Centre d'Etudes Doctorales : Sciences et Techniques de l'Ingénieur

N° d'ordre 48 */2020*

THESE DE DOCTORAT

Présentée par

Mme : Latifa Greche

Spécialité : Traitement d'images et informatique

Sujet de la thèse : Extraction de caractéristiques et apprentissage automatique pour la reconnaissance d'émotions humaines.

Thèse présentée et soutenue le 15/10/2020 devant le jury composé de :

Nom Prénom	Titre	Etablissement	
MAJDA AICHA	PES	Faculté des Sciences et Techniques -Fès	Présidente
OUM-EL-KHEIR AKTOUF	PH	Institut Polytechnique-Grenoble	Rapporteur
ANASS MANSOURI	PES	Ecole Nationale des Sciences Appliquées-Fès	Rapporteur
KHALID CHOUGDALI	PH	Ecole Nationale des Sciences Appliquées -Kénitra	Rapporteur
AKIL MOHAMED	Professeur émérite	Ecole Supérieure d'Ingénieurs en Electrotechnique et Electronique- Paris	Examinateur
ALI AHAITOUF	PES	Faculté des Sciences et Techniques -Fès	Examinateur
HAMID TAIRI	PES	Faculté des Sciences Dhar El Mehraz-Fès	Examinateur
NAJIA ES-SBAI	PES	Faculté des Sciences et Techniques -Fès	Directrice de thèse

Laboratoire d'accueil : Laboratoire Systèmes Intelligents, Géoressources et Energies Renouvelables (SIGER)

Etablissement : Faculté des Sciences et Techniques de Fès

Remerciements

Cet instant marque pour moi la fin d'une expérience exceptionnelle que j'ai pu confronter grâces à Dieu le tout-puissant, qui m'a guidé pour tracer mon chemin vers le succès et m'a donné le courage pour accomplir mon projet de thèse durant ces longues années d'études.

La réalisation de cette thèse n'aurait tout simplement pas été possible sans le soutien de Mme Najia Es-Sbai, Professeur à l'Université Sidi Mohamed Ben Abdellah, mon encadrante. J'adresse mes remerciements les plus chaleureux à elle, car elle m'a encadrée pendant ces années dans une atmosphère familiale, en accordant des encouragements à continuer pendant les moments difficiles. Je la remercie pour la patience, la confiance et le temps qu'elle m'a accordés, et pour les conseils qu'elle m'a prodigués, et aussi de m'avoir donnée l'opportunité rencontrer M. Mohammed Akil, professeur émérite de l'école ESIEE Paris, qui a donné des impulsions à ma thèse.

J'exprime toute ma gratitude à M. Mohammed Akil de m'avoir aidée en permanence à mener à bien mes travaux de recherche en accordant des remarques pertinentes et des conseils aussi bien morales que scientifiques, et surtout pour m'avoir invitée au laboratoire informatique Gaspar Monge à l'école ESIEE Paris pour faire avancer mes travaux recherche.

J'apprécie profondément les discussions stimulantes lors des réunions avec Rostom Kachouri, Professeur à l'école ESIEE Paris. Je suis reconnaissante à lui parce qu'il n'a pas hésité de contribuer à l'amélioration de ce travail en donnant des remarques et des conseils pertinentes.

Je tiens à remercier Mme Aktouf Oum-El-Kheir, M. Mansouri Anass et M. Chougdali Khalid qui m'ont fait l'honneur d'être rapporteur de mon manuscrit de thèse.

J'exprime ma reconnaissance aux membres du jury M. Ahaitouf Ali, M. Akil Mohammed, Mme Aicha Majda et M. Tairi Hamid qui ont accepté d'examiner cette thèse.

J'exprime ma reconnaissance au directeur M. Ali Ahaitouf et les membres du laboratoire SIGER, pour la liberté d'expression et le support accordés aux thésards(es) dans ce laboratoire. Sans oublier aussi les invitations déjeuner qui s'organisent chaque année dans l'objectif de rapprocher les thésards(es) des Professeurs. Je souhaite plus de succès et de rayonnement au laboratoire SIGER.

Un grand merci à monsieur le doyen Mustapha IJJAALI et le vice doyen, chargé de la recherche scientifique et de la coopération, M. El Mestafa EL HADRAMI pour tous les efforts déployés pour la promotion de la recherche scientifiques à la Faculté des Sciences et Techniques de Fès et les facilitations administratives qu'ils offrent aux thésards(es) surtout lorsqu'il s'agit d'une coopération ou une mobilité nationale ou internationale.

Je remercie mes parents, grâce à qui j'ai eu la chance de pouvoir continuer mes études jusqu'au plus haut stade. Merci pour votre amour inconditionnel et votre grand soutien

financier. Mille pages ne suffiront pas pour vous remercier

Je souhaite exprimer toute ma reconnaissance à mes sœurs qui m'ont soutenue tout au long de ces années et m'ont toujours encouragées à faire ce que je souhaite et à donner le meilleur de moi-même.

J'associe à ces remerciements mon oncle Hassane et sa femme Hakima qui m'ont ouvert non seulement la porte de leur maison, mais aussi leur cœur.

Un merci spécial à Salima, la fille la plus gentille du monde, et ses parents magnifiques, pour les aides et les moments familiaux que j'ai vécus avec eux.

Je voudrais aussi exprimer mon amitié aux personnes avec qui j'ai eu le plaisir de partager ses années de thèse Rajaa, Sarah, Sara, Zineb, Leila, Zahra, Iman, Hajar, oumaima, ouissal, Mahmoud, Hisham, Youssef, Omar, Ahamd, Abelhak, abdsamad, hamid et abdelaziz.

À mon père, ma mère et mes sœurs

Abstract

Automatic facial expression recognition has become a crucial technology in the computer vision field and its applications including identification and security, Medicine, and Monitoring. This research provides solutions to build an algorithmic pipeline enabling the recognition of universal facial expressions, as the neutral expression, disgust, fear, happiness, sadness, anger and surprise, in real world videos containing multiple faces and complex background. In order to do this, we propose an image analysis tool to analyse facial images applying a thorough study of a pre-selected feature extraction and classification methods. We investigated descriptors of face shape, texture, and contour in order to extract vectors of features describing the images and store them in a table. This helps analysing data when applying various classification methods, namely the support vector machine, the K nearest neighbours, the naïve Bayes, the linear discriminant analysis and the binary tree method. However, tuning descriptors and classifiers parameters is not an easy process to do it manually. To avoid the time-consuming step of manual configuration of descriptors and classifiers, we propose an algorithm covering different processing steps towards a fully automated analysis of images.

Considering the analysis result of the images, we propose a new pipeline to recognize the universal facial expressions of more than one person in real-world sequence videos. The pipeline uses the optimal analysis model returned by the data analysis tool, which is based on a Histogram of Oriented Gradient (HOG) descriptor and a Linear Discriminant Analysis (LDA). For further improvement of the LDA classifier, deep analysis on the data structure was carried out exploiting the pipeline. This helps testing linear and non-linear decision boundaries on the LDA clusters and setting up the adequate decision boundary while training the classifier.

To conduct our experimentation, we used Cohn-Kanade (CK+) database. An automatic evaluation over time is proposed, where labelled videos are utilized to investigate the suitability of the pipeline in real world condition. The pipeline results showed that the use of HOG descriptor and the LDA gives high recognition rate by an average of 94.66%. It should be noted that the proposed pipeline achieves an average processing time of 0.018 second, without requiring any device that speeds up the image processing.

Résumé

La reconnaissance automatique des expressions faciales est devenue une technologie cruciale dans le domaine de la vision par ordinateur et ses applications, notamment l'identification et la sécurité, la médecine et la surveillance. Le travail de cette thèse concerne le développement d'un système de reconnaissance des expressions faciales universelles, comme l'expression neutre, le dégoût, la peur, la joie, la tristesse, la colère et la surprise, dans des vidéos du monde réel contenant des visages. Pour y arriver, nous proposons un algorithme pour analyser les images en utilisant des descripteurs d'image et des classifieurs. Cependant, l'ajustement de ces méthodes n'est pas un processus facile à effectuer manuellement. Alors pour éviter l'étape de configuration manuelle des descripteurs et des classifieurs, nous proposons un algorithme exécutant les différentes étapes de la chaîne de traitement vers une analyse entièrement automatisée des images.

Ensuite, en se basant sur le résultat d'analyse des images, nous proposons un nouveau pipeline pour reconnaître les expressions universelles de plus d'une personne dans des vidéos du monde réel. Le pipeline utilise le modèle d'analyse optimal, renvoyé par l'algorithme d'analyse de données, qui se compose du descripteur HOG et du classifieur LDA. Pour améliorer le classifieur, une analyse de la structure des données a été réalisée en exploitant le pipeline. Cela, en testant et évaluant des fonctions séparatrices linéaires et non-linéaires sur les clusters de données pendant l'entraînement du classifieur.

Pour mener notre expérimentation, nous avons utilisé la base de données CK+. Une évaluation automatisée correspondant à la cadence vidéo est proposée, où des vidéos étiquetées de la base de données MMI sont utilisées pour étudier la pertinence du pipeline dans des conditions réelles. Les résultats du pipeline ont montré que l'utilisation du descripteur HOG et l'algorithme d'apprentissage LDA donne un taux de reconnaissance élevé de 94,66 % en moyenne. Il convient de noter que le pipeline proposé atteint un temps de traitement moyen de 0,018 secondes, sans nécessiter aucun dispositif qui accélère le traitement de l'image.

ملخص

عمومًا، يتضمن الاعتراف عدة خطوات على النحو التالي : استحصال الصورة، المعالجة المبدئية، تقطيع الصورة، استخلاص الميزات والتصنيف. أصبح التعرف التلقائي على تعبيرات الوجه تقنية مهمة في مجال رؤية الكمبيوتر وتطبيقاته وفي ذلك تحديد هوية الأفراد، الطب والمراقبة. يتطلب نظام التعرف على تعبيرات الوجه خوارزمية تتضمن خطوتين رئيسيتين : استخلاص ميزات الوجه وتصنيف التعبير. يعد تصميم نظام التعرف التلقائي على تعبيرات الوجه أمرًا صعبًا نظرًا للعدد الهائل من الصور اللازمة للاختبار واستخدامه في التطبيقات الواقعية. حتما يمكن أن تؤدي جلسة تجريبية واسعة إلى خوارزمية ملائمة، لا سيما لتحديد أفضل الطرق لاستخراج وتصنيف الميزات لتحقيق التعرف على تعبيرات الوجه بدقة عالية.

يوفر هذا البحث حلولًا لبناء خوارزمية تستطيع التعرف على تعبيرات الوجه، مثل التعبير المحايد، الاشمئزاز، الخوف، السعادة، الحزن والمفاجأة، في مقاطع الفيديو الواقعية التي تحتوي على وجوه متعددة وخلفية معقدة.

في المساهمة الأولى لهذه الأطروحة، نقترح أداة تحليل الصور لتحليل صور تعبيرات الوجه بتطبيق دراسة شاملة لطرق استخراج الميزات وتصنيف تعبير الوجه. في الخطوة الأولى، لاستخراج ناقلات الميزات التي تصف تعابير الوجه السبعة وتخزينها في جدول، قمنا بدراسة طرق استخراج شكل الوجه والملمس والحواف. في الخطوة الثانية، يتم تحليل الجداول التي تم الحصول عليها

باستخدام خمس طرق تصنيف مختلفة، وهي : مصنف شبكات دعم التمييز، مصنف بايز الساذج، مصنف أقرب جار، مصنف شجرة القرار ومصنف تحليل التمييز الخطي. لكل طريقة تصنيف صيغ رياضية خاصة بها والتي ستساعد في إنشاء مصنفات مختلفة. لكن ضبط معلمات طرق استخراج الميزات والتصنيف ليست عملية سهلة للقيام بذلك يدويا نضرا لأنها تستغرق وقتًا طويلاً. لتجنب دالك، نقترح خوارزمية تغطي خطوات المعالجة المختلفة نحو تحليل مؤتمت بالكامل لصور التعبير.

في المساهمة الثانية لهذا العمل، نقترح نضام جديدة للتعرف على تعابير الوجه لأكثر من شخص واحد في مقاطع الفيديو العالم الحقيقي. أولاً، يقوم نظام التعرف بادخال فيديو ويقوم بخطوات كشف الوجه وتتبعه. ثانيًا، يتم تقطيع مناطق الاهتمام مثل العينين والفم من الوجه المكتشف من أجل تقليل كمية الميزات التي سيتم استخراجها. ثالثًا، لتحديد التعابير، يستخدم نظام التعرف نموذج التحليل الأمثل الذي تم إيجاده بواسطة أداة تحليل البيانات، والذي يعتمد على طريقة واصف شكل الوجه ومصنف التحليل الخطي للتمييز. تتمثل ميزة مصنف تحليل التمييز الخطي في أنها تعمل على تخفيض عدد الميزات الناتجة عن واصف شكل الوجه وتصنيف التعبيرات معا. هذا ساعدنا على إجراء تحليل عميق على بنية البيانات من خلال استغلال نظام التعرف على تعابير الوجه لإظهار توزيع فئات البيانات واختبار طرق فصل خطية وغير خطية لفصل فئات التعبير خلال تدريب المصنف.

لإجراء تجربتنا، اقتراحنا تقييما تلقائيا يعمل في الوقت الحقيقي على إيقاع الفيديو، حيث يتم استخدام مقاطع فيديو لتقييم نضام التعرف على تعابير الوجه في مقاطع فيديو العالم الحقيقي. أظهرت نتائج تقييم النظام أن استخدام واصف شكل الوجه ومصنف التحليل الخطي للتمييز يعطي معدل اعتراف مرتفع يبلغ ٦٦.٩٤ في المئة. وتجدر الإشارة إلى أن خط الأنابيب المقترح يحقق وقت معالجة يبلغ متوسطه ٠.١٨٠٠ ثانية، دون الحاجة إلى أي جهاز يسرع معالجة الصورة.

Table des matières

Table des figures

Liste des tableaux

Nomenclature

Liste des abréviations

LDA	Linear Discriminant Analysis, page vii
AB	Arbre binaire, page 69
Adaboost	Adabtative Boosting, page 24
AR face	Aleix and Robert face database, page 18
CAFE	California Facial Expressions, page 18
CCD	Charge Coupled Device, page 2
CK	Cohn-Kanade, page vii
CLBP	Compound Local Binary Pattern, page 31
CMOS	Complementary Metal Oxide Semiconductor, page 2
DaFEx	Database of Facial Expression, page 17
DTP	Directinal Ternary Pattern, page 30
fps	frame per second, page 2
HOG	Histogram of Oriented Gradient, page vii
JACFEE	Japanese and Caucasian Facial Expression of Emotion, page 19
k-ppv	k-plus proches voisins, page 68
KDEF	The Karolinska Directed Emotional Faces, page 18

KFDB	Korean Face Database, page 18
LBP	Local Binary Pattern, page 4
MSFDE	Montreal Set of Facial Displays of Emotion , page 17
NB	Naïve Bayésienne ou encore Naïf de Bayes, page 39
PCA	principal Component Analysis, page 37
PFA	Pictures of Facial Affect, page 18
RVB	Rouge Vert Bleu, page 23
SIFT	scale Invariant Feature Transform, page 32
SVM	La méthode du Séparateur à Vaste Marge, page 62
TFEID	Taiwanese Facial Expression Database, page 17
TSV	Teinte Saturation Valeur, page 23
UTVD	University of Texas Video Database, page 19

Notation des éléments en jeu dans l'apprentissage

(x, y)	Les coordonnées des pixel, page 45
$I_(x, y)$	L'intensité d'image $I \in \Re^2$, page 45
χ	L'espace de representation des exemples $X \in \Re^P$, page 54
Ω	Les données d'apprentissage $\Omega \in \Re^{N \times P}$, page 52
C	Le Vecteur des étiquettes des exemples $C = \{y_1, `\ldots, y_N\} \in \Re^N$ pour $y \in \{1, 2, ..., 7\}$, page 52
N	Le nombre total des images d'apprentissage, page 52
P	Le nombre total des attributs ou encore des caractéristiques extraites par le descripteur d'image. Ce nombre varie delon le descripteur d'image, page 52
X	Un exemple dans la base d'apprentissage ou encore le vecteur des caractéristiques extrait de l'image du visage $X \in \Re^P$, page 52

x_j	Le vecteur du j-ième attribut de Ω ou encore caractéristique extraite par le descripteur d'image $x \in \Re^N$, page 52
$x_{i,j}$	Le j-ième attribut du ième exemple , page 52

Notation pour les algorithmes d'aprentissage

Ψ	L'espace d'hypothèses h, page 54
$h_{optimale}$	Hypothèse optimale obtenue par l'algorithm d'apprentissage, page 54
Ω'	les données Ω après réduction de dimensions par l'algorithme de Fisher $\Omega' \in \Re^{N \times 2}$, page 57
b	le biais, page 54
$H(x_j)$	L'entropie du nœud parent, page 69
t	Le seuil de division de données , page 69
W	Le vecteur des poids des attributs $W = \{w_1, \ldots, w_P\} \in \Re^P$, page 53

Notation pour les metriques d'évaluation

$\hat{e}(h(X))$	L'erreur empirique de l'appretissage, page 78
$\xi_{i,i}$	Les occurrences de bonne classification de la classe i, page 80
$a'_{i,j}$	Les occurrences du faux positif, page 80
$a_{i,j}$	Les occurrences du faux négatif, page 80

Introduction générale

En vision par ordinateur, l'extraction de caractéristiques (ou feature extraction en anglais) permet de résoudre le problème de trouver l'ensemble de propriétés visuelles le plus compact et le plus informatif de l'image, afin d'améliorer le stockage et le traitement des données. L'extraction de caractéristiques est une composante du traitement de l'image qui va de pair avec la classification [1]. Ceci parce que la notion du vecteur caractéristique, extrait de l'image, est le moyen le plus courant et le plus pratique de représentation des données pour les problèmes de classification. Grâce à cela, les données peuvent être stockées dans des tableaux, dont les lignes représentent les entrées du classifieur nommées aussi des exemples et les colonnes représentent les caractéristiques visuelles extraites. Chaque caractéristique résulte d'une mesure quantitative appelée aussi un attribut.

Les chercheurs en informatique et statistique, s'intéressant à l'analyse prédictive, unissent leurs efforts pour faire avancer les domaines d'extraction de caractéristiques et de classification. Les progrès réalisés à la fois au niveau de ces deux domaines ont permis de concevoir des systèmes de reconnaissance, capables d'effectuer des tâches qui ne pouvaient pas être effectuées dans le passé. Pour cela, aujourd'hui, on peut par exemple exécuter les instructions d'un système intelligent qui détecte le visage humain [2], le suit dans une vidéo [3], identifie la personne [4] ainsi que son expression [5] pour des raisons de sécurité et de surveillance dans l'automobile et la santé.

En général, le processus de la reconnaissance repose sur deux étapes cruciales d'analyse d'images qui sont : l'étape d'extraction de caractéristiques et l'étape de classification. En plus de ces deux étapes, la construction de tout système de reconnaissance [6–8]dépend de l'utilisation d'une quantité importante d'images, comme par exemple l'utilisation d'une quantité de 168,359 images de taille 320 x 240 afin de sensibiliser le système aux classes d'expression faciales [7]. Par ailleurs, en présence d'une variété de bases de données, il est

naturel que le choix des techniques d'extraction de caractéristiques et de classification dépend du domaine d'application. Ceci parce que les objets [1] dans chaque base d'images se distinguent par des propriétés visuelles qui lui sont propres comme la couleur, la dimension, la forme, les contours, la texture, etc. De ce fait, la reconnaissance d'une catégorie particulière d'objets impose l'utilisation du descripteur d'image approprié à la nature des objets dans la base de données.

Les défis qui se posent au premier stade de construction de tels systèmes concernent le choix, d'une part, de la technique d'extraction de caractéristiques de l'objet à distinguer du reste de l'image et, d'autre part, la technique de classification permettant l'identification de la catégorie à laquelle il appartient. La construction d'un système de reconnaissance de l'expression émotionnelle, objet de cette thèse, est concerné par ces défis [9]. Pour analyser l'expression du visage à partir d'images faciales, des bases d'images sont fournis par des psychologues sous le format des séquences vidéo [10–14] ou sous format d'images [12,15–20] d'un ensemble de participants montrant différentes classes d'expressions émotionnelle, comme les expressions de la joie, la surprise, la peur, la colère, la tristesse, le dégoût et l'état neutre du visage. Dans ce cas, l'analyse d'image exige de détecter le visage en premier. Ensuite de décrire la variation de l'intensité au niveau des pixels délimitant le contour des traits du visage afin de donner une représentation compacte de l'expression. Finalement, construire à partir des exemples de données d'entraînement un classifieur permettant la séparation des classes d'expressions émotionnelle. Tout cela pour que le système de reconnaissance puisse identifier l'expression dans une nouvelle image n'appartenant pas aux données d'entraînement, qui s'utilisent que pendant la phase d'entrainement du classifieur par l'algorithme d'apprentissage..

Outre le défi du choix des techniques d'extraction des caractéristiques et de classification, pour qu'un système de reconnaissance d'expression émotionnelle fonctionne en temps réel, c.-à-d. à la cadence vidéo, cela nécessite d'analyser les images acquises en prenant en compte la contrainte de temps laquelle dépend du réglage de la vitesse de la caméra utilisée, comme illustré dans la figure 0-1. En réalité, la vitesse peut atteindre une cadence de 10^7 fps (frame per second) dans le cas des caméras à capteurs d'image CCD ou CMOS [21, 22], dont la taille de l'image est de 312 x 260 pixels. Cependant, le réglage de la vitesse à l'une des cadences standard suivantes : 24, 30, 60 fps est souvent utilisée dans le cas des caméras à usage quotidien. Citons, à titre d'exemple, les caméras du téléphone mobile et les caméras de

1. Dans le domaine de la vision par ordinateur, les bases d'images contiennent au moins deux objets représentant les images positives et négatives,

surveillance. La vitesse de 24 fps est la limite inférieure pour garantir une animation fluide dans la vidéo [23], or l'identification de l'expression exige un temps d'analyse d'image qui ne dépasse pas une durée de 0,04 secondes. Cela, doit être effectué afin de permettre par la suite d'achever un temps de traitement qui convient à la cadence vidéo au niveau du système reconnaissance d'émotion, quelle que soit la complexité de l'arrière-plan ou le nombre de visages présents dans la scène.

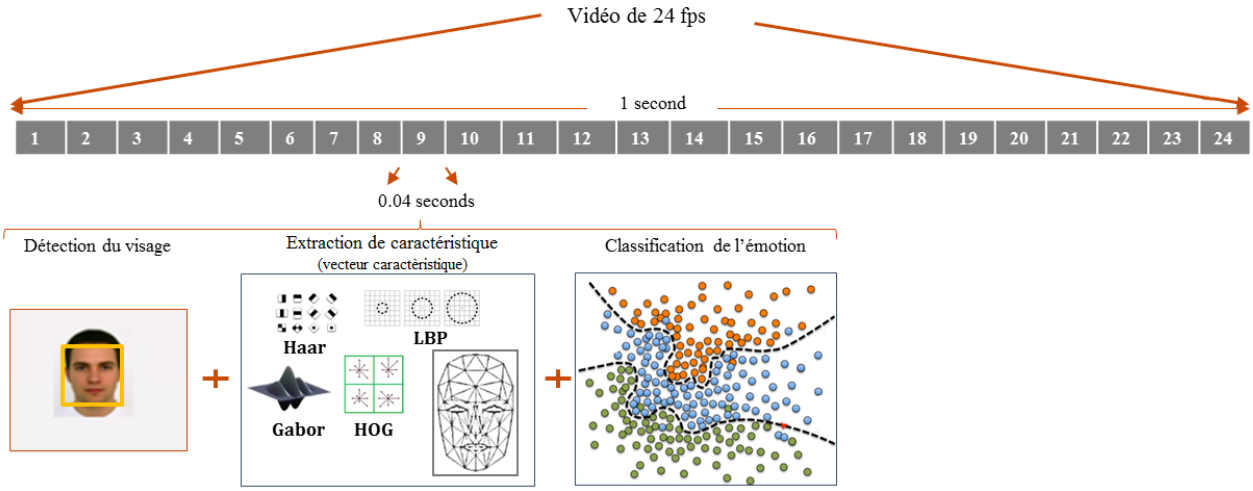


FIGURE 0-1 – Vue d'ensemble des étapes d'un système de reconnaissance d'émotions humaines en temps réel, allant de la détection du visage, en passant par l'extraction du vecteur des caractéristiques décrivant le visage jusqu'à la classification de l'expression émotionnelle.

Au vu des défis mentionnés dans les paragraphes ci-dessus et dans le cadre d'une démarche de construction d'un système de reconnaissance d'émotions des personnes dans des scènes vidéo, nous analysons une base de données comportant un ensemble de 574 images, regroupées en sept classes expressions : la joie, la surprise, la peur, la colère, la tristesse, le dégoût et l'état neutre, par le biais d'une variété de techniques d'analyse d'images. Après la détection du visage sur l'image à l'entrée du système, vient l'étape d'extraction du vecteur des caractéristiques qui décrit l'apparence du visage. En effet, sur l'image du visage de face, on trouve des contours qui délimitent les traits et on s'attend aussi à voir des lignes qui représentent des rides. Les contours sur l'image sont dus à la variation du niveau de l'intensité entre les pixels de la teinte de la peau et les pixels des traits. Comme la courbure délimitant les sourcils, les yeux et la bouche change en fonction de l'expression apparue, il est plus logique de rechercher les orientations du contour des traits du visage que de rechercher leur contour [24]. On peut remarquer ceci par exemple dans les cas des expressions du sourire et

de la surprise. Dans le premier cas, le contour qui délimite la bouche est constitué de courbes peu arquées, alors que dans le deuxième cas, le contour prend la forme de la lettre O.

Dans ce cadre, on peut distinguer deux catégories de descripteurs d'image, une catégorie qui repose sur une analyse spatiale et une autre reposant sur une analyse spectrale. Chaque catégorie permet d'exploiter un certain type d'information. Dans la première catégorie, le descripteur d'image opère dans le domaine spatial où l'analyse du visage se fait directement sur le plan des valeurs des pixels de l'image numérique. On peut procéder au traitement en divisant l'image en une grille de cellule de petite taille, par exemple, de 8 x 8 pixels. Ensuite, calculer l'histogramme de la direction du gradient de l'image pour les pixels dans chacune des cellules. En combinant les histogrammes extraits de chaque cellule, on peut construire l'apparence globale du visage qui décrit la distribution des orientations du contour et supprime tout autre détail existant dans le domaine spatial original de l'image. Cette technique est appelée descripteur HOG [25]. On peut aussi utiliser un descripteur de la texture comme le descripteur des motifs locaux binaires LBP (local binary pattern) [26]. Celui-ci permet de comparer le niveau de gris d'un pixel avec le niveau de gris des pixels voisins. En effet, si une zone avec un nombre restreint de pixels est intégralement grise, le contraste est nul. Au contraire, si la différence du niveau de gris entre les pixels voisins est élevée, la valeur du contraste augmente. Avec l'extraction de telle motifs locaux binaires on peut décrire le contraste au niveau du contour du visage tout en calculant la distribution des LBPs dans les cellules de la grille d'image.

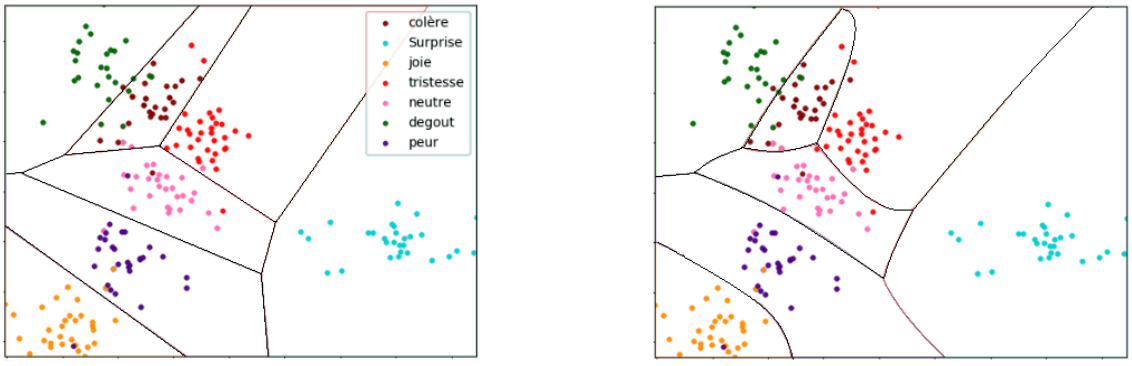
Aussi, une description qui exprime la fréquence spatiale dans l'image s'avère intéressante. C'est le cas lorsque l'on veut extraire l'orientation du contour dans le domaine spectral. L'utilisation de la technique d'ondelette bidimensionnelle de Gabor [27] permet de représenter les lignes régulières sur l'image du visage, lesquelles sont des contours délimitant les régions des pixels où l'intensité varie peu dans l'image spatiale. Le point fort de cette technique est sa capacité d'imiter le cortex visuel primaire chez l'être humain qui a pour rôle l'extraction des particularités géométriques de l'image perçue.

Le choix d'utiliser de tels descripteurs est motivé par divers facteurs. En plus de la capacité d'extraire les orientations du contour et de les présenter sous forme de vecteur des caractéristiques de taille fixe pour toutes les images, ces descripteurs sont tolérants à certaines transformations de l'image. Ces dernières peuvent être géométriques comme la rotation, la translation et le changement d'échelle ou autres comme le changement de luminosité ou de contraste. Quand il s'agit d'extraire la forme et l'apparence du visage, il importe que les

descripteurs d'image soient invariants à ces transformations.

Après l'étape de l'analyse d'images par des descripteurs qui transforment l'ensemble des images en un tableau contenant des vecteurs caractéristiques, vient ensuite l'étape du choix pertinent des méthodes de classification des données d'entraînement. En effet, sans l'analyse des données de tableaux, obtenus via les descripteurs, en faisant usage d'une variété d'algorithmes de classification, il est difficile par exemple de déterminer le classifieur qui peut se généraliser le mieux sur nos données, ainsi que savoir si les classes de données sont facilement séparables dans le classifieur d'expression émotionnelle.

Pour surmonter ces difficultés, une analyse de la structure de données en utilisant plus d'algorithme de classification que possible est essentiel. À ce niveau, notre choix est motivé par le fait que les méthodes de classification devront, d'une part, être supervisées, car nos données contiennent des exemples étiquetés. D'autre part, on veut procéder à la séparation des classes de données en testant une frontière de décision linéaire et non-linéaire, qui peut être polynomial ou quadratique comme illustré dans la figure 0-2. Cela, parce qu'on ne peut pas savoir si les classes de données sont linéairement ou non linéairement séparables qu'après avoir appliqué une variété de frontières de décision, et avoir testé la capacité des méthodes de classification à produire un classifieur d'expression émotionnelle présentant un taux de reconnaissance élevé.



(a) Frontière de séparation linéaire

(b) Frontière de séparation non-linéaire

FIGURE 0-2 – Exemple des frontière de décision appliquer sur le diagramme de dispersion montrant la structure de données d'expressions faciales

Cette thèse s'intéresse à la construction d'un système de reconnaissance d'émotions humaines. Pour ce faire, il nous semble, d'une part, primordial de réaliser une analyse de la base de données des expressions faciales par le biais d'une variété de méthodes d'analyse utilisées

dans la littérature. Cependant, trouver les meilleures valeurs pour tous les paramètres des méthodes utilisées dans l'analyse de données est une tâche fastidieuse, car il peut y avoir plusieurs valeurs à tester et elles peuvent s'influencer mutuellement. Ainsi, automatiser la recherche des paramètres nous fait économiser beaucoup de temps et d'efforts. En proposant un outil d'analyse de données, on peut même généraliser l'idée de rechercher des valeurs optimales des paramètres à l'idée de rechercher le descripteur et le classifieur d'expression qui maximisent le taux de reconnaissance et minimisent le temps de traitement. Par exemple, nous pourrions vouloir essayer une variété de valeurs des paramètres, en recherchant la combinaison descripteur-classifieur qui donne un taux de reconnaissance d'émotion élevé. Pour faciliter cette recherche, notre algorithme d'analyse de données repose sur une procédure de recherche, automatique, de la combinaison descripteur-classifieur optimale.

Organisation de la thèse :

Cette thèse, dont le plan est détaillé ci-après, s'organise en quatre chapitres :

Le premier chapitre présente un bref historique montrant les premières études qui ont permis de faire de la reconnaissance d'émotions humaines, un objet d'études notamment dans le domaine de la psychologie et ensuite dans le domaine de la vision par ordinateur. Ensuite, nous dressons un état de l'art décrivant les techniques de traitement existantes au niveau des phases principales qui forment les systèmes de reconnaissance d'émotions humaines, à savoir : la phase de recherche du visage, La phase de l'extraction du vecteur des caractéristiques du visage et la phase de construction du classifieur d'émotions.

Le deuxième chapitre aborde les notions théoriques des méthodes utilisées pour réaliser l'analyse des jeux de données tout en mettant en exergue les paramètres qui contrôlent leur fonctionnement. Nous décrivons l'ensemble des opérations effectuées par les descripteurs du visage pour créer des tables contenant les données nécessaires pour l'entraînement des algorithmes de classification.

Le troisième chapitre décrit l'algorithme que nous avons proposé pour automatiser la recherche de la valeur des paramètres des méthodes d'analyse permettant le fonctionnement optimal sur les jeux de données. L'algorithme est conçu aussi pour pouvoir comparer les performances des méthodes d'analyse, en termes de taux de reconnaissance et temps de traitement, et pour choisir les meilleurs, c'est-à-dire le meilleur taux de reconnaissance avec le minium de temps requis.

Dans le dernier chapitre, nous proposons un pipeline reposant sur le modèle d'analyse optimale renvoyé par l'outil d'analyse de données introduit dans le deuxième et le troisième chapitre. Le pipeline est exploité aussi pour effectuer une analyse approfondie au niveau du classifieur d'expression émotionnelle pour améliorer le fonctionnement du pipeline. Enfin, le pipeline est validé en l'exécutant et l'évaluant sur des séquences vidéo contenant des scènes prise dans des conditions réelles.

Enfin, nous terminons ce manuscrit par une conclusion générale dans laquelle nous rappelons la problématique relative à notre travail de recherche et résumons les chapitres. Aussi, nous présentons les perspectives émanant des limites dégagées de ce travail de thèse, et ce, en proposant des pistes de recherche, en cours et à venir.

Publications

Publications indexées :

- L. Greche, M. Akil, R. Kachouri, Na. ES-Sbai, A new pipeline for the recognition of universal expressions of multiple faces in a video sequence, Journal Real-Time Image Processing (JRTIP), 1-14 26 Juin 2019, Q2, IF 2.588 Scopus.
- L. Greche, A. Taamouch, M. Akil, Na. ES-Sbai, Tuning image descriptors and classifiers : the case of emotion recognition, accepted for WITS'2020 Conf, To appear in Springer.

Publications IEEE Xplore :

- L. Greche, M. Jazouli, N. Es-Sbai, A. Majda, A. Zarghili, « Comparison Between Euclidean and Manhattan Distance Measure for Facial Expressions Classification» 29 Mai 2017 IEEE Xplore, DOI : 10.1109/WITS.2017.7934618.
- L. Greche, N.Es-Sbai, L. Egons, "Histogram of oriented gradient and multi Layer Feed Forward Neural Network for facial expression identification", IEEEXPLORE, DOI : 10.1109/CADIAG.2017.8075680, 23 oct. 2017.
- L. Greche, N.Es-Sbai, L. Egons "Automatic system for facial expression recognition using Histogram of Oriented Gradient and Normalized Cross Correlation", IEEEXPLORE, 26 May 2016, DOI : 10.1109/IT4OD.2016.7479316, Electronic ISBN : 978-1-4673-7689-1.
- L. Greche, N. Es-Sbai, L. Egons, "Performance review of a multi-layer feed-forward neural network and normalized cross correlation for facial expression identification", 24 Avril 2017 IEEEXPLORE, DOI 10.1109/SITIS.2016.43.

Publication non indexée :

- L. Greche, N. Hamaoui, N.Es-Sbai, "Facial expression recognition on Android",Mediterranean Telecommunication Journal, vol. 5, no.2, pp.87-92, Fez, Morocco, June. 2015.

Communications orales :

- L. Greche, A. Taamouch, M. Akil, Na. ES-Sbai, Tuning image descriptors and classifiers : the case of emotion recognition,onférence internationale Wireless Technologies Embedded and intelligent Systems (WITS'2020), 14-16 October 2020, ENSA-USMBA Fès .
- L. Greche, N. Es-Sbai, "Development of a real time pipeline for face expression recognition", Workshop systèmes embarqués et application 2019, FST-USMBA Fès.
- L. Greche, M. Jazouli, N. Es-Sbai, A. Majda, A. Zarghili, "Comparison Between Euclidean and Manhattan Distance Measure for Facial Expressions Classification" conférence internationale Wireless Technologies Embedded and intelligent Systems (WITS 2017), 19-20 Avril 2017, FST-USMBA Fès
- L. Greche, N.Es-Sbai, "Etudes comparative des systèmes de reconnaissance des émotions", 3ème édition de la journée des doctorants, 2017.
- L. Greche, N.Es-Sbai, L. Egons, "Automatic system for facial expression recognition using Histogram of Oriented Gradient and Normalized Cross Correlation", international conference on Information Technology for Organisations Development (IT4OD), Mars 30– 1 Avril , 2016, ENSA, Fès.
- L. Greche, N.Es-Sbai, "Reconnaissance des expressions basée réseau de neurones", 2ème édition de la journée des doctorants, 2016.
- L. Greche, N. Hamaoui, N. ES-SBAI, "Facial expression recognition on Android", Wireless Technologies Embedded and intelligent Systems (WITS 2015), 29-30 Avril 2015, FST-USMBA Fès.

Communications par poster :

- L. Greche, N. Es-Sbai, “Exploring data with multiples image descriptors and machine learning algorithms”, Workshop systèmes embarqués et applications 2019
- L.Greche, N. Es-Sbai, “facial expression recognition and motion detection ”, Journée du pôle de recherche en technologie de l’information et de communication, systèmes et modélisation, 2015.

Chapitre 1

La reconnaissance des émotions humaines : état de l'art

1.1 Introduction

Les avancées technologiques ont permis aux utilisateurs humains d'interagir avec les ordinateurs. En plus du clavier et de la souris, de nouvelles modalités d'interaction homme ordinateur telles que le geste et la voix font leur apparition. Malgré des avancées importantes, un élément nécessaire à l'interaction naturelle reste encore un sujet de recherche actif : les émotions.

Nous commençons par présenter un bref historique exposant les recherches réalisées sur les expressions émotionnelles dans le domaine psychologique. Nous décrivons ensuite le système de la reconnaissance des émotions selon trois phases. Dans la première phase, nous présentons les techniques de recherche du visage, parmi lesquelles il y a les techniques de détection du visage et de segmentation des régions d'intérêts contenant les traits du visage, ainsi que des techniques de suivi de visage dans le cas de l'utilisation des vidéos. La deuxième phase, nous donnons les techniques d'extraction de caractéristiques utilisées pour la description du visage. Dans la troisième phase, nous exposons les approches de classification qui permettent de construire un classifieur de l'expression émotionnelle.

1.2 la reconnaissance des émotions humaines : dans le domaine de la psychologie

Avant que la communauté de la vision par ordinateur ouvre le débat sur le sujet de la reconnaissance des émotions humaines, de nombreux efforts ont été réalisés par des psychologues pour interpréter et analyser les expressions du visage, plus particulièrement celles représentant les émotions des individus. Ici, la reconnaissance de l'expression du visage ne doit pas être confondue avec la reconnaissance des émotions humaines. Alors que la reconnaissance des expressions du visage concerne la classification des mouvements des traits du visage en classes qui sont purement basées sur des informations visuelles, les émotions humaines sont le résultat des facteurs psychologiques. Contrairement à la reconnaissance des expressions, la reconnaissance des émotions ou encore expression émotionnelle est une tentative d'interprétation et nécessite souvent la compréhension de la situation de l'être humain.

Dans la section 1.2.1 nous mettons le point sur la signification des termes expression et émotion. Ensuite dans la section 1.2.2 nous introduisons un aperçu sur les jeux de données que les chercheurs de la vision par ordinateur utilisent pour contribuer à la construction ou l'amélioration des systèmes de reconnaissance des émotions.

1.2.1 Expression et émotion

Souvent, les individus exposent différentes expressions lorsqu'ils interagissent avec les personnes qui les entourent. Une expression peut indiquer tout mouvement au niveau des traits de visage, comme elle peut indiquer une émotion particulière ressentie chez la personne. À travers les recherches réalisées à propos des émotions chez l'être humain, les psychologues ont pu déterminer quelques émotions relatives à certaines expressions. Cela a pu être déterminé en effectuant une étude interculturelle [28–30] dans laquelle un groupe diversifié d'individus ayant des cultures occidentales et non-occidentales, sont soumis à des expérimentations, parfois, sans qu'ils en soient conscients. Ces expérimentations peuvent se faire de deux façons différentes.

Dans certains papiers [10, 11, 13, 15, 16, 18, 20, 31–34] l'expérimentateur dicte directement l'émotion que le participant doit afficher pour pouvoir prendre des enregistrements vidéo ou des images et les analyser à l'aide d'un groupe de psychologues. Dans d'autres papiers [13, 35, 36] on peut par exemple demander à chaque individu de regarder des vidéos

conçus pour générer différentes émotions. Le moment où chaque individu est sous l'expérimentation un groupe de psychologues l'observent, analysent les expressions et déterminent les différentes émotions qu'ils ont identifiées pour faire un vote à la fin de l'expérimentation. Ensuite, les individus sont invités à voir les enregistrements de leur vidéo dans l'objectif de décrire leurs émotions pendant l'expérimentation. Pour juger les émotions finales à chaque étape de l'expérimentation, certains chercheurs prennent en compte l'agrément des observateurs à propos des notes qu'ils ont prises [37] ou la description des individus eux-mêmes [38] ou les deux à la fois [28,30]. Lorsque les votes vont en majorité pour une expression émotionnelle particulière, celle-ci est dite **émotion universelle** [1], **émotion basique**, ou encore une **émotion prototype**. Dans le tableau 1.1 nous résumons un ensemble d'études réalisées par différents psychologues pour rechercher des émotions universelles.

TABLE 1.1 – Les émotions universelles relatives à chaque psychologue

psychologues	**émotions universelles**
James [39]	Peur, grief, amour, rage
Watson [40]	Peur, amour, rage
Mowrer [41]	souffrance, plaisir
Arnold [42]	Colère, aversion, courage, déjection, désire, désespoir, peur, haine, espoir, amour, tristesse
Robert [43, 44]	Acceptation, colère, anticipation, dégout, joie, peur, tristesse, surprise
Gray [45]	Rage and terreur, anxiété, joie
Panksepp [46]	Attente, peur, rage, panique
Ekman [47]	Colère, dégout, **peur**, joie, **tristesse**, **surprise**
Tomkins [48]	Colère, intérêt, mépris, dégout, détresse, peur, joie, honte, surprise
Weiner [49]	Joie, **tristesse**
Oatley [50]	Colère, dégout, anxiété, joie, tristesse
Frijda [51]	Désire, joie, intérêt, surprise, émerveillement, chagrin
Izard [52]	Colère, mépris, dégout, détresse, peur, culpabilité, intérêt, joie, honte, surprise
McDougall [53]	**Colère**, dégout, allégresse,peur, subjection, tendresse, émerveillement

Malgré les multiples choix des émotions universelles sur le tableau, les chercheurs de la communauté de la vision par ordinateur se limitent souvent à l'utilisation de celles prédéfinies

1. Il convient de noter que tous les chercheurs ne souscrivent pas à la notion d'émotion de base. La notion des émotions de base reste quelque peu controversée, un certain nombre de chercheurs contestant son utilité terminologique ou l'idée qu'un petit ensemble des émotions de base peut être identifié.

comme universelles par Ekman [47] à savoir la joie, la surprise, la tristesse, le dégoût, la colère, la peur et l'expression neutre. Ceci, à cause de la grande disponibilité des jeux de données (Voir section 1.2.2.1) relatives à ce genre d'émotion et le manque de base de données contenant d'autres expressions des émotions comme celle de la rage, du désespoir, etc.

1.2.2 De la psychologie vers la vision par ordinateur : reconnaissance des émotions universelles

Après des années de recherches sur les émotions universelles dans le domaine de la psychologie, une connaissance s'est établie dans ce domaine, ce qui a permis de fournir l'ensemble des informations, comme des règles de codage des mouvements du visage [54] et des bases contenant des enregistrements d'individus affichant des émotions sous différentes conditions [12, 16, 36], utiles pour la construction des systèmes de reconnaissance des émotions. Cette section est partagée en deux parties : dans la première, nous introduisons des bases de données qui ont été fournies gratuitement à la communauté de la vision par ordinateur. Dans la deuxième partie, nous présentons quelques domaines d'application de la reconnaissance des émotions.

1.2.2.1 Les jeux de données

Les enregistrements des émotions sont souvent stockés sous format d'image ou une vidéo montrant l'émotion de la personne filmée selon trois phases (voir figure 1-1). La première est la phase du début (ou onset) qui commence par l'expression neutre puis l'intensité de l'émotion augmente de plus en plus. La deuxième est la phase du sommet (ou apex) qui indique le maintien d'une intensité élevée de l'expression d'émotion pendant quelques secondes. Finalement la phase de l'offset qui est la phase de retour à l'expression neutre.

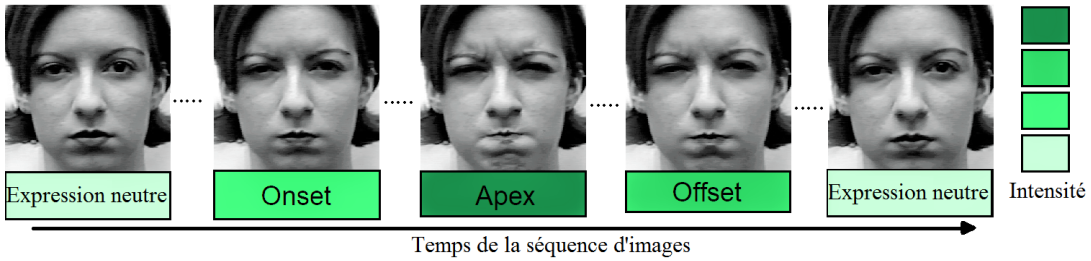


FIGURE 1-1 – Exemple extrait de la référence [55] montrant les trois phases d'affichage de l'émotion dans une séquence d'image de la base de données CK+

Dans la table 1.2 nous présentons les jeux de données fournies à la communauté de la vision par ordinateur tout en mentionnant les particularités de chaque base de données. Par exemple, certains chercheurs [37] tendent à enregistrer les données en éloignant les participants des conditions de laboratoire dans l'objectif de les laisser exprimer leurs émotions naturellement. Ceci permet de donner des émotions plus au moins spontanées. D'autres chercheurs créent des bases de données tout en y ajoutant des contraintes telle que : la variation d'éclairage de la chambre [18,20,56] pendant l'expérimentation ; l'ajout des occultations partielles sur le visage [16–18,20,33,56] par : des lunettes, un pivotement du visage, les cheveux ou la barbe ; la diversification de l'origine des individus [10,31,57] en incluant des Africains, des Européens et des Asiatiques.

TABLE 1.2 – Quelques bases de données contenants différentes classes des émotions

	Émotions	**Particularité de la base de données**
MMI [13]	joy, surprise, sadness, disgust, anger and fear	Il contient 2900 enregistrements vidéo du modèle temporel complet de l'émotion, commençant du visage neutre, passant à travers une série de phases : début, sommet et offset.
DaFEx [11]	happiness, surprise, fear, sadness, anger and disgust	Comprend 1008 vidéos d'expressions produites par 8 individus italiens. Chaque émotion a été enregistrée en trois intensités (faible, moyenne et élevée) et dans deux conditions différentes : (1) condition d'énonciation dans laquelle les acteurs ont prononcé des phrases supplémentaires et (2) condition de non-énonciation.
MSFDE [31]	happiness, sadness, anger, fear, disgust, and embarrassment	contient des images de 20 participants européens, asiatiques et africains.
Yale Face [16]	happiness, sadness, sleepy, surprise, and wink	contient 165 images en niveaux de gris de 15 individus, chaque individu exprime une émotion sous des conditions différentes : avec ou sans lunettes, différents angles de vue de caméra.
TFEID [33]	anger, contempt, disgust, fear, happiness, sadness surprise	La base de données comprend 7200 images de 40 individus. Les émotions sont affichées en deux intensités haute et basse, ainsi que deux angles de vision 0 et 45.

Maryland Data [10]	happiness, surprise, fear, sadness, anger and disgust	Cette base de données contient deux ensembles de données : (1) L'ensemble de laboratoire, qui comprend 40 participants ayant différentes cultures, origines et apparences. Les participants ont été autorisés à bouger la tête sans passer à la position de profil. De plus, on leur a demandé d'éviter le discours. Chaque vidéo contient de 1 à 3 émotions. (2) enregistrements vidéo de la télévision contenant également de la parole.
CAFE Data [34]	anger, disgust, happy, maudlin (for sad), fear, surprise	La base de données se compose de deux versions normalisées : une corrigée gamma et l'autre un histogramme égalisé pour enlever le contraste de l'image.
PFA [15]	happiness, surprise, fear, sadness, anger, disgust	La base de données comprend 110 images du visage en position frontale et transformées en niveaux de gris. Les émotions diffèrent en intensité dans les images.
CAS-PEAL [20]	smile, frown, surprise, close eyes, open mouth	Cette base de données contient plus de 99000 images de plus de 1000 individus avec des poses et des éclairages variables. Neuf caméras, espacées dans un cercle semi-circulaire horizontal, ont capturé simultanément les images à travers les différentes poses.
KFDB [18]	Blink, anger, surprise, and happiness	La base de données contient des images de 1000 participants coréens. Au total, 7 caméras CCD ont été placées en demi-cercle autour du participant. Trois conditions d'imagerie ont été prises en compte : (1) éclairage : deux couleurs de lumière différentes fluorescentes et incandescentes, et huit directions d'éclairage, (2) expressions : quatre expressions émotionnelles différentes, et (3) poses avec accessoires : les images ont été prises avec et sans casquette et lunettes.
KDEF [17]	happy, angry, afraid, disgusted, sad, surprised	La base de données contient un ensemble de 4900 images couleur prises par 70 individus. Chaque émotion est photographiée sous 5 angles différents.
AR face [56]	smile, anger, scream	Plus de 3000 images couleur de plus de 100 personnes. Les images ont été prises dans différentes conditions d'éclairage et d'occlusion.

CK+ [12]	joy, surprise, sadness, disgust, disgust, anger and fear	Extension de la base de données CMU-Pittsburg en augmentant le nombre de séquences d'images et de participants et en ajoutant également des séquences spontanées de sourires.
JACFEE [57]	anger, contempt, disgust, fear, happiness, sadness, and surprise	La base de données comprend des images 56 personnes différentes des japonais et caucasiens.
UTVD [35]	happiness, sadness, fear, disgust, anger, puzzlement, laughter, surprise, boredom, and disbelief	La base de données contient un total de 284 participants (208 femmes, 76 hommes). Pour enregistrer la dynamique des expressions des émotions des participants, ces derniers sont invités à voir une vidéo de 10 minutes destinée à susciter différentes émotions. Finalement, les 'motions des participants sont enregistrées pour une durée de 5 secondes. Aussi, qu'une vidéo de "regard fixe" de 5 secondes qui ne contient aucun mouvement du visage mais plutôt d'autres mouvements de la tête ou des yeux à enregistrer pour tous les participants.

1.2.2.2 Application de la reconnaissance des émotions

Avec l'émergence de nouvelles technologies et des machines ayant une bonne capacité de calcul, la reconnaissance des émotions par la vision par ordinateur est devenue possible. Elle peut être utilisée dans des applications, comme :

- **les jeux informatiques [58] :** ce domaine peut bénéficier de la reconnaissance d'émotion. Par exemple un jeu qui permet de savoir si le joueur est heureusement surpris, confus ou intrigué et on peut réagir aux états du joueur et changer la trajectoire du jeu vidéo serait certainement plus divertissant qu'un jeu qui ne les connaît pas.
- **L'animation virtuelle [59, 60] :** le visage est considéré comme une source d'information importante pour la création des jeux vidéo et l'animation des avatars (personnes virtuelles). Le suivi du changement de l'expression émotionnelle permet de capter les mouvements utiles pour faire animer le visage des avatars et de les doter d'un caractère humain.
- **La sécurité [61–63] :** la reconnaissance des émotions permet l'amélioration du système biométrique de l'identification des personnes par la détection du visage qui se base sur l'expression neutre pour identifier des personnes. En addition à cela, l'identification des enfants qui subissent une violence à travers l'évaluation comportementale sur la capacité des enfants à apprendre des émotions en interagissant avec des adultes dans différents contextes

sociaux [64].

- **L'apprentissage [65] :** création d'interface interaction homme ordinateur apprenant aux enfants les émotions universelles et évaluant leur capacité d'imiter les émotions affichées par l'ordinateur.
- **La surveillance médicale [66] :**la surveillance des patients dans les unités de soins intensifs pour les signes de douleur, de tristesse, de colère et d'anxiété

Dans la littérature plusieurs chercheurs [5, 67] proposent des approches de construction des systèmes de reconnaissance d'émotion. Cependant, des interrogations se posent vis-à-vis de la robustesse de ces systèmes et leurs capacités à avoir un niveau de reconnaissance comparable à celui des êtres humains. Mais, est ce que les machines peuvent-elles reconnaître les expressions du visage comme les humains ? Aujourd'hui les machines peuvent imiter, suivre et comprendre l'origine de quelques mouvements au niveau des traits du visage, mais, il est encore difficile de donner un sens à l'expression apparue si cette dernière n'appartient pas à l'une des émotions universelles comme la joie, la peur, la tristesse, la surprise, la colère et le dégoût. La diversité culturelle des individus rend la reconnaissance des émotions difficile même pour l'être humain. Dans ces travaux de recherche, nous nous intéressons essentiellement à proposer une solution pour la reconnaissance des expressions universelles qui sont la joie, la peur, la tristesse, la surprise, la colère, le dégoût et l'expression neutre.

1.3 Description générale du système de la reconnaissance des émotions humaines

Aujourd'hui, le progrès dans la vision par ordinateur permet de traiter et analyser les images pour trouver le visage et déterminer son émotion. Dans la littérature, nous avons constaté que la plupart des systèmes de reconnaissance des émotions humaines se développent, généralement, sur trois phases principales. La première phase consiste à **rechercher et localiser la zone du visage** dans l'image acquise. Ensuite, dans la deuxième phase, l'apparence du visage est analysée par des descripteurs afin d'extraire le vecteur des caractéristiques permettant la **description de l'expression** du visage détecté. Finalement, dans la troisième phase le système **reconnait l'expression** apparente sur le visage, et cela, après avoir réalisé un entraînement avec un algorithme d'apprentissage sur la base de données. Suite à cela, nous avons établi la figure 1-2 qui montre la structure adoptée pour le reste du chapitre et qui est divisée en trois sections, dans lesquelles nous présentons les techniques de traitement et

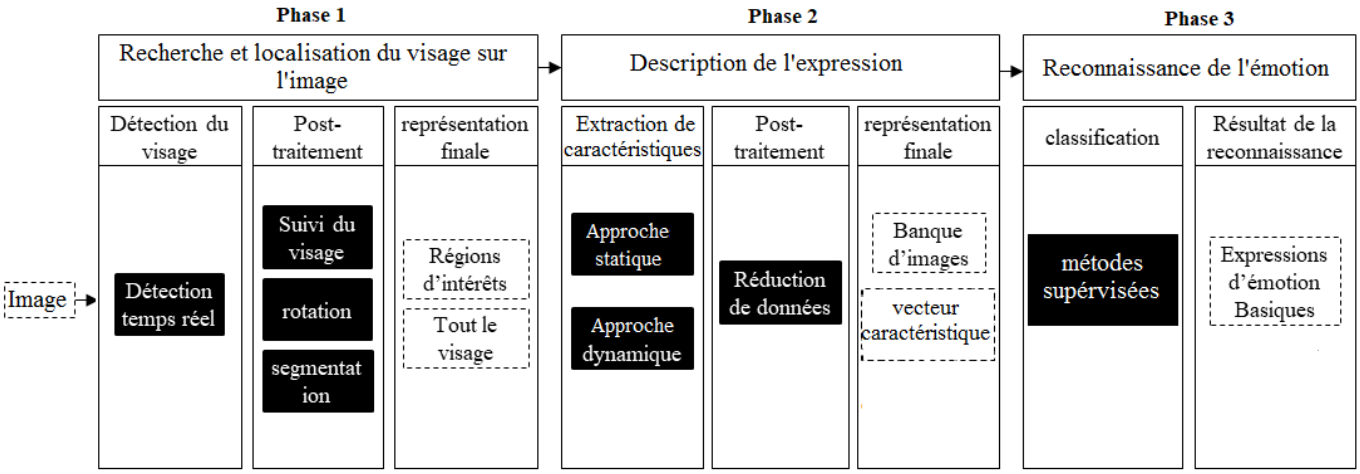


FIGURE 1-2 – Preprésentation générale de la structure des systèmes de reconnaissance qui résume le cadre générique d'analyse de l'expression du visage. Les rectangles noirs représentent les techniques de traitement et celles encadrées en pointillé représentent les données de sorties de chaque phase

d'analyse d'image existantes en les organisant en des sous-phases de la figure 1-2. La structure dans cette figure montre implicitement la piste que nous avons suivie lors du développement de la solution proposée dans le chapitre 4 pour la reconnaissance des émotions universelles dans des vidéos. Bien que nos contributions concernent particulièrement la deuxième et la troisième phase, il est intéressant d'examiner les techniques existantes de détection et de post-traitement du visage, car notre choix des techniques relatives à cette phase est fondé principalement sur la recherche bibliographique menée dans la section 1.4. Dans la section 1.5, nous présentons des catégories de descripteurs permettant la description de l'expression du visage. Ainsi que les méthodes post-traitement relatives à la réduction de dimension de données lorsque les descripteurs extraient une quantité importante d'informations. Finalement, dans la troisième phase, nous fournissons les approches existantes pour la construction des classifieurs des émotions universelles par des méthodes de classification.

1.4 Phase de la recherche du visage sur l'image

La phase de la détection du visage est une étape préalable et nécessaire à tout système de reconnaissance des émotions humaines, car elle permet de trouver la zone du visage sur l'image et de le préparer pour la phase de description de l'expression. Nous divisons cette section en deux parties. Dans la première, nous citons les différentes techniques existantes au

niveau des étapes du processus de construction du détecteur du visage. Dans la deuxième, nous présentons les techniques de post-traitement du visage détecté, à savoir la technique du suivi du visage dans une vidéo, de la rotation du visage et de segmentation des régions d'intérêt comme les yeux, le nez et la bouche.

1.4.1 Les techniques de recherche du visage

Le processus de la recherche du visage repose sur une étape d'apprentissage du détecteur, suivi d'une étape de détection du visage sur l'image. Dans la figure 1-3 nous donnons l'enchaînement des étapes de la recherche du visage ainsi que les différentes techniques existantes dans chaque étape. Dans ce qui suit, nous décrivons d'une façon succincte ces étapes, tout en citant quelques techniques relatives à chacune des étapes.

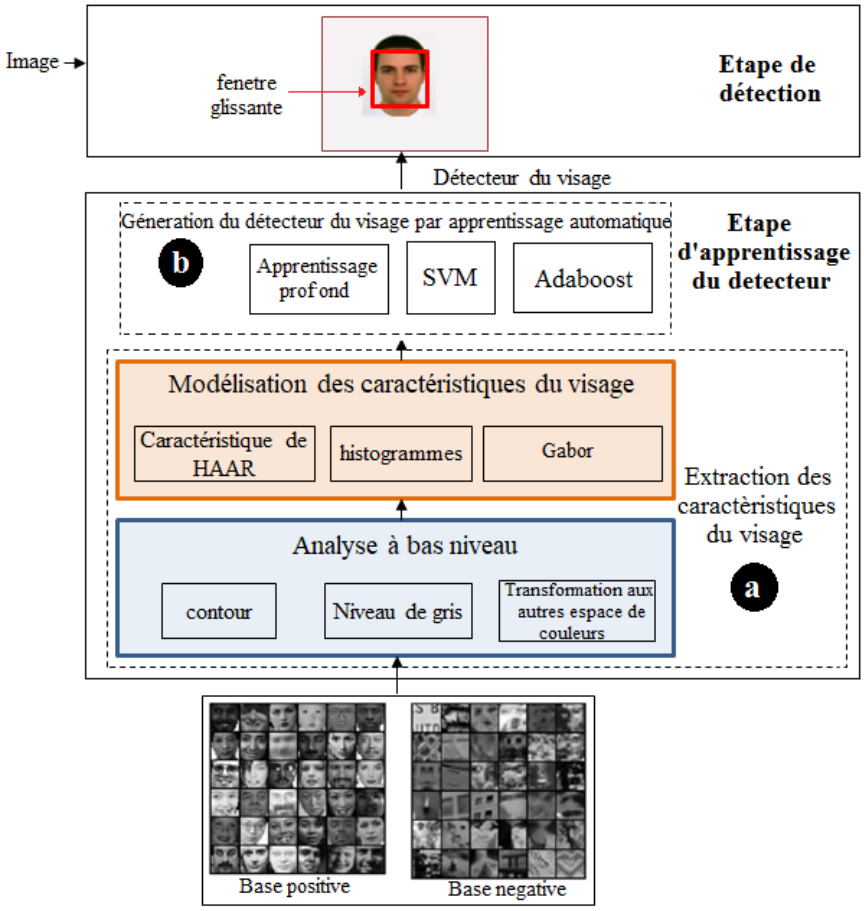


FIGURE 1-3 – Processus de la recherche du visage sur lequel nous montrons les différentes techniques utilisées au niveau des sous-étapes a et b de l'étape d'apprentissage du détecteur.

1.4.1.1 Étape d'apprentissage du détecteur

Cette étape permet au classifieur d'apprendre à différencier entre un visage et un non-visage. Sur le schéma de la figure 1-3, l'apprentissage du détecteur commence par la préparation d'une base d'**images positives** qui contient des images de visage et une autre base d'**images négatives** qui contient des images de non-visage. Ensuite, un ensemble de techniques de traitement d'image peut rentrer en jeu sur deux étapes (Voir étapes a et b de la figure 1-3) :

a. **Extraction des caractéristiques :** au début de l'ère des images numériques, les techniques d'extraction des caractéristiques étaient liées en particulier à des méthodes de traitement d'image de **bas-niveau** permettant essentiellement l'extraction du **contour** [68,69] et/ou la segmentation des régions englobant des pixels ayant des caractéristiques communes. Pour la segmentation des régions d'intérêts, une transformation du mode RVB (Rouge Vert Bleu) de l'image en d'**autres modes de couleur** est exploités pour pouvoir différencier facilement entre les parties du visage. Par exemple, la transformation de l'image RVB en **niveaux de gris** rend les traits du visage tels que les sourcils, les pupilles et les lèvres plus sombres que le reste du visage [68]. Ensuite, avec un seuillage bien précis, on peut segmenter les régions les plus sombres du reste de l'image. Aussi, la conversion de l'espace RVB d'image en d'autres espaces comme l'espace TSV (pour Teinte Saturation Valeur) [70] et YC_rC_b (Y est le signal de luminance et la chrominance C_r qui est Y moins le bleu et la chrominance C_b qui est Y moins le rouge) [71] permettent de segmenter les pixels connexes ayant la couleur de la peau en appliquant un seuillage approprié sur les pixels de l'image transformée. Le point fort de ces méthodes de segmentation réside dans la rapidité de l'extraction des caractéristiques du visage. Cependant, ces méthodes peuvent causer des pertes d'informations à la moindre variation d'illumination, d'occlusion et de la couleur de l'arrière-plan, surtout si la couleur est similaire à celle de la peau. Ces problèmes sont résolus avec l'apparition des descripteurs invariants aux changements d'illumination, comme les descripteurs qui se basent, d'une part, sur la transformation en ondelettes de **Gabor** [72] ou **de Haar** [2] (Voir exemples de caractéristiques de Haar dans la figure1-4), et d'autre part, sur le calcul des **histogrammes** décrivant la variation de l'orientation du contour [73] ou la variation de l'intensité [74] dans des zones locales sur l'image ou les deux à la fois [75].

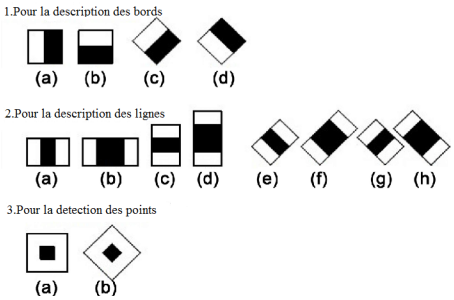


FIGURE 1-4 – Quelques caractéristiques de Haar. Les pixels dans les régions rectangulaires noires sont soustraits des valeurs des pixels dans les régions rectangulaires blanches, le résultat représente la valeur de chacune des caractéristiques [76].

b. **Génération du détecteur du visage :** à ce niveau, différentes méthodes de classification peuvent être exploitées pour construire le classifieur binaire du visage en exploitant les caractéristiques extraites des images positives et négatives. On dit binaire puisqu'on a seulement deux classes : soit le visage existe soit il n'existe pas. Pour la génération du détecteur du visage, l'algorithme de la méthode d'apprentissage ajuste ses paramètres et hyperparamètres en concordance avec les données d'apprentissage. Dans la figure 1-3 , nous mentionnons particulièrement les méthodes de classification qui ont démontré une bonne vitesse de détection dans la littérature. Par exemple, en se basant sur la méthode **SVM**, Kumar et al. [4] indiquent dans leur article avoir détecté le visage sur l'image à une vitesse de 10 fps. Alors que, D. Maio et al. [77] atteignent une vitesse de détection de 13 fps après l'utilisation d'une mesure de similarité [78] pour examiner la ressemblance entre l'image cible et les images candidates. Un progrès notable a été réalisé au sujet de la détection du visage dès l'apparition de la technique de Viola et Jones [6], qui s'est implémentée sur différents appareils tels que les téléphones portables, les tablettes, etc. Ceci grâce aux méthodes de calcul rapides utilisées dans cette technique, comme :

- L'image intégrale : qui accélère le calcul des caractéristiques de Haar [76].
- La cascade de classifieurs **Adaboost** (Adaptive Boosting) [79] : qui accélère la sélection des caractéristiques de Haar relatives au visage.

Actuellement, la communauté travaillant sur le sujet de la détection du visage s'est orientée vers l'utilisation de l'**apprentissage profond**. En effet, l'avancement des recherches sur les réseaux de neurones artificiels [80] au fil du temps, a permis de passer d'une classification basée sur un réseau de neurones classique [81] à une classification

basée sur un réseau de neurones à convolution [82]. Ceci grâce à la capacité de ce dernier à auto-extraire les informations caractéristiques de l'image du visage et de réaliser l'apprentissage en même temps. Par exemple l'étude réalisée par Garcia et al. [82] montre que le réseau de neurones à convolution peut trouver des visages très variables, c.-à-d. pivotés jusqu'à ±20 degrés par rapport au plan de l'image et tournés jusqu'à ±60 degrés, dans des images complexes du monde réel, et ceci sans la mise en œuvre d'une étape préalable pour l'extraction des caractéristiques. Malgré ces avantages, la vitesse de détection du visage par le réseau de neuronnes à convolution, reste faible et presque huit fois plus lente [82] que la méthode de Viola et Jones [2].

1.4.1.2 La phase de la détection du visage

Pour la détermination finale de la zone du visage une **fenêtre glissante** (Voir figure 1-3) contenant le détecteur du visage qui balaye l'image dans le but de chercher des visages dans l'image. Prenons l'exemple de la technique de Viola et Jones. Cette technique utilise une fenêtre glissante, de taille initiale 24x24 pixels et balaye l'image de gauche à droite et de haut en bas pour chercher le visage. Ensuite, la fenêtre augmente sa taille et puis balaye l'image à nouveau afin de rechercher des visages de plus grandes tailles. Au niveau de la fenêtre, plus de 160 000 caractéristiques de Haar sont calculées, dont chacune correspond à la différence entre les sommes de pixels en dessous des deux régions rectangulaires adjacentes en noir et blanc (voir figure 1-5). Ces caractéristiques sont obtenues en considérant différents arrangements de rectangles adjacents de même taille et de même forme. Finalement, pour trouver le visage une cascade de classifieurs calcule la probabilité qu'une caractéristique de Haar soit une caractéristique qui représente un visage.

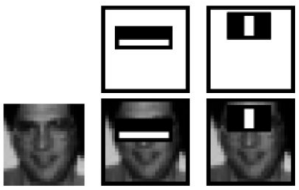


FIGURE 1-5 – Représentation de deux caractéristiques de Haar relatives à un visage [6]

1.4.2 Post-traitement du visage

Après la détection du visage, vient l'étape de préparation du visage pour l'étape de description de son expression. À ce niveau, des études ont proposé des techniques de suivi afin de réduire le temps de détection du visage dans une séquence vidéo, des techniques de rotation pour l'ajustement de visage, et des techniques de segmentation du visage permettant de réduire la région du visage à traiter.

1.4.2.1 Le suivi du visage

Dans un environnement dynamique comme la vidéo, où le bruit, l'illumination et la localisation du visage peuvent changer significativement d'une trame à l'autre, des méthodes de suivi du visage sont nécessaires pour garder la continuité de la détection dans les trames de la vidéo. Un ensemble de méthodes de suivi ont été documentées puis classifiées dans l'étude réalisée par Yilmaz et al. [83]. Cette étude détaille le processus du suivi qui repose, lui aussi, sur une phase d'extraction des informations caractéristiques relatives à l'objet visé, et une phase de calcul de leurs déplacements d'une trame à l'autre.Certaines méthodes reposent sur des techniques rapides d'extraction des caractéristiques comme la détection de la couleur de la peau [84, 85] et la détection des points d'intérêts [3, 86]. Les méthodes qui se basent sur la couleur de peau appliquent, généralement, une transformation de l'image RVB en d'autres espaces de couleur comme l'espace TSV dans l'objectif de segmenter la région des pixels contenant la teinte de peau en appliquant un seuillage approprié sur l'image. Cependant, ces méthodes sont très sensibles aux conditions réelles, telles que le fond ayant une couleur similaire à la peau et les effets d'éclairage [83]. Grâce à l'invariance aux changements d'éclairage, le suivi basé sur l'extraction des points d'intérêts [86] s'est avéré efficace pour une large gamme d'objets y compris le visage. À ce niveau, les détecteurs de points sont utilisés pour détecter une distribution de points sur l'image. Ces points peuvent être des coins, les maxima locaux et les minima locaux localisés autour des traits du visage. Le suivi est garanti si tous les points d'intérêts existent encore dans les trames consécutives de la séquence vidéo, mais au moment où 50% du total des points détectés disparaissent sur une trame cela indique que le visage n'existe pas. Alors, le détecteur du visage scanne à nouveau toute la trame.

1.4.2.2 Rotation du visage

Le visage peut être capturé de face, mais il se peut qu'il soit pivoté d'un certain angle, dans ce cas, la description de l'expression peut être difficile à extraire surtout si le descripteur utilisé

est invariant à la rotation. Certains chercheurs se tournent vers des techniques d'alignement du visage. À titre exemple Carcagni et al. [87] ont proposé de délimiter le visage par une ellipse tout en détectant la couleur de la peau du visage, lorsque les axes de l'ellipse sont tournés par rapport au plan de l'image une rotation devrait être effectuée afin d'aligner l'ellipse dans sa position verticale comme indiqué dans la figure 1-6.

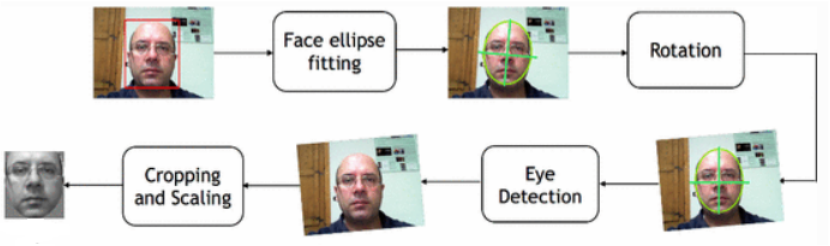


FIGURE 1-6 – Étapes de recadrage et d'alignement du visage par Carcagni et al. [87]

1.4.2.3 Segmentation des régions d'intérêts

Une fois que le visage est détecté sur l'image, le détecteur renvoie les coordonnées et les dimensions de la fenêtre qui englobe le visage. À ce niveau quelques chercheurs utilisent la fenêtre du visage en entier pour effectuer la reconnaissance alors que d'autres préfèrent segmenter le visage en des régions d'intérêt, contenant des traits comme les yeux et la bouche, en vue de réduire la région du visage à traiter. Lekdioui et al. [90] ont appliqué une technique automatique pour segmenter sept régions d'intérêts (Voir figure 1-7d) représentant les composantes faciales impliquées dans l'expression des émotions, plus précisément : le sourcil gauche et droit, l'œil gauche et droit, entre les sourcils, le nez et la bouche. Cette technique repose

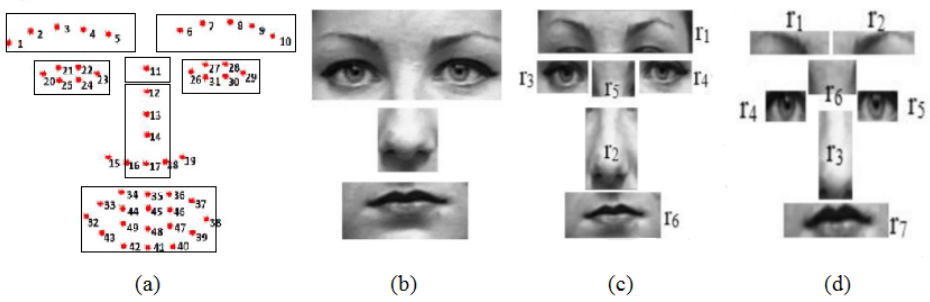


FIGURE 1-7 – Décomposition des régions d'intérêts (a). Les régions d'intérêts étudiées par : Youssif et al. [88](b), Donia et al. [89](c), Lekdioui et al. [90](d)

sur l'algorithme "Intra-Face" [91] qui détecte 49 points autour des régions des sourcils, des yeux, du nez et de la bouche (Voir figure 1-7a). En utilisant les coordonnées des 49 points, on peut créer différentes décompositions faciales pour obtenir les régions d'intérêt,comme, les régions $r_1, r_2, ..., r_7$ dans la figure 1-7d ou les régions $r_1, r_2, ..., r_6$ dans la figure 1-7c. Dans cette optique, Lekdioui et al. [90] ont testé, évalué et comparé trois décompositions du visage en utilisant différentes configurations de paramètres, dont la taille des régions d'intérêt et le nombre des régions d'intérêt, comme le montrent les figures 1-7b, c, et d. Les configurations des régions d'intérêts des figures 1-7b et c sont inspirées des études [88] et [89], respectivement. Après avoir comparé les résultats des trois décompositions, le meilleur résultat a été obtenu dans le cas de la décomposition en sept régions d'intérêt (Voir d dans la figure 1-7) avec un taux de reconnaissance de 92.03%.

1.5 Description du visage : extraction du vecteur caractéristique

Nous commençons ce paragraphe par la question suivante : qu'est-ce qui caractérise une expression ? Lorsqu'un individu est exposé à des situations influençant son état émotionnel, les expressions du visage changent d'un état à un autre. Ce changement, stimulé par les mouvements des muscles, se traduit par le changement de l'apparence des traits du visage. Nous pouvons diviser les changements de l'apparence des traits du visage en deux catégories : la première catégorie regroupe les changements de l'apparence des traits permanents qui se traduisent notamment par tous les mouvements possibles des sourcils, des yeux et de la bouche. Cependant, la deuxième catégorie est liée à l'apparition des traits transitoires (voir figure 1-8) telles que les rides, résultant de mouvements des muscles du visage, qui disparaissent une fois que le visage devient neutre.

FIGURE 1-8 – Types des rides comme 1 : ride du front. 2 : ride du lion. 3 : ride du menton. 4 : patte d'oie. 5 : sillons nasogéniens

À l'instar de la tâche de détection et du suivi du visage, la phase de description de l'expression du visage repose aussi sur une étape d'extraction des caractéristiques visuelles. Des techniques de description de l'expression du visage ont été proposées au fil des années. Les méthodes d'extraction des caractéristiques peuvent opérer d'une façon holistique en traitant le visage dans sa globalité ou d'une façon locale en traitant des zones locales sur le visage qui ne concernent que les éléments présentant le maximum de variations sous l'effet des expressions du visage. En plus de ces deux catégories de méthodes, holistique et locale, les descripteurs de l'expression du visage peuvent être classifiés en une multitude de classes. Comme par exemple, les méthodes statiques [92] ou dynamiques [93], les méthodes de description de la forme [94] ou de la texture [95] et celles opérant dans le domaine spatial [90,96] ou fréquentiel [97,98]. À ces différentes catégories s'ajoute, aussi, la catégorie des méthodes hybrides [99–102]. Sous cette diversité de catégories il reste difficile de distinguer les méthodes adéquates qui permettent une description fine de l'expression du visage, à moins de les comparer expérimentalement. Nous citons dans ce qui suit les méthodes les plus importantes en ce qui concerne les méthodes d'extraction des caractéristiques d'image en les classifiant en classes et sous-classes comme l'illustre l'organigramme de la figure 1-9.

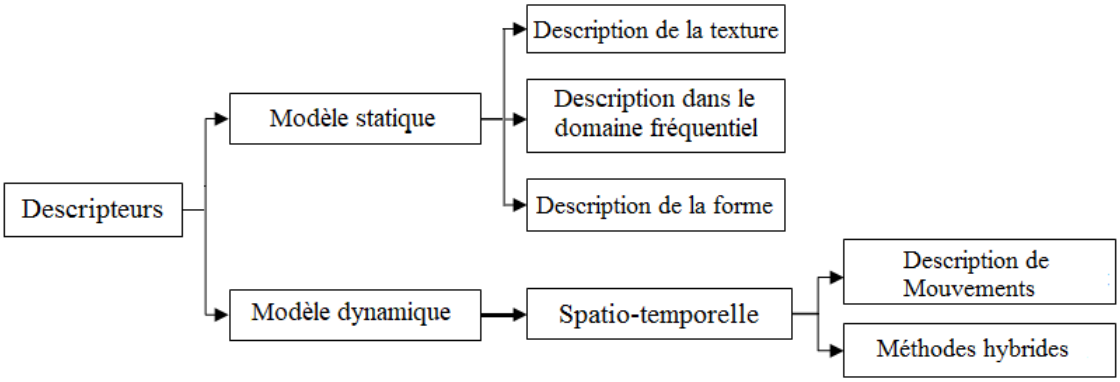


FIGURE 1-9 – Structuration des méthodes d'extraction de l'apparence du visage

1.5.1 Méthodes statiques pour la description du visage

Les méthodes de description statique concernent le traitement d'image 2D indépendamment des changements qui se produisent lorsque l'expression du visage change dans le temps. Nous les classons en trois catégories, à savoir : description de la texture, description fréquentielle et description de la forme du visage.

1.5.1.1 Méthodes de description de la texture du visage

Parmi ces méthodes, on trouve les descripteurs de **la texture** qui appliquent une transformation sur chaque pixel de l'image, et produisent une nouvelle image à partir d'une opération indépendante sur chacun des pixels [103]. Ceci se fait dans le domaine spatial ainsi que dans le domaine fréquentiel. Le descripteur des motifs locaux binaires (Local Binary Pattern LBP) [104], est le descripteur de texture le plus utilisé. Le principe de ce descripteur est de comparer le niveau d'intensité de chaque pixel de l'image avec les niveaux d'intensité des pixels voisins. Ceci se fait par un seuillage binaire des pixels voisins par rapport au pixel central. Ce descripteur se distingue par sa simplicité de calcul et son invariance à la rotation et sa tolérance aux variations d'éclairage (voir figure 1-10) qui vient du fait que le descripteur applique un ajustement de contraste par la technique d'égalisation de l'histogramme.

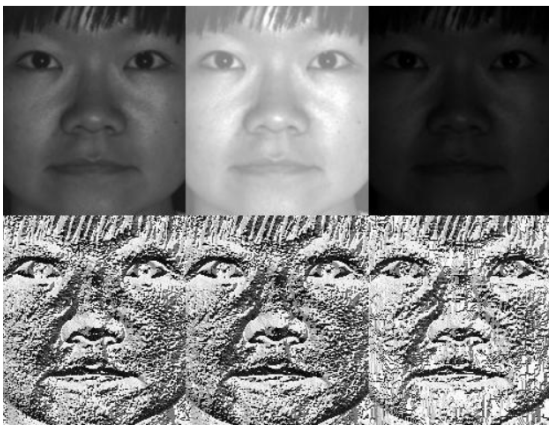


FIGURE 1-10 – L'invariance du descripteur LBP : en haut, l'image subit une modification artificielle de l'éclairage. En bas, nous avons une description LBP de l'image.

Le descripteur LBP a été utilisé pour décrire l'expression du visage, pour la première fois, par Shan et al. [105] et Feng et al. [106]. Ensuite, il a été étudié dans le cas d'images d'expression à basse résolution par Shan et al. [107], où une méthode de sélection a été appliquée à ce descripteur pour pouvoir sélectionner les motifs locaux binaires les plus pertinents pour chaque expression. Des versions modifiées du descripteur LBP ont été proposées pour l'analyse de l'expression, telles que : le descripteur LTP (Local ternary patterns) [108] qui utilise trois valeurs seuils 1, 0 et -1 au lieu de deux valeurs 0 et 1 ce qui le rend plus précis et moins sensible au bruit. D'autres améliorations ont été réalisées sur le descripteur LTP concernant la reconnaissance de l'expression. Par exemple, Ahmed et al. [109] proposent le descripteur DTP (directinal ternary pattern) qui exploite une étape de détection des contours, puisque l'image issue du détecteur des contours est plus descriptive de l'expression que l'image ayant

les valeurs de l'intensité [110]. Une autre version modifiée de LBP est le descripteur CLBP (Compound Local Binary Pattern) [96] qui prend en considération l'information de la magnitude des pixels voisins.

1.5.1.2 Méthode de description dans le domaine fréquentiel

En traitement d'image, l'aspect fréquentiel peut aussi révéler les orientations des contours de l'image. Dans une image, une variation d'intensité élevée sur un petit nombre de pixels indique une fréquence spatiale élevée dans la modélisation fréquentielle de cette image [111]. Par conséquent, les pixels ayant de hautes fréquences spatiales représentent les contours et les bords de l'image [112]. Dans cette optique, certains chercheurs utilisent des filtres comme la transformée de Fourier [113] ou la transformée en ondelettes bidimensionnelles comme l'ondelette de Gabor [114,115], la transformée en cosinus discrète [116], l'ondelette discrète [117] et l'ondelette stationnaire [98, 118]. Malgré l'invariance de ces ondelettes bidimensionnelles devant la translation de l'image et le changement d'illumination et de pose, celles-ci présentent aussi un inconvénient majeur qui est la grande quantité de données produites après avoir appliqué la transformation par une ondelette sur l'image du visage. A titre d'exemple le descripteur de Gabor qui génère une banque d'images composée d'un ensemble d'images de différentes échelles et orientations pour une seule image (Voir figure 1-11).

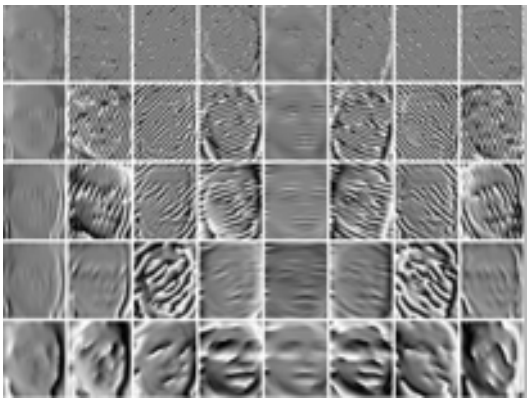


FIGURE 1-11 – Banque d'images générée par l'ondelette bidimensionnelle de Gabor

1.5.1.3 Méthode de description de la forme du visage

Les descripteurs de la forme du visage ont pour rôle de désigner aussi les contours de l'image par le calcul de la distribution de l'intensité au niveau des pixels qui présentent une grande variance d'intensité. Dans le domaine spatial, la forme du visage peut être extraite

par des descripteurs [25, 119] qui utilisent le gradient de l'image dans l'objectif de qualifier la variation de l'intensité au niveau des zones locales dans l'image comme au niveau de toute l'image. Au niveau local, le descripteur SIFT (scale Invariant Feature Transform) a été utilisé par Beretti et al [119] pour détecter des points autour des traits du visage représentant principalement les zones de pixels où la variation de l'intensité est maximale, comme dans le cas des contours et des points d'intérêts représentant les coins et les minima et maxima locaux. Ce descripteur est caractérisé par l'invariance aux changements d'illumination, d'échelle et de rotation. Dalal et al. [25] ont amélioré ce descripteur en calculant les caractéristiques du descripteur SIFT sur une grille dense de cellules uniformément espacées, c.-à-d. en divisant l'image en des blocs de taille égale et en calculant l'histogramme du gradient dans chaque bloc. Cette amélioration a donné le descripteur HOG qui a été utilisé dans un premier temps pour la détection des personnes dans une scène, ensuite utilisé par la communauté travaillant sur la reconnaissance des émotions humaines.

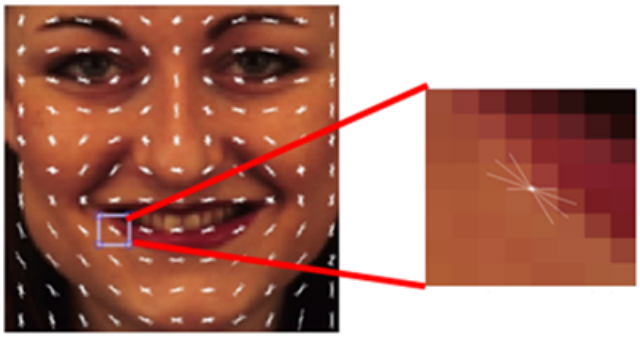


FIGURE 1-12 – Visualisation des caractéristiques de HOG

Le descripteur HOG a été étudié et comparé à de nombreux descripteurs dans certaines études [87, 90, 92, 120–122]. Carcagni et al. [87] ont présenté une étude compréhensive relative au descripteur de HOG, dans laquelle celui-ci a été comparé aux descripteurs LBP et CLBP [123]. L'étude a montré la pertinence du descripteur de HOG. Lekdioui et al. [90] ont comparé le descripteur de HOG à d'autres descripteurs tels que le LBP, le LTP et les combinaisons des descripteurs à savoir HOG-LBP, HOG-LTP et LBP-LTP. L'évaluation de la performance en termes de taux de reconnaissance a montré que l'association des caractéristiques LTP (descripteur de texture) avec celles de HOG (descripteur de forme) améliore le taux de reconnaissance des expressions universelles qui atteint la valeur de 93.34%.

1.5.2 Modèle dynamique de l'apparence du visage

Contrairement aux méthodes statiques, dans les méthodes de description dynamique, le traitement d'images se fait en suivant les changements d'expression dans une séquence d'images, assurant ainsi une modélisation dynamique du mouvement des traits du visage de l'état neutre à l'état expressif. La modélisation dynamique ajoute l'aspect temporel de l'expression du visage au descripteur de l'image statique du visage. La durée de l'expression se caractérise par trois modes [124] : le début appelé aussi onset qui est la durée où l'expression neutre transite à l'expression maximale, le sommet indique la durée du maintien d'une intensité maximale d'expression et la régression appelé aussi offset qui est la durée du retour à l'expression neutre. Pour la modélisation dynamique de l'expression du visage, certaines études reposent sur des méthodes spatio-temporelle [125–131] que nous catégorisons en deux : les méthodes de détection de mouvement des pixels [132, 133], Méthodes des trois plans orthogonaux [126, 127, 134] et les méthodes du suivi des points d'intérêts [91, 135, 136].

1.5.2.1 Le mouvement des pixels

La dynamique de l'expression du visage peut être décrite par la détection du mouvement des pixels du visage dans une séquence vidéo. Différentes méthodes d'extraction de mouvement entre deux images successives ont été proposées, dont la technique du flot optique [10, 137, 138] qui permet de mesurer le déplacement des pixels d'une image à l'autre et la technique de la différence d'images [132, 139]. Lien et al. [137] ont analysé le mouvement des traits du visage à plusieurs échelles à l'aide du flot optique après avoir appliqué une transformation par ondelette discrète sur l'image du visage. Cependant, Otsuka et al. [138] ont estimé le mouvement sur des régions d'intérêts contenant les yeux et la bouche, dont le vecteur des caractéristiques est obtenu en associant la transformée de Fourier 2-D et la technique du flot optique pour extraire le mouvement des contours des régions d'intérêts.

L'avantage de la technique du flot optique, c'est qu'elle peut calculer la vitesse de chaque pixel individuel pour deux images consécutives dans la séquence vidéo, cependant, elle ne peut pas décrire efficacement le cycle complet d'une action sur le visage. Dans cette optique, Fan et al. [133] proposent d'améliorer le flot optique, en le combinant avec la technique d'enregistrement de l'historique du mouvement (MHI- Motion History Image) [140], pour qu'il capte l'historique des mouvements de l'expression du visage dans une séquence vidéo, comme illustré dans la figure 1-13. L'avantage de combiner la technique MHI et le flot optique réside dans le fait que d'une part l'intensité de chaque pixel de l'image obtenu par la technique

FIGURE 1-13 – Séquence d'images allant du visage neutre (à gauche) jusqu'à l'expression de la surprise (au milieu). L'image à droite illustre l'historique du mouvement de l'expression [133]

MHI indique les mouvements récents, on peut voir ceci dans la figure 1-13 dont les pixels en nivau de gris indiquent l'ancien mouvement des traits et, d'autre part, la technique du flot optique calcule l'orientation du mouvement des pixels pour distinguer le mouvement des traits permanents (comme l'ouverture de la bouche et le haussement des sourcils) du mouvement de pose de la tête.

1.5.2.2 Méthodes des trois plans orthogonaux

Une autre famille des méthodes spatio-temporelles repose sur la description de la texture [125–127, 141] et de la forme [134] dans un volume d'images extrait d'une vidéo. L'idée c'est qu'une vidéo composée d'une séquence d'images comprend trois axes orthogonaux x, y pour représenter le plan spatial et t qui indique le temps. La figure 1-14 illustre le principe d'extraction des caractéristiques au niveau des trois plans orthogonaux. En utilisant les trois plans orthogonaux XY, Yt et Xt : le plan XY fournit l'apparence spatiale de l'image et les plans Xt et Yt enregistrent des informations temporelles et de mouvement. Par exemple, en

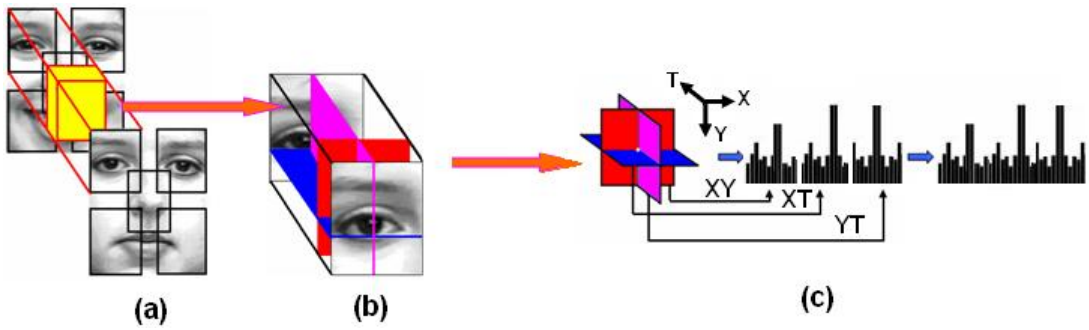


FIGURE 1-14 – Procédure de description des changements d'apparence du visage [125] : le volume du premier bloc en jaune est extrait de la vidéo (a), description de la forme au niveau des trois plans orthogonaux (b), concaténation des résultats (c)

appliquant les méthodes de description statique par le descripteur HOG [125] sur ces trois plans, on peut extraire la variation de l'orientation des contours quand l'expression du visage change.

1.5.2.3 Le suivi des points d'intérêts

Cette approche peut extraire un vecteur des caractéristiques constitué d'un ensemble de points géométriques [136] relevés du contour des traits du visage. En se basant sur ceci, l'algorithme de suivi [135] fournit un modèle d'expression faciale dynamique pour les séquences vidéo représentant les expressions basiques comme indiqué dans la figure 1-15. Ici le rôle de l'algorithme du suivi, c'est de trouver la localisation des points déjà détectés dans l'image suivante de la vidéo, en mesurant le flot optique au voisinage des points.

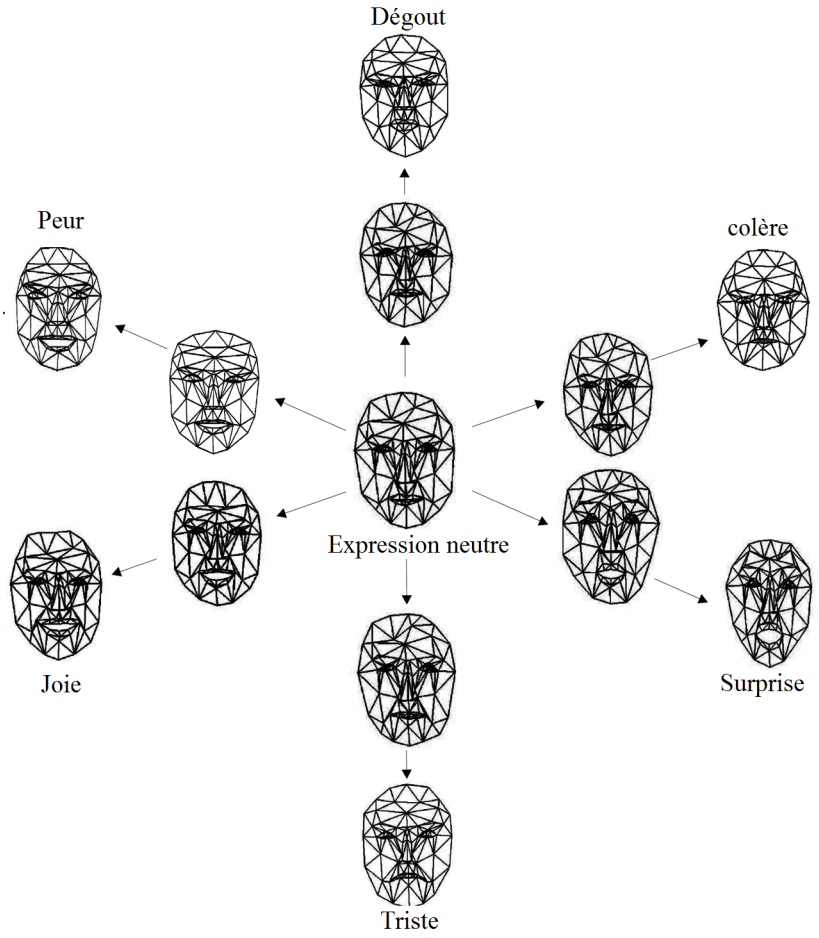


FIGURE 1-15 – Masques de points extraits des séquences vidéo représentant les modèles dynamiques des expressions de la joie, la surprise, la peur, la tristesse, le dégoût et la tristesse

Dans le cadre de l'étude de la dynamique des expressions [94] que nous avons réalisée en collaboration avec le laboratoire des Systèmes Intelligents et Applications, à la FST de Fès, qui s'intéresse à l'étude des émotions chez les enfants atteints de la maladie d'autisme, nous avons construit une base de données enregistrant la dynamique de l'expression d'un groupe de douze personnes. Cela est réalisé en utilisant la caméra Kinect dotée d'un système d'analyse du visage [142] qui permet d'analyser l'image pour détecter et suivre 121 points montrés dans la figure 1-16.

FIGURE 1-16 – Ensemble de 121 points détectés et enregistrés à l'aide de la caméra Kinect

Pour construire la base de données, au lieu d'enregistrer des séquences d'images pour chaque personne, nous avons enregistré des séquences de vecteurs composés des coordonnées de 121 points typiques indiqués dans la figure 1-15. Chaque séquence capte le changement de l'expression, c.-à-d. le déplacement des points, en commençant par l'état neutre puis en augmentant progressivement l'intensité de l'expression d'émotion. Les enregistrements obtenus, qui durent de 5 à 7 secondes, permettent de quantifier les mouvements des points relatifs à chaque expression en déterminant un seuil de déplacement maximal et minimal des points par rapport à l'expression neutre de chaque personne. Les seuils trouvés [94, 143] pour chaque expression constituent le vecteur des caractéristiques que nous utilisons pour entraîner un réseau de neurones. Selon le résultat de l'évaluation du réseau de neurones [94], le taux de reconnaissance d'émotion peut atteindre 100% seulement lorsque les premières images de la séquence contiennent l'expression neutre, puisqu'on se base sur celle-ci pour estimer le seuil du déplacement des points relatifs aux émotions universelles.

Bien que les méthodes dynamiques fournissent, en général, plus d'informations utiles pour l'analyse d'expression, l'aspect dynamique nécessite de considérer l'expression neutre comme référence pour pouvoir prédire les expressions exposées [144]. Ceci limite l'utilisation des modèles dynamiques, car dans le cas des scènes du monde réel l'expression est imprévisible.

1.5.3 Post-traitement : réduction de données

Un vecteur caractéristique ne doit disposer que de l'information offrant une discrimination de l'apparence de l'expression du visage. En plus du fait qu'une technique inappropriée d'extraction des informations caractéristiques conduit à un échec de la classification, un nombre élevé d'attributs dans le vecteur caractéristique décrivant le visage augmente la dimension de données ce qui augmente encore le temps de classification.

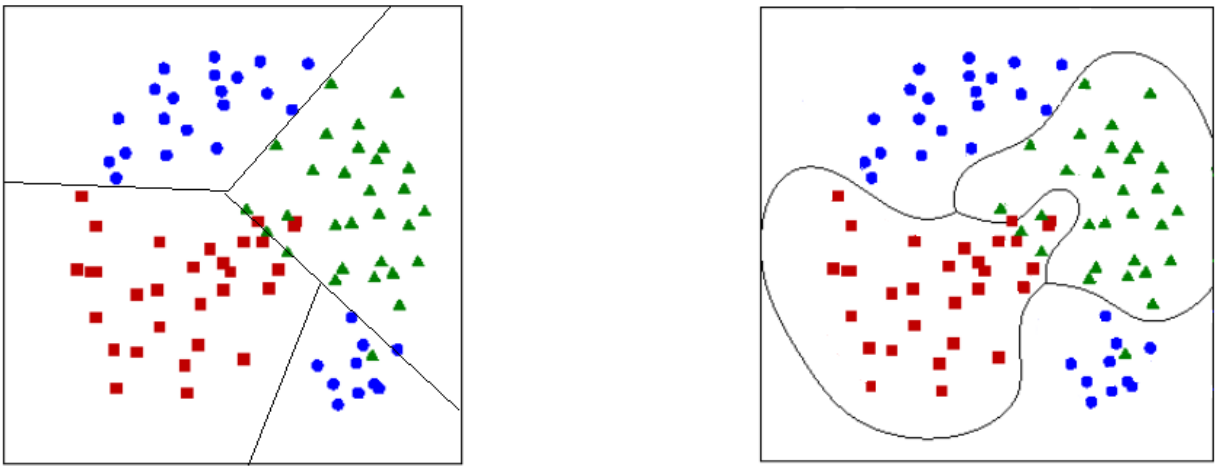
Le problème de données dont le nombre d'attributs élevé peut être résolu par l'utilisation des méthodes de réduction de la dimensionnalité du vecteur caractéristique. Parmi ces méthodes, on trouve la méthode d'analyse en composantes principales (PCA principal component analysis) [145] et l'analyse discriminante [146, 147]. Ces techniques transforment les données en calculant la matrice de covariance dans l'objectif de mesurer la variance des classes de la base de données. En conséquence, ceci peut réduire l'éspace de représentation des données, ayant plusieurs attributs, à des nouvelles données ayant seulement deux attributs. Ces derniers dérivent des anciens attributs à travers le calcul statistique appliqué sur les données (avant la réduction).

La différence entre la méthode PCA et l'analyse discriminante réside dans le fait que durant la réduction des dimensions de données, la méthode PCA se concentre sur la maximisation de la variance entre tous les exemples de données sans prendre en considération leur classe d'appartenance, alors que la méthode d'analyse discriminante minimise la variance entre les exemples appartenant à la même classe et maximise la variance entre ceux de différentes classes.

La méthode Adaboost [79] est une autre technique de réduction des dimensions de données, qui n'applique pas une transformation sur les vecteur des caractéristiques, mais qui y sélectionne les attributs les plus impliqués et utiles pour construire le classifieur de l'expression. Cette technique a été utilisée par Shan et al. [107] pour réduire le vecteur des caractéristiques produit par le descripteur de texture LBP. L'idée de cette méthode est d'associer chaque attribut par un classifieur, ensuite la sélection des attributs se fait en se basant sur l'attribut qui présente un taux d'erreur minimal. Cette méthode est susceptible de dégrader la qualité de l'apprentissage du classifieur de l'expression, surtout si la variance entre les attributs sélectionnés des exemples de même classe est élevée.

1.6 Reconnaissance de l'expression par les méthodes de classification

En arrivant à la phase finale du processus de construction des systèmes de reconnaissance des émotions de la figure 1-2, la reconnaissance de l'expression du visage s'effectue à travers des méthodes de classification. À ce niveau, les algorithmes d'apprentissage sont supervisés car les données d'apprentissage contiennent des classes que l'on cherche à séparer pour construire le classifieur d'expression. Sur le plan conceptuel, on peut imaginer, comme indiqué dans la figure 1-17, un algorithme de classification supervisé qui représente n'importe quelle ligne de séparation, linéaire ou non-linéaire, après le calcul des paramètres de cette ligne.



(a) Séparation linéaire

(b) Séparation non-linéaire

FIGURE 1-17 – Exemples de méthodes d'apprentissage reposant sur une séparation linéaire et non-linéaires de trois classes

Cependant, chaque méthode de classification utilise ses propres formules mathématiques (Ceci est détaillé dans le chapitre 2) pour calculer les paramètres de la ligne de séparation, étant donné les valeurs des attributs de données d'apprentissage. De multiples méthodes de classification ont été utilisées pour analyser et reconnaître l'expression du visage [5, 148, 149]. Nous décomposons ces méthodes en deux approches principales : approche paramétrique et approche non-paramétrique. Nous allons maintenant examiner certaines méthodes paramétriques, puis passer aux méthodes non-paramétriques.

1.6.1 L'approche paramétrique

Les méthodes de classification dans ce type d'approche commencent par une description de données d'apprentissage, tout en faisant une hypothèse sur la forme de la ligne qui va être recherchée pour la séparation des classes. L'approche paramétrique peut supposer que la frontière entre les classes sera une ligne droite dans le cas des données à deux dimensions comme le montre la figure 1-17, un hyperplan dans le cas de données à trois dimensions, ou une sphère dans le cas de données de grandes dimensions. Ensuite, elle peut rechercher les meilleures valeurs pour les paramètres qui décrivent la ligne à la frontière des classes afin qu'elle corresponde le mieux aux données d'apprentissage. Après l'apprentissage du classificateur, celui-ci stocke les paramètres qui décrivent les données et les utilise pour classifier de nouvelles images qui ne sont pas utilisées dans les données d'apprentissage.

Au niveau des méthodes relatives aux approches paramétriques, on trouve la méthode de classification Naïve Bayésienne (BN) [150] et la méthode de Fisher lorsque celle-ci est utilisée pour la classification [151] au lieu d'être utilisée à des fins de réduction de dimension de données [152]. S. M. Lajevardi et al. [101] ont proposé la méthode Naïve Bayésienne pour entraîner un classifieur d'expressions en exploitant des données encodées par le filtre de Gabor et le descripteur LBP. Cette technique de classification suppose que les attributs de données d'apprentissage suivent une distribution gaussienne. Lorsque les paramètres de la distribution gaussienne des classes sont déterminés par l'algorithme d'apprentissage de cette méthode, la classification se fait rapidement, puisque l'on ne retient que la forme des distributions gaussiennes des classes. D'un autre côté, B. Hariharan et al [153] ont également montré que l'utilisation du descripteur HOG conjointement à la méthode d'analyse discriminante permet de construire un système rapide de reconnaissance des émotions universelles non seulement au niveau du temps d'entraînement, mais aussi du temps de classification.

L'avantage des méthodes paramétriques, c'est qu'elles s'entraînent et classifient rapidement. Ceci vient du fait qu'elles gardent les paramètres des lignes de séparation qu'elles trouvent durant l'entraînement afin de les utiliser pour la classification et se débarrassent de toute la quantité de données d'apprentissage. Aussi, elles fonctionnent particulièrement bien lorsque les données ont de nombreux attributs, car dans des espaces de grande dimension, les échantillons sont plus éloignés ce qui facilite la séparation des classes [150].

1.6.2 L'approche non-paramétrique

Cette approche ne fait pas d'hypothèse sur la forme de la ligne séparatrice. Au contraire, elle garde généralement la plupart ou la totalité des données d'apprentissage et cherche une organisation des exemples des classes qui facilitera la classification de nouvelles données à leur arrivée.

Parmi les méthodes de classification qui reposent sur cette approche : la méthode du k-plus proches voisins [154], l'arbre binaire [155] et la méthode SVM [156]. Au niveau de la recherche bibliographique réalisée dans la section 1.5, nous avons remarqué que la méthode d'apprentissage SVM [87,90,157] est la méthode la plus utilisée pour évaluer la performance de reconnaissance des descripteurs d'images voire même les comparer. Cela vient du fait que cette méthode applique une transformation de données d'apprentissage en exploitant la fonction noyau : comme le noyau Gaussien, polynomial ou sigmoïde, permettant ainsi de faciliter la séparation des exemples d'apprentissage appartenant à des classes différentes. Cependant, il n'est pas aussi évident de décider laquelle des fonctions noyaux fonctionne mieux pour la méthode SVM, à moins de les tester ensuite les comparer. De surcroît, on ne peut pas savoir laquelle des autres méthodes d'apprentissage paramétrique ou non-paramétrique peuvent se généraliser mieux sur nos données d'entraînement, à moins de les tester et les comparer.

Le point fort de ces méthodes, c'est qu'elles permettent d'utiliser la plupart ou la totalité des données d'apprentissage à chaque fois que l'on veut classifier des nouvelles données qui n'appartient pas aux données d'apprentissage. Cela peut fournir une grande assistance durant la construction du classifieur, en particulier lorsque les données contiennent du bruit, c.-à-d. lorsque les classes de données sont difficiles à séparer. Cependant, lorsque la quantité de données d'apprentissage augmente le temps d'apprentissage augmente aussi, car généralement l'algorithme parcourt tous les exemples de données d'apprentissage, stockés, pour déterminer la meilleure classification pour un exemple qui n'appartient pas à l'ensemble d'apprentissage.

1.7 Conclusion

Dans ce chapitre, nous avons présenté les techniques de traitement et d'analyse utilisées dans la littérature au niveau des trois principales phases qui composent les systèmes de la reconnaissance des émotions universelles à savoir : la phase de détection du visage, la phase de description et la phase d'apprentissage

Dans la première phase, nous avons introduit quelques techniques de la détection du visage, ainsi que des techniques qui suivent l'étape de la détection, comme les techniques d'ajustement du visage à la position normale lorsque celui-ci se trouve pivoter, les techniques de suivi du visage dans le cas d'une détection dans une vidéo, ainsi que les techniques de segmentation du visage en des régions d'intérêts.

Dans la deuxième phase, nous avons exposé des méthodes de description du visage en les regroupant en des méthodes statiques et autres dynamiques. Les méthodes description statique traitent les pixels par des descripteurs d'image pour générer un vecteur de caractéristiques décrivant les contours, la texture et la forme du visage. L'approche de description dynamique de l'expression traite un volume composé d'une séquence de quelques images. Cela pour pouvoir suivre dans le temps le mouvement des contours qui délimitent les traits du visage. Nous avons vu que la description dynamique offre des informations supplémentaire par rapport aux méthodes statiques, comme la quantité de déplacement des propriétés géométrique décrivant les contours de l'image lors du changement de l'expression. Malgré cela, l'approche de description dynamique de l'expression échoue pendant la classification d'expression lorsque l'expression neutre n'est pas retrouvée dès la première détection du visage. Ceci parce que la dynamique de l'expression doit toujours être comparée à l'état neutre pour pouvoir quantifier l'émotion.

Dans la troisième phase, nous avons regroupé les méthodes de classification, qui permettent de construire un classifieur des émotions universelles, en des méthodes qui se basent sur deux approches différentes. Une approche paramétrique qui construit un classifieur à partir des hypothèses qu'elle fait concernant la forme de la frontière séparation des classes, et une autre approche non-paramétrique qui nécessite l'accès à toutes ou une partie des données d'apprentissage pour classifier de nouveaux exemples. Cela nous permet d'avoir une idée sur les méthodes de classification rapide de ceux qui sont lentes.

Dans le chapitre prochain, nous allons présélectionner quelques descripteurs d'images et méthodes d'apprentissage afin de réaliser une étape d'analyse de données des émotions universelles. Cette démarche est nécessaire pour pouvoir ajuster, évaluer et comparer les techniques pré-sélectionnées et en identifiées celles qui aboutissent à un temps de traitement minimal et un taux de reconnaissance élevé.

Chapitre 2

Analyse des données par des descripteurs du visage et des méthodes de classification

2.1 Introduction

Le progrès dans le domaine de la vision par ordinateur, permet d'obtenir des méthodes d'analyse du visage humain par la description de sa forme, sa texture et ses contours, pour obtenir un vecteur de caractéristiques stockable dans un tableau contenant d'autres exemples de vecteurs caractéristiques catégorisés selon les classes d'émotion. La notion du vecteur de caractéristiques permet d'organiser les données de façon à ce qu'elles puissent être utilisées dans l'apprentissage du classifieur.

En effet, la construction d'un système de reconnaissance d'émotions fonctionnant à la cadence vidéo 24 fps est, souvent, confrontée à deux difficultés majeures :

1. Comment décider quelle méthode de description du visage peut aboutir à une description optimale des propriétés géométriques impliquées dans l'expression du visage ?
2. Comment aussi décider quelle méthode de classification permet de construire un classifieur optimal pouvant être généralisé sur les données d'apprentissage et reconnaître avec précision les expressions du visage dans des images n'appartenant pas à la base d'apprentissage ?

La réponse à ces questions impose tout d'abord d'analyser les données d'expressions qui contiennent des images regroupées en sept classes d'émotion universelles : la joie, la peur, la tristesse, la surprise, la colère, le dégoût et l'expression de l'état neutre de l'émotion. Pour pouvoir réaliser l'analyse des images, nous répartissons cette tâche en deux chapitres : le chapitre 2 et 3. Alors que le présent chapitre décrit l'ensemble des méthodes utilisées pour l'analyse des images, le chapitre 3 décrit l'algorithme que nous avons construit pour automatiser l'analyse d'images.

Nous commençons ce chapitre par la détermination des méthodes d'extraction du vecteur des caractéristiques et des méthodes de classification. Dans la première section (2.2) nous introduisons trois types de descripteurs qui permettent l'extraction de la forme, la texture et le contour du visage. Ainsi, nous présentons les bases de données que nous avons choisies pour réaliser l'analyse des données d'expression. Dans la deuxième section (2.3), nous soulevons les méthodes de classification procédant à la séparation des classes des expressions de façon linéaire ou non-linéaire. Au niveau de ces deux sections, nous pressentons le principe théorique des méthodes tout en soulignant les principaux paramètres qui contrôlent le fonctionnement des descripteurs et des méthodes de classification.

2.2 Description du visage : extraction des caractéristiques visuelles de l'image du visage

Les émotions stimulent initialement les muscles du visage. Par conséquent, le mouvement des muscles apparaît à travers deux types de changements de l'apparence du visage impliqués lorsqu'une nouvelle émotion se produit. Le premier changement est lié aux mouvements des traits permanents, tels que les sourcils, les yeux, le nez et la bouche, tandis que le deuxième changement concerne les traits transitoires comme les rides qui apparaissent uniquement lorsque l'émotion se produit. Dans ce cas, la meilleure façon de décrire l'expression du visage est d'extraire les informations de position qui peuvent être formées lors du calcul de la distribution locale de l'orientation des contours, la variation du niveau de gris ou de détection du contour.

2.2.1 Description de la forme du visage : les caractéristiques HOG

Le descripteur HOG [7, 87, 90] permet d'extraire l'apparence de la forme du visage dans l'image en suivant le processus donné sur la figure 2-1. Dans cette figure, la forme est décrite

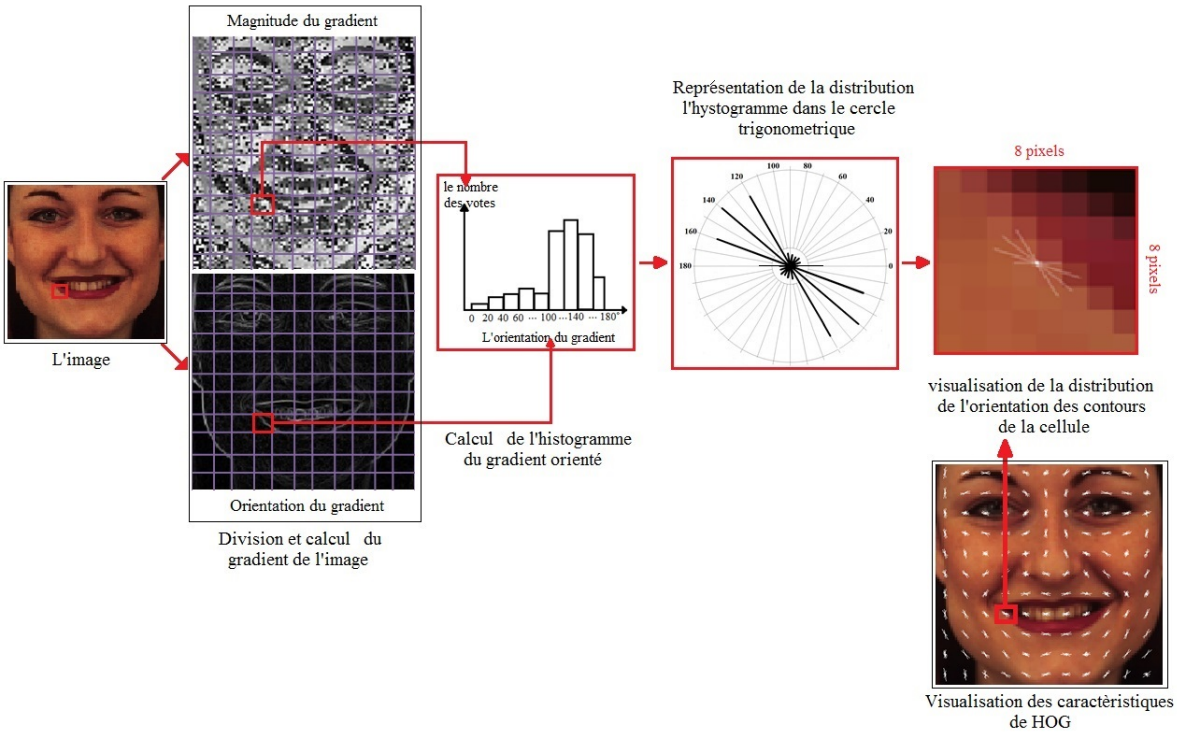


FIGURE 2-1 – Processus de description des caractéristiques de HOG d'une image de visage

par le calcul de la distribution de l'orientation de contour sur un petit ensemble restreint de pixels, résultant de la division de l'image du visage en cellules adjacentes de taille S égale à 8 x 8 pixels. Plus précisément, la description de la forme s'effectue sur la base du calcul des gradients horizontal et vertical de l'image, et ce, en appliquant un filtre dérivatif centré [-1 0 1] dans les directions horizontale et verticale. L'intérêt de l'utilisation du gradient consiste à calculer, sur chaque pixel de coordonnées (x, y) : l'orientation du gradient qui indique la direction dans laquelle le niveau de gris varie le plus et la norme du gradient qui indique l'intensité de la variation. Soit $I_{(x,y)}$ l'intensité de l'image au pixel de coordonnées (x,y), on calcule l'orientation du gradient $\theta_{(x,y)}$ et la norme du gradient $M_{(x,y)}$ de la façon suivante :

$$\theta_{(x,y)} = \text{atan2}(g(y), g(x)) = \begin{cases} \arctan(\frac{g(y)}{g(x)}) & \text{if } g(x) > 0, \\ \arctan(\frac{g(y)}{g(x)}) + \pi & \text{if } g(x) < 0 \text{ and } g(y) \geq 0, \\ \arctan(\frac{g(y)}{g(x)}) - \pi & \text{if } g(x) < 0 \text{ and } g(y) < 0, \\ +\frac{\pi}{2} & \text{if } g(x) = 0 \text{ and } g(y) > 0, \\ -\frac{\pi}{2} & \text{if } g(x) = 0 \text{ and } g(y) < 0, \\ \text{indéfinie} & \text{if } g(x) = 0 \text{ and } g(y) = 0. \end{cases} \tag{2.1}$$

$$M_{(x,y)} = \sqrt{g_x^2 + g_y^2} \tag{2.2}$$

$$g_x = \frac{I_{(x+1,y)} - I_{(x-1,y)}}{2} \tag{2.3}$$

$$g_y = \frac{I_{(x,y+1)} - I_{(x,y-1)}}{2} \tag{2.4}$$

$g_x \in \Re$ le gradient horizontal et $g_y \in \Re$ le gradient vertical, $-\pi < atan2(.,.) \leq \pi$ est une fonction à deux arguments g(y) et g(x), elle est une variante de la fonction arc tangente.

En effet, l'orientation $\theta_{(x,y)} \in]-\pi; \pi]$ et la magnitude $M_{(x,y)} \in \Re$, mesurés par les équations (2.1) et (2.2), contribuent à la construction des histogrammes représentant la distribution de l'orientation des gradients dans chaque cellule de la façon suivante : les orientations des gradients $\theta_{(x,y)}$ votent pour neuf classes de l'histogramme. Les classes sont également reparties entre les orientations 0° et 180° en fixant la largeur de chaque classe en $L = 20°$. D'autre part, la norme du gradient $M_{(x,y)}$ a pour rôle de donner un poids aux votes. À ce niveau, plus les variations autour du pixel sont élevées plus la norme du gradient est élevée. L'idée ici, c'est que dans les cellules qui contiennent des bords du visage où le niveau de gris des pixels varie beaucoup, certaines classes de l'histogramme reçoivent plus de poids que d'autres. Cela permet de donner une description distincte de la variation de la direction de contour sur la cellule comme illustré dans la figure 2-1. En combinant le résultat de toutes les cellules, la description de la forme globale du visage peut être achevée lorsque les paramètres du descripteur de HOG sont ajustés.

2.2.2 Description de la texture : les caractéristiques LBP

Ce descripteur compare le niveau de gris du pixel situé au centre d'une zone de l'image, de taille 3 x 3 pixels, avec le niveau de gris des pixels voisins de la même zone. L'avantage de comparer la luminance sur des petites cellules consiste à trouver des motifs locaux sur une petite échelle de l'image comme les coins et les bords. L'idée du descripteur LBP est de diviser l'image en une grille de k cellules $\{c_1, \ldots, c_k\}$. Ensuite, sur chacune des cellules, on mesure les caractéristiques LBP tout en appliquant une binarisation sur des zones rectangulaires, dont la taille est ajustable selon la valeur d'un rayon R (voir figure 2-2).

La binarisation est obtenue en affectant la valeur $b_i = 0$ aux pixels lorsque la valeur d'intensité de ces pixels voisins est inférieure à la valeur du pixel central, sinon les pixels

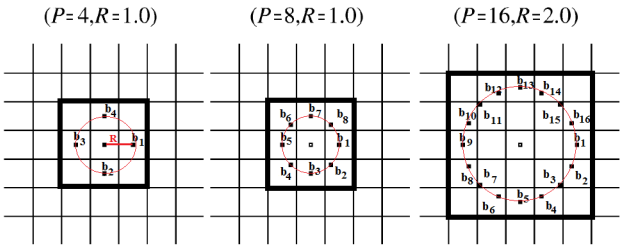


FIGURE 2-2 – Exemple d'extraction de trois motifs LBPs en variant le paramètre P indiquant le nombre de voisins et R le rayon déterminant la taille du motif LBP

voisins prennent la valeur $b_i = 1$. Pour chaque pixel de la cellule, le descripteur LBP mesure la valeur de la série binaire résultante $\sum_{i=0}^{P-1} b_i 2^i$ tout en considérant un certain nombre de voisins P repartis le long d'un cercle de rayon R égal à la distance entre le pixel central de la cellule et les pixels voisins. R et P sont des paramètres ajustables par l'utilisateur, la figure 2-2 donne un exemple graphique du motif binaire en faisant varier les paramètres P et R.

On peut distinguer deux types de séries binaires : les LBP uniformes et non-uniformes. Dans la présentation circulaire des LBP uniformes les transitions $0-1$ et $1-0$ sont limitées au plus à deux transitions. Cependant, les LBP non-uniformes peuvent contenir plus de deux transitions qui, généralement, résultent d'une région d'image bruyante (par exemple un background complexe). Le point intéressant est que les LBP uniformes sont conçus pour fonctionner comme des détecteurs des micro-motifs tels que les maxima et les minimas, les bords et les coins (voir figure 2-3).

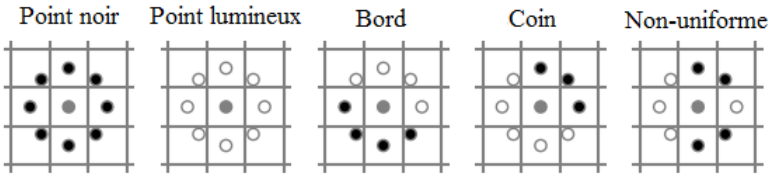


FIGURE 2-3 – Les quatre motifs à gauche illustre un exemple de motifs uniformes tandis que celui à droite représente un exemple de motif non-uniforme

Après avoir remplacé chaque pixel par sa valeur LBP, on passe à l'étape du recueil des statistiques d'occurrences de LBP sous forme d'histogramme. En effet, la structure de l'histogramme local des cellules est déterminée par l'ensemble des LBP uniformes, car les informations importantes sur l'image comme la variation du niveau de luminance et de contraste sont extraites par le motif uniforme. Autrement dit, dans chaque classe de l'histogramme local, on calcule le nombre d'occurrences de chaque LBP uniforme, tandis que le nombre d'occurrences

de tous les LBP non-uniformes est accumulé dans une seule classe de l'histogramme. Ceci nous permet de savoir si les caractéristiques LBP qui sont majoritaires dans chaque cellule sont celles représentant les bords de l'image.

2.2.3 Description de contours : les filtres de gabor

La fréquence spatiale permet de décrire les distributions périodiques de l'intensité dans une image. À ce niveau, les hautes fréquences spatiales correspondent à des motifs distincts sur l'image telle que les détails fins comme les contours. Dans le domaine fréquentiel, l'image est vue comme étant formée en superposant une série d'ondes sinusoïdales de différentes fréquences orientées dans différentes directions. D'une part, la valeur des pixels indique l'intensité ou l'amplitude d'une telle onde et, d'autre part, la position du pixel informe sur la fréquence et l'orientation de l'onde. En pratique, on ne désire sélectionner que certaines ondes ayant une fréquence et une amplitude élevées et une orientation spécifique qui peuvent aboutir ensemble à la discrimination de contours.

Le filtre de Gabor [158] est l'un des nombreux filtres dits passe-bande qui permettent de détecter le contour d'une image. Ce descripteur a été utilisé dans des applications d'analyse d'images qui exigent une grande précision de description du contenu de l'image, notamment la reconnaissance du texte [159], la détection de distorsion architecturale dans les mammographies [160] et l'imagerie cardiaque [161].

Un seul filtre de Gabor est considéré comme un détecteur de contour à une orientation θ et une fréquence spatiale f spécifique. Généralement, une image peut être filtrée par de nombreux filtres de Gabor avec différentes valeurs des paramètres θ et f. L'ensemble des filtres est appelé banque de Gabor, dont chacun des filtres est réglé à une valeur θ et f, permettant ainsi de parcourir au mieux les différentes fréquences qui existent dans l'image. Lorsque les filtres de Gabor sont bien ajustés, ils peuvent approximer le comportement du système visuel humain [162]. Ceci parce que le filtre de Gabor émule le système visuel humain en créant une banque de filtres où chacun répond à des contours ayant une certaine orientation et fréquence spatiale.

Le filtre de Gabor décrit une image I à travers une convolution de l'image avec l'ondelette de Gabor $g(x, y)$ qui est une sinusoïde complexe à modulation gaussienne :

$$g(x, y, f, \theta, \sigma, \Psi, \gamma) = \frac{1}{2\sqrt{\pi}\sigma} e^{-(\frac{x'^2+\gamma^2 y'^2}{2\sigma^2})} e^{i(2\pi f x'+\Psi)} \tag{2.5}$$

Avec

$$x' = xcos(\theta) + ysin(\theta) \tag{2.6}$$

$$y' = xsin(\theta) + ycos(\theta) \tag{2.7}$$

où x et y sont les coordonnées du pixel de l'image I, $\sigma \in \Re$ l'écart type, $f \in \Re$ la fréquence du filtre, $\theta \in [0; \pi]$ l'angle de rotation du filtre, $\gamma \in [0.23; 1]$ le rapport de forme spatial et $\Psi \in [0; \pi]$ la phase de la sinusoïde.

Pour garantir un fonctionnement optimal du filtre de Gabor $g(x, y, f, \theta, \sigma, \Psi, \gamma)$ l'ajustement des paramètres est nécessaire. Dans ce qui suit, nous donnons l'intervalle des valeurs qui ont été testées pour chacun des paramètres ainsi que la valeur optimale des paramètres.

- **L'écart type de l'enveloppe gaussienne σ :** est le paramètre qui change la taille de la région d'image analysée par le filtre. L'ajustement de ce paramètre dépend de la valeur choisie au niveau du paramètre de la fréquence spatiale f. La relation qui lie ces deux paramètres est b qui est la largeur de la bande de fréquence spatiale à demi-réponse, du filtre de Gabor, qui s'écrit :

$$\sigma = \frac{1}{f\pi}\sqrt{\frac{ln(2)}{2}}\frac{2^b + 1}{2^b - 1} \tag{2.8}$$

  b est en octaves

- **La fréquence du filtre f :** exprimée en cycle/pixel et peut s'exprimer aussi en fonction de la longueur d'onde $f = \frac{1}{\lambda}$. Ce paramètre ajuste la sinusoïde du filtre à une fréquence spatial qui laisse passer les hautes fréquences, ayant une bande passante b qui est approximativement égale à 1.2 octaves, et rejettent les autres. L'une des raisons de choisir la valeur 1.2 pour b, c'est de construire une banque de filtres de Gabor biologiquement inspirée de certaines mammifères, comme les chats et les macaques [163]. Pour cela, les fréquences des filtres de la banque de Gabor sont ajustées aux valeurs suivantes $f \in \{0.06, 0.1, 0.2, 0.3, 0.4\}$.

- **L'angle de rotation de l'enveloppe gaussienne θ :** exprimé en degré ou radian, est le paramètre qui ajuste le filtre à des orientations permettant de rechercher les contours orientés dans une direction particulière. Les valeurs valides relatives à ce paramètre sont des nombres réels compris entre 0° et 180° et peuvent être séparées de façon équidistante. Après l'ajustement de ce paramètre, les valeurs θ de la banque de Gabor

sont $\theta \in [0 : \frac{\pi}{8} : \pi]$ en choisissant un pas de $\frac{\pi}{8}$. Ici, plus on augmente le nombre d'orientations utilisées plus on obtient une représentation continue des contours. Mais ceci augmente le nombre des filtres dans la banque de Gabor ce qui nécessite aussi un calcul énorme pour construire le vecteur des caractéristiques relatif à ce descripteur.

- **le rapport de forme spatiale γ :** spécifie l'ellipticité du facteur gaussien. Après la réalisation des tests sur plusieurs valeurs de γ, les valeurs typiques de ce paramètre se situent entre $0.2 < \gamma < 1$. Ce paramètre affecte l'ellipticité du filtre, mais n'influence pas beaucoup les résultats de reconnaissance.

- **la phase de la sinusoïde Ψ :** détermine la symétrie de la fonction de Gabor concernée : pour $\Psi = 0°$ et $\Psi = 180°$, la fonction est symétrique. Ceci est important dans le filtrage d'énergie de Gabor, car en comparant au filtre de Gabor, le filtre d'énergie de Gabor donne une réponse plus lisse au contour de l'image. Les filtres d'énergie de Gabor produisent une réponse pour chaque filtre de sa banque.

$$E(x, y, \theta) = \sqrt{(I * G_{(\theta,0)})^2(x, y) + (I * G_{(\theta,\pi)})^2(x, y)} \tag{2.9}$$

($(I * G)(x, y)$ est la convolution de l'image I(x,y) avec le filtre de Gabor $G(x, y, \theta, \pi)$.

La réponse du filtre en énergie, dans l'équation (2.9), est obtenue par la somme de convolutions de l'image I avec $G(x, y, \theta, \Phi)$ obtenues à partir de deux décalages de phase différents $\Phi = 0$ et $\Phi = \pi$. Après l'ajustement des paramètres du descripteur de Gabor, la figure 2-4 montre le résultat du filtre d'énergie de Gabor appliqué aux images du visage avec deux émotions différentes.

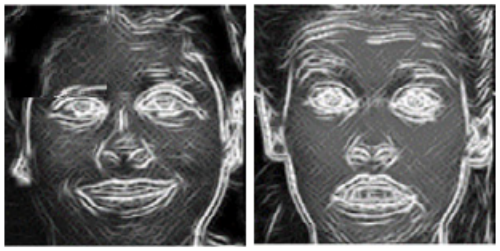


FIGURE 2-4 – Démonstration du filtrage de l'énergie de Gabor appliqué aux images d'expression.

Ainsi, l'association des filtres de Gabor permet de couvrir plus largement l'espace fréquentiel et, donc, d'extraire le contour de l'image suivant les orientations $\theta \in [0 : \frac{\pi}{8} : \pi]$.

2.2.4 Jeux de données utilisées pour l'analyse

Les images utilisées dans l'expérimentation ont été collectées à partir de deux bases de données différentes afin de diversifier les exemples au sein de l'ensemble d'apprentissage. La base d'images CK+ [12] est spécifiquement acquise pour l'analyse de l'expression faciale. Elle a été utilisée pour le développement, l'évaluation et la comparaison des systèmes de reconnaissance de l'expression du visage [87, 90, 164, 165]. Le choix de cette base est motivé par le fait que cette base contient des échantillons diversifiés au sein des mêmes classes en termes de sexe, d'origine ethnique et d'âge. En plus de la diversité des échantillons, la base CK + contient des séquences d'images de sept expressions faciales lesquelles commencent par l'expression neutre et se terminent par un visage expressif exposant l'expression de : joie, surprise, colère, peur, dégoût, tristesse ou état neutre. Le nombre d'images dans les séquences varie de 9 à 60 images. En utilisant des séquences d'images on peut tirer profit du changement de l'expression au cours du temps. Par conséquent, on peut exploiter les images où l'intensité de l'expression est faible lors de la transition de l'expression de l'état neutre jusqu'à l'état expressif, et ce, pour améliorer l'apprentissage des classifieurs.

Par ailleurs, la base d'images Yale Face [16] a également été utilisée. Cette base contient 165 exemples d'images de 15 individus apparus dans 11 conditions d'observation différentes. Les exemples contiennent des images de visage qui expriment différentes expressions comme la joie, la tristesse, l'ennui, la surprise et un clin d'œil sous différentes conditions d'éclairement et avec des occlusions partielles. L'avantage de l'utilisation de la base de Yale Face conjointement à la base de données CK+ est simplement d'augmenter la diversité des échantillons en ajoutant au jeu de données CK+ des images à visages partiellement camouflées.

2.2.4.1 Préparation des données pour l'apprentissage des classifieurs

L'étiquetage des images selon l'expression du visage fait partie du processus de préparation des données. Cela, se fait en regroupant les images de tous les individus dans des dossiers classifiés selon les classes d'émotions universelles. Avant de commencer l'entraînement des algorithmes de classification, la position du visage est ajustée au centre du cadre qui englobe le visage. Ensuite, toutes les images RGB sont transformées en niveau de gris pour qu'ensuite les descripteurs transforment à leur tour chacune des images de visage en un vecteur de caractéristiques X. L'ensemble des vecteurs caractéristiques et leurs étiquettes, notées y, sont stockés dans des tableaux notés D_i, comme l'illustre la figure 2-5.

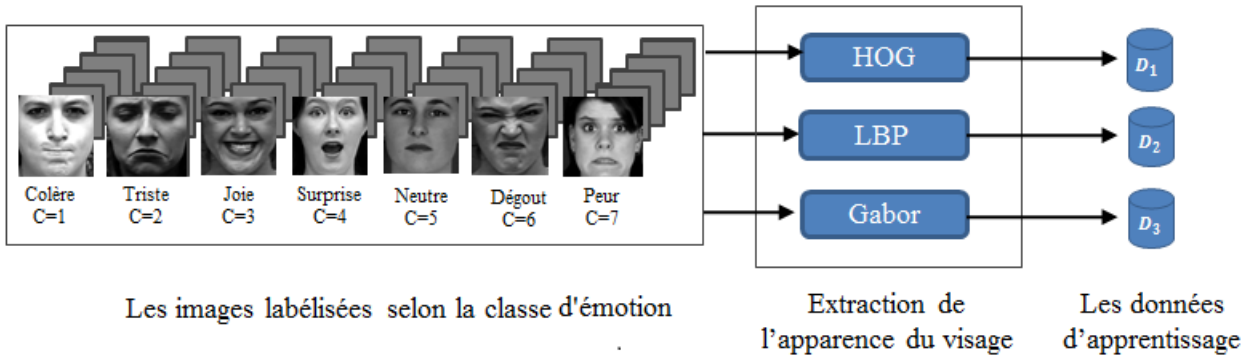


FIGURE 2-5 – Construction des données d'apprentissage

Mathématiquement, les données Ω issues de chacun des descripteurs utilisés dans l'analyse s'écrivent comme :

$$\Omega = \Omega^{D_i}_{(N,P)} = \begin{array}{c} \\ X_1 \\ X_2 \\ \vdots \\ X_N \end{array} \begin{array}{c} \begin{array}{cccccc} x_1 & x_2 & \cdots & x_P & C \end{array} \\ \begin{pmatrix} x_{1,1} & x_{1,2} & \cdots & x_{1,P} & y_1 \\ x_{2,1} & x_{2,2} & \cdots & x_{2,P} & y_2 \\ \vdots & \ddots & \vdots & \vdots & \\ x_{N,1} & x_{N,2} & \cdots & x_{N,P} & y_N \end{pmatrix} \end{array} \tag{2.10}$$

Par la suite, nous gardons les mêmes notations pour exprimer :

- Ω : les données d'apprentissage.
- D_i : le i-ième descripteur d'image pré-sélectionné
- X_i : le i-ième exemple ou encore le i-ième vecteur de caractéristiques extrait par le descripteur d'image.
- P : le nombre de caractéristiques extraites par le descripteur d'image.
- x_j : le j-ième attribut de données d'apprentissage. Dans la suite, lorsqu'on parle de la réduction de dimension de données d'apprentissage, on désigne par ceci la réduction du nombre d'attributs.
- C : le vecteur cible contenant les étiquettes $y_i \in \{1, 2, ..., 7\}$ des exemples X_i, dont chaque nombre indique une expression (voir figure 2-5).
- N : le nombre total des exemples utilisés pour l'expérimentation qui vaut 547 et qui représente les images extraites des séquences de la base CK+ [12] et les images de la base Yale-Face [16]. Cette quantité d'images est utilisée pour éviter l'élevation du temps d'analyse des images, à l'ordre de 6 mois de calculs, particulierement lorsqu'il s'agit de rechercher

la combinaison optimale des paramètres des descripteurs et des classifieurs, voir plus de détails à la section 3.3.3

TABLE 2.1 – Le nombre des exemples des classes construites à partir des bases de données Ck+ et Yale-face

	Colère	Dégout	Peur	Joie	état neutre	Tristesse	Surprise
Nombre d'exemples	33	43	36	103	71	91	197

2.3 Méthodes de classification

L'apprentissage est le processus qui consiste à chercher une fonction cible $f(X_i) = y_i$ qui interpole l'ensemble de données $\Omega_{(N,P)}$ de couples entrée-sortie $Z_i = (X_i, y_i)$, qui s'écrit de la façon suivante : $\Omega_{(N,P)} = (X_1, y_1), (X_2, y_2), \ldots, (X_N, y_N)$, dont $X_i \in \Re^P$ et $y_i \in \{1, 2, \ldots, 7\}$. Étant donné l'ensemble données $\Omega_{(N,P)}$, on cherche alors à construire une fonction cible $f(X)$ à partir d'une approximation $h_w(X)$ simplificatrice, appelée hypothèse, qui lie les étiquettes y_i aux attributs X_i tout en convergeant vers y_i, on note :

$$\forall X_i \in \Omega_{(N,P)}, b \in \Re | h_w(X_i) = \sum_{j=1}^{p} w_j x_{i,j} + b \approx f(X_i) = y_i \tag{2.11}$$

b le biais[1]*,* $W = \{w_1, w_2, ..., w_P\} \in \Re^P$ *le vecteur des poids lié au vecteur des caractéristiques* $X = \{x_1, x_2, ..., x_P\}$

Avant l'apprentissage, les paramètres de poids w_i et le biais b prennent des valeurs aléatoires, mais durant l'apprentissage ces paramètres subissent des mises à jour jusqu'à ce que l'erreur empirique suivante converge vers zéro.

$$e = h_w(X_i) - y_i \tag{2.12}$$

L'objectif de l'algorithme d'apprentissage est de fixer les paramètres de poids $W \in \Re^P$ et le

1. Le biais est le paramètre qui mesure la tendance d'un système à mener un mauvais apprentissage. Une petite valeur du biais qui tend vers zéro indique le phénomène de sur-apprentissage (overfitting), et une grande valeur »1 du biais indique le phénomène de sous-apprentissage (Underfitting). Ces deux phénomènes sont les causes principales des mauvaises performances des modèles prédictifs générés par les algorithmes de classification

biais $b \in \Re$ afin de chercher l'hypothèse optimale $h_{optimale}$ qui sépare au mieux les classes de données comme illustré dans la figure 2-6). Dans cette figure, l'algorithme d'apprentissage du classifieur transforme l'espace d'entrée χ des données d'entraînement en un espace d'hypothèses Ψ dans lequel on cherche l'hypothèse optimale $h_{optimale}$ qui minimise le risque d'avoir des exemples mal classés. L'espace d'hypothèses nous permet de trouver toutes les frontières de décision optimales qui séparent les classes de nos données d'apprentissage.

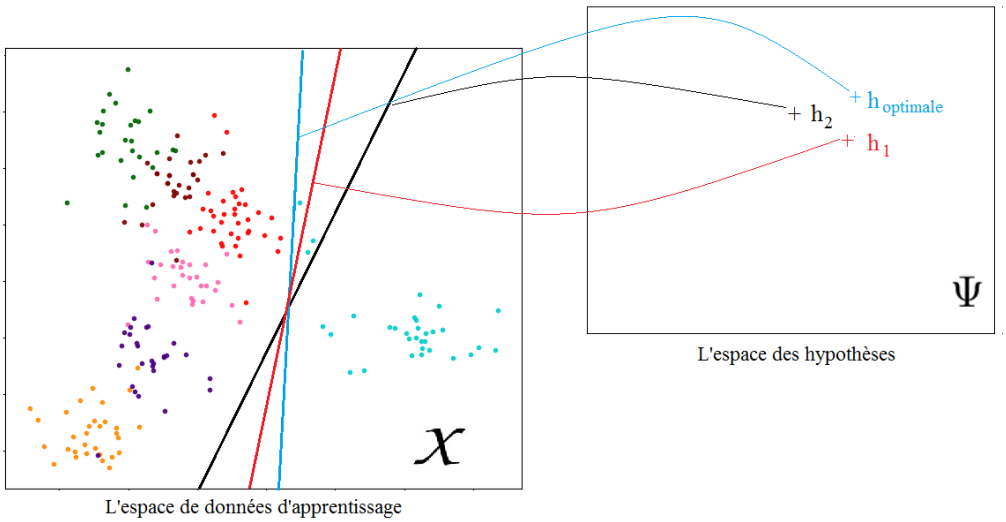


FIGURE 2-6 – Illustration d'un espace d'hypothèses Ψ. Chaque hypothèse h_i correspond à une partition de l'espace d'entrée χ des sept classes de données d'apprentissage. Les points illustrent les exemples X

Géométriquement, la frontière de décision, nommée aussi fonction séparatrice ou encore hyperplan séparateur [2] ou ligne séparatrice est définie comme la droite qui sépare les groupes des exemples appartenant aux différentes classes (clusters en anglais) d'expression. Mathématiquement, la frontière de décision est égale à la fonction $h_w(X)$ lorsque l'équation (2.11) s'annule, on écrit :

$$b \in \Re | \sum_{j=1}^{p} w_j x_j + b = 0 \tag{2.13}$$

b le biais, $w_j \in \Re$, pour $j = \{1, \ldots, P\}$, le poid de l'attributs $x_j \in \Re$

D'après l'équation (2.13) l'hyperplan séparateur est linéaire, dont les x_i sont ses coordonnées dans l'espace de dimension $\Re^P$. Pourtant, dans le chapitre 3, nous menons une évaluation de l'efficacité des frontières de décision non-linéaires sur les clusters de données

2. Lorsque les données d'apprentissage ont un nombre d'attributs qui dépasse strictement deux attributs ont dit hyperplan séparateur. Sinon on dit ligne ou droite séparatrice.

construits par la méthode de Fisher. Dans les sous-sections suivantes, nous décrivons les méthodes de classification [154, 166–169] que nous avons pré-sélectionnées pour analyser les données d'expressions encodées par les descripteurs d'image. Contrairement aux descripteurs, les méthodes de classification ont des hyperparamètres [3]. La différence est que les algorithmes des méthodes de classification ajustent eux-mêmes, durant l'entraînement, leurs paramètres comme par exemples les poids W, le biais b et l'hypothèse h à partir des données d'apprentissage, tandis que l'ajustement des hyperparamètres, comme l'hyperparamètre k dans la méthode du k-plus proches voisin, nécessitent l'intervention de l'utilisateur pour définir et tester un ensemble de valeurs pour chacun des hyperparamètres.

2.3.1 La méthode Fisher

Dans la littérature, souvent, la méthode Fisher est utilisée pour la réduction de la dimension des données plutôt que pour la classification. Le choix d'utiliser cette méthode pour la classification dépend de la relation entre les classes lorsque les données d'origine (non réduite) sont transformées, par un calcul statistiques, en un nouvel ensemble de données de faibles dimensions. En réduisant les dimensions des données d'apprentissage Ω par la méthode de Fisher et en visualisant les clusters construits dans la section 2.3.1.1, nous démontrons dans la section 2.3.1.2 qu'on peut appliquer une séparation linéaire des clusters et construire un classifieur d'expression

2.3.1.1 Processus de réduction de dimension de données d'apprentissage

L'idée d'utiliser cette méthode [146] est de transformer par un calcul statistiques la matrice $\Omega_{(N,P)}$ ayant P attributs en une autre matrice ayant moins d'attributs. En appliquant cette transformation, on peut représenter les données comme un nuage de points dans un diagramme de dispersion permettant ainsi une vision plus claire de la distribution des clusters dans un espace à deux axes (voir figure 2-7). Cela nous permet de comprendre la relation géométrique entre les exemples des clusters de données au cours de la phase d'apprentissage, de sorte que l'on puisse séparer ces clusters efficacement et ainsi construire un classifieur approprié pour les expressions de visage.

3. Pour la raison d'organisation des idées, dans l'introduction de ce chapitre nous avons parlé que des paramètres, mais en réalité la notion d'ajustement des paramètres varie entre le descripteur d'image et le classifieur. Ce dernier contient des paramètres que l'algorithme du classifieur lui-même ajuste durant l'entraînement en utilisant les données d'apprentissage. D'autre part, les paramètres des descripteurs et les hyperparamètres des classifieurs nous les ajustons en utilisant l'algorithme que nous proposons dans le chapitre 3

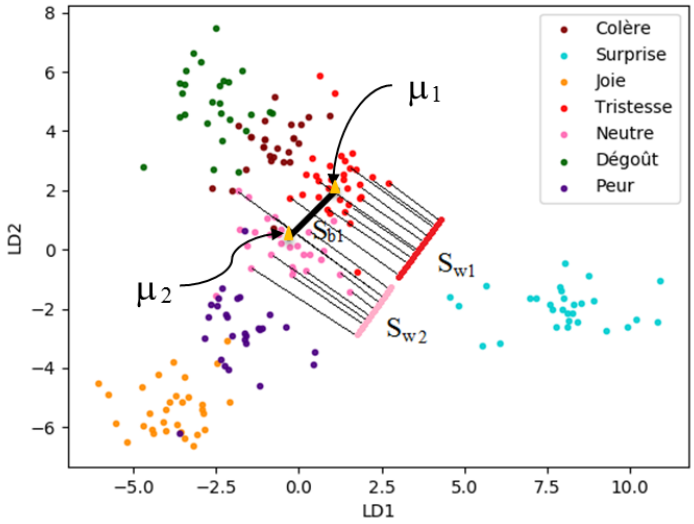


FIGURE 2-7 – Diagramme de dispersion de données montrant la variance inter-classe et intra-classe. Les points jaunes représentent la moyenne des classes

La création du nouvel espace d'attributs à partir de données $\Omega_{(N,P)}$ se fait par l'application des calculs statistiques sur les données, tels que le calcul de la moyenne de chaque classe μ_C, la moyenne de la totalité des classes μ et la covariance de chaque classe avec elle-même.

$$\mu_C = \frac{1}{M_c}[\sum_{j=1}^{M_c} x_{(j,1)}, \sum_{j=1}^{M_c} x_{(j,2)}, ..., \sum_{j=1}^{M_c} x_{(j,P)}] \tag{2.14}$$

$$\mu = \frac{1}{N}[\sum_{j=1}^{N} x_{(j,1)}, \sum_{j=1}^{N} x_{(j,2)}, ..., \sum_{j=1}^{N} x_{(j,P)}] \tag{2.15}$$

M_c le nombre des exemples dans chaque classe, N le nombre total des exemples de toutes les classes et $x_{(j,i)} \in \Re$ un attribut de données Ω

L'intérêt du calcul de la matrice de covariance est de déterminer la quantité de variances dans les données. Alors, parvenir à mesurer la quantité de variances revient à multiplier la matrice de covariance de chaque classe par sa probabilité P_C, qui est le nombre d'exemples dans la classe divisé par le nombre total des exemples dans les données. En additionnant le résultat pour toutes les classes, on obtient la variance intra-classe $S_W \in \Re^{P \times P}$.

$$S_w = \sum_{C=1}^{7} \sum_{i=1}^{M_c} P_C.(X_i - \mu_C)(X_i - \mu_C)^T \tag{2.16}$$

$P_C = \frac{M_c}{N}$ est la probabilité de chaque classe, $X_i \in \Re^P$ le ième exemple de données Ω, $\mu_C \in \Re^P$ la moyenne de chaque classe C (Voir équation (2.14)*)*

De point de vue géométrique, sur la figure 2-7, la variance S_{W1} (Resp. S_{W2}) est la distance en rouge (Resp. en rose) contenant la totalité des projections des points de la classe de tristesse (Resp. la classe de l'état neutre) sur une droite séparatrice linéaire. Avec la mesure S_{W_i}, nous pouvons savoir si les exemples appartenant à la même classe sont étroitement regroupés. Par exemple plus la valeur de S_{W1} est petite, plus on a des points dans le nuage de l'expression de tristesse qui sont proches les uns des autres.

Comme la variance intra-classe nous informe sur la dispersion des exemples de même classe, nous devons également mesurer la variance inter-classe $S_b \in \Re$ afin de savoir si les clusters peuvent être facilement séparés. Par exemple, sur la figure 2-7 la variance inter-classe S_{b1} fait référence à la distance entre les points jaunes, dont chacun d'eux indique le point moyen des classes de l'expression neutre et celle de la tristesse. On définit $S_b \in \Re^{P \times P}$ par :

$$S_b = \sum_{C=1}^{7} (\mu_C - \mu)(\mu_C - \mu)^T \tag{2.17}$$

$\mu_C \in \Re^P$ la moyenne de chaque classe C (Voir équation (2.14)*), $\mu \in \Re^P$ la moyenne de données (Voir équation* (2.15)*).*

Généralement, avoir une petite valeur de S_W et une grande valeur de S_b signifie qu'il est facile de séparer les classes. Donc, rendre le ratio $S_W^{-1} S_b$ aussi grande que possible est la clé pour réduire le nombre d'attributs P jusqu'à obtenir deux attributs notés $LDA_1 \in \Re^N$ et $LDA_2 \in \Re^N$ qui contiennent presque toute l'information de classification des N exemples dans $\Omega \in \Re^{P \times N}$. Les deux attributs sont donc calculés en résolvant les vecteurs propres généralisés de la matrice $S_W^{-1} S_b$ avec l'équation (2.18) et en sélectionnant les vecteurs propres υ ayant des valeurs propres maximales de λ.

$$A\upsilon = \lambda\upsilon \tag{2.18}$$

$A = S_W^{-1} S_b$, tout en supposant que S_W^{-1} existe.
Les deux vecteurs propres qui ont des valeurs propres maximales, représentent les valeurs des deux attributs des données réduites $\Omega' \in \Re^{N \times 2}$. Ces deux attributs sont ensuite tracés dans le diagramme de dispersion 2D de la figure 2-7, dont l'axe d'attribut LDA_1 représentant les valeurs du premier vecteur propre et l'axe d'attribut LDA_2 représentent les valeurs du

deuxième.

$$\Omega'_{(N,2)} = \begin{array}{c} \begin{matrix} LDA_1 & LDA_2 & \end{matrix} \\ \begin{pmatrix} x'_{1,1} & x'_{1,2} & y_1 \\ x'_{2,1} & x'_{2,2} & y_2 \\ \vdots & \vdots & \vdots \\ x'_{N,1} & x'_{N,2} & y_N \end{pmatrix} \end{array} \tag{2.19}$$

En appliquant un calcul statistique de la moyenne et de covariance sur les classes de données $\Omega' \in \Re^{N\times 2}$ tout en utilisant les équations (2.21), (2.22) et (2.20), on peut décrire les données et comprendre la répartition des exemples dans les classes. Ces statistiques, nous serons utiles pour déterminer les droites séparatrices entre les classes de données $\Omega' \in \Re^{N\times 2}$ et seront utilisées aussi dans le chapitre 3 pour rechercher des frontières de décision non-linéaires par les techniques de l'analyse discriminante quadratique (Voir section 4.2.5.2) et Naïve Bayésienne (Voir section 4.2.5.3).

$$\Sigma = \begin{pmatrix} \sigma_1^2 & \sigma_{1,2} \\ \sigma_{2,1} & \sigma_2^2 \end{pmatrix} \tag{2.20}$$

$$\sigma_i^2 = \frac{1}{M_c}\sum_{j=1}^{M_c}(x'_{j,i} - \bar{x'_i})^2 \tag{2.21}$$

$$\sigma_{i,k} = \frac{1}{M_c}\sum_{j=1}^{M_c}(x'_{j,i} - \bar{x'_i})(x'_{j,k} - \bar{x'_k})^T \tag{2.22}$$

$\bar{x'_i} = \frac{\sum_{j=1}^{M_c} x'_{(j,i)}}{M_c} \in \Re$ *la moyenne de l'attribut* x'_i*, pour* $i = \{1,2\}$*,* $\sigma_{i,k} \in \Re$*, pour* $k = \{1,2\}$ *la covraiance des attributs* x'_1 *et* x'_2*. Lorsque* $i{=}k$ $\sigma_{i,k} \in \Re$ *s'écrit* $\sigma_i^2 \in \Re$*.*

2.3.1.2 Construction du classifieur : analyse discriminante linéaire

L'analyse discriminante linéaire repose sur l'hypothèse que les données proviennent d'une distribution multivariée (Voir figure 2-8) on peut trouver des droites séparatrices linéaires en calculant la densité conditionnelle des classes $p(X = x'|C = c)$ en utilisant la loi de la distribution normale suivante :

$$p(X = x'/C = c) = f_c(x') = \frac{1}{\Sigma_c\sqrt{2\pi}} e^{-\frac{(X-\mu'_c)\Sigma_c^{-1}(X-\mu'_c)^T}{2}} \tag{2.23}$$

$\sigma_c = \{\sigma_1, \sigma_2\} \in \Re^2$, *pour* $c \in \{1, 2, ..., 7\}$, *est la variance relative à la classe c des données* $\Omega'_{(N,2)}$. $\mu'_c = \{\bar{x'_1}, \bar{x'_2}\} \in \Re^2$ *la moyenne de chaque classe C.*

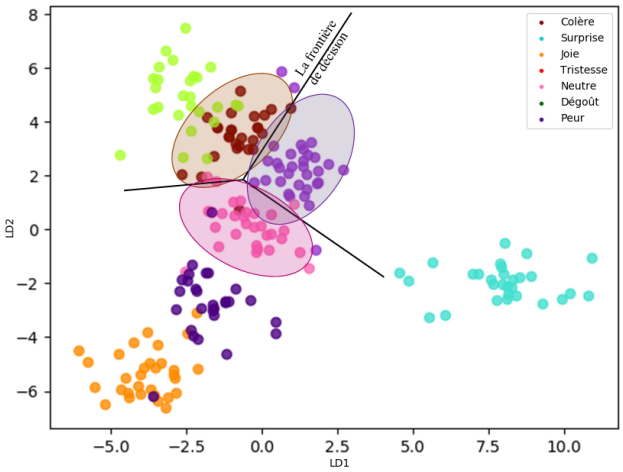


FIGURE 2-8 – Exemple de principe de séparation linéaire tout en supposant que les données ont une distribution normale multivariée.

En ayant recours à la règle de Bayes pour déterminer la probabilité des classes étant donné les exemples X, on note :

$$p(C = c/X = x') = \frac{f_c(x)p(c)}{p(X = x')} \tag{2.24}$$

$p(c)$ *la probabilité de la classe qui vaut le rapport du nombre d'échantillons dans la classe c au nombre total des échantillons.* $p(X = x')$ *est la probabilité de données X.*

En remplaçant, dans l'équation (2.24), f(x) par son expression donnée en 2.23 et en absorbant tout ce qui ne dépend pas de c dans une constante notée β, on obtient :

$$p(C = c/X = x') = \beta p(c) e^{-\frac{(X-\mu'_c)\Sigma_c^{-1}(X-\mu'_c)^T}{2}} \tag{2.25}$$

En introduisant le logarithme des deux côtés de l'équation (2.25) :

$$\log(p(C = c/X = x')) = \log(\beta) + \log(p(c)) - \frac{(X-\mu'_c)\Sigma_c^{-1}(X-\mu'_c)^T}{2} \tag{2.26}$$

En absorbant tout ce qui ne dépend pas de c dans une autre constante notée β', on obtient

l'équation suivante :

$$\log(p(C = c/X = x')) = \beta' + \log(p(c)) - \frac{\mu'_c \Sigma_c^{-1} \mu_c'^T}{2} + X\Sigma_c^{-1}\mu_c'^T \tag{2.27}$$

En définissant une fonction objectif $\delta_c(x)$ qui détermine l'hyperplan séparateur :

$$\delta_c(x) = \log(p(c)) - \frac{\mu'_c \Sigma_c^{-1} \mu_c'^T}{2} + X\Sigma_c^{-1}\mu_c'^T \tag{2.28}$$

La frontière de décision est donc l'ensemble des points pour lesquels les fonctions objectifs des classes adjacentes sont égales. Soit donc la classe j adjacente à une autre classe i, on note alors :

$$\delta_i(x) = \delta_j(x) \tag{2.29}$$

$$\log(p(c_j)) - \frac{\mu'_j \Sigma_j^{-1} \mu_j'^T}{2} + X\Sigma_j^{-1}\mu_j'^T = \log(p(c_i)) - \frac{\mu'_i \Sigma_i^{-1} \mu_i'^T}{2} + X\Sigma_i^{-1}\mu_i'^T \tag{2.30}$$

La figure 2-8 illustre un exemple de la frontière de décision entre trois classes. La droite de séparation est donc la droite passant par les points d'intersection entre les distributions multivariées $f_c(x)$ des classes. En calculant l'equation 2.30 pour toutes les classes adjacentes, nous obtenons le résultat de séparation présenté dans la figure 2-9

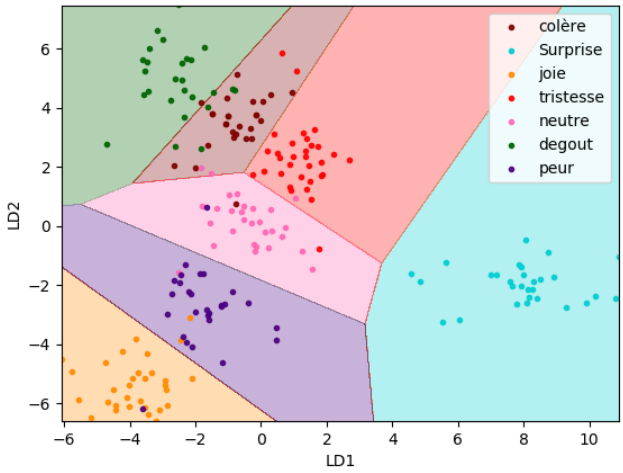


FIGURE 2-9 – Visualisation géométrique du résultat de séparation des classes d'expressions par l'analyse discriminante linéaire

Dans cette section, nous avons introduit la méthode de Fisher à la fois comme approche de réduction de nombre d'attributs des données et aussi comme un classifieur permettant, grâce à la fonction objectif, une séparation linéaire des classes.

2.3.2 La méthode de classification Naïve Bayésienne

Le classifieur naïf de Bayes [168] utilise un calcul probabiliste qui repose essentiellement sur le théorème de Bayes (Voir équation (2.24)) pour trouver l'hypothèse $h_{optimal} \in \Psi$ qui minimise l'erreur commise (Voir équation (2.12)) vis-à-vis de la fonction cible f, c.-à-d. de trouver l'application de l'espace d'entrée χ dans l'espace des hypothèses Ψ qui généralise le mieux les couples de $\Omega = \{(X_1, y_1), (X_2, y_2), \ldots, (X_N, y_N)\}$, avec $X_i = \{x_{i,1}, x_{i,2}, \ldots, x_{i,P}\}$ (Voir équation (2.10)).

La règle de décision bayésienne consiste par définition à trouver l'hypothèse $h_{optimal}$, dans Ψ, qui a la plus la grande probabilité connaissant les données d'apprentissage Ω, $h_{optimal}$. Elle est définie par l'équation suivante :

$$h_{optimal}(X) = \underset{C \in \{1,\ldots,7\}}{\arg\max}\, p(C|X), \tag{2.31}$$

$p(C|X)$ la probabilité à posteriori de la classe d'expression C sachant les exemples X et peut être calculée en utilisant la formule de Bayes dans l'équation (2.24).

Quand on suppose que les attributs x de description $X = \{x_1, x_2, \ldots, x_P\}$ dans l'éspace d'entrée χ sont indépendants les uns des autres, étant donné la classe d'expression C, on peut décomposer $p(X|C)$ en $\{p(X = X_1|C = c), \ldots, p(X = x_P|C = c)\}$, soit :

$$p(X/C) = \prod_{j=1}^{P} p(X = x_j|C) \tag{2.32}$$

$x_j \in \Re$ le jème attribut de l'exemple X dans Ω, et P le nombre des attributs du vecteur des caractéristiques.

En remplaçant $p(X/C)$ par son expression dans l'équation (2.24), on obtient :

$$P(C/X) = \frac{\prod_{j=1}^{P} P(X = x_j|C = c)P(C)}{P(X)} \tag{2.33}$$

Le dénominateur $P(X)$ peut être ignoré puisqu'il s'agit d'une constante positive par rapport aux classes C, lorsqu'on remplace l'équation (2.31) dans 2.33. Ainsi, pour trouver la règle d'apprentissage du classifieur naïf de Bayes, cela consiste à sélectionner la classe C pour

laquelle l'équation suivante est un maximum :

$$h_{optimal}(x) = \underset{C\in\{1,\ldots,7\}}{\arg\max} P(C) \prod_{j=1}^{P} P(X = x_j|C = c) \tag{2.34}$$

$P(X = x|C = c)$ et $P(C)$ sont deux paramètres ajustables selon les données utilisées. En supposant que les attributs associés à chacune des classes ont une distribution normale, la densité de probabilité de $X \in \Re^P$ peut être calculée en utilisant la fonction de Gauss :

$$p(X = x \mid C = c) = \frac{1}{\sqrt{2\pi\Sigma_c^2}} e^{-\frac{(x-\mu_c)\Sigma_c^{-1}(x-\mu_c)^T}{2}} \tag{2.35}$$

$\Sigma_c \in \Re^{P\times P}$ *est la matrice covariance des attributs de chaque classe C, elle est calculée en utilisant l'équation* (2.21). $\mu_c = \{\bar{x_1}, \ldots, \bar{x_P}\} \in \Re^P$ *la moyenne des attributs* $\Omega \in \Re^P$ *dans chaque classe C*

Le classifieur naïf de Bayes s'entraîne sur les exemples d'apprentissage en cherchant la densité de probabilité des attributs, cela permet de classifier un nouvel exemple, qui n'appartient pas aux données d'entraînement, rapidement. Nous verrons dans les sections 4.2.6 et 3.4 que lorsqu'on maximise la variance interclasse de donnée Ω par la méthode de réduction de dimension de Fisher, le classifieur naïf de Bayes produit des frontières de séparation qui augmente le taux de reconnaissance de 92% à 95%.

2.3.3 La méthode du Séparateur à Vaste Marge (SVM)

La méthode originale de SVM permet de créer des frontières de décision linéaires après avoir maximisé la marge M, qui est la distance entre l'hyperplan séparateur (cas de données de plusieurs dimensions) ou la ligne séparatrice (cas de deux dimensions) et les points des échantillons les plus proches. Pour une séparation linéaire des classes de données, la méthode SVM se base sur l'équation (2.36), qui produit des hyperplans linéaires, et cela, après avoir déterminé les poids $W = \{w_1, \ldots, w_P\} \in \Re^P$ en appliquant cette fois-ci la fonction signe sur l'hypothèse $h_w(X)$, on note alors :

$$\forall X_i \in \Omega, b \in \Re | h_w(X_i) = sign(\sum_{j=1}^{p} w_j x_{i,j} + b) \tag{2.36}$$

$$\sum_{j=1}^{p} w_j x_{i,j} + b \begin{cases} > 0 \\ < 0 \end{cases} \Longrightarrow y_i = \begin{cases} +1 \\ -1 \end{cases} \tag{2.37}$$

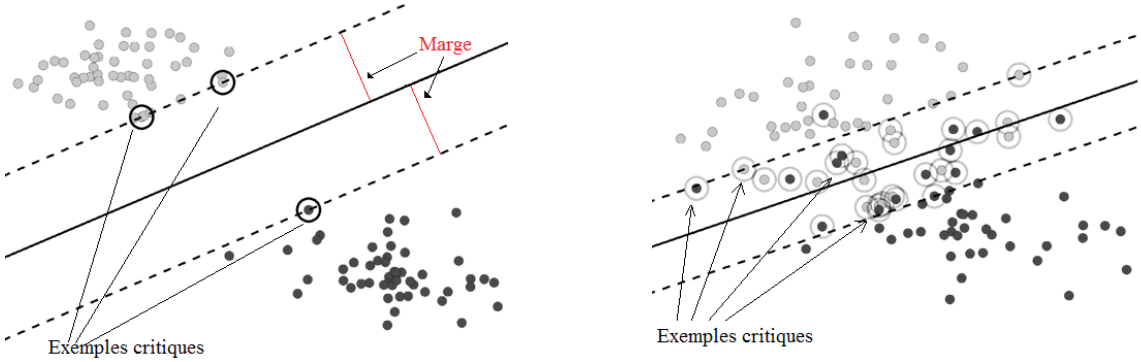
$x_{i,j} \in \Re$ le jème attribut du ième exemple X_i, $b \in \Re$ le biais, $y_i \in \{1, -1\}$ l'étiquette de l'exemple X_i.

Un SVM trouve une frontière de décision aussi éloignée que possible de deux clusters adjacents en maximisant la marge M. La recherche de frontière de décision optimale revient donc à résoudre le problème d'optimisation suivant qui porte sur les paramètres W et b :

$$\begin{cases} \text{Minimiser} & \frac{1}{2}\|W\|^2 \\ \text{sous les contraintes} & sign(\sum_{j=1}^{p} w_j x_j + b) > 1 \end{cases} \tag{2.38}$$

$W = \{w_1, \ldots, w_P\} \in \Re^P$ le vecteur des poids w_j associé aux attributs x_j, $\|W\| = \sqrt{\sum_{i=1}^{P} w_i^2}$ la norme Euclidienne indiquant la distance entre l'hyperplan séparateur et les exemples proches de l'hyperplan, $b \in \Re$ le biais.

La figure 2-10a illustre un exemple simplifié de frontière de décision qui est une ligne séparatrice que nous avons tracée en 2D. Dans cette figure (2-10a), les exemples encerclés sont appelés des exemples critiques (support vectors en anglais). La première étape de la méthode SVM est de déterminer ces exemples critiques pour, ensuite, définir la ligne pointillée. Ensuite, dans la deuxième étape, la méthode SVM recherche la ligne séparatrice qui est la ligne



(a) sans chevauchement de points, C=10^{10}

(b) présence de chevauchement de points, C=10^5

FIGURE 2-10 – ligne séparatricede SVM. La valeur de C spécifie le nombre d'exemples autorisés près de la limite la ligne continue

continue se trouvant au milieu des lignes pointillées. La distance entre la ligne continue et les lignes pointillées est la marge M.

Dans l'exemple de la figure 2-10a nous avons expliqué le principe de la méthode SVM en simplifiant le chevauchement de données. Cependant, dans notre cas, les données contiennent un chevauchement. Alors, comment peut-on séparer des classes qui se chevauchent, comme dans la figure 2-10b. La méthode SVM contient un hyperparamètre "clearance" appelé C qui contrôle le compromis entre le nombre d'erreurs de classement et la largeur de la marge. Plus la valeur de C est grande, plus la zone est vide autour de la ligne continue. Plus la valeur de C est petite, plus de points apparaissent dans la zone autour de la ligne continue. La méthode SVM est sensible à la valeur de C et à la structure de données que nous utilisons. Pour cette raison, nous devons ajuster cet hyperparamètre à la valeur convenable, c.-à-d. maximiser la marge M en gardant les erreurs de classification minimales. Généralement, cela signifie que nous devons essayer de nombreuses valeurs de C et les évaluer en exploitant la technique de "la validation croisée" présentée dans la section 3.1.1 du chapitre 3.

2.3.3.1 Le cas de non-linéarité de données

En réalité, lorsqu'on a des données contenant des exemples X_i définis comme un ensemble de points dans l'espace $\Re^P$ où le nombre d'attributs P dépasse trois attributs, il n'est pas aussi évident ni de visualiser (Comme dans le cas de la méthode Fisher sur la figure 2-7) le nuage des points des exemples des données pour observer les frontières des classes et ni de savoir si les classes sont linéairement ou non-linéairement séparables. Lorsque les données sont non-linéairement séparables, on peut réaliser une redescription de l'espace d'entrée χ et rendre les données Ω linéairement séparables dans un nouvel espace de redescription des attributs. Ceci, en utilisant l'une des fonctions noyau du tableau 2.2.

TABLE 2.2 – Quelques fonctions kernel aboutissant à la redescription des données d'apprentissage

Fonction noyau	Formule mathématique	1c|Paramètres de la fonction
Fonctions à base radiale	$K(X, X_i) = exp(-\frac{(X-X_i)^2}{2\sigma^2})$	l'écart-type σ^2 est un paramètre qui devait être ajusté selon les données
polynomiale	$K(X, X_i) = (1 + XX_i^T)^l$	le paramètre l est l'ordre du polynôme. celui-ci peut être ajusté après avoir testé plusieurs valeurs de l. Si l=1 cela donne une droite de séparation linéaire
fonctions sigmoïdes	$K(X, X_i) = tanh(X^{'}X_i - 1)$	

L'intérêt du passage par un espace de redescription des attributs x_i, c'est-à-dire de transformer par des fonctions mathématiques l'espace d'entrée χ de dimension p en un autre espace ayant une dimension p' qui est supérieure à p ou éventuellement infinie, est d'augmenter la probabilité de pouvoir trouver des hyperplans qui isolent les classes l'une de l'autre par une ligne linéaire. À cet effet, le défi est donc de déterminer quelles dimensions supplémentaires nous pouvons utiliser. Ce qui revient à dire quelle fonction kernel du tableau 2.2 permettrait une séparation adéquate des classes.

En plus de l'hyperparamètre C de la méthode SVM, la fonction noyau est aussi un autre hyperparamètre à ajuster par l'utilisateur. Le choix de la fonction noyau à utiliser pour le classifieur SVM ainsi que l'ajustement de ses paramètres est pénible sans assistance informatique. Pour cela, nous faisons recours à notre algorithme d'analyse d'images détaillé dans le chapitre 3 pour tester les fonctions noyau du tableau 2.2 afin d'en sélectionner celle qui fonctionne le mieux en donnant un taux de reconnaissance élevé.

La figure 2-11 résume le fonctionnement des séparateurs à vastes marges et montre le rôle des fonctions noyau. Lors de l'apprentissage, les exemples critiques sont retenus pour définir la frontière de décision. Quand une nouvelle image de visage est présentée au système, celle-là est comparée aux exemples critiques en exploitant l'une des fonctions noyaux. Comme la méthode SVM appartient à la famille des méthodes connexionnistes son fonctionnement est un peu similaire à un réseau de neurones. Alors, la sortie est calculée en faisant une somme pondérée de ces comparaisons, où les poids α_i et b sont déterminés lors de l'apprentissage.

Les classifieurs SVM sont lents à entraîner, mais ils sont rapides lors de la prévision.

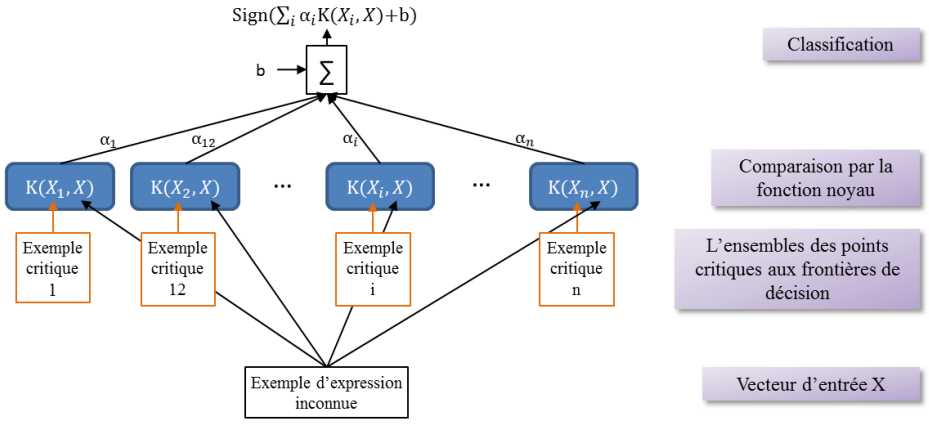


FIGURE 2-11 – Exemple de classification des images en utilisant la fonction noyau par SVM.

Car, une fois le classifieur est formé, l'algorithme d'apprentissage se débarrasse de la plupart des exemples d'apprentissage, et se base que sur l'utilisation des exemples critiques, qui déterminent la forme d'hyperplan séparateur, pour classifier les exemples du test.

2.3.3.2 Cas de classes multiples

La méthode SVM, originalement développée par Vapnik [156], est une solution destinée à des problèmes de classification binaire, où les classes ne prennent que deux valeurs soit +1 (Resp. vrai) ou -1 (Resp. faux). Cependant, les tâches de reconnaissance du monde réel ont plus de deux classes, y compris notre cas d'étude dans laquelle on classifie sept classes d'expressions. Pour cela, nous allons examiner, dans cette section quelques techniques de réorganisation de données d'entraînement en vue d'aboutir à une classification multi-classe par la méthode SVM.

Mis à part la méthode SVM, la plupart des méthodes de classification que nous avons examinées dans ce chapitre 1 pour l'analyse de données, génèrent un seul classifieur même si nous exploitons des données ayant sept classes. Pour classifier sept classes d'expression, le classifieur binaire de SVM peut être transformé en classifieur multi-classe à l'aide de la technique "un contre tous" [170] ou la technique de classification par paire [171].

La figure 2-12 illustre l'idée de la technique un contre tous. En utilisant cette technique, on construit sept classifieurs binaires, chacun étant entraîné pour séparer une seule classe du

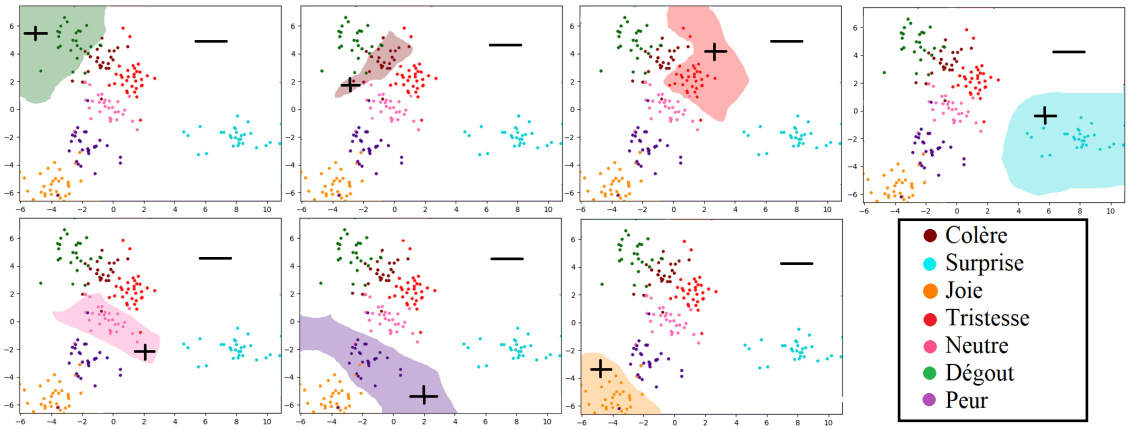


FIGURE 2-12 – Exemple de classification SVM par la méthode un-contre-tous, montant les frontières de décision de SVM pour les sept classifieurs binaires d'expression. Chaque classifieur trouve une frontière séparant points d'une seule classe et les points des autres classes. Le singe (+) indique les exemples positifs et le signe (-) indique les exemples négatifs

reste des classes. Ensuite, pour identifier l'expression dans une image, on teste les classifieurs des sept expressions du visage. Ainsi, on détermine la classe d'expression selon le classifieur qui atteint un taux de reconnaissance maximal.

La technique de classification par paire, au contraire, consiste à former un classifieur pour chaque paire de classes possible. En effet, pour un nombre de classes égale à C, on obtient $\frac{(C-1)C}{2}$ classifieurs binaires. Ce nombre de classifieurs est généralement supérieur au nombre de classifieurs générés par la technique "une classe contre tous". Par exemple, lorsque C vaut 7 qui est le nombre des classes d'expression, nous formerons 21 classifieurs binaires, comme présenté dans la figure 2-13.

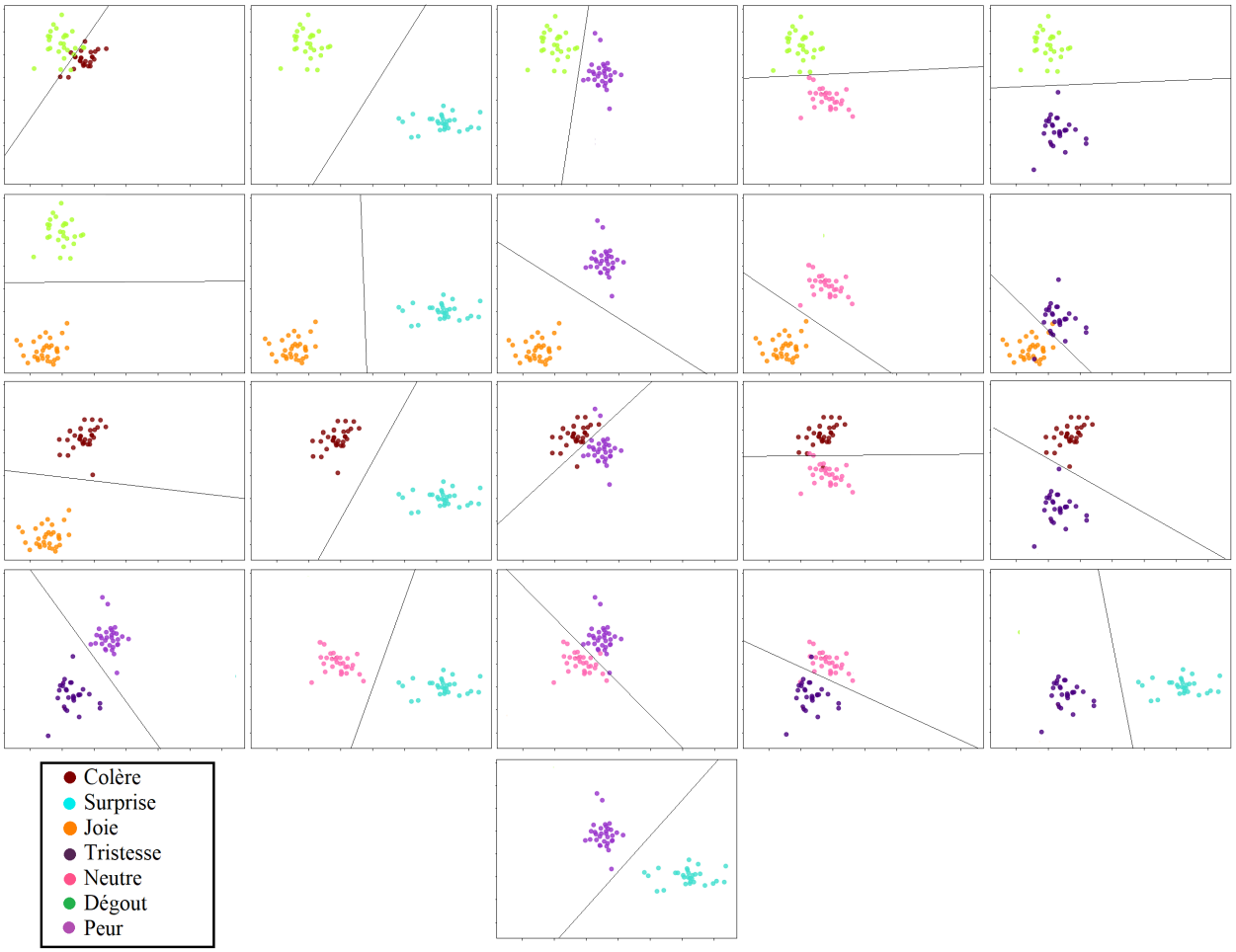


FIGURE 2-13 – Exemple de la technique de classification par paire de SVM. 21 classifieur sont construites à partir des paires combinaisons des expressions. Chaque classifieur cherche une frontière linéaire entre deux classes.

Dans ce cas nous analysons l'image du visage par 21 classifieurs au niveau de la deuxième technique tandis qu'au niveau de la première 7 classifieurs sont utilisés pour déterminer l'expression du visage. Ceci peut ne pas influencer la performance en termes de taux de reconnaissance, pourtant plus du temps est consommé lorsqu'on analyse l'image par 21 classifieurs pour identifier l'expression. Parce que pour classer un nouvel exemple, nous l'analysons en exploitant tous les 21 classificateurs, puis nous sélectionnons l'étiquette qui revient le plus souvent. En d'autres termes, chaque classificateur vote pour l'une des sept classes et nous déclarons que le gagnant est la classe avec le plus de votes.

2.3.4 La méthode du k-plus proches voisins (k-ppv)

L'algorithme du k-plus [154] proches voisins consistent à supposer qu'un exemple requête X_r dont on cherche la classe est similaire à ses voisins. En effet, chaque exemple dans l'ensemble d'entraînement est représenté dans un espace de $\Re^p$ comme un point X_i, où p dépend du nombre d'attributs extraits de l'image. À travers le calcul de la distance entre $X_r \in \Re^P$ et les exemples $X_i \in \Omega$ en utilisant l'équation (2.39), on cherche le nombre k des exemples X_i les plus proches de l'exemple requête $X_r \in \Re^P$. Cela est réalisé en cherchant les k exemples donnant une distance $d^L(X_i, X_r)$ minimale.

$$d^L(X_i, X_r) = (\sum_{j=1}^{p} |x_{i,j} - x_{r,j}|^L)^{\frac{1}{L}} \tag{2.39}$$

P le nombre des exemples dans les données Ω, L determine le type de la distance : lorsque L=1, la distance est celle de Manhattan, si L=2 la distance est Euclidienne.

L'exemple requête X_r est classifié donc en prenant en considération la classe qui reçoit plus de votes de la part des k plus proches voisins. Le choix de l'hyperparamètre k et la métrique de calcul de la distance entre les X_i et X_r n'est plus arbitraire, pour cela l'ajustement de ces deux variables sur la base de données est nécessaire afin d'atteindre une reconnaissance élevée par l'algorithme du k-plus proches voisins. Le classifieur k-plus proches voisins est rapide à former, car il enregistre généralement chaque échantillon d'entraînement. En revanche, la prédiction est lente, car l'algorithme doit rechercher les voisins les plus proches de l'échantillon que nous voulons classer en calculant la distance, en utilisant l'équation (2.39), entre cet échantillon et tous les exemples de données d'apprentissage.

2.3.5 La méthode d'arbre binaire pour la classification des expressions

En partant du principe qu'il existe des exemples qui ont des valeurs d'attributs similaires, mais qui peuvent appartenir à des classes différentes, on peut subdiviser l'ensemble d'apprentissage $\Omega_{m,p}$ en deux sous-ensembles tout en choisissant les valeurs d'attributs qui peuvent mener à la construction de l'arbre binaire (AB) [169].

Dans la figure 2-14, la construction de l'arbre binaire commence donc par la création d'un nœud racine x_{484} qui divise les exemples de données en deux sous-ensembles. Ensuite, on continue par la création progressive des branches en ajoutant des nœuds à l'arbre. Tout cela, pour pouvoir trouver à la fin les feuilles de l'arbre qui indiquent la classe d'expression relatives à chacune des branches. À chaque étage et sur chaque nœud de l'arbre, un attribut dit "attribut parfait" est sélectionné seulement s'il aboutit à une bonne catégorisation des données, compte tenu des attributs qui le précèdent. Cela peut être réalisé grâce au calcul de l'entropie H et le gain d'information g en cherchant les attributs parfaits conduisant à la construction des nœuds de l'arbre binaire.

L'entropie H définit la quantité de l'hétérogénéité dans les données Ω. Si l'entropie H est élevée cela indique qu'il est nécessaire créer des nœuds permettant la répartition de données en des groupes homogènes. L'entropie des données s'écrit :

$$H(x) = -\sum_{i=1}^{7} p_i \log_2 p_i \tag{2.40}$$

p_i, pour $i = \{1, \ldots, 7\}$, est le pourcentage des exemples dans chaque classe i.

En utilisant le gain d'information g, indiqué dans l'équation (2.41), on peut sélectionner l'attribut qui partitionne au mieux les données tout en réduisant l'hétérogénéité des données. Autrement dit, l'attribut qui procure un gain d'information maximal par rapport aux autres attributs. Soit donc $H(x_j)$ l'entropie du nœud parent, $x_j \in \Re^N$ dont N est le nombre des exemples de données après partitionnement selon $x_{(j,k)}$, et $H(x_j|x_{(j,k)})$ l'entropie des nœuds fils conditionnée par la valeur de division candidate $x(j,k) \in x_j$, qu'on peut nommer seuil de division t :

$$g(x_j, t) = H(x_j) - H(x_j|t) \tag{2.41}$$

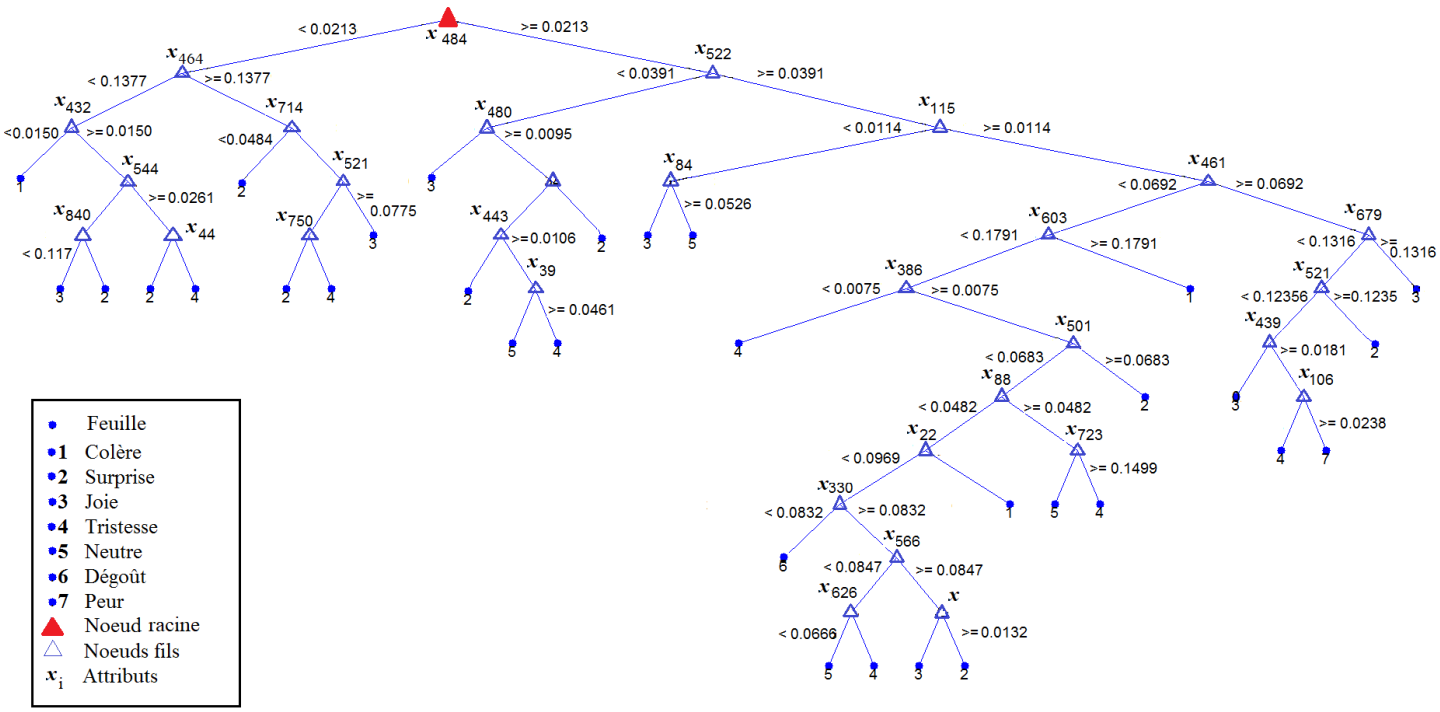


FIGURE 2-14 – Construction de l'arbre binaire en utilisant les données d'apprentissage Ω obtenues après la description des images par le descripteur de HOG. Le nœud racine est obtenu après la sélection de la valeur de l'attribut $x_{484} \in \Omega$ qui maximise le gain d'information après la division

$$H(x_j|t) = -\sum_{i=1}^{7} p_i(x_j \leq t) \log_2 \left(p_i(x_j \leq t)\right) - \sum_{i=1}^{7} p_i(x_j > t) \log_2 \left(p_i(x_j > t)\right) \tag{2.42}$$

$p_i(x_j \leq t)$ et $p_i(x_j > t)$ représentent le pourcentage des exemples dans chaque classe i après la division de données selon la valeur candidate testée $t = x(j,k) \in x_j$

Par conséquent, chercher parmi toutes les valeurs candidates de $t = x(j,k) \in x_j$ celui qui possède le grand gain d'information revient donc à trouver celui qui minimise l'entropie conditionnée $H(x_j|x_{(j,k)})$. Autrement dit, trouver l'attribut $x(j,k) \in x_j$ d'indice $k^* \in \{1, 2, ..., N\}$ qui donne :

$$k^* = \underset{k=\{1,..,N\}}{\arg\min}\ H(x_j|x_{(j,k)}) \tag{2.43}$$

Le coût de calcul associé à la création de l'arbre est $\mathcal{O}(d\ell \log \ell)$ où d est le nombre d'attributs

qui construisent l'arbre [4] et ℓ est le nombre des nœuds dans l'arbre binaire, est relativement faible, aussi bien son coût d'utilisation qui est encore plus bas : $\mathcal{O}(\log \ell)$. Ce qui est important pour l'entraînement, car l'utilisation de l'arbre binaire formé pour classifier un exemple requête X_r doit être aussi rapide que possible. Cela suffit pour rendre les arbres de décision utilisables pour la classification.

2.4 Conclusion

Dans ce chapitre, nous avons posé le cadre théorique des méthodes de description du visage humain et de classification que nous avons choisies pour réaliser l'analyse d'images d'émotions.

Premièrement, nous avons montré qu'on peut représenter l'image du visage par des vecteurs de caractéristiques décrivant la forme par le descripteur HOG, la texture par le descripteur LBP et les contours du visage par le filtre de Gabor. Grace à la notion du vecteur de caractéristiques, nous avons construit les données d'apprentissage à travers le stockage des vecteurs des caractéristiques dans un tableau de données.

Deuxièmement, nous avons montré qu'on peut analyser le tableau de données d'expressions par des méthodes de classification différentes. Cela en faisant une redescription de données d'apprentissage par les fonctions mathématiques des méthodes de classification, détaillées dans la section 2.3. Ces méthodes de classification prennent en entrée les attributs de données pour chercher des frontières de décision séparant les classes d'expression. Nous avons donc choisi d'exploiter les méthodes de classification suivantes :

- La méthode de Fisher réduit le nombre des attributs en appliquant une transformation de données par un calcul statistiques (Voir section 2.3.1) qui maximise la variance interclasse et minimise la variance intraclasse. Cela est réalisé dans le but de trouver une séparation linéaire entre les classes de données.
- La méthode du séparateur à vaste marge, contrairement à la méthode de Fisher, elle augmente les dimensions de données pour chercher des hyperplans séparateurs linéaires. L'augmentation de dimensions se fait en appliquant une transformation sur les attributs du tableau de données par une fonction noyau comme le noyau polynomial et Gaussien.
- La méthode du k-plus proches voisins n'applique aucune transformation aux données d'ap-

4. le nombre d n'est pas forcément égale au nombre des attributs dans la base de données. Un attribut peut être utilisé dans plusieurs nœuds

prentissage mais la classification des exemples se fait en cherchant des voisins majoritaires qui appartient à la même classe.

- La méthode de classification Naïve Bayésienne produit un classifieur de type probabiliste. Le classifieur utilise le théorème de Bayes pour chercher la probabilité conditionnelle de la classe étant donné les exemples, mais en supposant que les attributs sont indépendants.
- La méthode d'arbre binaire introduit l'idée de hiérarchie sur les attributs, c.-à-d. sélectionner les attributs qui renvoient le gain d'informations (Voir section 2.3.5) le plus élevé pour construire les nœuds et les branches de l'arbre et puis trouver les feuilles qui indiquent la classe des expressions.

Dans le chapitre suivant nous effectuons un ajustement des paramètres et hyperparamètres soulignés dans les sections 2.2 et 2.3, respectivement. En particulier, nous proposons un algorithme pour couvrir cette étape d'ajustement ainsi qu'une étape de comparaison de performance des descripteurs d'images et des classifieurs vers une analyse entièrement automatisée des images.

Chapitre 3

Recherche du descripteur et du classifieur d'expression : vers une analyse d'images entièrement automatisée

Introduction

Comme les descripteurs d'image et les méthodes de classification disposent des paramètres et hyperparamètres qui nécessitent un réglage à une valeur optimale, nous voulons essayer de nombreuses valeurs de ces paramètres pour rechercher la combinaison des valeurs qui donne la meilleure performance de reconnaissance. Cependant, nous ne pouvons pas rechercher ces paramètres manuellement.

Ainsi, pour automatiser ce processus de recherche, nous avons besoin d'un algorithme qui fonctionne en deux étapes. La première étape automatise la sélection des combinaisons des valeurs de paramètres et hyperparamètres, en choisissant les valeurs optimales pour configurer les combinaisons descripteur-classifieur [1] qui se construisent lors de l'analyse des images. Ensuite, la deuxième étape opère pour évaluer les modèles d'analyse générés par l'algorithme d'analyse des images afin d'en sélectionner le plus optimal. On peut alors choisir le modèle

1. Dans la suite du manuscrit nous faisons référence à chacune des combinaisons de paire descripteur-classifieur en tant que "modèle d'analyse"

qui aboutit à un taux de reconnaissance élevé sur nos données d'apprentissage.

Par ailleurs, on dit qu'un modèle d'analyse apprend et se généralise sur la base de données s'il reconnaît les exemples qui ne font pas partie de l'ensemble d'entraînement. Après l'apprentissage, si on atteint un taux de reconnaissance élevé, dépassant 90%, le modèle peut être exploité pour qu'il identifie les expressions dans des images du monde réel. Cependant, il n'est pas aussi simple d'atteindre un tel taux de reconnaissance. Car un mauvais choix de la méthode de validation du classifieur et de la métrique d'évaluation du taux de reconnaissance de l'expression, en plus d'une mauvaise configuration des paramètres, peut donner une généralisation qui atteint 90% mais moins capable d'identifier le vrai positif de l'expression. Pour cela, nous commençons par examiner dans les sections 3.1 les techniques d'évaluation de performance de reconnaissance reposant sur des techniques de validation des classifieurs et des métriques de mesure de performance de reconnaissance bien choisis. Dans la section 3.2, nous décrivons en détail la procédure de la recherche du modèle d'analyse optimal qui repose sur une étape importante de recherche exhaustive des valeurs de paramètres et hyperparamètres optimaux.

3.1 Techniques d'évaluation

L'ajustement des paramètres et des hyperparamètres des descripteurs d'image et des méthodes de classification préselectionnées exigent de mettre en œuvre une démarche d'évaluation des performances en termes de taux de reconnaissance. Dans ce cas, deux facteurs devront être pris en compte puisqu'ils influencent le résultat de l'analyse des données. Dans la section 3.1.1, nous abordons le premier facteur qui est lié au choix de la méthode de validation du modèle d'analyse. Tandis que dans la section 3.1.2, nous adressons le deuxième facteur qui concerne le choix de la métrique d'estimation du taux de reconnaissance.

3.1.1 Techniques de validation des modèles d'analyse

Pour pouvoir tester la capacité de généralisation des modèles d'analyse que nous avons utilisés lors de l'analyse des images, la répartition des données devrait être effectuée de manière à pouvoir utiliser le maximum de données pour la phase d'entraînement, aussi bien pour la phase de validation et la phase du test, en évitant aussi la sélection des mêmes exemples d'images dans les trois phases. Dans les sous-sections suivantes, nous examinons quelques techniques de validation en appuyant sur la technique choisie pour réaliser l'analyse

de données.

La technique la plus classique pour échantillonner les données, c'est d'utiliser la méthode "hold-out" que nous avons illustrée dans la figure 3-1. En effet, après l'extraction de la forme du visage de données étiquetées via le descripteur d'image, cette technique divise les données Ω en trois sous-ensembles.

- Le premier sous-ensemble concerne les données d'entraînement qui constitue typiquement 60% de données Ω et à partir duquel l'algorithme d'apprentissage entraîne le classifieur de l'expression. Durant l'entraînement l'algorithme d'apprentissage essaye de trouver une relation entre les exemples X_i avec leurs étiquettes y_i en recherchant les hypothèses $h(X_i)$ qui minimisent l'erreur $e(h(X))$.

$$e(h(X)) = \sqrt{\frac{1}{N'}\sum_{i=1}^{N'}(y_i - h(X_i))^2} \tag{3.1}$$

  N' est le nombre des exemples dans les données d'entraînement.

- Le deuxième sous-ensemble représente les données de validation ayant une quantité typique d'exemples égale à 20% de données d'apprentissage. Généralement, le rôle de l'ensemble de validation est l'ajustement de la configuration des paramètres et hyperparamètres du modèle d'analyse étudié. Cela se fait en répétant l'entraînement pour plusieurs valeurs des paramètres et hyperparamètres en enregistrant la performance de reconnaissance pour chaque valeur et puis en sélectionnant les valeurs qui rendent à la fois le descripteur et le classifieur performants.

- le troisième sous-ensemble est celui du test. Ce sous-ensemble permet d'évaluer la capacité du modèle d'analyse de classifier des nouveaux exemples que le classifieur n'a jamais vu auparavant - c.-à-d. les exemples qu'il n'a pas utilisés pendant l'entraînement et la validation.

Malgré la simplicité de mesure du taux de reconnaissance de l'expression en un seul cycle d'apprentissage [2], cette méthode présente cependant l'inconvénient de ne pas exploiter toutes les données pendant l'entraînement. La validation hold-out peut mener un apprentissage de faible qualité si on ne garde que 60% de données. Cette technique de validation peut aussi conduire à une fausse estimation du taux de reconnaissance de l'expression si l'on utilise que

2. Dans le cas de la validation par la méthode hold-out, un cycle d'apprentissage désigne le passage par trois étapes : l'entraînement, la validation et le test du classifieur.

peu de données, comme par exemple 20% de données, pour la validation et le test. Il est vrai qu'au niveau de la validation hold-out, si nous voulons entraîner le modèle d'analyse, nous devons diviser nos données Ω en trois parties, ne laissant que 60% pour l'entraînement du classifieur, mais en réalité, on peut faire mieux. Le point c'est que, même si nous devons encore retenir certains exemples de nos données Ω afin de créer un ensemble du test, nous pouvons éviter de créer un ensemble de validation. La technique pour éviter cela est d'utiliser la validation croisée.

L'idée de la validation croisée est de répéter dans une boucle l'entraînement et le test plusieurs fois. Mais au lieu d'essayer différents ensembles de paramètres pour évaluer le modèle d'analyse. L'objectif sera d'utiliser la boucle de base pour évaluer dans quelle mesure le modèle d'analyse répondra aux nouvelles données, mais sans faire usage d'un ensemble de validation. Cela permet d'utiliser toutes nos données pour l'entraînement et le test, mais pas toutes en même temps.

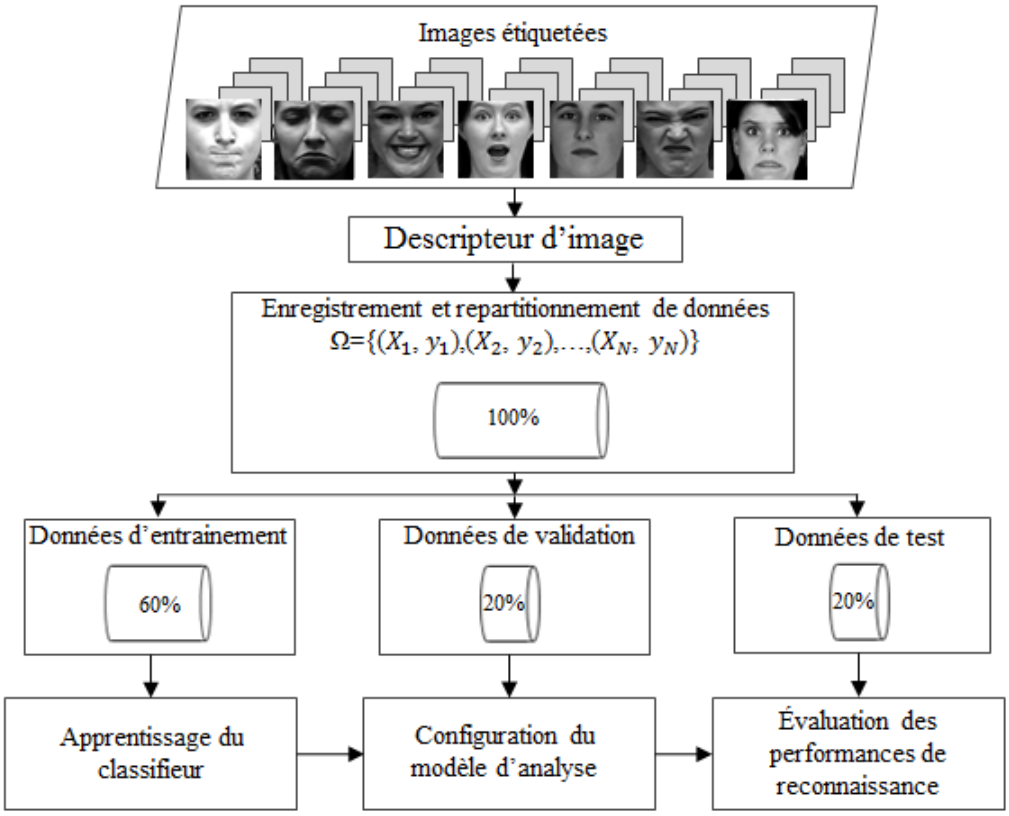


FIGURE 3-1 – Partitionnement de données dans la validation par la technique Hold-out.

3.1.1.1 Généralisation par la validation croisée à n-plis

La validation croisée à n plis (n-fold cross validation en anglais) peut remédier aux problèmes de la méthode hold-out, et ce en faisant apparaître chacun des exemples de la base de données tantôt dans le sous-ensemble d'entraînement et tantôt dans le sous-ensemble de test. L'idée de la validation croisée à n plis, que nous avons exposée dans la figure 3-2, est de diviser les données d'apprentissage Ω en n sous-ensembles contenant un nombre d'exemples égales. Ensuite, retenir (n-1) sous-ensembles pour effectuer l'apprentissage et un seul sous-ensemble

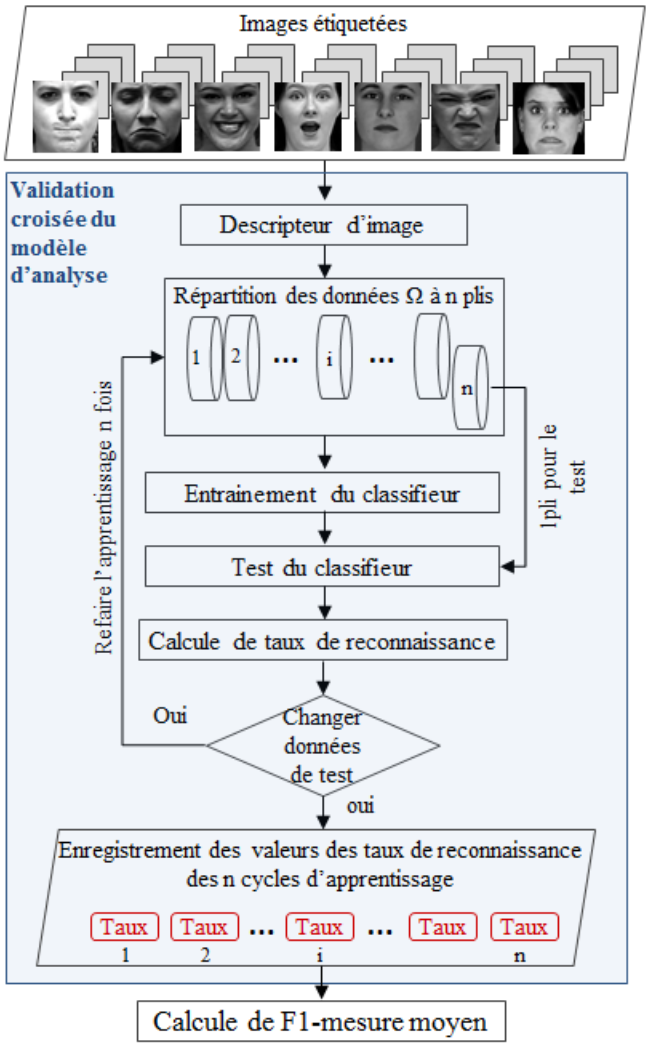


FIGURE 3-2 – Principe de la validation croisée à n plis pour évaluer la performance des descripteurs et du classifieur.

pour réaliser le test. En répétant le cycle d'apprentissage [3] n fois tout en faisant varier le sous-ensemble de test. On peut donc mesurer l'erreur empirique relative à l'entraînement :

$$\hat{e}(h(X)) = \frac{1}{n}\sum_{i=1}^{n} e_i(h(X)) \tag{3.2}$$

$e_i(h(X))$ *est l'erreur* $e(h(X))$ *calculée par l'équation* (3.1) *pour les n répartition de données d'entraînement possible.*

À ce niveau, en plus de l'erreur empirique d'autres métriques d'évaluation de performance peuvent être utilisées comme par exemple la matrice de contingence, la précision (precision en anglais), le rappel (recall en anglais) et le taux de reconnaissance de l'expression (F1-mesure), discutés dans la section 3.1.2.

En choisissant un nombre de plis n compris entre 5 et 10, la validation croisée n-plis aura l'avantage d'avoir une mesure d'erreurs un peu plus précise sur l'ensemble de données puisque l'apprentissage est réalisé sur un nombre d'exemples élevé si l'on compare avec la validation hold-out.

Par ailleurs, le point gênant dans la validation croisée n plis est qu'avec des hypothèses h(X) estimées sur n sous-ensembles d'apprentissage différents, on peut quand même, dans certains cas, générer des hypothèses différentes les unes des autres. En conséquence, une variance élevée dans l'espace des hypothèses mènera à un apprentissage sans valeur. Pour éviter ceci, le mieux est de réaliser l'apprentissage et le test sur tous les exemples de données en utilisant la technique de la validation croisée leave-one-out.

3.1.1.2 Généralisation par la validation croisée Leave-One-Out (LOO)

La technique leave-one-out est un cas particulier de celle de n-plis car le nombre de plis n est égale au nombre total des exemples N dans les données utilisées Ω. Dans ce cas, on répète l'apprentissage N-1 fois tout en retirant chaque fois un seul exemple pour le test durant chaque cycle d'apprentissage jusqu'à ce qu'on teste tous les exemples de données Ω. Ceci produit des hypothèses h(X) moins variables.

Même si dans la validation croisée leave-one-out l'apprentissage est répété plusieurs fois par rapport à la validation croisée 10-plis cela augmente le temps d'analyse de données, et ce,

3. Dans la validation croisée, un cycle d'apprentissage se réalise sur deux étapes : l'entraînement et le test seulement.

relativement aux méthodes de classification et à la quantité de données utilisées. Cependant, cette technique permet de gagner en exactitude du calcul des hypothèses h(X).

Suite à notre présentation des techniques de validation existantes, nous avons décidé d'exécuter l'algorithme d'analyse de données en faisant appel aux techniques de validation 10-plis et leave-one-out. Ensuite d'enregistrer les résultats d'analyse pour observer l'effet de varier les techniques de validation croisée sur le résultat de reconnaissance mesuré pour chacun des modèles d'analyse générés par l'algorithme d'analyse de données.

3.1.2 Métriques d'évaluation

Jusqu'ici l'évaluation des méthodes d'analyse d'image est discutée en prenant en compte l'estimation de l'erreur empirique $\hat{e}(h(X))$ calculée pendant l'apprentissage du classifieur. Toutefois, lorsqu'il s'agit de prendre une décision, il est intéressant de prendre en compte d'autres métriques d'évaluation en plus de celle de l'estimation du taux de l'erreur empirique.

3.1.2.1 Tableau de contingence

Comme nous l'avons mentionné dans l'introduction de ce chapitre, avec l'usage de l'algorithme d'analyse d'images on cherche à ajuster et comparer la performance des descripteurs et des classifieurs sur des images d'expressions universelles d'émotion. Dans ce sens, l'ajustement a pour but de faire varier les valeurs des paramètres des descripteurs et les hyperparamètres des classifieurs, les tester et mesurer leur performance pour ensuite prendre une décision et qualifier ceux qui donnent un taux de reconnaissance élevé. Pour ce faire, l'évaluation commence par la construction du tableau de contingence 3.1 pour toutes les classes de données. Les lignes L_i du tableau correspondent aux classes références, tandis que les colonnes C_i correspondent aux résultats estimés par le classifieur. Chaque cellule du tableau représente le nombre des exemples des classes de référence qui sont estimées comme appartenant aux

TABLE 3.1 – La tableau de contingence montrant le nombre des exemples correctement et mal classifiés

		Classes estimées			
		C_1	C_2	...	C_7
Classes de reference	L_1	$\xi_{1,1}$	$a_{1,2}$	...	$a_{1,7}$
	L_2	$a'_{2,1}$	$\xi_{2,2}$		$a_{2,7}$
	⋮	⋮		⋱	
	L_7	$a'_{7,1}$	$a'_{7,2}$	...	$\xi_{7,7}$

classes C_i.

3.1.2.2 Mesure de la précision et du rappel

Pour simplifier la lecture des résultats du tableau de contingence, on calcule d'une part la précision du classifieur qui mesure la probabilité que la classe existe lorsque le résultat du test du classifieur est positif. Une qualité importante de la précision, lorsqu'elle est inférieur à 1, elle indique le pourcentage des exemples identifiés comme des vrais positifs, mais qui sont en réalité des exemples d'autres classes.

$$Précision = \frac{\sum_{i=1}^{7} \xi_{i,i}}{\sum_{i=1}^{7} \xi_{i,i} + \sum_{i=1}^{7} \sum_{j=1,j<i}^{7-1} a'_{i,j}} \tag{3.3}$$

$\xi_{i,i}$ le nombre de bonne classification de la classe i, $a_{i,j}$ le nombre du faux négatif, $a'_{i,j}$, le nombre du faux positif

D'autre part, le rappel indique le pourcentage des exemples positifs correctement reconnus. Lorsque le rappel est inférieur à un, cela signifie que le classifieur n'a pas bien classifié l'expression du visage.

$$Rappel = \frac{\sum_{i=1}^{7} \xi_{i,i}}{\sum_{i=1}^{7} \xi_{i,i} + \sum_{i=1}^{7-1} \sum_{j>i}^{7} a_{i,j}} \tag{3.4}$$

3.1.2.3 Mesure du taux de reconnaissance : F1-mesure

En réalité, obtenir un système de reconnaissance performant signifie pouvoir maximiser la précision et le rappel à la fois. Avec le calcul de la moyenne harmonique de la précision et le rappel, on peut obtenir une métrique d'évaluation appelée F1-mesure. L'avantage d'utiliser cette métrique réside dans le fait qu'une augmentation (Resp. diminution) au niveau de la valeur de la précision et le rappel, tout en étant proches entre eux, engendre aussi bien une augmentation (Resp. diminution) de la valeur du métrique F1-mesure [172]. Cela, permet de dire que lorsque F1-mesure est au maximum le rappel et la précision sont à la fois élevés.

$$F1 - mesure = \frac{2.rappel.précision}{recall + précision} \tag{3.5}$$

Le choix de la métrique F1-mesure est fait dans l'objectif de fournir une mesure bien précise lorsqu'on a des données multi-classes dont le nombre d'exemples dans les classes n'est pas équilibré dans la base de données.

3.2 Description de l'algorithme proposé pour l'analyse des images

L'illustration que nous avons établie dans la figure 3-3 offre une vue générale de la structure de l'algorithme d'analyse des images. A ce niveau, nous montrons les principales composantes qui interagissent dans cette structure, y compris le bloc contenant l'algorithme que nous avons proposé pour gérer l'analyse des images depuis le recueil des entrées, constituées d'images étiquetées, des descripteurs et des classifieurs, jusqu'à la prise de la décision finale. Les autres composantes telles que les descripteurs et les classifieurs sont déjà adressées dans le chapitre 3, dans lequel nous avons vu que la reconnaissance de l'expression du visage est une tâche composite qui se fait sur deux étapes : une **étape de description du visage** et une autre **étape d'apprentissage des classifieurs**. Dans cette optique, nous avons bâti l'algorithme d'analyse d'images pour être capable d'utiliser, de tester, de configurer et d'évaluer une variété de descripteurs et des méthodes de classification à la fois. En tant qu'utilisateur et avant le démarrage de l'algorithme d'analyse de données, on commence tout d'abord par importer **les images étiquetées** selon l'expression du visage. Ensuite, on sélectionne dans les étapes 1 et 2 de la figure 3-3 les méthodes avec lesquelles on souhaite réaliser l'analyse des

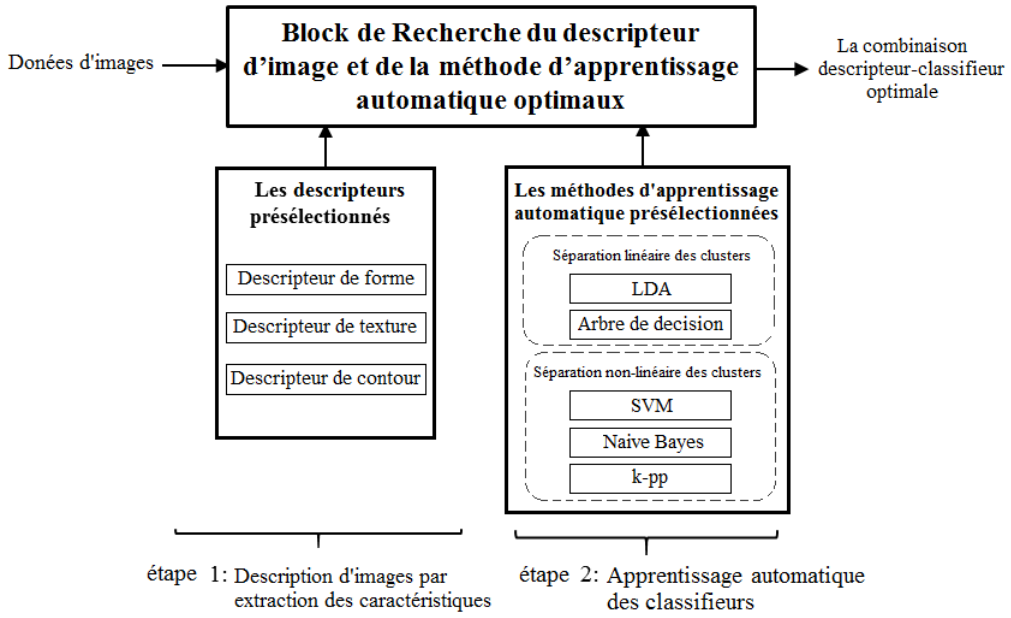


FIGURE 3-3 – Schéma simplifié montrant les composantes de l'algorithme informatique proposé pour l'analyse de données d'apprentissage

images. Suite à cela, le **bloc** crée un ensemble de modèles d'analyse en construisant des combinaisons de paire descripteur-classifieur. En conséquence, l'objectif du bloc est de rechercher parmi ces modèles d'analyse celui qui apprend mieux sur les données d'apprentissage.

3.3 Le bloc responsable de la recherche du modèle d'analyse optimal

Pour simplifier la compréhension du fonctionnement de l'algorithme du bloc, nous avons tracé l'organigramme de la figure3-4. Cet organigramme représente l'enchaînement des opé-

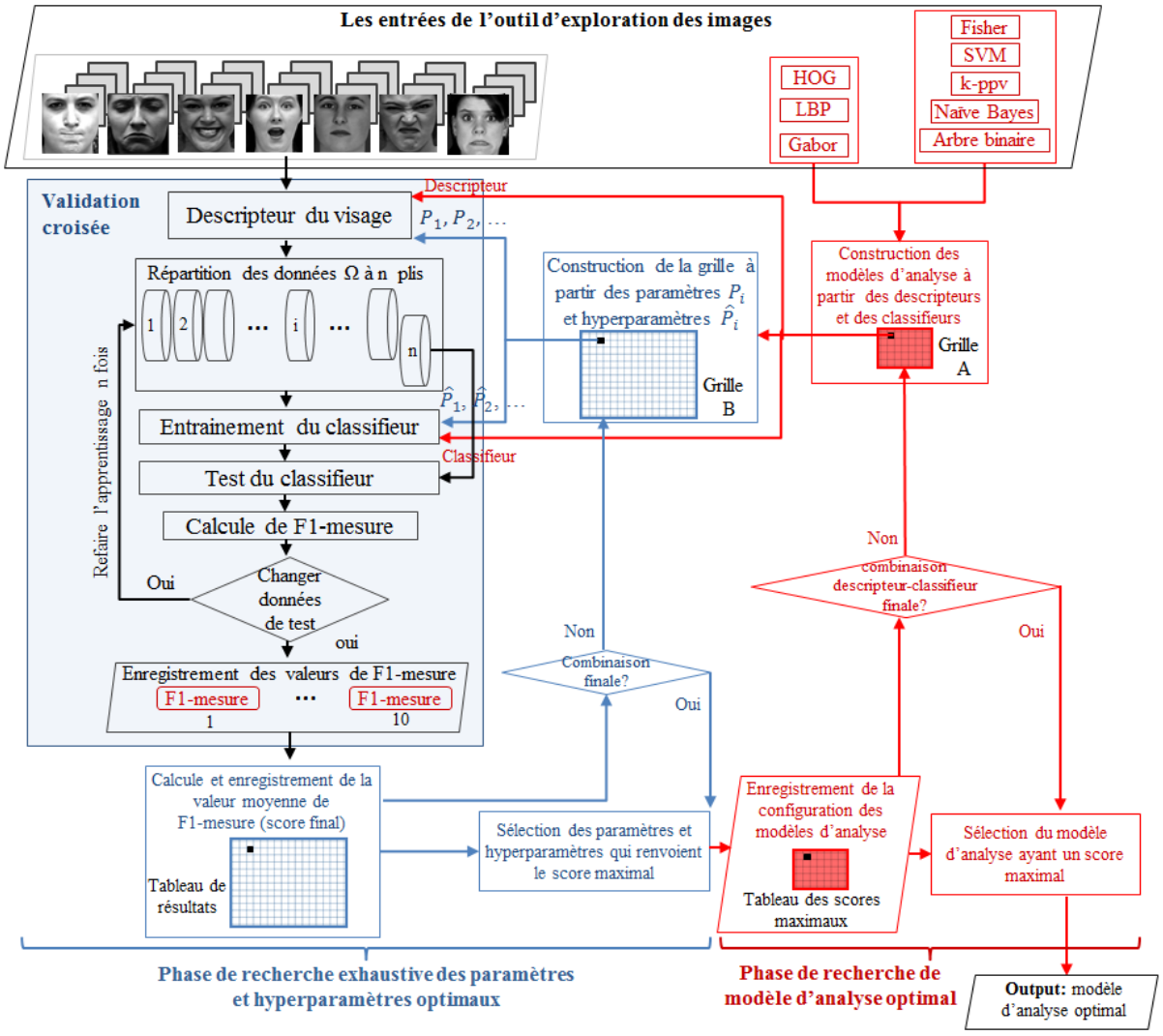


FIGURE 3-4 – Organigramme de l'algorithme proposé pour automatiser l'analyse des images.

rations et des décisions effectuées lors de l'analyse des images étiquetées, qui sont réalisées sur deux phases. La première en bleu opère pour rechercher les paramètres et hyperparamètres optimaux de chacun des modèles d'analyse. La deuxième, en rouge, consiste à rechercher le modèle d'analyse qui donne un score de reconnaissance maximal et un temps de classification minimal. Cette section (3.3) est organisée en suivant la structure de l'organigramme de la figure3-4, dont les sous-sections décrivent en détail l'ensemble des opérations effectuées dans les deux phases de l'organigramme.

3.3.1 Initialisation de variables

En premier, nous importons les entrées qui sont : les images regroupées en sept classes d'expression, les descripteurs et les méthodes de classification. Ensuite, l'algorithme initialise lui-même les variables qui nécessitent un ajustement, tels que les paramètres P_i des descripteurs d'image et les hyperparamètres $\hat{P}_i$, et l'intervalle des valeurs $[v_1, v_2, ..., v_k]$ que nous voulons tester pour chaque P_i et $\hat{P}_i$.

3.3.2 Construction des modèles d'analyse

L'algorithme combine chacun des descripteurs avec chacune des méthodes de classification, comme le montre la figure 3-5. Dans cette figure, nous avons pré-sélectionné trois descripteurs et cinq méthodes de classification, ce qui donne au total quinze modèles d'analyse, que l'on peut utiliser pour analyser nos images. La recherche du modèle d'analyse optimal consiste à réaliser un apprentissage de chaque modèle d'analyse en utilisant une boucle qui permet de configurer tout d'abord les modèles d'analyse de la grille A, illustrée dans la fi-

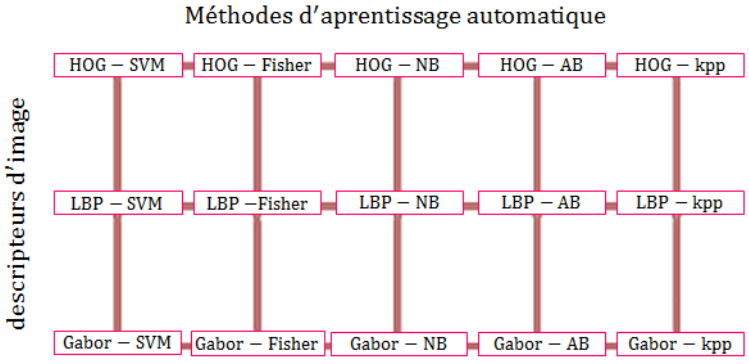


FIGURE 3-5 – Construction de la grille A des modèles d'analyse à l'aide des descripteurs et les méthodes méthode de classification. AB indique l'arbre binaire.

gure 3-5 (voir aussi3-4). Mais, avant toute comparaison chaque modèle d'analyse nécessite un ajustement. Pour cela nous effectuons tout d'abord une recherche exhaustive des valeurs des paramètres et hyperparamètres qui rendent le fonctionnement du modèle d'analyse optimal.

3.3.3 Construction de la grille des paramètres et hyperparamètres

Pour la recherche des valeurs des paramètres et hyperparamètres optimales, nous avons d'abord créé une grille régulière que nous nommons grille B. Cette grille construit des combinaisons à partir de la plage de valeurs des paramètres et hyperparamètres. Chacune des combinaisons est représentée aussi comme un point sur la grille B. Un exemple de grille construite à partir de trois paramètres est illustré sur la figure 3-6. Pour chaque combinaison de trois valeurs, une validation croisée et une mesure de la performance de reconnaissance sont réalisées. Il y a dans ce cas 3x3x3=27 combinaisons à tester et à comparer.

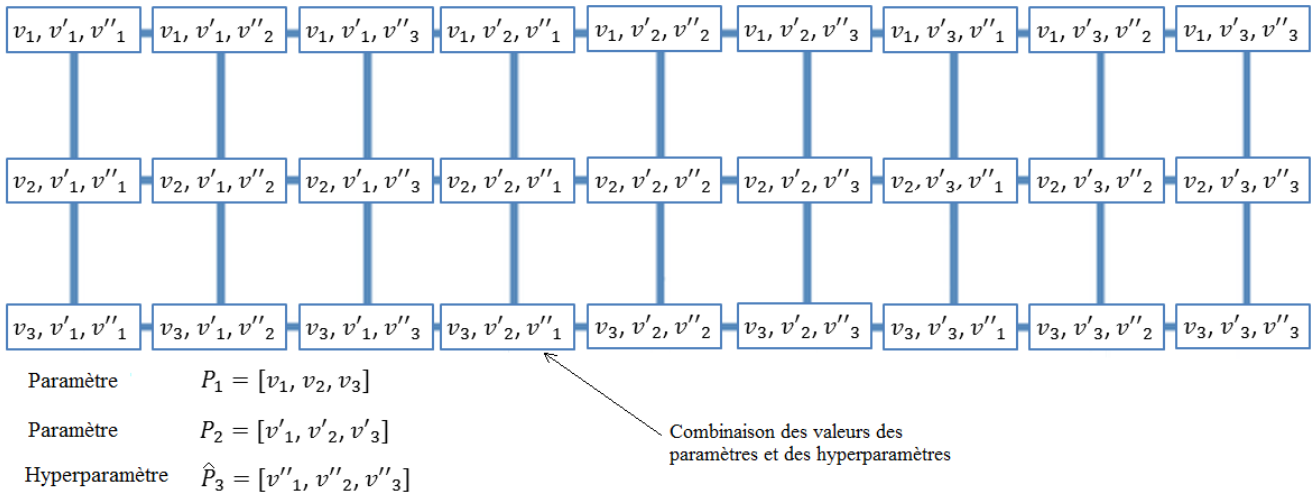


FIGURE 3-6 – Exemple d'une recherche de grille régulière à l'aide de deux trois paramètres, chacun ayant 3 valeurs.

Comme nous ajoutons plus de paramètres et hyperparamètres ou plus de valeurs de paramètres et hyperparamètres, la taille de la grille B augmente de même. Cela signifie l'que le nombre de tests par validation croisée sera élevé aussi, chose qui rend l'analyse de données une étape lente, qui peut prendre des semaines pour être achevée. Par exemple, dans le cas du modèle d'analyse composé du descripteur HOG et du classifieur de Fisher nous avons six paramètres à ajuster. Si nous choisissons de tester 7 valeurs pour chacun des paramètres, l'algorithme construit 7^6 combinaisons différentes. Ainsi, s'il faut une seule seconde pour entraîner le modèle d'analyse à une combinaison de valeur des paramètres, la recherche des

valeurs optimales prendra presque un jour et demi de calcul. Compte tenu du nombre des modèles d'analyse qui doivent être ajustés le temps s'élève jusqu'au trois semaines en utilisant un processeurs Intel Core i7, CPU @ 2.20 GHz et une mémoire RAM de 16 Go.

Au lieu de tester toutes les combinaisons de la grille B et pour réduire le temps d'analyse d'image nous proposons, dans les perspectives (voir section 4.5), de réaliser une sélection arbitraire d'une combinaison de valeurs dans la grille B. Faire, ensuite, le test sur la combinaison sélectionnée et continuer le test des combinaisons voisines seulement lorsque le taux de reconnaissance du model d'analyse augmente à chaque test. Sinon, si le taux de reconnaissance diminue, alors l'algorithme sélectionne arbitrairement une autre combinaison et cherche dans le voisinage de cette combinaison.

3.3.4 Phase de la recherche exhaustive des valeurs optimales des paramètres et hyperparamètres

Dans la phase de recherche exhaustive, nous commençons par produire la grille B. Ensuite, nous exécutons la boucle principale au balayage des combinaisons de paramètres et hyperparamètres dans la grille B. A chaque itération de la boucle, on fixe les paramètres du descripteur et les hyperparamètres de la méthode de classification provenant de la grille A pour créer nos classifieurs d'expressions.

Sur la figure 3-4, nous pouvons voir (en haut à gauche) les images qui composent notre ensemble d'apprentissage ayant les sept classes d'expressions universelles. Après l'extraction du vecteur caractérisant l'appartenance du visage de toutes les images, nous construisons la matrice Ω. Celle-ci est ensuite partitionnée en n plis. Nous fournissons également le nombre de plis à utiliser lors de la validation croisée. Le classifieur étant entraîné et testé en appliquant la validation croisée 10-plis, les valeurs de F1-mesure issues des étapes de validation croisée sont enregistrées puis moyennées pour mesurer le score final pour ce modèle. Ce score est enregistré par la suite dans un tableau de résultats, comme il est indiqué sur la figure 3-4 (en bas à gauche), le tableau a une taille similaire à la taille de la grille B. Cela permet d'enregistrer les scores trouvés lorsqu'on va répéter l'entraînement et l'évaluation par validation croisée pour toutes les valeurs des combinaisons de la grille B. L'enregistrement de résultats s'arrête lorsque toutes les combinaisons de la grille B sont parcourues selon un ordre fixe de gauche à droite et de haut en bas comme illustré sur la figure 3-7.

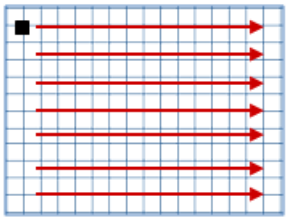


FIGURE 3-7 – l'ordre de sélection des combinaisons des valeurs de paramètres et hyperparamètres dans la grille B. Le point noir représente la première combinaison

Maintenant que toutes les combinaisons de la grille B sont testées et les valeurs du métrique F1-mesure sont enregistrées, nous pouvons rapidement voir quel score a été attribué à chaque combinaison. En conséquence, nous examinons tous les scores enregistrés au cours du processus de recherche pour trouver les valeurs des paramètres et hyperparamètres qui produisent le meilleur résultat.

3.3.5 Recherche du modèle d'analyse optimal

Une fois la phase de la recherche des valeurs optimales des paramètres et hyperparamètres est terminée, la deuxième phase qui est la phase de recherche du modèle d'analyse optimal reprend le relais. À ce niveau, nous faisons appel à une boucle qui permet de parcourir et d'ajuster tous les modèles d'analyse construits dans la grille A.

À chaque itération de la boucle, nous sélectionnons un modèle d'analyse de la grille selon un ordre fixe de gauche à droite et de haut en bas, tout comme dans le cas de la figure 3-7. Ensuite, nous l'ajustons en appliquant le processus de la section 3.3.4. Finalement, nous enregistrons la combinaison de valeurs des paramètres et hyperparamètres qui a aboutie à la meilleure configuration du modèle d'analyse ainsi que le score final qui lui est associé. Ce score est aussi enregistré dans un tableau que nous marquons dans la figure 3-4 "tableau des scores maximaux", qui a la même taille que la grille A.

Après que la boucle parcourt tous les modèles d'analyse, le tableau enregistrant les scores maximaux des modèles d'analyse se remplit. Nous pouvons finalement comparer les scores. Suite à cela, nous pouvons déterminer la sortie de l'algorithme qui renvoie le meilleur modèle d'analyse dans la grille A, et ce, en identifiant le score maximal.

3.4 Résultats expérimentaux de la recherche du modèle d'analyse optimal

Dans cette section, nous comparons la capacité de généralisation des modèles d'analyse construits dans la grille A. Le tableau 3.2 présente les scores maximaux obtenus par chaque modèle d'analyse après leur ajustement en exécutant l'algorithme d'analyse des images une fois en appliquant la validation croisée 10-plis et une autre fois en appliquant la validation croisée Leave-one-out.

TABLE 3.2 – Résultats obtenus lors de la construction du tableau de scores maximaux de la figure 3-4

	Mesure des scores par validation croisée (%)					
	10-plis			leave-one-out		
	HOG	LBP	Gabor	HOG	LBP	Gabor
k-plus proches voisins	85.78	62.67	81.78	74.50	52.90	71.02
Fisher	**96.44**	82.67	92.89	**93.02**	78.77	–
Arbre binaire	73.33	53.33	63.11	70.01	55.06	56.24
Naïf de Bayes	92.00	80.50	89.78	82.14	62.88	77.54
SVM	89.27	79.17	82.79	92.55	77.13	90.62

En ce qui concerne les descripteurs d'images, nous remarquons, d'après le tableau 3.2 que le descripteur de la forme du visage (HOG) est plus performant que le descripteur de la texture (LBP) et le descripteur du contour du visage (Gabor), quel que soit le classifieur auquel ils sont associés. En fait, même si les descripteurs HOG et Gabor atteignent des scores élevés, supérieurs à 90%, lorsqu'ils sont associés au classifieur Fisher, le descripteur Gabor consomme plus de temps et extrait plus d'information spatiales sur les images. Pour cette raison et en plus de calculs coûteux de la technique de validation croisée Leave-one-out, l'algorithme informatique d'analyse d'images n'a pas été en mesure d'accomplir l'apprentissage du classifieur Fisher associé au descripteur de Gabor. Pour cela, le score est manquant dans le tableau 3.2.

D'un autre côté, en observant les lignes du tableau 3.2 et en comparant les scores des classifieurs on trouve que la méthode de Fisher se généralise mieux, avec un taux de 96.44%, sur les données Ω par rapport aux autres classifieurs du tableau 3.2.

3.5 Discussion des résultats

Pour comparer les résultats de reconnaissance des modèles d'analyse que nous avons obtenus par l'algorithme d'analyse avec ceux de la littérature, nous avons dressé le tableau 3.3. Dans ce tableau, nous comparons les scores obtenus par les deux méthodes de validation à savoir : la méthode à 10 plis et la méthode LOO, ainsi que les temps de description et de classification pour différents modèles d'analyse. Dans la littérature, peu de chercheurs passent par l'étape d'analyse des images d'expressions émotionnelles avant d'avoir proposé leur système de reconnaissance d'émotions humaines. Par exemple, Lekdioui et al. [90] ont

TABLE 3.3 – Comparaison des résultats des modèles d'analyses avec les résultats des modèles d'analyses de l'état-de-l'art en exploitant la base d'images CK+.

Modèle d'analyse	approches	score de 10-plis (%)	score de LOO (%)	temps de description (ms)	temps de classification (ms)	hardware
HOG-kppv	MT[a]	85.78	75.50	2.60	133.50	CPU 2.2GHz 16Go RAM
	[173]2013	95.86	–	–	–	CPU 2.2GHz 2Go RAM
HOG-Fisher	MT	96.44	93.02	2.60	0.78	–
HOG-AB	MT	73.33	–	2.60	11.20	–
HOG-NB	MT	92.00	82.14	2.60	2314.00	–
HOG-SVM	MT	89.27	92.55	2.60	700.70	–
	[174]2015	83.7	–	–	–	CPU 3.4GHz 16Go RAM
	[175]2015	94.1	–	24.00	1240.00	–
	[90]2017	90.28	–	196.24	–	CPU 2.4GHz 4Go RAM
	[25]2015	94.1	–	0.24	1.24	–
LBP-kppv	MT	62.67	52.90	0.12	325.30	–
LBP-Fisher	MT	82.67	78.77	0.12	44.50	–
	[6]2009	73.4	–	30.00	–	–
LBP-AB	MT	53.33	–	0.12	10.00	–
LBP-NB	MT	80.50	62.88	0.12	1997.60	–
	[101]2012	82.00	–	0.17	–	CPU 2.4 GHz 2Go RAM
LBP-SVM	MT	79.17	77.13	0.12	5.80	–
	[107]2019	80.2	–	30.00	–	–
	[90]2017	91.33	–	–	–	–
	[87]2015	91.7	–	0.4	0.72	CPU 3.4GHz 16Go RAM
Gabor-kppv	MT	81.78	71.02	500.90	652.50	–
	[176]2012	91.51	–	–	–	CPU 2.66GHz 8Go RAM
Gaobr-Fisher	MT	92.89	–	500.90	1084.40	–
	[177]2006	—	80.7%	500.90	1084.40	Pentium IV 3GHz
Gabor-AB	MT	63.11	–	500.90	19.00	–
Gabor-NB	MT	89.78	77.54	500.90	115.73	–
	[101]2012	77.90	–	1550.00	–	–
Gabor-SVM	MT	82.79	90.62	500.90	206.60	–
	[107]2009	86.8	–	30000.00	–	–
	[178]2014	95.3	–	54.30	10.05	DualCore 3.2GHz 4Go RAM
	[177]2006	–	89.1%	–	–	

[a] MT : et le modèle d'analyse que nous avons testé en exploitant l'algorithme d'analyse que nous avons proposé

utilisé un ensemble de descripteurs constitué de trois descripteurs HOG, LTP et LBP et leurs paire combinaisons pour analyser les données d'apprentissage. Avant l'évaluation la performance des trois descripteurs et les paires combinaisons, leurs paramètres ont été testé et ajusté manuellement.

Dans un autre article [87], P. Cargani et al ont proposé une étude approfondie de l'application HOG conjointement avec le classifieur SVM dans le problème de la reconnaissance d'émotions humaines, souligne que le descripteur de HOG pourrait être efficacement exploité à cette fin. En particulier, les auteurs ont souligné qu'un ensemble approprié de paramètres HOG peut faire de ce descripteur l'un des plus appropriés pour caractériser les particularités de l'expression faciale. Pour cela, ils ont réalisé une session expérimentale en exploitant un pipeline algorithmique pour rechercher les paramètres optimaux du descripteur de HOG.

Grâce à l'algorithme d'analyse automatique nous avons pu constater que si l'on peut associer le classifieur Fisher avec le descripteur HOG on peut constituer un pipeline approprié qui mène à une reconnaissance d'expression du visage dans le monde réel. Cela par ce que, selon le tableau 3.3, le descripteur de HOG donne un bon compromis entre taux et temps de reconnaissance, lorsqu'il est associé avec le classifieur Fisher.

3.6 conclusion

Ce chapitre propose un algorithme pour automatiser la procédure d'analyse d'images introduite dans le chapitre 2. L'analyse d'image porte sur l'utilisation des descripteurs de la forme, la texture et les contours du visage pour extraire le vecteur des caractéristiques. En stockant les vecteurs des caractéristiques des images dans un tableau de données pour chacun des descripteurs d'images, trois tableaux de données ayant différents nombre et valeurs d'attributs sont obtenus. Ces données sont analysées par les méthodes de classification de : Fisher, séparateur à vaste marge, k-plus proches voisins, naïve de Bayésienne et d'arbre binaire.

L'objectif de l'étape d'analyse d'images d'expression est d'identifier le modèle d'analyse composé d'une paire combinaison descripteur-classifieur qui renvoie un taux de reconnaissance élevé et un temps de traitement minimal. Pour ce faire, nous avons proposé un algorithme pour automatiser l'analyse de données. Le rôle principal de l'algorithme d'analyse est de réaliser une recherche exhaustive des valeurs des paramètres et hyperparamètres qui rendent les modèles d'analyse optimaux. Cela se fait d'une façon itérative, c.-à-d. pour chaque

modèle d'analyse à chaque itération l'algorithme test par validation croisée une combinaison de valeurs des paramètres et hyperparamètres. Ensuite en testant toutes les valeurs mutuelles des paramètres et hyperparamètres l'algorithme sélectionne la combinaison des valeurs qui enregistre un score élevé. En comparant les scores obtenus après l'ajustement de tous les modèles d'analyse, l'algorithme d'analyse peut identifier le modèle d'analyse qui renvoie un score maximal.

Le résultat obtenu après l'analyse de données a montré que la combinaison du descripteur de la forme, extrayant les caractéristiques de HOG, et le classifieur de Fisher donne le score le plus élevé qui vaut 96.44%. Aussi, la comparaison du résultat obtenu avec les résultats des autres travaux dans la littérature a montré que le classifieur Fisher peut être utilisé pour classifier les expressions d'émotion. Non seulement parce qu'il donne le score le plus élevé, mais aussi parce qu'il a un temps de classification minimal qui vaut 0.78 millisecondes.

Le chapitre suivant propose d'apporter des améliorations au modèle d'analyse optimal, renvoyé par algorithme d'analyse de données, et ce, afin de proposer un pipeline identifiant l'expression dans des vidéos. le plus optimal. On peut alors choisir le modèle qui aboutit à un taux de reconnaissance élevé sur nos données d'apprentissage.

Chapitre 4

Le pipeline proposé pour la reconnaissance d'émotions dans une vidéo

4.1 Introduction

Dans le chapitre précédent, nous avons accompli l'étape d'analyse d'images qui nous a aidée à obtenir le modèle d'analyse optimal, composé de pair descripteur-classifieur. En conséquence, dans ce chapitre, nous passons à l'évaluation du pipeline dans le cas des vidéos contenant plusieurs personnes dans la scène [179]. Mais avant tout, dans la section 4.2 nous proposons et décrivons notre pipeline dans lequel nous ajoutons au modèle d'analyse trois autres étapes de pré-traitement, qui permettent de :

1. Trouver le (ou plusieurs s'ils y sont) visage après l'acquisition de la scène.
2. Suivre le visage au cours du temps jusqu'à sa disparition de la séquence d'image
3. Extraire les régions d'intérêt sur le visage pour réduire la zone du traitement du descripteur de la forme du visage

Nous montrons aussi dans la section 4.2 qu'on peut exploiter le pipeline pour réaliser une analyse approfondie sur le classifieur. Par cela, nous entendons la réalisation d'une analyse de la structure des données pour comprendre comment les classes de données sont regroupées et pouvoir par la suite améliorer davantage la performance, et ce, en ajustant la frontière de

décision du classifieur. Ici, par structure des données on désigne la relation géométrique entre les classes de données dans un espace bidimensionnel. Enfin, nous décrivons la procédure et exposons les résultats d'évaluation du pipeline sur une série de vidéo de la base de données MMI [13].

4.2 Le pipeline proposé pour la reconnaissance de l'expression à la cadence vidéo

Nous commençons ce chapitre par la description du fonctionnement du pipeline que nous présentons dans la figure 4-1. Le pipeline est basé principalement sur le modèle d'analyse jugé optimal qui se compose du descripteur de HOG et du classifieur Fisher. Pour garantir un fonctionnement à la cadence vidéo, une étape de pré-traitement garantissant le suivi du visage et une autre étape d'extraction des régions d'intérêts sont ajoutées au modèle d'analyse. Dans cette section, nous détaillons l'ensemble des opérations effectuées lors du fonctionnement du pipeline en donnant comme entrée une vidéo.

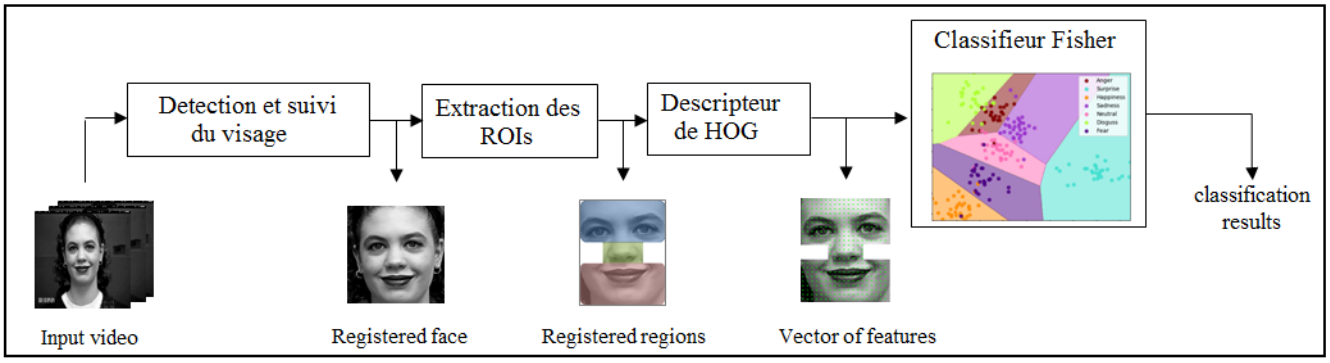


FIGURE 4-1 – Vue d'ensemble du fonctionnement du pipeline proposé exploitant une vidéo

4.2.1 Recherche et suivi du visage dans une vidéo

Comme nous nous intéressons à la détection du visage de face nous avons utilisé l'algorithme rapide de Viola-Jones [180]. Cet algorithme se caractérise par des techniques rapides de calcul de centaines de milliers de caractéristiques de Haar et de sélection de celles propres à un visage dans la fenêtre glissante. L'ensemble des opérations de la méthode de Viola et Jones, se fait dans une fenêtre glissante initiale de taille 24 x 24 qui balaye l'image et qui augmente sa taille à chaque nouveau balayage afin de chercher des visages de différentes

tailles et dans différents endroits de l'image. Si la fenêtre glissante ne trouve pas le visage dans la première image acquise, elle continue sa recherche dans les trames suivantes. Si un ou plusieurs visages sont détectés pour la première fois, une fenêtre englobante est définie sur les visages détectés.

Pour obtenir une détection à la cadence de la vidéo, le suivi du visage est assuré par la méthode de Kanade-Lucas-Tomasi (KLT [3]) aussi bien pour les visages déjà détectés dans les images précédentes que pour les nouveaux visages détectés dans les trames suivantes. Le suivi est réalisé en quatre étapes. Dans la première étape, l'idée est d'analyser la variation du niveau de gris à l'intérieur de la fenêtre englobante du visage détecté, et ce, pour extraire un nuage de 100 de points répartis autour des contours comme le montre la figure 4-2.

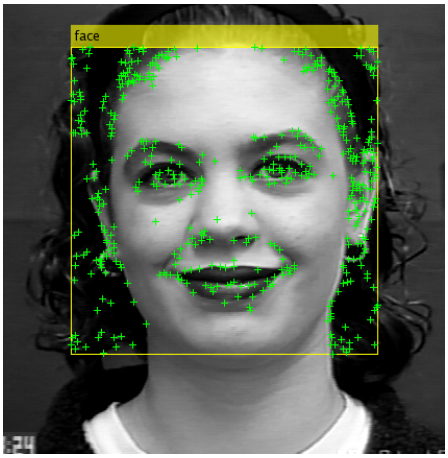


FIGURE 4-2 – Distribution des points retrouvée dans la fenêtre englobante représentée en jaune.

Ensuite dans la deuxième étape, le calcul du déplacement des points entre des images consécutives garantit le suivi du visage. Comme la dynamique du visage peut causer la disparition de quelques points au cours du temps, la troisième étape essaye de maintenir le nombre de point initial tout en extrayant de nouveaux points à chaque saut de 5 images pour remplacer les points perdus. Finalement, la quatrième étape maintient le suivi des nouveaux et des anciens points en répétant la première et la deuxième étape. Lorsque les deux tiers des points disparaissent brusquement, on dit que le visage n'existe plus dans la scène. En conséquence, le suivi s'arrête et le détecteur du visage reprend la main pour scanner toute l'image.

Le rôle le plus important du suivi du visage est de minimiser le temps de détection en réduisant la zone d'image où le détecteur de Viola-Jones va chercher de nouveaux visages. En

d'autres termes, les régions d'image soumises au contrôle du suiveur du visage ne sont pas traitées par le détecteur du visage, ce qui permet de gagner en temps de calcul. Le temps est encore réduit en sautant un certain nombre de trames lors de l'appel du détecteur du visage. Pour cela, après avoir mesuré la vitesse du traitement relative au pipeline tout en sautant différent nombre de trames à chaque fois que le pipeline est ré-exécuté, nous avons décidé de faire fonctionner le détecteur du visage après un saut de 10 images en assurant le suivi des visages sur chacune des 10 images sautées.

Le point intéressant à ce niveau c'est que, même si le détecteur du visage saute 10 images, l'étape de suivi garantit le suivi des visages d'une image à l'autre sans sauter aucune image, ce qui aboutit à un temps de traitement convenable à la cadence vidéo. Cependant, le défi consiste à trouver de nouveaux visages lors du saut des 10 images. En analysant ce point dans l'aspect temporel, le saut de 10 images dans le cas de vidéos ayant une vitesse de 24 fps signifie attendre 0,42 secondes avant de mettre à jour l'étape de détection des visages pour rechercher de nouveaux visages, sachant que le suivi est maintenu à ceux déjà détectés pendant les 0.42 secondes. Lors de l'exécution du pipeline, l'effet du retard de 0,42 seconde n'est pas observable visuellement et on peut avoir une détection à la cadence vidéo en dépit de ce retard, car il n'est lié qu'au début de la détection du visage mais une fois le visage détecté l'expression est analysée d'une image à l'autre jusqu'à disparition du visage.

4.2.2 Segmentation des régions d'intérêts sur le visage

Une fois le suivi est assuré dans la scène vidéo, on localise des régions d'intérêt, qui fournissent plus d'informations sur la forme du visage lors du changement de l'expression, telle que la région du front et les yeux, la région du nez et la région de la bouche. Par conséquent, les parties du visage non-pertinentes sont éliminées.

Tant que le visage est détecté dans sa position de face, les régions d'intérêt peuvent être localisées d'une façon facile et rapide par l'approche géométrique [181] que par des détecteurs des traits du visage. La méthode géométrique est basée sur le fait que le visage humain a la même configuration géométrique chez les gens et qu'il y a une proportionnalité entre les dimensions des traits, la largeur et la longueur du visage.

Dans cette optique, nous avons segmenté les régions d'intérêts du visage tout en divisant le visage détecté par 6 x 6 blocs égaux, puis, en sélectionnant les pixels qui se trouvent sous les blocs colorés en bleu, vert et rouge (voir figure 4-3). Par exemple, la région des yeux est

délimitée par l'intersection des blocs 2 et 5 qui se trouve sur l'axe horizontal avec les blocs 2 et 3 qui se trouvent sur l'axe vertical.

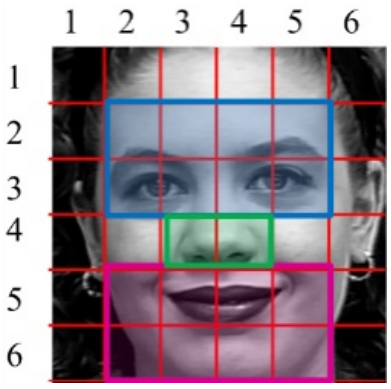


FIGURE 4-3 – Subdivision du visage et sélection des régions d'intérêts

4.2.3 Extraction de la forme du visage : descripteur de HOG

D'après les résultats de l'analyse d'images obtenus dans le chapitre 3, le descripteur de HOG est choisi pour la description de la forme de l'expression du visage. Alors, pour générer le vecteur caractéristique, les trois ROIs sont encore partitionnées en petits blocs comme indiqué dans la figure 4-4. Dans chaque bloc, la distribution des directions des contours est mesurée à l'aide d'un histogramme orienté h_i. En concaténant l'ensemble des h_i dans un seul histogramme H on obtient le vecteur caractéristique H=X=[x_1,x_2,x_3,...,x_n], où n est le nombre total des caractéristiques de HOG extraites des ROIs.

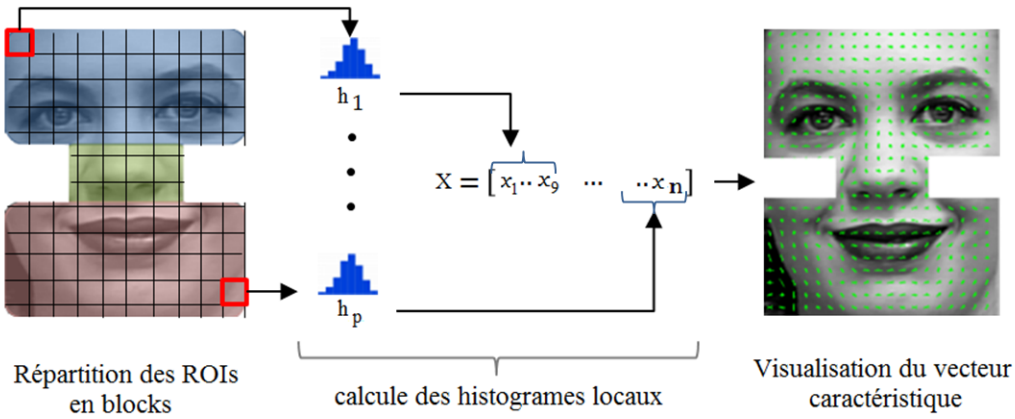


FIGURE 4-4 – Extraction des caractéristiques des HOG dans les régions d'intérêts

4.2.4 Entraînement du classifieur d'expression par méthode de Fisher

Une utilisation efficace du descripteur HOG nécessite l'utilisation d'une méthode de classification robuste et rapide. Habituellement, la méthode de Fisher est utilisée pour la réduction des dimensions de données. En fait, le choix d'utiliser la méthode de Fisher à des fins de réduction de dimension ou des fins de classification dépend fortement de la relation entre les classes lorsque les données originales subissent une transformation au niveau des attributs pour créer un nouvel ensemble de données ayant moins d'attributs. Généralement, les nouveaux attributs sont dérivés des anciens à travers le calcul statistiques que nous avons présenté dans la section 2.3.1. Dans la suite, nous montrons comment on exploite le pipeline pour réaliser des analyses approfondies sur la structure de données lors de l'entraînement du classifieur Fisher

4.2.4.1 Réduction de données : méthode de Fisher

Comme la reconnaissance d'émotion ne peut pas être réalisée que sur deux phases distinctes : une phase d'entraînement du classifieur et une phase de classification. Nous montrons dans la figure 4-5 le schéma de la phase d'entraînement du classifieur Fisher. La figure comprend les étapes de pré-traitement, en plus à l'étape de réduction de dimension effectuée.

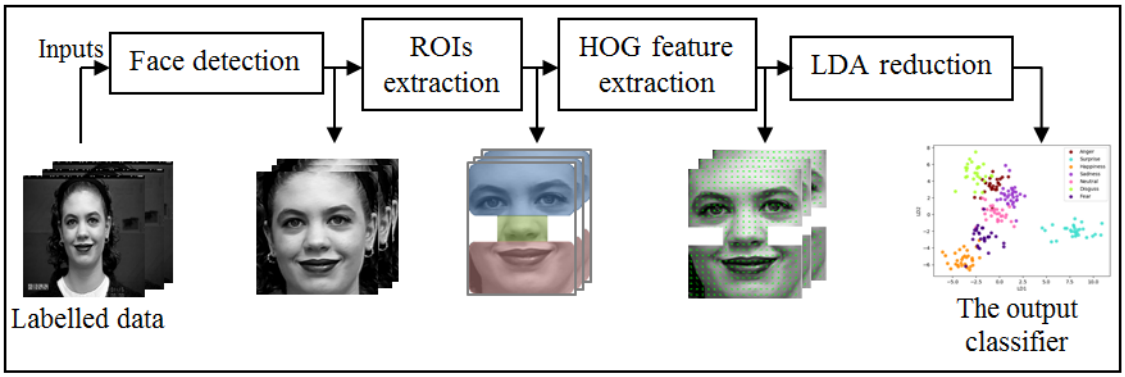


FIGURE 4-5 – Vue globale de la phase d'entraînement du classifieur

Le schéma de la figure 4-5 prend, donc, comme entrée l'ensemble des images. Ensuite, chaque image passe par les étapes de la détection du visage, d'extraction des régions d'intérêt et d'extraction de la forme du visage. À la sortie du descripteur de HOG, le vecteur caractéristique extrait de chaque image conjointement à son étiquette y_i sont stockés dans la ligne du tableau $\Omega_{(N,P)}$, dont N est le nombre d'images utilisées et P le nombre total des

caractéristiques de HOG extraites du visage :

$$\Omega_{(N,768)} = \begin{pmatrix} x_{1,1} & x_{1,2} & \cdots & x_{1,768} & y_1 \\ x_{2,1} & x_{2,2} & \cdots & x_{2,768} & y_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ x_{N,1} & x_{N,2} & \cdots & x_{N,768} & y_N \end{pmatrix} \tag{4.1}$$

En se basant sur la méthode de Fisher détaillée dans la section 2.3.1, on transforme les données $\Omega_{(N,768)}$, ayant 768 attributs, en données $\Omega'_{(N,2)}$ ayant deux attributs nommés x_0 et x_1

$$\Omega' = \begin{matrix} x'_0 & x'_1 & \\ \end{matrix} \begin{pmatrix} x'_{1,1} & x'_{1,2} & y_1 \\ x'_{2,1} & x'_{2,2} & y_2 \\ \vdots & \vdots & \vdots \\ x'_{N,1} & x'_{N,2} & y_N \end{pmatrix} \tag{4.2}$$

Les données Ω' peuvent être représentées dans un diagramme de dispersion permettant ainsi une vision plus détaillée de la structure de données dans un plan bidimensionnel. Dans la figure 4-6 nous donnons le diagramme de dispersion, cela nous permet de comprendre la relation géométrique entre les classes de données dans la phase d'entraînement.

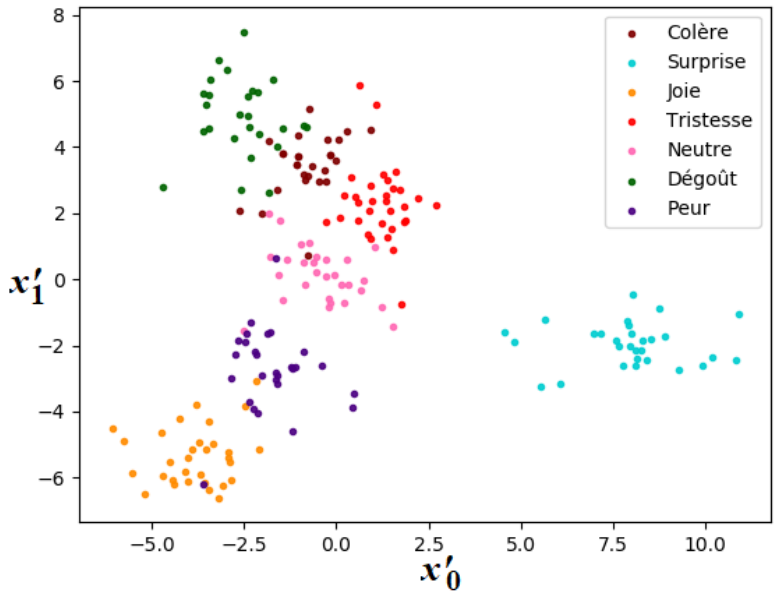


FIGURE 4-6 – Diagramme de dispersion montrant les clusters construits après extraction des caractéristiques de HOG et réduction de dimension par la méthode de Fisher

En effet, dans le cas de l'existence de chevauchements importants entre les classes dans le diagramme de dispersion, cela peut exclure la méthode Fisher de l'option d'être utilisée en tant que classifieur. Cependant, lorsque les données sont séparables, c.-à-d. lorsque nous avons moins de chevauchement entre les classes, la méthode de Fisher peut produire un classifieur efficace si on arrive à trouver une frontière de décision convenable qui augmente le taux de reconnaissance lors du test.

Le diagramme de dispersion de la figure 4-6 montre que les sept classes d'émotion peuvent être regroupées séparément. Cependant, il y a un chevauchement entre les clusters sur certaines zones du diagramme, particulièrement aux frontières des classes d'expression de joie et de peur. A ce niveau, nous cherchons à construire un classifieur qui sépare au mieux les points appartenant à la même classe de ceux qui appartiennent aux autres. Dans ce contexte, le pipeline a été exploité pour tester quelques fonctions de séparation qui procèdent à la séparation des classes d'expressions de façons linéaire et non-linéaire.

4.2.5 Étude approfondie sur la méthode de Fischer pour l'amélioration de la reconnaissance

Nous avons mentionné dans le chapitre 3 que la méthode de Fisher se base sur l'hypothèse que les classes de données ont une distribution multivariée gaussienne. Ceci permet de tracer des frontières de décision linéaires entre les clusters créés après la réduction du nombre des attributs de données d'apprentissage. Cependant, dans cette section, nous analysons la structure de données tout en testant et évaluant des frontières de décision linéaires et non linéaires.

4.2.5.1 Séparation linéaire : méthode d'arbre binaire

On peut utiliser l'arbre binaire pour chercher une frontière de décision qui sépare les clusters des données $\Omega' = \{(X'_0, y_0), \ldots, (X'_{N-1}, y_{N-1})\}$, $X'_i \in \Re^2$, de façon linéaire. Dans la section 2.3.5 nous avons exploité la méthode de l'arbre binaire pour construire un classifieur d'expression lorsque l'espace d'entrée χ contenait des données Ω de dimensions $\Re^P$.

Nous montrons dans cette section que la méthode d'arbre binaire peut créer des frontières de décision, géométriques, séparant les classes de $\Omega' \in \Re^2$.

L'arbre binaire découpe l'espace d'entrée $\Omega' \in \Re^2$ en région dont les côtes sont des lignes droites perpendiculaires aux axes x'_0 et x'_1 comme le montre la figure 4-7. Ces lignes sont déterminées par l'arbre binaire en cherchant le nœud racine et les nœuds des branches et des feuilles.

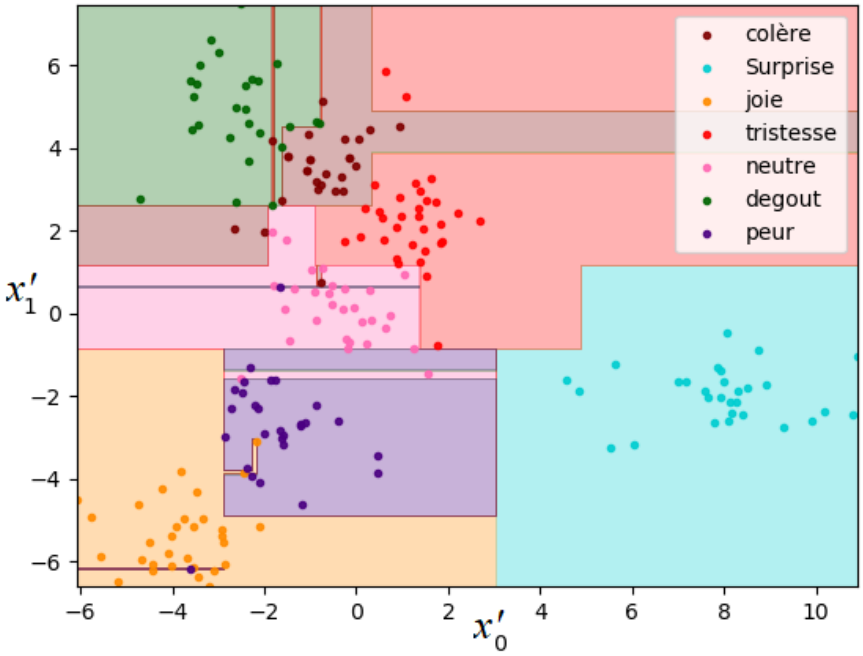


FIGURE 4-7 – Résultat de séparation par la fonction séparatrice de l'arbre binaire

Dans la figure 4-8 nous détaillons la phase de la construction de l'arbre binaire tout en visualisant une partie de l'arbre. Au niveau du nœud racine, étant donné le mélange des exemples des sept classes [colère, Surprise, joie, tristesse, neutre, dégoût, peur] avec un nombre d'exemples qui vaut respectivement [27,29,31,32,29, 26,28], l'algorithme de l'arbre binaire réalise des tests sur les attributs x'_0 et x'_1.

Dans chaque test s'agit de calculer l'entropie H des attributs en utilisant l'équation (2.40), qui mesure l'hétérogénéité du nœud. Par exemple, si le nœud contient un mélange hétérogène, comme le nœud racine indiqué dans la figure 4-8, l'entropie est élevée. Mais, lorsque le mélange est pur, c.-à-d. contient que les exemples d'une seule classe comme les feuilles indiquées dans la figure 4-8, l'entropie est nulle.

Comme l'objectif de la méthode d'arbre binaire est de chercher des nœuds purs permettant de classifier les expressions, les tests servent donc à sélectionner l'attribut qui a une entropie minimale qui divise le mélange hétérogène en fonction d'un seuil. Une technique pour trouver

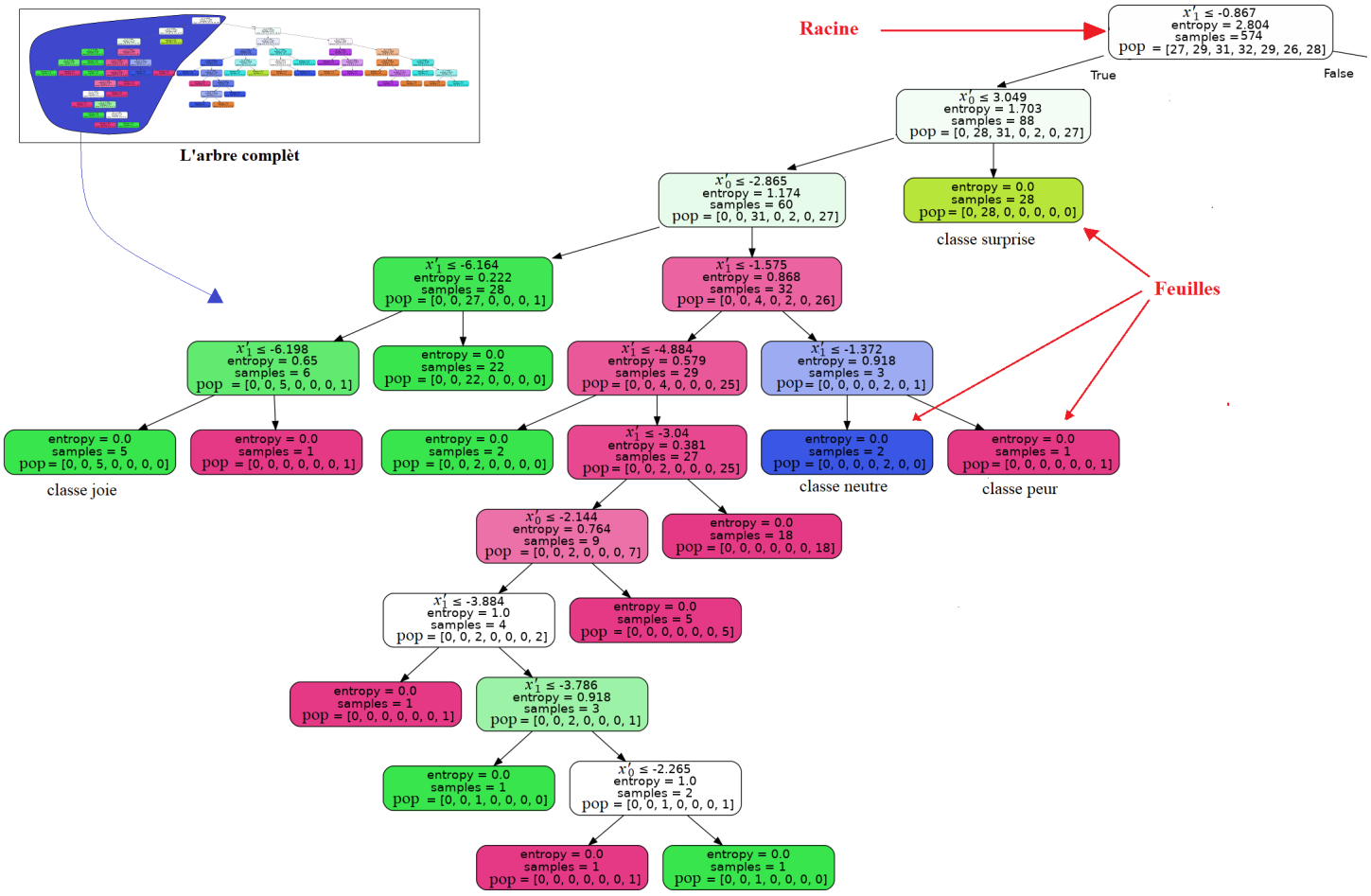


FIGURE 4-8 – Visualisation de la partie gauche de l'arbre binaire construite à partir de données Ω'. Dans chaque nœud nous montrant : le seuil du partitionnement de données selon les attributs X'_0 et X'_1 , l'entropie des attributs H , le nombre des exemples restant après le partitionnement de données, et le mélange [colère, Surprise, joie, tristesse, neutre, dégoût, peur]

le seuil consiste à trier les valeurs de l'attribut peu importe l'ordre croissant ou décroissant. Ensuite, en considérant les points médians entre deux valeurs adjacentes comme une position de division candidate, nous calculons le gain d'information après la division g(t) pour chaque candidat en utilisant l'équation (4.3).

$$g(t) = H(parent) - \frac{m_g}{M} H_g(fils/t) - \frac{m_d}{M} H_d(fils/t) \tag{4.3}$$

H(parent) l'entropie du nœud parent avant la division, m_g et m_g sont le nombre total des exemples dans le mélange du nœud fils gauche et droit respectivement, M est le nombre total

des exemples du nœud parent, $H_g(fils/t) \geq 0$ *et* $H_d(fils/t) \geq 0$ *l'entropie des nœuds fils gauche et droit, respectivement, après la division du mélange du nœud parent selon t.*

Le rôle du gain d'information consiste à informer combien du désordre est supprimé après la division du mélange du nœud parent. Minimiser le désordre au niveau du mélange des nœuds fils revient maximiser le gain d'information g(t), et en choisissant le seuil t qui minimise l'entropie des nœuds fils $H_g(fils/t)$ et $H_d(fils/t)$.

Lors de la construction de l'arbre binaire, à chaque division de données deux nœuds fils sont générés, dont les exemples ayant une valeur d'attribut inférieure ou égale au seuil t se placent dans le nœud fils à gauche et le reste des exemples se placent dans le nœud fils à droite. Pour continuer le branchement des nœuds, l'algorithme répète les tests sur les attributs x'_0 et x'_1 lorsque le mélange des nœuds après la division est hétérogène, et arrête les tests quand les nœuds sont purs.

Pour expliquer la frontière de décision tracée par l'arbre binaire, nous présentons dans la figure 4-9 les linges séparatrices obtenues lors des deux premières divisions effectuées sur les données.

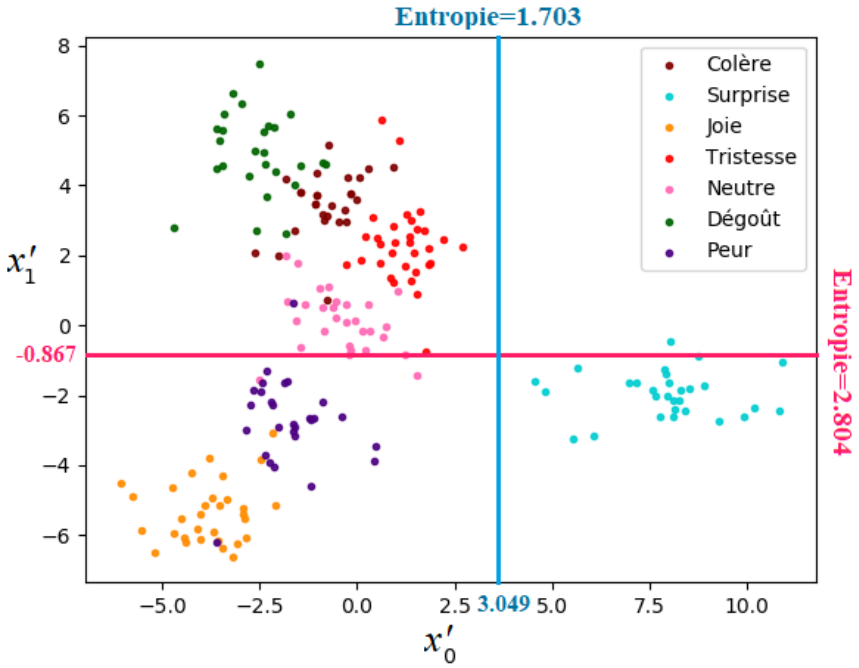


FIGURE 4-9 – Partitionnement de l'espace d'entrée $\Omega' \in \Re^2$ obtenu par les seuils du nœud racine et du nœud fils gauche de l'arbre binaire construit

En prenant en compte les résultats des tests du nœud racine présentés dans la figure 4-8,le seuil $t = -0.867$ est exploité pour effectuer le premier partitionnement de données selon l'axe x'_1. Géométriquement, ce partitionnement de données est illustré dans la figure 4-9 par une droite perpendiculaire à l'axe x'_1 passant par la valeur $x'_1 = -0.867$ qui sépare les exemples des classes joie, surprise et peur des exemples des autres classes.

En considérant cette fois-ci le résultat du test du nœud fils de la racine (voir figure 4-8), le deuxième partitionnement est réalisé en traçant fois-ci une droite perpendiculaire à l'axe x'_0 qui passe par le seuil $t = 3.049$ comme illustré dans la figure 4-9. En faisant l'union des régions partitionnées appartenant à la même classe, on obtient la frontière de décision finale du classifieur de la figure 4-7.

L'arbre binaire peut gérer des frontières étranges entre les classes, mais au prix d'un arbre plus profond. Plus l'arbre est profond plus on a des lignes qui séparent les points intrus des points du même classe.

4.2.5.2 Séparation non-linéaire : analyse discriminante quadratique

Au niveau de la méthode d'analyse discriminante linéaire (voir section 2.3.1.2), on se base sur l'hypothèse selon laquelle les classes ont la même covariance pour trouver la fonction objectif $\delta_c(X)$ d'équation (2.28). Cependant, cette hypothèse peut être assez restrictive, car en réalité, les classes de données ont une covariance différente les unes des autres. Dans l'analyse discriminante quadratique de Fisher, nous estimons une moyenne μ_c et une matrice de covariance σ_c^2 propre à chaque classe séparément. Ceci permet d'exprimer la fonction objectif $\delta'_c(X)$ par l'équation suivante :

$$\delta'_c(X) = \underbrace{\log(p(c) - \frac{\mu_c^2}{2\Sigma_c} + \frac{X\mu_c}{\Sigma_c}}_{\delta_c(X)} - \frac{X^2}{\Sigma_c} - \frac{1}{2}\log(|\Sigma_c|) \tag{4.4}$$

$c \in \{1, \ldots, 7\}$ la classe d'expression, $\mu_c \in \Re^2$ le vecteur moyen de la classe c calculé par l'équation (2.14)*, $\Sigma_c \in \Re^{2\times 2}$ la variance des attributs x'_0 et x'_1 calculée par l'équation* (2.21)*, $p(c)$ la probabilité de la classe c*

En faisant apparaître la partie rouge dans l'équation (2.28), la fonction objectif est maintenant quadratique en x, aux frontières de décision obtenues dans la figure 4-10 ci-dessous.

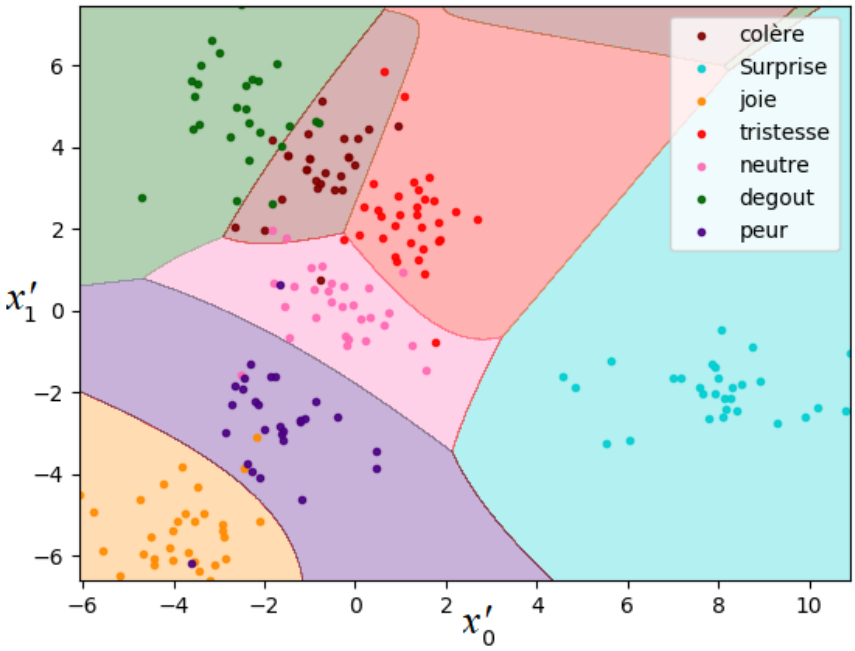


FIGURE 4-10 – Résultat de séparation par l'analyse quadratique de Fisher

4.2.5.3 Séparation non-linéaire : méthode Naïve Bayésienne

Tout comme l'analyse discriminante linéaire et quadratique de Fisher le classifieur naïf de Bayes est également basé sur la règle de décision de Bayes. Cependant, une hypothèse simplificatrice est faite sur les attributs individuels de x'_i, et ce, en considérant que les x'_i sont indépendants. À travers cette simplification, la fonction densité conditionnelle des classes $P(X = x|C = c)$ de l'équation (2.25) devient :

$$P(X = x'|C = c) = \prod_{j=1}^{2} f_{cj}(x'_j) = \prod_{j=1}^{2} \frac{1}{\sigma_{cj}\sqrt{2\pi}} e^{-\frac{(x'_j - \mu_{cj})^2}{2\sigma_{cj}^2}} \tag{4.5}$$

f_{cj} la densité de probabilité du j-ième attribut dans la classe c, σ_{cj} la covariance des attributs calculée par l'équation (2.22), *$\mu_{cj} \in \Re$ la moyenne du j-ième attribut dans la classe c calculée par l'équation* (2.14)

En remplaçant l'équation (4.5) dans l'équation (2.23), la fonction objectif $\delta_c^{"}(X)$ peut s'exprimer de la façon suivante :

$$\delta_c^{"}(X) = \log(p(c)) - \sum_{j=1}^{2} \left[\log(\sigma_{cj}) + \frac{(x - \mu_{cj})^2}{2\sigma_{cj}^2} \right] \tag{4.6}$$

$p(c)$ la probabilité de la classe d'expression c pour $c \in \{1, \ldots, 7\}$

La ligne séparatrice entre deux classes C_i et C_j, dans ce cas, se définit alors par les points où les densités conditionnelles à priori relatives aux classes C_i et C_j sont égales. Aux limites des classes d'expression, on a :

$$X \in \Re^p | \delta_i^{"}(X) - \delta_j^{"}(X) = 0 \tag{4.7}$$

D'après l'équation (4.6) la puissance carrée de $(x - \mu_{cj})$ montre que la ligne séparatrice est quadratique aussi. Si l'on compare les règles de décision de la figure 4-11 avec celles de la figure 4-10, on remarque qu'elles sont un peu similaire sauf qu'on a peu de déviation entre les lignes séparatrices des deux techniques. Cela vient de la simplification "naïve" qui rend les attributs indépendants et qui fait apparaître les σ_{cj} dans la fonction objectif.

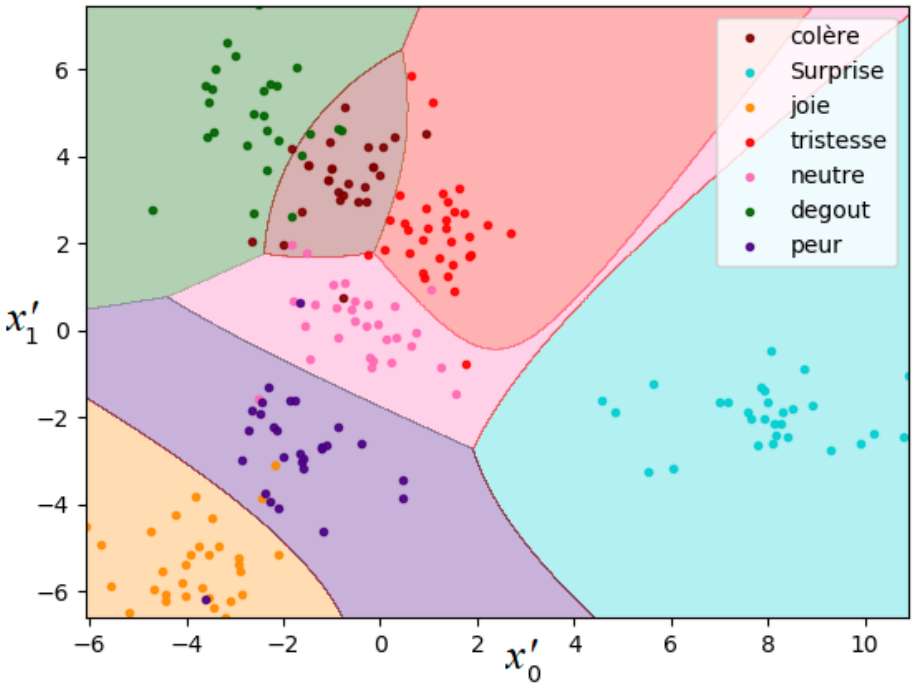


FIGURE 4-11 – Résultat de séparation par la fonction séparatrice du classifieur naïf de Bayes

4.2.5.4 Séparation non-linéaire : méthode de k plus proches voisins

La frontière de décision du classifieur k-plus proches voisin se base essentiellement sur le diagramme de Voronoï. L'idée ici, c'est de former des lignes séparatrices par l'intersection des médiatrices perpendiculaires de chaque paire de points appartenant à des classes différentes. La figure 4-12, illustre le principe de fonctionnement du diagramme de Voronoï dans le cas de 1-ppv, où k est égal à 1.

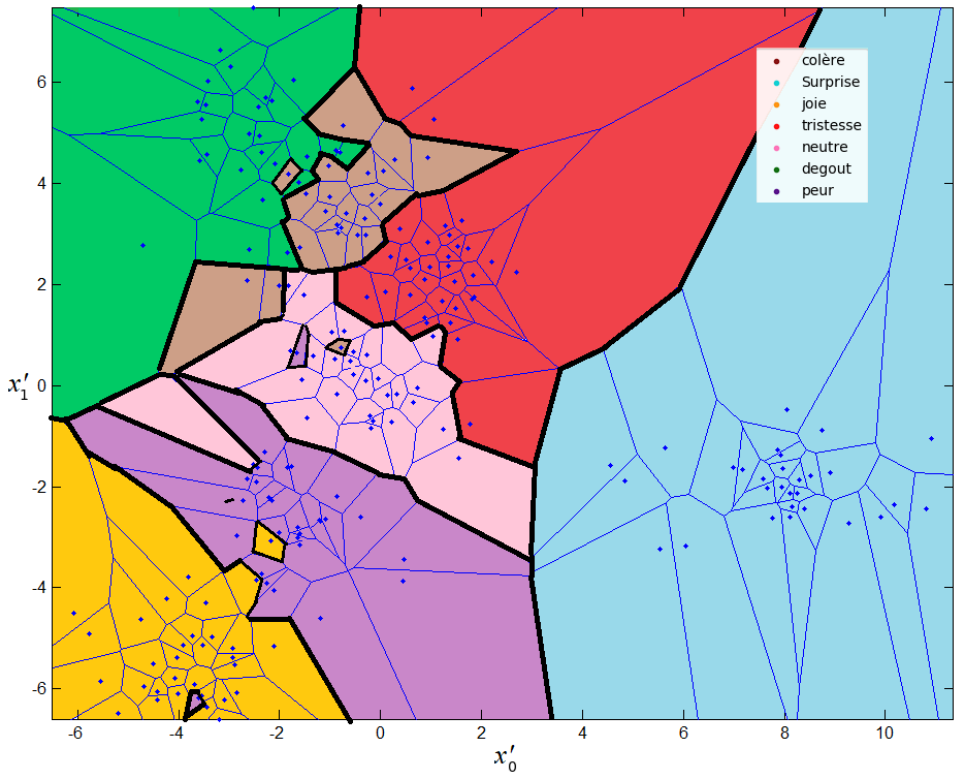
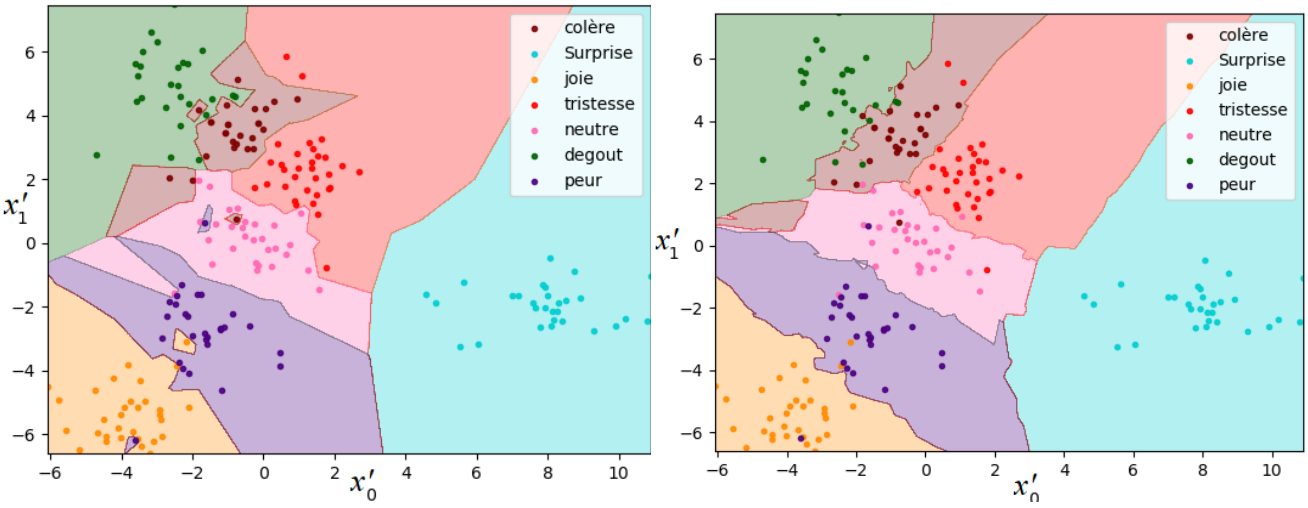


FIGURE 4-12 – Ensemble de points appartenant aux sept classes d'expressions et leurs zones de Voronoï qui sont des polygones convexes. La séparatrice entre les classes par la frontière de décision 1-ppv est en trait gras, qui est la ligne séparatrice entre les polygones convexes par l'union des lignes de Voronoï des exemples de chaque classe

En effet, le diagramme de Voronoï permet de tracer par morceaux des lignes médiatrices perpendiculaires entre chaque paire de points voisins pour créer polygones convexes dite "zone de Voronoï" pour chaque point. Ensuite, pour définir la frontière de décision, les zones de Voronoï qui appartiennent à la même classe sont fusionnées, c.-à-d. les limites séparant les points de la même classe sont supprimées tandis que les autres sont gardées comme le montre la figure 4-13a. Dans le cas ou k>1 (voir figure 4-13b) les séparatrices sont aussi des lignes séparatrices par morceaux.



(a) 1-ppv

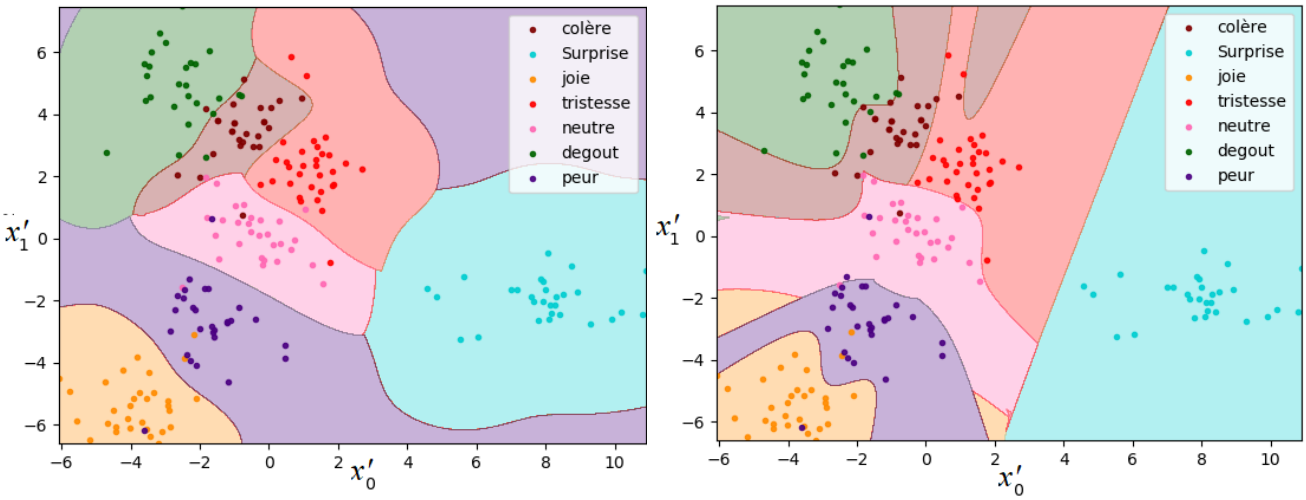
(b) 8-ppv

FIGURE 4-13 – Résultat de la fonction séparatrice du k-plus proches voisins, à gauche : séparation des classes en utilisant un seul voisin et à droite séparation en utilisant huit voisins

L'effet du changement du nombre k est illustré sur la figure 4-13. Le choix de k ne peut pas être arbitraire. En prenant k petit (égale à 1 par exemple), ceci rend la méthode k-plus proches voisins sensible au bruit, généralement on appelle ce phénomène un sur-apprentissage. À ce niveau, le classifieur peut atteindre une précision de 100% sur l'ensemble d'apprentissage mais la précision diminue dramatiquement pour des données qui n'appartiennent pas à l'ensemble d'apprentissage. Au contraire, si l'on prend un nombre de k aussi grand (figure 4-13b), la précision diminue puisque des points voisins qui appartiennent à des classes différentes les unes des autres sont prises en compte. Ce qui ne nous permet pas de savoir quelle est la classe majoritaire au voisinage des points. En effet, la méthode k-plus proches voisins ne représente pas explicitement les lignes séparatrices comme les autres techniques de séparation, il peut donc gérer tout type de structure compliquée formée par les exemples des données d'apprentissage.

4.2.5.5 Séparateur à Vaste Marge

Dans notre cas les données contiennent sept classes, nous serons donc confrontés au problème de chevauchements entre les clusters. Pour voir l'effet du choix du noyau sur la performance de reconnaissance, nous avons testé deux fonctions noyau : le noyau RBF et le noyau polynomial. Sur la figure 4-14 nous représentons l'effet du noyau sur le diagramme de



(a) Noyau RBF

(b) Noyau polynomial de degré 4

FIGURE 4-14 – Effet de deux différents noyaux lorsqu'ils sont appliqués au diagramme de dispersion des sept classes d'émotion.

dispersion sur les sept classes d'émotion. L'utilisation des fonctions noyaux nous a conduit à trouver des frontières de décision non-linéaires et compliquées, en terme du calcul, par rapport à celles produites nativement par la version primale du SVM dans la figure 4-15 qui se basant sur l'équation (2.36) pour chercher les lignes séparatrices linéaire.

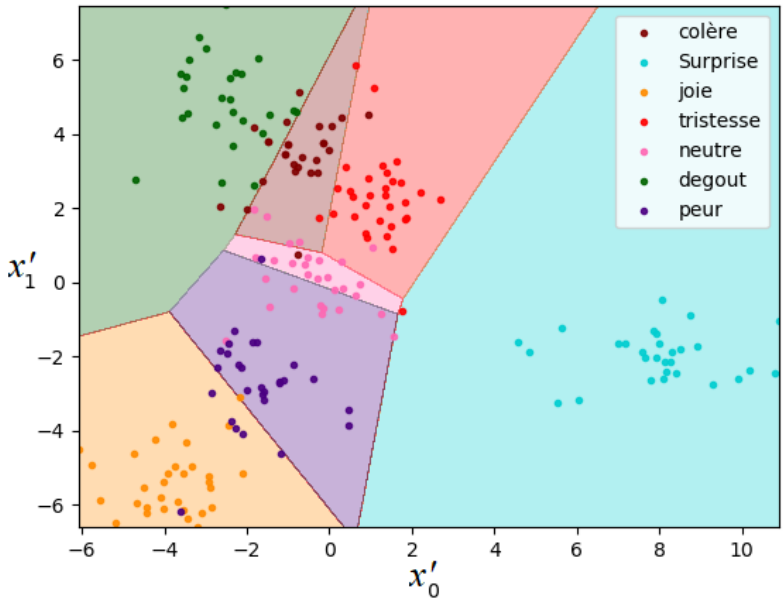


FIGURE 4-15 – Frontière de décision linéaire produite par la version primale de la méthode SVM.

Dans la méthode SVM, le choix du noyau est prépondérant. En effet, sur les figures 4-14a et 4-14b nous remarquons que les noyaux du SVM présentent un inconvénient qui limite leur mise en pratique. Le problème est lié au manque d'interprétabilité des lignes séparatrices produites. Par exemple, dans la figure 4-14a la frontière de décision du noyau RBF segmente les régions des points appartenant à la même classe, et nous remarquons aussi que même si aucun point n'existe dans la région se trouvant à droite en haut, celle-ci est définie comme une région de la classe d'expression de la peur. Cela, fait l'origine des fausses reconnaissances : lorsqu'un point de test appartenant aux classes de la surprise et la tristesse se projette et s'identifie dans cette région.

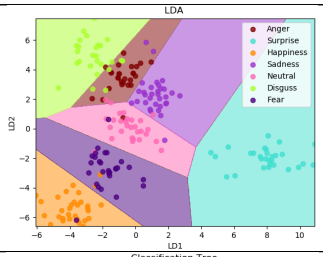
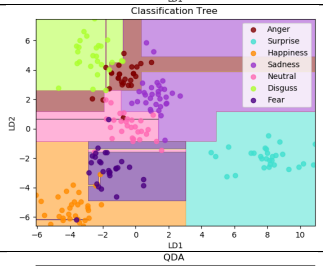
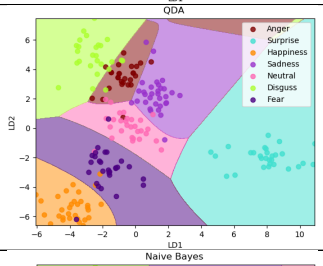
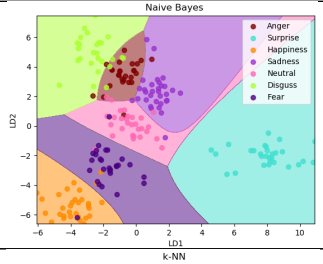
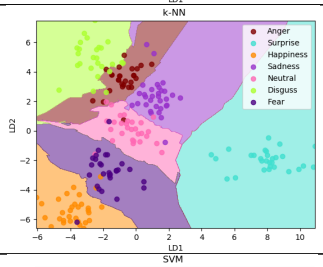
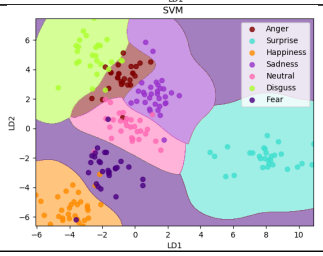
4.2.6 Le choix de la frontière de décision adéquate

Après avoir testé différentes techniques de séparation des classes, nous les évaluons pour pouvoir choisir la technique séparation qui nous permet de bien classifier des images qui n'appartiennent pas à l'ensemble d'entraînement. Le tableau 4.1 résume l'analyse reposant sur le test des fonctions séparatrices que nous avons menées sur la structure de données.

D'après les resultats du tableau, l'utilisation des fonctions séparatrices telles que l'analyse discriminante quadratique, le noyau RDF, le k-plus proches voisins ou de l'arbre binaire n'améliore pas la reconnaissance. Le problème ici, c'est que même si ces fonctions séparatrices semblent être géométriquement correctes, c.-à-d. arrivent à segmenter chaque classe de l'autre avec moins d'erreurs dans les diagrammes de dispersion, ces fonctions échouent pendant la classification des nouvelles images qui n'appartiennent pas à l'ensemble d'apprentissage.

La raison c'est que ces fonctions créent des lignes séparatrices pour segmenter des régions en donnant l'étiquette d'expression même si aucun point de données n'est pas disponible dans ces régions. Quand une région dans le diagramme de dispersion ne contient aucun point, il est mieux et de donner à cette région l'étiquette de la région voisine. Ceci est accompli par l'utilisation de méthode l'analyse discriminant linéaire qui sépare les clusters en supposant que les classes de données ont une distribution normal multivarié.

TABLE 4.1 – Résultats de généralisation des classifieurs de Fisher en utilisant différentes techniques de séparation de données

Fonction séparatrices	**Score (%)**	**Frontière de décision**
Séparation linéaire : analyse discriminante linéaire	96.44	
Séparation linéaire : Arbre de decision	43.00	
Séparation non-linéaire : Fisher quadratique basé sur le théorème de bayes	80.00	
Séparation non-linéaire : Naïf de Bayes	95.00	
Séparation non-linéaire : k-plus proches voisins	77.00	
Séparation non-linéaire : noyau RDF du SVM	60.00	

Dans le tableau 4.1, Le problème de séparation de données se manifeste clairement dans le cas du diagramme de dispersion montrant la frontière de décision de la fonction noyau RBF. En regardant les lignes séparatrices de cette technique, on remarque que la région supérieure à droite du diagramme de dispersion est identifiée comme faisant partie de la classe des visages tristes alors qu'aucun point appartenant à cette classe ne se trouve dans cette région. Ainsi, dans ce cas, si une nouvelle image du visage que l'on cherche à classifier son expression est projetée dans cette région, celle-ci sera classée incorrectement.

L'analyse du tableau 4.1 montre que la séparation des clusters de la méthode de Fisher en utilisant l'analyse discriminante linéaire, semble adéquate à la séparation des sept classes d'expression et la classification de nouvelles images qui n'appartiennent pas à l'ensemble d'apprentissage. C'est pour cette raison que nous privilégions cette méthode et nous l'utiliserons pour classifier les expressions des visages

4.2.7 Classification de l'expression du visage : classifieur de Fisher

Une fois que la fonction séparatrice est fixée, le pipeline utilise le classifieur pour reconnaître les expressions du visage en temps réel à la cadence vidéo. A cette étape, lorsqu'une nouvelle image de vecteur caractéristique X_r est projetée à son emplacement dans le diagramme de dispersion, l'expression de X est déterminée selon la classe C qui donne la probabilité postérieure la plus élevée, on note :

$$C = \underset{c\in\{1,2,...,7\}}{\arg\max}\ P(C = c/X = x) \tag{4.8}$$

$P(C = c/X = x)$ *probabilité de la classe étant donné l'exemple* X

Une fois que le classifieur reconnaît l'émotion des visages détectés, le pipeline affiche l'expression au-dessus de chacune des fenêtres englobante contenant le visage. En plus de cela, le pipeline enregistre le résultat de la reconnaissance et l'emplacement des visages afin d'évaluer le pipeline.

4.3 Évaluation du fonctionnement du pipeline sur vidéo

Après l'élaboration du pipeline, vient l'étape de l'évaluation du fonctionnement du pipeline par exploitation des vidéos. Pour ce faire, il nous a été primordial de mettre une procédure d'évaluation pour cela, nous avons construit le schéma présenté dans la figure 4-16. Ce schéma est composé des mêmes étapes du pipeline proposé dans la figure 4-1, dans lequel nous ajoutons deux blocs pour pouvoir effectuer l'évaluation du pipeline. Le premier bloc vise à évaluer le détecteur du visage pour confirmer si le visage est bien détecté tandis que le second bloc évalue le classifieur en précisant si l'expression du visage est reconnue.

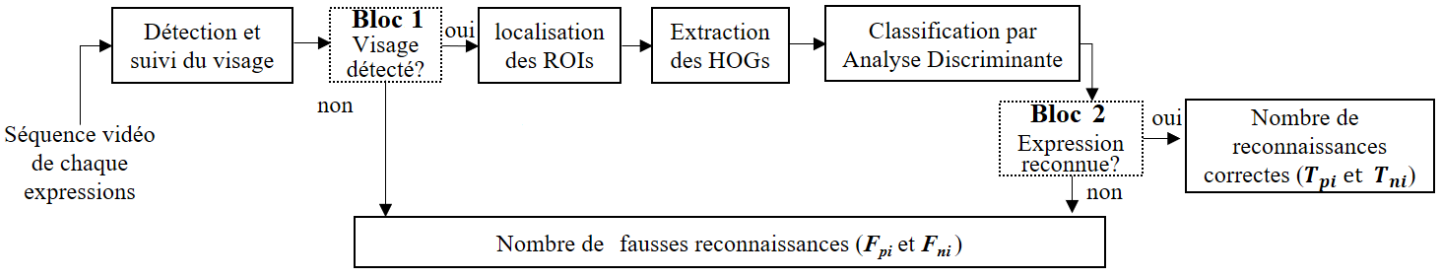


FIGURE 4-16 – Schéma d'évaluation du pipeline à la cadence vidéo

Durant l'évaluation, le pipeline prend d'abord en entrée un ensemble de vidéos étiquetées. Chacune des vidéos est composée d'une séquence d'images contenant des visages affichant le changement d'émotions de l'état neutre jusqu'à l'état où l'intensité de l'émotion est élevée. Les vidéos que nous avons exploitées sont extraites de la base de données de MMI [36]. Cette base contient des séquences vidéo de 75 participants de différentes origines : européenne, asiatique et africaine. Chaque participant affiche diverses expressions d'émotion, y compris les sept expressions d'émotions universelles. Cette base contient aussi des visages partiellement occultés par les cheveux et les lunettes et des vidéos enregistrées dans diverses conditions d'éclairage. Nous décrivons dans ce qui suit le fonctionnement des deux blocs que nous avons ajoutés dans la figure 4-16.

4.3.1 Évaluation du détecteur du visage : bloc 1

Pour évaluer le détecteur du visage en utilisant une vidéo, le bloc 1 compare le résultat du détecteur qui est la position du visage dans l'image aux éléments d'un vecteur référence, que nous avons préalablement construit, contenant la position du visage de chaque image

de la vidéo. Ici, nous avons manuellement construit, les vecteurs références de 18 vidéos de la MMI que nous avons exploités pour le test. Comme le visage des personnes participants aux vidéos MMI se trouve aligné au centre de l'image, on peut facilement définir la position du visage pour toutes les images de vidéos, et ce, en déterminant la position du visage dans la première image de la vidéo. Dans le cas où la position du visage donnée par le détecteur ne correspond pas à la position du visage dans le vecteur de référence, la reconnaissance de l'expression du visage échoue. Dans ce cas la sortie du bloc 1 est un faux positif F_{p_i}, dont le détecteur déclare avoir trouvé un visage alors qu'il n'existe pas dans l'image, ou un faux négatif F_{n_i}, à ce niveau le détecteur estime que le visage n'existe pas alors qu'il existe. Les résultats F_{p_i} et F_{n_i} sont ensuite enregistrés comme des sorties négatives "negatif outputs" pour être exploitées dans le calcul de performance de reconnaissance. Sinon, si la position du visage trouvé par le détecteur correspond à celle du vecteur de référence, le visage détecté passe par les étapes restantes du pipeline, jusqu'au deuxième bloc qui évalue les résultats de reconnaissance de l'émotion.

4.3.2 Évaluation du classifieur : bloc 2

Dans ce bloc, l'évaluation est réalisée en comparant le résultat de classification au niveau de chaque image de la vidéo et l'étiquette dans un vecteur référence, que nous avons aussi préalablement déterminé pour chacune des 18 vidéos. La construction du vecteur référence pour les vidéos, nécessite la détermination, manuelle, de deux transitions de l'expression du visage T_1 et T_2. Un exemple de construction du vecteur référence est présenté dans la figure 4-17.

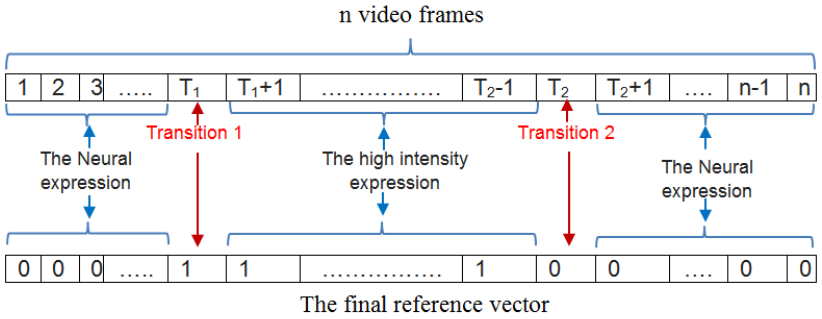


FIGURE 4-17 – Schéma simplifié montrant l'étape de construction du vecteur de référence

Dans la figure 4-17, T_1 et T_2 sont déterminés de sorte que, d'une part, les images entre les transitions T_1 et T_2 prennent une étiquette égale à y_i de l'une des 6 émotions suivantes : la

joie, la surprise, la colère, la tristesse, la peur et le dégoût et, d'autre part, les images à gauche de T_1 et à droite de T_2 prennent une étiquette de l'expression neutre. Par exemple, dans la figure 4-18, la première transition T_1 représente le passage de l'état neutre à l'expression de la colère, tandis que la deuxième transition T_2 correspond au retour à l'état neutre.

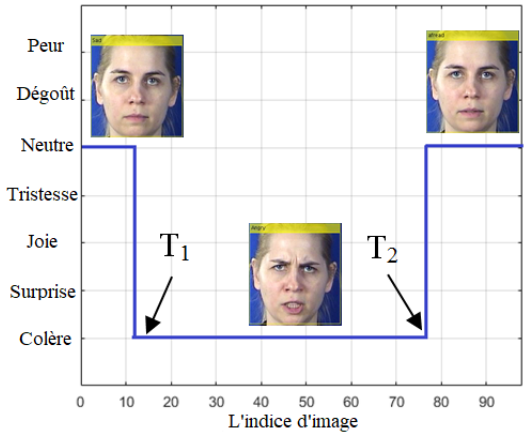


FIGURE 4-18 – Vecteur de référence de l'expression de la colère construit à partir de la vidéo "S001-100.avi"

En effet, dans la plupart des séquences de la base MMI la durée moyenne de l'émotion est égale à 6 secondes, elle commence par une expression neutre qui change après quelques secondes pour achever une expression de haute intensité, puis revient à l'état initial qui représente l'expression neutre. Cependant, dans quelques vidéos, l'expression commence et se termine par une intensité minimale de l'expression telle que le cas dans la figure 4-19.

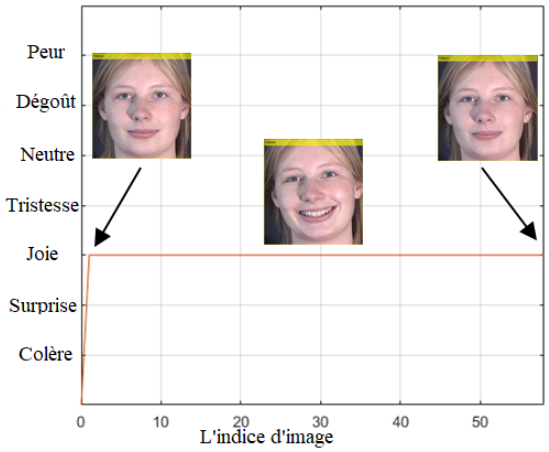


FIGURE 4-19 – Résultats de reconnaissance en utilisant la vidéo "S036-005.avi" tirée de la base de données MMI

Pour réaliser l'évaluation nous supposons qu'une séquence est correctement classée lorsque les images contenant l'expression neutre sont correctement reconnues et les images contenant l'émotion de base de la vidéo sont également correctement classées par le classifieur. Finalement, pour toutes les vidéos exploitées dans l'expérimentation, le vecteur référence est construit manuellement, conformément aux transitions T_1 et T_2, puis stocké dans le bloc 2 afin que ce dernier puisse comparer la sortie du classifieur $\widehat{y}_i$ avec les éléments du vecteur de référence y_i comme suit :

$$S_i = \begin{cases} y_i & if\ \widehat{y}_i = y_i \\ \widehat{y}_i & if\ \widehat{y}_i \neq y_i \end{cases} \tag{4.9}$$

y_i est l'émotion dans le vecteur de référence à la ième image, $\widehat{y}_i$ est l'émotion identifiée par le classifieur à la même image.

Le bloc peut décider si l'expression est correcte ou non en utilisant la condition suivante. Si $S_i = y_i$ alors le S_i résultant est positif, ce qui signifie que l'expression du visage est correctement classée (voir schéma 4-16). Sinon, si $S_i = \widehat{y}_i$ alors S_i est une sortie négative, alors l'émotion est mal classée.

4.4 Résultats de l'évaluation et discussion

Dans l'ensemble, les sorties positives et négatives sont utilisées pour construire le tableau de contingence et calculer la métrique F1-mesure [182]. Le tableau 4.2 rapporte le tableau de contingence résultante après l'évaluation automatique du pipeline en exploitant 18 vidéos extraites de la base de données MMI.

TABLE 4.2 – le résultat de l'évaluation pipeline résumé dans le tableau de contingence.

	joie	surprise	peur	dégoût	colère	tristesse	neutre
Joie	95.7%	0	3.1%	0	0	0	1.2%
Surprise	1.3%	96%	1.9%	0	0	0	0.8%
peur	5.12%	7.01%	85.09%	0	0.08%	0.1%	2.6%
dégoût	0	0	0	89.92%	3.58%	3.1%	3.4%
colère	0	0	0.84%	1.6%	91.1%	4.02%	2.44%
tristesse	0	0	1.3%	2.9%	8%	83.49%	4.31%
neutre	0.3%	0.3%	5.5%	0.9%	8.94%	10.23%	73.83%

Sachant que le classifieur est entraîné par les bases de données CK+ et "Yale Face", l'évaluation avec des vidéos de la base MMI a montré que le pipeline est précis lorsqu'il

classifie l'expression de la joie, la surprise et la colère. Ces trois expressions ont des scores supérieurs à 90%, ce qui est normal, car elles sont des expressions faciles à identifier [183,184].

Cependant, le score relatif aux expressions de la peur, le dégoût et la tristesse est borné entre 80% et 90%. Il peut y avoir plusieurs raisons pour lesquelles certaines expressions ne sont pas correctement classées. La première raison est la faible intensité des expressions lorsqu'on passe d'une expression à l'autre, particulièrement lors de la transition de l'expression. Une autre raison est les similitudes dans l'apparence de certains traits du visage au niveau de ces trois expressions, comme par exemple quand les lèvres sont rétrécies et les sourcils sont abaissés, cela crée une confusion entre les expressions de dégoût, de colère et de tristesse.

Le pipeline a été conçu aussi pour visualiser le résultat de reconnaissance pendant le fonctionnement du pipeline au cours du temps. Nous démontrons ceci dans les figures 4-19 et 4-20. Au niveau de la figure 4-20, une classification correcte est obtenue pour les images contenant l'expression de base, alors que la classification est instable en raison de l'incertitude du pipeline lors du changement de l'expression du visage.

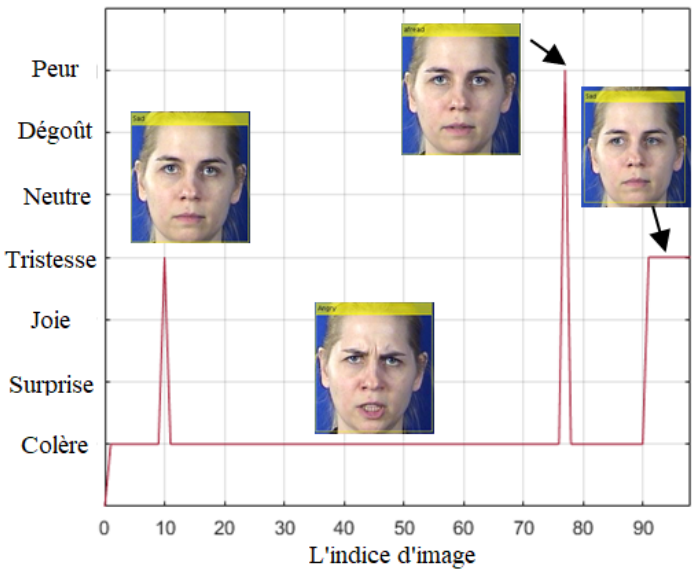


FIGURE 4-20 – Résultat de reconnaissance de l'émotion du visage au cours du temps, que nous avons obtenu pour la vidéo "S001-100.avi" de la base de données MMI

Le pipeline proposé est conçu pour reconnaître l'émotion de multiples personnes dans le cas où plus d'une seule personne apparaissent dans la scène vidéo. Un exemple de reconnaissance d'expression sur quatre images extraites d'une vidéo est présenté dans la figure 4-21, où trois personnes sont filmées à l'aide de la caméra frontale d'un téléphone portable.

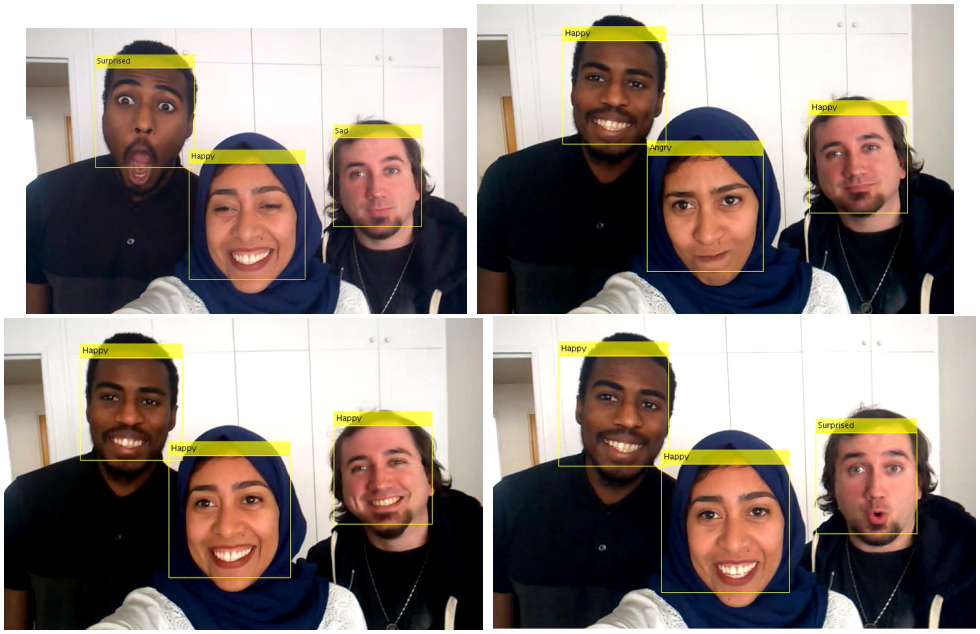


FIGURE 4-21 – Reconnaissance de l'expression après la mise en œuvre pipeline sur une vidéo filmée par une caméra frontale d'un téléphone portable

Après avoir testé le pipeline par certaines vidéos du monde réel, il convient de noter que l'analyse d'image atteint un temps de moyen de 0,018 secondes pour un visage. Le coût en termes de temps est mesuré en considérant une moyenne de 100 images tout en utilisant des processeurs Intel Core i7, CPU @ 2,20 GHz et une mémoire RAM de 16 Go. Cela signifie que le pipeline peut être exécuté pour une cadence vidéo de 24 fps. Cependant, plus les visages sont multipliés dans la scène filmée, plus le pipeline consomme de temps. En effet, le temps d'analyse d'images varie en fonction de certaines variables qui sont la résolution de l'image et la complexité du fond de la scène. Cela signifie que dans le cas de scènes contenant des régions du visage de grande dimension ou contenant de nombreux objets en mouvement en arrière-plan, le pipeline prend beaucoup de temps durant l'analyse des images de la scène.

En résumé, après l'analyse des données effectuées dans le chapitre 3 et l'évaluation du pipeline proposé, nous soulignons qu'une réduction potentielle des caractéristiques HOG par l'algorithme de Fisher peut accélérer le fonctionnement du pipeline avec le maintien d'un score de reconnaissance d'émotion assez élevé.

4.5 conclusion

En ajoutant des étapes de prétraitement de détection et du suivi du visage ainsi que de segmentation des régions des yeux et de la bouche au modèle d'analyse obtenue dans le chapitre 3, nous avons proposé un pipeline pour identifier les expressions universelles des personnes dans des vidéos du monde réel.

En exploitant le pipeline, nous avons pu analyser la relation géométrique entre les clusters construits par la méthode de Fisher. Cela est réalisé en testant et évaluant des techniques de séparation des clusters linéaires et non linéaires. Nous avons montré qu'une séparation linéaire des clusters du classifieur, par la technique d'analyse discriminante linéaire, rend le classifieur d'expressions performant en termes de taux reconnaissance et en complexité de calcul.

Nous avons validé le pipeline en proposant une procédure d'évaluation du fonctionnement du pipeline par exploitation des vidéos au lieu des images. En calculant le nombre d'occurrences des expressions identifiées et non identifiées dans les séquences vidéo, nous avons établi le tableau de contingence et calculer le taux de reconnaissance du pipeline qui vaut 87.88%. Pour améliorer le pipeline, nous avons ajouté un code qui aide à enregistrer les images des expressions qui ne sont pas identifiées, les ajouter à la base d'entraînement et refaire l'apprentissage du classifieur.

Conclusions et perspectives

Pour conclure, nous présentons un extrait de rappel concernant la problématique de la thèse ainsi qu'un résumé des chapitres de ce manuscrit. Ensuite, nous proposons des pistes de recherche que ce soit des recherches en cours ou pour l'avenir, qui mettent en perspective les travaux présentés dans cette thèse.

1 Rappel de la problématique

L'objectif de cette thèse est le développement d'un système de vision par ordinateur pour la reconnaissance d'émotion universelle des personnes dans une vidéo. Cela signifie que l'analyse d'image devrait être effectuée en temps correspondant à la cadence vidéo. Sachant que la cadence de la vidéo diffère d'une caméra à l'autre, la limite inférieure pour obtenir une vidéo fluide correspond à l'ajustement de l'enregistrement vidéo à 24 images par second. À cette vitesse, l'analyse d'image qui comprend les étapes de la détection et la description du visage et la reconnaissance de l'expression devrait être rapide, tout en finissant l'analyse d'image en une durée de 0.042 secondes quelle que soit la complexité des images dans la scène vidéo. Donc pour chercher le modèle d'analyse approprié à cette tâche, il est primordial d'effectuer une étape d'analyse d'images des expressions d'émotion par une variété de descripteurs décrivant différentes propriétés visuelles telles que la forme les contours et la texture, ainsi qu'une variété de méthodes de classification. Pour réussir l'analyse des images, une recherche exhaustive de valeurs optimales de paramètres des descripteurs et des hyperparamètres des méthodes de classification devrait être effectuée. L'existence d'un nombre important de variables à gérer durant l'analyse, exige de bâtir un algorithme permettant d'automatiser non seulement la recherche des valeurs optimales des paramètres et hyperparamètres. Mais aussi de trouver la combinaison descripteur-classifieur constituant le modèle d'analyse qui renvoie un taux de reconnaissance maximal et un temps de traitement

minimale.

2 Résumé des chapitres

2.1 Reconnaissance des émotions humaines : état d'art

Dans le premier chapitre, nous avons commencé par déterminer les classes d'émotions qu'on peut exploiter, en passant en revue les études interculturelles réalisées dans le domaine psychologique. Ces études ont pu fournir des bases de données que les chercheurs dans le domaine de la vision par ordinateur l'utilisent actuellement afin de proposer des solutions pour le problème de la reconnaissance d'émotions. Nous avons aussi passé en revue plusieurs techniques de traitement et d'analyse d'images existantes dans la littérature qui permettent de construire un système de reconnaissance d'émotions universelles. Après l'étude bibliographique réalisée dans ce chapitre, nous avons pu déterminer la piste à suivre pour bâtir notre solution, que nous avons développée sur trois grandes étapes qui sont : l'étape de la détection du visage, l'étape d'extraction du vecteur caractéristique et l'étape de la classification de l'émotion.

2.2 Analyse de données par des descripteurs du visage et des méthodes de classification

Dans ce chapitre, nous avons donné le cadre théorique des méthodes que nous avons pré-sélectionnées pour accomplir l'étape d'analyse d'images des expressions. À ce niveau, nous avons décrit le fonctionnement des descripteurs que nous avons choisis dans le but d'extraire les vecteurs de caractéristiques qui décrivent la forme, la texture et les contours du visage. Ensuite, nous avons introduit cinq algorithmes de classification en décrivant les fonctions mathématiques qui s'utilisent pour la redescription des données d'apprentissage et la construction un classifieur d'émotion. L'intérêt d'utiliser une diversité d'algorithmes de classification est qu'ils servent à tester les différentes fonctions de séparation de classes pour comprendre la structure de données pendant l'étape d'analyse d'images.

2.3 Recherche du descripteur et du classifieur d'expression : vers une analyse d'images entièrement automatisée

Dans ce chapitre nous avons proposé un algorithme qui automatise le processus d'analyse d'images. Pendant l'analyse, l'algorithme commence par une recherche exhaustive de valeurs optimales des paramètres des descripteurs et des hyperparamètres des méthodes de classification. Ensuite, il identifie le meilleur modèle d'analyse finale, en termes de taux reconnaissance et temps de classification.

2.4 Le pipeline proposé pour la reconnaissance d'émotions dans une vidéo

En se basant sur le résultat de l'analyse du chapitre précédent, dans ce chapitre nous avons bâti un pipeline capable d'identifier les émotions universelles de plusieurs personnes dans des vidéos. Ensuite, pour valider le pipeline nous avons élaboré un procédé de test et d'évaluation du fonctionnement du pipeline sur un ensemble de vidéo.

3 Perspectives

3.1 Au niveau du pipeline : en cours

Le test du fonctionnement du pipeline a soulevé des points qui devraient être améliorés. Il est nécessaire d'exécuter le pipeline en permanence, dans des scènes réelles, que l'expression faciale soit une expression universelle ou non. Pour cela, nous avons pensé à l'amélioration du pipeline pour qu'il puisse reconnaître d'autres expressions telles que l'expression de la fatigue, visant par ceci une application dans le domaine de la conduite intelligente . Ainsi, nous travaillons sur le codage des mouvements des traits du visage pour pouvoir identifier les micro-expressions. Pour ce faire, nous nous basons sur le système FAC [185] qui décode tous les mouvements possibles des muscles du visage en les classifiant, théoriquement, en un ensemble d'unités d'action AU (Action Unit en anglais), dont chaque AU décrit un mouvement spécifique d'une région particulière du visage comme le montre le tableau 4.3.

TABLE 4.3 – Liste de quelques unités d'action qui codent les mouvements subtils des traits du visage

Unité d'action	**Mouvement associé à l'unité d'action**
AU_1	Remontée de la partie interne des sourcils
AU_2	Remontée de la partie externe des sourcils
AU_4	Abaissement et rapprochement des sourcils
AU_5	Ouverture entre la paupière supérieure et les sourcils
AU_6	Remontée des joues
AU_7	Tension de la paupière
AU_9	Plissement de la peau du nez vers le haut
AU_{12}	Étirement du coin des lèvres
AU_{15}	Abaissement des coins externes des lèvres
AU_{16}	Ouverture de la lèvre inférieure
AU_{20}	Étirement externe des lèvres
AU_{23}	Tension refermante des lèvres
AU_{26}	Ouverture de la mâchoire

Cela nous permettra de rendre le pipeline capable non seulement d'analyser l'émotion de la personne, mais aussi de générer des visages virtuels qui imitent les mouvements des utilisateurs.

Pour pouvoir réaliser ceci, techniquement nous allons améliorer la méthode d'extraction du vecteur caractéristique introduite dans la section 4.2.3. Notre vision est qu'au lieu d'extraire

les caractéristiques de HOG sur toutes les régions d'intérêt, nous proposons les extraire au niveau des cercles en rouges dans la figure 4-22. Cela nous aidera à voir le changement de l'orientation des contours lorsque l'expression du visage change. En appréhendant comment l'orientation des contours change, on pourra quantifier les unités d'action du tableau 4.3

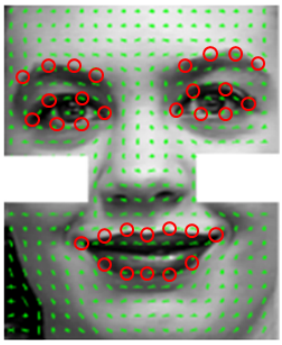


FIGURE 4-22 – L'amélioration que l'on envisage à réaliser

3.2 Au niveau de l'algorithme d'analyse d'images

La mise en œuvre de l'algorithme qui automatise d'analyse des images nous a montré les points à améliorer au niveau de l'algorithme. Nous proposons les deux modifications suivantes.

3.2.1 Réduction du temps de l'analyse : utilisation du parallélisme

Nous envisageons d'utiliser le parallélisme dans l'objectif de pouvoir réduire le temps d'analyse qui est assez élevé : à l'ordre de quelques semaines. En effet, l'algorithme du réglage et d'évaluation des modèles d'analyse repose sur un processus opérationnel qui est parallèle : il ne nécessite aucune intercommunication, car les exécutions des tâches sont indépendantes sur le même ensemble de données. Deux processus particuliers de cette nature sont le balayage des paramètres et les tests. Dans les balayages de paramètres (Voir section 3.3), le modèle d'analyse est exécuté plusieurs fois sur le même ensemble de données avec des paramètres différents, suivi d'une évaluation sur un ensemble de validation. Pendant les procédures de test par la technique de validation croisée (Voir section 3.1.1.1), les entraînements et les tests du modèle d'analyse sont effectués à plusieurs reprises sur différents sous-ensembles de données. L'utilité du parallélisme est évidente pour la tâche du réglage et d'évaluation des modèles d'analyse.

3.2.2 Recherche des paramètres et hyperparamètres optimaux des modèles d'analyse

Dans la section 3.3.4 nous avons réalisé une recherche exhaustive des paramètres et hyperparamètres optimaux, par laquelle nous testons toutes les combinaisons possibles des valeurs des paramètres et hyperparamètres. À ce niveau, nous pouvons penser à réaliser une sélection arbitraire d'une combinaison des valeurs de la grille B et ensuite la tester sur le modèle d'analyse. Si ce dernier donne un score élevé, on continue le test des autres combinaisons qui se trouvent au voisinage de la combinaison déjà testée. Sinon on continue la sélection arbitraire. Ici, on peut mettre un critère d'arrêt de test lorsque la combinaison sélectionnée donne un score satisfaisant à l'utilisateur.

Bibliographie

[1] Alexis Zubiolo. *Extraction de caractéristiques et apprentissage statistique pour l'imagerie biomédicale cellulaire et tissulaire*. PhD thesis, Nice, 2015.

[2] Paul Viola and Michael Jones. Robust real-time face detection. In *null*, page 747. Citeseer, 2001.

[3] Carlo Tomasi and Takeo Kanade. Detection and tracking of point features. 1991.

[4] Vinay P Kumar and Tomaso Poggio. Learning-based approach to real time tracking and analysis of faces. In *Automatic Face and Gesture Recognition, 2000. Proceedings. Fourth IEEE International Conference on*, pages 96–101. IEEE, 2000.

[5] Marco Leo, Pierluigi Carcagnì, Pier Luigi Mazzeo, Paolo Spagnolo, Dario Cazzato, and Cosimo Distante. Analysis of facial information for healthcare applications : A survey on computer vision-based approaches. *Information*, 11(3) :128, 2020.

[6] Paul Viola and Michael J Jones. Robust real-time face detection. *International journal of computer vision*, 57(2) :137–154, 2004.

[7] Daniel McDuff, Rana Kaliouby, Thibaud Senechal, May Amr, Jeffrey Cohn, and Rosalind Picard. Affectiva-mit facial expression dataset (am-fed) : Naturalistic and spontaneous facial expressions collected. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 881–888, 2013.

[8] Adam Yala, Constance Lehman, Tal Schuster, Tally Portnoi, and Regina Barzilay. A deep learning mammography-based model for improved breast cancer risk prediction. *Radiology*, page 182716, 2019.

[9] Samta Jain Goyal, Arvind K Upadhyay, RS Jadon, and Rajeev Goyal. Real-life facial expression recognition systems : a review. In *Smart Computing and Informatics*, pages 311–331. Springer, 2018.

[10] Michael J Black and Yaser Yacoob. Recognizing facial expressions in image sequences using local parameterized models of image motion. *International Journal of Computer Vision*, 25(1) :23–48, 1997.

[11] *Dafex : Un database di espressioni facciali dinamiche*, 2004.

[12] Patrick Lucey, Jeffrey F Cohn, Takeo Kanade, Jason Saragih, Zara Ambadar, and Iain Matthews. The extended cohn-kanade dataset (ck+) : A complete dataset for action unit and emotion-specified expression. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2010 IEEE Computer Society Conference on*, pages 94–101. IEEE, 2010.

[13] Michel Valstar and Maja Pantic. Induced disgust, happiness and surprise : an addition to the mmi facial expression database. In *Proc. 3rd Intern. Workshop on EMOTION (satellite of LREC) : Corpora for Research on Emotion and Affect*, page 65. Paris, France, 2010.

[14] Kathrin Kaulard, Douglas W Cunningham, Heinrich H Bülthoff, and Christian Wallraven. The mpi facial expression database—a validated database of emotional and conversational facial expressions. *PloS one*, 7(3), 2012.

[15] P Ekman and WV Friesen. Pictures of facial affect consulting psychologists press. *Palo Alto, CA*, 1976.

[16] A Georghiades, P Belhumeur, and D Kriegman. Yale face database. *Center for computational Vision and Control at Yale University, http ://cvc. yale. edu/projects/yalefaces/yalefa*, 2 :6, 1997.

[17] Daniel Lundqvist, Anders Flykt, and Arne Öhman. The karolinska directed emotional faces (kdef). *CD ROM from Department of Clinical Neuroscience, Psychology section, Karolinska Institutet*, 91(630) :2–2, 1998.

[18] Hwang Bon-Woo, Hyeran Byun, Roh Myoung-Cheol, and Lee Seong-Whan. Performance evaluation of face recognition algorithms on the asian face database, kfdb. In *International Conference on Audio-and Video-Based Biometric Person Authentication*, pages 557–565. Springer, 2003.

[19] Meredith Minear and Denise C Park. A lifespan database of adult facial stimuli. *Behavior Research Methods, Instruments and Computers*, 36(4) :630–633, 2004.

[20] Wen Gao, Bo Cao, Shiguang Shan, Xilin Chen, Delong Zhou, Xiaohua Zhang, and Debin Zhao. The cas-peal large-scale chinese face database and baseline evaluations. *IEEE Transactions on Systems, Man, and Cybernetics-Part A : Systems and Humans*, 38(1) :149–161, 2007.

[21] Liang Gao, Jinyang Liang, Chiye Li, and Lihong V Wang. Single-shot compressed ultrafast photography at one hundred billion frames per second. *Nature*, 516(7529) :74, 2014.

[22] PWW Fuller. An introduction to high speed photography and photonics, 2009.

[23] Paul Read and Mark-Paul Meyer. *Restoration of motion picture film*. Elsevier, 2000.

[24] Giuseppe Papari and Nicolai Petkov. Edge and line oriented contour detection : State of the art. *Image and Vision Computing*, 29(2-3) :79–103, 2011.

[25] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *international Conference on computer vision & Pattern Recognition (CVPR'05)*, volume 1, pages 886–893. IEEE Computer Society, 2005.

[26] Di Huang, Caifeng Shan, Mohsen Ardabilian, Yunhong Wang, and Liming Chen. Local binary patterns and its application to facial image analysis : a survey. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 41(6) :765–781, 2011.

[27] Albert Cruz, Bir Bhanu, and Ninad S Thakoor. Facial emotion recognition with anisotropic inhibited gabor energy histograms. In *2013 IEEE International Conference on Image Processing*, pages 4215–4219. IEEE, 2013.

[28] Paul Ekman and Wallace V Friesen. Constants across cultures in the face and emotion. *Journal of personality and social psychology*, 17(2) :124, 1971.

[29] Paul Ekman. Strong evidence for universals in facial expressions : A reply to russell's mistaken critique. 1994.

[30] Pamela J Naab and James A Russell. Judgments of emotion from spontaneous facial expressions of new guineans. *Emotion*, 7(4) :736, 2007.

[31] M Beaupré, N Cheung, and U Hess. La reconnaissance des expressions émotionnelles faciales par des décodeurs africains, asiatiques, et caucasiens. In *Poster presented at the annual meeting of the Société Québécoise pour la Recherche en Psychologie, Hull, Quebec, Canada*, 2000.

[32] Takeo Kanade, Jeffrey F Cohn, and Yingli Tian. Comprehensive database for facial expression analysis. In *Proceedings Fourth IEEE International Conference on Automatic Face and Gesture Recognition (Cat. No. PR00580)*, pages 46–53. IEEE, 2000.

[33] Li-Fen Chen and Yu-Shiuan Yen. Taiwanese facial expression image database. *Brain Mapping Laboratory, Institute of Brain Science, National Yang-Ming University, Taipei, Taiwan*, 2007.

[34] Matthew N. Dailey, Garrison W. Cottrell, and Judith Reilly. CAlifornia Facial Expressions (CAFE), 2001.

[35] Alice J O'Toole, Joshua Harms, Sarah L Snow, Dawn R Hurst, Matthew R Pappas, Janet H Ayyad, and Hervé Abdi. A video database of moving faces and people. *IEEE Transactions on pattern analysis and machine intelligence*, 27(5) :812–816, 2005.

[36] Maja Pantic, Michel Valstar, Ron Rademaker, and Ludo Maat. Web-based database for facial expression analysis. In *2005 IEEE international conference on multimedia and Expo*, page 5. IEEE, 2005.

[37] Paul Ekman, Wallace V Friesen, Maureen O'sullivan, Anthony Chan, Irene Diacoyanni-Tarlatzis, Karl Heider, Rainer Krause, William Ayhan LeCompte, Tom Pitcairn, Pio E Ricci-Bitti, et al. Universals and cultural differences in the judgments of facial expressions of emotion. *Journal of personality and social psychology*, 53(4) :712, 1987.

[38] Mohammad Soleymani, Martha Larson, Thierry Pun, and Alan Hanjalic. Corpus development for affective video indexing. *IEEE Transactions on Multimedia*, 16(4) :1075–1089, 2014.

[39] William James. What is an emotion ? *Mind*, 9(34) :188–205, 1884.

[40] John Broadus Watson. Behaviorism, rev. 1930.

[41] Orval Mowrer. Learning theory and behavior. 1960.

[42] Magda B Arnold. Emotion and personality. 1960.

[43] Plutchik Robert. Emotion : Theory, research, and experience. vol. 1 : Theories of emotion. 1980.

[44] Robert Plutchik. The nature of emotions : Human emotions have deep evolutionary roots, a fact that may explain their complexity and provide tools for clinical practice. *American scientist*, 89(4) :344–350, 2001.

[45] Jeffrey A Gray. Précis of the neuropsychology of anxiety : An enquiry into the functions of the septo-hippocampal system. *Behavioral and Brain Sciences*, 5(3) :469–484, 1982.

[46] Jaak Panksepp. Toward a general psychobiological theory of emotions. *Behavioral and Brain sciences*, 5(3) :407–422, 1982.

[47] Paul Ekman. What emotion categories or dimensions can observers judge from facial behavior ? *Emotions in the human face*, pages 39–55, 1982.

[48] SS Tomkins. Affect theory. a kr scherer i p. ekman (eds.), approaches to emotion. *Hill sdale : Lawrence Erlbaum*, 1984.

[49] Bernard Weiner and Sarah Graham. An attributional approach to emotional development. *Emotions, cognition, and behavior*, pages 167–191, 1984.

[50] Keith Oatley and Philip N Johnson-Laird. Towards a cognitive theory of emotions. *Cognition and emotion*, 1(1) :29–50, 1987.

[51] Nico H Frijda, Batja Mesquita, Joep Sonnemans, and Stephanie Van Goozen. The duration of affective phenomena or emotions, sentiments and passions. 1991.

[52] Carroll E Izard. *The psychology of emotions*. Springer Science & Business Media, 1991.

[53] William McDougall. *An introduction to social psychology*. Psychology Press, 2015.

[54] E Friesen and P Ekman. Facial action coding system : a technique for the measurement of facial movement. *Palo Alto*, 1978.

[55] Dawood Adel AL CHANTI and Alice Caplier. Deep learning for spatio-temporal modeling of dynamic spontaneous emotions. *IEEE Transactions on Affective Computing*, 2018.

[56] AM Martinez and R Benavente. The ar face database. computer vision center(cvc), barcelona, spain : Technical report 24, 1998.

[57] David Matsumoto. Japanese and caucasian facial expressions of emotion (jacfee) and neutral faces (jacneuf). *Intercultual and Emotion Research Laboratory, Department of Psychology*, 1988.

[58] Christine L Lisetti and Diane J Schiano. Automatic facial expression interpretation : Where human-computer interaction, artificial intelligence and cognitive science intersect. *Pragmatics & cognition*, 8(1) :185–235, 2000.

[59] Ryohei Nakatsu, Joy Nicholson, and Naoko Tosa. Emotion recognition and its application to computer agents with spontaneous interactive capabilities. In *Proceedings of the seventh ACM international conference on Multimedia (Part 1)*, pages 343–351, 1999.

[60] Yisi Liu, Olga Sourina, and Minh Khoa Nguyen. Real-time eeg-based emotion recognition and its applications. In *Transactions on computational science XII*, pages 256–277. Springer, 2011.

[61] Hyung-Soo Lee and Daijin Kim. Expression-invariant face recognition by facial expression transformations. *Pattern recognition letters*, 29(13) :1797–1805, 2008.

[62] Wei Quan, Bogdan J Matuszewski, Lik-Kwan Shark, and Djamel Ait-Boudaoud. Facial expression biometrics using statistical shape models. *EURASIP Journal on Advances in Signal Processing*, 2009(1) :261542, 2009.

[63] Shiyang Cheng, Irene Kotsia, Maja Pantic, and Stefanos Zafeiriou. 4dfab : A large scale 4d database for facial expression analysis and biometric applications. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5117–5126, 2018.

[64] Seth D Pollak and Pawan Sinha. Effects of early experience on children's recognition of facial displays of emotion. *Developmental psychology*, 38(5) :784, 2002.

[65] Min Fan, Jianyu Fan, Sheng Jin, Alissa N Antle, and Philippe Pasquier. Emostory : A game-based system supporting children's emotional development. In *Extended Abstracts of the 2018 CHI Conference on Human Factors in Computing Systems*, pages 1–6, 2018.

[66] Sheryl Brahnam, Chao-Fa Chuang, Frank Y Shih, and Melinda R Slack. Machine recognition and representation of neonatal facial displays of acute pain. *Artificial intelligence in medicine*, 36(3) :211–222, 2006.

[67] Ciprian Adrian Corneanu, Marc Oliu Simón, Jeffrey F Cohn, and Sergio Escalera Guerrero. Survey on rgb, 3d, thermal, and multimodal approaches for facial expression recognition : History, trends, and affect-related applications. *IEEE transactions on pattern analysis and machine intelligence*, 38(8) :1548–1568, 2016.

[68] Guangzheng Yang and Thomas S Huang. Human face detection in a complex background. *Pattern recognition*, 27(1) :53–63, 1994.

[69] Kin Choong Yow and Roberto Cipolla. Feature-based human face detection. *Image and vision computing*, 15(9) :713–736, 1997.

[70] Christophe Garcia and George Tziritas. Face detection using quantized skin color regions merging and wavelet packet analysis. *IEEE Transactions on multimedia*, 1(3) :264–277, 1999.

[71] Rein-Lien Hsu, Mohamed Abdel-Mottaleb, and Anil K Jain. Face detection in color images. *IEEE transactions on pattern analysis and machine intelligence*, 24(5) :696–706, 2002.

[72] Lin-Lin Huang, Akinobu Shimizu, and Hidefumi Kobatake. Robust face detection using gabor filter features. *Pattern Recognition Letters*, 26(11) :1641–1649, 2005.

[73] Kobi Levi and Yair Weiss. Learning object detection from a small number of examples : the importance of good features. In *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, volume 2, pages II–II. IEEE, 2004.

[74] Hongming Zhang, Wen Gao, Xilin Chen, and Debin Zhao. Object detection using spatial histogram features. *Image and Vision Computing*, 24(4) :327–341, 2006.

[75] Xiaoyu Wang, Tony X Han, and Shuicheng Yan. An hog-lbp human detector with partial occlusion handling. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 32–39. IEEE, 2009.

[76] Rainer Lienhart and Jochen Maydt. An extended set of haar-like features for rapid object detection. In *Proceedings. international conference on image processing*, volume 1, pages I–I. IEEE, 2002.

[77] Dario Maio and Davide Maltoni. Real-time face location on gray-scale static images. *Pattern Recognition*, 33(9) :1525–1539, 2000.

[78] K-K Sung and Tomaso Poggio. Example-based learning for view-based human face detection. *IEEE Transactions on pattern analysis and machine intelligence*, 20(1) :39–51, 1998.

[79] Robert E Schapire. Explaining adaboost. In *Empirical inference*, pages 37–52. Springer, 2013.

[80] Jürgen Schmidhuber. Deep learning in neural networks : An overview. *Neural networks*, 61 :85–117, 2015.

[81] Henry A Rowley, Shumeet Baluja, and Takeo Kanade. Neural network-based face detection. *IEEE Transactions on pattern analysis and machine intelligence*, 20(1) :23–38, 1998.

[82] Christophe Garcia and Manolis Delakis. Convolutional face finder : A neural architecture for fast and robust face detection. *IEEE Transactions on pattern analysis and machine intelligence*, 26(11) :1408–1423, 2004.

[83] Alper Yilmaz, Omar Javed, and Mubarak Shah. Object tracking : A survey. *Acm computing surveys (CSUR)*, 38(4) :13, 2006.

[84] Patrick Pérez, Carine Hue, Jaco Vermaak, and Michel Gangnet. Color-based probabilistic tracking. In *European Conference on Computer Vision*, pages 661–675. Springer, 2002.

[85] Abdullah Bulbul, Zeynep Cipiloglu, and Tolga Capin. A color-based face tracking algorithm for enhancing interaction with mobile devices. *The Visual Computer*, 26(5) :311–323, 2010.

[86] Chris Harris and Mike Stephens. A combined corner and edge detector. In *Alvey vision conference*, volume 15, pages 10–5244. Citeseer, 1988.

[87] Pierluigi Carcagnì, Marco Del Coco, Marco Leo, and Cosimo Distante. Facial expression recognition and histograms of oriented gradients : a comprehensive study. *SpringerPlus*, 4(1) :645, 2015.

[88] Aliaa AA Youssif and Wesam AA Asker. Automatic facial expression recognition system based on geometric and appearance features. *Computer and Information Science*, 4(2) :115, 2011.

[89] Manar MF Donia, Aliaa AA Youssif, and Atallah Hashad. Spontaneous facial expression recognition based on histogram of oriented gradients descriptor. *Computer and Information Science*, 7(3) :31, 2014.

[90] Khadija Lekdioui, Rochdi Messoussi, Yassine Ruichek, Youness Chaabi, and Raja Touahni. Facial decomposition for expression recognition using texture/shape des-

criptors and svm classifier. *Signal Processing : Image Communication*, 58 :300–312, 2017.

[91] Xuehan Xiong and Fernando De la Torre. Supervised descent method and its applications to face alignment. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 532–539, 2013.

[92] Latifa Greche, Najia Es-Sbai, and Egons Lavendelis. Performance review of a multilayer feed-forward neural network and normalized cross correlation for facial expression identification. In *2016 12th International Conference on Signal-Image Technology & Internet-Based Systems (SITIS)*, pages 223–229. IEEE, 2016.

[93] Xiaohua Huang, Guoying Zhao, Wenming Zheng, and Matti Pietikäinen. Towards a dynamic expression recognition system under facial occlusion. *Pattern Recognition Letters*, 33(16) :2181–2191, 2012.

[94] Latifa Greche, Maha Jazouli, Najia Es-Sbai, Aicha Majda, and Arsalane Zarghili. Comparison between euclidean and manhattan distance measure for facial expressions classification. In *2017 International Conference on Wireless Technologies, Embedded and Intelligent Systems (WITS)*, pages 1–4. IEEE, 2017.

[95] Xiaoming Zhao and Shiqing Zhang. Facial expression recognition using local binary patterns and discriminant kernel locally linear embedding. *EURASIP journal on Advances in signal processing*, 2012(1) :20, 2012.

[96] Faisal Ahmed, Emam Hossain, ASM Hossain Bari, and ASM Shihavuddin. Compound local binary pattern (clbp) for robust facial expression recognition. In *Computational Intelligence and Informatics (CINTI), 2011 IEEE 12th International Symposium on*, pages 391–395. IEEE, 2011.

[97] Weilong Chen, Meng Joo Er, and Shiqian Wu. Illumination compensation and normalization for robust face recognition using discrete cosine transform in logarithm domain. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 36(2) :458–466, 2006.

[98] Huma Qayyum, Muhammad Majid, Syed Muhammad Anwar, and Bilal Khan. Facial expression recognition using stationary wavelet transform features. *Mathematical Problems in Engineering*, 2017, 2017.

[99] Zhengyou Zhang, Michael Lyons, Michael Schuster, and Shigeru Akamatsu. Comparison between geometry-based and gabor-wavelets-based facial expression recognition using multi-layer perceptron. In *Automatic Face and Gesture Recognition, 1998. Proceedings. Third IEEE International Conference on*, pages 454–459. IEEE, 1998.

[100] Mikel D Rodriguez, Javed Ahmed, and Mubarak Shah. Action mach a spatio-temporal maximum average correlation height filter for action recognition. In *Computer Vision*

and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on, pages 1–8. IEEE, 2008.

[101] Seyed Mehdi Lajevardi and Zahir M Hussain. Automatic facial expression recognition : feature extraction and selection. *Signal, Image and video processing*, 6(1) :159–169, 2012.

[102] Yongqiang Yao, Di Huang, Xudong Yang, Yunhong Wang, and Liming Chen. Texture and geometry scattering representation-based facial expression recognition in 2d+ 3d videos. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, 14(1s) :18, 2018.

[103] Mark Nixon and Alberto S Aguado. *Feature extraction and image processing for computer vision*. Academic Press, 2012.

[104] Timo Ojala, Matti Pietikäinen, and David Harwood. A comparative study of texture measures with classification based on featured distributions. *Pattern recognition*, 29(1) :51–59, 1996.

[105] Caifeng Shan, Shaogang Gong, and Peter W McOwan. Robust facial expression recognition using local binary patterns. In *Image Processing, 2005. ICIP 2005. IEEE International Conference on*, volume 2, pages II–370. IEEE, 2005.

[106] Xiaoyi Feng, M Pietikainen, and Abdenour Hadid. Facial expression recognition with local binary patterns and linear programming. *Pattern Recognition And Image Analysis C/C of Raspoznavaniye Obrazov I Analiz Izobrazhenii*, 15(2) :546, 2005.

[107] Caifeng Shan, Shaogang Gong, and Peter W McOwan. Facial expression recognition based on local binary patterns : A comprehensive study. *Image and vision Computing*, 27(6) :803–816, 2009.

[108] Xiaoyang Tan and Bill Triggs. Enhanced local texture feature sets for face recognition under difficult lighting conditions. *IEEE transactions on image processing*, 19(6) :1635–1650, 2010.

[109] Faisal Ahmed and Md Hasanul Kabir. Directional ternary pattern (dtp) for facial expression recognition. In *2012 IEEE International Conference on Consumer Electronics (ICCE)*, pages 265–266. IEEE, 2012.

[110] Taskeed Jabid, Md Hasanul Kabir, and Oksam Chae. Robust facial expression recognition based on local directional pattern. *ETRI journal*, 32(5) :784–794, 2010.

[111] Maitine Bergounioux. Quelques méthodes de filtrage en traitement d'image. 2011.

[112] D Kishore, S Srinivas Kumar, and Ch Srinivasa Rao. A neural network approach for content-based image retrieval using moments of image transforms. In *Soft Computing and Signal Processing*, pages 625–633. Springer, 2019.

[113] Dehai Zhang, Da Ding, Jin Li, and Qing Liu. Pca based extracting feature using fast fourier transform for facial expression recognition. In *Transactions on Engineering Technologies*, pages 413–424. Springer, 2015.

[114] Michael Lyons, Shigeru Akamatsu, Miyuki Kamachi, and Jiro Gyoba. Coding facial expressions with gabor wavelets. In *Automatic Face and Gesture Recognition, 1998. Proceedings. Third IEEE International Conference on*, pages 200–205. IEEE, 1998.

[115] Chengjun Liu and Harry Wechsler. Gabor feature based classification using the enhanced fisher linear discriminant model for face recognition. *IEEE Transactions on Image processing*, 11(4) :467–476, 2002.

[116] Bin Jiang, Guo-Sheng Yang, and Huan-Long Zhang. Comparative study of dimension reduction and recognition algorithms of dct and 2dpca. In *Machine Learning and Cybernetics, 2008 International Conference on*, volume 1, pages 407–410. IEEE, 2008.

[117] Sidra Batool Kazmi, M Arfan Jaffar, et al. Wavelets-based facial expression recognition using a bank of support vector machines. *Soft Computing*, 16(3) :369–379, 2012.

[118] Ayşegül Uçar, Yakup Demir, and Cüneyt Güzeliş. A new facial expression recognition based on curvelet transform and online sequential extreme learning machine initialized with spherical clustering. *Neural Computing and Applications*, 27(1) :131–142, 2016.

[119] Stefano Berretti, Boulbaba Ben Amor, Mohamed Daoudi, and Alberto Del Bimbo. 3d facial expression recognition using sift descriptors of automatically detected keypoints. *The Visual Computer*, 27(11) :1021, 2011.

[120] Latifa Greche and Najia Es-Sbai. Automatic system for facial expression recognition based histogram of oriented gradient and normalized cross correlation. In *2016 International Conference on Information Technology for Organizations Development (IT4OD)*, pages 1–5. IEEE, 2016.

[121] Latifa Greche, Najia Es-Sbai, and Egons Lavendelis. Histogram of oriented gradient and multi layer feed forward neural network for facial expression identification. In *2017 International Conference on Control, Automation and Diagnosis (ICCAD)*, pages 333–337. IEEE, 2017.

[122] Ahmet Kucuk, Juan M Banda, and Rafal A Angryk. A large-scale solar dynamics observatory image dataset for computer vision applications. *Scientific data*, 4 :170096, 2017.

[123] Ihsan Ullah, Muhammad Hussain, Ghulam Muhammad, Hatim Aboalsamh, George Bebis, and Anwar M Mirza. Gender recognition from face images with local wld descriptor. In *2012 19th International Conference on Systems, Signals and Image Processing (IWSSIP)*, pages 417–420. IEEE, 2012.

[124] Jeffrey F Cohn, Karen Schmidt, Ralph Gross, and Paul Ekman. Individual differences in facial expression : Stability over time, relation to self-reported emotion, and ability to inform person identification. In *Proceedings of the 4th IEEE International Conference on Multimodal Interfaces*, page 491. IEEE Computer Society, 2002.

[125] Guoying Zhao and Matti Pietikainen. Dynamic texture recognition using local binary patterns with an application to facial expressions. *IEEE transactions on pattern analysis and machine intelligence*, 29(6) :915–928, 2007.

[126] Timur R Almaev and Michel F Valstar. Local gabor binary patterns from three orthogonal planes for automatic facial expression recognition. In *2013 Humaine Association Conference on Affective Computing and Intelligent Interaction*, pages 356–361. IEEE, 2013.

[127] Bihan Jiang, Michel F Valstar, Brais Martinez, and Maja Pantic. A dynamic appearance descriptor approach to facial actions temporal modeling. *IEEE Trans. Cybernetics*, 44(2) :161–174, 2014.

[128] Richard J Prokop and Anthony P Reeves. A survey of moment-based techniques for unoccluded object representation and recognition. *CVGIP : Graphical Models and Image Processing*, 54(5) :438–460, 1992.

[129] Y Zhu, Liyanage C De Silva, and Chi Chung Ko. Using moment invariants and hmm in facial expression recognition. *Pattern Recognition Letters*, 23(1-3) :83–91, 2002.

[130] Seyed Mehdi Lajevardi and Zahir M Hussain. Higher order orthogonal moments for invariant facial expression recognition. *Digital Signal Processing*, 20(6) :1771–1779, 2010.

[131] Yi Ji and Khalid Idrissi. Automatic facial expression recognition based on spatiotemporal descriptors. *Pattern Recognition Letters*, 33(10) :1373–1380, 2012.

[132] Beat Fasel and Juergen Luettin. Recognition of asymmetric facial action unit activities and intensities. In *Proceedings 15th International Conference on Pattern Recognition. ICPR-2000*, volume 1, pages 1100–1103. IEEE, 2000.

[133] Xijian Fan and Tardi Tjahjadi. A dynamic framework based on local zernike moment and motion history image for facial expression recognition. *Pattern Recognition*, 64 :399–406, 2017.

[134] Junkai Chen, Zenghai Chen, Zheru Chi, and Hong Fu. Facial expression recognition in video with multiple feature fusion. *IEEE Transactions on Affective Computing*, 9(1) :38–50, 2018.

[135] Jean-Yves Bouguet et al. Pyramidal implementation of the affine lucas kanade feature tracker description of the algorithm. *Intel Corporation*, 5(1-10) :4, 2001.

[136] Irene Kotsia and Ioannis Pitas. Facial expression recognition in image sequences using geometric deformation features and support vector machines. *IEEE transactions on image processing*, 16(1) :172–187, 2006.

[137] Jenn-Jier James Lien. *Automatic recognition of facial expressions using hidden Markov models and estimation of expression intensity.* University of Pittsburgh, 1998.

[138] Takahiro Otsuka and Jun Ohya. Spotting segments displaying facial expression from image sequences using hmm. In *Proceedings Third IEEE International Conference on Automatic Face and Gesture Recognition*, pages 442–447. IEEE, 1998.

[139] W Fellenz, J Taylor, Nicolas Tsapatsoulis, and S Kollias. Comparing template-based, feature-based and supervised classification of facial expressions from static images. *Computational Intelligence and Applications*, 19(9) :9, 1999.

[140] Michel Valstar, Maja Pantic, and Ioannis Patras. Motion history for facial action detection in video. In *2004 IEEE International Conference on Systems, Man and Cybernetics (IEEE Cat. No. 04CH37583)*, volume 1, pages 635–640. IEEE, 2004.

[141] Ville Ojansivu and Janne Heikkilä. Blur insensitive texture classification using local phase quantization. In *International conference on image and signal processing*, pages 236–243. Springer, 2008.

[142] Zhengyou Zhang. Microsoft kinect sensor and its effect. *IEEE multimedia*, 19(2) :4–10, 2012.

[143] Latifa Greche, Nabil Hamaoui, and ES-Sbai Najia. Facial expression recognition on android. *Revue Méditerranéenne des Télécommunications*, 5(2), 2015.

[144] Fadi Dornaika, Elena Lazkano, and Basilio Sierra. Improving dynamic facial expression recognition with feature subset selection. *Pattern Recognition Letters*, 32(5) :740–748, 2011.

[145] Hervé Abdi and Lynne J Williams. Principal component analysis. *Wiley interdisciplinary reviews : computational statistics*, 2(4) :433–459, 2010.

[146] Peter N Belhumeur, João P Hespanha, and David J Kriegman. Eigenfaces vs. fisherfaces : Recognition using class specific linear projection. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, (7) :711–720, 1997.

[147] Muhammad Hameed Siddiqi, Rahman Ali, Muhammad Idris, Adil Mehmood Khan, Eun Soo Kim, Min Cheol Whang, and Sungyoung Lee. Human facial expression recognition using curvelet feature extraction and normalized mutual information feature selection. *Multimedia Tools and Applications*, 75(2) :935–959, 2016.

[148] Beat Fasel and Juergen Luettin. Automatic facial expression analysis : a survey. *Pattern recognition*, 36(1) :259–275, 2003.

[149] Jyoti Kumari, R Rajesh, and KM Pooja. Facial expression recognition : A survey. *Procedia computer science*, 58(1) :486–491, 2015.

[150] Jake VanderPlas. *Python data science handbook : Essential tools for working with data*. " O'Reilly Media, Inc.", 2016.

[151] Muhammad Hameed Siddiqi, Rahman Ali, Adil Mehmood Khan, Young-Tack Park, and Sungyoung Lee. Human facial expression recognition using stepwise linear discriminant analysis and hidden conditional random fields. *IEEE Transactions on Image Processing*, 24(4) :1386–1398, 2015.

[152] Min Tang and Feng Chen. Facial expression recognition and its application based on curvelet transform and pso-svm. *Optik-International Journal for Light and Electron Optics*, 124(22) :5401–5406, 2013.

[153] Bharath Hariharan, Jitendra Malik, and Deva Ramanan. Discriminative decorrelation for clustering and classification. In *European Conference on Computer Vision*, pages 459–472. Springer, 2012.

[154] Thomas Cover and Peter Hart. Nearest neighbor pattern classification. *IEEE transactions on information theory*, 13(1) :21–27, 1967.

[155] Leo Breiman. *Classification and regression trees*. Routledge, 2017.

[156] V Vapnik and Vlamimir Vapnik. Statistical learning theory wiley. *New York*, pages 156–160, 1998.

[157] Bharat Richhariya and Deepak Gupta. Facial expression recognition using iterative universum twin support vector machine. *Applied Soft Computing*, 76 :53–67, 2019.

[158] Rajiv Mehrotra, Kameswara Rao Namuduri, and Nagarajan Ranganathan. Gabor filter-based edge detection. *Pattern recognition*, 25(12) :1479–1494, 1992.

[159] AG Ramakrishnan, S Kumar Raja, and HV Raghu Ram. Neural network-based segmentation of textures using gabor features. In *Proceedings of the 12th IEEE Workshop on Neural Networks for Signal Processing*, pages 365–374. IEEE, 2002.

[160] Rangaraj M Rangayyan and Fábio J Ayres. Gabor filters and phase portraits for the detection of architectural distortion in mammograms. *Medical and biological engineering and computing*, 44(10) :883–894, 2006.

[161] Tristan Glatard, Johan Montagnat, and Isabelle E Magnin. Texture based medical image indexing and retrieval : application to cardiac imaging. In *Proceedings of the 6th ACM SIGMM international workshop on Multimedia information retrieval*, pages 135–142. ACM, 2004.

[162] S Marĉelja. Mathematical description of the responses of simple cortical cells. *JOSA*, 70(11) :1297–1300, 1980.

[163] Javier R Movellan. Tutorial on gabor filters. *Open Source Document*, 2002.

[164] SL Happy and Aurobinda Routray. Automatic facial expression recognition using features of salient facial patches. *IEEE transactions on Affective Computing*, 6(1) :1–12, 2014.

[165] Marian Bartlett, Gwen Littlewort, Tingfan Wu, and Javier Movellan. Computer expression recognition toolbox. In *2008 8th IEEE international conference on automatic face & gesture recognition*, pages 1–2. IEEE, 2008.

[166] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The elements of statistical learning : data mining, inference, and prediction*. Springer Science & Business Media, 2009.

[167] Klaus-Robert Müller, A Smola, Gunnar Rätsch, B Schölkopf, Jens Kohlmorgen, and Vladimir Vapnik. Using support vector machines for time series prediction. *Advances in kernel methods—support vector learning*, pages 243–254, 1999.

[168] Pedro Domingos and Michael Pazzani. On the optimality of the simple bayesian classifier under zero-one loss. *Machine learning*, 29(2-3) :103–130, 1997.

[169] Leo Breiman, Jerome Friedman, Charles J Stone, and Richard A Olshen. *Classification and regression trees*. CRC press, 1984.

[170] Ryan Rifkin and Aldebaro Klautau. In defense of one-vs-all classification. *Journal of machine learning research*, 5(Jan) :101–141, 2004.

[171] Gidudu Anthony, Hulley Gregg, and Marwala Tshilidzi. Image classification using svms : one-against-one vs one-against-all. *arXiv preprint arXiv :0711.2914*, 2007.

[172] Didier Nakache and Elisabeth Métais. Evaluation : nouvelle approche avec juges. In *INFORSID*, volume 5, pages 555–570, 2005.

[173] Xiaohua Wang, Chao Jin, Wei Liu, Min Hu, Liangfeng Xu, and Fuji Ren. Feature fusion of hog and wld for facial expression recognition. In *Proceedings of the 2013 IEEE/SICE International Symposium on System Integration*, pages 227–232. IEEE, 2013.

[174] Xijian Fan and Tardi Tjahjadi. A spatial-temporal framework based on histogram of gradients and optical flow for facial expression recognition in video sequences. *Pattern Recognition*, 48(11) :3407–3416, 2015.

[175] Marian Stewart Bartlett, Gwen Littlewort, Ian Fasel, and Javier R Movellan. Real time face detection and facial expression recognition : development and applications to human computer interaction. In *2003 Conference on computer vision and pattern recognition workshop*, volume 5, pages 53–53. IEEE, 2003.

[176] Wenfei Gu, Cheng Xiang, YV Venkatesh, Dong Huang, and Hai Lin. Facial expression recognition using radial encoding of local gabor features and classifier synthesis. *Pattern recognition*, 45(1) :80–91, 2012.

[177] Gwen Littlewort, Marian Stewart Bartlett, Ian Fasel, Joshua Susskind, and Javier Movellan. Dynamics of facial expression extracted automatically from video. In *2004 Conference on Computer Vision and Pattern Recognition Workshop*, pages 80–80. IEEE, 2004.

[178] Ligang Zhang, Dian Tjondronegoro, and Vinod Chandran. Random gabor based templates for facial expression recognition in images with facial occlusion. *Neurocomputing*, 145 :451–464, 2014.

[179] Latifa Greche, Mohamed Akil, Rostom Kachouri, and Najia Es-Sbai. A new pipeline for the recognition of universal expressions of multiple faces in a video sequence. *Journal of Real-Time Image Processing*, pages 1–14, 2019.

[180] Paul Viola, Michael J Jones, and Daniel Snow. Detecting pedestrians using patterns of motion and appearance. in null. *IEEE*, 19 :57, 2003.

[181] Shi-Hong Jeng, Hong Yuan Mark Liao, Chin Chuan Han, Ming Yang Chern, and Yao Tsorng Liu. Facial feature detection using geometrical face model : an efficient approach. *Pattern recognition*, 31(3) :273–282, 1998.

[182] Marina Sokolova and Guy Lapalme. A systematic analysis of performance measures for classification tasks. *Information processing & management*, 45(4) :427–437, 2009.

[183] Philipp Michel and Rana El Kaliouby. Real time facial expression recognition in video using support vector machines. In *Proceedings of the 5th international conference on Multimodal interfaces*, pages 258–264. ACM, 2003.

[184] Montse Pardàs and Antonio Bonafonte. Facial animation parameters extraction and expression recognition using hidden markov models. *Signal Processing : Image Communication*, 17(9) :675–688, 2002.

[185] Paul Ekman and Erika L Rosenberg. *What the face reveals : Basic and applied studies of spontaneous expression using the Facial Action Coding System (FACS)*. Oxford University Press, USA, 1997.