (Eq. 3.36 and Eq. 4.4). The local orientation θ represents the dominant direction in the image at any given point. This representation comes from the formulation of the monogenic signal which assumes images as intrinsically one-dimensional signals [72]². This means that the monogenic signal is useful for modelling image features such as edges and lines that have variation in one direction only, but cannot model image features such as corners that have variation in two directions (intrinsically 2D signals) [219]. This is one important limitation of the monogenic signal [24].

One of the advantages of the monogenic signal is that, in the same fashion of the analytical signal, it preserves the split of identity. This means that, the local phase is invariant to changes of the local orientation, and the local orientation is invariant to changes of the local structure (which means that we can split them). If we can recover the correct local direction from the local orientation, we have an ideal split of identity with respect to energetic, geometric, and structural information of the signal. However there are a couple of problems to estimate the correct local direction. The first one is that the estimation of local orientation is unstable if the local phase ϕ is close to 0. The second problem is that it's not possible to find an absolute estimation for the local direction but rather the relative estimation. The absolute solution can be obtained by adding some constraints on the smoothness of the phase and orientation (such as the ones used for motion estimation in Sec. 3.1.1). However, as it was previously discussed in Sec. 3.3.1, it is better to keep a quaternionic phase representation $(\phi \cos(\theta), \phi \sin(\theta))$ rather than trying to divide into local phase and orientation to avoid the sign ambiguity problem.

The main question that we need to ask is whether the orientation component of the quaternionic phase can be used to differentiate between opposing motions. Let's take Fig. 5.1a, an airplane image which is translated one pixel in any given direction. We present the resulting filtered quaternionic phase with a false-color representation in which the image saturation represents the phase ϕ component and the hue represents the orientation θ component (see Fig.5.1b). The areas of low amplitude were masked using the technique presented in Sec. 3.4.2 for better visualization. The image is translated one pixel up (Fig. 5.1c), down (Fig. 5.1d), to the left (Fig. 5.1e) and to the right (Fig. 5.1f). As we can see, by using the quaternionic phase representation, we were able to represent motion in different directions (as evidenced by the different hues from the false-color image representations from the middle and lower rows of Fig. 5.1). It is worth nothing that, if we observe the tail of the plane (which has a diagonal edge), the orientation of the quaternionic phase is not the same as the one of the translation but rather it is perpendicular to the orientation of the edge. This means that the oriented quaternionic phase is affected by the aperture problem³. Nevertheless, when we compare two opposing motions (Fig. 5.1c vs. Fig. 5.1d or Fig. 5.1e vs. Fig. 5.1f), their respective orientated quaternionic phases are also opposite. This becomes important

²The intrinsic dimension is the number of degrees of freedom necessary to describe a local structure [219]

³This could be corrected by adding some motion constraints such as the ones used for motion estimation in Sec. 3.1.1. However, this is out of the scope of this work.

for ME recognition, where different MEs represent motions in different directions. For example, when analysing the eyebrows movement during an ME of surprise, the eyebrows raise. On the other hand during an ME of anger, the eyebrows are contracted (lowered).

5.1.2 Mean Oriented Riesz Image Pair

In the previous section we showed how using a relative quaternion phase estimation we can differentiate motion from different directions. However, we only analysed the motion between two consecutive frames. Considering the MEs are captured as video sequences of several frames, we also need to analyse the temporal evolution of these motions. However, some considerations must be made before proposing an ME modelling scheme.

The first consideration is whether it is necessary to analyse the whole ME sequence in order to classify it. An ME can be divided into two sequences: an onset phase that goes from onset to apex (the face goes from a neutral state to a state of peak expressibility), and an offset phase that goes from apex to offset (the face goes from peak expressibility to a neutral state). One thing to consider is that the second sequence use the same facial muscles as it follows the reverse motion of the first sequence. Furthermore, the duration from onset to apex is shorter than the one from apex to offset, thus, the spatial displacement of the ME between consecutive frames is more evident in the first sequence compared to the second sequence. What's more, there are some MEs that have a very slow offset phase (some even remain in apex for seconds) [232]. Thus we decide to focus on modelling only the sequence that goes from onset to apex.

The second consideration is the length of the captured MEs. Depending on the image capture device, a recorded ME can be composed from 5 to 100 consecutive frames (see Table 1.1). Let's take as example the two most currently used ME databases: SMIC and CASME II. SMIC videos were captured at a frame rate of 100 fps while the CASME II videos were captured at 200 fps. Thus, a given ME captured in the CASME II database will be composed of twice the number of frames. Consequently, an ME will have half the spatial displacement between consecutive frames in the CASME II database compared to SMIC. This will affect the sensibility of algorithms which analyse motion between consecutive frames such as our Riesz pyramid subtle motion analysis method (Sec. 3.4). Some authors have proposed to mitigate this effect, normalizing the duration of MEs by temporally interpolating the video sequence [241].

The third consideration is the facial ROIs to analyse. In our previous work in ME spotting we selected a series of local ROIs (see Sec. 4.1.4) and masked them using the local amplitude from the Riesz pyramid in order to isolate areas of potential noise (see Sec. 4.2.1). Although this approach was effective for ME spotting, it has the drawback of ignoring certain facial areas of low amplitude which might have some interesting information while an ME is

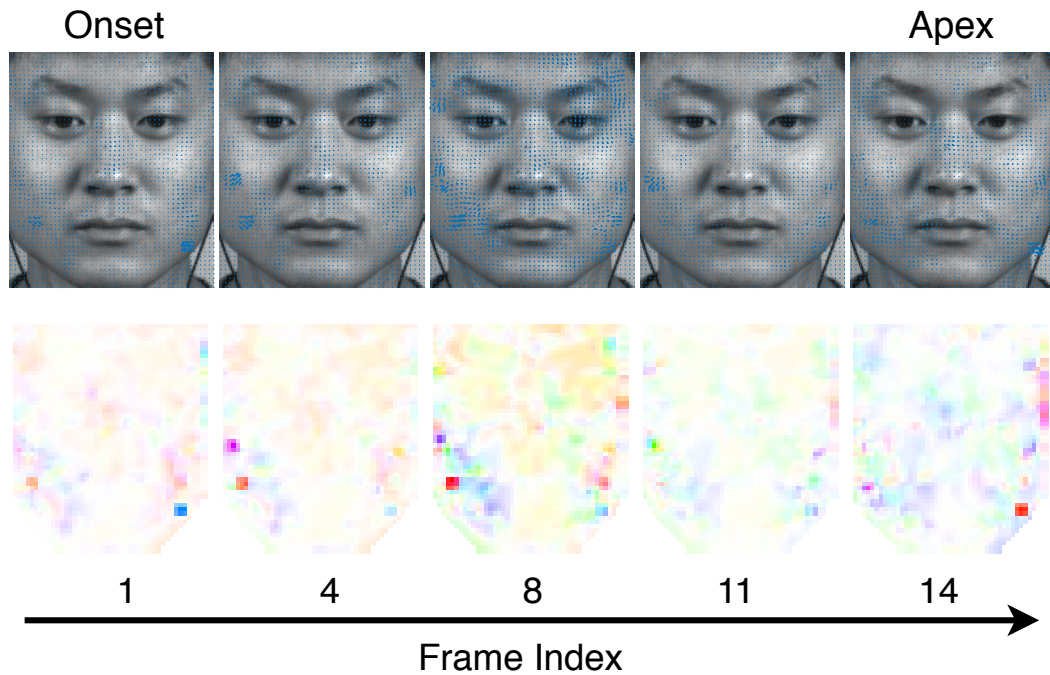


Figure 5.2.: Riesz Oriented Phase of an Onset-Apex sequence. The top row shows the oriented phase as displacement vectors and the bottom row shows their respective false-color representation

taking place (such as the cheek areas). In theory, we could solve the latter problem by selecting a face grid ROI approach (see Sec. 2.2.3). However, as stated in Sec. 3.4.2, we need to deal with the high variance of the quaternionic phase $(\phi \cos(\theta), \phi \sin(\theta))$ in areas of low local amplitude A . Let's take Fig. 5.2 as an example. We show different frames of an ME sequence that goes from onset to apex, calculate the quaternionic phase for the sequence and represent the motion values as vectors (top row) and as a false-color representation (low row). In theory, the sequence should represent the temporal evolution of a positive ME (cheeks and corner of the lips raising). As we can see, our current method can detect certain motion in the cheek areas (specially in frame 8). However, it also wrongfully detects some square areas as if they had a higher motion (the small squares of high saturation around the cheek areas and near the external border of the face). These areas appear as a consequence of the spatio-temporal filtering scheme proposed in Sec. 3.3.3, which is able to filter noisy data in areas of medium and high local amplitude, but it spreads the noise in areas of low amplitude (that's the main reason behind the amplitude masking methods proposed in Sec. 3.4.2 and Sec. 4.2.1). One thing that we can observe is that in the areas of true motion, the orientation stays similar or changes slowly overtime following a tendency, whilst in the noisy areas, the orientation changes randomly between frames.

Taking the aforementioned considerations into account, we propose to model the temporal evolution of the ME in a single image paired called the **mean oriented (MOR) image pair**.

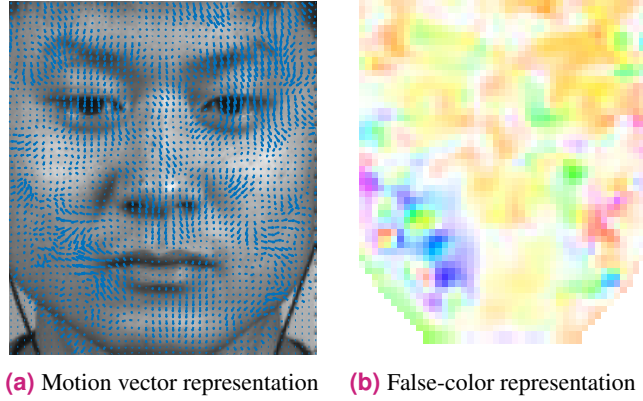


Figure 5.3.: MOR Image Pair

We simply calculate the filtered quaternionic phase of an ME sequence from onset to apex and then we average the results through the time axis.

$$\overline{\phi \cos(\theta)} = \frac{1}{apex - onset + 1} \sum_{t=onset}^{apex} \phi_t \cos(\theta_t) \quad (5.1)$$

$$\overline{\phi \sin(\theta)} = \frac{1}{apex - onset + 1} \sum_{t=onset}^{apex} \phi_t \sin(\theta_t) \quad (5.2)$$

The main intuition is that by temporally averaging the filtered quaternionic phase, the real motion of each pixel is modelled in a single orientation and magnitude while reducing the effect of wrongfully detected motion due to noise. Let's take the video sequence from Fig. 5.2 and produce its MOR image pair (Fig. 5.3). As we can see, the new image pair describes the motion in the cheeks for a positive ME but now the noisy areas have been reduced.

5.2 Mean Oriented Riesz Features

We propose a method to extract the orientated phase called **Mean Oriented Riesz Features (MORF)**. Using the taxonomy for feature extraction methods from Sec. 2.3, MORF would be a Geometrical Dynamic Pre-designed Local descriptor. This section goes as follows: Sec. 5.2 describes the implementation of our proposed feature extraction method; Sec. 5.2.1 introduces a couple of variations to our proposed feature extraction method.

5.2.1 Implementation Details

We apply the same face registration step documented in Sec. 4.1 to locate the face in an image and track its facial landmarks. However, instead of selecting local ROIs, we aim

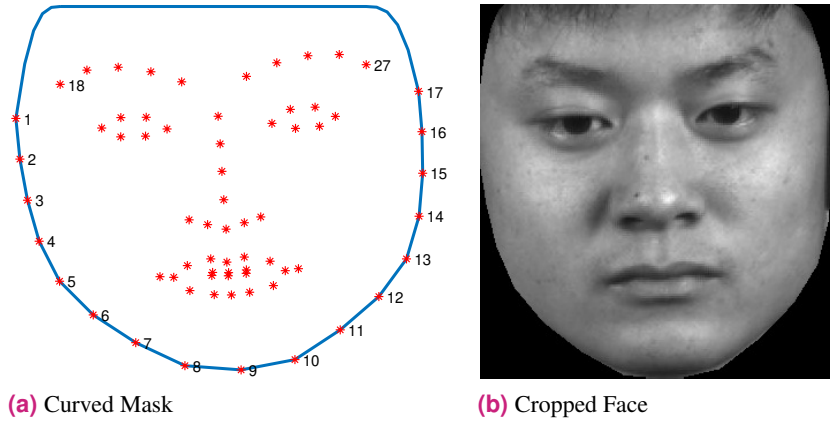


Figure 5.4.: Face Masking

to divide the face into a grid of rectangular ROIs. Before we do that, we create a curved mask that isolate image areas that might go outside of the desired face border. We use the fiducial points from the AAM (Fig. 4.4a) to delimit the curved mask (Fig. 5.4a). For the lower border we just use the fiducial points 1 to 17. For the upper border, we model a curve that goes from the fiducial face border (point 1 in the left side of the face and 17 in the right side of the face), to the point in top of the cropped image that is horizontally aligned to the most external fiducial point of each eyebrow (point 18 in the left side and 27 in the right side). The resulting cropped face can be seen in Fig. 5.4b.

The cropped face sequence is processed using the Riesz pyramid to obtain the filtered quaternionic phase (Sec. 3.4.1) and we use Eq. 5.1 and Eq. 5.2 to obtain the MOR image pair⁴. We divide the face into a grid of equally sized non-overlapping rectangle areas. As we can see in Fig. 5.5a, each pixel in the image pair represents a motion vector with a magnitude and angle. We can extract them from the oriented phase by:

$$\overline{\phi}_R = \sqrt{(\overline{\phi}_R \cos(\theta_R))^2 + (\overline{\phi}_R \sin(\theta_R))^2} \quad (5.3)$$

$$\overline{\theta}_R = \arctan\left(\frac{\overline{\phi}_R \sin(\theta_R)}{\overline{\phi}_R \cos(\theta_R)}\right) \quad (5.4)$$

where $\overline{\phi}_R$ is a matrix containing the phase of every pixel, $\overline{\theta}_R$ is a matrix containing the dominant orientation of every pixel and R correspond to the level of the Riesz pyramid level we are extracting the oriented phase from. The next step is to create the histogram of oriented phase for each one of the rectangular blocks. For each pixel, a bin is selected based on the orientation θ and a weighted vote is cast based on the value of the phase ϕ (Fig. 5.5b). The final histogram is created by concatenating the histograms of all the given blocks (Fig. 5.5c).

⁴We have chosen to show the results of the third level of the Riesz Pyramid only since they show more significant ME motion (Sec. 4.4.3).

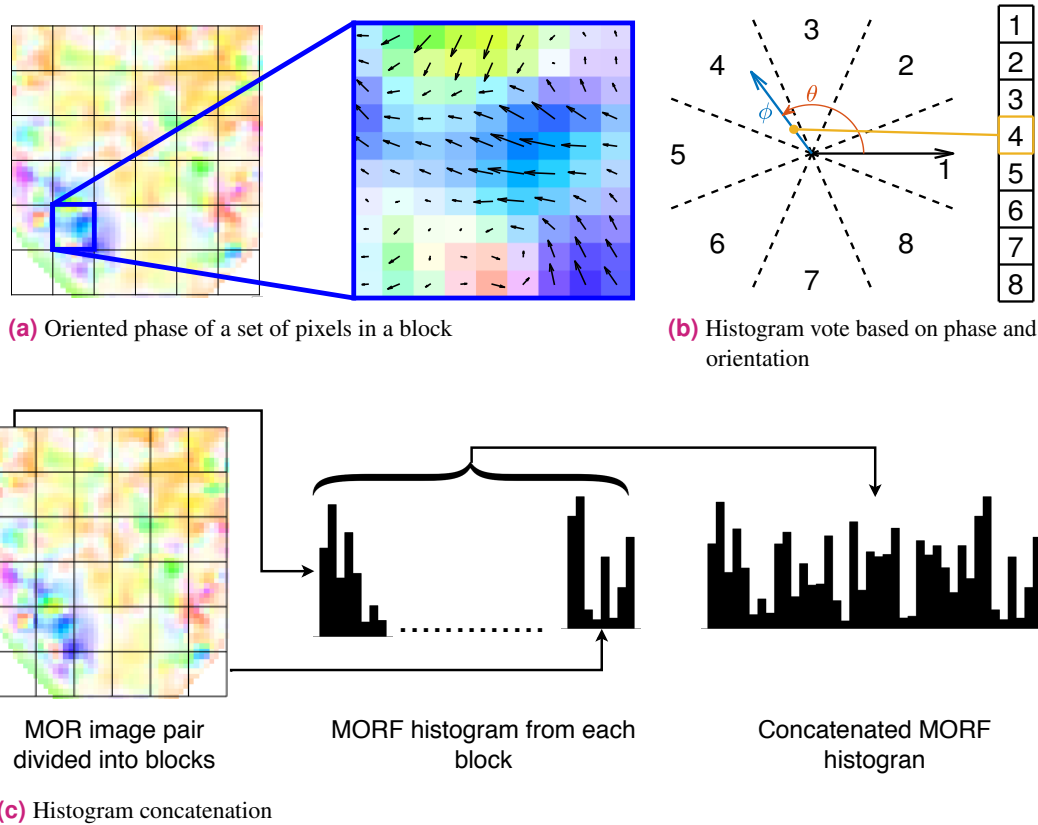


Figure 5.5.: Implementation of the MORF descriptor

The MORF descriptor depends on three parameters: G which determines the grid division of ($G \times G$) ROIs, O which determines the number of orientations bins of the descriptor and R which determines the level of the Riesz pyramid we are going to extract. Thus $\text{MORF}_{G,O,R}$ produces a feature vector of $G^2 \times O$ length⁵. For the bin construction, we decompose the orientation space into a set of subsets Θ_i of the same size (π/O) where:

$$\Theta_k = \left[\frac{2\pi(k-1) - \pi}{O}, \frac{2\pi(k-1) + \pi}{O} \right) \quad \text{for } 1 \leq k \leq O \quad (5.5)$$

In Fig. 5.6, partition modes for different values of O are displayed. The complete implementation of the MORF operator for different parameters can be found in Algorithm 3 (the inputs $\overline{\phi_R \cos(\theta_R)}$ and $\overline{\phi_R \sin(\theta_R)}$ are obtained from Eq. 5.1 and Eq. 5.2 respectively).

5.2.2 Modifications on MORF

As we previously described, the original MORF descriptor is the computation of the oriented phase of the MOR image pair. We propose to extend the computation of MORF using different data and methodologies already discussed in previous section:

⁵Since we are extracting data from one level of the pyramid at the time, R does not affect the length of the feature vector

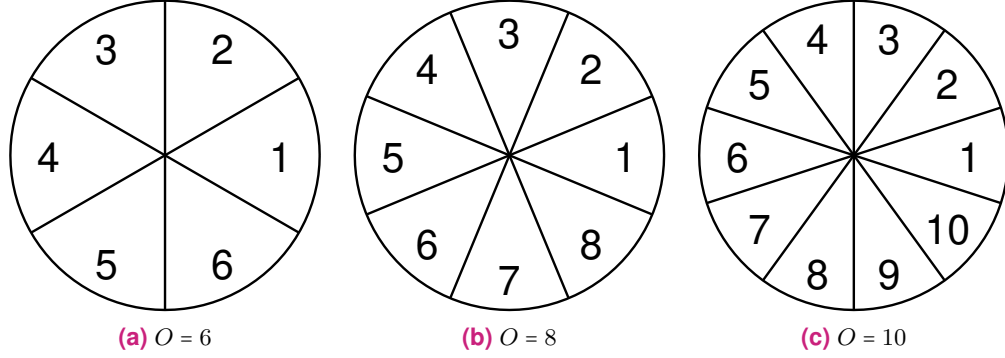


Figure 5.6.: Orientation bin construction

Input : $\overline{\phi_R \cos(\theta_R)}, \overline{\phi_R \sin(\theta_R)}, O, G$
Output : MORF normalized histogram H_n

- 1 Extract mean phase $\overline{\phi_R} = \sqrt{(\overline{\phi_R \cos(\theta_R)})^2 + (\overline{\phi_R \sin(\theta_R)})^2}$
- 2 Extract mean orientation $\overline{\theta_R} = \arctan\left(\frac{\overline{\phi_R \sin(\theta_R)}}{\overline{\phi_R \cos(\theta_R)}}\right)$
- 3 Create a set of orientation subsets $\Theta_k = \left[\frac{2\pi(k-1)-\pi}{O}, \frac{2\pi(k-1)+\pi}{O} \right)$ for $1 \leq k \leq O$
- 4 Extract the width and height of matrix $[w, h] = \text{size}(\overline{\phi_R \cos(\theta_R)})$
- 5 **for** $i \leftarrow 1$ **to** G **do**
- 6 **for** $j \leftarrow 1$ **to** G **do**
- 7 Create grid of rectangular divisions where

$$\mathbf{x} = \left(\frac{w \times (i-1)}{G} \leq x < \frac{w \times i}{G}, \frac{h \times (j-1)}{G} \leq y < \frac{h \times j}{G} \right)$$
- 8 **foreach** $z \in \mathbf{x}$ **do**
- 9 **if** $\overline{\phi_R}(z) > 0$ **then**
- 10 Find Θ_k that corresponds to orientation subset $\overline{\theta_R}(z)$ then
- 11 $P(i, j, k) = P(i, j, k) + \overline{\phi_R}(z)$
- 12 **end**
- 13 **end**
- 14 **end**
- 15 **end**
- 16 Reshape P into a 1-D vector $H = \text{reshape}(P)$
- 17 Normalize the histogram $H_n = \frac{H}{\sum H}$

Algorithm 3: MORF descriptor

- **Fused MORF:** In this approach, we just concatenate the results of two or more MORF from different levels of the Riesz pyramid. The idea is to use the oriented phase calculated from different sub-bands to potentially complement the information for modelling an ME. For the notation, we will simply specify which levels of the pyramid have been used in the parameter R (for instance, Fused MORF $_{G,O,2&3}$ means that the information of the second and third level of the pyramid has been fused). The length of the feature vector depends on the number of fused levels ($G^2 \times O \times$ number of levels).
- **Amplified MORF:** In this approach, we follow the amplification process of Sec. 3.3.4. The main idea is to make the subtle phase changes (which represent ME) more visible without amplifying the noise. First, we multiply the quaternionic filtered phase $(\phi \cos(\theta), \phi \sin(\theta))$ by a magnification factor (α) , then, after performing a quaternion exponentiation on it, we extract the amplified quaternionic phase:

$$(\sin(\alpha\phi) \cos(\theta), \sin(\alpha\phi) \sin(\theta)) \quad (5.6)$$

Finally, we use this representation to calculate the MOR image pair. We can further extend this approach by fusing the results from different levels of the pyramid to obtain the **Amplified Fused MORF**

- **Amplitude Weighted MORF:** In this approach, we multiply the quaternionic filtered phase $(\phi \cos(\theta), \phi \sin(\theta))$ with the local amplitude A before calculating the MOR image pair (Eq. 5.1 and Eq. 5.2). The main idea is to minimize the impact of the noise by assigning weights to the pixels depending on their correspondent local amplitude.
- **Amplitude Masked MORF:** In this approach, we isolate areas of the quaternionic filtered phase using our amplitude masking method (Sec. 3.4.2) before calculating the MOR image pair. The main idea is to minimize the impact of the noise by rejecting pixels of low local amplitude.

5.3 Experimental Results and Discussions

In this section, we describe the experimental procedures. We propose to do 3 sets of experiments:

1. **Experiment 1: Partial Macro-Expressions experiment.** We analyse the effect of spatial intensity for facial expression classification. For that we decide to classify “partial macro-expressions”, that is, videos which have been trimmed to show only a part of the total facial expression. We test and compare our proposed descriptors

to geometrical, appearance, pre-trained features descriptors. From the results we selected the two descriptors with better performance for the next experiment.

2. **Experiment 2: MORF vs. MOOF ME classification.** We classify ME expressions from two spontaneous ME databases testing different parameters for MORF and MOOF (a variation of MORF based on Optical flow). We evaluate the general accuracy of both methods and the effects of the parameter variations in two ME database.
3. **Experiment 3: MORF variations** We perform a thorough classification of ME expressions from two spontaneous ME databases using different variations of the MORF descriptor. We evaluate the general and per-class accuracy and compare our results with the state of the art.

5.3.1 Experiment 1: Partial Macro-Expressions classification

Before we start analysing MEs, we would like to test our method in different levels of facial expression intensity. However, there aren't available databases with different levels of spatial intensity between macro and micro expression. Thus, we decide to analyse "partial macro-expressions", that is, videos which have been trimmed to show only a fraction of the total facial expression. The aim of this experiment is to analyse the response of our method when the facial expressions becomes subtler and subtler. We also analyse the response of other classical descriptors of the state of the art.

Database

For our experimentation, we selected the extended Cohn-Kanade (CK+) dataset [128]. The CK+ database, contains 593 facial expressions from 210 adults. Videos were collected with Panasonic AG-7500 cameras at a resolution of 640×480 and a frame-rate of 30 fps. Subjects of different ages, genders and ethnicities were used as participants. Participants were instructed by an experimenter to perform a series of 23 facial displays; these included single action units and combinations of action units. Finally, 327 facial expressions were labelled into 7 basic emotions: anger (45), contempt (18), disgust(59), fear (25), happiness (69), sadness (28) and surprise (83). Each of the sequences contains images from onset (neutral frame) to peak expression (last frame). The partial macro-expression sequences are created by cutting videos that go from the onset to a new apex which is a fraction of the original apex $[\text{onset}, \text{round}(\frac{\text{apex}-\text{onset}+1}{s})]$ where $2 \leq s \leq 5$ and the total duration is $T = \text{round}(\frac{\text{apex}}{s})$ (see Fig. 5.7).



Figure 5.7.: Macro-expressions sequence at different time-stamps

Evaluation Procedure

We apply the face registration and cropping step described in Sec. 5.2. The cropped face sequence is processed using the Riesz pyramid to obtain the filtered quaternionic phase. For the spatio-temporal filtering we use the same parameters as the ones in Sec. 4.4.1 except the order of the FIR filter which is 6 this time (because the database has a frame-rate of 30 fps). Considering that the spatial resolution of the CK+ database is the same as CASME II and SMIC database, we process the third level of the pyramid to obtain the MOR image pair (we use the results of Sec. 4.4.3 as selection criteria). For the MORF descriptor, we evaluate different levels of grid division ($G = [6, 8]$) and orientation binning ($O = [6, 8, 10]$).

We decide to compare the MORF descriptor with a similar descriptor based on optical flow. First we process the motion of the videos using TV-L1 optical flow [159] with the attachment parameter $\lambda = 50$. We decided to use TV-L1 OF instead of the Lucas-Kanade approach used in Sec. 3.5.2 because it is more robust against parameter variation⁶. Then, we use the obtained results to create a mean oriented OF image pair and process it in a similar way to MORF descriptor to obtain Mean Oriented Optical Flow or MOOF (we evaluate the same parameters as MORF).

We also decided to compare the MORF descriptor against a pre-trained model for feature extraction. For that we use AlexNet, a convolutional neural network which won the 2012 ImageNet Challenge [172]. The neural network, which has 60 million parameters and 650,000 neurons, consists of five convolutional layers, some of which are followed by max-pooling layers, and three fully-connected layers with a final 1000-way softmax [105]. We aim to use this pre-trained model to extract features from the mean oriented image pair. However, the input for AlexNet are 227×227 RGB images. Thus, we transform and resize the MOR and MOOF image pairs into false-color representation images where the saturation represents the phase and the hue represents the angle (see Fig. 5.3b). Additionally, we also evaluate the apex frame of the “partial macro-expression” videos. We extract the feature representation from the seventh fully connected layer $fc7$. The extracted features of the MOR image pair, MOOF image pair and input image are called $Alex_{MOR}$, $Alex_{MOOF}$ and

⁶For more information we refer the reader to Sec. E.1

Alex₁ respectively. We also compare our results with the HOG descriptor. For each block produced by the HOG function there are 2×2 cells of $[8, 8]$ pixels with 9 orientation bins⁷. We extracted the features from the apex frame of the “partial macro-expression” videos.

For classification we trained an SVM using a RBF kernel. The hyperparameters of the SVM are tuned by performing 30 iterations of a Bayesian optimization process before applying the kernel function⁸. The results were tested by a 10-fold validation method.

Results

The classification results for different partial macro-expressions are presented in Fig. 5.8. Considering that the CK+ database is unbalanced, we test both the general and average per-class accuracy. As we can see from Fig. 5.8a, all the methods show a high accuracy when analysing complete and partial macro-expressions with half of the frames (between 80 – 90% of general accuracy for a complete video and between 70 – 80% of general accuracy for a video of duration $T = \frac{\text{apex}}{2}$). However, the accuracy of all methods decrease as the duration of the partial macro-expression decreases.

MORF and MOOF show a similar performance both for general and average per-class accuracy standards (except when $T = \frac{\text{apex}}{5}$) and they both have a better performance compared to the other methods⁹. For complete videos, the HOG features and the AlexNet ones showed to have a similar accuracy (except for Alex_{MOR}). Furthermore, HOG features showed a higher accuracy at $T = \frac{\text{apex}}{2}$ compared to the AlexNet ones but it has the lowest accuracy when $T = \frac{\text{apex}}{5}$. From the three sets of features extracted by AlexNet, the performance of Alex_{MOR} was superior to Alex_{MOOF} (except when $T = \frac{\text{apex}}{5}$) and Alex₁ has the lowest performance (except when $T = \frac{\text{apex}}{2}$) from the three methods.

The most misclassified facial expressions from the database were contempt, fear and disgust which are also the classes with least samples. Also, contempt was many times misclassified as happy which can be explained by the similarity of lip movement for both expressions (lip corner raised).

The features extracted using AlexNet (Alex_{MOR} and Alex_{MOOF}) were less accurate than the hand-crafted ones proposed in this work. One of the reason might be that the original AlexNet was trained to classify 1000 different classes of still images. That means that the layer *fc7* supposedly extracts the top-level features that the model has "discovered" during

⁷For a more detailed explanation about these parameters, the reader is referred to the web page of the function `extractHOGFeatures` for Matlab

⁸For more details about the training model, we refer the reader to Sec. D

⁹The results for MORF and MOOF in Fig. 5.8 correspond to the average value and standard deviation of the results obtained after testing different sets of parameters

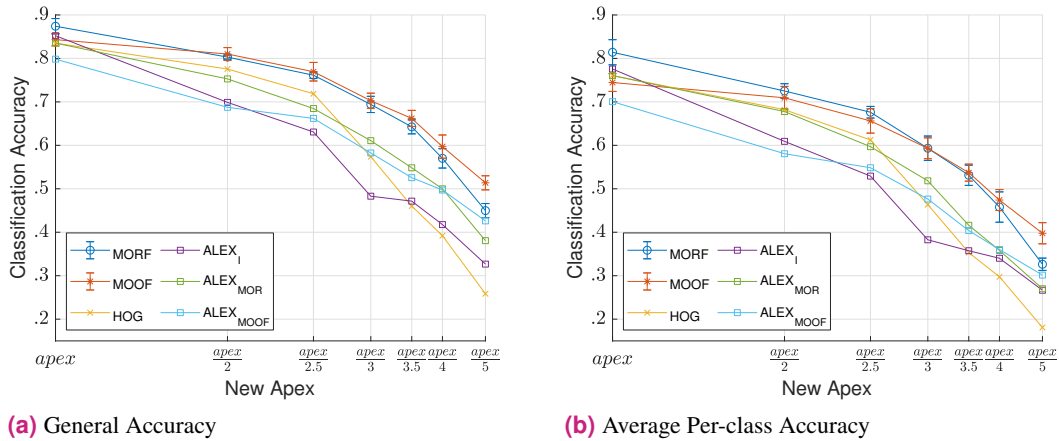


Figure 5.8.: Results of partial macro-expressions classification

its training that separate such different objects as kite, jellyfish or tiger [172]. However the synthetic images generated from MOR and MOOF image points might not be perceived as very different from each other according to AlexNet (specially when the new apex is short).

The HOG and Alex_I features were extracted only from the last frame of the partial macro-expression videos, thus their accuracy depends on how distinctive are the facial structural characteristics for the new apex. As we can see from Fig. 5.7, when the new apex is short, there are not many face variations compared to a neutral state (onset). What's more, neither of the methods measure any change or motion between frames.

Something worth noting is the hyperparameter optimization process time depended in great part of the dimensionality of the features. While MORF and MOOF feature histogram had a maximum length of 640 (see Sec. 5.2.1), AlexNet gave us a feature of length 4096 (given by the *fc7* layer) and HOG features had a length of 28224¹⁰.

In the end, the MORF and MOOF descriptors have shown to have overall better performances compared to the other methods. When we analyse the effects of the parameters on the descriptors among different values for the new apex (see Table. 5.1 and Table. 5.2), we can see that the accuracy of the descriptors doesn't change dramatically using different set of parameters. If we analyse the cumulative results of the different set of parameters (stacked bar in Fig. 5.9), we might see that overall for MORF, the best results are obtained when $G = 8$ and $O = [8, 10]$ (MORF_{8,8} and MORF_{8,10}), while for MOOF the best results are obtained when $G = 8$ and $O = 6$ (MOOF_{8,6}). We decide for the next experimentation step, to evaluate these two descriptors for real MEs.

¹⁰The length of the HOG features involve different aspects like image size, block size, number of orientation bins among others

Partial Macro-Expressions								
Parameters		MORF						
Grid Division	Orientation Bins	<i>apex</i>	$\frac{apex}{2}$	$\frac{apex}{2.5}$	$\frac{apex}{3}$	$\frac{apex}{3.5}$	$\frac{apex}{4}$	$\frac{apex}{5}$
$G = 6 \times 6$	$O = 6$	0.8029	0.7245	0.6820	0.5417	0.4987	0.4298	0.3133
	$O = 8$	0.8389	0.7484	0.6610	0.5883	0.5206	0.4470	0.3425
	$O = 10$	0.7611	0.7308	0.6831	0.6016	0.5274	0.4179	0.3116
$G = 8 \times 8$	$O = 6$	0.8213	0.7305	0.6618	0.5924	0.5246	0.4546	0.3355
	$O = 8$	0.8250	0.7187	0.6955	0.6179	0.5517	0.5074	0.3160
	$O = 10$	0.8359	0.6976	0.6736	0.6177	0.5640	0.4913	0.3390

Table 5.1.: Parameter evaluation results for MORF

Partial Macro-Expressions								
Parameters		MOOF						
Grid Division	Orientation Bins	<i>apex</i>	$\frac{apex}{2}$	$\frac{apex}{2.5}$	$\frac{apex}{3}$	$\frac{apex}{3.5}$	$\frac{apex}{4}$	$\frac{apex}{5}$
$G = 6 \times 6$	$O = 6$	0.7395	0.7170	0.6394	0.6146	0.5316	0.4740	0.4200
	$O = 8$	0.7275	0.6722	0.6289	0.5645	0.5161	0.4580	0.3791
	$O = 10$	0.7251	0.7265	0.6404	0.5810	0.5169	0.4540	0.3709
$G = 8 \times 8$	$O = 6$	0.7354	0.6992	0.6938	0.6274	0.5667	0.5204	0.4328
	$O = 8$	0.7651	0.6946	0.6451	0.5953	0.5447	0.4644	0.3848
	$O = 10$	0.7746	0.7467	0.6897	0.5762	0.5472	0.4739	0.3985

Table 5.2.: Parameter evaluation results for MOOF

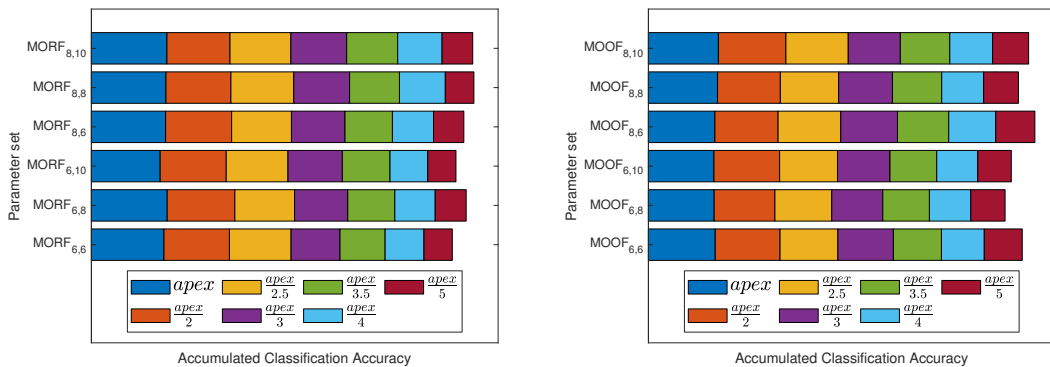


Figure 5.9.: Parameter variation of MORF and MOOF

5.3.2 Experiment 2: MORF vs MOOF for ME classification

In this step we want to test the classification accuracy on the descriptors who showed a better performance in the previous experiment: the MORF and MOOF descriptors. The aim of the experiment is to quickly evaluate the performance of both descriptors to classify ME sequences. The input are ME sequences from SMIC-HS and CASME II databases.

Database

For our experimentation, we selected two spontaneously elicited ME databases: the Spontaneous Micro-Expression Database (SMIC-HS) [113]¹¹ and the improved Chinese Academy of Sciences Micro-expression (CASME II) [230] (More details about both databases can be read in Sec. 1.4.2). The SMIC-HS database has 164 MEs labelled into 3 emotions: positive (51), negative (70) and surprise (43). The CASME II database has 246 MEs labelled into 5 emotions: happiness (32), disgust (63), surprise (25), repression (27) and others (99).

Evaluation procedure

The input are ME sequences cut from onset to apex from SMIC-HS and CASME II databases. The SMIC-HS database doesn't label the apex frame, thus the apex frame is inferred as the medium frame between onset and offset ($\text{apex} = \frac{\text{offset} + \text{onset} + 1}{2}$). We apply the face registration and cropping step described in Sec. 5.2. The cropped face sequence is processed using the Riesz pyramid to obtain the filtered quaternionic phase. For the spatio-temporal filtering we use the same parameters as the ones in Sec. 4.4.1. Considering that different sub-bands of an image could give different information about MEs, we decided to process the second, third and fourth level of the pyramid (see Fig. 3.6) to obtain the MOR image pairs (We don't process the first level since it has the information of the highest frequency sub-band and seem to carry an important amount of undesired noise).

To obtain the MOOF descriptor, we process the motion of the videos using TV-L1 optical flow [159]. We evaluate different values for the attachment parameter $\lambda = \{25, 50, 200\}$. Then we use the obtained results to create mean oriented OF image pairs and process them in a similar way to the MORF descriptor to obtain Mean Oriented Optical Flow or MOOF.

For classification we used the same SVM scheme described in Sec. 5.3.1. The results were tested by a 10-fold validation method.

¹¹We decided to use only the videos captured using the high speed camera

Results

The classification results for the SMIC-HS database can be seen in Table. 5.3 and the results for the CASME II can be seen in Table. 5.4. As we can see, for both datasets in any given set of parameters, MORF descriptor had a better accuracy compared to MOOF.

The MOOF descriptor, which in the previous experiment had one of the best performances, underperformed with both ME spontaneous datasets barely surpassing random guess (0.33 in SMIC-HS and 0.2 in CASME II). There are different factors that might explain these results. Firstly, the TV-L1 optical flow method used to extract the MOOF descriptor was a coarse-to-fine OF method. Coarse-to-fine means to simply build image pyramids for each image pair (by downsampling), apply the optical flow method on each layer of the pyramid and finally all the detected displacements are refined at the original scale. That means that all displacements (both coarse or fine) are fused, which would mean that subtler ME motion might be diminished by a larger macro-movements (face displacements, eye blinking, etc.). Secondly, the calculation of TV-L1 optical flow is heavily affected by its attachment parameter λ . This parameter determines the smoothness of the output, i.e. the smaller this parameter is, the smoother the obtained solution. Consequently for smaller values of λ the optical flow is underestimated for subtler motion. On the other hand, for bigger values of λ , the method becomes more sensitive to noise and the results are unstable flow fields [159] (which might explain why the accuracy results were better for MOOF at $\lambda = 25$ compared to $\lambda = 200$)¹².

The MORF results in the SMIC-HS database shows that the best performance was found in the third level of the Riesz pyramid, albeit when $G = 6$, and that the fourth level has the worst performance. This results are similar to the ones reported in Fig. 4.13a. However, the results in the CASME II show that, even though the two best accuracy scores can be found in the second level of the pyramid, there is not a clear advantage of choosing the second level of the pyramid over the third one (although the fourth level showed the worst performance). This goes in direct contradiction to what was reported in Fig. 4.13b. Furthermore, it is not possible to see any significant advantage for choosing a specific set of parameters (either for G or O) for the MORF descriptor.

As we can see, it is not possible to conclude whether one pyramid level gives the most relevant information about MEs for classification. It seems that, depending on the circumstances, extracting information from certain levels of the Riesz pyramid yield better results. In the next experiment we will test whether a combination of pyramid levels might improve the performance of our system.

¹²We further test the variation of λ in Sec. E.2

SMIC Database							
Parameters		MORF			MOOF		
Grid division	Orientation bins	Riesz Level			λ		
		2	3	4	25	50	200
$G = 6 \times 6$	$O = 6$	0.5000	0.6000	0.5259	0.4111	0.3667	0.3185
	$O = 8$	0.5519	0.5593	0.5222	0.4185	0.3556	0.3407
	$O = 10$	0.4963	0.5963	0.5407	0.4074	0.3778	0.3333
$G = 8 \times 8$	$O = 6$	0.5667	0.5370	0.4926	0.4000	0.3852	0.3852
	$O = 8$	0.5333	0.5222	0.4704	0.4148	0.3815	0.3185
	$O = 10$	0.5667	0.5556	0.4778	0.4148	0.4222	0.3185

Table 5.3.: MORF vs MOOF classification for the SMIC-HS database

CASME II Database							
Parameters		MORF			MOOF		
Grid division	Orientation bins	Riesz Level			λ		
		2	3	4	25	50	200
$G = 6 \times 6$	$O = 6$	0.5820	0.5466	0.4405	0.2419	0.2403	0.2409
	$O = 8$	0.5177	0.5305	0.5273	0.2450	0.2584	0.2223
	$O = 10$	0.5531	0.5466	0.4984	0.1741	0.2797	0.2247
$G = 8 \times 8$	$O = 6$	0.5498	0.5659	0.4920	0.2188	0.2246	0.1781
	$O = 8$	0.5209	0.5723	0.5498	0.2613	0.2722	0.2195
	$O = 10$	0.5916	0.5273	0.5498	0.2451	0.2366	0.1637

Table 5.4.: MORF vs MOOF classification for the CASME II database

5.3.3 Experiment 3: MORF variations

In this step we wanted to do a thorough classification test on the MORF descriptors and some variations of it. The aim of the experiment is to cross-validate the classification accuracy of different variations of the MORF descriptor and compare it to the state of the art. The input are ME sequences from SMIC-HS and CASME II databases.

Evaluation Procedure

We use the same type of input, databases, face registration and cropping algorithms and Riesz processing procedure as the previous experiment (see Sec. 5.3.2). We process the second, third and fourth level of the pyramid to obtain the features from the MOR image pairs.

We decided to test a couple variations of the MORF descriptors, namely, Fused MORF, Amplified MORF and Fused Amplified MORF (see Fig. 5.2.2)¹³. For the fused MORF descriptor we tested 3 combinations of Riesz pyramid levels: second and third levels (2&3), third and fourth levels (3&4) and second, third and fourth levels (2&3&4). For the Amplified

¹³Amplitude weighted and amplitude masked MORF are tested in Sec. E.3

MORF descriptor, we tested two values for the magnification factor ($\alpha = \{5, 10\}$). For the Amplified Fused MORF we use a combination of the previously mentioned parameters.

For classification we used the same SVM scheme described in Sec. 5.3.1. However, this time the results were tested by a Leave-One-Subject-Out (LOSO) cross-validation.

Results

The results for the SMIC-HS database can be seen in Table. 5.5. As we can see, the best results for the simple MORF descriptor are obtained at the third level of the pyramid, specially when the oriented phase has been amplified with an amplification factor $\alpha = 10$ (classification accuracy: 0.5895). For the fused MORF descriptor the best results are obtained when we mix the results of the second and third level of the pyramid, specially when the phase has been amplified with an amplification factor $\alpha = 5$ (classification accuracy: 0.5745). When we analyse the confusion matrix for the best result (see Fig. 5.10a), we can see that the most accurately recognized ME (“surprise”) is surprisingly the class with least samples in the database. Furthermore, we see that our trained model had difficulties to separate “positive” from “negative” MEs.

The results for the CASME database can be seen in Table. 5.6. As we can see, the best results for the simple MORF descriptor are obtained at the third level of the pyramid, however, the best results are obtained when oriented phase is not amplified (classification accuracy: 0.5650). For the fused MORF descriptor the best results are obtained when we mix the results of the second, third and fourth level of the pyramid, specially when the phase has been amplified with an amplification factor $\alpha = 5$ (classification accuracy: 0.6137). When we analyse the confusion matrix for the best result (see Fig. 5.10b), we can see that the most accurately recognized MEs are “surprise” and “others”. The most misclassified ME were “happiness” and “repression”. These results could be explained by the fact that the database is unbalanced (the number of samples for the classes “disgust” and “others” double and triple respectively the number of samples for “happiness”, “surprise” and “repression”). However, “surprise” has a great recognition rate regardless having significantly less samples compared to “disgust” and “others”.

Parameter Analysis: The results of the parameter evaluation for the SMIC-HS dataset can be found in Table. 5.7 and for the CASME II dataset can be found in Table. 5.8. We can observe that the results show a similar behaviour compared to the previously reported in Table. 5.5 (most of the highest results are obtained for MORF in the third level of the pyramid and for fused MORF when we combine the second and third levels of the pyramid specially when the oriented phase has been amplified) and Table. 5.6 (most of the highest results are obtained for MORF in the third level of the pyramid and for fused MORF when

SMIC-HS				
Descriptor	Pyramid Level	Non Amplified	Amplified MORF	
			$\alpha = 5$	$\alpha = 10$
MORF	2	0.5451	0.5669	0.5539
	3	0.5547	0.5733	0.5895
	4	0.4978	0.5051	0.4982
Fused MORF	2&3	0.5673	0.5745	0.5697
	3&4	0.5147	0.5176	0.5438
	2&3&4	0.5313	0.5309	0.5475

Table 5.5.: ME classification for SMIC-HS

CASME II				
Descriptor	Pyramid Level	Non Amplified	Amplified MORF	
			$\alpha = 5$	$\alpha = 10$
MORF	2	0.5364	0.5488	0.5489
	3	0.5650	0.5447	0.5078
	4	0.5374	0.5257	0.5278
Fused MORF	2&3	0.5499	0.5489	0.5221
	3&4	0.5696	0.6005	0.5654
	2&3&4	0.5754	0.6137	0.5940

Table 5.6.: ME classification for CASME II



Figure 5.10.: Confusion Matrices at the best recognition rate by the LOSO cross-validation

SMIC-HS Database Non-amplified							
Parameters		MORF			Fused MORF		
Grid division	Orientation bins	Riesz Levels					
		2	3	4	2&3	3&4	2&3&4
$G = 6 \times 6$	$O = 6$	0.5467	0.5632	0.4892	0.5547	0.5115	0.5426
	$O = 8$	0.5592	0.5531	0.5172	0.5572	0.5333	0.5378
	$O = 10$	0.5442	0.5547	0.4857	0.5455	0.5067	0.5087
$G = 8 \times 8$	$O = 6$	0.5459	0.5653	0.5071	0.5685	0.5119	0.5289
	$O = 8$	0.5499	0.5798	0.4692	0.5661	0.5378	0.5248
	$O = 10$	0.5374	0.5608	0.4840	0.5653	0.5289	0.5341
SMIC-HS Database Amplified $\alpha = 5$							
Parameters		Amplified MORF			Amplified Fused MORF		
Grid division	Orientation bins	Riesz Levels					
		2	3	4	2&3	3&4	2&3&4
$G = 6 \times 6$	$O = 6$	0.5661	0.5564	0.5046	0.5624	0.5107	0.5362
	$O = 8$	0.5749	0.5455	0.5261	0.5661	0.5410	0.5442
	$O = 10$	0.5806	0.5523	0.4881	0.5475	0.5139	0.5398
$G = 8 \times 8$	$O = 6$	0.5677	0.5851	0.5208	0.5830	0.5305	0.5277
	$O = 8$	0.5713	0.5721	0.5119	0.5830	0.5358	0.5406
	$O = 10$	0.5669	0.5612	0.4933	0.5758	0.5459	0.5370
SMIC-HS Database Amplified $\alpha = 10$							
Parameters		Amplified MORF			Amplified Fused MORF		
Grid division	Orientation bins	Riesz Pyramid Levels					
		2	3	4	2&3	3&4	2&3&4
$G = 6 \times 6$	$O = 6$	0.5564	0.5519	0.5091	0.5709	0.5131	0.5164
	$O = 8$	0.5600	0.5527	0.5285	0.5616	0.5402	0.5341
	$O = 10$	0.5483	0.5467	0.5006	0.5471	0.5160	0.5244
$G = 8 \times 8$	$O = 6$	0.5556	0.5891	0.4994	0.5875	0.5184	0.5277
	$O = 8$	0.5438	0.5786	0.4937	0.5648	0.5483	0.5515
	$O = 10$	0.5418	0.5689	0.5091	0.5661	0.5358	0.5382

Table 5.7.: Recognition rates with respect to different set of parameters for the SMIC-HS dataset

we combine the second, third and fourth levels of the pyramid no matter what level of amplification we use for the oriented phase). It is difficult to say whether a set of parameters is superior than other, but it seems that the best results are slightly better when $G = 8$ and $O = \{6, 8\}$. What's more the variance between results of a given level (or combination of levels) of the Riesz pyramid is very low.

State of the Art Comparison: Although we have already provided a summary of the results of ME classification in the state of the art (see Table. 2.3), not all of those results are comparable to ours since some methods used different metrics, datasets and evaluation methodologies. Therefore, we only directly compare our classification results with methods that follow the following conditions:

- They were tested in SMIC-HS and CASME II datasets (the complete dataset).
- They were testing ME recognition (not comparing between ME and non-MEs).

CASME II Database Non-amplified							
Parameters		MORF			Fused MORF		
Grid division	Orientation bins	Riesz Levels					
		2	3	4	2&3	3&4	2&3&4
$G = 6 \times 6$	$O = 6$	0.5340	0.5530	0.4902	0.5548	0.5465	0.5753
	$O = 8$	0.5179	0.5291	0.5285	0.5514	0.5509	0.5683
	$O = 10$	0.5249	0.5263	0.4990	0.5628	0.5463	0.5670
$G = 8 \times 8$	$O = 6$	0.5548	0.5561	0.5072	0.5776	0.5907	0.6094
	$O = 8$	0.5431	0.5634	0.5600	0.5844	0.5922	0.5969
	$O = 10$	0.5535	0.5307	0.5224	0.5501	0.5665	0.5727
CASME II Database Amplified $\alpha = 5$							
Parameters		Amplified MORF			Amplified Fused MORF		
Grid division	Orientation bins	Riesz Pyramid Levels					
		2	3	4	2&3	3&4	2&3&4
$G = 6 \times 6$	$O = 6$	0.5354	0.5511	0.4842	0.5620	0.5597	0.5728
	$O = 8$	0.5350	0.5267	0.5322	0.5701	0.5491	0.5798
	$O = 10$	0.5566	0.5416	0.5075	0.5680	0.5650	0.5925
$G = 8 \times 8$	$O = 6$	0.5590	0.5598	0.5078	0.5618	0.6029	0.6013
	$O = 8$	0.5561	0.5580	0.5660	0.5706	0.5966	0.6008
	$O = 10$	0.5543	0.5341	0.5221	0.5489	0.5668	0.5745
CASME II Database Amplified $\alpha = 10$							
Parameters		Amplified MORF			Amplified Fused MORF		
Grid division	Orientation bins	Riesz Pyramid Levels					
		2	3	4	2&3	3&4	2&3&4
$G = 6 \times 6$	$O = 6$	0.5552	0.5504	0.4964	0.5709	0.5634	0.5686
	$O = 8$	0.5340	0.5247	0.5223	0.5615	0.5382	0.5587
	$O = 10$	0.5359	0.5367	0.4976	0.5499	0.5600	0.5761
$G = 8 \times 8$	$O = 6$	0.5262	0.5558	0.5013	0.5628	0.5956	0.6052
	$O = 8$	0.5246	0.5637	0.5580	0.5764	0.5979	0.5907
	$O = 10$	0.5098	0.5086	0.5158	0.5047	0.5514	0.5587

Table 5.8.: Recognition rates with respect to different set of parameters for the CASME II dataset

- Evaluated ME classification accuracy (not F-scores).
- Implemented a Leave-One-Subject-Out (LOSO) cross-validation.

The results can be found in Table. 5.9. Among all comparative algorithms, the Fusion Motion Boundary Histograms (FMBH) proposed by [125] has the best recognition rate for both datasets (71.95% for SMIC-HS and 69.11% for CASME II). Our proposed approach, albeit it does not yield the best possible results, is still able to surpass several appearance and geometrical based descriptors in both datasets (such as STLBP-IP [89], LBP-TOP [142], MOP-LBP [216], MDMO [123] and Facial Dynamic Maps [229]). One thing to consider is that most of the methods presented in Table. 5.9 are based in very well known algorithms (LBP, HOG, optical flow). On the other hand, our proposed method is based on the monogenic signal, which it has not been very explored. As a matter of fact, comparing our method to [148] (the only other method based on the monogenic signal), the ME recognition performance is increased by 15.11%. Furthermore, our results are very similar to the ones obtained using convolutional neural networks [100] (which had to artificially augment the database). Finally, considering the complexity of the task at hand (the ME recognition rate of any given method didn't surpassed 72% of accuracy in neither database), our results are able to hold relatively well against the state of the art.

5.3.4 Discussion

For both the SMIC-HS and the CASME II dataset, we obtained the best results for the single level MORF descriptor (also for the amplified single level MORF in the case of the SMIC-HS dataset) when we extracted features from the 3rd level of the Riesz pyramid. These results coincide with the parameter evaluation results for ME spotting (see Sec. 4.4.3) which also deem the 3rd level of the pyramid as the one who gives the most information from MEs. Nevertheless, we obtained better results from the fused MORF descriptor which means that different levels of the pyramid contain relevant complementary information (second and third levels for the SMIC-HS dataset and second, third and fourth levels for the CASME II database). However, at this point we cannot confidently establish a set of rules for fusing pyramid levels for ME recognition. If we consider the fact that for ME spotting parameter evaluation in the CASME II dataset, the best results were obtained in both the third and fourth level of the pyramid (see Sec. 4.4.3), it could be argued that the best results for ME recognition can be obtained when we fuse the most relevant Riesz level (third level in SMIC-HS and fourth in CASME II) with the previous ones.

One possible cause of error in our calculations are facial macro-movements. Similarly to what was discussed in Sec. 4.4.4, when our method is unable to remove the effect of macro-movements, our system will register them as part of the facial motion. Considering

¹⁴The database has been artificially augmented

Micro Expression Classification Methods					
Feature Extraction		Classifier		Results	
Category	Approach	Category	Approach	SMIC-HS	CASME II
Appearance features	STCLQP [91]	Support Vector Machine	Linear Kernel	64.02%	58.39%
	LBP-SIP [217]			64.02%	–
	STLBP-IP [89]		Chi-square kernel	57.93%	59.51%
	STLBP-IIP [90]			63.41%	64.78%
	LBP [112]		LSVM	60.37%	64.78%
	HOG [112]			61.59%	63.97%
	HIGO [112]			68.29%	67.21%
	LBP-TOP [142]		RBF kernel	–	51.00%
	LBP-TOP & Intensity variation [153]			–	51.91%
	MOP-LBP [216]			–	45.75%
	LBP-SIP [217]			–	66.40%
	Riesz Wavelet [148]			–	46.15%
	MOP-LBP [216]		Polynomial kernel	50.00%	–
	2D Gabor Filter [240]	Others	SRC	–	64.88%
Geometrical features	RHPM [5]	Support Vector Machine	LibSVM	–	65.35%
	BI-WOOF + Riesz phase [119]		Linear kernel	68.29%	62.55%
	Optical Strain & Wiener Filter [115]			53.56%	–
	MDMO [123]		Polynomial kernel	58.97%	51.69%
	Facial Dynamics Map [229]		RBF kernel	54.88%	41.96%
	FMBH [125]			71.95%	69.11%
Hybrid Features	DLSTD [213]	Support Vector Machine	Non Specified	67.68%	65.44%
	OSW-LBP-TOP [116]		Polynomial kernel	63.95%	–
Learned Features	Feature Mapping [83]	Support Vector Machine	Non Specified	63.95%	–
	Learned [100]	Deep Learning	CNN & LSTM	–	60.98% ¹⁴
Proposed Approach	Amplified Fused MORF	Support Vector Machine	RBF kernel	58.95%	61.37%

Table 5.9.: ME recognition accuracy of comparable methods of the state of the art

that facial macro-movements (such as face translation) maintains the same orientation and phase over several frames, these macro movements are preserved when we create the MOR image pair (Eq. 5.1 and Eq. 5.2) and the MORF descriptor will wrongfully capture these motions as part of the ME.

Another possible cause of error is the face registration scheme used as pre-processing step for ME recognition. The first problem is that different authors will use different face registration algorithms. Since each method will offer a different level of accuracy, response time and stability, the results of an ME recognition framework might be affected by the use of different face registration methods. The second problem is the feature correspondence problem. These methods normally track the face by extracting and matching features between consecutive frames. During the feature matching stage, there is the risk of finding ambiguous matches which over time cause the feature to flicker. Furthermore, even if the correspondence problem produce a displacement in a sub-pixel resolution, when the localized face is cropped it cuts the image in pixel resolution. That means that a cropped image sequence might be flickering by one pixel to any direction which might be wrongfully interpreted by our system as a micro-movement.

5.4 Chapter Conclusions

In this chapter, we presented a facial micro-expression recognition method based on the quaternionic representation of the multi-scale oriented phase element from the Riesz pyramid. Our main contributions are:

- We adapt the oriented phase difference calculated in previous chapter to create a simple micro-expression representation. We produce an image pair model that models the temporal evolution of MEs using the phase and orientation components of the monogenic signal while reducing the effects of image noise.
- We propose a novel weighted descriptor that extracts orientation and phase information features with low dimensionality.

A series of experiments showed that our classification method is able to compete with other well-known and more developed methods from the state of the art. Furthermore, The results of our parameter analysis experiment showed that this method is robust to changes in parameters. The results of our experiment also showed that, the results can be improved by simply combining and amplifying the oriented phase from different levels of the Riesz pyramid, suggesting that a greater recognition rate can be achieved by further development and experimentation.

Conclusions and Perspectives

In this work, we propose a novel framework for subtle motion and micro-expression analysis. After describing the nature of micro-expression and challenges of studying it, we offer an extensive review of the state of the art methods of the target. We then propose to use the multi-scale components derived from the Riesz pyramid as the basis for subtle motion analysis. We first propose a method for general subtle motion spotting. Then, we proceed to adapt the previous method for micro-expression spotting. Finally, we propose a feature extraction method for micro-expression description. The micro-expression spotting and feature extraction methods are evaluated and tested in publicly available micro-expression databases, while for the subtle motion method we create our own database of subtle motions.

The main contributions are summarized hereafter:

- **Framework for subtle motion analysis using the Riesz Pyramid:** From the approximated multi-scale monogenic obtained from the Riesz pyramid, the quaternionic phase difference is extracted from two consecutive frames for subtle motion analysis and spotting. The Riesz pyramid is constructed using a Laplacian pyramid representation and an approximate Riesz transform to reduce the computation time. A temporal filtering scheme is designed which can enhance motions of interest without producing delays or undesired artifacts. In addition, a masking method based on the image local amplitude is proposed in order to isolate regions where subtle motions might take place while masking areas sensible to image noise. The quaternionic phase difference is turned into a 1-D signal which is used for temporal and frequency analysis of subtle motions. Experiments show that our proposed method is more robust against noise than classical approaches like LBP, HOG and optical flow while requiring much less computation time than those traditional texture and motion features.
- **Framework for micro-expression spotting using the Riesz Pyramid:** The filtering scheme and amplitude masking method proposed for the subtle motion analysis framework were adapted to analyse facial micro-expressions. A heuristic micro-expression spotting method was proposed which analyses the motion in local face regions of interest. This method is able to separate real micro-expressions from subtle eye movements decreasing the quantity of possible false positives. Experiments show that our method is robust against changes in parameters and surpasses the accuracy of comparable methods in the state of the art.

- **Framework for micro-expression classification:** A facial micro-expression recognition method based on the quaternionic oriented phase representation of the multi-scale monogenic signal is proposed. The temporal evolution of a micro-expression is modelled as an image pair that contains the mean oriented phase component of the monogenic signal which aims to reduce the effects of image noise. Furthermore, this model is transformed into an easily-adaptable and low-dimensional feature descriptor which can also represent the amplification of the oriented phase and also the fusion of multi-scale oriented phase representation. Experiments show that our method is robust against changes in parameters and some preliminary results are given showing that the proposed descriptor is promising.

Future Work

Some interesting future works can be derived from this thesis. Firstly, a better facial tracking system should be integrated to our framework. Although, our methodology was tested in two datasets which faces were captured in controlled environments (SMIC [113] and CASME II [230]), in a real life application we might have to deal with different head-poses, occlusions and illumination changes (such was the case when we worked in the project DAME as described in Sec. A). Fortunately, there are some interesting face tracking methods that have appeared recently in the state of the art which are publicly available for different operating systems and programming languages and also are easily adaptable (such as [OpenPose](#)). Furthermore, the obtained face registration coordinates should be processed over time in order to compensate the feature correspondence problem and create smoother trajectories (so the cropped faces do not jitter).

Secondly, our micro-expression spotting method, which proved to perform well in micro-expression databases comprised of short videos, has yet to be tested in a database of longer videos. Currently, there is only one publicly available database (CAS(ME)² [168]) which has not only longer videos but it also captures both micro and macro-expressions. Thus, several improvements have to be done to our spotting method. Firstly we need to find a way to differentiate between micro and macro expressions. In the case of our peak analysis method, we would have to exploit the differences between peak heights (which represent motion intensity) and peak widths (which represents motion duration). Secondly, the peak analysis method (see Sec. 4.3) would have to be changed since its peak selection method depends on thresholds (Eq. 4.9 and Eq. 4.10) calculated under the assumption that the biggest motion detected in the eyebrow or mouth area corresponds to a micro-expression (this leads to some errors as discussed in Sec. 4.4.4). We are currently doing some experiments where we process the feature signals using wavelets which might help to indicate the duration and the intensity of the spotted events which might help us for both cases. Furthermore, we could improve the spotting method by training a model which could classify spotted events

as either ME or non-ME. What's more, we could apply a Bayesian optimization scheme, similar to the one used for hyperparameter tuning for machine learning, to automatically determine the best parameters for micro-expression spotting.

Thirdly, the micro-expression recognition framework should be further improved. Although our proposed classification method yield some interesting initial results, when we compare it to the current state of the art we realize there is room for more improvement. The first improvement would be to adapt our micro-expression spotting algorithm as a pre-processing step for classification. During the development of our work we observed that some temporal labels (onset, apex and offset) for the micro-expression databases were not accurate (and the SMIC database does not provide apex labels). Thus, we could integrate our spotting method to detect the new apex before we model the mean oriented Riesz image pairs. Also, we should consider to, instead of extracting the quaternionic oriented phase between consecutive frames, to extract it between onset and apex, which would represent a bigger spatial displacement.

Fourthly, using another machine learning method such as deep learning might result in obtaining better micro-expression recognition rates. However, training a convolutional neural network with the current limited data might result in undesired overfitting. This could be solved in two different ways: Either by using a pre-trained model as the starting point for a new model (transfer learning) or by adding more data. The second task specifically, could be achieved in the future by implementing data augmentation techniques, creating new spontaneous micro-expressions databases (like the one that could potentially be obtained in the project DAME) or by synthetically modelling and generating realistic micro-expressions.

Fifthly, the Riesz pyramid has demonstrated to be a powerful tool for motion analysis and it could be used to create a novel motion estimation technique. Phase-based motion estimation techniques have been already proposed in the past [4, 71, 77], however, a method based on the approximate Riesz pyramid would calculate the oriented phase difference in a faster and less over-complete manner. Furthermore, none of the aforementioned methods take into consideration the effect of noise in areas of low local amplitude. Thus, an effective Riesz based motion estimation technique should use the local amplitude as a constraint (either as a weighting factor or as a mask) in order to avoid to erroneously model phase noise as motion.

Lastly, other alternatives to the Riesz pyramid should also be considered for subtle motion analysis. As discussed in Sec. 5.1.1, one important limitation of the monogenic signal is that its formulation assumes images as intrinsically one-dimensional signals, thus, it can correctly calculate the dominant orientation of straight lines and edges but it fails to determine intrinsically two-dimensional patterns such as corners and junctions. Thus a different formulation for both intrinsically one-dimensional and two-dimensional signals should be looked upon (like one proposed by [219]). However, such formulation should also

be implemented with an approximate transformer in order to keep the speed advantages of our proposed formulation (see Sec. 3.2.4).

Bibliography

- [1] N. K. El Abbadi and A. A. A. Qazzaz. “Detection and Segmentation of Human Face”. In: *IJARCCCE* (Feb. 2015), pp. 90–94 (cit. on p. 19).
- [2] S. J. Ahn, J. Bailenson, J. Fox, and M. Jabon. “Using automated facial expression analysis for emotion and behavior prediction”. In: *The Routledge Handbook of Emotions and Mass Media* (2010), p. 349 (cit. on p. 23).
- [3] A. R. Ajaya, P. Petchimuthu, and V.K. Kavitha. “Emotion Analysis using Thermal Images based on Kernel Eigen Spaces”. In: *IJCA Proceedings on Emerging Technology Trends on Advanced Engineering Research - 2012 ICETT.2* (Jan. 2013), pp. 41–45 (cit. on p. 26).
- [4] M. Alessandrini, O. Bernard, A. Basarab, and H. Liebgott. “Multiscale optical flow computation from the monogenic signal”. In: *IRBM. Digital Technologies for Healthcare* 34.1 (Feb. 2013), pp. 33–37 (cit. on p. 121).
- [5] B. Allaert, I. M. Bilasco, and C. Djeraba. “Consistent Optical Flow Maps for Full and Micro Facial Expression Recognition”. In: Feb. 2017 (cit. on pp. 24, 39, 93, 116).
- [6] C. H. Anderson, J. R. Bergen, P. J. Burt, and J. M. Ogden. *Pyramid Methods in Image Processing*. 1984 (cit. on pp. xi, 48).
- [7] Carlos Arango. “Analysis and detection of facial micro-expressions using the Riesz pyramid”. In: *Journée Visage, geste, action et comportement*. TELECOM ParisTech, Dec. 2017 (cit. on p. 179).
- [8] Carlos Andres Arango, Olivier Alata, Rémi Emonet, Anne-Claire Legrand, and Hubert Konik. “Subtle Motion Analysis and Spotting using the Riesz Pyramid”. In: *Proceedings of the 13th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications - Volume 5: VISAPP, INSTICC*. SciTePress, 2018, pp. 446–454 (cit. on p. 179).
- [9] M. Arif-Rahu and M. J. Grap. “Facial expression and pain in the critically ill non-communicative patient: State of science review”. In: *Intensive & critical care nursing : the official journal of the British Association of Critical Care Nurses* 26.6 (Dec. 2010), pp. 343–352 (cit. on p. 1).

- [10] A. Asthana, S. Zafeiriou, S. Cheng, and M. Pantic. “Robust Discriminative Response Map Fitting with Constrained Local Models”. In: *2013 IEEE Conference on Computer Vision and Pattern Recognition*. June 2013, pp. 3444–3451 (cit. on p. 20).
- [11] A. Azcarate, F. Hageloh, K. Sande, and R. Valenti. “Automatic facial emotion recognition”. In: *Universiteit van Amsterdam* (June 2005) (cit. on pp. 21, 23, 31).
- [12] G. Balakrishnan, F. Durand, and J. Guttag. “Detecting Pulse from Head Motions in Video”. In: *2013 IEEE Conference on Computer Vision and Pattern Recognition*. June 2013, pp. 3430–3437 (cit. on p. 41).
- [13] J. L. Barron, D. J. Fleet, and S. S. Beauchemin. “Performance of optical flow techniques”. en. In: *International Journal of Computer Vision* 12.1 (Feb. 1994), pp. 43–77 (cit. on p. 65).
- [14] M. S. Bartlett, J. C. Hager, P. Ekman, and T. J. Sejnowski. “Measuring facial expressions by computer image analysis”. eng. In: *Psychophysiology* 36.2 (Mar. 1999), pp. 253–263 (cit. on p. 31).
- [15] M. S. Bartlett, G. Littlewort, I. Fasel, and J. R. Movellan. “Real Time Face Detection and Facial Expression Recognition: Development and Applications to Human Computer Interaction.” In: *Conference on Computer Vision and Pattern Recognition Workshop, 2003. CVPRW '03*. Vol. 5. June 2003, pp. 53–53 (cit. on p. 31).
- [16] G. Bebis, A. Gyaourova, S. Singh, and I. Pavlidis. “Face recognition by fusing thermal infrared and visible imagery”. In: *Image and Vision Computing* 24.7 (July 2006), pp. 727–742 (cit. on p. 33).
- [17] A. Bertholon. “Développement d’une méthode d’exploration des micro-expressions faciales dans l’éveil de coma des patients cérébrolésés graves”. anglais. Thèse d’exercice. France: Université Jean Monnet (Saint-Étienne). Faculté de médecine Jacques Lisfranc, 2018 (cit. on pp. 145, 146, 152).
- [18] A. Bertholon, C. Arango Duque, O. Alata, et al. “Validation in healthy subjects of a clinical protocol for the evaluation of facial micro-expressions in severely brain injured patients awakening from coma”. In: *Annals of Physical and Rehabilitation Medicine* 61 (2018). 12th World Congress of the International Society of Physical and Rehabilitation Medicine. Paris. 8-12 July 2018, e426 (cit. on p. 179).
- [19] V. Bettadapura. “Face Expression Recognition and Analysis: The State of the Art”. In: *arXiv:1203.6722 [cs]* (Mar. 2012). arXiv: 1203.6722 (cit. on pp. vii, 2).
- [20] S. Bharadwaj, T. I. Dhamecha, M. Vatsa, and R. Singh. “Computationally Efficient Face Spoofing Detection with Motion Magnification”. In: *2013 IEEE Conference on Computer Vision and Pattern Recognition Workshops*. June 2013, pp. 105–110 (cit. on p. 49).
- [21] Bioinformatics.Org. *Hidden Markov Model* (cit. on p. 32).

- [22] D. Borza, R. Danescu, R. Itu, and A. Darabant. “High-Speed Video System for Micro-Expression Detection and Recognition”. eng. In: *Sensors (Basel, Switzerland)* 17.12 (Dec. 2017) (cit. on pp. 21, 22, 26, 29, 30, 32, 38).
- [23] S. Bouaziz, Y. Wang, and M. Pauly. “Online Modeling for Realtime Facial Animation”. In: *ACM Trans. Graph.* 32.4 (July 2013), 40:1–40:10 (cit. on p. 20).
- [24] C. P. Bridge. “Introduction To The Monogenic Signal”. In: *arXiv:1703.09199 [cs]* (Mar. 2017). arXiv: 1703.09199 (cit. on pp. 53, 54, 95).
- [25] J. Brownlee. *Boosting and AdaBoost for Machine Learning*. en-US. Apr. 2016 (cit. on p. 31).
- [26] P. Burt and E. Adelson. “The Laplacian Pyramid as a Compact Image Code”. In: *IEEE Transactions on Communications* 31.4 (Apr. 1983), pp. 532–540 (cit. on p. 48).
- [27] C. Busso, Z. Deng, Se. Yildirim, et al. “Analysis of Emotion Recognition Using Facial Expressions, Speech and Multimodal Information”. In: *Proceedings of the 6th International Conference on Multimodal Interfaces. ICMI '04*. New York, NY, USA: ACM, 2004, pp. 205–211 (cit. on p. 23).
- [28] D. Chai and K. N. Ngan. “Face segmentation using skin-color map in videophone applications”. In: *IEEE Transactions on Circuits and Systems for Video Technology* 9.4 (June 1999), pp. 551–564 (cit. on p. 19).
- [29] R. Chaudhry, A. Ravich, G. Hager, and R. Vidal. “Histograms of oriented optical flow and binet-cauchy kernels on nonlinear dynamical systems for the recognition of human actions”. In: *In IEEE Conference on Computer Vision and Pattern Recognition (CVPR. 2009)* (cit. on p. 23).
- [30] S. Yu Chekmenev, H. Rara, and Aly A. Farag. “Non-contact, wavelet-based measurement of vital signs using thermal imaging”. In: *The first international conference on graphics, vision, and image processing (GVIP), Cairo, Egypt. 2005*, pp. 107–112 (cit. on p. 153).
- [31] J. G. Chen, N. Wadhwa, Y.-J. Cha, et al. “Modal identification of simple structures with high-speed video using motion magnification”. In: *Journal of Sound and Vibration* 345 (June 2015), pp. 58–71 (cit. on pp. 41, 49).
- [32] S. W. Chew, P. Lucey, S. Lucey, et al. “Person-independent facial expression detection using Constrained Local Models”. In: *2011 IEEE International Conference on Automatic Face Gesture Recognition and Workshops (FG 2011)*. Mar. 2011, pp. 915–920 (cit. on p. 23).
- [33] T. F. Clark, P. Winkielman, and D. N. McIntosh. “Autism and the extraction of emotion from briefly presented facial expressions: stumbling at the first step of empathy”. eng. In: *Emotion (Washington, D.C.)* 8.6 (Dec. 2008), pp. 803–809 (cit. on pp. 1, 8).

- [34] I. Cohen, A. Garg, and T. S. Huang. “Emotion Recognition from Facial Expressions using Multilevel HMM”. In: *In Neural Information Processing Systems*. 2000 (cit. on pp. 23, 32).
- [35] I. Cohen, N. Sebe, A. Garg, L. S. Chen, and T. S. Huang. “Facial expression recognition from video sequences: temporal and static modeling”. In: *Computer Vision and Image Understanding*. Special Issue on Face Recognition 91.1–2 (July 2003), pp. 160–187 (cit. on pp. 23, 31, 32).
- [36] T. F. Cootes, G. J. Edwards, and C. J. Taylor. “Active appearance models”. en. In: *Computer Vision — ECCV’98*. Ed. by Hans Burkhardt and Bernd Neumann. Lecture Notes in Computer Science 1407. Springer Berlin Heidelberg, June 1998, pp. 484–498 (cit. on p. 20).
- [37] T. F. Cootes, C. J. Taylor, D. H. Cooper, and J. Graham. “Active Shape Models-Their Training and Application”. In: *Computer Vision and Image Understanding* 61.1 (Jan. 1995), pp. 38–59 (cit. on p. 20).
- [38] C. A. Corneanu, M. Oliu, J. F. Cohn, and S. Escalera. “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* PP.99 (2016), pp. 1–1 (cit. on pp. vii, ix, 2, 6, 17, 23, 24, 32–34).
- [39] D. Cristinacce and T. F. Cootes. “Feature Detection and Tracking with Constrained Local Models”. en. In: British Machine Vision Association, 2006, pp. 95.1–95.10 (cit. on p. 20).
- [40] C. Crivelli and A. J. Fridlund. “Facial Displays Are Tools for Social Influence”. English. In: *Trends in Cognitive Sciences* 22.5 (May 2018), pp. 388–399 (cit. on p. 6).
- [41] E. Cuevas, D. Zaldívar, M. Pérez-Cisneros, H. Sossa, and V. Osuna. “Block matching algorithm for motion estimation based on Artificial Bee Colony (ABC)”. In: *Applied Soft Computing*. Swarm intelligence in image and video processing. 13.6 (June 2013), pp. 3047–3059 (cit. on p. 45).
- [42] S. Cun. *Dual tvl1 optical flow matlab version*. original-date: 2017-02-14T09:16:37Z. Sept. 2018 (cit. on p. 69).
- [43] N. Dalal and B. Triggs. “Histograms of oriented gradients for human detection”. In: *IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2005. CVPR 2005*. Vol. 1. June 2005, 886–893 vol. 1 (cit. on pp. 25, 65).
- [44] C. Darwin and P. Prodger. *The Expression of the Emotions in Man and Animals*. en. Google-Books-ID: TFRtLZSHMcYC. Oxford University Press, 1998 (cit. on p. 6).
- [45] B. Davis. *Facial Expressions in Nonverbal Communication: Importance & Explanation*. June 2015 (cit. on p. 1).

- [46] A. K. Davison, C. Lansley, N. Costen, K. Tan, and M. H. Yap. “SAMM: A Spontaneous Micro-Facial Movement Dataset”. In: *IEEE Transactions on Affective Computing* PP.99 (2016), pp. 1–1 (cit. on pp. 10, 13, 14).
- [47] A. K. Davison, C. Lansley, C. C. Ng, K. Tan, and M. H. Yap. “Objective Micro-Facial Movement Detection Using FACS-Based Regions and Baseline Evaluation”. In: *arXiv preprint arXiv:1612.05038* (2016) (cit. on pp. 21, 28, 30).
- [48] A. K. Davison, C. Lansley, C. C. Ng, K. Tan, and M. H. Yap. “Objective Micro-Facial Movement Detection Using FACS-Based Regions and Baseline Evaluation”. In: *ResearchGate* (Dec. 2016) (cit. on p. 90).
- [49] A. Demertzi, S. Laureys, and M. Boly. “Coma, persistent vegetative states, and diminished consciousness”. In: *Encyclopedia of consciousness* (2009), pp. 147–156 (cit. on pp. 145, 146).
- [50] M. M. F. Donia, A. A. A. Youssif, and A. Hashad. “Spontaneous Facial Expression Recognition Based on Histogram of Oriented Gradients Descriptor”. In: *Computer and Information Science* 7.3 (July 2014) (cit. on pp. 25, 31, 38).
- [51] A. Dosovitskiy, P. Fischer, E. Ilg, et al. “FlowNet: Learning Optical Flow With Convolutional Networks”. In: 2015, pp. 2758–2766 (cit. on p. 46).
- [52] S. Du, Y. Tao, and A. M. Martinez. “Compound facial expressions of emotion”. In: *Proceedings of the National Academy of Sciences of the United States of America* 111.15 (Apr. 2014), E1454–E1462 (cit. on p. 6).
- [53] C. A. Duque, O. Alata, R. Emonet, A. C. Legrand, and H. Konik. “Micro-Expression Spotting Using the Riesz Pyramid”. In: *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*. Mar. 2018, pp. 66–74 (cit. on p. 179).
- [54] Eilert-Akademie fur emotionale Intelligenz. *7 basic emotions*. [Online; accessed June 18, 2018]. 2012 (cit. on p. 6).
- [55] P. Eisert and B. Girod. “Facial Expression Analysis for Model-Based Coding of Video Sequences”. In: *In Proceedings Picture Coding Symposium (PCS 97)*. 1997, pp. 33–38 (cit. on p. 20).
- [56] P. Ekman. “Basic Emotions”. en. In: *Handbook of Cognition and Emotion*. Wiley-Blackwell, 2005, pp. 45–60 (cit. on p. 7).
- [57] P. Ekman. “Facial Clues to Deceit”. en. In: *Telling Lies: Clues to Deceit in the Marketplace, Politics, and Marriage*. Google-Books-ID: 7I_wDDfrwCgC. W.W. Norton, 2001, pp. 123–161 (cit. on p. 7).
- [58] P. Ekman. “Facial Expressions”. en. In: *Handbook of Cognition and Emotion*. Wiley-Blackwell, 2005, pp. 301–320 (cit. on pp. viii, 6).
- [59] P. Ekman. *Telling Lies: Clues to Deceit in the Marketplace, Politics, and Marriage*. English. 1 edition. New York: W. W. Norton & Company, Jan. 2009 (cit. on p. 8).

- [60] P. Ekman and W. Friesen. “Facial action coding system: a technique for the measurement of facial movement.” In: *Consulting Psychologists, San Francisco* (1978) (cit. on pp. viii, 9, 10).
- [61] P. Ekman and W. V. Friesen. “Constants across cultures in the face and emotion”. In: *Journal of Personality and Social Psychology* 17.2 (1971), pp. 124–129 (cit. on p. 6).
- [62] P. Ekman and W. V. Friesen. “Nonverbal leakage and clues to deception”. eng. In: *Psychiatry* 32.1 (Feb. 1969), pp. 88–106 (cit. on pp. viii, 7, 93).
- [63] P. Ekman and W. V. Friesen. *Unmasking the Face: A Guide to Recognizing Emotions From Facial Expressions*. English. Cambridge, MA: Malor Books, Apr. 2015 (cit. on p. 8).
- [64] P. Ekman and E. L. Rosenberg. “Smiles when Lying”. en. In: *What the Face Reveals: Basic and Applied Studies of Spontaneous Expression Using the Facial Action Coding System (FACS)*. Oxford University Press, Apr. 2005, pp. 201–216 (cit. on pp. vii, 1).
- [65] P. Ekman and E. L. Rosenberg. *What the Face Reveals: Basic and Applied Studies of Spontaneous Expression Using the Facial Action Coding System*. Inglés. 2nd ed. Oxford ; New York: Oxford University Press, Feb. 2005 (cit. on p. 78).
- [66] P. Ekman and E. L. Rosenberg. *What the Face Reveals: Basic and Applied Studies of Spontaneous Expression Using the Facial Action Coding System (FACS)*. en. Oxford University Press, Apr. 2005 (cit. on pp. viii, 6, 8, 10).
- [67] J. Endres and A. Laidlaw. “Micro-expression recognition training in medical students: a pilot study”. In: *BMC Medical Education* 9 (July 2009), p. 47 (cit. on pp. 1, 41).
- [68] S. S. Farfade, M. Saberian, and L.-J. Li. “Multi-view Face Detection Using Deep Convolutional Neural Networks”. In: *arXiv:1502.02766 [cs]* (Feb. 2015). arXiv: 1502.02766 (cit. on pp. 18, 19).
- [69] B. Fasel and J. Luetttin. “Automatic facial expression analysis: a survey”. In: *Pattern Recognition* 36.1 (Jan. 2003), pp. 259–275 (cit. on pp. vii, 2, 17).
- [70] J. Fei and I. Pavlidis. “Analysis of breathing air flow patterns in thermal imaging”. eng. In: *Conference proceedings: ... Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Annual Conference* 1 (2006), pp. 946–952 (cit. on p. 153).
- [71] M. Felsberg. “Optical Flow Estimation from Monogenic Phase”. en. In: *Complex Motion*. Ed. by Bernd Jähne, Rudolf Mester, Erhardt Barth, and Hanno Schar. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2007, pp. 1–13 (cit. on p. 121).
- [72] M. Felsberg and G. Sommer. “The monogenic signal”. In: *IEEE Transactions on Signal Processing* 49.12 (Dec. 2001), pp. 3136–3144 (cit. on pp. 52, 53, 95).

- [73] D. Fernández-Espejo and A. M. Owen. “Detecting awareness after severe brain injury”. In: *Nature Reviews Neuroscience* 14 (2013), pp. 801–809 (cit. on p. 145).
- [74] D. J. Fleet and A. D. Jepson. “Computation of component image velocity from local phase information”. en. In: *International Journal of Computer Vision* 5.1 (Aug. 1990), pp. 77–104 (cit. on pp. 45, 50).
- [75] M. G. Frank, C. J. Maccario, and V. Govindaraju. “Behavior and Security”. en. In: *Protecting Airline Passengers in the Age of Terrorism*. Ed. by Paul Seidenstat and Francis X. Splane. ABC-CLIO, 2009, pp. 86–106 (cit. on p. 8).
- [76] T. R. Gault and A. A. Farag. “A Fully Automatic Method to Extract the Heart Rate from Thermal Video”. In: *2013 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. June 2013, pp. 336–341 (cit. on p. 153).
- [77] T. Gautama and M.M. Van Hulle. “A phase-based approach to the estimation of the optical flow field using spatial filtering”. In: *IEEE Transactions on Neural Networks* 13.5 (Sept. 2002), pp. 1127–1136 (cit. on pp. 45, 50, 121).
- [78] Joseph T. Giacino, S. Ashwal, N. Childs, et al. “The minimally conscious state: definition and diagnostic criteria”. eng. In: *Neurology* 58.3 (Feb. 2002), pp. 349–353 (cit. on p. 146).
- [79] Y. Guo, C. Xue, Y. Wang, and M. Yu. “Micro-expression recognition based on CBP-TOP feature with ELM”. In: *Optik - International Journal for Light and Electron Optics* 126.23 (Dec. 2015), pp. 4446–4451 (cit. on pp. 26, 39, 93).
- [80] H. Tao and T.S. Huang. “Explanation-based facial motion tracking using a piecewise Bezier volume deformation model”. In: *IEEE Comput. Soc*, 1999, pp. 611–617 (cit. on p. 20).
- [81] D. Habbal, O. Gosseries, Q. Noirhomme, et al. “Volitional electromyographic responses in disorders of consciousness”. eng. In: *Brain Injury* 28.9 (2014), pp. 1171–1179 (cit. on p. 146).
- [82] W. R. H. Hamilton. “On quaternions; or on a new system of imaginaries in algebra”. In: *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science* 29.192 (Aug. 1846), pp. 113–122 (cit. on p. 155).
- [83] J. He, J.-F. Hu, X. Lu, and W.-S. Zheng. “Multi-task Mid-level Feature Learning for Micro-expression Recognition”. In: *Pattern Recogn.* 66.C (June 2017), pp. 44–52 (cit. on pp. 32, 40, 93, 116).
- [84] S. He, S. Wang, W. Lan, H. Fu, and Q. Ji. “Facial Expression Recognition Using Deep Boltzmann Machine from Thermal Infrared Images”. In: *2013 Humaine Association Conference on Affective Computing and Intelligent Interaction (ACII)*. Sept. 2013, pp. 239–244 (cit. on pp. 32, 153).

- [85] B. Hernández, G. Olague, R. Hammoud, L. Trujillo, and E. Romero. “Visual learning of texture descriptors for facial expression recognition in thermal imagery”. In: *Computer Vision and Image Understanding*. Special issue on Advances in Vision Algorithms and Systems beyond the Visible Spectrum 106.2–3 (May 2007), pp. 258–269 (cit. on pp. 20, 25).
- [86] B. K.P. Horn and B. G. Schunck. *Determining Optical Flow*. Tech. rep. Cambridge, MA, USA: Massachusetts Institute of Technology, 1980 (cit. on pp. 23, 45).
- [87] C. Hu, Y. Chang, R. Feris, and M. Turk. “Manifold Based Analysis of Facial Expression”. In: *Conference on Computer Vision and Pattern Recognition Workshop, 2004. CVPRW '04*. June 2004, pp. 81–81 (cit. on p. 23).
- [88] C. Huang, H. Ai, Y. Li, and S. Lao. “High-Performance Rotation Invariant Multiview Face Detection”. In: *IEEE Trans. Pattern Anal. Mach. Intell.* 29.4 (Apr. 2007), pp. 671–686 (cit. on p. 18).
- [89] X. Huang, S. J. Wang, G. Zhao, and M. Pietikäinen. “Facial Micro-Expression Recognition using Spatiotemporal Local Binary Pattern with Integral Projection”. In: () (cit. on pp. 26, 38, 93, 115, 116).
- [90] X. Huang, S. Wang, X. Liu, et al. “Spontaneous Facial Micro-Expression Recognition using Discriminative Spatiotemporal Local Binary Pattern with an Improved Integral Projection”. In: *ResearchGate* (Aug. 2016) (cit. on pp. 26, 31, 38, 93, 116).
- [91] X. Huang, G. Zhao, X. Hong, W. Zheng, and M. Pietikäinen. “Spontaneous Facial Micro-expression Analysis Using Spatiotemporal Completed Local Quantized Patterns”. In: *Neurocomput.* 175.PA (Jan. 2016), pp. 564–578 (cit. on pp. 26, 38, 93, 116).
- [92] S. V. Ioannou, A. T. Raouzaïou, V. A. Tzouvaras, et al. “Emotion recognition through facial expression analysis based on a neurofuzzy network”. In: *Neural Networks. Emotion and Brain* 18.4 (May 2005), pp. 423–435 (cit. on p. 31).
- [93] S. Ioannou, V. Gallese, and A. Merla. “Thermal infrared imaging in psychophysiology: Potentialities and limits”. en. In: *Psychophysiology* 51.10 (Oct. 2014), pp. 951–963 (cit. on p. 153).
- [94] M. Irani and P. Anandan. “About Direct Methods”. In: *Proceedings of the International Workshop on Vision Algorithms: Theory and Practice. ICCV '99*. London, UK, UK: Springer-Verlag, 2000, pp. 267–277 (cit. on p. 43).
- [95] R. Irani, K. Nasrollahi, and T. B. Moeslund. “Improved Pulse Detection from Head Motions Using DCT”. English. In: Institute for Systems, Technologies of Information, Control, and Communication, Jan. 2014 (cit. on p. 41).
- [96] M. Iwasaki and Y. Noguchi. “Hiding true emotions: micro-expressions in eyes retrospectively concealed by mouth movements”. In: *Scientific Reports* 6 (Feb. 2016), p. 22049 (cit. on pp. 1, 41).

- [97] Xitong Jia, Xianye Ben, Hui Yuan, Kidiyo Kpalma, and Weixiao Meng. “Macro-to-micro transformation model for micro-expression recognition”. In: *International Journal of Computational Science and Engineering* (Mar. 2017) (cit. on pp. 26, 93).
- [98] S. E.i Kahou, X. Bouthillier, P. Lamblin, et al. “EmoNets: Multimodal deep learning approaches for emotion recognition in video”. en. In: *Journal on Multimodal User Interfaces* (Aug. 2015), pp. 1–13 (cit. on pp. 33, 34).
- [99] S. Kamate and N. Yilmazer. “Application of Object Detection and Tracking Techniques for Unmanned Aerial Vehicles”. In: *Procedia Computer Science*. Complex Adaptive Systems San Jose, CA November 2-4, 2015 61 (Jan. 2015), pp. 436–441 (cit. on p. 69).
- [100] D. H. Kim, W. J. Baddar, and Y. M. Ro. “Micro-Expression Recognition with Expression-State Constrained Spatio-Temporal Feature Representations”. In: *Proceedings of the 2016 ACM on Multimedia Conference*. MM ’16. New York, NY, USA: ACM, 2016, pp. 382–386 (cit. on pp. 33, 40, 93, 115, 116).
- [101] Y. Koda, Y. Yoshitomi, M. Nakano, and M. Tabuse. “A facial expression recognition for a speaker of a phoneme of vowel using thermal image processing and a speech recognition system”. In: *RO-MAN 2009 - The 18th IEEE International Symposium on Robot and Human Interactive Communication*. Sept. 2009, pp. 955–960 (cit. on pp. 19, 25, 31).
- [102] W. Koehrsen. *A Conceptual Explanation of Bayesian Hyperparameter Optimization for Machine Learning*. June 2018 (cit. on p. 166).
- [103] S. G. Kong, J. Heo, B. R. Abidi, J. Paik, and M. A. Abidi. “Recent advances in visual and infrared face recognition—a review”. In: *Computer Vision and Image Understanding* 97.1 (Jan. 2005), pp. 103–135 (cit. on p. 19).
- [104] M. Kopaczka, K. Acar, and D. Merhof. “Robust Facial Landmark Detection and Face Tracking in Thermal Infrared Images using Active Appearance Models.” In: *VISIGRAPP (4: VISAPP)*. 2016, pp. 150–158 (cit. on p. 153).
- [105] A. Krizhevsky, I. Sutskever, and G. E. Hinton. “ImageNet Classification with Deep Convolutional Neural Networks”. In: *Advances in Neural Information Processing Systems* 25. Ed. by F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger. Curran Associates, Inc., 2012, pp. 1097–1105 (cit. on p. 104).
- [106] S. Kumano, K. Otsuka, J. Yamato, E. Maeda, and Y. Sato. “Pose-Invariant Facial Expression Recognition Using Variable-Intensity Templates”. en. In: *International Journal of Computer Vision* 83.2 (Nov. 2008), pp. 178–194 (cit. on p. 23).
- [107] J. Lee and S. Y. Shin. “General construction of time-domain filters for orientation data”. In: *IEEE Transactions on Visualization and Computer Graphics* 8.2 (Apr. 2002), pp. 119–128 (cit. on p. 57).

- [108] Y.-J. Lee, Y.-. Yeh, and H.-K. Pao. “An Introduction to Support Vector Machines”. In: *National Taiwan University of Science and Technology* (2010) (cit. on pp. 161, 163, 164).
- [109] H. Li, Z. Lin, X. Shen, J. Brandt, and G. Hua. “A convolutional neural network cascade for face detection”. In: *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2015, pp. 5325–5334 (cit. on p. 18).
- [110] J. Li and M. Oussalah. “Automatic face emotion recognition system”. In: *2010 IEEE 9th International Conference on Cybernetic Intelligent Systems (CIS)*. Sept. 2010, pp. 1–6 (cit. on pp. 25, 31).
- [111] X. Li, X. Hong, A. Moilanen, et al. “Reading Hidden Emotions: Spontaneous Micro-expression Spotting and Recognition”. In: *arXiv:1511.00423 [cs]* (Nov. 2015). arXiv: 1511.00423 (cit. on p. 26).
- [112] X. Li, X. Hong, A. Moilanen, et al. “Towards Reading Hidden Emotions: A Comparative Study of Spontaneous Micro-expression Spotting and Recognition Methods”. In: *IEEE Transactions on Affective Computing PP.99* (2017), pp. 1–1 (cit. on pp. 22, 25, 26, 28, 30, 38, 63, 65, 73, 85, 90, 93, 116).
- [113] X. Li, T. Pfister, X. Huang, G. Zhao, and M. Pietikainen. “A Spontaneous Micro-expression Database: Inducement, collection and baseline”. In: *2013 10th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG)*. Apr. 2013, pp. 1–6 (cit. on pp. ix, xiii, 12, 83, 108, 120, 152).
- [114] J. J. Lien, T. Kanade, J. F. Cohn, and C. C. Li. “Detection, tracking, and classification of action units in facial expression”. In: *Robotics and Autonomous Systems* 31.3 (May 2000), pp. 131–146 (cit. on p. 32).
- [115] S. T. Liong, R. C. W. Phan, J. See, Y. H. Oh, and K. Wong. “Optical strain based recognition of subtle emotions”. In: *2014 International Symposium on Intelligent Signal Processing and Communication Systems (ISPACS)*. Dec. 2014, pp. 180–184 (cit. on pp. 24, 31, 39, 116).
- [116] S. T. Liong, J. See, R. C.-W. Phan, et al. “Subtle Expression Recognition Using Optical Strain Weighted Features”. en. In: *Computer Vision - ACCV 2014 Workshops*. Ed. by C. V. Jawahar and Shiguang Shan. Lecture Notes in Computer Science 9009. Springer International Publishing, Nov. 2014, pp. 644–657 (cit. on pp. 24, 27, 31, 39, 116).
- [117] S. T. Liong, J. See, K. Wong, et al. “Automatic apex frame spotting in micro-expression database”. In: *2015 3rd IAPR Asian Conference on Pattern Recognition (ACPR)*. Nov. 2015, pp. 665–669 (cit. on pp. 21, 28, 30, 90).
- [118] S.-T. Liong, J. See, K. Wong, and R. C. -W. Phan. “Less is more: Micro-expression recognition from video using apex frame”. In: *Signal Processing: Image Communication* 62 (Mar. 2018), pp. 82–92 (cit. on pp. 24, 39, 93).

- [119] S. Liong and K. Wong. “Micro-expression recognition using apex frame with phase information”. In: Dec. 2017, pp. 534–537 (cit. on pp. xiv, 27, 39, 93, 116).
- [120] C. Liu, A. Torralba, W. T. Freeman, F. Durand, and E. H. Adelson. “Motion Magnification”. In: *ACM SIGGRAPH 2005 Papers*. SIGGRAPH ’05. New York, NY, USA: ACM, 2005, pp. 519–526 (cit. on pp. 41, 49).
- [121] C. Liu, J. Yuen, and A. Torralba. “SIFT Flow: Dense Correspondence across Scenes and Its Applications”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 33.5 (May 2011), pp. 978–994 (cit. on p. 46).
- [122] P. Liu, S. Han, Z. Meng, and Y. Tong. “Facial Expression Recognition via a Boosted Deep Belief Network”. In: IEEE, June 2014, pp. 1805–1812 (cit. on p. 32).
- [123] Y.-J. Liu, J.-K. Zhang, W.-J. Yan, et al. “A Main Directional Mean Optical Flow Feature for Spontaneous Micro-Expression Recognition”. In: *IEEE Transactions on Affective Computing* 7.4 (Oct. 2016), pp. 299–310 (cit. on pp. 24, 39, 93, 115, 116).
- [124] Z. Liu and S. Wang. “Emotion Recognition Using Hidden Markov Models from Facial Temperature Sequence”. en. In: *Affective Computing and Intelligent Interaction*. Ed. by Sidney D’Mello, Arthur Graesser, Björn Schuller, and Jean-Claude Martin. Lecture Notes in Computer Science 6975. Springer Berlin Heidelberg, 2011, pp. 240–247 (cit. on pp. 32, 153).
- [125] H. Lu. “Video Analysis for Micro-Expression Spotting and Recognition”. Ph.D. INSA de Rennes, Apr. 2018 (cit. on pp. 23, 24, 29, 30, 39, 115, 116).
- [126] H. Lua, K. Kpalma, and J. Ronsin. “Micro-expression detection using integral projections”. In: *Journal of WSCG* 25 (Jan. 2017), pp. 87–96 (cit. on pp. 25, 28, 30, 85, 90).
- [127] B. D. Lucas and T. Kanade. “An Iterative Image Registration Technique with an Application to Stereo Vision”. In: *Proceedings of the 7th International Joint Conference on Artificial Intelligence - Volume 2*. IJCAI’81. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1981, pp. 674–679 (cit. on p. 45).
- [128] P. Lucey, J.F. Cohn, T. Kanade, et al. “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 (CVPRW)*. June 2010, pp. 94–101 (cit. on p. 103).
- [129] S. Majerus, H. Gill-Thwaites, K. Andrews, and S. Laureys. “Behavioral evaluation of consciousness in severe brain damage”. eng. In: *Progress in Brain Research* 150 (2005), pp. 397–413 (cit. on p. 145).
- [130] A. Marchewka, Ł. Żurawski, K. Jednoróg, and A. Grabowska. “The Nencki Affective Picture System (NAPS): Introduction to a novel, standardized, wide-range, high-quality, realistic picture database”. en. In: *Behavior Research Methods* 46.2 (June 2014), pp. 596–610 (cit. on p. 148).

- [131] G. G. Mateos. “Procesamiento de caras humanas mediante integrales proyectivas”. es. <http://purl.org/dc/dcmitype/Text>. Universidad de Murcia, 2007 (cit. on p. 25).
- [132] D. Matsumoto and H. S. Hwang. “Evidence for training the ability to read microexpressions of emotion”. en. In: *Motivation and Emotion* 35.2 (June 2011), pp. 181–191 (cit. on p. 1).
- [133] D. Matsumoto and H. S. Hwang. “Evidence for training the ability to read microexpressions of emotion”. en. In: *Motivation and Emotion* 35.2 (Apr. 2011), pp. 181–191 (cit. on p. 8).
- [134] I. Matthews and S. Baker. “Active Appearance Models Revisited”. In: *Int. J. Comput. Vision* 60.2 (Nov. 2004), pp. 135–164 (cit. on p. 20).
- [135] P. Maurel, A. McGonigal, R. Keriven, and Patrick Chauvel. “3D model fitting for facial expression analysis under uncontrolled imaging conditions”. In: *19th International Conference on Pattern Recognition, 2008. ICPR 2008*. Dec. 2008, pp. 1–4 (cit. on p. 20).
- [136] A. J. McLeod, J. S. H. Baxter, S. de Ribaupierre, and T. M. Peters. “Motion magnification for endoscopic surgery”. In: vol. 9036. 2014, pp. 90360C–90360C–8 (cit. on p. 49).
- [137] R. Meyer and C. House. *Preprocessing and Descriptor Features for Facial Micro-Expression Recognition*. Tech. rep. Stanford University, June 2015 (cit. on pp. 18, 26, 38, 93).
- [138] S. Meyer, O. Wang, H. Zimmer, M. Grosse, and A. Sorkine-Hornung. “Phase-based frame interpolation for video”. In: *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2015, pp. 1410–1418 (cit. on p. 9).
- [139] *Micro Expression Recognition Training Tools - Detect Lies - Humintell*. en-US (cit. on p. 1).
- [140] A. Moilanen, G. Zhao, and M. Pietikainen. “Spotting Rapid Facial Movements from Videos Using Appearance-Based Feature Difference Analysis”. In: *2014 22nd International Conference on Pattern Recognition (ICPR)*. Aug. 2014, pp. 1722–1727 (cit. on p. 28).
- [141] T. Nakano, M. Kato, Y. Morito, S. Itoi, and S. Kitazawa. “Blink-related momentary activation of the default mode network while viewing videos”. en. In: *Proceedings of the National Academy of Sciences* 110.2 (Jan. 2013), pp. 702–706 (cit. on p. 22).
- [142] A. C. L. Ngo, Y.-H. Oh, R. C. -W. Phan, and J. See. “Eulerian emotion magnification for subtle expression recognition”. In: *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Mar. 2016, pp. 1243–1247 (cit. on pp. 26, 38, 93, 115, 116).
- [143] H. Nguyen, F. Chen, K. Kotani, and B. Le. “Fusion of Visible Images and Thermal Image Sequences for Automated Facial Emotion Estimation”. In: *ResearchGate* 10.3&4 (Nov. 2014), pp. 294–308 (cit. on pp. 21, 26, 33, 34, 153).

- [144] H. Nguyen, F. Chen, K. Kotani, and B. Le. “Human Emotion Estimation Using Wavelet Transform and t-ROIs for Fusion of Visible Images and Thermal Image Sequences”. en. In: *Computational Science and Its Applications – ICCSA 2014*. Ed. by Beniamino Murgante, Sanjay Misra, Ana Maria A. C. Rocha, et al. Lecture Notes in Computer Science 8584. Springer International Publishing, June 2014, pp. 224–235 (cit. on pp. 25, 33, 34).
- [145] H. Nguyen, K. Kotani, F. Chen, and B. Le. “A Thermal Facial Emotion Database and Its Analysis”. en. In: *Image and Video Technology*. Ed. by Reinhard Klette, Mariano Rivera, and Shin’ichi Satoh. Lecture Notes in Computer Science 8333. Springer Berlin Heidelberg, Oct. 2013, pp. 397–408 (cit. on p. 25).
- [146] H. Nguyen, K. Kotani, F. Chen, and B. Le. “Estimation of human emotions using thermal facial information”. In: vol. 9069. 2014, 906900–906900–5 (cit. on p. 153).
- [147] S Nikolay. *Efficient LLBP (Line Local Binary Pattern)* (cit. on p. 69).
- [148] Y.-H. Oh, A. C. L. Ngo, J. See, et al. “Monogenic Riesz wavelet representation for micro-expression recognition”. In: *2015 IEEE International Conference on Digital Signal Processing (DSP)*. July 2015, pp. 1237–1241 (cit. on pp. xiv, 26, 38, 93, 115, 116).
- [149] T. Ojala, M. Pietikainen, and T. Maenpaa. “Multiresolution gray-scale and rotation invariant texture classification with local binary patterns”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 24.7 (July 2002), pp. 971–987 (cit. on p. 65).
- [150] OpenCV. *Introduction to Support Vector Machines — OpenCV 2.4.13.6 documentation* (cit. on p. 31).
- [151] A. M. Owen, M. R. Coleman, M. Boly, et al. “Detecting awareness in the vegetative state”. eng. In: *Science (New York, N.Y.)* 313.5792 (Sept. 2006), p. 1402 (cit. on p. 146).
- [152] S. Y. Park, S. Ho Lee, and Y. M. Ro. “Subtle Facial Expression Recognition Using Adaptive Magnification of Discriminative Facial Motion”. In: *Proceedings of the 23rd ACM International Conference on Multimedia*. MM ’15. New York, NY, USA: ACM, 2015, pp. 911–914 (cit. on pp. 26, 31, 73, 77, 93).
- [153] S. Park and D. Kim. “Subtle facial expression recognition using motion magnification”. In: *Pattern Recognition Letters* 30.7 (May 2009), pp. 708–716 (cit. on pp. 11, 27, 38, 39, 49, 73, 116).
- [154] D. Patel, G. Zhao, and M. Pietikäinen. “Spatiotemporal Integration of Optical Flow Vectors for Micro-expression Detection”. en. In: *Advanced Concepts for Intelligent Vision Systems*. Ed. by Sebastiano Battiato, Jacques Blanc-Talon, Giovanni Gallo, et al. Lecture Notes in Computer Science 9386. Springer International Publishing, 2015, pp. 369–380 (cit. on pp. 28, 30).
- [155] *Paul Ekman Group*. en-US (cit. on pp. 1, 7).

- [156] J. W. Peirce. “Generating stimuli for neuroscience using PsychoPy”. English. In: *Frontiers in Neuroinformatics* 2 (2009) (cit. on p. 148).
- [157] J. W. Peirce. “PsychoPy—Psychophysics software in Python”. In: *Journal of Neuroscience Methods* 162.1-2 (May 2007), pp. 8–13 (cit. on p. 148).
- [158] M. Peng, C. Wang, T. Chen, G. Liu, and X. Fu. “Dual Temporal Scale Convolutional Neural Network for Micro-Expression Recognition”. In: *Frontiers in Psychology* 8 (Oct. 2017) (cit. on pp. 33, 40, 93).
- [159] J. S. Pérez, E. Meinhardt-Llopis, and G. Facciolo. “TV-L1 Optical Flow Estimation”. en. In: *Image Processing On Line* 3 (July 2013), pp. 137–150 (cit. on pp. 45, 46, 104, 108, 109, 167).
- [160] Béatrice Pesquet-Popescu, Marco Cagnazzo, and Frédéric Dufaux. *Motion Estimation Techniques*. TELECOM ParisTech (cit. on p. 42).
- [161] T. Pfister, X. Li, G. Zhao, and M. Pietikainen. “Recognising spontaneous facial micro-expressions”. In: *2011 IEEE International Conference on Computer Vision (ICCV)*. Nov. 2011, pp. 1449–1456 (cit. on pp. 18, 26, 31, 39, 93).
- [162] M. Pietikäinen. “Local Binary Patterns”. en. In: *Scholarpedia* 5.3 (2010), p. 9775 (cit. on p. 25).
- [163] S. Polikovsky, Y. Kameda, and Y. Ohta. “Detection and measurement of facial micro-expression characteristics for psychological analysis”. In: *Kameda’s Publication* 110 (2010), pp. 57–64 (cit. on pp. 8, 11, 26, 31, 39).
- [164] S. Polikovsky, Y. Kameda, and Y. Ohta. “Facial micro-expressions recognition using high speed camera and 3D-gradient descriptor”. In: *3rd International Conference on Crime Detection and Prevention (ICDP 2009)*. Dec. 2009, pp. 1–6 (cit. on pp. 8, 11, 26, 31, 39).
- [165] S. Porter and L. ten Brinke. “The truth about lies: What works in detecting high-stakes deception?” en. In: *Legal and Criminological Psychology* 15.1 (Feb. 2010), pp. 57–75 (cit. on p. 8).
- [166] J. Portilla and E. P. Simoncelli. “A Parametric Texture Model Based on Joint Statistics of Complex Wavelet Coefficients”. In: *Int. J. Comput. Vision* 40.1 (Oct. 2000), pp. 49–70 (cit. on p. 48).
- [167] J. Posner, J. A. Russell, and B. S. Peterson. “The circumplex model of affect: An integrative approach to affective neuroscience, cognitive development, and psychopathology”. In: *Development and psychopathology* 17.3 (2005), pp. 715–734 (cit. on p. 7).
- [168] F. Qu, S. J. Wang, W. J. Yan, et al. “CAS(ME)2: A Database for Spontaneous Macro-expression and Micro-expression Spotting and Recognition”. In: *IEEE Transactions on Affective Computing* (2017), pp. 1–1 (cit. on pp. xv, 8, 13, 120, 153).

- [169] R. Raghavendra, M. Avinash, S. Marcel, and C. Busch. “Finger vein liveness detection using motion magnification”. In: *2015 IEEE 7th International Conference on Biometrics Theory, Applications and Systems (BTAS)*. Sept. 2015, pp. 1–7 (cit. on p. 49).
- [170] T. H. Rassem and B. E. Khoo. “Completed local ternary pattern for rotation invariant texture classification”. In: *The Scientific World Journal 2014* (2014) (cit. on p. 69).
- [171] M. Riegel, Ł. Żurawski, M. Wierzba, et al. “Characterization of the Nencki Affective Picture System by discrete emotional categories (NAPS BE)”. en. In: *Behavior Research Methods* 48.2 (June 2016), pp. 600–612 (cit. on p. 148).
- [172] O. Russakovsky, J. Deng, H. Su, et al. “ImageNet Large Scale Visual Recognition Challenge”. In: *International Journal of Computer Vision (IJCV)* 115.3 (2015), pp. 211–252 (cit. on pp. 104, 106).
- [173] T. A. Russell, E. Chu, and M. L. Phillips. “A pilot study to investigate the effectiveness of emotion recognition remediation in schizophrenia using the micro-expression training tool”. eng. In: *The British Journal of Clinical Psychology* 45.Pt 4 (Nov. 2006), pp. 579–583 (cit. on pp. 1, 41).
- [174] A. Samal and P. A. Iyengar. “Automatic recognition and analysis of human faces and facial expressions: a survey”. In: *Pattern Recognition* 25.1 (Jan. 1992), pp. 65–77 (cit. on pp. vii, 2).
- [175] Paul Sastrasinh. *Dense Realtime Optical Flow on the GPU*. Final Project. Brown University, Apr. 2011 (cit. on p. 46).
- [176] S. Sayad. *KNN Classification* (cit. on p. 31).
- [177] C. Schnakers, J. T. Giacino, M. Løvstad, et al. “Preserved covert cognition in noncommunicative patients with severe brain injury?” eng. In: *Neurorehabilitation and Neural Repair* 29.4 (May 2015), pp. 308–317 (cit. on p. 146).
- [178] C. Schnakers, A. Vanhaudenhuyse, J. Giacino, et al. “Diagnostic accuracy of the vegetative and minimally conscious state: Clinical consensus versus standardized neurobehavioral assessment”. In: *BMC Neurology* 9 (July 2009), p. 35 (cit. on p. 145).
- [179] T. Senechal, V. Rapp, and L. Prevost. “Facial Feature Tracking for Emotional Dynamic Analysis”. en. In: *Advanced Concepts for Intelligent Vision Systems*. Ed. by Jacques Blanc-Talon, Richard Kleihorst, Wilfried Philips, Dan Popescu, and Paul Scheunders. Lecture Notes in Computer Science 6915. Springer Berlin Heidelberg, Aug. 2011, pp. 495–506 (cit. on p. 20).
- [180] C. Shan, S. Gong, and P. W. McOwan. “Facial expression recognition based on Local Binary Patterns: A comprehensive study”. In: *Image and Vision Computing* 27.6 (May 2009), pp. 803–816 (cit. on p. 25).

- [181] A. Shiel, S. A. Horn, B. A. Wilson, et al. “The Wessex Head Injury Matrix (WHIM) main scale: a preliminary report on a scale to assess and monitor patient recovery after severe head injury”. eng. In: *Clinical Rehabilitation* 14.4 (Aug. 2000), pp. 408–416 (cit. on p. 146).
- [182] M. Shreve. “Automatic Macro- and Micro-Facial Expression Spotting and Applications”. Graduate Theses and Dissertations. University of South Florida, Jan. 2013 (cit. on pp. 28, 30).
- [183] M. A. Shreve. “Automatic macro-and micro-facial expression spotting and applications”. PhD thesis. 2013 (cit. on p. 24).
- [184] M. Shreve, S. Godavarthy, D. Goldgof, and S. Sarkar. “Macro- and micro-expression spotting in long videos using spatio-temporal strain”. In: *2011 IEEE International Conference on Automatic Face Gesture Recognition and Workshops (FG 2011)*. Mar. 2011, pp. 51–56 (cit. on pp. 11, 18, 28).
- [185] M. Shreve, S. Godavarthy, V. Manohar, D. Goldgof, and S. Sarkar. “Towards macro- and micro-expression spotting in video using strain patterns”. In: *2009 Workshop on Applications of Computer Vision (WACV)*. Dec. 2009, pp. 1–6 (cit. on pp. 11, 28).
- [186] M. Shreve, N. Jain, D. Goldgof, et al. “Evaluation of Facial Reconstructive Surgery on Patients with Facial Palsy Using Optical Strain”. en. In: *Computer Analysis of Images and Patterns*. Ed. by Pedro Real, Daniel Diaz-Pernil, Helena Molina-Abril, Ainhoa Berciano, and Walter Kropatsch. Lecture Notes in Computer Science 6854. Springer Berlin Heidelberg, 2011, pp. 512–519 (cit. on p. 24).
- [187] T. Simon, H. Joo, I. Matthews, and Y. Sheikh. “Hand Keypoint Detection in Single Images using Multiview Bootstrapping”. In: *CVPR. 2017* (cit. on pp. 19, 20).
- [188] E.P. Simoncelli and W.T. Freeman. “The steerable pyramid: a flexible architecture for multi-scale derivative computation”. In: , *International Conference on Image Processing, 1995. Proceedings*. Vol. 3. Oct. 1995, 444–447 vol.3 (cit. on p. 48).
- [189] E.P. Simoncelli, W.T. Freeman, E.H. Adelson, and D.J. Heeger. “Shiftable multi-scale transforms”. In: *IEEE Transactions on Information Theory* 38.2 (Mar. 1992), pp. 587–607 (cit. on p. 48).
- [190] Y. Song, L.-P. Morency, and R. Davis. “Learning a Sparse Codebook of Facial and Body Microexpressions for Emotion Recognition”. In: *Proceedings of the 15th ACM on International Conference on Multimodal Interaction. ICMI '13*. New York, NY, USA: ACM, 2013, pp. 237–244 (cit. on pp. 27, 31).
- [191] J. Steel, M. Youssef, R. Pfeifer, et al. “Health-related quality of life in patients with multiple injuries and traumatic brain injury 10+ years postinjury”. eng. In: *The Journal of Trauma* 69.3 (Sept. 2010), 523–530, discussion 530–531 (cit. on p. 145).
- [192] R. A. Stevenson and T. W. James. “Affective auditory stimuli: Characterization of the International Affective Digitized Sounds (IADS) by discrete emotional categories”. en. In: *Behavior Research Methods* 40.1 (Feb. 2008), pp. 315–321 (cit. on p. 148).

- [193] C. P. Sumathi, T. Santhanam, and M. Mahadevi. “Automatic facial expression analysis a survey”. In: *International Journal of Computer Science and Engineering Survey* 3.6 (2012), p. 47 (cit. on p. 17).
- [194] D. Sun, S. Roth, and M.J. Black. “Secrets of optical flow estimation and their principles”. In: *2010 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2010, pp. 2432–2439 (cit. on p. 45).
- [195] Hai Tao and Thomas S. Huang. “A PIECEWISE BÉZIER VOLUME DEFORMATION MODEL AND ITS APPLICATIONS IN FACIAL MOTION CAPTURE”. en. In: *Advances in Image Processing and Understanding*. Vol. 52. WORLD SCIENTIFIC, Nov. 2002, pp. 39–56 (cit. on p. 20).
- [196] Techopedia. *What is an Artificial Neural Network (ANN)? - Definition from Techopedia*. en (cit. on p. 31).
- [197] S. Thakur, S. Paul, A. Mondal, S. Das, and A. Abraham. “Face detection using skin tone segmentation”. In: *2011 World Congress on Information and Communication Technologies (WICT)*. Dec. 2011, pp. 53–60 (cit. on p. 19).
- [198] Ying-Li Tian, Takeo Kanade, and Jeffrey F. Cohn. “Facial Expression Analysis”. en. In: *Handbook of Face Recognition*. Springer New York, 2005, pp. 247–275 (cit. on pp. vii, 2).
- [199] C. Tomasi and T. Kanade. *Detection and Tracking of Point Features*. Tech. rep. International Journal of Computer Vision, 1991 (cit. on p. 75).
- [200] Y. Tong, Y. Wang, Z. Zhu, and Q. Ji. “Robust Facial Feature Tracking Under Varying Face Pose and Facial Expression”. In: *Pattern Recogn.* 40.11 (Nov. 2007), pp. 3195–3208 (cit. on p. 20).
- [201] A. Torralba. *Lecture 4 Motion Filters Spatial Pyramids*. 2016 (cit. on p. 50).
- [202] T.-K. Tran, X. Hong, and G. Zhao. “Sliding Window Based Micro-expression Spotting: A Benchmark”. en. In: *Advanced Concepts for Intelligent Vision Systems*. Lecture Notes in Computer Science. Springer, Cham, Sept. 2017, pp. 542–553 (cit. on pp. 29, 30, 90).
- [203] L. Trujillo, G. Olague, R. Hammoud, and B. Hernandez. “Automatic Feature Localization in Thermal Images for Facial Expression Recognition”. In: *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Workshops*. June 2005, pp. 14–14 (cit. on pp. 19, 20, 25).
- [204] G. Tzimiropoulos and M. Pantic. “Optimization Problems for Fast AAM Fitting in-the-Wild”. In: *2013 IEEE International Conference on Computer Vision (ICCV)*. Dec. 2013, pp. 593–600 (cit. on pp. xiii, 20, 21, 75, 150).
- [205] R. Verma. *K-Means Algorithm* (cit. on p. 31).

- [206] P. Viola and M. Jones. “Rapid object detection using a boosted cascade of simple features”. In: *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2001. CVPR 2001*. Vol. 1. 2001, I–511–I–518 vol.1 (cit. on pp. xiii, 18, 74).
- [207] N. Wadhwa, M. Rubinstein, F. Durand, and W. T. Freeman. “Phase-based Video Motion Processing”. In: *ACM Trans. Graph.* 32.4 (July 2013), 80:1–80:10 (cit. on pp. xi, 41, 50–52, 54, 56, 61, 63, 64, 73).
- [208] N. Wadhwa, M. Rubinstein, F. Durand, and W. T. Freeman. *Quaternionic representation of the riesz pyramid for video magnification*. Tech. rep. 2014 (cit. on pp. 57, 78, 150, 156).
- [209] N. Wadhwa, M. Rubinstein, F. Durand, and W.T. Freeman. “Riesz pyramids for fast phase-based video magnification”. In: *2014 IEEE International Conference on Computational Photography (ICCP)*. May 2014, pp. 1–10 (cit. on pp. xi, 41, 51, 55, 58–60, 63, 64, 150).
- [210] C. W. Wang, A. Ahmed, and A. Hunter. “Vision analysis in detecting abnormal breathing activity in application to diagnosis of obstructive sleep apnoea”. eng. In: *Conference proceedings: ... Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Annual Conference 1* (2006), pp. 4469–4473 (cit. on pp. x, 41).
- [211] J. Wang, S. M. Drucker, M. Agrawala, and M. F. Cohen. “The Cartoon Animation Filter”. In: *ACM SIGGRAPH 2006 Papers*. SIGGRAPH '06. New York, NY, USA: ACM, 2006, pp. 1169–1173 (cit. on p. 49).
- [212] N. Wang, X. Gao, D. Tao, and X. Li. “Facial Feature Point Detection: A Comprehensive Survey”. In: *arXiv:1410.1037 [cs]* (Oct. 2014). arXiv: 1410.1037 (cit. on p. 20).
- [213] S. J. Wang, W. J. Yan, G. Zhao, X. Fu, and C. G. Zhou. “Micro-Expression Recognition Using Robust Principal Component Analysis and Local Spatiotemporal Directional Features”. en. In: *Computer Vision - ECCV 2014 Workshops*. Ed. by Lourdes Agapito, Michael M. Bronstein, and Carsten Rother. Lecture Notes in Computer Science 8925. Springer International Publishing, Sept. 2014, pp. 325–338 (cit. on pp. 27, 39, 116).
- [214] S.-J. Wang, W.-J. Yan, X. Li, et al. “Micro-Expression Recognition Using Color Spaces”. In: *IEEE Transactions on Image Processing* 24.12 (Dec. 2015), pp. 6034–6047 (cit. on pp. 26, 31, 38).
- [215] S. Wang, H. Chen, W. Yan, Y. Chen, and X. Fu. “Face Recognition and Micro-expression Recognition Based on Discriminant Tensor Subspace Analysis Plus Extreme Learning Machine”. en. In: *Neural Processing Letters* 39.1 (Feb. 2013), pp. 25–43 (cit. on pp. 25, 31).

- [216] Y. Wang, J. See, R. C.-W. Phan, and Y.-H. Oh. “Efficient Spatio-Temporal Local Binary Patterns for Spontaneous Facial Micro-Expression Recognition”. In: *PLoS ONE* 10.5 (May 2015) (cit. on pp. 26, 27, 38, 93, 115, 116).
- [217] Yandan Wang, John See, Raphael C.-W. Phan, and Yee-Hui Oh. “LBP with Six Intersection Points: Reducing Redundant Information in LBP-TOP for Micro-expression Recognition”. en. In: *Computer Vision – ACCV 2014*. Ed. by D. Cremers, I. Reid, H. Saito, and M.-H. Yang. Lecture Notes in Computer Science 9003. Springer International Publishing, Nov. 2014, pp. 525–537 (cit. on pp. 26, 31, 38, 93, 116).
- [218] P. Werner, A. Al-Hamadi, R. Niese, et al. “Automatic Pain Recognition from Video and Biomedical Signals”. In: *2014 22nd International Conference on Pattern Recognition (ICPR)*. Aug. 2014, pp. 4582–4587 (cit. on p. 33).
- [219] L. Wietzke, G. Sommer, and O. Fleischmann. “The geometry of 2D image signals”. In: *2009 IEEE Conference on Computer Vision and Pattern Recognition*. June 2009, pp. 1690–1697 (cit. on pp. 95, 121).
- [220] W. K. Wong, J. H. Hui, J. B. M. Desa, et al. “Face detection in thermal imaging using head curve geometry”. In: *2012 5th International Congress on Image and Signal Processing (CISP)*. Oct. 2012, pp. 881–884 (cit. on p. 19).
- [221] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma. “Robust Face Recognition via Sparse Representation”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 31.2 (Feb. 2009), pp. 210–227 (cit. on p. 31).
- [222] H.-Y. Wu, M. Rubinstein, E. Shih, et al. “Eulerian Video Magnification for Revealing Subtle Changes in the World”. In: *ACM Trans. Graph.* 31.4 (July 2012), 65:1–65:8 (cit. on pp. xi, 41, 49, 50, 52, 56, 73).
- [223] Q. Wu, X. Shen, and X. Fu. “The Machine Knows What You Are Hiding: An Automatic Micro-expression Recognition System”. en. In: *Affective Computing and Intelligent Interaction*. Ed. by Sidney D’Mello, Arthur Graesser, Björn Schuller, and Jean-Claude Martin. Lecture Notes in Computer Science 6975. Springer Berlin Heidelberg, Oct. 2011, pp. 152–162 (cit. on pp. 25, 32, 38).
- [224] Y. Wu. “Optical flow and motion analysis”. In: *Advanced Computer Vision Notes Series* 6 (2001), pp. 3–7 (cit. on p. 43).
- [225] Y. Wu, Z. Wang, and Q. Ji. “Facial Feature Tracking Under Varying Facial Expressions and Face Poses Based on Restricted Boltzmann Machines”. In: *2013 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2013, pp. 3452–3459 (cit. on p. 20).
- [226] Z. Xia, . Feng, J. Peng, X. Peng, and G. Zhao. “Spontaneous micro-expression spotting via geometric deformation modeling”. In: *Computer Vision and Image Understanding. Spontaneous Facial Behaviour Analysis* 147 (June 2016), pp. 87–94 (cit. on pp. 29, 30).

- [227] L. Xie, X. Liu, Z. Wang, et al. “Micro-expression Cognition and Emotion Modeling Based on Gross Reappraisal Strategy”. cn. In: *JOURNAL OF INFORMATION & COMPUTATIONAL SCIENCE* 12.6 (2015), pp. 2117–2132 (cit. on pp. 25, 31).
- [228] X. Xiong and F. De la Torre. “Supervised Descent Method and Its Applications to Face Alignment”. In: *2013 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2013, pp. 532–539 (cit. on p. 20).
- [229] F. Xu, J. Zhang, and J. Z. Wang. “Microexpression Identification and Categorization using a Facial Dynamics Map”. In: *IEEE Transactions on Affective Computing* PP.99 (2016), pp. 1–1 (cit. on pp. 24, 39, 93, 115, 116).
- [230] W. J. Yan, X. Li, S. J. Wang, et al. “CASME II: An Improved Spontaneous Micro-Expression Database and the Baseline Evaluation”. In: *PLoS ONE* 9.1 (Jan. 2014) (cit. on pp. ix, xiii, 13, 56, 83, 108, 120, 149, 152).
- [231] W. J. Yan, S. J. Wang, Y. H. Chen, G. Zhao, and X. Fu. “Quantifying Micro-expressions with Constraint Local Model and Local Binary Pattern”. en. In: *Computer Vision - ECCV 2014 Workshops*. Ed. by Lourdes Agapito, Michael M. Bronstein, and Carsten Rother. Lecture Notes in Computer Science 8925. Springer International Publishing, Sept. 2014, pp. 296–305 (cit. on pp. 21, 25, 28, 30).
- [232] W. J. Yan, Q. Wu, J. Liang, Y. H. Chen, and X. Fu. “How Fast are the Leaked Facial Expressions: The Duration of Micro-Expressions”. en. In: *Journal of Nonverbal Behavior* 37.4 (July 2013), pp. 217–230 (cit. on pp. viii, 9, 77, 84, 96).
- [233] W. J. Yan, Q. Wu, Y. J. Liu, S. J. Wang, and X. Fu. “CASME database: A dataset of spontaneous micro-expressions collected from neutralized faces”. In: *2013 10th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG)*. Apr. 2013, pp. 1–7 (cit. on pp. viii, 5, 12, 149).
- [234] Y. Yoshitomi. “Facial expression recognition for speaker using thermal image processing and speech recognition system”. In: *ResearchGate* (Jan. 2010) (cit. on pp. 25, 34).
- [235] Y. Yoshitomi, N. Miyawaki, S. Tomita, and S. Kimura. “Facial expression recognition using thermal image processing and neural network”. In: , *6th IEEE International Workshop on Robot and Human Communication, 1997. RO-MAN '97. Proceedings*. Sept. 1997, pp. 380–385 (cit. on p. 31).
- [236] C. Zach, T. Pock, and H. Bischof. “A Duality Based Approach for Realtime TV-L1 Optical Flow”. en. In: *Pattern Recognition*. Ed. by Fred A. Hamprecht, Christoph Schnörr, and Bernd Jähne. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2007, pp. 214–223 (cit. on pp. 46, 69).
- [237] M. D. Zeiler, G. W. Taylor, L. Sigal, I. Matthews, and R. Fergus. “Facial Expression Transfer with Input-Output Temporal Restricted Boltzmann Machines”. In: *Advances in Neural Information Processing Systems* 24. Ed. by J. Shawe-Taylor,

- R. S. Zemel, P. L. Bartlett, F. Pereira, and K. Q. Weinberger. Curran Associates, Inc., 2011, pp. 1629–1637 (cit. on p. 32).
- [238] Z. Zeng, M. Pantic, G.I. Roisman, and T.S. Huang. “A Survey of Affect Recognition Methods: Audio, Visual, and Spontaneous Expressions”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 31.1 (Jan. 2009), pp. 39–58 (cit. on pp. vii, 2).
- [239] G. Zhao and M. Pietikainen. “Dynamic Texture Recognition Using Local Binary Patterns with an Application to Facial Expressions”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 29.6 (June 2007), pp. 915–928 (cit. on pp. 26, 93).
- [240] Hao Zheng. “Micro-Expression Recognition based on 2D Gabor Filter and Sparse Representation”. In: *Journal of Physics: Conference Series* 787 (Jan. 2017), p. 012013 (cit. on pp. 25, 31, 39, 116).
- [241] Z. Zhou, G. Zhao, and M. Pietikainen. “Towards a practical lipreading system”. In: *2011 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2011, pp. 137–144 (cit. on pp. 9, 96).
- [242] X. Zhu and D. Ramanan. “Face detection, pose estimation, and landmark localization in the wild”. In: *2012 IEEE Conference on Computer Vision and Pattern Recognition*. June 2012, pp. 2879–2886 (cit. on pp. 20, 150).

Project DAME

In this section, we present the collaborative effort of Laboratoire Hubert Curien and the "Centre Hospitaliere Universitaire" (CHU) de Saint-Etienne for the DAME project. We collaborated with Dr. Pascal Giraux, professor and hospital doctor at CHU Saint Etienne, and Dr. Alexandre Bertholon, whose thesis was the basis of the presented work [17]. This section goes as follows: Sec. A.1 gives an introduction to the problem and enumerates the objectives of the DAME system. Sec. A.2 explain the micro-expression extraction procedure, including the elicitation protocol, list of materials, etc. Sec. A.3 explains how the videos were processed using the Riesz Pyramid method proposed in this thesis and introduces to a micro-expression visualization tool designed specifically for this project. Sec. A.4 explain the process of manually detecting and analysing micro-expressions. Sec. A.5 shows the preliminary obtained results. In Sec. A.6 we discuss the results and future perspectives for the project.

A.1 Introduction

Assessing the level of consciousness in severe acquired brain injury is a major issue in the Neurological Intensive Care Unit or in Neuro-rehabilitation units [73, 129, 178, 191]. However, characterization of the consciousness state is a challenging task for clinicians. Consciousness is a complex concept, and its different states are usually divided in different levels of arousal (i.e., wakefulness or vigilance) and awareness (of the environment and of the self) [17]. Different disorders of consciousness (DoC), following a severe brain injury, can be described and classified using this criteria [49]:

- **Coma:** Patients in coma cannot be awakened even when intensively stimulated and, hence, are not aware of the environment and of themselves.
- **Vegetative State (VS):** or unresponsive wakefulness syndrome (UWS), is characterized by the return of arousal without signs of awareness. In this case, patients regain sleep–wake cycles. However, their motor, auditory, and visual functions are restricted to mere reflexes and show no adapted emotional responses.
- **Minimally Concious State (MCS):** In this state, there are noncommunicating patients that show inconsistent, but discernible signs of behavioural activity that is

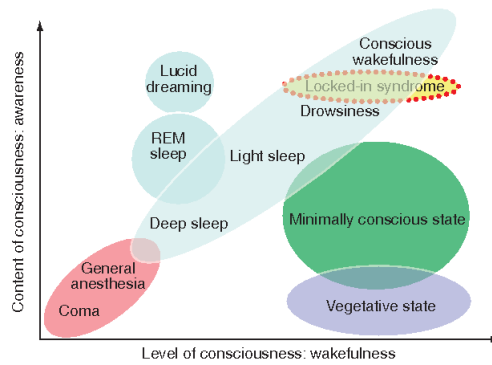


Figure A.1.: Simplified illustration of the two major components of consciousness and the way they correlate within the different DoC [49]

more than reflexive. These signs can be yes/no verbalized or gestural responses to orders, sketchy but intelligible verbal expressions, motor skills or adapted affective manifestations [78].

Functional magnetic resonance imaging (fMRI) [151], electroencephalography (EEM) [177] and electromyography (EMG) [81] offer promising opportunities to study consciousness and could bring complementary information for a better assessment of DoC. However these techniques are not often readily available, especially for repeated assessment like in DoC evaluation [17]. Facial movements and expressions have been used in the usual clinical scales, in the context of coma awakening and the assessment of consciousness state [181]. However, not much has been done in the area of detecting and recovering these expressions during coma awakening.

Emotional perceptions, and their facial transcriptions as facial expressions, could be recovered at an early stage of the coma emergence. In the context of severe global motor deficiency, micro-expressions could potentially be elicited even from patients presenting a lack of visible facial movements. This hypothesis is supported by the demonstration of higher facial EMG responses in volitional movements of the face in VS/UWS and MCS patients [81].

Thus, the purpose of this study was to develop and validate a method of eliciting, capturing and analysing micro-expressions in patients with DoC. In this project we integrate the knowledge and experience of experts in the physical medicine and rehabilitation with experts in computer vision. This project was comprised of:

- A device suitable for bedside use in acute care unit and rehabilitation ward, which records high-speed video recording, coupled to a computerized stimulus delivery station.

- A multimodal stimulation protocol designed to evoke emotional responses, using calibrated visual stimuli, auditory stimuli and tactile stimuli.
- A software visualization system which helped to detect and process micro-expressions from the captured videos.

Our main role in this project was the implementation of the micro-expressions analysis framework, the visualization tool and the stimuli delivery software. Furthermore, we also collaborated in the design of the image acquisition system.

A.2 Micro Expression Elicitation Experiment

This feasibility study was conducted with normal healthy volunteers, as the first step to explore and evaluate ME detection in an acute care unit environment. In a second step, the same system will be used with patients suffering acute and chronic DoC in the rehabilitation unit.

The subjects were exposed to a multimodal stimulation paradigm while they were filmed. It consisted of one control run and 3 runs of stimulation, lasting about 5 minutes each, delivered in a constant order: a control run with no stimulation (NS), a visual stimulation run (PI), an auditory stimulation run (SO), a tactile stimulation run (TA).

A.2.1 Participants

Fifteen healthy adult volunteers participated in this feasibility study, in an acute care unit environment. No participants had active neurological disorder or treatment. One participant had a history of neonatal meningitis without actual consequences. Data from one participant had to be discarded due to improper handling of the device. The final dataset includes 14 volunteers (6 males, 8 females, age : $30.1 \pm 11, 3$ SD). Due to the time-consuming analysis process, this feasibility study rely on the first seven patients, in the purpose of a primary and prompt validation of the stimulation protocol and ME detection.

A.2.2 List of Materials

For the experiment, a work station was built to be used for image acquisition and stimulus delivery station (See Fig. A.2). It was designed to be operated at the bedside of a patient in a hospital's intensive care unit. The work station is comprised of a movable metallic structure which carries the acquisition and stimuli delivery equipment, and a CPU work station that controls the whole process.

For the image acquisition step, two cameras were used to record video in a parallel mode: a gray scale high-speed optical camera (1936 × 1216 resolution, 100 fps, model GO-2400M-USB, JAI Ltd, Japan) and a thermal camera (320 × 240, 60 fps, FLIR A315, FLIR System, USA). The recorded scene was illuminated by 2 LED lamps positioned at both sides of the bed to avoid flickering light which might be captured by the high-speed camera. A C++ interface program was developed to control the recordings of both cameras. The videos were recorded frame by frame without image compression. As recordings were performed at the bedside of a intensive care unit, the video capture conditions cannot be fully controlled and there were some fluctuations regarding the viewing angle, head-pose and lighting.

For the stimulation elicitation step, the complete stimulation protocol was programmed using python and Psychopy software [156, 157]. For the visual stimulation step, a computer screen was located in front of the patient. The images used for this step were taken from the NAPS database (Nencki Affective Picture System) [130]. Each picture from NAPS had been described by Riegel [171] characterized into:

- Emotion categories among the 6 emotions: happiness, sadness, anger, disgust, surprise, fear and neutral with a label from 1 to 7.
- Valence with a score from 1 to 9.
- Arousal with a score from 1 to 9.

For the auditory stimulation step, a set of headphones connected to the work station were provided to the subjects. The sounds used for this step were taken from the IADS database (International Affective Digital Sounds) [192].

A.2.3 Stimulation paradigm design

The multimodal stimulation paradigm consisted in a control run and 3 runs of stimulation, lasting about 5 minutes each, delivered in a constant order: a control run with no stimulation (NS), a visual stimulation run (PI), an auditory stimulation run (SO), a tactile stimulation run (TA). Every stimulation run was comprised of 6 blocks of stimuli separated by 15 seconds of pause to respect an emotional wash out. The stimuli of a given block had the same emotional valence, and the blocks of each run were randomly displayed.

In the visual run (PI), for each block, 6 pictures that represented the same emotion were shown to the subject. Stimuli were adapted to each subject gender. Valence was chosen to be the most positive for the happiness block (mean rate of selected pictures: 7/9), the most negative for disgust and sadness block (mean rate: 2/9) and neutral for anger and surprise block (mean rate: 5/9). Finally, high arousal pictures were picked up to maximizing the



Figure A.2.: DAME image acquisition and stimuli delivery station

evoked emotions. Since head injured patients are considered as vulnerable, some pictures were rejected due to their traumatic potential. Each picture was displayed for 5 seconds.

For the auditory run (SO), Stimuli were selected in the same way as the Visual Stimuli run, according to their valence (positive or negative) and then their arousal. Stimuli were also adapted to each subject gender. The run was composed 3 positive blocks and 3 negative blocks of 5 sounds each. All stimuli with sexual contents were removed. Each sound had a duration of 5 seconds.

The tactile block (TA) was composed of 6 groups divided in two body sections to randomly stimulate: right or left upper or lower body part, for 15 seconds each. A soft brush was used to operate tactile stimulation.

To elicit ME, we set up a similar procedure as CASMEII database [230, 233]: The subject were asked to keep a neutral face and try to suppress their facial movements when they had one¹. In order to measure the ability of the stimuli to evoke emotional perceptions, participants were asked to:

1. Label the emotion felt during the block (anger, sadness, happiness, etc.).
2. Label the valence felt during the block (positive or negative).

¹The main difference of our procedure compared to the ones in [230, 233] is that no monetary compensations were offered to the participants

3. Score from 0 to 10 the average and maximal arousal they felt for each run.

A.3 Video Processing

In order to detect and crop the faces in the video we used the method proposed by [242] (The Viola and Jones detector used in Sec. 4.1 was not able to perform well due to the angle and head pose of certain captured faces). Then, We used the active appearance model proposed by [204] to locate a series of fiducial points. Then, we created a series of ROIs as described in Sec. 4.1.4. Next, we process and filter the cropped videos using the Riesz pyramid as described in Sec. 4.2. From the filtered quaternionic phase difference obtained we:

- Generate a series of temporal signals based on the face ROIs as described in Sec. 4.3.
- Take the masked oriented quaternionic phase difference and display them as a fake-color representation video, where the color saturation represents the phase ϕ and the hue the dominant orientation θ .
- Magnify the quaternionic phase difference and reconstruct the face videos using the Riesz based technique of [209, 208].

We developed a visualization interface program in Matlab to analyse and compare the processed data. It allowed us to display different processed data at the same screen. Let's take Fig. A.3 as an example. It shows the moment when a ME is taken place in the left corner of the mouth². The first row contains the original cropped video, the amplified version of the video, the oriented phase video. The second row contains a control panel, a plot containing the curves corresponding to the subtle motions in the eyes areas, and a second plot containing the curves corresponding to the subtle motions in the eyebrows and mouth areas. The first row show the images corresponding to the current frame (t) of the 3 videos. The plots in the second row show a section of the whole processed curve that goes from $[t - N, t + N]$, being N a given number of frames and the current frame (highlighted by a vertical dashed line) correspond to the central point in the curve. The control panel has different functionalities. It allow us to:

- Select and upload the set of videos and curves that are going to be displayed.
- Select the current time in the videos and plot. The user has the choice to update the current time by typing the desired value or clicking on any of the plots.

²A black rectangle has been added to the face image for identity privacy reasons

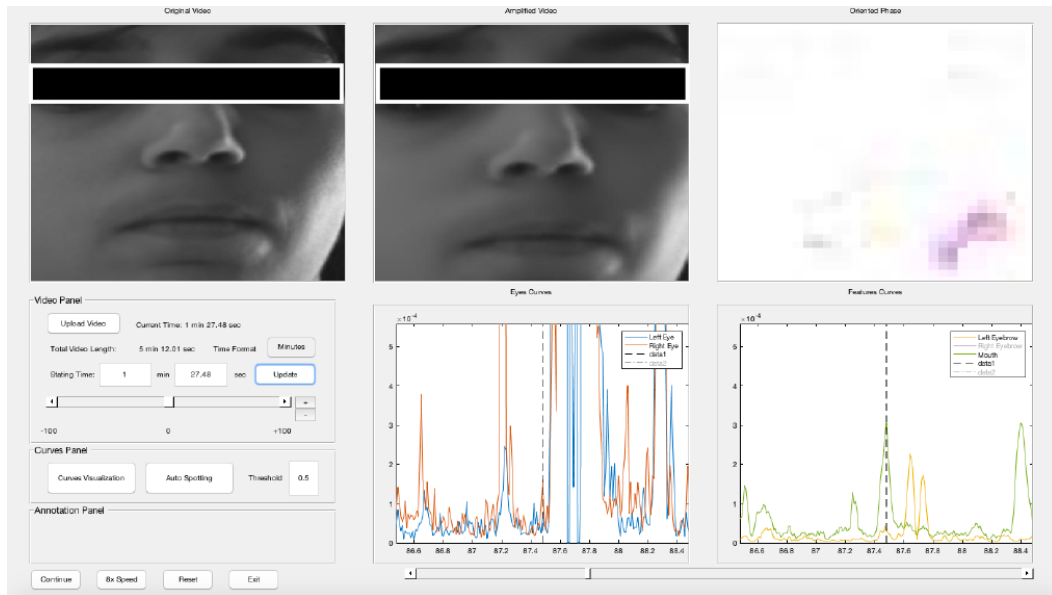


Figure A.3.: DAME visualization tool

- Select the length of the time window of the displayed processed curves (the value of N).
- Go forward or backwards in the video either by using the given scroll bars or by playing the video frame by frame.

A.4 Micro-Expression Analysis

The videos were analysed in two steps. Firstly, the experimenters observed the recorded unprocessed videos. Then they must report any detected ME with their timing, their AU and emotional valence scoring. Also they must report macro-expressions and non-emotional movements were written down. They could repeat, pause, and replay sections of the whole video as many times as required. Secondly, the experimenter used the visualization tool, where the phase signal graphs were used to detect temporal events and the normal, amplified and oriented phase videos were used to confirm if it was a ME or dismiss it as non-emotional event.

Both analysis were performed in blind condition for stimulus parameters by two experimenters. To be characterized as a ME, a candidate facial movement should have:

- A low spatial amplitude.
- Well defined temporal phases (onset, apex, offset).

- A duration between 100ms and 500ms or between 500ms and 1 second with a fast-onset duration fewer than 260ms

For the purpose of the study, all head and eye movements have been considered as non-emotional without further consideration. The observers did not take any ME recognition training prior to the analysis.

A.5 Preliminary Results

24 recordings from the 7 subjects were analysed from which 348 facial movements were noticed, and 192 ME were detected. Due to some technical difficulties regarding recording conditions (subject head movements) and the whole system sensibility, only sixteen processed videos were analyzed from which 319 facial movements were noticed and 156 ME were detected. On average, about 9 ME and 3.6 macro-expressions were found on a block of stimuli. The average ME duration was 251.32 ± 87.24 ms. All the results of this study can be found in [17]. The main results show that the overall DAME system, including the video recording and stimulation devices and multimodal stimulation protocol was able to elicit a significant number of ME for each tested healthy subject, as detected by our analysis methods. These preliminary results validate the first step before testing of the DAME system with real patients with DoC.

A.6 Discussion and Future Perspectives

Although, the initial results are promising, there is a number improvements that need to be done that require our attention for future iterations of the DAME project. In this section, we will discuss the difficulties and possible improvements regarding the acquisition system and video processing step. For a detailed discussion regarding the ME elicitation protocol or ME analysis we refer the reader to [17].

The micro-expression analysis method used for this project was initially tested in datasets which were captured in controlled environments (such as CASME II [230] and SMIC [113]). The captured data with DAME was less controlled, with possible changes in the illumination, head pose and angle. Furthermore, a future implementation with real patients of DoC might require to deal with patients being filmed while using breathing and respiratory equipments which represent different levels of face occlusions. Thus, a more robust face detection and tracking system must be created before processing the captured DAME data.

Another thing to consider is the length of the videos. The micro-expression analysis method used for this project was tested in datasets with short videos. Longer video requires not

only to detect micro-expressions but also to dismiss several movements that might be just physiological motions and not represent real emotions. Furthermore, we need to design a system that is able to differentiate between micro and macro expressions. Fortunately, with the growing interest in the study of MEs, some new databases like CAS(ME)² [168] provide long videos with both macro and micro expressions.

For video analysis, we only used the gray scale images even though thermal images were also captured during the recording sessions. Thermal imaging has been used in the past for analysing facial macro expressions [84, 124, 143, 146] although not much has been done for ME. One of the main difficulties working with thermal images is localizing the faces. Although, there are some methods which exploit the differences of the thermal signature of a human face compared to the background in order to detect the face (See Sec. 2.2.1), these methods assume that the captured subject are aligned in a frontal position in a controlled environment (and as we have previously mentioned, that might not be necessarily the case). What's more, bandages, tubes and respiratory equipments that might be required for a patient well being, might occlude and show a different thermal signature compared to the rest face. Furthermore, traditional facial landmarks tracking methods such as ASM and AAM, which are trained with gray scale images, fail to accurately locate facial landmarks in thermal images (The facial topography captured by a gray scale camera differs to the one captured by a thermal camera). Fortunately, some authors have proposed to train an AAM using their own thermal imaging database and manual annotation [104]. Despite the aforementioned difficulties, the use of thermal imaging should not be discouraged. For instance, thermal imaging could point out very subtle facial motions, where the underlying facial musculature is active (blood flow), but not active enough to actually move the skin [93]. Additionally, thermal imaging could be used to obtain complementary physiological information which could help to do a better assessment of the patient state. For instance, human breathing patterns can be analyzed by measuring the thermal signaling variations from the nostrils during exhalation [70]. Also, heart rate could be analyzed by measuring the thermal signal variations of the face over time [30, 76].

There are also other type of data, other than thermal imaging, that would help us create a multimodal ME analysis system. For instance, analysing the neck area might help us discard certain mouth movements that appear as a result of saliva swallowing. Furthermore, certain aspects of emotion could be measured using different medical devices (which are already available in a hospital) such as electrocardiograms (ECG) to capture the heart rate.

Although, we were able to elicit and capture some MEs from the recorded videos, we believe the extracted information needs to be better standardized and annotated. However, after we perform certain needed changes in the methodology, we could potentially obtain a compacted database which can be shared with the scientific community. One change would be to cut the experiment session videos into smaller chunks. As mentioned in Sec. A.2.3, each recording session lasted about 5 minutes. That became a problem for the manual

annotators, which had to both rewatch, pause and rewind the natural videos several times and also inspect a very long continuous signal using the visualization tool (it will take them around 60 minutes to extract all MEs in one single video). Thus, the videos should be divided into 6 portions (one for each block of stimuli). In order to correctly cut the videos, either the recording and stimuli delivery programs should be temporally synchronized (So the videos could be cut using the time-stamps given by both programs) or put a visual indicator in the field of view of the recording cameras that signalizes the start and/or end of a block of stimuli (like the clapper board used while shooting a film). Another improvement would be to add an annotation tool to the visualization software. This would facilitate the work of the annotators and even save some time. Two major improvements would be needed for this upgrade. Firstly, the annotation tool must include an updated version of the ME spotting presented in this thesis which has addressed the issues presented in Sec. 4.4.4. This would help the annotators to quickly detect subtle facial events and speed up the annotation process. Secondly, an annotation panel should be added to the visualization interface, in which the annotator can write down the onset, apex and offset of a detected ME and the type of emotion is representing.

Quaternions

As mentioned in Sec. 3.2.3, the monogenic signal is represented using quaternions. Quaternions are a generalization of the complex numbers, in which there are three imaginary units, denoted i , j and k , so that each quaternion is characterized by four numbers, one real and three imaginary [82]:

$$a + bi + cj + dk \quad (\text{B.1})$$

where a, b, c, d are real numbers and i, j, k are the fundamental quaternion units. Let be two quaternions $\mathbf{q} = q_1 + q_2i + q_3j + q_4k$ and $\mathbf{r} = r_1 + r_2i + r_3j + r_4k$. The addition and subtraction of both of quaternion is affected by:

$$\mathbf{q} \pm \mathbf{r} = q_1 \pm r_1 + (q_2 \pm r_2)i + (q_3 \pm r_3)j + (q_4 \pm r_4)k \quad (\text{B.2})$$

Quaternion multiplication is associative and distributive with addition and can therefore be fully defined by the following property of the imaginary units:

$$-1 = i^2 = j^2 = k^2 = ijk \quad (\text{B.3})$$

Consequently the multiplication is non-commutative. Thus:

$$\begin{aligned} \mathbf{qr} &= (q_1r_1 - q_2r_2 - q_3r_3 - q_4r_4) \\ &\quad i(q_1r_2 + q_2r_1 + q_3r_4 - q_4r_3) \\ &\quad j(q_1r_3 - q_2r_4 + q_3r_1 + q_4r_2) \\ &\quad k(q_1r_4 + q_2r_3 - q_3r_2 + q_4r_1) \end{aligned} \quad (\text{B.4})$$

Following Eq. B.4, for a quaternion \mathbf{q} , its conjugate \mathbf{q}^* , norm $\|\mathbf{q}\|$ and inverse \mathbf{q}^{-1} are defined as:

$$\mathbf{q}^* = q_1 - iq_2 - jq_3 - kq_4 \quad (\text{B.5})$$

$$\|\mathbf{q}\| = \sqrt{q_1^2 + q_2^2 + q_3^2 + q_4^2} \quad (\text{B.6})$$

$$\mathbf{q}^{-1} = \frac{\mathbf{q}^*}{\|\mathbf{q}\|^2} \quad (\text{B.7})$$

B.1 Complex Exponential and Logarithms

The exponential function of a complex number $z = a + bi$ can be defined through a power series:

$$e^z = \sum_{n=0}^{\infty} \frac{z^n}{n!} = e^a (\cos(b) + i \sin(b)) \quad (\text{B.8})$$

The complex logarithm is supposed to be the inverse function of the complex exponential function. However, for a function to have an inverse, it must be injective. There is one problem, though: the complex exponential function is not injective ($e^{i2\pi} = e^{i4\pi} = 1$). One solution is to restrict the imaginary part of the logarithm in the range $[-\pi, \pi)$ (with this restriction we will obtain the principal value of $\log z$). Then, the logarithm of a complex number z is:

$$\log(z) = \log(\sqrt{a^2 + b^2}) + i \arctan\left(\frac{b}{a}\right) \quad (\text{B.9})$$

where \arctan correspond to the multi-valued inverse tangent. In the case of a unit complex number, the first term is 0 and:

$$\log(z) = i \arctan\left(\frac{b}{a}\right) \quad (\text{B.10})$$

which is just the principle value of the phase of the complex number. Therefore, complex logarithm and exponentiation give a useful way to go from a complex number to its phase and back again [208].

The exponential function of a quaternion $\mathbf{q} = q_1 + \mathbf{v}$ (where $\mathbf{v} = iq_2 + jq_3 + kq_4$) is defined by its power series as with the complex number (Eq. B.8):

$$e^{\mathbf{q}} = \sum_{n=0}^{\infty} \frac{\mathbf{q}^n}{n!} = e^{q_1} \left(\cos(\|\mathbf{v}\|) + \frac{\mathbf{v}}{\|\mathbf{v}\|} \sin(\|\mathbf{v}\|) \right) \quad (\text{B.11})$$

The inverse of this function is:

$$\log(\mathbf{q}) = \log(\|\mathbf{q}\|) + \frac{\mathbf{v}}{\|\mathbf{v}\|} \arccos\left(\frac{q_1}{\|\mathbf{q}\|}\right) \quad (\text{B.12})$$

In the case of a unit quaternion, where $\|\mathbf{q}\| = 1$, this simplifies to:

$$\log(\mathbf{q}) = \frac{\mathbf{v}}{\|\mathbf{v}\|} \arccos(q_1) \quad (\text{B.13})$$

Subtle Motion Dataset

In this section, we describe the construction of the subtle motion sequences dataset used in the experiments in Sec. 3.5.2. The image sequences were created by taking a static image sample, repeating it for several frames, then one or two instances of subtle motion are added and then the resulting image is again repeated for several frames.

Two types of motions are simulated:

- **Global motion:** they were obtained by either translating the whole image one pixel or zooming in (by resizing the image one pixel in each direction and then cropping the image one pixel in each direction to keep the original size) during two frames (see Fig. C.1a).
- **Local motion:** One object of the image was translated one pixel in any direction while the other objects remain static (see Fig. C.1b).

We used 17 sample images (see Fig. C.2) which were either artificially created (top row) or natural images downloaded from the internet (middle and bottom rows). For the natural sample images we selected image with simple or blurry background (middle row) and textured background (bottom row).

We created 18 sequences, one from each image except for one image (Fig. C.2 first image top row) from which we created 2 sequences (one moving the animals and one moving the donut). We used the 4 first images in top row to create 5 sequences of local motion and

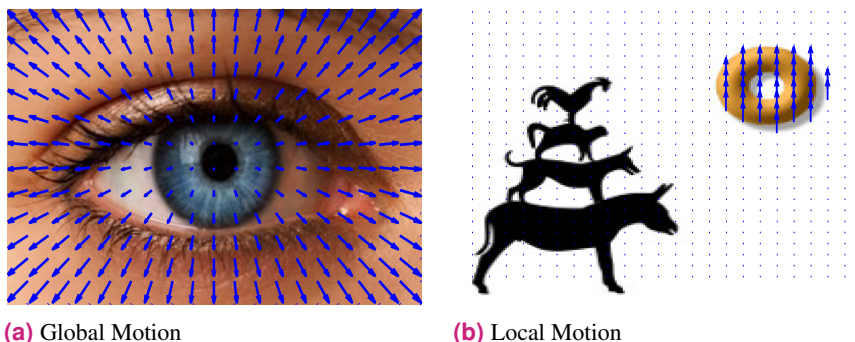


Figure C.1.: Simulated Motion

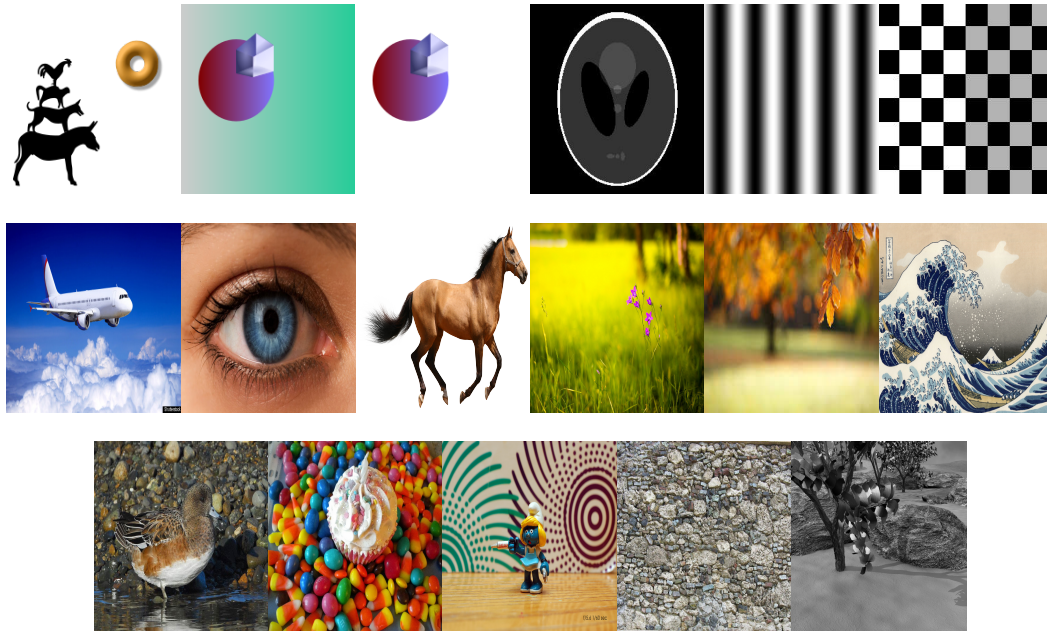


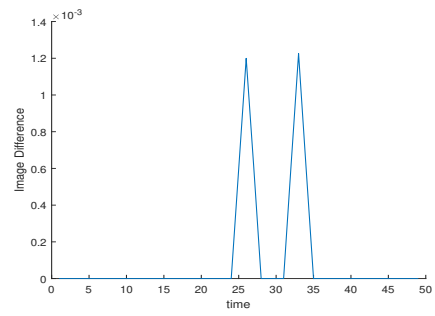
Figure C.2.: Sample images used in the dataset

the rest were used for global motion (although, it can be argued that globally translating an image with plain background such as the horse image in the middle row of Fig. C.2 can be considered as local motion).

Each sequence was composed of 50 frames, from which the first 24 frames were static, then the motion was simulated from frames 25 to 26 or 27. If the sequence had a second instance of motion, it would be simulated from frames 33 to 35. The rest of the frames would be copies of the last simulated motion image. The sequences were created in this way so any method, even one as basic as subtracting consecutive frames, is able to create a signal where we can easily detect when the subtle motion takes place before we add any type of noise (see Fig. C.3a and Fig. C.3b). However once we added different levels of image noise, the ability to correctly spot the subtle motion would depend on the robustness of the method to distinguish subtle motion from image noise. For example, we can see how, under the presence of noise (Fig. C.3c), image difference fails to create a signal where we can correctly spot subtle motion (Fig. C.3d).



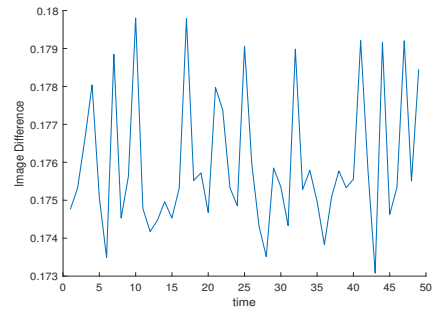
(a) Image without noise



(b) Image difference without noise



(c) Noisy image



(d) Image difference with noise

Figure C.3.: Differences between image sequence with and without noise

Support Vector Machine

For the classification step, we chose to use the well known support vector machine thanks to their tendency to separate the data in hyperplanes that can be linearly separable. Also it can generalize well as long as the dimensionality of the data is not too high. This section goes as follows: Sec. D.1 presents the classical formulation; Sec. D.2 the non-linear extension of SVM and the kernel trick; Finally, Sec. D.3 talks about different approaches for optimizing the machine's hyperparameters. The formulation presented in this section can be found in [108].

D.1 Classical Formulation

Support Vector Machines (SVM) are supervised learning models for classification and regression models. They are linear classifiers although they can solve linear and non-linear problems. The idea of SVM is simple: to create a hyperplane or set of hyperplanes that can separate data into different classes. The goal of the algorithm is to construct a hyperplane that represents the largest separation, or margin, between classes.

For the binary classification case, we are given a dataset consisting of m points in the n -dimensional real space \mathbb{R}^n . Each point in the dataset comes with a class label y , $+1$ or -1 , indicating one of two classes, Y_+ and Y_- to which the point belongs. We represent these data points by an $m \times n$ matrix Y , where the i^{th} row of the matrix A , \mathbf{x}_i , corresponds to the i^{th} data point.

D.1.1 Hard-margin

Let's start the formulation with a strictly separable case (there exist a hyperplane that can separate the data Y_+ and Y_-). In this case the two classes can be separated by a pair of parallel bounding planes [108] (see Fig. D.1a):

$$\begin{aligned} \mathbf{w}^\top \mathbf{x} + b &\geq +1, & \text{for } \mathbf{x} \in Y_+ \\ \mathbf{w}^\top \mathbf{x} + b &\leq -1, & \text{for } \mathbf{x} \in Y_- \end{aligned} \tag{D.1}$$

where \mathbf{w} is the normal vector to the planes and b determines their location relative to the origin. The first plane of Eq. D.1 bounds the class Y_+ and the second plane bounds the class Y_- .

SVM achieves a better prediction ability via maximizing the margin between two bounding planes. Hence, SVM searches for a separating hyperplane by maximizing $\frac{2}{\|\mathbf{w}\|_2}$. It can be done by means of minimizing $\frac{1}{2}\|\mathbf{w}\|_2^2$ and leads to a quadratic program, as follows:

$$\begin{aligned} \min_{(\mathbf{w}, b) \in \mathbb{R}^{n+1}} \quad & \frac{1}{2} \|\mathbf{w}\|_2^2 \\ \text{s. t.} \quad & y_i (\mathbf{w}^\top \mathbf{x}_i + b) \geq 1 \quad \text{for } i = 1, 2, \dots, m \end{aligned} \quad (\text{D.2})$$

The linear separating hyperplane is the plane:

$$\mathbf{w}^\top \mathbf{x} + b = 0 \quad (\text{D.3})$$

in the middle of the bounding planes Eq. D.1 as shown in Fig. D.1a. For the linearly separable case there exists an optimal solution (\mathbf{w}^*, b^*) . The data points on the bounding planes, $\mathbf{w}^{*\top} \mathbf{x} + b^* = \pm 1$, are called support vectors. If we remove any point which is not a support vector, the training result will not be changed. This optimization problem is known as the primal problem.

D.1.2 Soft-Margin

If the classes are linearly inseparable then the two planes bound the two classes with a “soft margin” determined by a non-negative slack vector variable ξ , that is:

$$\begin{aligned} \mathbf{w}^\top \mathbf{x}_i + b + \xi_i &\geq +1, \quad \text{for } \mathbf{x}_i^\top \in Y_+ \\ \mathbf{w}^\top \mathbf{x}_i + b - \xi_i &\leq -1, \quad \text{for } \mathbf{x}_i^\top \in Y_- \end{aligned} \quad (\text{D.4})$$

The 1-norm of the slack variable ξ , $\sum_{i=1}^m \xi_i$, is called the penalty term. The principle is to determine a separating hyperplane that not only correctly classifies the training data, but also performs well on a testing set (As shown in Fig. D.1b). With a soft margin, the Eq. D.2 can be extended such as:

$$\begin{aligned} \min_{(\mathbf{w}, b, \xi) \in \mathbb{R}^{n+1+m}} \quad & \frac{1}{2} \|\mathbf{w}\|_2^2 + C \sum_{i=1}^m \xi_i \\ \text{s. t.} \quad & y_i (\mathbf{w}^\top \mathbf{x}_i + b) + \xi_i \geq 1 \quad \xi_i \geq 0 \quad \text{for } i = 1, 2, \dots, m \end{aligned} \quad (\text{D.5})$$

where $C > 0$ is a positive parameter that balances the weight of the penalty term $\sum_{i=1}^m \xi_i$ and the margin maximization term $\frac{1}{2} \|\mathbf{w}\|_2^2$.

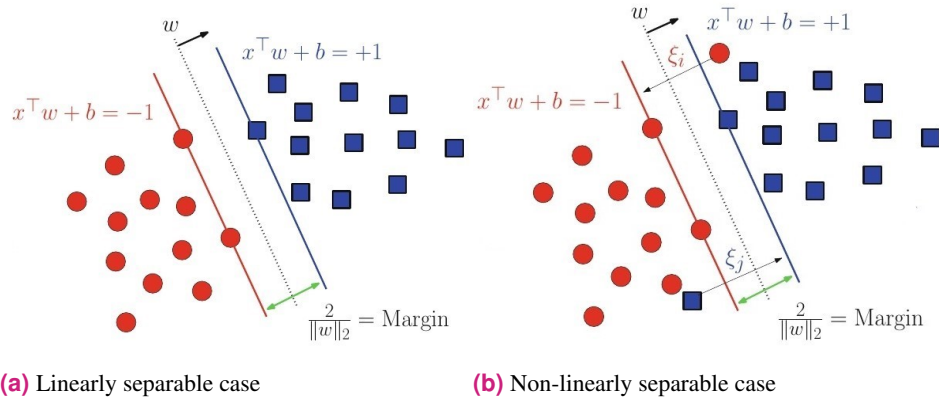


Figure D.1.: Linear vs. Non-linearly separable cases

D.2 Non-linear Extension

In many cases, a dataset, as collected in a vector form full of attributes, cannot be well separated by a linear separating hyperplane. However, it is likely that the dataset becomes linearly separable after mapped into a higher dimensional space by a non-linear map. A nice property of SVM methodology is that the non-linear map does not need to be explicitly known; still, we can apply a linear algorithm to the classification problem in the high dimensional space. In order to do so, we need to understand the dual problem of Eq. D.2 and the “kernel trick” [108].

D.2.1 Dual Problem

In the previous section we talked about the primal problem formulation. An alternative formulation called the dual problem of Eq. D.5 is as follows:

$$\begin{aligned}
 \max_{c \in \mathbb{R}^m} \quad & \sum_{i=1}^m c_i - \frac{1}{2} \sum_{i,j=1}^m y_i y_j c_i c_j \langle \mathbf{x}_i, \mathbf{x}_j \rangle \\
 \text{s. t.} \quad & \sum_{i=1}^m y_i c_i = 0 \quad 0 \leq c_i \leq C \quad \text{for } i = 1, 2, \dots, m
 \end{aligned} \tag{D.6}$$

where $\langle \mathbf{x}_i, \mathbf{x}_j \rangle$ is the inner product of \mathbf{x}_i and \mathbf{x}_j . The primal variable \mathbf{w} is given by:

$$\mathbf{w} = \sum_{\{i|c_i>0\}}^m y_i c_i \mathbf{x}_i \tag{D.7}$$

where the c_i corresponds to a training point \mathbf{x}_i . The normal vector \mathbf{w} can be expressed in terms of a linear combination of training data points which have corresponding positive dual

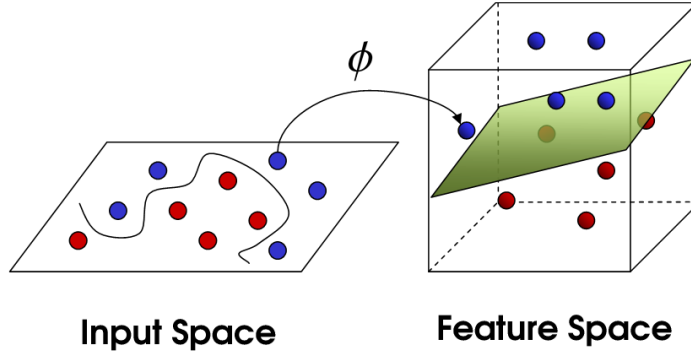


Figure D.2.: Illustration of non-linear SVM

variables c_i (namely, the support vectors). By the Karush-Kuhn-Tucker complementarity conditions [108]:

$$\begin{aligned} c_i &\geq 0, & y_i(\mathbf{w}^\top \mathbf{x}_i + b) + \xi_i - 1 &\geq 0 \\ C &\geq c_i, & \xi_i &\geq 0, \quad \text{for } i = 1, 2, \dots, m \end{aligned} \quad (\text{D.8})$$

and b can be simply determined by taking any training point, x_i , such that $i \in I := \{k | 0 < c_k < C\}$ and obtain:

$$b = y_i - \mathbf{w}^\top \mathbf{x}_i = y_i - \sum_{j=1}^m (y_j c_j \langle \mathbf{x}_j, \mathbf{x}_i \rangle) \quad (\text{D.9})$$

D.2.2 Kernel trick

From the dual SVM formulation (Eq. D.6), all we need to know is simply the inner product between training data vectors. Let us map the training data points from the input space \mathbb{R}^n to a higher-dimensional feature space F by a non-linear map $\Phi(\mathbf{x}) \in \mathbb{R}^\ell$ where ℓ is the dimensionality of the feature space F . Based on the above observation, if the inner product $\Phi(\mathbf{x}_i)^\top \Phi(\mathbf{x}_j) \forall i, j = 1, 2, \dots, m$ is known, then the linear SVM algorithm can be performed in the feature space F . The separating hyperplane will be linear in the feature space F but is a non-linear surface in the input space \mathbb{R}^n (See Fig. D.2). Note that the non-linear map Φ does not need to be explicitly known and it can be achieved by applying a kernel function. Let $k(\mathbf{x}, \mathbf{z}) : \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}^m$ be an inner product kernel, then a non-linear map Φ can be constructed such that $k(\mathbf{x}_i, \mathbf{x}_j) = \Phi(\mathbf{x}_i)^\top \Phi(\mathbf{x}_j)$, $i, j = 1, 2, \dots, m$. The resulting dual non-linear SVM formulation becomes:

$$\begin{aligned} \max_{\mathbf{c} \in \mathbb{R}^m} & \sum_{i=1}^m c_i - \frac{1}{2} \sum_{i,j=1}^m y_i y_j c_i c_j k(\mathbf{x}_i, \mathbf{x}_j) \\ \text{s. t.} & \sum_{i=1}^m y_i c_i = 0 \quad 0 \leq c_i \leq C \quad \text{for } i = 1, 2, \dots, m \end{aligned} \quad (\text{D.10})$$

The non-linear separating hyperplane is defined by the solution of Eq. D.10 as follows:

$$\sum_{j=1}^m y_j c_j k(\mathbf{x}_i, \mathbf{x}_j) + b = 0 \quad (\text{D.11})$$

where

$$b = y_i - \sum_{j=1}^m y_j c_j k(\mathbf{x}_i, \mathbf{x}_j), \quad i \in I := \{k | 0 < c_k < C\} \quad (\text{D.12})$$

Let us see some common kernels used with SVMs:

- **Linear Kernel:** $k(\mathbf{x}, \mathbf{z}) = \mathbf{x}^\top \mathbf{z}$.
- **Polynomial Kernel:** $k(\mathbf{x}, \mathbf{z}) = (\mathbf{x}^\top \mathbf{z} + b)^d$ where d is the degree of the exponentiation.
- **Gaussian Radial Basis function (RBF):** $k(\mathbf{x}, \mathbf{z}) = e^{-\gamma \|\mathbf{x} - \mathbf{z}\|_2^2}$ where γ is the width parameter of Gaussian kernel.

D.3 Hyperparameter Tuning

In machine learning, the parameters are the variables that are learned and adjusted from the training data (\mathbf{w} and b in the case of SVM). However, there are some other parameters that can't be directly learned from the training process. These parameters that express a "higher level" properties are called hyperparameters. Hyperparameters are usually fixed before the actual training process begins. Some examples of hyperparameters are the number of trees in a random forest, the number of hidden layers in a neural network, the learning rate in logistic regression, etc. In the case of SVM, some important hyperparameters are the box constraint (C in Eq. D.2) which is a parameter that controls the maximum penalty imposed on margin-violating observations and the kernel scale (the elements of the predictor data \mathbf{x} are divided by this value).

Hyperparameters are optimized or "tuned" by setting different values for the hyperparameters, training different models, and deciding which one works best by testing the models. There are different approaches:

- **Grid Search:** It is an exhaustive search through a manually specified search of hyperparameters. Grid search train the algorithm in each set of possible combinations of hyperparameters and measure their performance. Finally the hyperparameter which gives the maximum score is the final output. The main drawbacks is that this method suffers the curse of dimensionality.

- **Random Search:** This method searches for random combinations of hyper parameter values for a number of defined iterations. It takes less time than grid search but it doesn't use information from prior experiments and there is not guarantee to obtain the best possible combination of hyperparameterers
- **Bayesian Optimization:** It sees the hyperparameter tuning as a blackbox optimization problem. It proposes to use a surrogate function to approximate the blackbox function (which is easier to optimize). Bayesian methods work by finding the next set of hyperparameters to evaluate on the actual blackbox function by selecting hyperparameters that perform best on the surrogate function [102].

Additional Classification Experiments

In this section, we show some additional partial macro-expression and ME recognition experiments. This section serves a complement for Sec. 5. This section goes as follows: Sec.E.1 compares the results of partial macro-expressions for the MOOF descriptor using Lucas-Kanade and TV-L1 approaches for optical flow. In Sec. E.2, we evaluate the regularization parameter λ of TV-L1 optical flow for ME recognition. Sec. E.3 shows the ME recognition rate of the amplitude weighted and amplitude masked variation of the MORF descriptor. Finally, Sec. E.4 presents one last experiment masking the eye regions. All the experiments in this section were classified using SVMs with RBF kernels and the results were tested by a 10-fold validation method.

E.1 Lucas-Kanade vs. TV-L1 optical flow

During the development of this thesis, we have mainly compared our method with two types of optical flow: Lucas-Kanade and TV-L1 [159]. For the experiment in Sec. 3.5.2, we could only test the efficiency of Lucas-Kanade optical flow approach due to the long processing time of the TV-L1 method. However, the advantage of using TV-L1 is that we can tune the attachment parameter λ to choose either a smooth or a sensible optical flow. Thus, we wanted to compare which optical flow method would yield better results for facial expression recognition. We compare both methods using the protocol of experiment one (Sec. 5.3.1). We use the same database and partial macro-expressions sequences for testing. The videos are processed using TV-L1 optical flow with three values for the attachment parameter $\lambda = \{25, 50, 200\}$ and using Lucas-Kanade optical flow with three values for the “Noise Threshold” parameter = $\{0.1, 0.2, 0.3\}$ ¹. Then we use the obtained results to create a mean oriented OF image pair and process it to obtain the MOOF features. The obtained results for Lucas-Kanade are called $MOOF_{LK-A}$, $MOOF_{LK-B}$ and $MOOF_{LK-C}$ (where A correspond to the first variation of the attachment parameter $\lambda = 25$ and so on). The obtained results for TV-L1 OF are called $MOOF_{TV-L1-A}$, $MOOF_{TV-L1-B}$ and $MOOF_{TV-L1-C}$ (where A correspond to the first variation of the “Noise Threshold”= 0.1).

¹According to Matlab: “The more you increase this number, the less an object’s movement affects the optical flow calculation.”

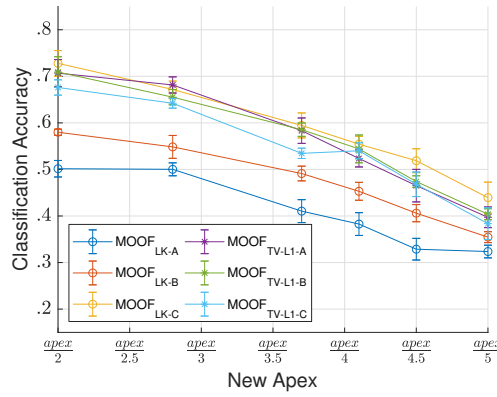


Figure E.1.: Results of partial macro-expressions classification

The results can be seen in Fig. E.1. It seems that $\text{MOOF}_{\text{LK-C}}$, $\text{MOOF}_{\text{TV-L1-A}}$ and $\text{MOOF}_{\text{TV-L1-B}}$ have the best overall results for the partial macro-expressions. However the TV-L1 results does not vary to much between each other, while the Lucas-Kanade results vary greatly depending on the value of the “Noise Threshold”. This is the main reason why we chose to compute the MOOF descriptor with TV-L1 optical flow for the experiments one and two of Sec. 5.3.

E.2 TV-L1 Optical Flow Parameter Evaluation

We wanted to measure the effect of the TV-L1 optical flow parameter λ for ME recognition. We use the same protocol of the second experiment (Sec. 5.3.2) to create the MOOF descriptor but this time we vary the value of $\lambda = \{5, 10, 15, 25, 50, 200\}$.

The results can be seen in Table. E.1 and Table. E.1². For both databases, if we move the parameter from the highest to lowest value, we can see that there is an increase in accuracy from $\lambda = 200$ until $\lambda = 15$, then it starts to decrease for $\lambda = 10$ and $\lambda = 5$. The results seem to suggest that for smaller values of λ , the optical flow is more robust against noise, but after certain value the smooth motion flow underestimates subtler motions (as initially discussed in Sec. 5.3.2). In the end, the obtained results for both databases are lower than the ones using the MORF descriptor, thus, we decide not to pursue further experimentation with this method.

E.3 Amplitude based MORF

In this experiment, we want to test the amplitude based variations of the MORF descriptor for ME recognition. We compare both methods using the protocol of experiment two

²The results for $\lambda = \{25, 50, 100\}$ are the same ones presented in Sec. 5.3.2

SMIC-HS Database							
Parameters		MOOF					
Grid division	Orientation bins	λ					
		5	10	15	25	50	200
$G = 6 \times 6$	$O = 6$	0.3963	0.4111	0.4111	0.4000	0.3704	0.2963
	$O = 8$	0.3778	0.3704	0.4000	0.4074	0.3333	0.2963
	$O = 10$	0.3519	0.4222	0.4370	0.4185	0.3815	0.3148
$G = 8 \times 8$	$O = 6$	0.3741	0.4148	0.4481	0.4111	0.3963	0.3259
	$O = 8$	0.3630	0.3889	0.4593	0.4222	0.3704	0.3222
	$O = 10$	0.3667	0.3889	0.4259	0.3963	0.3815	0.3148

Table E.1.: TV-L1 MOOF results for different values of λ for the SMIC-HS database

CASME II Database							
Parameters		MORF			MOOF		
Grid division	Orientation bins	λ					
		5	10	15	25	50	200
$G = 6 \times 6$	$O = 6$	0.3794	0.3344	0.3569	0.3441	0.3248	0.2669
	$O = 8$	0.3441	0.3794	0.3601	0.3537	0.3280	0.2862
	$O = 10$	0.3633	0.3215	0.3248	0.3569	0.3151	0.3473
$G = 8 \times 8$	$O = 6$	0.4019	0.3408	0.3698	0.3376	0.3312	0.2830
	$O = 8$	0.3183	0.3730	0.3923	0.3537	0.3762	0.2830
	$O = 10$	0.3762	0.3408	0.3601	0.3473	0.3601	0.3087

Table E.2.: TV-L1 MOOF results for different values of λ for the CASME II database

(Sec. 5.3.2). We use the same databases and evaluation procedure described in Sec. 5.3.2. We obtain the amplitude weighted MORF by multiplying the quaternionic filtered phase $(\phi \cos(\theta), \phi \sin(\theta))$ obtained from a Riesz pyramid level with their respective local amplitude A (see Fig. E.2a) before calculating the MOR image pair (see Fig. E.2b). We obtain the amplitude masked MORF by multiplying the quaternionic filtered phase with an amplitude mask using the method of Sec. 3.4.2 (see Fig. E.2c) with a threshold $\beta = 0.1$ before calculating the MOR image pair (see Fig. E.2d). We also propose to create an amplitude weighted and Masked MORF by just combining the previous methods.

The results can be seen in Table. E.3. As we can see the best results are obtained for the amplitude weighted MORF and the worst results for the amplitude masked MORF (The combination of amplitude weighted and masked MORF obtained slightly better results than masked MORF except for CASME II at the fourth level of the Riesz pyramid). In the case of weighted MORF, the facial areas with high amplitude such as the mouth, eyes and eyebrows are highlighted and have a cast a heavier vote in the final ME feature vector. However, the weight of areas like the cheeks, which provide interesting information about the ME motion, is heavily reduced. In the case of masked MORF, low amplitude areas that produce unwanted noise are ignored but at the expense of areas which contain important information about ME. Furthermore, if we check the results of ME spotting parameter analysis (Sec. 4.4.3), the optimal value for the parameter β for the second level of the pyramid should be higher ($0.2 \leq \beta \leq 0.3$). In the end, the obtained results are lower than the ones using the normal

Amplitude Weighted MORF							
Parameters		SMIC-HS			CASME II		
Grid division	Orientation bins	Riesz Level			Riesz Level		
		2	3	4	2	3	4
$G = 6 \times 6$	$O = 6$	0.4593	0.4306	0.4402	0.5356	0.5131	0.4569
	$O = 8$	0.4067	0.4785	0.4641	0.5019	0.4457	0.5169
	$O = 10$	0.3971	0.4163	0.4498	0.5206	0.4869	0.4082
$G = 8 \times 8$	$O = 6$	0.4880	0.4641	0.4354	0.4494	0.4607	0.4307
	$O = 8$	0.4498	0.4785	0.4689	0.5019	0.4532	0.5056
	$O = 10$	0.5167	0.4545	0.4306	0.4981	0.4382	0.4869
Amplitude Masked MORF							
Parameters		SMIC-HS			CASME II		
Grid division	Orientation bins	Riesz Level			Riesz Level		
		2	3	4	2	3	4
$G = 6 \times 6$	$O = 6$	0.3062	0.3349	0.3828	0.1386	0.2434	0.4682
	$O = 8$	0.2871	0.4067	0.2967	0.1236	0.2210	0.4981
	$O = 10$	0.2632	0.3301	0.3541	0.1311	0.2247	0.4345
$G = 8 \times 8$	$O = 6$	0.2967	0.3684	0.3254	0.1536	0.2397	0.4082
	$O = 8$	0.2919	0.3493	0.3493	0.1423	0.2547	0.4607
	$O = 10$	0.3014	0.3206	0.3158	0.1423	0.1760	0.4419
Amplitude Weighted + Amplitude Masked MORF							
Parameters		SMIC-HS			CASME II		
Grid division	Orientation bins	Riesz Level			Riesz Level		
		2	3	4	2	3	4
$G = 6 \times 6$	$O = 6$	0.3732	0.4402	0.4115	0.1835	0.2846	0.3633
	$O = 8$	0.3349	0.4306	0.3971	0.1798	0.2734	0.4757
	$O = 10$	0.3062	0.4928	0.4258	0.1723	0.2622	0.4082
$G = 8 \times 8$	$O = 6$	0.3062	0.4928	0.4258	0.1723	0.2622	0.4082
	$O = 8$	0.3589	0.4641	0.3923	0.2285	0.2809	0.4757
	$O = 10$	0.3014	0.4737	0.4163	0.1798	0.2809	0.4082

Table E.3.: Recognition rates with respect to different set of parameters for the amplitude based MORF descriptors

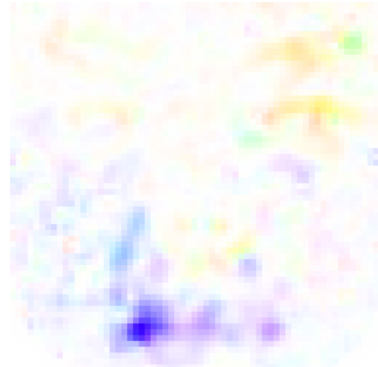
MORF descriptor, thus, we decide not to pursue further experimentation with these methods.

E.4 Masked Eyes MORF

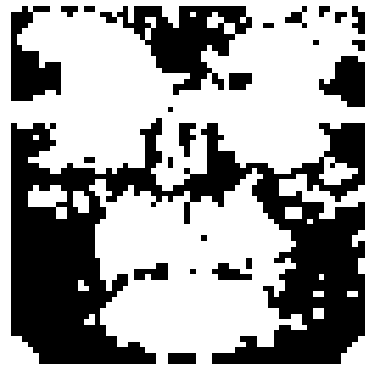
In this experiment we wanted to test the effect of the eyes region for ME recognition. As discussed in Sec. 4.3, eye blinks and eye gaze changes could be wrongfully considered as MEs. Thus, we decided to mask these regions, extract the MORF vector, and compare it to the feature vectors obtained using the whole face. First, we combine the curved mask from Sec. 5.2.1 with a mask created using the eyes regions from Sec. 4.1.4 (see Fig. E.2e). Then, we mask the quaternionic filtered phase with the mask and create the MOR image pair (See Fig. E.2f). We do the same evaluation procedure for experiment two (Sec. 5.3.2).



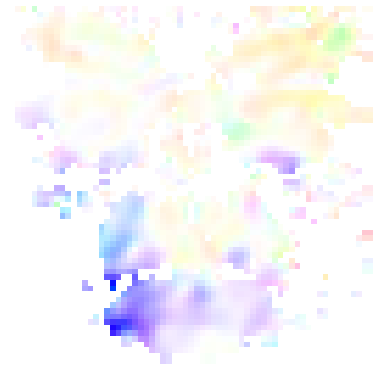
(a) Local Amplitude



(b) Amplitude Weighted MOR Image Pair



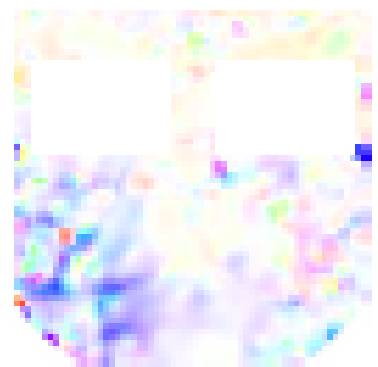
(c) Local Amplitude Mask



(d) Masked MOR Image Pair



(e) Curved mask without eyes



(f) Masked MOR Image pair

Figure E.2.: Masks and Masked MOR Image pairs for different MORF variations

Eyes Masked MORF							
Parameters		SMIC-HS			CASME II		
Grid division	Orientation bins	Riesz Level			Riesz Level		
		2	3	4	2	3	4
$G = 6 \times 6$	$O = 6$	0.4778	0.4926	0.3963	0.4984	0.5145	0.5113
	$O = 8$	0.4185	0.4815	0.3556	0.4630	0.5080	0.5949
	$O = 10$	0.4296	0.5407	0.3963	0.5177	0.5080	0.5563
$G = 8 \times 8$	$O = 6$	0.4444	0.5074	0.3519	0.5145	0.5241	0.5659
	$O = 8$	0.4704	0.5481	0.4111	0.5498	0.5273	0.5273
	$O = 10$	0.4630	0.5111	0.3852	0.5691	0.5466	0.5402

Table E.4.: MORF recongition rates with masked eyes

The results of the classification can be seen in Table. E.4. In general, the results of MORF with masked eyes are lower than the ones of normal MORF (Table. 5.3 and Table. 5.4). This seems to confirm that, regardless of the eye blinks and eye gaze changes, the eye areas contains useful information for ME recognition.

List of Figures

1.1	7 basic emotions [54]	6
1.2	Graphical representation of affect/valence model	7
1.3	Intensity of facial expressions [168]	8
1.4	Facial Action Units (AUs) of upper and lower face [60]	10
2.1	Taxonomy for AFER in computer vision	18
2.2	Face detection example of faces in different poses, with different illuminations and occlusions	19
2.3	Face registration examples	21
2.4	Facial ROIs	22
2.5	Optical Strain example	24
2.6	LBP-TOP is a 3D variant of LBP which considers the co-occurrence statistic on three orthogonal planes: XY, XT and YT. [216]	27
2.7	Heuristic ME spotting general framework	28
2.8	Trained ME spotting general framework	29
2.9	General Framework for different modality fusion approaches.	34
2.10	ME research over the years	36
3.1	Feature matching	43
3.2	Phase Correlation	44
3.3	Block Matching Algorithm [41]	45
3.4	Lucas-Kanade Dense Optical Flow [175]	46
3.5	Noise reduction effect in subtle motion estimation	47
3.6	Visual representation of image pyramid with 4 levels	49
3.7	Eulerian motion amplification framework [222]	50
3.8	Phase-based motion amplification framework [207]	51
3.9	Motion Magnification Results.	52
3.10	Local amplitude and quaternionic phase of different levels of the Riesz pyramid	56
3.11	Subtle motion analysis framework.	59
3.12	Non-periodic signal filtering representation	60

3.13	A comparison of different filter responses for subtle motion detection.	61
3.14	Isolating relevant quaternionic phase using the amplitude mask.	62
3.15	Subtle motion detection for breathing measurement for a baby monitoring system.	64
3.16	Subtle motion spectral analysis of a vibrating drum.	65
3.17	Performance curves of different subtle motion spotting techniques in presence of different levels of Gaussian noise	68
3.18	Performance curves of different subtle motion spotting techniques in presence of different levels of Salt and Pepper noise	68
4.1	ME spotting framework.	74
4.2	Facial features bounding boxes and their centroids.	75
4.3	Detecting the edge between lips.	76
4.4	Face ROI localization and tracking	76
4.5	Masking facial regions of interest.	79
4.6	Micro-expression spotting process.	80
4.7	Filter gains for FIR filters	84
4.8	ROC curves for ME spotting on SMIC-HS and CASME II datasets	85
4.9	Result surfaces for the different levels of the Riesz pyramid for the SMIC-HS dataset	87
4.10	Contour plots for the results of the different levels of the Riesz pyramid for the SMIC-HS dataset	87
4.11	Result surfaces for the different levels of the Riesz pyramid for the CASME II dataset	88
4.12	Contour plots for the results of the different levels of the Riesz pyramid for the CASME II dataset	88
4.13	Parameter Evaluation results in SMIC-HS and CASME II datasets	89
4.14	Face registration failure case.	91
4.15	Failure case when our system has wrongfully detected the peaks produced by macro-movements as MEs	91
4.16	Failure case when our system have created a tall threshold which ignores the peak produced by the true ME	92
5.1	False-Color representation of oriented quaternionic phase response to different motions	94
5.2	Riesz Oriented Phase of an Onset-Apex sequence	97

5.3	MOR Image Pair	98
5.4	Face Masking	99
5.5	Implementation of the MORF descriptor	100
5.6	Orientation bin construction	101
5.7	Macro-expressions sequence at different time-stamps	104
5.8	Results of partial macro-expressions classification	106
5.9	Parameter variation of MORF and MOOF	107
5.10	Confusion Matrices at the best recognition rate by the LOSO cross-validation	112
A.1	Simplified illustration of the two major components of consciousness and the way they correlate within the different DoC [49]	146
A.2	DAME image acquisition and stimuli delivery station	149
A.3	DAME visualization tool	151
C.1	Simulated Motion	157
C.2	Sample images used in the dataset	158
C.3	Differences between image sequence with and without noise	159
D.1	Linear vs. Non-linearly separable cases	163
D.2	Illustration of non-linear SVM	164
E.1	Results of partial macro-expressions classification	168
E.2	Masks and Masked MOR Image pairs for different MORF variations	171

List of Tables

1.1	Estimated micro-expression duration under different frame rates	9
1.2	ME posed databases	12
1.3	ME spontaneous datasets	13
2.1	Heuristic ME spotting methods	30
2.2	Trained ME spotting methods	30
2.3	ME classifier methods	40
3.1	Running times (in seconds) of comparable MATLAB implementations of subtle motion spotting methods	70
4.1	AUC values of the ME spotting experiments using different methods	85
5.1	Parameter evaluation results for MORF	107
5.2	Parameter evaluation results for MOOF	107
5.3	MORF vs MOOF classification for the SMIC-HS database	110
5.4	MORF vs MOOF classification for the CASME II database	110
5.5	ME classification for SMIC-HS	112
5.6	ME classification for CASME II	112
5.7	Recognition rates with respect to different set of parameters for the SMIC-HS dataset	113
5.8	Recognition rates with respect to different set of parameters for the CASME II dataset	114
5.9	ME recognition accuracy of comparable methods of the state of the art	116
E.1	TV-L1 MOOF results for different values of λ for the SMIC-HS database	169
E.2	TV-L1 MOOF results for different values of λ for the CASME II database	169
E.3	Recognition rates with respect to different set of parameters for the amplitude based MORF descriptors	170
E.4	MORF recongition rates with masked eyes	172

List of Publications

Carlos Andres Arango, Olivier Alata, Rémi Emonet, Anne-Claire Legrand, and Hubert Konik. “Subtle Motion Analysis and Spotting using the Riesz Pyramid”. In: *Proceedings of the 13th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications - Volume 5: VISAPP*. INSTICC. SciTePress, 2018, pp. 446–454.

C. A. Duque, O. Alata, R. Emonet, A. C. Legrand, and H. Konik. “Micro-Expression Spotting Using the Riesz Pyramid”. In: *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*. Mar. 2018, pp. 66–74.

A. Bertholon, C. Arango Duque, O. Alata, R. Emonet, A.C. Legrand, H. Konik, et al. “Validation in healthy subjects of a clinical protocol for the evaluation of facial micro-expressions in severely brain injured patients awakening from coma”. In: *Annals of Physical and Rehabilitation Medicine* 61 (2018). 12th World Congress of the International Society of Physical and Rehabilitation Medicine. Paris. 8-12 July 2018, e426.

National Conference

Carlos Arango. “Analysis and detection of facial micro-expressions using the Riesz pyramid”. In: *Journée Visage, geste, action et comportement*. TELECOM ParisTech, Dec. 2017.

Résumé Les micro-expressions sont des expressions faciales brèves et subtiles qui apparaissent et disparaissent en une fraction de seconde. Ce type d'expressions reflèterait "l'intention réelle" de l'être humain. Elles ont été étudiées pour mieux comprendre les communications non verbales et dans un contexte médicale lorsqu'il devient presque impossible d'engager une conversation ou d'essayer de traduire les émotions du visage ou le langage corporel d'un patient. Cependant, détecter et reconnaître les micro-expressions est une tâche difficile pour l'homme. Il peut donc être pertinent de développer des systèmes d'aide à la communication exploitant les micro-expressions. De nombreux travaux ont été réalisés dans les domaines de l'informatique affective et de la vision par ordinateur pour analyser les micro-expressions, mais une grande majorité de ces méthodes repose essentiellement sur des méthodes de vision par ordinateur classiques telles que les motifs binaires locaux, les histogrammes de gradients orientés et le flux optique. Étant donné que ce domaine de recherche est relativement nouveau, d'autres pistes restent à explorer. Dans cette thèse, nous présentons une nouvelle méthodologie pour l'analyse des petits mouvements (que nous appellerons par la suite mouvements subtils) et des micro-expressions. Nous proposons d'utiliser la pyramide de Riesz, une approximation multi-échelle et directionnelle de la transformation de Riesz qui a été utilisée pour l'amplification du mouvement dans les vidéos à l'aide de l'estimation de la phase 2D locale. Pour l'étape générale d'analyse de mouvements subtils, nous transformons une séquence d'images avec la pyramide de Riesz, extrayons et filtrons les variations de phase de l'image. Ces variations de phase sont en lien avec le mouvement. De plus, nous isolons les régions d'intérêt où des mouvements subtils pourraient avoir lieu en masquant les zones de bruit à l'aide de l'amplitude locale. La séquence d'image est transformée en un signal 1D utilisé pour l'analyse temporelle et la détection de mouvements subtils. Nous avons créé notre propre base de données de séquences de mouvements subtils pour tester notre méthode. Pour l'étape de détection de micro-expressions, nous adaptons la méthode précédente au traitement de certaines régions d'intérêt du visage. Nous développons également une méthode heuristique pour détecter les micro-événements faciaux qui sépare les micro-expressions réelles des clignotements et des mouvements subtils des yeux. Pour la classification des micro-expressions, nous exploitons l'invariance, sur de courtes durées, de l'orientation dominante issue de la transformation de Riesz afin de moyennner la séquence d'une micro-expression en une paire d'images. A partir de ces images, nous définissons le descripteur MORF (Mean Oriented Riesz Feature) constitué d'histogrammes d'orientation. Les performances de nos méthodes sont évaluées à l'aide de deux bases de données de micro-expressions spontanées.

Abstract Micro-expressions are brief and subtle facial expressions that go on and off the face in a fraction of a second. This kind of facial expressions usually occurs in high stake situations and is considered to reflect a humans real intent. They have been studied to better understand non-verbal communications and in medical applications where is almost impossible to engage in a conversation or try to read the facial emotions or body language of a patient. There has been some interest works in micro-expression analysis, however, a great majority of these methods are based on classically established computer vision methods such as local binary patterns, histogram of gradients and optical flow. Considering the fact that this area of research is relatively new, much contributions remains to be made. In this thesis, we present a novel methodology for subtle motion and micro-expression analysis. We propose to use the Riesz pyramid, a multi-scale steerable Hilbert transformer which has been used for 2-D phase representation and video amplification, as the basis for our methodology. For the general subtle motion analysis step, we transform an image sequence with the Riesz pyramid, extract and filter the image phase variations as proxies for motion. Furthermore, we isolate regions of interest where subtle motion might take place and mask noisy areas by thresholding the local amplitude. The total sequence is transformed into a 1D signal which is used for temporal analysis and subtle motion spotting. We create our own database of subtle motion sequences to test our method. For the micro-expression spotting step, we adapt the previous method to process some facial regions of interest. We also develop a heuristic method to detect facial micro-events that separates real micro-expressions from eye blinkings and subtle eye movements. For the micro-expression classification step, we exploit the dominant orientation constancy from the Riesz transform to average the micro-expression sequence into an image pair. Based on that, we introduce the Mean Oriented Riesz Feature descriptor. The accuracy of our methods are tested in two spontaneous micro-expressions databases. Furthermore, we analyse the parameter variations and their effect in our results.