N° d'ordre :

République Algérienne Démocratique & Populaire

Ministère de l'Enseignement Supérieur & de la Recherche Scientifique

Université Djillali Liabes

Faculté des Sciences Exactes- Sidi Bel Abbès

THESE

DE DOCTORAT EN SCIENCES

Présentée par : **Yacine GAFOUR**

Spécialité : **Informatique**

Option : **Intelligence Artificielle**

Intitulé de la thèse

Apprentissage automatique pour une classification des images basée sur les descripteurs

Soutenue le 16/12/2020

Devant le jury composé de :

Président	Réda ADJOUDJ	Professeur	UDL - SBA
Directeur de thèse	Djamel BERRABAH	MCA	UDL - SBA
Co-directeur de thèse	Abdelkader GAFOUR	Professeur	UDL - SBA
Examinateur	Sidi Mohamed BENSLIMANE	Professeur	ESI - SBA
Examinateur	Djamel AMAR BENSABER	MCA	ESI - SBA
Examinateur	Nabil KESKES	MCA	ESI - SBA

Année universitaire : 2020 -2021

Je dédie ce travail

A mes très chers parents qui m’ont tout donné sans rien attendre en retour et pour m’avoir soutenu et encouragé dans ce tournant de ma vie

A mes chers frères et sœurs source de joie et de bonheur

A toute ma famille, source d’espoir et de motivation

A tous mes amis

Merci à eux de me rappeler sans cesse ce qu’est la vraie richesse, qui m’avez toujours soutenu et encouragé durant ces années d’études.

Remerciements

Le travail présenté dans cette thèse a été effectué au sein du laboratoire EEDIS (Evolutionary engineering and distributed information systems) dirigé par le professeur Sofiane BOUKLI HACENE que je tiens à remercier, pour m'avoir accueilli dans ce laboratoire.

Je tiens à exprimer mon profond sentiment de gratitude et mes remerciements les plus sincères à mon directeur de thèse Mr Djamel BERRABAH pour son vif intérêt, sa bienveillance, son aide infaillible, ses conseils inspirants, ses encouragements constants et son inspiration avec mon travail à toutes les étapes, pour mener à bien cette thèse. C'est un grand plaisir pour moi d'avoir la chance de travailler avec lui parce qu'il m'a guidé avec ses précieuses suggestions, a éclairé la voie dans mes moments les plus sombres et m'a beaucoup encouragé dans les domaines académiques et non académiques.

Je tiens également à remercier le co-directeur de thèse Mr Abdelkader GAFOUR d'avoir tout en moi fais confiance d'accepter de m'encadrer en première année de doctorat.

Je tiens également à le remercie pour les qualités scientifiques et pédagogiques de son enseignement pendant ma préparation mon diplôme d'ingéniorat.

Je remercie Mr Réda ADJOUDJ, professeur à l'Université Djillali Liabès de Sidi Bel Abbès qui a bien voulu me faire l'honneur d'examiner ce travail de recherche et de présider le jury.

Je suis extrêmement reconnaissant envers les membres de jury,

Mr Sidi Mohamed BENSLIMANE, professeur à l'Ecole Supérieur en Informatique (ESI)- Sidi Bel Abbes

Mr Djamel AMAR BENSABER, maître de conférences "A" à l'Ecole Supérieur en Informatique (ESI)- Sidi Bel Abbes ;

Mr Nabil KESKES, maître de conférences "A" à l'Ecole Supérieur en Informatique (ESI)- Sidi Bel Abbes ; d'avoir accepté d'examiner mes travaux de thèse, et pour le temps qu'ils ont consacré à ce fait.

Je remercie également l’ensemble des enseignants et les membres du personnel non enseignant du département d'informatique de la faculté des sciences exactes de l’Université Djillali Liabès de Sidi Bel Abbès.

Je remercie aussi l’ensemble des enseignants et les membres du personnel non enseignant du département d'informatique de la faculté des mathématiques et d’informatique de l’Université Ibn Khaldoun-Tiaret.

Enfin, je tiens à remercier très sincèrement tous les membres du laboratoire EEDIS, et tous ceux qui m’ont aidé de près ou de loin à la réalisation de ce travail.

Résumé

Le développement rapide des appareils numériques (téléphones portables, cameras etc.) a mené à une augmentation explosive des données multimédia (images et vidéos) de haute qualité à gérer. L'énorme quantité de ces données doit être interprétées et récupérées par les grandes entreprises. En effet, elles ont besoin de méthodes efficaces pour exploiter le contenu de ces données et le transformer en connaissances précieuses afin d'avoir une compréhension visuelle rapide des images et des vidéos.

Dans cette thèse, nous définissons plusieurs buts qui sont intéressants pour comprendre le contenu visuel des images afin de réaliser les tâches de la classification d'images et la reconnaissance d'objets. Par conséquent, nous proposons des modèles et des approches dédiées à ces taches en utilisant l'apprentissage automatique et en se basant sur des descripteurs représentatifs du contenu de l'image. Ces descripteurs sont obtenus par le processus d'extraction de caractéristiques à partir de l'image. Dans ce contexte, nous présentons les deux approches suivantes.

Dans la première approche, nous proposons un nouveau modèle pour améliorer les performances du descripteur A-KAZE pour la classification des images. Nous établissons d'abord la connexion entre le descripteur A-KAZE et le modèle *Bag of features* (BoF). Ensuite, nous adoptons le *Spatial Pyramid Matching* (SPM) pour introduire des informations spatiales durant l'exploitation du descripteur A-KAZE afin de renforcer sa robustesse. Nous présentons dans la seconde approche un nouveau modèle pour la reconnaissance faciale. Cette approche est basée sur un nouvel ensemble de variantes du descripteur LBP que nous avons proposé et que nous avons appelé *Honeycomb-Local Binary Pattern* (Ho-LBP). En effet, la présentation des images en utilisant un ensemble de variantes du descripteur Ho-LBP aide le classificateur à mieux apprendre. De plus, ces variantes sont combinées pour améliorer les performances du processus de la reconnaissance faciale.

Mots clés : Apprentissage automatique, descripteurs, classification des images, reconnaissance faciale.

Abstract

The rapid development of digital devices (cell phones, cameras, etc.) has led to an explosive increase in high quality multimedia data (images and videos) to be managed. The enormous amount of this data must be interpreted and retrieved by large companies. Indeed, they need efficient methods to exploit the content of this data and transform it into valuable knowledge in order to have a rapid visual understanding of images and videos.

In this thesis, we define several goals that are interesting for understanding the visual content of images in order to perform the tasks of image classification and object recognition. Therefore, we propose models and approaches dedicated to these tasks using machine learning and based on descriptors representative of the content of the image. These descriptors are obtained by the process of extracting features from the image. In this context, we present the following two approaches.

In the first approach, we propose a new model to improve the performance of the A-KAZE descriptor for image classification. We first establish the connection between the A-KAZE descriptor and the Bag of features (BoF) model. Then, we adopt the Spatial Pyramid Matching (SPM) to introduce spatial information during the exploitation of the A-KAZE descriptor in order to reinforce its robustness. We present in the second approach a new model for facial recognition. This approach is based on a new set of variants of the LBP descriptor that we have proposed and named Honeycomb-Local Binary Pattern (Ho-LBP). Indeed, presenting the images using a set of variants of the Ho-LBP descriptor helps the classifier to learn better. In addition, these variants are combined to improve the performance of the facial recognition process.

Keywords: Machine learning, descriptors, image classification, facial recognition.

ملخص

أدى التطور السريع للأجهزة الرقمية (الهواتف المحمولة والكاميرات وما إلى ذلك) إلى زيادة هائلة في بيانات الوسائط المتعددة عالية الجودة (الصور ومقاطع الفيديو) التي يتعين إدارتها. لابد من تفسير الكم الهائل من هذه البيانات واسترجاعها من قبل الشركات الكبيرة. في الواقع، يحتاجون إلى طرق فعالة لاستغلال محتوى هذه البيانات وتحويلها إلى معرفة قيمة من أجل الحصول على فهم مرئي سريع للصور ومقاطع الفيديو.

نحدد في هذه الرسالة عدة أهداف مثيرة للاهتمام وهذا لفهم المحتوى المرئي للصور لغرض أداء مهام تصنيف الصور والتعرف على محتوياتها. في هذا السياق نقترح نماذجا وأساليبا مخصصة لهذه المهام باستخدام التعلم الآلي وبناءً على الواصفات التي تمثل محتوى الصورة. يتم الحصول على هذه الواصفات من خلال عملية استخراج الميزات من الصورة. في هذا السياق ، نقدم المنهجين التاليين.

في النهج الأول نقترح نموذجًا جديدًا لتحسين أداء واصف $A\text{-}KAZE$ لتصنيف الصور. أولاً أنشأنا اتصالا بين واصف $A\text{-}KAZE$ ونموذج حقيبة الميزات (BoF). بعد ذلك نعتمد مطابقة الهرم المكاني (SPM) من خلال استغلال واصف $A\text{-}KAZE$ لتعزيز متانته من خلال تقديم المعلومات المكانية. نقدم في النهج الثاني نموذجًا جديدًا للتعرف على الوجه. استعملنا في هذا النهج مجموعة جديدة من المتغيرات لوصف LBP التي اقترحناها والتي أطلقنا عليها اسم النمط الثنائي المحلي على شكل خلايا النحل ($Ho\text{-}LBP$). إن استغلال الصور باستخدام مجموعة من المتغيرات للموصف $Ho\text{-}LBP$ يساعد المصنف على التعلم بشكل أفضل. بالإضافة إلى ذلك يتم الجمع بين هذه المتغيرات لتحسين أداء التعرف على الوجه.

الكلمات المفتاحية: التعلم الآلي ، الواصفات ، تصنيف الصور ، التعرف على الوجه.

Table des matières

IV. Approches de l'intelligence artificielle pour la classification des images 52

Liste des figures

Liste des tableaux

I. Introduction générale

1 Modélisation et classification des images

L'apprentissage automatique, aussi appelé *Machine Learning* (ML) en anglais, est un sous domaine de l'intelligence artificielle qui applique des algorithmes pour synthétiser les relations fondamentales entre les données et les informations (Kaifeng et al., 2020).

L'application des algorithmes par ML pour le but de transformer des données empiriques en modèle (données de l'apprentissage) utilisable. Ce modèle est basé sur les propriétés établies ou bien tirées des données de formation afin de permettre à la ML d'effectuer une analyse prédictive. ML aide au développement des programmes qui améliorent leurs performances pour une tâche spécifique en se basant sur des correspondances sous-jacentes entre les données d'entrées et les résultats attendus pendant l'apprentissage. La question à savoir que ML doit être répond est comment construire un programme informatique en utilisant des données historiques, pour résoudre un problème donné et améliorer automatiquement l'efficacité du programme en utilisant l'apprentissage.

Ces dernières années, de nombreuses applications d'apprentissage automatique ont été développées afin de classer de nouvelles structures (*patterns* en anglais), allant des programmes d'exploration de données qui apprennent à détecter les transactions frauduleuses, des études neurobiologiques, aux véhicules autonomes qui apprennent à conduire sur les routes publiques aux systèmes de filtrage des informations etc.

L'apprentissage supervisé est un modèle d'apprentissage conçu pour faire des prédictions. Il utilise un ensemble de données d'apprentissage pour développer un modèle de prédiction en fonction des données d'entrée et des valeurs de sortie. La technique utilisée dans cet apprentissage est basée sur l'extraction des relations sous-jacentes entre des attributs indépendants qui représentent des données d'entrée et un attribut dépendant désigné la valeur de

sortie (l'étiquette). Un algorithme d'apprentissage supervisé prend un ensemble connu de données d'entrée et ses réponses connues aux données (sortie) pour apprendre le modèle de classification / régression. Il entraîne ensuite un modèle pour générer une prédiction de la réponse à de nouvelles données ou à l'ensemble de données de test.

La classification des images est un problème d'apprentissage supervisé qui peut définir un ensemble de classes cibles (catégories des images) et former un modèle pour les reconnaître à l'aide des exemples des images étiquetées. Cette classification fait référence à un processus en vision par ordinateur qui peut classer une image en fonction de son contenu visuel. Elle est en croissance et devient une tendance chez les développeurs de technologies actuelles, en particulier avec l'augmentation des données dans différentes parties de l'industrie telles que l'agriculture, le tourisme, les soins de santé, les produits pharmaceutiques, l'automobile, le commerce électronique, et les jeux. Une classification robuste des images reste un défi dans les applications en vision par ordinateur. La classification des images est normalement effectuée en deux composantes principales : l'extraction des caractéristiques et la classification.

Les caractéristiques sont des propriétés distinctives des objets dans les images, ces caractéristiques aident les modèles à la prise de décisions telles que la détection, la reconnaissance et la classification. Extraction des caractéristiques d'un objet dans une image permet de décrire et de représenter efficacement un l'objet dans un espace de dimension réduit. Les caractéristiques extraites peuvent faire référence à des emplacements spécifiques dans les images, tels que les coins de bâtiments, les sommets des montagnes, ou d'autres caractéristiques d'objets selon l'application considérée. Ces types de caractéristiques locales sont souvent appelées des points clés (ou points d'intérêt) et sont souvent décrites par des pixels entourant l'emplacement du point clé.

Le processus d'extraction des caractéristiques influence la tâche d'apprentissage automatique et une sélection rigoureuse des caractéristiques décide de la précision du système de classification. Ainsi, la recherche de ces caractéristiques est appelée détection de caractéristiques. Une fois qu'un point clé est détecté, la région entourant le point clé détecté est utilisée pour décrire les caractéristiques locales de ce point clé en construisant un descripteur des point clés détectés, également appelé descripteur de caractéristiques. Ce descripteur est généralement formé en extrayant des caractéristiques décrivant les caractéristiques locales du voisinage du point clé détecté. Les descripteurs de caractéristiques doivent être invariants à la rotation, à l'éclairage, à la

translation et / ou à la mise à l'échelle. Bien que, de nombreux classificateurs différents aient été proposés pour effectuer la classification en utilisant ces descripteurs, tels que la machine à vecteurs de support, aussi appelé *Support Vector Machines* (SVM) en anglais, le k-plus proche voisin ou *K-Naerest Neighbor* (kNN) en anglais, les forets aléatoires ou bien *Random Forest* (RF) et *Naive Bayes* (NB).

L'objectif principal de cette thèse est de construire des systèmes de classification automatique en utilisant l'apprentissage supervisé pour analyser les modèles et faire des prédictions précises basées sur les observations précédentes pour les problèmes de reconnaissance et de classification des images. La précision des systèmes dépend des caractéristiques fiables qui doivent être indépendantes et non corrélées. Les descripteurs génériques les plus récemment proposés, tels que *Local Binary Pattern* (LBP) et *Scale Invariant Feature Transform* (SIFT), *Accelerate-KAZE* (A-KAZE), *Speeded Up Robust Features* (SURF) ont également été incorporés pour représenter les caractéristiques des objets dans les images. Ces descripteurs ont démontré des performances de classification plus élevées.

2 Problématique et contributions de la thèse

La représentation et la description des images en utilisant des caractéristiques locales sont très importante étape pour résoudre les problèmes d'analyse et la classification des images. Ces caractéristiques doivent être robustes aux changements et aux distorsions d'une image parce qu'elles sont des facteurs décisifs de divers problèmes de classification des images.

La forte variabilité intra-classe des images appartenant à la même classe et la faible variabilité inter-classe des images appartenant aux différentes classes rendent ces problèmes difficiles. De nombreuses caractéristiques et algorithmes de vision par ordinateur ont été proposés pour faire face à ces problèmes. Le but des algorithmes d'extraction de caractéristiques est d'identifier les caractéristiques qui représentent le mieux l'image et contiennent moins de paramètres. Avec les caractéristiques spécifiées, l'image peut être exprimée de manière significative en utilisant moins de paramètres. Une classification plus rapide et réussie peut être effectuée avec moins de cout de calcul en éliminant les paramètres non important. Les caractéristiques de bas niveau sont des caractéristiques plus simples dans l'image et demandent moindre de coût de calcul. Le choix des caractéristiques à utiliser pour la classification des images dépond du problème à résoudre.

Pour cette raison, il existe de nombreux algorithmes d'extraction de caractéristiques avec différentes approches dans la littérature. En tant que problème classique de reconnaissance de formes, la classification des objets et la reconnaissance faciale se composent principalement de deux sous-problèmes critiques : l'extraction de caractéristiques et le choix de classificateurs, qui ont tous les deux font l'objet d'une étude approfondie. En général, la description des caractéristiques des objets ou bien des visages joue un rôle relativement plus important dans la classification. En effet, si de mauvaises caractéristiques sont utilisées, et même le meilleur classificateur ne parviendra pas à obtenir de bons résultats de classification.

Il reste difficile de trouver des descripteurs de caractéristiques généralisés capables de capturer des caractéristiques discriminantes dans l'image pour la classification des objets. Dans cette situation, l'information spatiale joue un rôle clé dans l'analyse et la compréhension la description des objets dans une image. Afin d'extraire des caractéristiques de bas niveau d'une image avec succès.

Dans cette thèse, nous utilisons un modèle de présentation de caractéristiques pyramidale dans laquelle nous construisons un *Bag of features* (BoF) basé sur une pyramide hiérarchique afin de classifier des images. A-KAZE est considéré comme le premier descripteur de détecter des entités en construisant un espace d'échelle en utilisant une diffusion non linéaire. A-KAZE utilise un descripteur binaire appelé *Modified-Local Difference Binary* (MLDB), qui est très efficace et invariant aux changements de rotation et d'échelle. La représentation binaire est générée par comparaison entre les grilles de trois canaux (une luminance et deux dérivées de premier ordre) afin de décrire les points clés détectés dans l'image. Cette représentation ne permet pas de considérer l'information spatiale entre les objets dans l'image, ce qui permet de réduire des performances de la classification des images. Dans cette thèse, nous abordons une nouvelle approche pour améliorer les performances du descripteur A-KAZE pour la classification d'images. Nous établissons d'abord la connexion entre le descripteur A-KAZE et le modèle BoF. Ensuite, nous adoptons *Spatial Pyramid Matching* (SPM) en exploitant le descripteur A-KAZE pour renforcer sa robustesse en introduisant des informations spatiales. Les résultats des expériences sur plusieurs bases de données montrent que le descripteur A-KAZE basé sur SPM donne des résultats très satisfaisants par rapport à d'autres approches existantes dans l'état de l'art.

Des efforts considérables ont été déployés sur des tâches de reconnaissance faciale qui nous intéressent dans cette thèse, où des caractéristiques fiables doivent être extraites pour l'identification biométrique. En effet, la plupart des algorithmes de reconnaissance faciale qui existent aujourd'hui ont réalisé d'excellentes performances, mais ils restent loin de la reconnaissance faciale faite par l'être humain. Dans les cas non contrôlés, les grandes variations des images faciales telles que l'expression, l'éclairage, le vieillissement, la posture et l'occlusion rendent difficile la reconnaissance de la personne, car les variations entre les classes sont relativement faibles ; que les variations intra-classe peuvent être très importantes où, dans chaque classe, nous trouvons un ensemble des images de visages de la même personne. Changer la luminance et la pose d'un même visage peut changer radicalement l'apparence de l'identité de la personne, donc la reconnaissance faciale est une tâche difficile avec ces changements, cela peut affecter la précision de la reconnaissance.

Dans cette thèse, nous proposerons une nouvelle approche où nous nous sommes adaptés à la structure en nid d'abeille aux pixels de l'image en niveaux de gris dans les images qui représentent le visage. Nous utilisons dans cette approche un nouveau descripteur basé sur des modèles binaires locaux. Le nouveau descripteur est une extension du descripteur de texture LBP. Dans ce travail, nous montrons comment améliorer des performances de la reconnaissance faciale en choisissant une solution inspirée des *Honeycombs*, tout en exploitant le descripteur LBP qui sera appliqué à la représentation hexagonale des cellules d'une ruche. Cette exploitation permet de donner naissance à un nouveau descripteur appelé *Honeycombs-Local Binary Pattern* (Ho-LPB). Les résultats expérimentaux sur nombreuses bases de données faciales montrent que notre approche est robuste pour résoudre le problème de la variation de pose et du changement de luminance pour la reconnaissance faciale.

3 Organisation de la thèse

La thèse se concentre sur la description et la représentation efficaces des images en utilisant les descripteurs de caractéristiques pour fournir une meilleure reconnaissance et précision de classification. La thèse est organisée comme suit :

Chapitre 1 : Introduction générale.

La thèse se débute par une introduction générale sur la classification des images et illustre leurs composantes principales. Ce chapitre a également analysé le contexte du problème, défini

l'énoncé du problème et identifié les objectifs de la recherche. La méthodologie de recherche, les approches proposées et la contribution des travaux de recherche sont également expliqués dans ce chapitre.

Chapitre 2 : Introduction à la vision par ordinateur.

Ce chapitre fournit des connaissances introductives sur la vision par ordinateur que permet d'extraire, d'analyser et de comprendre automatiquement les informations utiles à partir d'une image. Ce chapitre illustre le fonctionnement du système visuel humain afin de pouvoir développer des algorithmes robustes et efficaces pour accomplir la compréhension visuelle automatique.

Chapitre 3 : Détecteurs des caractéristiques locales, descripteurs et représentation des images.

Dans ce chapitre, nous donnons un aperçu des méthodes et des algorithmes proposés pour représenter les images par les chercheurs. Ensuite, présente les propriétés des caractéristiques et donne un aperçu des différentes méthodes existantes de détection et de description. De plus, nous expliquons les notations de base et les concepts mathématiques pour détecter et décrire les caractéristiques de l'image. Ainsi que, nous explorons également en détail quelle est la relation entre les détecteurs et les descripteurs. En fin, nous fournissons au lecteur une compréhension fondamentale de quelques modèles de représentation des images en utilisant ces caractéristiques.

Chapitre 4 : Approches de l'intelligence artificielle pour la classification des images.

Ce chapitre explore l'utilisation de l'apprentissage automatique pour classifier des images et ses méthodologies. Nous aborderons également divers concepts importants couvrant les principes fondamentaux du l'apprentissage automatique tels que les types d'apprentissage, les différences principales entre l'apprentissage automatique et l'apprentissage profond pour la classification des images ainsi que les mesures d'évaluation, etc.

Chapitre 5 : Nouvelle approche pour améliorer le processus de classification des objets.

Ce chapitre illustre une nouvelle approche pour améliorer la classification des images qui contiennent des objets. Ce chapitre commence par la présentation de certains types de caractéristique qui sont généralement exploités par un certains nombres d'applications. Puis, de

trouver le meilleur de ces caractéristiques à utiliser dans notre approche en les ajoutant des informations spatiales. Dans cette approche, les résultats de la classification sont comparés à l'aide de différents algorithmes d'extraction de caractéristiques qui peuvent extraire diverses caractéristiques des images. Ensuite, nous discutons les résultats obtenus, À la fin de ce chapitre, nous impliquons les remarques finales de cette approche.

Chapitre 6 : Extraction automatique des caractéristiques pour la reconnaissance faciale.

Ce chapitre est couvert au domaine de la reconnaissance faciale. Dans ce chapitre, nous montrons comment la littérature a introduit des techniques robustes dans la reconnaissance faciale. Ensuite, nous détaillons l'approche proposée pour la classification faciale. Une discutions des expériences comparatives faites par nous et concluons avec des commentaires et des recommandations.

Chapitre 7 : Conclusion générale.

Présente le résume de les contributions de recherche et les suggestions de recherche possibles pour les travaux futurs.

II. Introduction à la vision par ordinateur

1 Introduction

La technologie a toujours été le moteur de grandes innovations. En tant que créateurs, nous vivons actuellement dans l'une des ères les plus passionnantes où la technologie fait d'énormes progrès.

La vision par ordinateur (*Computer Vision* "CV" en anglais) est une discipline qui a débuté dans les années 1980 et qui a été utilisée dans le monde entier pour son potentiel et ses nombreuses applications. La vision par ordinateur comprend des méthodes et des techniques grâce auxquelles des systèmes de vision artificielle peuvent être construits et utilisés de manière raisonnable dans des applications pratiques. Ces méthodes de l'informatique comprennent les logiciels, le matériel et les techniques d'imagerie nécessaires (Davies, 2005). Ces dernières années, l'intelligence artificielle a permis de produire des résultats impressionnants et impeccables en remplaçant l'humain dans des tâches telles que la reconnaissance d'objets et les tâches de vision par ordinateur (Cohen, Feigenbaum, 1982).

Le terme "intelligence" fait référence à la capacité d'acquérir des connaissances et d'appliquer des compétences pour résoudre un problème donné. En outre, l'intelligence concerne également l'utilisation de la capacité mentale pour apprendre, résoudre et raisonner dans diverses situations (Vinciarelli, et al., 2015 ; Lake, Ullman, Tenenbaum, & Gershman, 2017).

L’étude de l’intelligence s’est bien développée au cours des dix dernières années. L'intelligence est intégrée à diverses fonctions cognitives telles que : langage, réflexion, perception, raisonnement, planification. L'intelligence humaine concerne la résolution de

problèmes, la réflexion et l'apprentissage. De plus, les humains ont des comportements complexes et simples qu'ils peuvent apprendre facilement tout au long de leur vie (Gottfredson, 1998).

2 Intelligence artificielle

2.1 Qu'est-ce que l'Intelligence Artificielle ?

Selon John McCarthy qui est le père de L'Intelligence Artificielle (IA), *"The science and engineering of making intelligent machines, especially intelligent computer programs"* (McCarthy, 2007). L'IA est un moyen de faire penser intelligemment un ordinateur, par exemple : un robot contrôlé par un ordinateur ou un logiciel d'une manière similaire que pensent les humains intelligents. L'IA consiste à étudier comment le cerveau humain apprend, pense et décide pour résoudre un problème. Puis, les résultats de cette étude seront utilisés comme base du développement de systèmes intelligents pour résoudre d'autre problème. Les racines de l'IA remontent aux philosophes classiques de la Grèce et à leurs efforts pour modéliser la pensée humaine en tant que système de symboles.

En 1950, Alan Turing, écrivit un article suggérant comment tester une machine «pensante» (Turing, 1950). Il pensait que si une machine pouvait tenir une conversation au moyen d'un téléimprimeur, imitant un être humain sans différences notables, la machine pourrait être décrite comme pensante. Son article a été suivi en 1952 par le modèle de Hodgkin, & Huxley, (1952) du cerveau en tant que neurones formant un réseau électrique dont des neurones individuels se déclenchant en pulsations. Ces événements, lors d'une conférence parrainée par le Dartmouth College en 1956, ont contribué à faire émerger le concept d'Intelligence Artificielle.

Le développement de l'IA a connu de nombreuses fluctuations. Le concept d'intelligence artificielle a été lancé en 1956. Dans les années 1970, le financement de la recherche sur l'IA a été interrompu après que plusieurs rapports aient critiqué le manque de progrès. Les chercheurs en IA ont dû faire face à deux limitations basiques, pas assez de mémoire et des vitesses de traitement trop faible par rapport aux normes actuelles. La recherche sur l'IA a repris dans les années 80, les États-Unis et la Grande-Bretagne fournissant des fonds pour concurrencer le nouveau projet informatique du Japon et leur objectif de devenir le leader mondial de la technologie informatique.

2.2 Qu'est-ce qui contribue à l'IA ?

L'IA est une technologie basée sur des disciplines telles que l'informatique, la linguistique, la psychologie, la biologie, l'ingénierie et les mathématiques. L'IA est principalement basée sur le développement des technologies informatiques associées à l'intelligence humaine, telles que l'apprentissage, le raisonnement et la résolution de problèmes. Parmi les objectifs de l'IA on a :

- La création des systèmes experts : Un système expert est un programme informatique qui simule l'action et le comportement d'un être humain ou d'une organisation possédant des connaissances et des expériences étendues dans un domaine particulier (Kumar, & Jain, 2013).
- L'implémentation de l'intelligence humaine dans les machines : L'IA est un moyen efficace permettant aux ordinateurs et aux logiciels à penser en utilisant des systèmes experts pour adopter un comportement intelligent et conseiller les utilisateurs en appliquant de l'apprentissage et de la réflexion (Shabbir, & Anwer, 2015).

2.3 Applications de l'IA

L'intelligence artificielle a été dominante dans divers domaines tels que :

- L'intelligence artificielle de jeu : joue un rôle crucial dans les jeux stratégiques où la machine peut penser à un grand nombre de positions possibles basées sur une connaissance heuristique, tels que le poker, les échecs etc.,
- Les systèmes experts : certaines applications intègrent des machines et des logiciels permettant de raisonner et de proposer des conseils aux utilisateurs.
- Les robots intelligents : les robots sont des machines ou des appareils fonctionnant automatiquement ou par télécommande. Ils ont des capteurs pour détecter les données physiques du monde réel telles que la température, la chaleur, la lumière, les mouvements, le son, les chocs, la pression etc. Ils sont capables d'apprendre de leurs erreurs et de s'adapter à un nouvel environnement afin d'accomplir les tâches confiées à un humain (Kurfess, 2005).
- La reconnaissance vocale : certains systèmes intelligents sont capables d'entendre et de comprendre la voix humaine captée. Ces systèmes permettent d'analyser le langage en termes de phrases et de leurs significations pendant qu'un humain le parle.

- La reconnaissance de l'écriture manuscrite est l'un des domaines de recherche les plus fascinants et les plus difficiles dans le domaine du traitement de l'image et de la reconnaissance des formes au cours des dernières années. Les applications dotées d'un système de reconnaissance de l'écriture manuscrite contribuent énormément à l'avancement du processus d'automatisation et peuvent améliorer l'interface entre l'homme et la machine.
- Les systèmes de vision : ces systèmes interprètent et comprennent les entrées visuelles sur l'ordinateur. Par exemple,
    - Les médecins utilisent un système expert clinique pour diagnostiquer le patient.
    - Un avion espion prend des photographies qui sont utilisées pour comprendre des informations spatiales.
    - La police utilise un logiciel informatique capable d'arrêter une personne criminel grâce aux systèmes de la reconnaissance faciale.

3 Du système visuel humain vers la vision par ordinateur

3.1 Qu'est-ce que la vision par ordinateur ?

La vision par ordinateur est un domaine d'étude qui cherche à développer des techniques permettant aux ordinateurs de "voir" et de comprendre le contenu des images numériques telles que les images et les vidéos.

❖ **Pourquoi avons-nous besoin de vision par ordinateur ?**

Les smart phones sont dotés de caméras qui facilitent la prise de photos et de vidéos, ce qui entraîne une croissance incroyable pour les réseaux sociaux modernes comme Facebook, YouTube, Twitter et Instagram. YouTube est peut-être le deuxième moteur de recherche en importance, des centaines d'heures de vidéos sont téléchargées chaque minute et des milliards de vidéos sont visionnées chaque jour.

Internet est constitué de texte, d'images et de vidéos. Il est relativement facile d'indexer et de rechercher du texte, mais pour indexer et rechercher des images, les algorithmes doivent savoir ce que les images contiennent. Pendant plus longtemps, le contenu des images et des vidéos est resté opaque et se décrit mieux à l'aide des méta-descriptions fournies par le programme de téléchargement. Pour tirer la meilleure partie des données de l'image, nous avons besoin des ordinateurs capables de visualiser, d'analyser et de comprendre le contenu.

❖ **La vision par ordinateur est différente du traitement d'image ?**

Le traitement d'image est le processus de création d'une nouvelle image à partir d'une image existante. Le contenu est généralement simplifié ou amélioré d'une manière ou d'une autre, mais il ne s'agit pas de comprendre le contenu de l'image. Le système de vérification sur ordinateur peut nécessiter certaines opérations associées au traitement des images en tant que configuration de l'image :

Les exemples de traitement des images incluent :

- Améliorer les propriétés optiques de l'image, telles que la luminosité ou la couleur.
- Couper les bordures d'une image, par exemple en centrant un objet dans une image.
- Supprimer le bruit d'une image.

Il existe de nombreuses applications de vision par ordinateur, notamment :

- Classification d'objets.
- Identification d'objets.
- Vérification d'objets.
- Détection d'objets.
- Segmentation d'objets.
- Reconnaissance d'objets.

3.2 Complexité du système de la vision par ordinateur

Computer vision est la transformation des données d'une caméra fixe ou vidéo en une nouvelle représentation afin de prendre une décision. La décision peut être la détection d'une personne dans cette scène, la reconnaissance faciale dans une image ou bien le comptage des cellules tumorales sur une diapositive. Une nouvelle représentation pourrait consister à supprimer le mouvement de l'arrière plan d'une séquence des images afin de segmenter un objet ou bien à transformer une image couleur en une image en niveaux de gris pour effectuer un traitement particulier sur cette image. Toutes ces transformations sont effectuées pour atteindre un objectif spécifique. La figure 1 montre une image d'un visage d'une femme sur un fond qui semble gris. En effet, ce que l'ordinateur voit n'est qu'une matrice de valeurs encadrées par le rectangle en couleur jaune.

Les actions ou décisions que la vision par ordinateur tente de prendre en fonction des données de la caméra sont exploitées dans le contexte d'une tâche spécifique ou d'un objectif ciblé. Certaines données dans cette matrice sont bruitées donc l'ordinateur ne reçoit que peu d'informations pertinentes de cette matrice.

La tâche primordiale dans la vision par ordinateur consiste donc à transformer cette matrice bruitée en perception des objets comme dans la figure 1 qui montre une femme souriante.

Figure 1. Résultat de la reconnaissance de l'expression faciale d'une femme (Nigam, Singh, & Misra, 2018).

Le défi de la vision par ordinateur est lié à la complexité du monde visuel. En effet, il est impossible de traiter une vue en deux dimensions (2D) d'un monde 3D. La figure 2 explique pourquoi la vision par ordinateur reste encore difficile par ce qu'un objet dans un système en vision par ordinateur peut être perçu dans diverses conditions d'éclairage, sous de multiples angles, partielle caché par d'autres objets. Les possibilités de produire une image de 2D sont indéfinies d'un monde 3D, tout comme un artiste dessinateur qui pourrait reproduire la réalité sous une infinité d'angles avec différents paramètres.

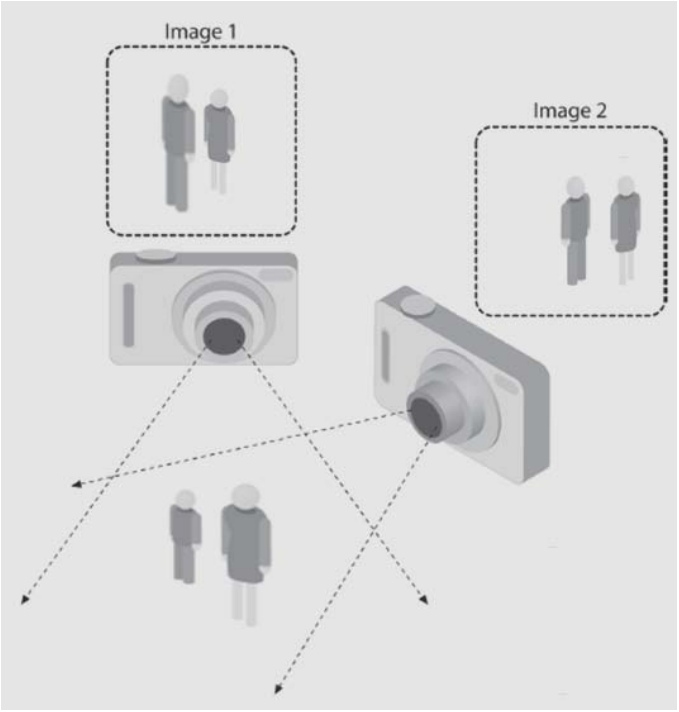


Figure 2. La complexité du système de la vision par ordinateur

Un véritable système de vision par ordinateur doit être en mesure d'en extraire des informations afin de percevoir le contenu dans n'importe quelles situations. Cependant, Les données reçues par l'ordinateur sont par fois bruitées ce qui provoque la dégradation de la qualité d'image. Cette dégradation est considérée comme un autre défi. Elle est liée à la complexité du monde visuel. Une telle dégradation dépend d'un éclairage insuffisant (conditions météorologiques, éclairage, réflexions, mouvements, bruit électrique dans le capteur ou d'autres composants électroniques, configuration mécanique et artefacts de compression après la capture de l'image) lors de la prise de l'image. Face à ces défis, la vision par ordinateur représente un véritable challenge scientifique. De fait, comment pouvons-nous progresser ?

3.3 Système visuel humain

La perception visuelle est un mécanisme complexe qui met en jeu plusieurs structures : l'œil, la rétine et le cerveau (voir la figure 3.). Le système visuel humain est le plus performant pour reconnaître des objets, des visages ou des paysages. Jusqu'à aujourd'hui, aucun autre système n'est aussi efficace ni aussi rapide pour identifier des objets sous des positions différentes ou des éclairages différents. Il est important de comprendre le fonctionnement du système visuel humain afin de pouvoir développer des algorithmes robustes et efficaces. Les algorithmes en

vision par ordinateur ont pour objectif d'analyser et de comprendre le contenu des images et des vidéos.

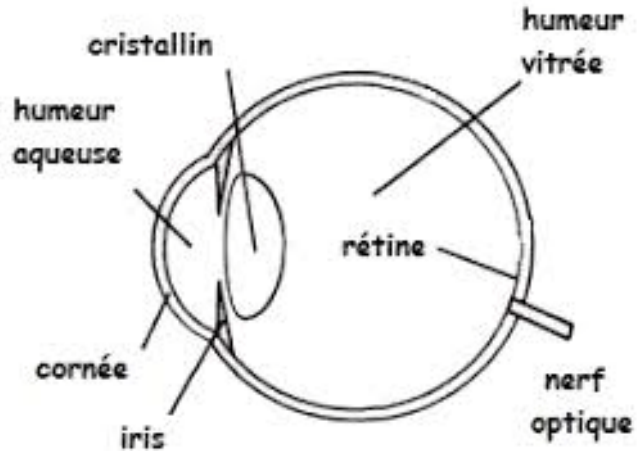


Figure 3. Œil humain

Les yeux humains fonctionnent comme un appareil photo. La lumière rebondit des objets vers l'œil à travers la cornée. La cornée aide à diriger la lumière vers la pupille et l'iris pour contrôler la quantité de lumière pénétrant dans l'œil. L'humeur vitrée est une masse gélatineuse sans couleur et transparente qui remplit l'espace situé entre le cristallin et la membrane de la rétine située sur la face postérieure de l'œil. La pupille est le petit cercle noir au centre de l'œil et l'iris est la partie colorée de l'œil. La lumière rentre par la pupille à travers le cristallin. Le cristallin fait la mise au point d'un objet c'est-à-dire, Il permet de focaliser sur des objets rapprochés (le muscle ciliaire resserre le cristallin, ce qui l'épaissit) et sur des objets éloignés (le muscle amincit le cristallin).en dirigeant la lumière vers la partie arrière de l'œil. La partie arrière de l'œil est appelée la rétine et elle contient des capteurs (photorécepteurs) qui transposent l'intensité lumineuse captée en impulsions électriques transmises via le nerf optique au cerveau. Ces capteurs sont appelés cônes et les bâtonnets. L'information visuelle excite les cônes et les bâtonnets et ils sont impliqués dans la vision des couleurs en présence de la lumière.

Les capteurs traduisent l'information visuelle en information électrique. Le nerf optique va envoyer cette information au cerveau. Les minuscules cellules nerveuses sont capables de prendre la forme électrique de l'image en face de nous et de l'envoyer au cortex visuel du cerveau (c'est une partie arrière du cerveau). Le cortex visuel du cerveau est responsable du décodage de l'information électrique provenant de la rétine. Il analyse et interprète la forme électrique de l'image, ce qui permet de former une carte visuelle.

La compréhension du mécanisme du système visuel humain repose sur la modélisation de ce dernier en vue d'en simuler son fonctionnement.

- L'œil humain est plus sensible aux changements de luminosité qu'aux changements de couleur.
- Notre système visuel est sensible au mouvement. Dans notre champ de vision, nous pouvons rapidement reconnaître si quelque chose bouge.
- Notre système visuel est plus sensible au contenu basse fréquence qu'au contenu haute fréquence. Le contenu basse fréquence correspond à des changements d'intensité lents. Tandis que, le contenu haute fréquence correspond à des changements d'intensité rapides.

Nous aurons sûrement remarqué qu'il est facile de voir s'il y a des taches sur une surface plane, mais il est difficile de repérer quelque chose comme cela sur une surface très texturée.

3.4 Comment les êtres humains comprennent-ils le contenu de l'image ?

Le cerveau humain est constitué de près d'un tiers des aires cérébrales visuelles et chacune spécialisée dans un type de traitement particulier, du plus perceptif au plus cognitif. Les informations visuelles sont transmises sous forme d'influx électrique de l'œil au cerveau. La scène va être reconstruite au niveau cérébral en fonction des différentes informations portant sur la couleur, le mouvement, la forme, la localisation spatiale etc. que les aires cérébrales vont analyser. Les aires visuelles vont réaliser des traitements de plus en plus complexes pour analyser les informations spatiales de la scène ainsi que les informations des détails et des couleurs impliquées dans la reconnaissance de formes. De plus, le cerveau visuel communique continuellement avec le reste des aires cérébrales comme celles du langage, de la mémoire ou des émotions qui amènent du sens à ce que nous voyons, et qui peuvent également influencer notre perception visuelle. C'est une hiérarchie des zones de notre cerveau qui nous aide à reconnaître des objets.

Grace à ce système visuel, nous pouvons reconnaître différents objets sans effort et pouvons regrouper des objets similaires. Nous pouvons faire ce regroupement parce que nous avons développé un mécanisme d'invariance vis-à-vis des objets de la même classe. Lorsque nous regardons un objet, notre cerveau extrait les points saillants comme par exemple l'orientation, la perspective, la taille et la luminance etc. Par exemple, une chaise qui fait le double de la taille

normale et qui tourne de soixante degrés est toujours une chaise. Nous pouvons facilement la reconnaître à cause de la façon dont nous la traitons.

Dans notre système visuel, nous construisons ces invariances hiérarchiques en ce qui concerne la position, l'échelle et le point de vue qui nous aident à être très robustes. Les humains ont tendance à se souvenir d'un objet en fonction de sa forme et de ses caractéristiques importantes. Peu importe, la manière dont l'objet est placé, nous pouvons toujours le reconnaître. Les machines ne peuvent pas le faire si facilement.

3.5 Pourquoi est-il difficile pour les machines de comprendre le contenu des images ?

Nous pouvons rencontrer de nombreux objets chaque jour et sans effort, nous pouvons les reconnaître presque immédiatement. Par exemple, lorsque vous voyez une table, vous n'attendez pas quelques minutes avant de vous rendre compte que c'est bien une table. D'un autre côté, les ordinateurs trouvent qu'il est très difficile d'accomplir cette tâche. Les chercheurs travaillent depuis des décennies pour comprendre pourquoi les ordinateurs ne sont pas aussi performants que nous ?

Pour obtenir une réponse à cette question, nous devons comprendre comment les humains le font. Nous avons parlé dans la section précédente comment les données visuelles pénètrent dans le système visuel humain et comment notre système les traite. Le problème resté toujours est que nous ne comprenons pas complètement comment notre cerveau analyse, organise et reconnaît ces données visuelles. Par ailleurs, la reconnaissance des objets par l'ordinateur commence par l'extraction de certaines caractéristiques des images. Ensuite, l'ordinateur va exploiter ces caractéristiques à l'aide des algorithmes d'apprentissage automatique.

L'extraction de caractéristiques se fait à partir des variations telles que la taille, la forme, la perspective, l'illumination, l'angle, l'occlusion, etc. Le problème rencontré dans la reconnaissance des objets par l'ordinateur est que la table elle-même est complètement différente lorsqu'elle est présentée d'un autre point de vue. Les humains peuvent facilement reconnaître qu'il s'agit d'une table, quelle que soit la manière dont elle nous est présentée. Alors, comment pouvons-nous expliquer cela à nos machines ?

Une solution consiste à stocker toutes les variations d'un objet, y compris les angles, les tailles, les couleurs, les perspectives, etc. Mais ce processus prend du temps et est très stressant !

De plus, il est pratiquement impossible de collecter des données à partir de toutes les variables d'objet. En outre, pour reconnaître ces objets, il faudra des machines qui consomment énormément de mémoire et beaucoup de temps pour créer un modèle capable de le faire. Même avec tout cela, si l'objet est partiellement caché, les ordinateurs ne pourront pas le reconnaître. C'est parce qu'ils pensent que c'est un nouvel objet.

4 Conclusion

La vision par ordinateur est utilisée dans divers domaines. À mesure que cette technologie évolue, les possibilités d'améliorer son utilisation devraient augmenter au cours des prochaines années. Toutes les machines pourront bientôt voir et penser exactement de la même manière que les humains.

De nombreuses applications de vision par ordinateur reposent sur la représentation des images avec un nombre réduit de points clés et des descripteurs. De plus, les descripteurs de caractéristiques locales s'avèrent être un bon choix pour les tâches de mise en correspondance des images. Pour certaines applications, telles que le calibrage de la caméra, le suivi, la récupération des images, la classification des images et la reconnaissance des objets est essentiel que les détecteurs de caractéristiques et les descripteurs soient robustes aux changements de point de vue ou de luminosité et aux distorsions d'image (par exemple, bruit ou illumination) la rotation et la mise à l’échelle des images (Chen, et al., 2010). Tandis que, d'autres tâches de reconnaissance visuelle particulière, telles que la détection ou la reconnaissance de visage, nécessitent l'utilisation de détecteurs et de descripteurs spécifiques (Viola, & Jones, 2004).

III. Détecteurs des caractéristiques locales, descripteurs et représentation des images

1 Introduction

Depuis la naissance de la vision par ordinateur, les détecteurs de caractéristiques occupent une place importante dans la recherche. En effet, nombreuses applications sont proposées dans des domaines telles que la classification et récupération des images (Liu, Bai, 2012), la représentation des images (Yap, Jiang, & Kot, 2010), la reconnaissance et l'appariement des objets (Andreopoulos, & Tsotsos, 2013), le suivi du mouvement (Takacs et al., 2013), la reconstruction des scènes en 3D (Moreels,& Perona, 2005), la localisation du robot (Murillo, Guerrero, & Sagues, 2007), la classification des textures (Lazebnik, Schmid, & Ponce, 2005) etc.

Une caractéristique dans une image représente une structure spécifique et significative dans l'image. Les caractéristiques peuvent aller d'un pixel unique aux bords de l'mage puisqu'aux contours des objets dans l'image. La détection de caractéristiques est de rechercher les structures ayant des propriétés locales significatives dans une image. Le résultat de la détection de caractéristiques est repéré généralement par les emplacements spécifiques dans une image, appelés points de caractéristiques. Les points de caractéristiques sont appelés les points clés (ou intérêts) dans l'image.

Ces emplacements sont détectés en fonction de leur invariance à la translation, rotation, échelle, à la transformation affine, à la présence de bruit, flou, etc. Tuytelaars et Mikolajczyk,

(2007) présentent un aperçu des détecteurs de points d'intérêt invariants. Les points de caractéristiques sont souvent utilisés par un descripteur pour représenter des images. Les points de caractéristiques représentent le voisinage autour de chaque point clé détectés dans un vecteur de caractéristique efficace et discriminant appelé le descripteur de caractéristique qui est invariante à certaines transformations (Hassaballah, Ali, Alshazly, 2016). Dans la littérature, certains détecteurs de caractéristiques bien connus fournissent des descripteurs de caractéristiques.

2 Propriétés des caractéristiques de l'image

Le but principal des caractéristiques locales est de fournir une correspondance entre les images en utilisant ces caractéristiques. Pour atteindre cet objectif, les détecteurs et les extracteurs des caractéristiques doivent avoir certaines propriétés dans les applications en vision par ordinateur (Hassaballah, Ali, & Alshazly, 2016).):

- **Robustesse** : La détection de caractéristiques devrait être aux mêmes emplacements et indépendamment de la rotation, du décalage, de la mise à l'échelle, de l'illumination, du bruit et des artefacts de compression.
- **Répétabilité** : Les caractéristiques détectées dans un même objet devrons être toujours les mêmes quelque soit les conditions de visualisation.
- **Précision** : Concernant les tâches de mise en correspondance des images, la détection de caractéristiques de l'image doit être localisée avec précision (mêmes emplacements de points clés).
- **Efficacité** : La détection de caractéristiques devrait être rapide pour les nouvelles images afin d'implémenter des applications en temps réel.
- **Quantité** : Il faut détecter la totalité des caractéristiques de l'image pour fournir une représentation globale de l'image.

3 Extraction des caractéristiques de l'image

Les détecteurs de caractéristiques peuvent être classés en quatre catégories : les détecteurs de coins, les détecteurs des blob, les détecteurs de caractéristiques invariantes affines et les descripteurs de caractéristiques (Loncomilla, Ruiz-del-Solar, & Martínez, 2016; Uchida, 2016).

La figure 4 montre quelques exemples où une partie particulièrement importante des caractéristiques locales et des représentations des images est montrée.

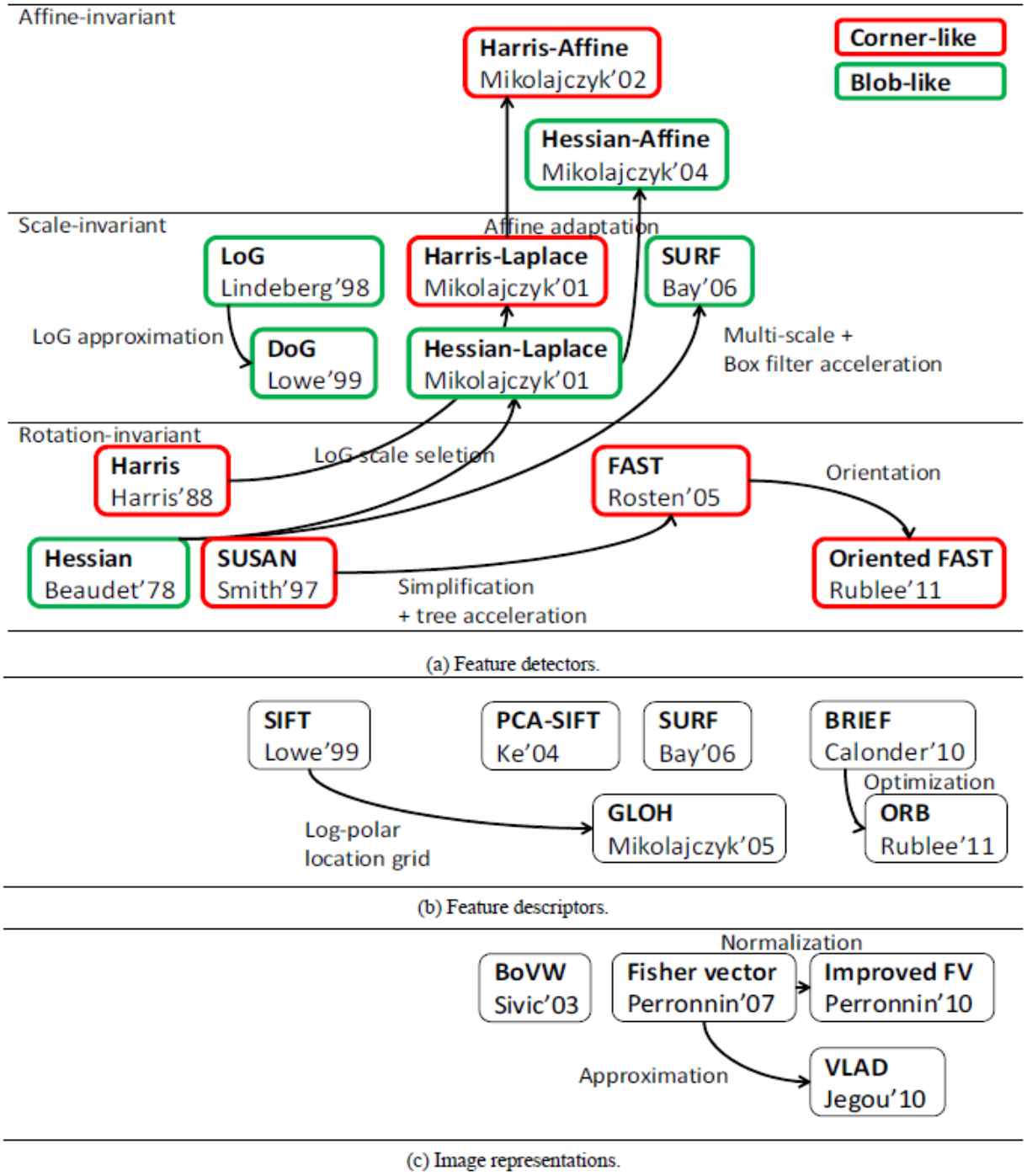


Figure 4. Aperçu de l'historique des caractéristiques locales et des représentations d'images (Loncomilla, Ruiz-del-Solar, & Martínez, 2016).

III. Détecteurs des caractéristiques locales, descripteurs et représentation des images

3.1 Détecteurs de caractéristiques invariants à la rotation

Les coins sont des caractéristiques préférées en vision par ordinateur en raison de leur contrainte bidimensionnelle et de leurs algorithmes rapides pour les détecter.

3.1.1 Détecteur Hessian

Le détecteur de Hessian (Beaudet, 1978.), ce détecteur recherche les emplacements dans une image possédant des dérivées fortes dans les deux directions orthogonales. Ce détecteur est basé sur la matrice des dérivées secondes, à savoir la matrice Hessian (H):

$$H(x, y, \sigma) = \begin{bmatrix} L_{xx}(x, y, \sigma) & L_{xy}(x, y, \sigma) \\ L_{xy}(x, y, \sigma) & L_{yy}(x, y, \sigma) \end{bmatrix} \quad (3.1)$$

Où $L(x, y, \sigma)$ est une image lissée par le noyau gaussien $G(x, y, \sigma)$:

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y) \quad (3.2)$$

Et L_{xx}, L_{xy} et L_{yy} sont des dérivées d'image de second ordre calculées à l'aide de la fonction gaussienne de l'écart type σ :

$$L_{xx}(x, y) = \frac{d^2}{dx^2} I(x, y) = L(x + 1, y) - 2L(x, y) + L(x - 1, y) \quad (3.3)$$

$$L_{yy}(x, y) = \frac{d^2}{dy^2} I(x, y) = L(x, y + 1) - 2L(x, y) + L(x, y - 1) \quad (3.4)$$

$$L_{xy}(x, y) = \frac{d^2}{dxdy} I(x, y) \quad (3.5)$$

$$= \frac{L(x + 1, y + 1) - L(x + 1, y - 1) - L(x - 1, y + 1) + L(x - 1, y - 1)}{4}$$

Le détecteur Hessian détecte le point (x, y) en tant que point de caractéristique tel que le déterminant de la matrice H des 8 pixels voisins du point (x, y) soit local-maximal avec :

$$\det(H) = L_{xx} L_{yy} - L_{xy}^2 \quad (3.6)$$

3.1.2 Détecteur Harris

Le principe du fonctionnement du détecteur de Harris (Harris, & Stephens, 1988) est de repérer un point dans lequel les directions de ces deux arêtes (en anglais *edges*) se changent, puisqu'il s'agit de l'intersection de deux arêtes.

Par conséquent, les angles représentent une forte variation du gradient dans l'image (dans les deux directions), ce qui peut être utilisé pour la détection des coins.

Une fenêtre w (x, y) (avec les déplacements *u* dans la direction *x* et *v* dans la direction *y*) qui va être déplacé dans une image en niveaux de gris *I* en calculant la variation d'intensité.

$$E(u,v) = \sum_{x,y} w(x,y)\,[I(x+u,y+v) - I(x,y)]^2 \qquad (3.7)$$

Où :

$I(x,y)$ est l'intensité à (x,y)

$I(x+u,y+v)$ est l'intensité de la fenêtre déplacée $(x+u,y+v)$

$w(x,y)$ est la fenêtre à la position(x,y).

Ensuite, des fenêtres à forte variation d'intensité qui présentent des fenêtres à coins sont recherchées. Par conséquent, nous devons maximiser l'équation ci-dessus doit être maximisée :

$$\sum_{x,y} [I(x+u,y+v) - I(x,y)]^2 \qquad (3.8)$$

Utilisation de l'extension Taylor:

$$E(u,v) \approx \sum_{x,y} \left[I(x,y) + uI_x + vI_y - (x,y)\right]^2 \qquad (3.9)$$

Développer l'équation :

$$E(u,v) \approx \sum_{x,y} u^2 I_x^2 + 2uvI_xI_y + v^2 I_y^2 \qquad (3.10)$$

Ce qui peut être exprimé sous la forme d'une matrice :

$$E(u,v) \approx [u \;\; v] \left(\sum_{x,y} w(x,y) \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} \right) \qquad (3.11)$$

Notons :

$$M = \sum_{x,y} w(x,y) \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix} \qquad (3.12)$$

Ici, I_x et I_y sont des dérivées de l'image respectivement dans les directions x et y. (Ceux-ci peuvent être facilement trouvés en utilisant l'opérateur de *Sobel*.

Donc, l'équation (3.11) devient :

$$E(u,v) \approx [u \;\; v] \; M \begin{bmatrix} u \\ v \end{bmatrix} \qquad (3.13)$$

Un score R est calculé pour chaque fenêtre afin de déterminer si elle peut éventuellement contenir un coin :

$$R = \det(M) - k(trace(M))^2 \qquad (3.13)$$

Où :

$$\det(M) = \lambda_1 \lambda_2 \qquad (3.14)$$

$$trace(M) = \lambda_1 + \lambda_2 \qquad (3.15)$$

k est une valeur généralement comprise entre 0,04 et 0,06.

Une fenêtre avec une valeur R supérieur à une certaine valeur est considérée comme un "coin".

λ_1 et λ_2 sont les valeurs propres de M. Ainsi, les valeurs propres déterminent si une région est un coin, un bord ou un plat.

- Quand R << 0, ce qui arrive quand λ_1 et λ_2 sont petits, la région est plate.

- Lorsque $R < 0$, ce qui se produit lorsque $\lambda_1 >> \lambda_2$ ou inversement, la région est une arête.
- Lorsque $R > 0$, ce qui arrive quand λ_1 et λ_2 sont grands et la région est un coin.

3.1.3 Détecteur SUSAN

Smith et Brady (1997) ont présenté une technique d'extraction des caractéristiques de bas niveau d'image appelée *Smallest Univalue Segment Assimilating Nucleus* (SUSAN). Cette technique est utilisée pour extraire des coins et des bords et elle permet de réduire le bruit de l'image.

Un coin est détecté en plaçant un masque circulaire de rayon fixe sur chaque pixel de l'image. Les pixels de la zone située sous le masque sont comparés au noyau qui est le pixel central du masque pour vérifier s'ils ont des valeurs d'intensité similaires ou différentes. La zone résultante de cette vérification est appelée *Univalue Segment Assimilating Nucleus* (USAN). La figure 5 représente un rectangle d'ombre sur un fond blanc, où le masque peut prendre de différents emplacements dans une image. La détection des coins peut être basée sur la zone USAN. Lorsque la région USAN est plus petite, le noyau présente un coin, comme l'emplacement «a» indiqué dans la figure 5.

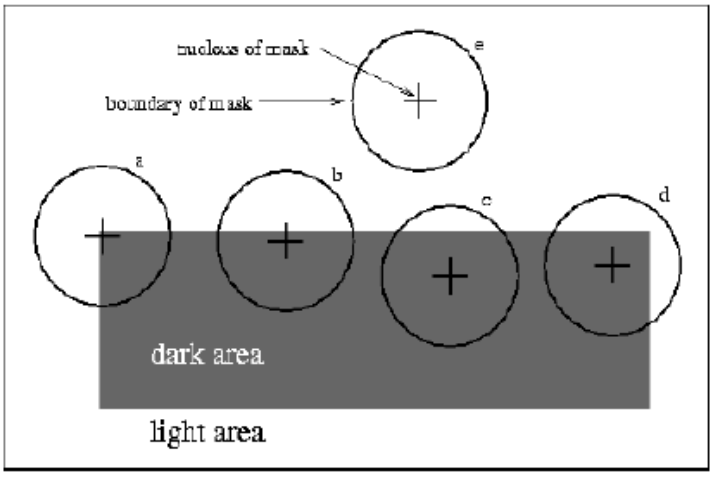


Figure 5. Quatre masques circulaires à différents endroits sur une image simple (Smith, & Brady, 1997)

Pour la détection de coins, la fonction de comparaison *C (r, r_0)* entre chaque pixel du masque et le noyau du masque est donné par l'équation 3.16

$$c(r, r_0) = \begin{cases} 1, & |I(r) - I(r_0)| \leq t \\ 0, & sinon \end{cases} \quad (3.16)$$

Un coin se trouve à des emplacements où le nombre de pixels dans USAN atteint un minimum local c'est-à-dire ce nombre est inférieur à une valeur de seuil spécifique *t*.

r_0 représente les coordonnées du noyau et *r* représente les coordonnées des autres pixels du masque ;

c (r, r_0) est le résultat de la comparaison ; *I (r)* est la valeur de l'intensité du pixel du point ;

t est la plus faible valeur pouvant être détecté par le détecteur SUSAN.

La taille de la région USAN est donnée par :

$$n(r_0) = \sum_{rec\,(r_0)} c(r, r_0) \quad (3.17)$$

Ensuite, *n* est comparé à un seuil géométrique *g*, qui est fixé à la moitié de la zone de masque (nombre total de pixels dans le masque). Pour détecter les coins, la fonction suivante est utilisée :

$$R(r_0) = \begin{cases} g - n(r_0), & n(r) \geq g \\ 0, & n(r) \geq g \end{cases} \quad (3.18)$$

Les performances du détecteur de coin SUSAN dépendent principalement de la fonction de comparaison similaire *c(r, r_0)*, qui peut être affecté par le changement de la luminance et des bruits.

3.1.4 Détecteur FAST

Features from Accelerated Segment Test (FAST) est un détecteur de coin développé à l'origine par (Rosten, & Drummond, 2006; Rosten, Porter, & Drummond; 2010). L'avantage principal du détecteur FAST est son efficacité en termes de temps de calcul. Il est très approprié

pour les applications de traitement vidéo en temps réel en raison de ses performances à haute vitesse.

Le détecteur FAST détecte les pixels plus clairs ou plus foncés que les pixels voisins basés sur le segment. Pour chaque pixel *P*, les intensités de 16 pixels sur un cercle de *Bresenham* de rayon 3 sont comparés à celui de *P*, et sont classés en trois types : plus brillant, similaire, et plus sombre. La figure 6 montre un cercle de 16 pixels (indiqué par des lignes pointillées blanches) autour du pixel *p* à tester.

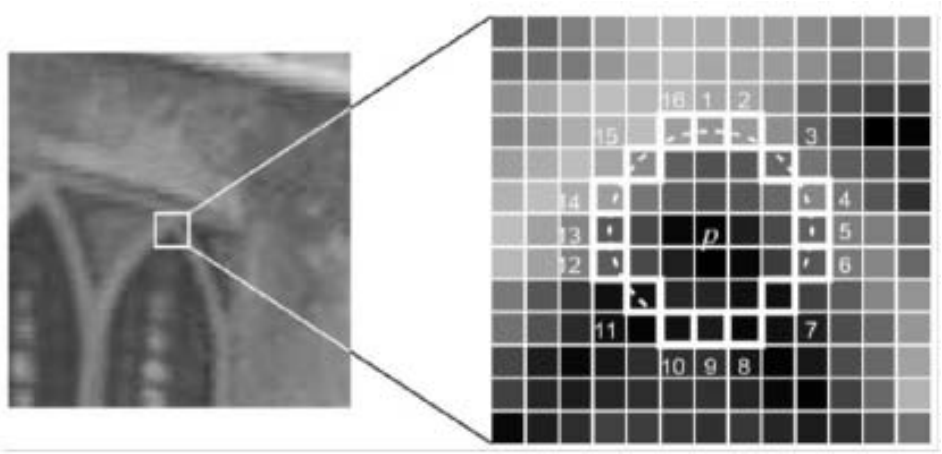


Figure 6. Image affichant le point d'intérêt sous test et le cercle de 16 pixels (Rosten, & Drummond, 2006; Rosten, Porter, & Drummond; 2010).

Un résumé de base de l'algorithme est présenté ci-dessous.

1. Sélectionner un pixel *p* dans l'image à identifier comme un point d'intérêt qui représente l'intensité I_p. Ce pixel peut être spécifié comme point d'intérêt ou non.

2. Sélectionner la valeur de seuil *t*.

3. Obtenir *t* qui représente la valeur de l'intensité du seuil.

4. Pour spécifier que *P* est un coint, il faut avoir 12 pixels contigus du cercle de 16 pixels qui ont tous une valeur supérieure ou inferieure au seuil *t*.

5. Pour alléger le traitement, un test à haute vitesse a été proposé. Ce test examine uniquement les quatre pixels en 1, 9, 5 et 13 (les premiers 1 et 9 sont testés, s'ils sont trop clairs

ou trop foncés. On passe à la vérification des points 5 et 13). Si *P* est un coin, alors au moins trois d'entre eux doivent tous être plus brillants ou plus foncés que I_p. Si la condition précédente n'est pas satisfaite, alors *p* ne peut pas être un coin. Ce détecteur lui-même présente des performances élevées, mais il a plusieurs faiblesses.

3.1.5 Détecteur AGAST

La détection générique des coins est basée sur le détecteur de caractéristiques *Accelerated Segment Test* (AGAST) qui a été proposée par Mair, Hager, Burschka, Suppa, et Hirzinger, (2010). Le détecteur AGAST est basé sur le même critère d'extraction de caractéristiques de FAST, mais il utilise un arbre de décision différent. AGAST permet de modifier automatiquement les arbres de décision en introduisant un algorithme de commutation d'arbre dynamique. De plus, Un arbre est formé à partir des zones homogènes et l'autre zone hétérogène. AGAST est formé sur la base d'un ensemble de données incluant toutes les combinaisons possibles de 16 pixels sur le cercle qui tourne au tour du pixel candidat. De cette manière, les performances AGAST augmente pour les scènes aléatoires, et fonctionne dans tous les environnements sans aucune étape de formation nécessaire. Cela rend AGAST très prometteur pour des applications de vision par ordinateur en temps réel.

3.2 Détecteurs des caractéristiques invariants à l'échèle

3.2.1 Laplacian of Gaussian (LoG)

Dans cette section, nous détaillons laplacien de gaussien, aussi appelé *Laplacian of Gaussian* (LoG) en anglais proposé par Lindeberg (1998), un filtre dérivé de second ordre. L'opérateur de laplacien est défini dans l'équation 3.19:

$$\nabla^2 = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} \qquad (3.19)$$

La détection des bords en utilisant des filtres du premier ordre est basée sur la localisation des maxima ou minima locaux. Par contre, l'opérateur laplacien détecte les bords aux passages par zéro, c'est-à-dire là où la valeur passe de négative à positive et vice-versa.

L'utilisation des approximations de différences finies pour la dérivée de premier ordre permet d'obtenir des noyaux laplacien des équations 3.20 et 3.21 :

$$\frac{\partial^2 f}{\partial x^2} = f(x+1) + f(x-1) - 2f(x) \rightarrow \underset{x\ kernel}{[1 \quad -2 \quad 1]} \qquad (3.20)$$

$$\frac{\partial^2 f}{\partial y^2} = f(y+1) + f(y-1) - 2f(y) \rightarrow \underset{y\ kernel}{\begin{bmatrix} 1 \\ -2 \\ 1 \end{bmatrix}} \qquad (3.21)$$

La somme des deux noyaux des équations 3.20 et 3.21 de laplacien indiqué dans la figure 7.

0	0	0
1	-2	1
0	0	0

+

0	1	0
0	-2	0
0	1	0

=

0	1	0
1	-4	1
0	1	0

Figure 7 Le noyau laplacien

Parmi les propriétés du laplacien est qu'il est très sensible au bruit. Afin de réduire l'effet du bruit, l'image est d'abord lissée avec un filtre gaussien, puis les passages par zéro sont recherchés en utilisant le noyau laplacien. Ce processus effectué en deux étapes est appelé opération du laplacien de gaussien (LoG).

Mais cela peut également être effectué en une seule étape. Au lieu de lisser d'abord une image avec un noyau gaussien puis trouver son noyau laplacien, il est possible d'obtenir le laplacien du noyau gaussien puis le convoluer avec l'image. Ceci est montré dans l'équation 3.22 où f est l'image et g est le noyau gaussien.

$$\nabla^2 (f * g) = f * \nabla^2 g = f * \frac{\sigma^2}{\sigma x^2} g + f * \frac{\sigma^2}{\sigma y^2} g \qquad (3.22)$$

LoG peut être donné par l'équation 3.23

$$Log(x,y) = -\frac{1}{\pi\sigma^4}\left[1 - \frac{x^2+y^2}{2\sigma^2}\right] e^{-\frac{x^2+y^2}{2\sigma^2}} \qquad (3.23)$$

3.2.2 Difference of Gaussian (DoG)

La différence de gaussiennes en anglais est *Difference of Gaussian* (DoG) (Lowe, 2004). L'utilisation de DoG dans une image est d'abord commencée par le lissage en utilisant la convolution avec un noyau gaussien d'une certaine largeur du paramètre σ_1

$$G_{\sigma 1}(x,y) = -\frac{1}{\sqrt{2\pi\sigma_1^2}}\exp\left(-\frac{x^2+y^2}{2\sigma_1^2}\right) \tag{3.24}$$

Pour obtenir l'image lissée g_1(x, y) :

$$g_1(x,y) = G_{\sigma 1}(x,y) * f(x,y) \tag{3.25}$$

Une deuxième image lissée peut être obtenue en utilisant une autre largeur différente σ_2:

$$g_2(x,y) = G_{\sigma 2}(x,y) * f(x,y) \tag{3.26}$$

Il est possible de montrer que la différence de ces deux images lissées gaussiennes donne la définition de la différence gaussienne (équation 3.27). Cette différence peut être utilisée pour détecter les contours de l'image.

$$\begin{aligned} g_1(x,y) - g_2(x,y) &= G_{\sigma 1} * f(x,y) - G_{\sigma 2} * f(x,y) = (G_{\sigma 1} - G_{\sigma 2}) * f(x,y) \\ &= DoG * f(x,y) \end{aligned} \tag{3.27}$$

DoG en tant qu'opérateur ou noyau de convolution est défini par l'équation 3.28

$$DoG \triangleq G_{\sigma 1} - G_{\sigma 2} = \frac{1}{\sqrt{2\pi}}\left(\frac{1}{\sigma_1}e^{-(x^2+y^2)/2\sigma_1^2} - \frac{1}{\sigma_2}e^{-(x^2+y^2)/2\sigma_2^2}\right) \tag{3.28}$$

3.2.3 Harris-Laplace

Harris et Stephens (1988) ont développé un détecteur combiné de coin (angle) et de bord pour répondre aux limites du détecteur de Moravec.

Les points de Harris sont invariants aux changements de rotation et d'éclairage. Mais, ces points ne sont pas invariants à l'échelle.

La matrice de second moment est utilisée dans le détecteur Harris-Laplace pour adapter la détection des coins Harris dans des différentes échelles. Cette matrice est représentée comme suit :

$$M(x, y, \sigma_I, \sigma_D) = \sigma_D^2 \, g(\sigma_I) \begin{bmatrix} I_x^2(x, y, \sigma_D) & I_x I_y(x, y, \sigma_D) \\ I_x I_y(x, y, \sigma_D) & I_y^2(x, y, \sigma_D) \end{bmatrix} \qquad (3.29)$$

Le paramètre σ_I détermine l'échelle actuelle à laquelle les points de coin de Harris sont détectés dans l'espace-échelle gaussien.

Le paramètre σ_D représente l'échelle dérivée qui fixe la taille des noyaux gaussiens utilisés pour calculer les dérivées.

I_x et I_y sont les dérivées d'image calculées dans leur direction respective en utilisant un noyau gaussien d'échelle σ_D.

La détection des coins dans des échelles multiples est calculée en utilisant le déterminant et la trace de la matrice de second moment adaptée comme

$$C(x, y, \sigma_I, \sigma_D) = \det\,[M(x, y, \sigma_I, \sigma_D)] - \alpha . trace^2\,[M(x, y, \sigma_I, \sigma_D)] \qquad (3.30)$$

La valeur de la constante α est comprise entre 0,04 et 0,06.

À chaque niveau de la représentation, les points d'intérêt sont extraits en détectant les maxima locaux dans le voisinage à 8 points (x, y).

Ensuite, un seuil est utilisé pour éliminer les points qui ont de petite information, car ils sont moins stables dans des conditions d'observation arbitraires

$$C(x, y, \sigma_I, \sigma_D) > threshold_{\,Harris} \qquad (3.31)$$

De plus, le laplacien de gaussien est utilisé pour trouver les maxima dans l'échelle.

Où, seuls les points pour lesquels la réponse du laplacien est supérieure à un seuil sont acceptés.

$$\sigma_I^2 \left| L_{xx}(x, y, \sigma_I) + L_{yy}(x, y, \sigma_I) \right| > threshold_{\ Laplacian} \qquad (3.32)$$

L'approche Harris-Laplace fournit un ensemble représentatif de points caractéristiques de l'image qui sont invariants aux changements d'échelle, à la rotation, à l'éclairage et à l'ajout de bruit. De plus, les points d'intérêt sont hautement reproductibles (répétabilité).

Il réduit également considérablement le nombre de points d'intérêt redondants par rapport à Harris multi-échelles. Cependant, le détecteur d'Harris-Laplace échoue dans le cas de transformations affines. Mais, il renvoie un nombre de points beaucoup plus faible que les détecteurs LoG ou DoG.

3.2.4 Hessian-Laplace

Le détecteur Hessian-Laplace proposé par Mikolajczyk et Schmid (2004), c'est un détecteur des points clés invariants à l'échelle dans une image. Ce détecteur utilise la matrice de Hessian pour localiser des points caractéristiques à l'aide du détecteur de Hessian à plusieurs échelles. Ensuite, ces points caractéristiques sont sélectionnés de la même manière que Harris-Laplace.

Le détecteur de Hessian-Laplace détecte des structures de type blob similaires aux détecteurs LoG ou DoG. En effet, l'utilisation du déterminant de la Hessian (Mikolajczyk et al., 2005) permet de détecter souvent des points caractéristiques sur les bords, contrairement à la Hessian-Laplace.

3.3 Détecteurs de caractéristiques affines invariantes

Mikolajczyk et al.(2005) ont présenté un état de l'art sur les détecteurs affines des régions covariantes et ils ont comparé leurs performances. Les détecteurs affines sont :

1) Le détecteur Harris-Affine; 2) Le détecteur Hessian-Affine; 3) Le détecteur MSER; 4) Le détecteur *Edge-Based Region* (EBR) et détecteur *Intensity Extremal-Based Region* (IBR).

3.3.1 Détecteur Harris-Affine

Le détecteur Harris-Affine (Mikolajczyk, & Schmid, 2002 ; Mikolajczyk, & Schmid ,2004; Schaffalitzky, & Zisserman, 2002), est un détecteur de caractéristiques invariant aux transformations de l'image affines. Les régions Harris-Affine résultantes sont caractérisées par des ellipses. Ce détecteur détecte tout d'abord les points caractéristiques à l'aide du détecteur Harris-Laplace. Ensuite, il affine itérativement ces régions en utilisant la seconde matrice de moment proposée par (Lindeberg, & Gårding, 1997; Mikolajczyk, & Schmid, 2001)

3.3.2 Détecteur Hessian-Affine

Le détecteur Hessian-Affine (Mikolajczyk, Schmid, 2004; Mikolajczyk, & Schmid, 2002) est un détecteur de caractéristiques affine-invariant qui est similaire au détecteur Harris-Affine. Il détecte tout d'abord les points caractéristiques à l'aide du détecteur Hessian-Laplace. Ensuite, il affine de manière itérative ces régions en utilisant la matrice du second moment comme cela est fait dans le détecteur Harris-Affine.

Le détecteur Harris-Affine est adopté pour obtenir des points caractéristiques et la région caractéristique d'une image. Les régions détectées en utilisant Hessian-Affine sont invariantes aux transformations des images affines. La localisation et l'échelle de ces régions sont estimées par le détecteur Hessian-Laplace.

Notez que Harris-Affine diffère de Harris-Laplace par l'adaptation affine, qui est appliquée aux régions de Harris-Laplace.

3.3.3 Détecteur MSER

Maximally Stable Extremal Region (MSER), ce détecteur a été proposé par Matas et al. (2004). La signification de «Extremal» indique que chaque pixel de la région MSER a une intensité plus claire ou sombre que tous les pixels de la zone extérieure (Patel, & Gurjwar, 2016). En d'autres termes, la région MSER est constituée de pixels homogènes par rapport à un seuil donné. La détection des régions MSER commence par le seuillage de l'intensité de l'image à toutes les 256 valeurs de gris. À chaque niveau de seuil, les pixels en dessous du seuil sont colorés en noir, tandis que les pixels au-dessus du seuil sont colorés en blanc.

La région «Maximally Stable» est celle qui maintient son état avec peu de changement sur plusieurs seuils d'intensité sélectionnés. Ces régions maximales qui sont stables sur une plage de seuils constituent les régions extrêmes maximales stables de l'image.

MSER est adopté comme détection BLOB ainsi qu'il est invariante à la transformation affine. Il détermine l'appariement entre les objets figurés dans deux images différentes dans des points de vue différents. En effet, Il permet d'obtenir de meilleures performances pour la correspondance entre les caractéristiques des objets (Kaushik, Rawat, & Bhalla, 2016). Ainsi que, il.est efficace pour reconnaître des objets.

3.3.4 Edge Based Regions (EBR) et Intensity Based Regions (IBR)

Tuytelaars et Gool (2004) ont proposé deux types de détecteurs complètement différents par rapport aux détecteurs cités ci-dessus. Le premier est appelé le détecteur de régions basées sur les bords, aussi appelé *Edge Based Regions* (EBR) en anglais et le second est le détecteur de région basé sur l'intensité, en anglais est *Intensity Based Regions* (IBR).

Le détecteur EBR exploite le comportement des arêtes autour d'un point d'intérêt. Elle exploite des coins de Harris et des bords qui se croisent à proximité. Son principe est basé sur la détection de point Harris (P) et sur l'exploitation des informations de bord à proximité de ce point. Le point de coin (P) et deux points se déplaçant le long des deux bords (P_1 et P_2) définissent un parallélogramme.

Alors que, le deuxième détecteur est IBR qui trouve des caractéristiques aux coins et aux bords, les IBR extraient des régions affines en fonction des propriétés d'intensité. IBR permettent d'explorer les pixels voisins autour d'un point extrême d'intensité dans l'image.

Le processus d'utilisation ce détecteur commence tout d'abord par le lissage de l'image, puis les extrema locaux sont sélectionnés en utilisant une suppression des valeurs des pixels non maximale. Ces points ne peuvent pas être détectés avec précision, mais ils sont résistants aux transformations monotones d'intensité.

3.4 Descripteurs de caractéristiques basés sur l'extraction des caractéristiques

Les points clés (intérêt) détectés sont fournis souvent des caractéristiques d'invariance d'échelle, de rotation et d'éclairage pour le descripteur. Le descripteur permet d'ajouter plus de détails et plus d'attributs d'invariance. Il admet d'extraire des caractéristiques autour de chaque point d'intérêt. Le descripteur fait la correspondance entre les caractéristiques, c'est-à-dire, il détermine la correspondance entre d'emplacements des caractéristiques dans différentes images. Plusieurs descripteurs cités dans la littérature sont basés sur la détection des points clés pour décrire des caractéristiques dans une image.

3.4.1 HOG

Histogram of Oriented Gradients (HOG) (Dalal & Triggs, 2005) est un descripteur de caractéristiques largement utilisé dans plusieurs domaines pour extraire des caractéristiques des objets à travers leurs formes. La forme et l'apparence d'un objet local peuvent souvent être décrites par la distribution d'intensité des gradients locale ou des directions des bords.

La première étape de la détection HOG consiste à diviser l'image sous forme de blocs (par exemple 16 × 16 pixels). Chaque bloc est divisé en petites cellules (par exemple 8 × 8 pixels) de sorte que la même cellule peut être dans plusieurs blocs. Pour chaque pixel de la cellule, les gradients verticaux et horizontaux sont obtenus. Pour ce faire, la méthode la plus simple consiste à utiliser des opérateurs verticaux et horizontaux 1-D Sobel:

$$\boldsymbol{Gx}(\boldsymbol{x},\boldsymbol{y}) = \boldsymbol{I}(\boldsymbol{y},\boldsymbol{x}+1) - \boldsymbol{I}(\boldsymbol{y},\boldsymbol{x}-1); \qquad \boldsymbol{Gy}(\boldsymbol{x},\boldsymbol{y}) = \boldsymbol{I}(\boldsymbol{y}+1,\boldsymbol{x}) - \boldsymbol{I}(\boldsymbol{y}-1,\boldsymbol{x}) \quad (3.33)$$

$\boldsymbol{Gx}(\boldsymbol{x},\boldsymbol{y})$ est le gradient horizontal et $\boldsymbol{Gy}(\boldsymbol{x},\boldsymbol{y})$ est le gradient vertical. $\boldsymbol{I}(\boldsymbol{y},\boldsymbol{x})$ est l'intensité des pixels aux coordonnées *x* et *y*. L'amplitude et la phase du gradient sont déterminées comme suit:

$$G(x,y) = \sqrt{G_x(x,y)^2 + G_y(x,y)^2} \quad , \quad \theta(x,y) = \arctan\left(\frac{G_y(x,y)}{G_x(x,y)}\right) \qquad (3.34)$$

En effet, pour chacune de ces cellules, un histogramme de gradients orientés est calculé en utilisant 9 orientations (ou bins): Une orientation généralement non signée est utilisée, donc les angles inférieurs à 0° sont augmentés de 180°.

Les histogrammes de gradient de chaque cellule qui constituent un bloc doivent être concaténés pour former un seul descripteur qui représente ce bloc en utilisant les normes *L1* ou *L2* :

$$L1 - norm = \frac{v}{\sqrt{|v|_2^2 + s^2}} \qquad L2 - nomr = \frac{v}{\sqrt{|v|_1 + s^2}}$$

$$L1\ sqrt - norm = \sqrt{\frac{v}{|v|_1 + s}} \qquad (3.35)$$

Une fois tous les blocs sont normalisés, nous prenons les histogrammes résultants, les concaténons et les traitons comme le vecteur de caractéristique final qui représente l'image.

3.4.2 SIFT

L'algorithme *Scale Invariant Feature Transform* (SIFT), proposé par Lowe (1999), est considéré parmi l'un des premiers descripteurs à fournir une technique complète de détection de points clés et d'extraction des caractéristiques. Il se déroule en quatre étapes :

Étape 1: Détection des extrema de l'échelle de l'espace.

Pour construire une représentation des caractéristiques de l'image invariante à l'échelle, SIFT établi une pyramide multi-résolution sur l'image d'entrée et applique la différence des opérateurs gaussiens (DoG) pour localiser les extrema locaux dans l'espace d'échelle.

Étape 2: Localisation les points clés.

Les points d'intérêt sélectionnés sont capables de réaliser l'invariance par rapport au contraste, le bruit local et la présence de bord dans le voisinage des pixels locaux. Les points clés sont localisés avec une précision sur l'échelle et l'espace. Les points d'intérêt extraits sont donc invariants à l'échelle.

Diverses méthodes et techniques peuvent être utilisées afin de sélectionner les meilleurs points d'intérêt y compris l'interpolation de courbure locale sur de petites régions pour détecter des réponses des bords.

Étape 3: Estimation de l'orientation.

Une région locale ou un patch de taille 16x16 pixels entourant les points d'intérêt détectés sont la base du vecteur qui va représenter l'image. L'amplitude des gradients locaux dans le patch 16x16 et les orientations du graduant sont calculées et stockées dans un vecteur de caractéristique HOG.

Étape 4: Descripteur des points clés.

Extraction des gradients de l'image locale à l'échelle sélectionnée autour du point-clé permet de former une représentation invariante de diverses distorsions géométriques qui peuvent changer la position des gradients locaux.

Les gradients de l'image d'une fenêtre 16 × 16, centrée à chaque point clé, sont ensuite calculés et regroupés en 4 × 4 sous-régions. La direction des gradients dans la même sous-région est ensuite quantifiée pour conclure un histogramme à huit cases. En rassemblant toutes les cases des seize histogrammes dans la fenêtre, on obtient un vecteur descripteur SIFT à 128 éléments.

3.4.3 PCA-SIFT

L'algorithme SIFT-PCA développée par Ke et Suthankar (2004). Cet algorithme permet de réduire la dimensionnalité du descripteur SIFT à un plus petit ensemble d'éléments (attributs). SIFT-PCA utilise l'analyse en composantes principales en anglais est *Principal Component Analysis* (PCA), basé sur les patchs de gradient normalisés plutôt que sur les histogrammes pondérés et lissés des gradients, comme utilisé dans SIFT. Le descripteur SIFT a été initialement signalé en utilisant 128 valeurs, mais en utilisant SIFT-PCA, la taille du vecteur est réduite.

Les étapes de base pour SIFT-PCA sont les suivantes :

1. Construire un espace propre basé sur les gradients des patchs d'image locaux 41x41 résultant en un vecteur d'éléments 3042 ; ce vecteur est le résultat du pipeline SIFT normal.

2. Calculer l'image gradients locaux pour les patchs.

3. Créer le vecteur de caractéristiques de taille réduite à partir de l'espace en utilisant PCA sur la matrice de covariance de chaque vecteur.

Il est démontré que SIFT-PCA apporte certaines améliorations par rapport à SIFT dans le domaine de la robustesse à la déformation d'image, et la plus petite taille du vecteur de caractéristiques entraîne une vitesse de correspondance plus rapide. Les auteurs notent que même si l'ACP en général n'est pas optimale lorsqu'elle est appliquée aux caractéristiques des correctifs d'image, la méthode fonctionne bien pour les correctifs de dégradé de style SIFT qui sont orientés et localisés dans l'espace d'échelle.

3.4.4 GLOH

Le descripteur *Gradient Location and Orientation Histogram* (GLOH) est introduit par Mikolajczyk, & Schmid (2005). Il est conçu pour augmenter la robustesse et le caractère distinctif du descripteur bien connu SIFT qui intègre à la fois l'apparence locale et les informations des caractéristiques locales.

GLOH est semblable au descripteur SIFT et HOG puisqu'il est basé sur l'évaluation des orientations des histogrammes locaux normalisés du gradient d'image dans une grille dense. Plus spécifiquement, le descripteur GLOH peut être obtenu en calculant le descripteur SIFT ou HOG pour une grille de localisation log-polaire à trois rayons et huit angles. Les orientations de gradient sont ensuite quantifiées en 16 parties et ainsi le descripteur résultant donne un histogramme de 272 cases. La taille est réduite à 128 par PCA. Ce dernier est utilisé pour réduire la dimension dans l'espace de représentation vectorielle

GLOH est une méthode de description de forme locale, très similaire au SIFT. Et la seule différence par rapport au SIFT est qu'il utilise une grille de localisation polaire logarithmique :

- 3 bacs en direction radiale
- 8 bacs en direction angulaire
- 16 cases pour la quantification de l'orientation du gradient

- Le descripteur GLOH est donc un vecteur de dimension supérieure avec un total de 17 (soit 2x8 + 1) * 16 = 272 cases.

3.4.5 SURF

Bay, Tuytelaars et Van Gool, (2006) ont proposé un algorithme pour représenter une image sous forme d'un descripteur très robuste appelé *Speeded Up Robust Features* (SURF). La caractéristique la plus importante de SURF est la rapidité. Cet algorithme garantit également des performances de répétabilité, de caractère distinctif et de robustesse. SURF utilise l'image intégrale, la réponse en ondelettes Haar et la matrice approximative de Hessian pour réduire considérablement le temps de calcul et en même temps pour augmenter sa robustesse. SURF en général peut être divisé en quatre étapes : détection et localisation des points clés, affectation de l'orientation, descripteur d'image local et correspondance des points clés. La description spécifique est la suivante :

3.4.5.1 Détection et localisation de points clés

Dans la localisation de points clés, l'algorithme utilise une matrice Hessian qui est approximative en utilisant des images intégrales, permettant ainsi des performances élevées sans perte de précision. Le détecteur, prend le nom de Fast-Hessian. L'algorithme SURF utilise un filtre gaussien qui permet une analyse spatiale et des facteurs d'échelle plus larges que l'algorithme SIFT.

3.4.5.2 Affectation de l'orientation

Afin d'être invariant à la rotation de l'image, une orientation dominante pour chaque point clé est identifiée. En effet, dans cette étape un voisinage circulaire centré avec un rayon fixe est calculé (à n'importe quelle échelle). Ensuite, dans chaque zone, une ondelette de Haar dans les directions x et y est calculée. La direction dominante est estimée en quantifiant la somme de toutes les réponses de Haar ondelettes au moyen d'une fenêtre coulissante avec une taille fixe (généralement π / 3).

À chaque position, les réponses horizontales et verticales dans la fenêtre coulissante sont additionnées et utilisées pour former un nouveau vecteur. Le plus long vecteur de ce type sur toutes les fenêtres est attribué comme orientation du point clé.

3.4.5.3 Descripteur d'image local

Une fois l'orientation de chaque point-clé est identifié, le descripteur lié est implémenté à l'étape du descripteur de point-clé. En particulier, la région est divisée en un nombre fixe de sous-régions (généralement, une grille de 4 × 4 secteurs).

Pour chacune des sous-régions, la réponse Haar-ondelettes dans la direction x et y est calculée. Afin d'augmenter la robustesse vis-à-vis des déformations géométriques, les réponses des ondelettes sont calculées avec une fonction gaussienne centrée sur le point d'intérêt. Enfin, les différentes réponses sont additionnées entre elles, formant ainsi un premier ensemble de valeurs liées au descripteur. Pour conserver également des informations sur la polarité des changements d'intensité, les valeurs absolues des sommes obtenues sont également calculées (c.-à-d. |dx| et |dy|) et forment un vecteur de caractéristiques. Ainsi, pour chaque point clé, il en résulte un vecteur descripteur de longueur 64. Enfin, le descripteur SURF est normalisé pour le rendre invariant aux différents changements.

3.4.6 KAZE

KAZE est introduit par Alcantarilla, Bartoli et Davison (2012). L'idée derrière sa création est de détecter et de décrire des caractéristiques des images et d'obtenir une meilleure précision de localisation et un caractère distinctif à différents niveaux d'échelle. KAZE introduit un nouveau schéma pour créer l'espace d'échelle en utilisant le filtrage de diffusion non-linéaire, appelé en anglais *Nonlinear Diffusion Filtering* (NDF) (Perona & Malik, 1990). Cela rend le flou dans les images localement adaptable aux points caractéristiques, réduisant ainsi le bruit et conservant simultanément les limites des objets dans l'image.

Les techniques précédentes, telles que SIFT et SURF trouvent des caractéristiques multi-échelles en construisant un espace à l'échelle gaussienne. La recherche des caractéristiques stables se fait à toutes les échelles à l'aide du filtrage de l'image par un noyau gaussien d'écart-type croissant. Les espaces à l'échelle gaussienne utilisés dans SIFT ou SURF ont la propriété indésirable de brouiller les bords des images, cela réduit la précision de localisation des points clés. En effet, le flou gaussien ne respecte pas les bords (les limites) des objets. Il permet de lisser les détails des objets de même degré que des bruits lors de l'application du flou gaussien

sur une l'image originale. Ceci permet de réduire la précision et le caractère distinctif de la localisation.

En revanche, KAZE rend le flou localement adaptable aux données de l'image au moyen de l'espace d'échelle non linéaire par un filtrage de diffusion non linéaire. Le filtrage de la diffusion pour que le bruit soit flou mais que les détails importants de l'image et l'objet des frontières ne seront pas affectées. La NDF est décrite par des équations différentielles partielles non linéaires (équations 3.36 et 3.37).

$$\frac{\partial L}{\partial t} = div\ (c(x,y,t).\nabla L) \qquad (3.36)$$

$$c(x,y,t) = g\left(\nabla L_\sigma(x,y,t)\right) \qquad (3.37)$$

L est la fonction de luminance (l'image)

div est l'opérateur de divergence

c(x ; y ; t) est une fonction de conductivité

Perona et Malik ont proposé de rendre la fonction de conductivité *c* dépendante de l'amplitude du gradient.

σ est le paramètre de flou (variance du gaussien)

Deux formules différentes pour la fonction de conductivité g_1(équation 3.38) et g_2(équation 3.39):

$$g_1 = \exp\left(-\frac{|\nabla L_\sigma|^2}{K^2}\right) \qquad (3.38)$$

$$g_2 = \frac{1}{1+\frac{|\nabla L_\sigma|^2}{K^2}}) \qquad (3.39)$$

$g_1$favorise les bords à fort contraste.

$g_2$favorise les régions étendues par rapport aux plus petites.

Le facteur de contraste *k* est calculé empiriquement comme le 70% de l'histogramme du gradient d'une version lissée des images originales, Il peut également être fixé à la main ou par un apprentissage.

L'algorithme KAZE suit les mêmes étapes que SIFT mais avec de petites différences à chaque étape. Il commence par construire l'espace à l'échelle non linéaire au moyen d'*Additive Operator Splitting* (AOS) qui est une solution (numérique) approximative de NDF. Parce qu'il n'existe pas une solution analytique pour résoudre l'équation aux dérivées partielles (EDP) de NDF. Ensuite, le détecteur KAZE est basé sur le déterminant normalisé de la matrice Hessian qui est calculée à plusieurs niveaux d'échelle. Les maxima de réponse du détecteur sont relevés sous forme de points caractéristiques à l'aide d'une fenêtre mobile. Enfin, l'orientation dominante de chaque point clé est calculée, puis la construction du descripteur de caractéristique d'une manière invariante en termes de rotation et d'échelle.

3.5 Descripteurs binaires

Un descripteur binaire est une représentation cohérente d'une région d'image ou d'un patch sous la forme d'une chaîne binaire résultant de comparaisons d'intensité de pixels à des emplacements prédéfinis dans le patch (Tan, Arshad,& Abdullah, 2019). Les descripteurs binaires ont un avantage particulier car ils accélèrent le temps d'extraction des caractéristiques et minimisent la capacité de stockage des caractéristiques extraites des images locales par rapport à d'autres descripteurs non binaire.

Le but des descripteurs binaires est de représenter les motifs locaux par un vecteur binaire qui peut être rapidement mis en correspondance en utilisant une distance spécifique par exemple la distance de Hamming.

3.5.1 BRIEF

Binary Robust Independent Elementary Features (BRIEF) est un descripteur décrit par une chaîne binaire (Calonder, Lepetit, Strecha, & Fua, (2010)). Le descripteur BRIEF est une description de chaîne de bits d'un patch d'image construit à partir d'un ensemble de tests d'intensité binaire. Il permet de décrire le point d'intérêt en utilisant un patch p de taille $S \times S$.

Tous les échantillons dans *p* sont des intensités de pixels lissées gaussiennes. La chaîne de bits $f_{nd}\,(p)$ de ce descripteur est le résultat de *nd* tests binaires concaténés

$$f_{nd}\,(P) = \sum_{t=1}^{nd} 2^{t-1}\ \tau\,(P;\ C, y_t) \qquad (3.40)$$

$$\tau(P; x_t, y_t) = \begin{cases} 1 & if\ \ P(x) < P(y) \\ 0 & sinon \end{cases} \qquad (3.41)$$

Où x et y sont les coordonnées des points d'échantillonnage.

Donc, cette approche permet d'utiliser la distance de Hamming qui permet de simplifier considérablement la tâche de correspondance. Les calculs de distance entre les caractéristiques binaires peuvent être effectués efficacement par l'opérateur XOR.

3.5.2 LBP

La technique du modèle *Local Binary Patterns* (LBP) est proposée par Ojala, Pietikäinen, & Harwood (1996). LBP fait partie des descripteurs utilisés pour extraire les caractéristiques locales. Il est devenu l'un des descripteurs d'analyse de texture les plus importants. LBP est une description de texture locale simple, efficace et robuste contre les variations de luminance. La description de la texture se fait par un modèle d'histogramme basé sur la représentation binaire calculée sur l'image complète ou bien une région d'une image.

Le principe du descripteur LBP est d'étudier la relation entre le pixel central, et ces pixels voisins ce qui permet de construire une description binaire autour du pixel central. Les approches basées sur LBP peuvent être utilisées dans différentes applications de vision par ordinateur où elles ont montré de meilleures performances dans la classification des textures. Dans les dernières tendances, ces modèles locaux sont de plus en plus adaptés à la reconnaissance faciale.

3.5.3 LDB

Local Difference Binary (LDB) est un descripteur binaire très efficace, robuste et distinctif qui a été proposé par (Yang, & Cheng. 2014). Ce descripteur calcule directement une chaîne

binaire en utilisant des tests de différence sur des bins spatiaux dans le patch. Le caractère distinctif et la robustesse de LDB sont obtenus en 3 étapes ;

Premièrement, LDB capture les motifs internes de chaque patch de l'image à travers un ensemble de tests binaires, chacun comparant l'intensité moyenne *I* et les gradients de premier ordre *dx* et *dy* d'une paire de grilles de l'image à l'intérieur du patch.

Deuxièmement, LDB utilise une stratégie de quadrillage pour capturer la structure à différentes granularités spatiales. Les grilles de niveau grossier peuvent supprimer le bruit à haute fréquence tandis que les grilles de niveau fin peuvent capturer des modèles locaux détaillés ce qui permet d'améliorer le caractère distinctif des caractéristiques.

Troisièmement, LDB sélectionne un sous-ensemble de bits hautement différents et distinctifs et les concatène pour former un descripteur LDB compact et unique.

3.5.4 LATCH

LATCH (Levi, & Hassner, 2015) est un descripteur binaire employé pour représenter des régions locales de l'image. Il s'est avéré que LATCH permet d'obtenir de meilleures performances par rapport à d'autres descripteurs binaires. Ce descripteur montre parfois des précisions de représentations beaucoup plus meilleures que celles de SIFT ou SURF.

Ce descripteur binaire utilise un ensemble de paires $S = (S_1, S_2, \ldots, S_N)$, où chaque S_i est une paire de deux emplacements de pixels définis dans un patch local. Pour chaque paire d'échantillonnage, le descripteur compare l'intensité à l'emplacement du premier pixel à celle du deuxième pixel. Si l'intensité est supérieure, il attribue 1 dans le descripteur final et 0 sinon. Ces comparaisons résultent en une chaîne binaire qui peut être comparée très efficacement en utilisant la distance de Hamming.

3.5.5 BRISK

Leutenegger et al. ont proposé l'algorithme *Binary Robust Invariant Scalable Key points (*BRISK*)* (Leutenegger, Chli, & Siegwart, 2011) comme alternative à l'extraction des

caractéristiques visuelles locales. Ces fonctionnalités sont très puissantes et sont similaires à celles de leurs prédécesseurs bien connus (par exemple, SIFT, SURF).

La première étape est la représentation de l'espace d'échelle : Cette représentation est obtenue en sous-échantillonnant de l'image originale en deux couches. L'une correspond aux octaves, tandis que l'autre correspond aux intra-octaves. La première octave est prise comme image d'entrée originale, et la première intra-octave est obtenue en sous-échantillonnant de l'image originale et en organisant les images résultantes en octaves et intra-octaves.

La deuxième étape est la détection de points clés : le principe de détection de points clés de l'algorithme BRISK est le même que celui de l'algorithme FAST. BRISK détecte les maxima ou minima locaux comme des points plus brillants ou plus sombre qu'un arc de pixels contigus autour de lui.

La troisième étape est la description du point clé : le descripteur BRISK consiste à construire un vecteur de 512 bits obtenus en comparant point à point l'intensité des échantillons prélevés au long d'un motif circulaire tourné autour du point clé. Si l'intensité du premier point est supérieure à celle du second, alors le bit correspondant est mis à 1; sinon, il est mis à 0.

3.5.6 A-KAZE

Accelerated KAZE (A-KAZE) est un détecteur et descripteur de caractéristiques, qui a été proposé par Alcantarilla, Nuevo, & Bartoli (2013). A-KAZE utilise des espaces à l'échelle non linéaire pour extraire les caractéristiques des images en utilisant le NDF. Un espace d'échelle non linéaire peut par exemple être obtenu en utilisant la technique efficace qui s'appelle AOS et la diffusion de conductance pour détecter les caractéristiques par le descripteur KAZE. Le NDF à son tour, rend le flou localement adaptable aux données de l'image et réduit ainsi le bruit tout en conservant les limites de l'objet. Cependant, les schémas AOS nécessitent la résolution d'un grand système d'équations linéaires pour obtenir une solution. En effet, cette solution demande beaucoup de calculs et prend beaucoup de temps.

Afin d'augmenter l'efficacité de calcul, il a été proposé de définir un espace d'échelle non linéaire à diffusion explicite rapide (en anglais *Fast Explicit Diffusion* -FED) en anglais, qui est intégré dans une forme pyramidale pour accélérer considérablement la détection des

caractéristiques dans l'espace à l'échelle non linéaire. En outre, les schémas FED sont faciles à mettre en œuvre et offrent une plus grande précision que les schémas AOS utilisés dans KAZE.

La diffusion d'une image se réalise d'une manière itérative afin de retrouver l'évolution filtrée mais préservée des bords de l'image originale, en choisissant soigneusement la fonction de conductivité. Un bon choix de la conductivité est le suivant:

$$c(x,y,t) = \left[1 + \left(\frac{\nabla L_\sigma(x,y,t)}{k}\right)^2\right]^{-1} \qquad (3.42)$$

Pour détecter l'ensemble des points clés, le déterminant de la matrice Hiessian est calculé pour chaque image filtrée L^i dans l'espace d'échelle non linéaire en utilisant l'équation (3.42)

$$L^i_{Hessian} = \sigma^2_{i,norm}\left(L^i_{xy}L^i_{yy} - L^i_{xy}L^i_{xy}\right) \qquad (3.43)$$

L'ensemble des points clés est trouvé en recherchant les maxima de la réponse du détecteur à l'échelle et à l'emplacement spatial. L'étape de description des caractéristiques est réalisée en utilisant une version modifiée du descripteur LDB et exploitant les informations de gradient et d'intensité de l'espace d'échelle non linéaire. Les caractéristiques d'AKAZE sont invariantes à l'échelle, à la rotation, à l'affine limitée et elles ont plus de caractère distinctif à différentes échelles en raison des espaces d'échelle non linéaires.

3.6 Quelques variantes du descripteur LBP

Ojala, Pietikäinen, & Maenpaa, (2002). ont proposé une approche multi-résolution basée sur des modèles binaires locaux pour la classification de la texture. Cette approche est invariante à l'échelle et à la rotation. C'est une approche très simple mais efficace. Une version étendue du LBP, appelée *BackGround Local Binary Patterns* (BGLBP) (Davarpanah, Khalid, Abdullah & Golchin, 2015), a été proposée pour gérer les changements d'éclairage des images, en particulier dans les environnements extérieurs. En fait, ces changements diminuent les performances d'extraction des caractéristiques en arrière-plan.

En ce qui concerne les variations de rotation, $LBP^{r_i}_{r,p}$ est une autre variante de LBP (Pietikäinen, Ojala & Xu, 2000). Elle a été obtenue en regroupant les mêmes motifs tournés du

descripteur LBP. En revanche, cette variante ne prend pas en compte les régions d'intérêt. Pour remédier à ce problème, un autre modèle est présenté par (Heikkilä, Pietikäinen & Schmid, 2006), ce modèle s'appelle *Center-Symmetric Local Binary Pattern* (CSLBP). Ce descripteur est défini par la combinaison des points clés du descripteur SIFT et du descripteur LBP. Il est construit d'une manière similaire à SIFT, sauf que ses caractéristiques individuelles sont différentes. Le CSLBP a été étendu pour créer un nouveau descripteur appelé *eXtended Center-Symmetric Local Binary Pattern* (XCSLBP) (Silva, Bouwmans & Frelicot, 2015). Ce descripteur est proposé pour la modélisation et la soustraction de l'arrière-plan dans les vidéos. Avec ce descripteur, une nouvelle stratégie de comparaison des pixels voisins est utilisée pour générer un histogramme court, tout en préservant la robustesse du visage aux changements d'éclairage.

D'autres descripteurs ont été proposés pour la modélisation de fond uniquement. Ils sont nommés respectivement : *Symmetri*c Scale Local Invariant Ternary Patterns (CSSILTP) (Wu, Liu, Luo, Su & Chen, 2014) et *Spatial Extended Center-Symmetric Local Binary Pattern* (SCSLBP) (Xue, Sun, & Song, 2010). Dans le premier descripteur, les propriétés spatio-couleur-texture sont utilisées pour détecter des personnes dans une vidéo. Le second est une extension du descripteur CS-LBP du domaine spatial au domaine spatio-temporel. Pour extraire les informations temporaires, un schéma amélioré d'estimation de la distribution temporelle a été associé au descripteur SCSLBP.

Un autre descripteur a également été proposé, mais cette fois, pour ne gérer que l'extraction de fond. Pour le définir, il est nécessaire de commencer par calculer le deuxième ordre du *Center-Symmetric Local Derivative Pattern* (CSLDP) (Xue, Song, Sun & Wu, (2011) ainsi que CSLBP. Ensuite, les histogrammes CSLBP et CSLDP sont concaténés pour former un autre descripteur hybride.

Un autre type de variantes LBP a été proposé. Dans ces variantes, des motifs binaires locaux sont remplacés par des motifs ternaires pour réduire la dimensionnalité. Le descripteur *Local Ternary Patterns* (LTP) est introduit par Tan, & Triggs, (2010). Il s'agit d'un descripteur invariable à la mise à l'échelle des images en niveaux de gris, mais il était plus résistant au bruit. Afin de soustraire l'arrière-plan et en particulier pour le déplacement des ombres douces, les chercheurs étendent le LTP en proposant un autre descripteur appelé *Scale Invariant Local*

Ternary Pattern (SI-LTP) (Liao, Zhao, Kellokumpu, Pietikainen & Li, 2010). Ces chercheurs ont montré que ce descripteur est efficace dans la gestion des variations d'éclairage.

Une autre idée était d'étudier les différences de pixels calculés localement : les Différences Angulaires (AD) et les Différences Radiales (RD) (Liu et al., 2016). Cette nouvelle variante du descripteur LBP a été utilisée pour capturer des informations de texture discriminantes par le développement de motifs dominants étiquetés. De plus, le modèle de *Local Directional Gradient Pattern* (LDGP) a été proposé (Chakraborty, Singh & Chakraborty, 2017). Ce descripteur étudie la relation entre un pixel de référence et quatre pixels voisins dans les quatre directions pour former un descripteur plus efficace. *Local Gradient HexaPattern* (LGHP) (Chakraborty et al., 2018) est un descripteur qui a été proposé pour extraire des informations discriminantes qui existent entre le pixel de référence et ses voisins ainsi que dans différentes directions dérivées, tout en identifiant la relation entre la référence pixel et ses voisins à différentes distances dans différentes directions des dérivations.

D'autres approches se sont concentrées sur la relation entre les caractéristiques LBP d'un pixel et celles de ses voisins. Parmi ces approches, il y a celles qui utilisent la distance de Hamming entre une paire de codes LBP situés au même endroit mais à deux échelles différentes pour mesurer les informations de la variation d'échelle. Ensuite, les filtres gaussiens sont appliqués pour générer des informations spatiales à l'échelle d'une image (Yuan et al., 2019). Il existe également une approche qui propose à comparer l'intensité d'un pixel et sa relation avec le pixel central dans une fenêtre 3 × 3. Ainsi, cette méthode permet de déterminer la relation d'un pixel central avec ses voisins adjacents (Banerjee, Bhunia, Bhattacharyya, Roy & Murala, 2018). Cette méthode génère un descripteur de texture basé sur la différence d'intensité de voisinage locale qui a été utilisée pour la recherche des images basées sur le contenu (CBIR). Le modèle *Local Zigzag Pattern* (LZP) (Roy et al., 2018) se caractérise par son efficacité à capturer des modèles de texture locale non uniformes. Ce descripteur a une forte relation angulaire entre deux pixels consécutifs par rapport au centre ainsi que deux pixels alternés par rapport à leur pixel intermédiaire.

Enfin, nous pouvons citer le descripteur nommé *Attractive-and-Repulsive Center-Symmetric Local Binary Patterns* (ARCS-LBP) (El merabet et al., 2019). Il s'agit d'un descripteur plus

robuste et stable qui a été utilisé pour préserver les caractéristiques de texture. Ce descripteur est formé de la combinaison de deux autres qui sont simples et plus utiles à la compréhension de l'image : *Attractive Center-Symmetric Local Binary Patterns* (ACS-LBP) et *Repulsive Center-Symmetric Local Binary Patterns* (RCS-LBP). Des expériences sur treize bases de données ont montré que l'utilisation du descripteur ARCS-LBP a très bien fonctionné par rapport à plusieurs autres descripteurs, y compris les caractéristiques d'apprentissage profond.

4 Représentation des caractéristiques de l'image

4.1 Bag-of-Words (BoW)

On considère que le monde peut être décrit au moyen d'un vocabulaire visuel (dictionnaire de "mots"). Dans sa version la plus simple, un document particulier est représenté par l'histogramme des occurrences des mots le composant : Dans un document donné, à chaque mot est affecté le nombre de fois où il apparaît dans le document (Sac). Un document est représenté par un vecteur de même taille que le vocabulaire visuel. La composante i indique le nombre d'occurrences du ième mot du dictionnaire dans le document.

4.2 Spatial Pyramid Matching (SPM)

L'approche d'appariement des pyramides spatiales en anglais est *Spatial Pyramid Matching* (SPM). Cette méthode a été proposée par Lazebnik, Schmid, & Ponce (2006). Dans leurs travaux, ils ont proposé d'étendre le BoF en pyramides spatiales afin de récupérer les informations spatiales dans l'image, en divisant l'image en sous-régions de plus en plus fines. En effet, ils ont construit un BoF qui représente l'image en concaténant le BoF des caractéristiques locales trouvées dans chaque sous-région (figure 8).

Dans cette méthode, les auteurs ont utilisé trois niveaux. Chaque niveau est obtenu par filtrage à l'aide du noyau gaussien. Par la suite, les auteurs ont calculé l'histogramme de chacun des trois niveaux. La distance entre deux histogrammes est donnée par la somme de la distance des trois niveaux en calculant les couples d'histogrammes correspondant aux mêmes niveaux de pyramide. Cette méthode est utilisée pour trouver une correspondance entre deux ensembles de vecteurs dans un espace d'entités. L'utilisation de la méthode SPM a montré qu'elle donne des résultats très satisfaisants par rapport à la BoF seule (Bosch, Zisserman, & Munoz, 2007).

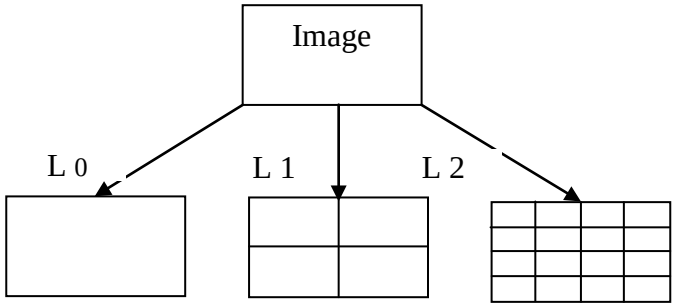


Figure 8. Représentation pyramidale spatiale des niveaux L0, L1 et L2 respectivement.

4.3 Sparse coding

Le codage parcimonieux ou *Sparse Coding* (SC) en anglais comme une première idée en vision par ordinateur est introduit par Olshausen et Field (Olshausen, & Field, 1996; Olshausen, & Field, 1997). SC a fait l'objet de nombreuses recherches ces dernières années. En SC, un vecteur de caractéristiques *Y* est représenté par la combinaison d'un ensemble minimal d'atomes provenant d'un dictionnaire *D* prédéfini d'atomes. Chaque colonne de *D* est un vecteur prototype (atome).

Étant donné une matrice de données d'entrée, le SC vise à trouver un ensemble de vecteurs de base (c'est-à-dire un dictionnaire) qui capturent la sémantique de haut niveau et les coordonnées par rapport au dictionnaire. Le SC présente plusieurs avantages pour la représentation des données. Il permet de représenter chaque point de données par une combinaison linéaire d'un petit nombre de vecteurs de base. Cette représentation offre plus de flexibilité dans la représentation du signal et plus d'efficacité dans des tâches telles que la compression de données et l'extraction de signal.

Le SC a été utilisé dans de nombreuses applications, telles que la reconnaissance faciale (Wright, Yang, Ganesh, Sastry, & Ma, 2009) la restauration d'images (Donoho, Elad, & Temlyakov, 2006), et la classification des images (Mairal, Bach, Ponce, Sapiro, & Zisserman, 2009; Raina, Battle, Lee, Packer,& Ng, 2007).

5 Conclusion

La détection et la description des caractéristiques sont des composants essentiels de diverses applications en vision par ordinateur. Au cours des dernières décennies, elles ont tiré une

attention considérable. L'extraction des caractéristiques locales est considérée comme un meilleur outil pour représenter des images. Cette représentation est pour la reconnaissance des objets, la correspondance entre les images en utilisant les points clés ainsi que pour classifier les objets spécifiques dans des catégories (classes). Cette représentation est proposée afin de trouve des caractéristiques invariantes à la mise à l'échelle et à la rotation de l'image, et qui sont également robustes aux changements d'éclairage. Une description est calculée pour chaque caractéristique locale en utilisant le voisinage local autour d'elle, puis elle est considérée comme un identifiant unique pour cette caractéristique.

Ce chapitre présente les notations de base et les concepts mathématiques pour détecter et décrire les caractéristiques de l'image. Puis, il décrit les propriétés des détecteurs de caractéristiques existants. Divers algorithmes existants pour détecter les points d'intérêt sont brièvement discutés. Parmi des algorithmes de description les plus fréquemment utilisés on peut citer HOG, SIFT, SURF, LBP.

IV. Approches de l'intelligence artificielle pour la classification des images

1 Introduction

Le développement dans le domaine de la technologie s'est amélioré au fil des ans. Aujourd'hui, nous entendons parler des termes de technologie comme l'intelligence artificielle, l'apprentissage automatique et l'apprentissage profond. Nous confondons souvent ces termes et les définissons de manière similaire. Mais ce n'est pas une définition précise car ces termes sont différents les uns des autres. Pour illustrer la relation entre ces termes, il est possible d'utiliser la relation d'appartenance montrée dans la figure 9.

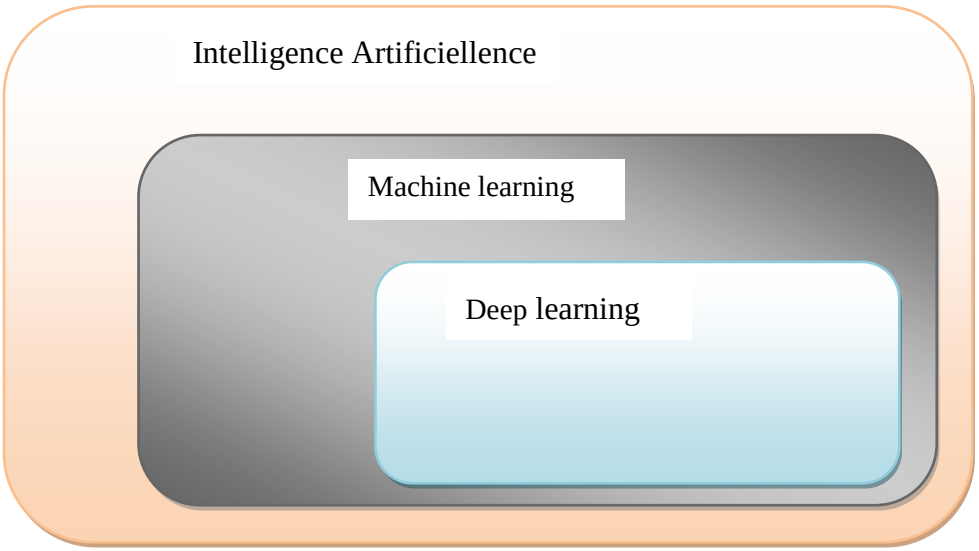


Figure 9. Concepts liés à l'intelligence artificielle.

• Intelligence artificielle (IA): le rectangle le plus large est l'idée à l'IA qui est apparue en premier dans ce domaine

• Apprentissage automatique ou *Machine Learning* (ML) en anglais : au milieu, il a prospéré plus tard après l'IA

• Apprentissage profond, aussi appelé *Deep Learning* (DL) en anglais : le plus petit rectangle est actuellement une extension de l'IA

Dans ce chapitre, nous allons discuter de la différence entre ces trois termes Intelligence artificielle, Apprentissage automatique et Apprentissage profond.

2 Intelligence artificielle

Intelligence artificielle (IA), comme son nom l'indique, l'intelligence est créée par l'homme. Un facteur majeur à considérer lors de la création d'un système intelligent est de percevoir comment ce système intelligent gère les environnements changeants et de s'y retrouver avec succès. Il s'est construit comme des machines complexes utilisant des propriétés informatiques et effectuant diverses actions pour imiter le comportement humain.

L'intelligence artificielle est un grand domaine dans lequel les ordinateurs sont formés pour montrer un comportement intelligent. L'intelligence artificielle pense et réagit de la même manière que les humains, telle qu'elle est conçue de manière similaire au cerveau humain. L'intelligence artificielle est la technologie pour l'avenir de l'humanité qui rend leur vie meilleure qu'avant.

Le rôle de ces technologies est similaire à celui des humains. Ces technologies préfèrent donc la meilleure solution pour les tâches que nous ne pouvons pas effectuer. Cela réduit efficacement le travail humain et peut les aider à créer de multiples solutions. Parmi des meilleurs cas d'application de l'intelligence artificielle sont la reconnaissance faciale, la détection et la classification des objets etc. Donc, l'intelligence artificielle est la technologie de l'avenir de l'humanité et rend leur vie meilleure qu'avant. Cependant, établir et utiliser l'intelligence artificielle d'une manière générale dans nos vies n'est pas possible jusqu'à présent car il existe de nombreuses caractéristiques du cerveau humain qui sont incapables de décrire.

3 Machine Learning pour la classification des images

L'apprentissage automatique est une technique qui fait partie de l'intelligence artificielle pour former des modèles complexes. Ces modèles peuvent faire fonctionner l'ordinateur ou le système de manière indépendante sans intervention humaine. L'apprentissage automatique

utilise des données pour alimenter un algorithme qui peut comprendre la relation entre l'entrée et la sortie. Lorsque la machine a terminé l'apprentissage, elle peut prédire la valeur ou la classe du nouveau point de données. L'une des principales idées derrière l'apprentissage automatique est que l'ordinateur peut être formé pour automatiser des tâches qui seraient impossible pour un être humain. Donc, c'est un domaine de recherche en informatique qui traite des méthodes d'identification et de mise en œuvre de systèmes et d'algorithmes par lesquels un ordinateur peut apprendre sur la base des exemples donnés en entrée. Cet apprentissage est jusqu'à présent le meilleur outil pour analyser, identifier et comprendre un modèle de données. Il peut prendre des décisions avec une intervention minimale humaine.

Le défi de l'apprentissage automatique est de permettre à un ordinateur d'apprendre à reconnaître automatiquement des modèles complexes et à prendre des décisions aussi intelligentes que possible. L'ensemble du processus d'apprentissage nécessite un ensemble de données de formation et un ensemble de données de test :

Ensemble de formation : il s'agit de la base de connaissances utilisée pour former l'algorithme d'apprentissage automatique. Au cours de cette phase, les paramètres du modèle d'apprentissage automatique peuvent être ajustés en fonction des performances obtenues.

Ensemble de test : utilisé uniquement pour évaluer les performances du modèle sur des données invisibles. La théorie de l'apprentissage utilise des outils mathématiques dérivés de la théorie des probabilités et de la théorie de l'information.

Il y a essentiellement trois paradigmes d'apprentissage qui seront brièvement discutés dans les sections suivantes (figure 10):

- Apprentissage supervisé
- Apprentissage non supervisé
- Apprentissage par renforcement.

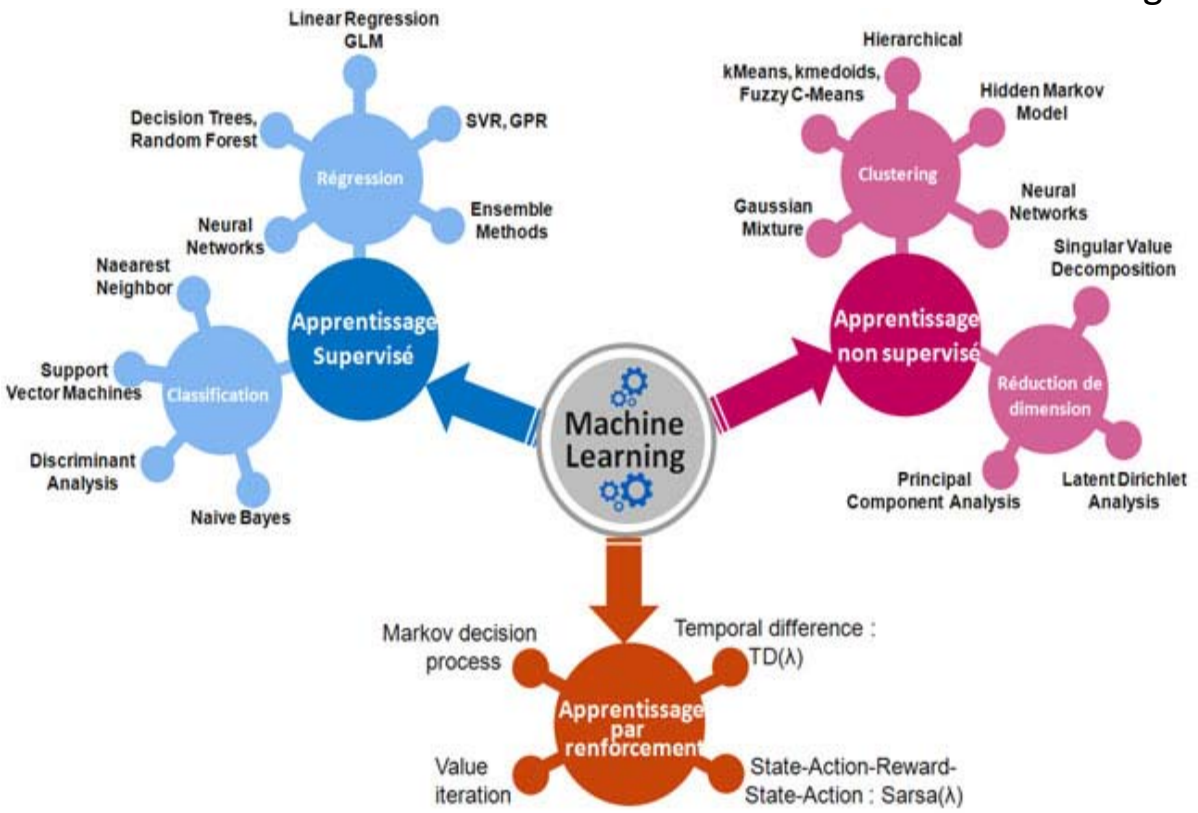


Figure 10. Principaux algorithmes d'apprentissage automatique.

3.1 Apprentissage supervisé

L'apprentissage supervisé est un type d'apprentissage automatique plus simple et très connu. Il est basé sur un certain nombre d'exemples pré-classifiés, dans lesquels chacun des intrants utilisés comme exemples d'apprentissage devrait appartenir à une classe (catégorie) connue a priori. Dans ce cas, la question cruciale est comment généraliser le problème de la généralisation de la prédiction pour tous les exemples.

Après d'apprentissage d'un échantillon d'exemples, le système devrait produire un modèle qui devrait bien fonctionner pour toutes les entrées possibles. L'ensemble d'exemples se compose de données étiquetées, c'est-à-dire d'objets et de leurs classes associées. Cet ensemble d'exemples étiquetés constitue donc l'ensemble d'apprentissage. Le système modifie ensuite ses paramètres internes afin de minimiser cette fonction d'erreur. Par la suite, la qualité du modèle est évaluée en utilisant un deuxième ensemble d'exemples étiquetés (l'ensemble de test) afin d'évaluer le pourcentage d'exemples correctement classés et le pourcentage d'exemples mal classés.

Le contexte d'apprentissage supervisé comprend la fonction de prédiction est appelée classificateur ou *classifier* en anglais. On parle dans ce cas d'un problème de classification, qui revient à attribuer une étiquette à chaque entrée lorsque l'ensemble des valeurs de sortie est un nombre entier. Mais aussi l'apprentissage de fonctions qui prédisent des valeurs numériques lorsque la sortie que l'on cherche à estimer est une valeur continue de type réel. Cette fonction de prédiction est appelée la fonction de régression ou bien *regression function* en anglais. Il existe donc plusieurs algorithmes d'apprentissage supervisé qui ont été développés pour la classification et la régression.

3.1.1 Classification

Un classificateur est un mécanisme à n entrées représentant les n entités $e_1, e_2, \ldots, e_n$ calculé à partir d'un objet à classer, et une (1) sortie. Un classificateur de classe S générera l'un des symboles $w_1, w_2, \ldots, w_S$ comme sortie, et l'utilisateur interprète cette sortie comme une décision concernant la classe de l'objet traité. Les symboles générés w_s sont les identificateurs de classes.

Les algorithmes de classification tels que SVM, K-NN, Naive bayes sont des algorithmes pour classifier des données. Ces données peuvent être des mots, des couleurs, des sons, etc. Dans un contexte de classification d'images, ces algorithmes peuvent être utilisés comme classificateurs basés sur des informations extraites des pixels de ces images.

3.1.1.1 Support Vector Machines

Support Vector Machine (SVM) est un type d'algorithmes de classification (Boser, Guyon, & Vapnik, 1992 ; Cortes, Vapnik, 1995 ; Vapnik, 2000; Burges, 1998). SVM peut être utilisé pour résoudre des problèmes de discrimination. Autrement dit, il peut décider à quelle classe appartient un échantillon. La technique de SVM consiste à créer un hyperplan dans un espace de grande dimension afin de séparer des données des classes. La séparation de données est réalisée par l'hyperplan qui a la plus grande classification des points de données des classes les plus proches. Ces points de données sont appelés la marge. L'erreur de généralisation du classificateur SVM dépend de la taille de la marge Boser, Guyon, et Vapnik (1992). La marge est la distance entre la frontière de séparation des points les plus proches.

L'algorithme d'apprentissage SVM construit un modèle sur la base de la marge fonctionnelle. La marge fonctionnelle fait un classificateur linéaire binaire. Le modèle appris est utilisé pour la classification linéaire et non linéaire. Le SVM peut être facilement

transformé en apprenants non linéaires. La classification non linéaire est effectuée à l'aide d'une fonction basée sur le noyau pour mapper l'entrée dans un espace d'entités de grande dimension (Cristianini, & Shawe-Taylor, 2000). Dans cet algorithme, la prise en charge des points de données de formation appartient chacune à deux classes. Et son objectif est de décider quelle classe est responsable d'un nouveau point de données.

3.1.1.2 K-Naerest Neighbor

Le principe de l'algorithme *K-Naerest Neighbor* (K-NN) (Cover, Hart, 1967 ; Duda, & Hart, 1973; Fukunaga, &Hostetler,1975) est de trouver un nombre prédéfini d'échantillons d'apprentissage les plus proches en terme de distance par rapport à la nouvelle entité (entrée), où chaque point de données est caractérisé par un ensemble de variables. C'est à dire, chaque point est tracé dans un espace de grande dimension, où chaque axe de l'espace correspond à une variable individuelle. Autrement dit, k-plus proche voisin consiste à récupérer les entités voisines les plus proches par rapport à la nouvelle entité et à affecter à une classe.

K-NN est sans doute le plus simple de tous les algorithmes de classification supervisée, cet algorithme continue de fonctionner assez bien pour des grandes tailles d'apprentissage. Il ne nécessite que le choix de K, le nombre de voisins à considérer lors de la classification. Le nombre K est généralement choisi comme la racine carrée de *N*, le nombre total de points dans l'ensemble de données d'entraînement. (Ainsi, si N est 400, K = 20).

Lorsque nous avons un nouveau point de données (test), nous voulons trouver les K voisins les plus proches qui sont les plus proches (c'est-à-dire les plus «similaires») en utilisant une distance. En général, la distance peut être n'importe quelle mesure métrique : la distance euclidienne standard est le choix le plus courant.

3.1.1.3 Naive Bayes

L'algorithme *Naive Bayes* (NB) (Fukunaga, 1990) est un algorithme d'apprentissage supervisé basé sur l'application du théorème de Bayes avec l'hypothèse d'indépendance conditionnelle entre chaque paire de caractéristiques pour calculer les probabilités d'appartenance à une classe.

Le théorème de Bayes est défini par la relation suivante :

$$P(y|x_1, \dots, x_n) = \frac{P(y)P(x_1, \dots, x_n|y)}{P(x_1, \dots, x_n)} \qquad (4.1)$$

En 2004, l'analyse du problème de classification bayésienne a montré qu'il existe des raisons théoriques pour l'efficacité, apparemment non raisonnable, des classificateurs bayésiens naïfs (Zhang, 2004).

L'algorithme *Naïve Bayes* suppose que y est une variable de classe et X est un vecteur d'entités dépendant (de taille n) où*:* $X = x_1, x_2 \dots, x_n$.Bien que, l'algorithme Naïve Bayes est simple dans sa conception et son développement. L'utilisation de cet algorithme a donné des résultats très encourageants, et plus utilisés dans le domaine de la classification des images.

3.1.2 Régression

Dans l'apprentissage automatique, les algorithmes de régression tentent d'estimer la fonction (f) des variables d'entrée (x) aux variables de sortie numériques ou continues (y). Par exemple, lorsque vous disposez d'un ensemble de données sur les maisons et que vous êtes invité à prédire leurs prix, il s'agit d'une tâche de régression car le prix sera une sortie (valeur) continue. Les exemples d'algorithmes de régression courants incluent *Linear Regression*, *Decision Tree, Random Forest* et *Support Vector Regression*.

3.1.2.1 Linear Regression

La régression linéaire ou *Linear Regression* (LR) tente de modéliser la relation entre deux variables en ajustant une équation linéaire aux données observées. Une variable est considérée comme une variable explicative et l'autre est considérée comme une variable dépendante (Yan, 2009). Par exemple, un modélisateur peut vouloir relier les poids des individus à leurs hauteurs à l'aide d'un modèle de régression linéaire. Une droite de régression linéaire a une équation de la forme Y = bX + a, où X est la variable explicative et Y est la variable dépendante. La pente de la ligne est b et a est l'ordonnée à l'origine (a est la valeur de y lorsque x = 0).

3.1.2.2 Decision Tree

L'apprentissage de l'arbre de décision aussi appelé *Decision Tree* (DT) en anglais, est une technique d'apprentissage automatique supervisé pour induire un arbre de décision à partir des données d'apprentissage. Un arbre de décision est un arbre binaire (arbre dans lequel chaque nœud a deux nœuds enfants). Dans les structures arborescentes, les feuilles représentent des classes (également appelées étiquettes de classe) (Baum, & Petrie, 1966).

Il peut être utilisé soit pour la classification ou bien pour la régression (Breiman, 2001) Pour la classification, chaque feuille d'arbre est marquée par une étiquette de classe ;

plusieurs feuilles peuvent avoir la même étiquette. Pour la régression, une valeur est également affectée à chaque feuille de l'arbre, de sorte que la fonction d'approximation est constante.

Pour atteindre un nœud feuille afin d'obtenir une réponse du vecteur d'entité en entrée, la procédure de prédiction commence par le nœud racine. À partir de chaque nœud non feuille, la procédure va vers la gauche (sélectionne le nœud enfant gauche comme nœud observé suivant) ou vers la droite en fonction de la valeur d'une certaine variable dont l'index est stocké dans le nœud observé.

3.1.2.3 Random Forest

Les forets aléatoires ou bien *Random Forest* (RF) en anglais, sont parmi les algorithmes qui sont utilisés dans la classification supervisée. Cet algorithme est proposé par Breiman (2001) au début des années 2000. Il s'agit de l'un des algorithmes qui restent efficaces lorsqu'il est appliqué à de grandes quantités de données. Il combine de manière complexe plusieurs mécanismes difficiles à appréhender comme le *Bagging* (Breiman, Friedman, Olshen, Stone, 1984).

Les forêts aléatoires combinent des prédicteurs ou des estimateurs des arbres, donnant naissance à ce que l'on appelle maintenant les arbres de décision. Plus généralement, ce sont des algorithmes établis dont leur principe général est de construire une collection de prédicteurs afin d'agréger ensuite toutes leurs prédictions.

En classification, l'agrégation consiste par exemple à faire un vote majoritaire parmi les étiquettes de classe fournis par les prédicteurs. Concernant la régression, la cible de régression prédite d'un échantillon d'entrée est calculée comme la cible de régression prédite moyenne des arbres dans la forêt.

3.1.2.4 Support Vector Regression

Régression Vectorielle de support appelé en anglais *Support Vector Regression* (SVR). L'algorithme de classification des vecteurs de support peut être étendu pour résoudre les problèmes de régression. Cet algorithme est appelé Régression Vectorielle de support (Drucker, Burges, Kaufman, Smola, & Vapnik, 1996 ; Vapnik, 1998). Comme son nom l'indique, le SVR est un algorithme de régression, nous pouvons donc utiliser SVR pour travailler sur des valeurs continues au lieu de classification.

L'objectif principal de SVR est de trouver des modèles dont les prédictions ont un écart maximal par rapport aux valeurs de la fonction de régression pour les données d'apprentissage (Vapnik, 1998). Bien que, SVR permet de définir un Hyper Plan qui est une ligne de séparation entre les classes de données. La ligne de séparation nous aidera à prédire la valeur continue ou la valeur cible.

3.2 Apprentissage non supervisé

Dans l'apprentissage non supervisé, un ensemble d'entrés est fourni au système de classification pendant la phase d'apprentissage contrairement au cas de l'apprentissage supervisé. Cet ensemble n'est pas étiqueté par la classe d'appartenance correspondante..

L'objectif principal de l'apprentissage non supervisé est de modéliser la structure des entrées de données qui sont observées pour avoir plus d'informations sur la distribution des données sous-jacentes. L'apprentissage non supervisé est utilisé particulièrement dans les problèmes de *clustering*, dans lesquels étant donné une collection d'objets, nous voulons être capables de comprendre et de montrer leurs relations.

Une approche standard utilisée dans ce type d'apprentissage consiste à définir une mesure de similitude entre deux objets, puis à rechercher tout cluster d'objets plus similaires les uns aux autres, par rapport aux objets des autres clusters. Cet apprentissage peut être utilisé pour le regroupement, l'estimation de la densité, la réduction de dimension etc.

3.2.1 Clustering

Clustering est une classification non supervisée des objets. C'est une technique courante d'analyse de données, utilisée dans de nombreux domaines, notamment l'apprentissage automatique, la reconnaissance de formes, l'analyse d'images et la bioinformatique.

Les algorithmes de *clustering* partitionnent les données en un certain nombre de clusters (catégories, sous-ensembles ou groupes). C'est un processus de regroupement des objets similaires en différents groupes, ou plus précisément, le partitionnement d'un ensemble de données en sous-ensembles selon une mesure de distance définie (distance Euclidienne par exemple).

3.2.1.1 K-means Clustering

Généralement, *K-means* est considéré comme l'une des techniques les plus efficaces de segmentation non supervisée (Jain, Dubes, 1988; Kaufman, & Rousseeuw, 1990). *K-means*

partitionne un certain nombre de points en k-groupes dans lesquels chaque point appartient au groupe qui a la moyenne la plus proche. Les étapes de *K-means* sont les suivantes :

1) Calculer la valeur moyenne de chaque groupe.

2) Affecter chaque point au groupe le plus proche en calculant la distance de chaque point par rapport à la valeur moyenne du groupe correspondant.

L'algorithme de *K-means* est un processus itératif pour calculer la valeur moyenne de chaque groupe et pour calculer la distance de chaque point par rapport au groupe le plus proche. Ce processus itératif est répété jusqu'à ce qu'aucune réduction supplémentaire ne puisse être obtenue dans la somme des erreurs carrées dans chaque groupe.

3.2.1.2 Hierarchical Ascendant Clustering

La seconde famille des algorithmes de *clustering* comprend les méthodes hiérarchiques top-down telles que le *Hierarchical Ascendant Clustering* (HAC) (Hastie, Tibshirani, Friedman, Franklin, 2001). HAC consiste à construire des groupes en partitionnant des individus (objets) de façon ascendante. Ce processus conduit à une bonne visualisation des résultats. Cependant, il a une complexité non linéaire. Cet algorithme consiste à agglomérer les groupes similaires afin d'avoir à la fin de cet algorithme un groupe contenant tous les individus ou bien tous les objets X_i $(i = 1, .., N)$.

Considérons $\rho^D = \{C_1, \ldots, C_D\}$ l'ensemble des groupes. Si $D = N$, $C_1 = x_1, \ldots, C_N = x_N$. Par la suite, à travers toutes les étapes du regroupement, on passera d'une partition ρ^D à une partition ρ^{D-1}.

Le résultat généré est décrit par un arbre de *clustering* hiérarchique qui s'appelle le dendrogramme. Les nœuds représentent les fusions successives entre les partitions. La hauteur des nœuds représente la valeur de la distance entre deux objets qui donne une signification concrète au niveau des nœuds appelée l'indexe. Ce dernier est généralement défini par les valeurs des distances (ou dissimilarité) pour chaque étape d'agrégation.

3.2.1.3 Hidden Markov Model

Le modèle de Markov caché ou *Hidden Markov Model* (HMM) en anglais, a été proposé pour la première fois par Baum, & Petrie (1966). HMM référe au nom du mathématicien russe Andrey Andreyevich Markov, qui a étudié et développé au début des années 1970 une grande partie de la théorie statistique. HMM est un modèle statistique qui utilise le processus

de Markov et contient des paramètres cachés et inconnus. Ce modèle stochastique est composé de deux suites de variables aléatoires, la première est cachée et la deuxième est observable qui permet de capturer des informations cachées à partir de symboles séquentiels observables.

Une chaîne de Markov est utile lorsque nous voulons calculer une probabilité pour une séquence d'événements observables. Dans de nombreux cas, les événements qui nous intéressent sont cachés et nous ne les observons pas directement.HMM a été utilisé pour la première fois dans la reconnaissance vocale et il a été appliqué avec succès à l'analyse de séquences biologiques à la fin des années 1980. De nombreuses techniques d'apprentissage automatique basées sur de HMM ont été appliquées avec succès à des problèmes tels que la reconnaissance optique de caractères, la reconnaissance vocale etc. HMM sont devenues un outil fondamental en bioinformatique grasse à leur simplicité conceptuelle, leur base statistique solide. HMM sont adaptées pour répondre à divers problèmes en classification des objets.

3.2.1.4 Gaussian Mixture Models

Les modèles *Gaussian Mixture Models* (GMM) font parti des approches de *clustering* qui utilisent des modèles probabilistes pour distribuer les points dans différents clusters (McLachlan, Basford, 1988). GMM comme son nom l'indique, est un mélange de plusieurs distributions gaussiennes. Ces Modèles permettent de mesurer l'incertitude ou la probabilité qui nous indique à quel point de données est associé à un groupe spécifique. GMM suppose qu'il existe un certain nombre de distributions gaussiennes, et chacune de ces distributions représente un cluster. Par conséquent, un modèle de mélange gaussien a tendance à regrouper les points de données appartenant à une seule distribution.

GMM peut être utilisé pour regrouper des données non étiquetées de la même manière que l'algorithme k- *means*. Une caractéristique importante de K-*means* est qu'il s'agit d'une méthode de *clustering* puissante, ce qui signifie qu'il associera chaque point à un et un seul cluster. Une limite à cette approche est qu'il n'y a pas de mesure d'incertitude ou de probabilité qui nous indique à quel point un point de données est associé à un cluster spécifique.

Un mélange gaussien est une fonction qui est composée de plusieurs gaussiens, chacun identifié par $k \in \{1,\ldots, K\}$, où K est le nombre de groupe de l'ensemble de données. Chaque k gaussien dans le mélange comprend les paramètres suivants :

- Un μ moyen qui définit son centre.
- Une covariance Σ qui définit sa largeur.
- Une probabilité de mélange π qui définit la taille de la fonction gaussienne.

3.2.2 Réduction de dimension

Dans les problèmes de l'apprentissage automatique, il y a souvent trop de facteurs sur la base de données sur la quelle la classification est effectuée. Ces facteurs sont essentiellement des variables appelées caractéristiques. Plus le nombre de caractéristiques est élevé, plus il est difficile de visualiser l'ensemble d'entraînement. Parfois, la plupart de ces caractéristiques sont corrélées et donc redondantes. C'est là que les algorithmes de réduction de dimensionnalité entrent en jeu.

La réduction de la dimensionnalité joue un rôle important dans les performances de classification. L'utilisation de la réduction de dimensionnalité est pour prendre des données de dimension supérieure et les représenter dans une dimension inférieure, en obtenant un ensemble de variables principales.

La réduction de la dimensionnalité est une technique puissante qui est largement utilisée dans l'analyse des données pour aider à sélectionner les bonnes caractéristiques, former efficacement les modèles utilisés et visualiser les données.

3.2.2.1 Principal Componet Analysis

Principal Component Analysis (PCA) est une technique importante à comprendre dans les domaines de la statistique et de la science des données (Wold, Esbensen, & Geladi, 1987). PCA est un moyen utilisé pour afficher des modèles dans des données multivariées. Il vise à afficher graphiquement les positions relatives des points de données dans moins de dimensions tout en conservant des informations pertinentes et à explorer les relations entre les variables dépendantes.

PCA a été utilisé pour réduire la dimensionnalité d'un ensemble de données en expliquant la corrélation entre de nombreuses variables en termes d'un plus petit nombre de vecteurs (composantes principales), sans perdre beaucoup d'informations (Abdi & Williams, 2010). La valeur propre donne une mesure de la signification du vecteur : les vecteurs et les valeurs propres les plus élevées sont les plus significatifs.

De nombreux objectifs de PCA concernent la recherche de relations entre les objets. On peut être intéressé, par exemple, à trouver des classes d'objets similaires. L'appartenance à la classe peut être connue à l'avance, mais elle peut également être trouvée en explorant les données disponibles.

3.2.2.2 Feature Selection

De nombreuses méthodes de sélection des caractéristiques sont disponibles dans la littérature en raison de la disponibilité de données avec des centaines de variables conduisant à des données de très grandes dimensions. Les algorithmes de sélection des caractéristiques ou *Feature selection* en anglais (Cai, Luo, Wang, & Yang, 2018), permettent d'améliorer des performances de prédiction ainsi que de réduire le temps de calcul et de mieux comprendre la corrélation entres les caractéristiques dans les applications d'apprentissage automatique ou de reconnaissance de formes.

Les algorithmes les plus célèbres de la sélection des caractéristiques utilisées dans la classe de *clustering* qui relèvent du *filtre* et du *wrapper*. Les algorithmes de modèle de filtre n'utilisent aucun algorithme de *clustering* pour tester la qualité des caractéristiques (Dy, 2008).). Ils évaluent le score de chaque caractéristique selon certains critères. Ensuite, ils sélectionnent les caractéristiques qui ont le score le plus élevé. Le modèle *wrapper* utilise un algorithme de *clustering* pour évaluer la qualité des caractéristiques sélectionnées (Dy, 2012). Il commence premièrement (1) par trouver un sous-ensemble de caractéristiques. Ensuite, (2) il évalue la qualité du clustering en utilisant le sous-ensemble sélectionné. Enfin, il répète (1) et (2) jusqu'à ce que la qualité souhaitée soit trouvée.

3.2.2.3 Linear Discriminant Analysis

Linear Discriminant Analysis (LDA) est un algorithme utilisé dans les statistiques, l'apprentissage automatique et la reconnaissance de formes pour trouver une combinaison linéaire de caractéristiques qui caractérise ou sépare deux ou plusieurs classes d'objets (Mokeev, Mokeev, 2015). LDA est une technique de réduction de dimensionnalité. Comme son nom l'indique, les techniques de réduction de la dimensionnalité réduisent le nombre de dimensions (c'est-à-dire les variables) dans un ensemble de données tout en conservant autant d'informations que possible.

Cette technique est utilisée lorsqu'il y a une ou plusieurs variables indépendantes. Le discriminant linéaire effectue un test de différence multivarié entre les groupes. Il est également utile pour déterminer le nombre minimum de dimensions nécessaires pour décrire

ces différences. LDA est souvent utilisé comme technique de réduction de dimensionnalité dans l'étape de prétraitement pour les applications de classification de modèles et d'apprentissage automatique. L'objectif est de projeter un ensemble de données sur un espace de dimension inférieure avec une bonne séparabilité de classe afin de réduire les coûts de calcul.

3.3 Apprentissage par renforcement

L'apprentissage par renforcement (en anglais : *Reinforcement Learning* (RL)) est une approche de l'intelligence artificielle qui met l'accent sur l'apprentissage du système à travers ses interactions avec l'environnement (Sutton, 1988 ; van Otterlo, Wiering, 2012; Francois-Lavet, Vincent ; et al. 2018). A l'aide de cet apprentissage, le système adapte ses paramètres en fonction des commentaires reçus de l'environnement. Ensuite, ce système va fourni des commentaires sur les décisions prises. Par exemple, un système qui modélise un joueur d'échecs qui utilise le résultat des étapes précédentes pour améliorer ses performances est un système qui apprend par renforcement.

La recherche actuelle sur l'apprentissage par renforcement est hautement interdisciplinaire et comprend des chercheurs spécialisés dans les algorithmes génétiques, les réseaux de neurones, la psychologie et l'ingénierie du contrôle. Deux types principaux de signaux de récompense sont :

- **Le signal de récompense négatif** : pénalise l'exécution de certaines activités et demande instamment de corriger l'algorithme pour ne plus recevoir de pénalités.
- **Le signal de récompense positif** : encourage la poursuite des performances d'une séquence d'action particulière.

Pour un ensemble, le système essaie de maximiser les récompenses positives et de minimiser les négatifs. Cependant, la nature des informations peut modifier la fonction du signal de récompense. Ainsi, les signaux de récompense peuvent être principalement classés en fonction des exigences de l'opération. Parmi les algorithmes d'apprentissage par renforcement, on compte :

- Recherche d'arbres Monte-Carlo (SCTM).
- Q-Learning.
- Agents d'acteurs critiques asynchrones (A3C).
- Différence temporelle (TD).

Parmi des applications de l'apprentissage par renforcement qu'on peut citer :

- Apprentissage automatique et traitement des données.
- Robotique pour l'automatisation industrielle.
- Planification de la stratégie commerciale.
- Contrôle des avions et contrôle du mouvement du robot.

Les principales raisons d'utiliser l'apprentissage par renforcement sont :

- Aider à trouver quelle situation nécessite une action.
- Aider à découvrir quelle action rapporte la récompense la plus élevée sur la plus longue période.

4 Réseaux de Neurones et Deep Learning

Les Réseaux de Neurones Artificiels aussi appelé *Artificial Neural Networks* (ANN) en anglais, sont une classe d'algorithmes d'apprentissage automatique qui apprennent sur des données et se spécialisent dans la reconnaissance des formes. Les ANN sont inspirés de la structure et la fonction du cerveau (Schwenker, Abbas, Gayar, Trentin, (eds), 2016). Le premier modèle d'un neurone artificiel a été proposé par McCulloch et Pitts (1943) en termes de modèle de calcul de l'activité nerveuse (Anderson, & Rosenfeld, 1988).

Dans les années 1950, l'algorithme de l'architecture *Perceptron* a été publié par (Rosenblatt, 1958; Rosenblatt, 1963). Ce modèle pouvait automatiquement apprendre les poids nécessaires pour classer une entrée (aucune intervention humaine requise). Un exemple de l'architecture *Perceptron* peut être vu dans la figure 11 qui présente un exemple de l'architecture d'un réseau *Perceptron* simple. Ce réseau accepte un certain nombre d'entrées, et cet exemple calcule une somme pondérée et applique une fonction pas à pas pour obtenir la prédiction finale. En effet, cette procédure d'apprentissage automatique est principalement constituée de la Descente du gradient stochastique ou *Stochastic Gradient Descent* (SGD) en anglais. Cette procédure est encore utilisée aujourd'hui pour entraîner les réseaux neuronaux profonds.

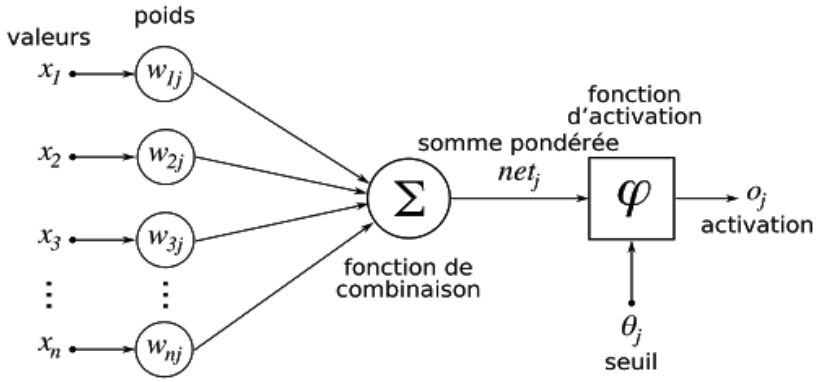


Figure 11. Exemple de l'architecture d'un réseau *Perceptron* simple.

Un réseau de neurones peut apprendre automatiquement, et sans connaissances prédéfinies explicitement codées par les programmeurs.

L'apprentissage se déroule en deux phases.

- La première phase consiste à créer une architecture du modèle.
- La deuxième phase vise à améliorer le modèle en utilisant une technique qui est appelée rétropropagation ou *Backpropagation* en anglais. C'est la clé de la façon dont un réseau de neurones apprend une tâche particulière.

Le réseau neuronal répète ces deux phases plusieurs fois jusqu'à ce qu'il atteigne un niveau de précision acceptable. La répétition de cette double phase est appelée une itération. Les réseaux de neurones peuvent être divisés en deux classes :

La première classe regroupe les réseaux de neurones non profond : Le réseau de neurones non profond ont une seule couche cachée entre la couche l'entrée et la sortie. Pour une représentation graphique de cette relation, voir la figure 12.

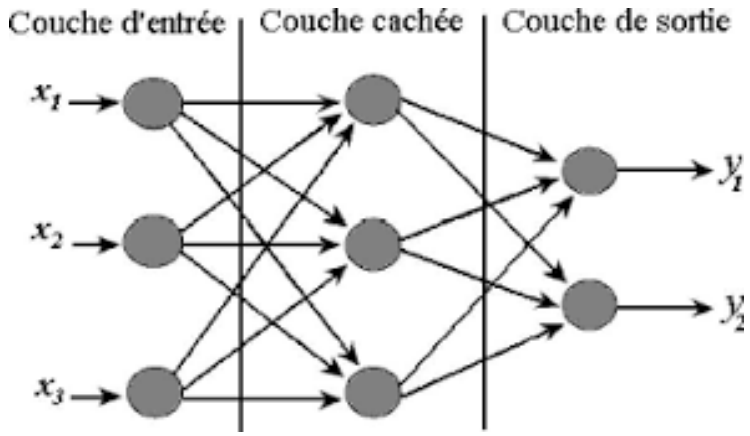


Figure 12. Architecture du réseau de neurones artificiels.

La deuxième classe regroupe les réseaux de neurones profonds dis en anglais *Deep Neural Networks* (DNN). Les réseaux de neurones profonds appartiennent à la famille des Réseaux de Neurones Artificiels. Les réseaux de neurones profonds ont plusieurs couches cachées. Parmi ces réseaux : le modèle Google LeNet, Face net pour la reconnaissance des images. La figure 13 montre une architecture DNN :

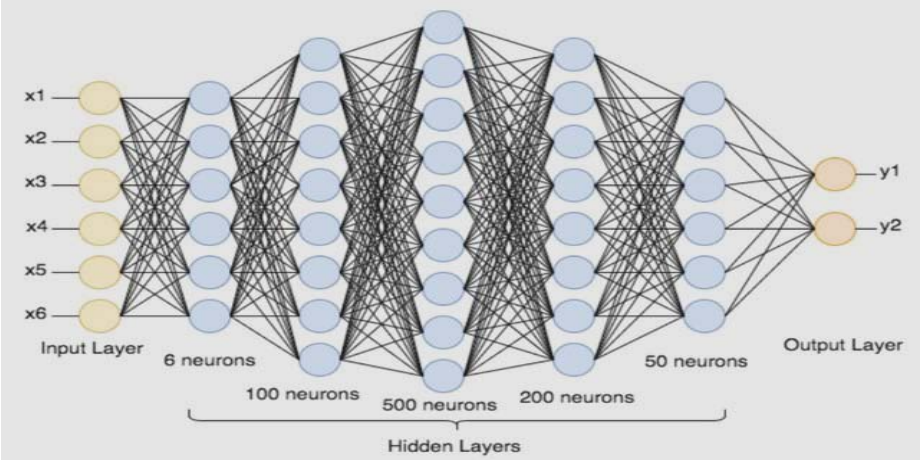


Figure 13. Architecture du réseau de neurones profonds (Bahi & Batouche, 2018)

Les DNN sont des réseaux de neurones très similaires aux réseaux de neurones de la première classe dont nous avons discuté, sauf qu'ils doivent implémenter un modèle plus complexe (un grand nombre de neurones, de couches cachées et de connexions). Le réseau de DNN a une architecture où les couches sont empilées les unes sur les autres. Le principe d'apprentissage pour les DNN est le même pour tous les problèmes d'apprentissage automatique (apprentissage supervisé).

Les réseaux de neurones profonds sont des réseaux de neurones artificiels qui utilisent l'apprentissage profond. Lorsque les procédures d'analyse normales ne sont pas applicables en raison de la complexité des données à traiter. L'apprentissage profond est l'un des derniers domaines de recherche en apprentissage automatique. Il est un sous domaine de l'apprentissage automatique, qui est à son tour un sous domaine de l'intelligence artificielle. L'apprentissage profond est une approche informatique qui imite le réseau de neurones dans un cerveau. Ce type d'apprentissage est appelé profond car il utilise plusieurs couches constituant l'architecture du modèle pour apprendre des données.

- La première couche est appelée la couche d'entrée
- La dernière couche est appelée couche de sortie
- Toutes les couches intermédiaires sont appelées couches cachées.

Chaque couche cachée est composée de neurones. Les neurones sont connectés les uns aux autres. Le neurone reçoit l'information d'entrée puis la transmet à la couche au-dessus de lui. La robustesse de l'information donnée au neurone dans la couche suivante dépend de la fonction d'activation, du poids et du biais.

Un réseau de neurones profond offre une meilleure précision dans de nombreuses tâches ; la détection et la classification des objets, la reconnaissance vocale etc. Pour plus de détails sur l'histoire des réseaux de neurones et de l'apprentissage en profondeur, il faut se référer à Goodfellow et al. (Goodfellow, Bengio, & Courville, 2016).

4.1 Types des réseaux de neurones profonds

Les réseaux de neurones profonds sont inspirés des réseaux de neurones biologiques. Ils ont été largement utilisés au cours des vingt dernières années dans divers domaines. Un apprentissage profond peut alors être défini comme des réseaux de neurones avec un grand nombre de paramètres et de couches. Dans cette section nous nous intéressons principalement aux quatre architectures qui sont ***Recurrent Neural Networks*** **(RNN),** ***Convolutional Neural Networks*** **(CNN),** ***Generative Adversarial Networks*** **(GANs) et** ***Deep Belief Networks*** **(DBN).**

4.1.1 Recurrent Neural Networks

Recurrent Neural Networks (RNN) sont des réseaux de neurones artificiels dont les connexions entre les neurones comprennent des boucles. Les RNN sont bien adaptés au traitement des séquences d'entrées. Les utilisent une architecture qui n'est pas différente de celle des ANN traditionnelles. La différence est que les RNN introduisent le concept de mémoire, où les sorties de certaines couches sont réinjectées dans les entrées d'une couche précédente en utilisant des liens dans le sens inverse. Cette réinjection permet l'analyse de données d'une manière séquentielles. De plus, les réseaux ANN traditionnels sont limités à une entrée de longueur fixe, tandis que les RNN n'ont pas une telle restriction. L'inclusion de liens entre les couches dans le sens inverse permet des boucles de rétroaction, qui sont utilisées pour aider le réseau à apprendre.

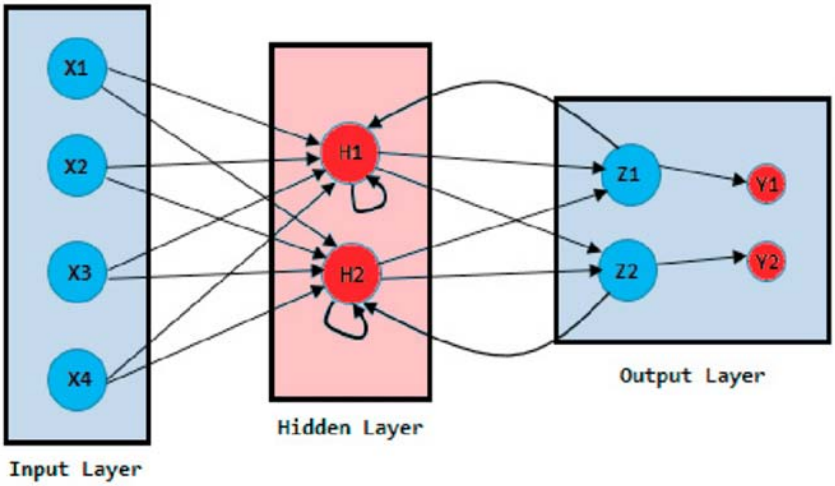


Figure 14. Architecture de *Recurrent Neural Network* (Bridle, 1990).

Les RNN sont capables de reconnaître de tirer un parti du contexte lié au temps. Par exemple, considérons un système qui apprend la reconnaissance de la parole, où le système va prendre en compte des mots prononcés récemment pour prédire la phrase suivante.

Les RNN ont récemment réussi à gérer des données séquentielles telles que la reconnaissance vocale (Graves, Mohamed, & Hinton, 2013), le traitement du langage naturel (Graves, & Jaitly, 2014), la reconnaissance d'actions (Sanin, Sanderson, Harandi, & Lovell, 2013), etc. La figure 14 montre l'un des types et les caractéristiques les plus importantes des RNN.

4.1.2 Convolutional Neural Networks

Convolutional Neural Network (CNN) (Lecun, Bottou, Bengio, &Haffner, 1998) est appliqué pour la première fois à la reconnaissance de caractères manuscrits. Un CNN est un cas particulier des réseaux de neurones profonds. Il est appliqué en vision par ordinateur. Le réseau neuronal CNN a plusieurs couches avec une architecture spécifique (figure 15). Ce type de réseaux est conçu pour extraire des caractéristiques de plus en plus complexes des données à chaque couche pour déterminer la sortie. Les CNN ont été utilisés comme extracteurs de caractéristiques de bas niveau d'une manière automatique. Cela permet d'éviter à son tour d'effectuer manuellement l'extraction des caractéristiques.

A l'intérieur du réseau CNN se trouvent des couches convolutives, qui appliquent des filtres glissants sur la hauteur et la largeur de l'image. Afin d'extraire des caractéristiques spécifiques telles que les coins, les bords, les caractéristiques de textures, etc. Le produit

scalaire des valeurs de pixels et de ces filtres donne un vecteur qui contient des valeurs qui sont exploitées à plusieurs itérations.

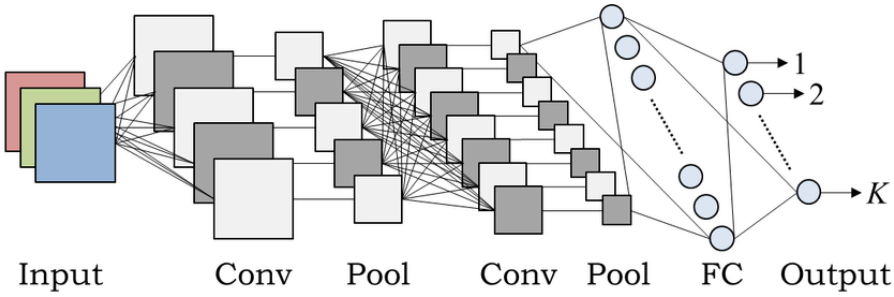


Figure 15. Architecture du réseau *Convolutional neural network* (Hidaka & Kurita, 2017).

Les réseaux CNN sont principalement utilisés lorsqu'il existe un ensemble de données non structurées. Les données non structurées peuvent avoir leur propre structure interne, mais ne se conforment pas parfaitement à une base de données ou à une feuille de calcul comme par exemple des images. Le CNN reçoit une image afin d'extraire des informations qui représente des objets dans cette image. Cette image en terme informatique est une collection de pixels. En effet, les pixels constituants une image sont des valeurs en entières de 0 à 255 pour l'image en niveaux de gris. Le codage de la couleur d'une image est composé de trois valeurs en entier, chaque valeur représentant la valeur d'une composante couleur par un entier de 0 à 255.

Pendant l'apprentissage des caractéristiques, le réseau identifiera des caractéristiques pertinentes grâce aux couches cachées, par exemple l'oreille ou bien la queue du chat, etc. Lorsque le réseau a parfaitement appris à reconnaître une image, il peut fournir une probabilité pour chaque image qu'il connaît. La catégorie (cible) avec la probabilité la plus élevée deviendra la prédiction du réseau.

4.1.3 Generative Adversarial Networks

Les *Generative Adversarial Networks* (GAN) sont une innovation récente dans l'apprentissage automatique. Les GAN ont été introduits par les chercheurs de l'Université de Montréal dans les travaux d'Ian Goodfellow et al. (2014). Les GAN sont des modèles génératifs qui permettent de créer de nouvelles instances de données qui ressemblent aux données utilisées dans l'apprentissage. Les GAN ont des architectures algorithmiques qui utilisent deux types de réseaux de neurones et qui sont opposant l'un à l'autre (les adversaires) : le générateur et le discriminateur. Le générateur est formé pour créer des images réelles

lorsque seul un ensemble des images bruitées sont données en entrées. Le Discriminateur est formé pour dire si l'image est réelle ou non.

Ils sont largement utilisés pour la reconstruction (génération) des images, la reconstruction de vidéos et la reconstruction de voix. L'utilisation des GAN pour le bien et le mal est énorme, car ils peuvent apprendre à imiter toute distribution de données. En effet, les GAN peuvent apprendre à créer du contenu similaire à l'être humain dans n'importe quel domaine : prose, musique, discours, images. Autrement dit, ils peuvent également être utilisés pour générer de faux contenus multimédias et constituent la technologie sous-jacente à *Deepfakes*. Par exemple, les GAN peuvent créer des images qui ressemblent à des photographies de visages humains, même si les visages n'appartiennent à aucune personne réelle.

4.1.4 Deep Belief Networks

Les *Deep Belief Networks* (DBN) (Rumelhart, & McClelland, 1987) sont des réseaux composés de plusieurs couches de type *Restricted Boltzmann Machine* (RBM) et sa dernière couche représente un classificateur. Dans RBM, il y a deux couches, à savoir la couche visible et la couche cachée. La *Restricted Boltzmann Machine* est un réseau de neurones à deux couches illustré sur la figure 16 (b), où la couche inferieur est utilisée pour l'entrée $x = (v_1, v_2, \dots, v_n)$ de n dimension. Cette couche est appelée la couche visible tandis que la couche supérieur est la variable cachée de m dimension.

La même figure montre que toute unité de la couche d'entrée et toute unité de la couche cachée sont complètement connectées. D'un autre côté, les unités de la couche d'entrée ne sont pas connectées les unes aux autres et les unités de la couche cachée ne sont pas connectées les unes aux autres. Chaque RBM se compose d'une couche visible (V) qui représente la couche d'entrée indiquée par (V) et une couche cachée indiquée par (H) connectée à un vecteur de poids (W) (Shao, Jiang, Wang, & Wang, 2017).

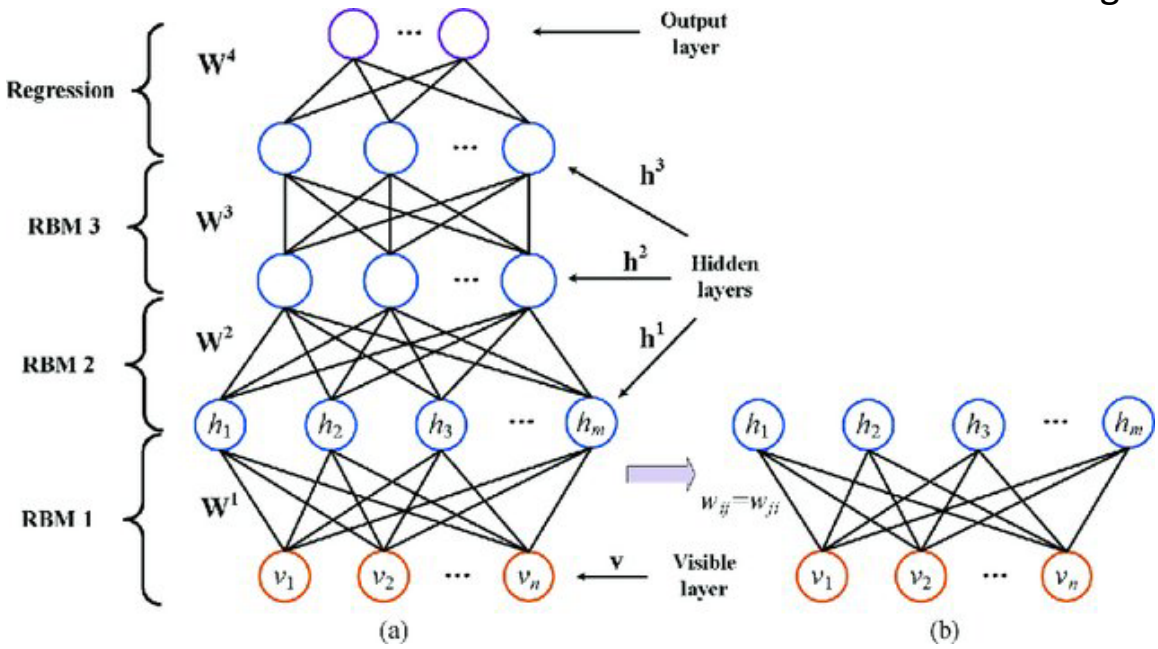


Figure 16. (a) DBN avec trois couches cachées (Shao, Jiang, Wang, & Wang, 2017)

(b) RBM avec n unités visibles et m unités cachées (Shao, Jiang, Wang, & Wang, 2017)

Il n'y a aucun lien entre les unités cachées, c'est pourquoi nous l'appelons *Restricted Boltzmann Machine*. Lorsqu'une RBM a appris, ses résultats des fonctions d'activations sont utilisés comme «données» pour former la prochaine RBM dans les DBN (figure 16. (a)). Autrement dit, À l'exception des premières et dernières couches, chaque couche d'un réseau de *Deep Belief Networks* a un double rôle : elle sert de couche cachée aux nœuds qui la précèdent et de couche d'entrée (ou «visible») aux nœuds qui viennent après. Il s'agit d'un réseau constitué de réseaux monocouches. Les DBN sont utilisés pour générer et reconnaître des images et des séquences vidéo.

4.2 Autoencoder

Un travail de recherche très intéressant concernant les réseaux de type encodeurs automatiques, qui s'appelle *Autoencoder* en anglais, a récemment été présenté par certains auteurs qui ont développé une stratégie efficace pour améliorer le processus d'apprentissage dans ce type de réseau (G.E. Hinton & R. Salakhutdinov, 2006). Ce type de réseaux de neurones à codage automatique entre dans la catégorie d'apprentissage non supervisé. En effet, les valeurs cibles sont définies pour être égales aux entrées. À l'intérieur de la structure de ce réseau, les informations d'entrées sont réduites vers une couche de codage centrale afin de trouver une représentation de faible dimension de l'ensemble de données.

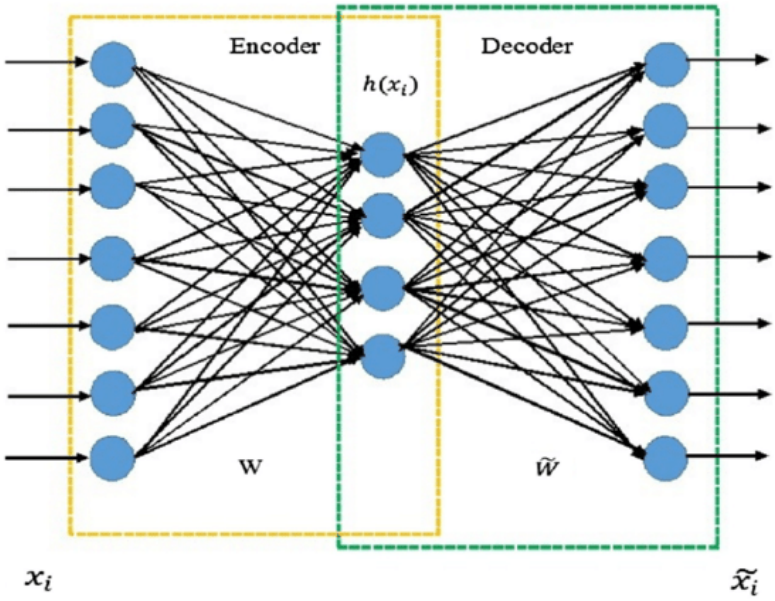


Figure 17. Phase d'encodeur et de décodeur dans le réseau *Autoencoder* (Ahmed, Wong, & Nandi, 2018).

La figure 17 montre le processus de fonctionnement d'un réseau de type encodeur automatique, il reconstruit l'entrée reçue en deux phases, une phase d'encodeur, qui correspond à une réduction de dimension pour l'entrée d'origine, et une phase de décodeur, capable de reconstruire l'entrée d'origine à partir de l'encodé (compressé). Dans la même figure, la couche nommée *h(x)* est la couche centrale qui se situe entre les deux parties de l'encodeur et du décodeur.

Les applications utiles des encodeurs automatiques sont le débruitage des données, la compression des données, la réduction de dimension pour la visualisation des données, etc. Les encodeurs automatiques sont efficacement utilisés pour résoudre de nombreux problèmes comme le débruitage des données, la compression des données et la réduction de dimension pour la visualisation des données, la reconnaissance faciale (Hinton, Krizhevsky, Wang, 2011), l'obtention du sens sémantique des mots (Liou, Huang, Yang, 2008).

5 Différence entre l'apprentissage automatique et l'apprentissage profond

L'apprentissage automatique et l'apprentissage profond sont deux sous-ensembles de l'intelligence artificielle qui attirent activement l'attention des chercheurs depuis plusieurs années. Ce qui est très intéressant maintenant est de comprendre quelle est la grande différence entre l'apprentissage automatique et l'apprentissage profond, et comment il est possible de choisir le meilleur des deux pour résoudre des problèmes dans le domaine de l'intelligence artificielle et de la reconnaissance de forme (voir le tableau 1).

	Apprentissage automatique	**Apprentissage profond**
Nombre des algorithmes existés	Nombreux	Peu
Temps de l'apprentissage	Court	Long
Taille de la base de données utilisée de l'apprentissage	Petite	Longue
Choisissez les caractéristiques	Oui	Non
Dépendances matérielles	Travailler sur une machine bas de gamme	Nécessite des machines puissantes dotées des GPU.
Les données dépendances	Excellentes performances sur un petit / moyen ensemble de données	Excellentes performances sur un grand ensemble de données
Ingénierie des caractéristiques	Besoin de comprendre les caractéristiques qui représentent les données	Pas besoin de comprendre les meilleures caractéristiques qui représentent les données
Temps d'exécution	De quelques minutes à quelques heures	Des heures jusqu'à des semaines.

Tableau 1. Différences entre l'apprentissage automatique et l'apprentissage profond

La principale différence entre l'apprentissage automatique et l'apprentissage profond est due à la façon dont les données sont présentées dans le système de classification. Les algorithmes d'apprentissage automatique nécessitent des données structurées, tandis que les réseaux d'apprentissage profond reposent sur des couches des réseaux de neurones artificiels. En outre, l'apprentissage automatique n'a pas besoin de gros volumes de données pour former l'algorithme ce qui n'est pas le cas pour l'apprentissage profond.

D'un autre côté, l'apprentissage profond nécessite un ensemble de données complet et diversifié. De plus, l'apprentissage automatique fournit un modèle plus rapide tandis que l'architecture d'apprentissage profond peut prendre des jours et des semaines pour s'entraîner. Par ailleurs, l'avantage de l'apprentissage profond par rapport à l'apprentissage automatique est qu'il est très précis. Par contre, dans d'apprentissage en profondeur, le réseau neuronal a appris à sélectionner les caractéristiques pertinentes. En effet, il n'est pas nécessaire de comprendre quelles caractéristiques ont la meilleure représentation des données. Mais, dans l'apprentissage automatique, il est indispensable de choisir manuellement des caractéristiques.

6 Conclusion

De l'intelligence artificielle à l'apprentissage automatique et puis à l’apprentissage profond, ce sont trois concepts liés à l'intelligence artificielle. Les trois concepts se combinent pour améliorer l'avenir de l'intelligence artificielle, mais ce n'est pas de l'intelligence artificielle. La nouvelle étape dans les processus de développement de systèmes d'intelligence artificielle se rapproche du défi de la réalisation de systèmes informatiques indépendants de l’intervention humaine. Autrement dit, ces systèmes sont capables d'imiter le comportement humain et la pensée.

V. Nouvelle approche pour améliorer le processus de classification des objets

1 Introduction

L'intelligence artificielle c'est la science qui essaie de rendre des machines aussi intelligentes que l'être humain pour reconnaître des formes existées dans des images et les classer dans les catégories souhaitées de manière simple et fiable. La reconnaissance des formes offre des solutions à de nombreux problèmes qui entrent dans la catégorie de la reconnaissance ou de la classification des images.

Souvent appelée «classification des images» ou «étiquetage des images», cette tâche essentielle est un élément fondamental de la résolution de nombreux problèmes d'apprentissage automatique basés sur la vision par ordinateur. La classification des images fait référence à un processus en vision par ordinateur qui peut classifier des images en fonction de leurs contenus visuels. Cette classification est une technique permet aux machines d'interpréter et de catégoriser ce qu'elles «voient» dans les images ou les vidéos. Par exemple, un algorithme de classification des images peut être conçu pour dire si une image contient une figure humaine ou non. Lorsque les humains regardent une photographie ou une vidéo, ils peuvent facilement repérer des personnes, des objets, des scènes et des détails visuels. Bien que, la détection d'un objet soit triviale pour les humains, une classification robuste des images reste un défi dans les applications de la vision par ordinateur.

Les images nécessite que leur classification ait généré une importance dans la classification des objets, car elles représentent un problème important dans de nombreuses

applications qui impliquent la détection du visage, la reconnaissance d'un caractère optique ou une empreinte digitale, la classification des anomalies dans les images médicales et dans tout autre domaine d'application à des fins de surveillance, d'inspection visuelle industrielle, de navigation automobile, de contrôle de robot, de télédétection et de nombreuses applications de vision par ordinateur.

2 Systèmes de la classification des images

La classification des images consiste à attribuer une étiquette à une image à partir d'un ensemble prédéfini de classes. Concrètement, cela consiste à analyser une image en entrée et à renvoyer une étiquette qui classifie cette l'image. L'étiquette provient toujours d'un ensemble prédéfini de classes possibles.

2.1 Classification des images en utilisant l'apprentissage automatique

Dans le système de classification des images en utilisant l'apprentissage automatique, nous pouvons diviser ce système en trois étapes : acquisition de données, prétraitement des données et classification des décisions, comme le montre la figure 18.

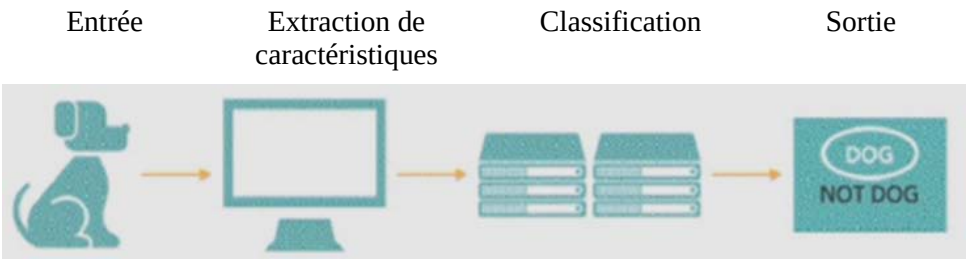


Figure 18. Apprentissage automatique traditionnel

Dans la première étape, les données sont acquises et prétraitées. Ensuite, dans un deuxième temps, le système extrait les caractéristiques et réduit également les dimensions de ses caractéristiques. Les extracteurs de certaines caractéristiques sont bien connus comme *Local Binary Pattern* et *Scale Invariant Feature Transform, Accelerate-KAZE, Speeded Up Robust Features* etc. La dernière étape est la classification. Le classificateur formé assigne l'image en entrée à l'une des classes des images en fonction des caractéristiques extraites.

Après avoir converti une image en vecteur de caractéristiques, un algorithme de classification prendra ce vecteur comme entrée et affichera une étiquette de classe. Le principe général de classification est que les algorithmes d'apprentissage traitent les vecteurs de caractéristiques comme des points dans un espace de dimension supérieure et essaient de

trouver des plans ou des surfaces qui partitionnent les espaces de dimension supérieure. Par conséquent, les exemples de la même classe se trouvent du même côté du plan ou de la surface.

2.2 Classification des images en utilisant l'apprentissage profond

À l'exception des méthodes qui utilisent l'apprentissage automatique, les approches de l'apprentissage profond offrent une précision étonnamment meilleure en vision par ordinateur. Les systèmes de classification basés sur l'apprentissage automatique permettent d'extraire des caractéristiques à partir des images en utilisant des algorithmes spécifiques comme par exemple SIFT. Donc, la partie cruciale de l'apprentissage automatique consiste à exploiter un ensemble de caractéristiques extraites pour que le système apprenne. Par contre, l'apprentissage profond résout ce problème en utilisant un réseau de neurones convolutionnels qui permet d'extraire et de classifier automatiquement les caractéristiques (figure 19).

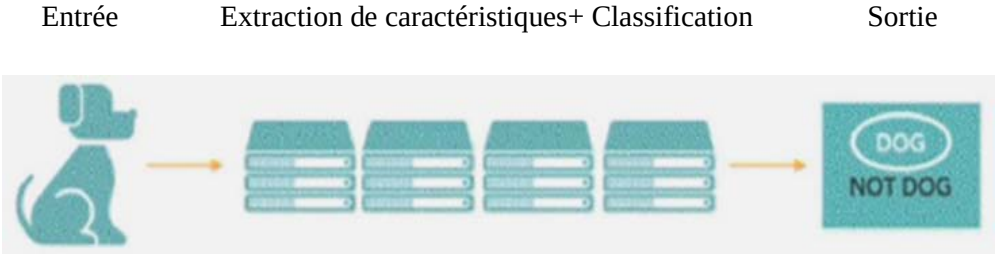


Figure 19. Apprentissage profond

La première couche d'un réseau neuronal convolutionnel apprendra de petits détails de l'image ; les couches suivantes combineront les caractéristiques précédentes pour créer des informations plus complexes. Dans le réseau neuronal convolutionnel, l'extraction des caractéristiques se fait à l'aide de filtres. Le réseau applique des filtres à l'image pour voir s'il y a une correspondance, c'est-à-dire que la forme de l'élément est identique à une partie de l'image. En effet, le processus d'extraction des caractéristiques se fait donc automatiquement.

3 Mesures pour évaluer les performances du système de classification

Cette section présente des mesures permettant d'évaluer dans quelle mesure le classificateur est bon (précis) pour prédire l'étiquette de classe des images. Les mesures du classificateur comprennent le taux de reconnaissance (précision globale), le rappel (ou sensibilité), la spécificité, la précision et le F-mesure.

Le tableau 2 montre plus de détails sur les formules utilisées pour calculer ces mesures. Ainsi que, supposons que nous utilisons notre classificateur sur un ensemble de tests des images étiquetés. *P* est le nombre des images positifs et N est le nombre des images négatifs.

La précision est une mesure spécifique, parce que ce mot est également utilisé comme un terme général pour désigner les capacités de la prédiction d'un classificateur.

- Vrais positifs ou *True Positive* en anglais (TP) : Soit TP le nombre de vrais positifs, ce sont les valeurs positives correctement prédites, ce qui signifie que la valeur de la classe réelle est oui et la valeur de la classe prédite est également oui.

Mesure	**Formule**
Taux de reconnaissance	$\frac{TP + TN}{P + N}$
Taux d'erreur	$\frac{TP + FN}{P + N}$
Rappel	$\frac{TP}{P}$
Spécificité	$\frac{TN}{N}$
Précision	$\frac{TP}{TP + FP}$
F, F-mesure, moyenne harmonique de précision et de rappel	$\frac{2 * précision * rappel}{précision + rappel}$

Tableau 2. Mesures d'évaluation (Han, & Kamber, 2005)

- Faux positifs ou *False Positive* en anglais (FP) : Soit FP le nombre de faux positifs, lorsque la classe réelle est non et que la classe prédite est oui, c à d. ce sont les valeurs négatives qui ont été incorrectement étiquetés comme positifs.
- Faux négatifs ou *False Negative* en anglais (FN) : Soit FN le nombre de faux négatifs, lorsque la classe réelle est oui, mais la classe prévue est non, c à d. ce sont les valeurs positives qui ont été mal étiquetés comme négatives.
- Vrais négatifs ou *True Negative* en anglais (TN): Soit TN le nombre de vrais négatives, Ce sont les valeurs négatives correctement prédites, ce qui signifie que la valeur de la classe réelle est non et que la valeur de la classe prédite est également non.

TP et TN permettent de dire quand le classificateur fait les choses correctement, tandis que FP et FN nous disent quand le classificateur se trompe (c.-à-d., une mauvaise classification).

La matrice de confusion (la table de contingence) (Fawcett, 2003) est un outil utile pour évaluer les performances d'un classificateur dans la reconnaissance et la prédiction des images de différentes classes. La matrice de confusion est un outil utile pour analyser dans quelle mesure votre classificateur peut reconnaître les images de différentes classes. C'est une table de taille *(n x n)*, où *n* est le nombre de classes. Chaque colonne de la matrice représente les classifications prévues et chaque ligne représente les classifications définies réelles.

	Classe prédite	
Classe Réelle	*True Positive*	*False Negative*
	False Positive	*True Negative*

Tableau 3. Matrice de confusion

Les entrées occupantes la diagonale de la matrice de confusion indiquent une bonne classification avec la plus grande précision et les entrées hors de la diagonale indiquent les erreurs de classification. Le tableau 3 indique la matrice de confusion pour deux classes avec quatre entrées.

4 Systèmes de classification des images

La classification des images est basée sur deux aspects très importants : les descripteurs et les classificateurs. Les descripteurs sont des vecteurs de caractéristiques qui peuvent représenter les images. De nombreuses applications en vision par ordinateur reposent sur la représentation des images par un certain nombre de points clés (Leutenegger, Chli, & Siegwart, 2011; Alahi, Ortiz, & Vandergheynst, 2012), c'est un véritable défi d'utiliser les points clés afin de représenter efficacement les images. Cette représentation doit être robuste, invariante à l'échelle, à la rotation et aux transformations affines.

Les approches de classification des images sont basées sur l'extraction de caractéristiques de bas niveau. Les caractéristiques extraites sont très utiles dans les applications de vision par ordinateur. L'extraction de caractéristiques commence par la détection des points clés et la construction du descripteur à la base de ces points. Le nombre de points clés détectés varie d'une image à l'autre, ce qui présente un problème dans l'utilisation de méthodes de

classification telles que SVM (Drif, & Ameur, 2016). Face à ce problème, Csurka et al. (2004) ont proposé une solution pour utiliser le modèle BoF pour représenter des images. En effet, ce modèle permet de générer des descripteurs de même taille. Aujourd'hui, ce modèle est très utilisé dans la recherche et la classification des images (Pérez, Bromberg, & Diaz, 2017; Chatterjee et al., 2018; Hu et al., 2019; Tiwari & Jain, 2019). L'utilisation de ce modèle fournit une représentation simple pour la classification des images et pour résoudre le problème créé par la désunion des caractéristiques locales. La précision de la classification dépend de la taille du BoF utilisé. Cette taille du BoF ne peut pas être ajustée empiriquement par des expériences (Jiang, Ngo et Yang, 2007).

Pérez et al., (2017) ont proposés une approche de classification des images basée sur bourgeons de vigne. Les auteurs de cette approche utilisent SIFT pour calculer des caractéristiques de bas niveau à l'aide de BoF. La méthode SPM a été proposée pour étendre la BoF en pyramides spatiales afin de récupérer les informations spatiales dans l'image. Cela se fait en divisant l'image en sous-régions de plus en plus fines. En effet, ils ont construit un BoF qui représente l'image en concaténant le BoF des caractéristiques locales trouvées dans chaque sous-région. Dans cette approche, les auteurs ont utilisé trois niveaux. Chaque niveau est obtenu par filtrage à l'aide du noyau gaussien. Par la suite, les auteurs ont calculé l'histogramme de chacun des trois niveaux. La distance entre deux histogrammes est donnée par la somme des distances des trois niveaux en calculant les couples d'histogrammes correspondant aux mêmes niveaux de pyramide. Cette méthode est utilisée pour trouver une correspondance entre deux ensembles de vecteurs dans un espace de caractéristiques. L'utilisation de la méthode SPM a montré qu'elle donne des résultats très satisfaisants par rapport à la BoF seule (Bosch, Zisserman et Munoz, 2007).

Plusieurs travaux ont été proposés pour améliorer la classification des images (Liu, Guo, Chamnongthai, & Prasetyo, 2017) ont proposé de combiner le descripteur de couleur appelé *Color Information Feature* (CIF) avec le descripteur LBP pour représenter les caractéristiques des images de texture de couleur. La fusion des deux descripteurs CIF et LBP donne des résultats prometteurs de récupération et de classification des images. Dans le même contexte, une approche d'appariement des pyramides spatiales basée sur le *Sparse Coding* du SIFT (ScSPM) est proposée pour la classification des images (Yang, Yu, Gong et Huang, 2009). Zheng, Zhao, Gao, et Wu (2018) ont proposé un modèle de classification conjointe qui s'appelle *Set-Level Joint Sparse Representation Classification* (SJSRC). Une autre approche a été proposée pour améliorer le *Sparse Coding* en se basant sur un ensemble

de patchs d'image. Cette approche est le cadre de la classification des images de bords ambigus en utilisant le *Sparse Coding* basé sur des correctifs de l'image (Lee, Kong et Lee, 2019).

Dictionary learning (DL) vise à apprendre des bases appropriées qui peuvent décrire efficacement les échantillons donnés (Xu, Li, Zhang, Yang, & You, 2017; Xu et al., 2013; Yali, Shiganga, Xili, & Xiaojun, 2019; Peng, Li, Liu, Wang et Li, 2018). DL est une classe de méthodes importantes dans l'apprentissage par dictionnaire. Leur tâche consiste à améliorer la classification des images en utilisant *Dictionary of Learning*. Comme dans (Yali et al., 2019), une approche d'apprentissage par dictionnaire appelé *Joint Local-Constraint and Fisher Discrimination based Dictionary Learning method (JLCFDDL)* est présentée afin d'améliorer les performances de la classification des images. Autres travaux sont proposés pour améliorer la classification des images en utilisant l'apprentissage pondéré ou *weighted learning* en aglais (Peng et al., 2018). Cet apprentissage utilise une matrice pondérée en diagonale pour construire un élément de contrainte afin de réduire l'auto-corrélation entre les échantillons d'apprentissage.

4.1 Classification des images à l'ère des Big data

La reconnaissance de formes à l'ère des Big data attirait beaucoup d'attention de nombreux chercheurs et appliquait dans divers domaines, notamment les soins de santé, la détection des anomalies et la sécurité etc. Dans ce contexte, Zerdoumi et al. (2018) exposent dans ses travaux en explorant les approches de reconnaissance d'images pour Big data. Big data pour la reconnaissance des formes des images se caractérisent par deux aspects: Le premier consiste à stocker de grandes quantités de données sur plusieurs machines, pas sur une seule. Deuxièmement, le manque de structure dans le concept de Big Data a rendu nécessaire l'utilisation d'outils et de techniques spécifiques.

Parmi ces outils, il y a Spark, Hadoop, MapReduce qui ont été utilisés comme des nouvelles techniques de reconnaissance de formes des images (Hashem et al., 2016; Hashem et al, 2017). Ces techniques sont utilisées pour résoudre certains des problèmes les plus critiques, à savoir comment exploiter efficacement d'énormes quantités de données. Ces outils et techniques actuels du Big Data sont moins susceptibles de résoudre complètement les vrais problèmes du Big Data. En effet, d'autres outils et des techniques de stockage et d'E/S plus progressives sont nécessaires pour des performances plus élevées afin de gérer la forte intensité de données cloud, informatique biologique et sociale, etc. (Shen, Liao,

Choudhary, Memik, & Kandemir, 2003 ; Gokhale, Cohen, Yoo, Miller, Jacob, Ulmer et Pearce, 2008).

De plus, l'intégration de différentes technologies sur la même plateforme augmente les risques de sécurité. (Salleh et Janczewski, 2016) ont fourni un article exposant l'état de l'art sur des problèmes de sécurité et de confidentialité des Big data, tandis que (Benjelloun et Ait Lahcen, 2015) ont présenté les défis de sécurité des Big data. Ils ont également présenté un état de l'art sur des méthodes, des mécanismes et des solutions utilisés pour protéger les systèmes d'information à forte intensité de données.

4.2 Classification des images à l'ère de Deep learning

Il existe une autre catégorie qui a fait l'objet d'études importantes. Les approches utilisant le *Deep learning* (Geert et al., 2017; Noord et Postma, 2017) donnent d'excellents résultats. Malheureusement, leurs performances dépendent directement de la taille et de la qualité des échantillons utilisés dans l'apprentissage, par exemple (Druzhkov & Kustikova, 2016; Choi, & Lee, 2019), qui ne sont pas nécessairement disponibles dans les applications pratiques. La grande taille et la qualité des échantillons utilisés en apprentissage permettent de générer un modèle de *Deep Learning*. Ce modèle nécessite des millions de paramètres à ajuster, ce qui limite l'application de ces approches dans de nombreux cas pratiques. Ainsi, ce type d'apprentissage nécessite la disponibilité de matériel efficace doté des processeurs avec une capacité de traitement et une puissance de calcul meilleures et plus rapides. En plus de cela, les GPU ont également été très utiles pour calculer des millions d'opérations matricielles à l'échelle, ce qui est l'opération la plus courante dans tout modèle d'apprentissage profond.

4.3 Points communs d'un système de classification des images

Les points communs d'un système de classification des images se caractérisent généralement par un descripteur de caractéristiques qui représente l'image et les méthodes d'apprentissage automatique pour apprendre à discriminer les classes des images en fonction de ces descripteurs. Les descripteurs de caractéristiques peuvent être extraits globalement de l'image afin de la représenter sous forme d'un histogramme. Un histogramme est l'un des moyens les plus couramment utilisés pour capturer des informations visuelles à partir d'une image. L'histogramme présente de nombreux avantages, tels que l'invariance en rotation de l'image et la robustesse des translations de l'image autour de l'axe de visualisation. De plus, le fait d'utiliser un seul histogramme représentant l'image pourrait entraîner une perte

d'information et la représentation de l'image ne comprend pas toutes ces caractéristiques. En d'autres termes, l'histogramme ne prend pas en compte les informations spatiales entre les objets de l'image. En effet, nous avons pensé à enrichir les caractéristiques de l'image incluses dans l'histogramme par d'autres caractéristiques calculées à partir de la partition de l'image. Cette solution fournit des informations supplémentaires pour capturer la distribution spatiale du contenu de l'image.

5 Approche proposée pour la classification des images

L'approche proposée pour la classification des images est basée sur le modèle spatial pyramidal (Lazebnik et al., 2006). Cette approche est principalement composée de quatre modules : extraction de caractéristiques, construction de BoF, application du modèle de pyramide spatiale et utilisation du classificateur *Random Forest* (RF). Tout d'abord, l'extraction de caractéristiques est effectuée pour obtenir le vecteur qui contient la description de l'image. Ensuite, le modèle BoF permet de définir un vocabulaire basé sur des caractéristiques extraites (Gafour Y. & Berrabah D., Benaissa M., 2019). Ensuite, le modèle spatial pyramidal est appliqué sur l'image et le BoF est calculé à chaque niveau pour obtenir les descripteurs qui représentant les images (Gafour Y. & Berrabah D., 2020). Dans le dernier module, nous utilisons le classificateur RF pour classer les images à l'aide de descripteurs calculés.

5.1 Processus de classification des images

5.1.1 Calcul du descripteur A-KAZE

Les caractéristiques locales ont un rôle très important dans une représentation efficace de l'image. Le choix des caractéristiques locales appropriées est nécessaire pour améliorer les performances de la classification des images. Pour avoir une meilleure discrimination, nous utilisons le descripteur A-KAZE, qui est très efficace. Il s'agit d'un descripteur rapide pour la détection et la description du changement d'échelle d'image dans des espaces non linéaires.

5.1.2 Bag of Features

Dans le cas des images, le dictionnaire est généralement composé de caractéristiques locales. Cela s'appelle Bag of Features (BoF). Ce modèle a été proposé en 2004 par Csurka, Dance, Fan, Willamowski, & Bray (2004) où la similitude visuelle est représentée par la

similitude des distributions de mots (c'est-à-dire des histogrammes clairsemés) sur les images comparées.

Les emplacements spatiaux des points clés du modèle BoF sont ignorés, ce qui est considéré comme son principal inconvénient. Pour cette raison, la plupart des travaux introduisent une étape de vérification géométrique pour comparer le contenu des images et identifier des images réellement similaires au sein des candidats préalablement trouvés par BoF. Il s'agit d'une tâche exigeante en calculs et de nombreuses tentatives ont été rapportées pour la simplifier (Sluzek, Kozera, 2014).

Dans cette approche, nous extrayons les caractéristiques locales en utilisant le descripteur A-KAZE. Nous utilisons un ensemble des images pour construire le vocabulaire visuel des caractéristiques locales en utilisant la méthode *K-means*. Ensuite, les caractéristiques locales de chaque image seront données au vocabulaire visuel pour construire le BoF qui contient la description de l'image ou d'une partie de celle-ci (figure 20).

5.1.3 Déterminer la taille du descripteur

Après avoir extrait les caractéristiques A-KAZE des images en niveaux de gris, un vocabulaire a été construit qui réduit le nombre de caractéristiques à l'aide de l'algorithme de classification *K-means*. Les groupes calculés après application *K-means* sont séparées par des caractéristiques similaires. Chaque centre de *cluster* est un mot de vocabulaire dans un dictionnaire de mots visuel. Le nombre de mots visuels est la taille du vocabulaire et correspond à *K* mots pour quantifier les caractéristiques dans l'algorithme de classification *K-means*.

5.1.4 Application de l’approche Spatial Pyramid Matching

Le SPM est appliqué à l'image et le BoF est calculé pour générer le vecteur caractéristique de chaque niveau. Chaque BoF calculé dans un niveau est utilisé pour capturer des informations spatiales dans l'image. Dans le SPM, nous avons plusieurs niveaux L (0, 1, 2 ... *L*) montrés dans la figure 8 (voir le chapitre III). Le nombre de partitions détermine le nombre de cellules d'espace. Enfin, BoF à tous les niveaux sont concaténés pour former un seul vecteur qui présente les caractéristiques globales, locales et spatiales de l'image.

V. Nouvelle approche pour améliorer le processus de classification des objets

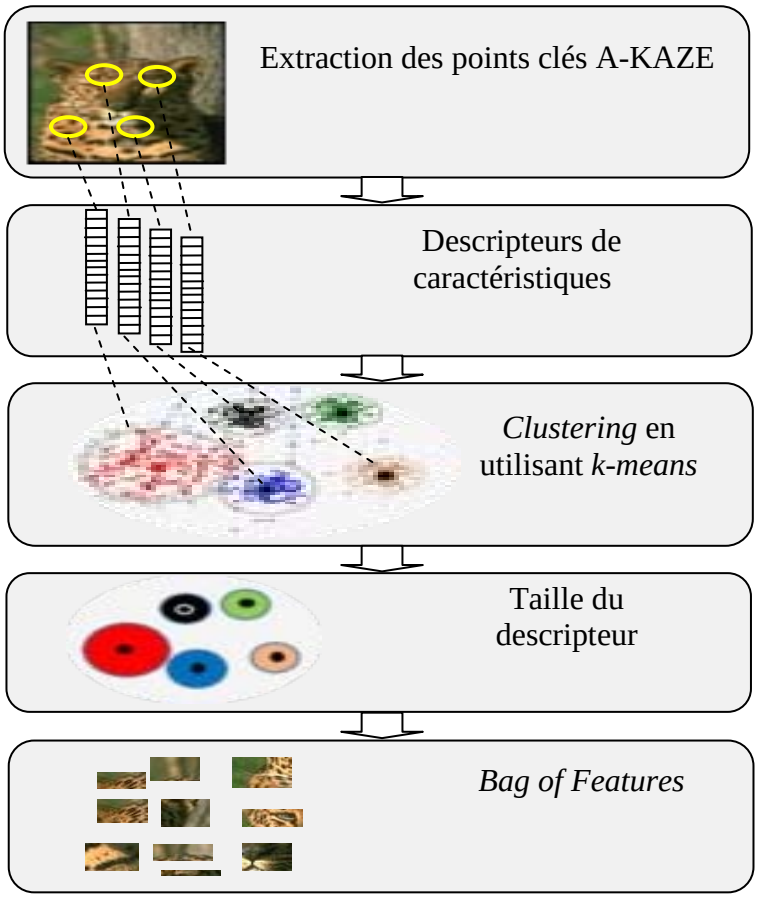


Figure 20. Modèle de BoF avec descripteur A-KAZE

Le système de classification proposé est illustré dans la figure 21. Pour utiliser le troisième niveau, nous devons diviser l'image en cellules d'image (4x4). Ensuite, nous appliquons le descripteur A-KAZE sur chaque cellule. Nos expériences ont montré que nous ne pouvons pas extraire des caractéristiques des images de petites tailles, c'est pourquoi nous avons limité le traitement aux deux niveaux (0, 1). SPM utilisé avec deux niveaux (0, 1) dans lequel les informations spatiales sont extraites et utilisées pour concaténer les histogrammes entre les sous-régions ensemble. Ainsi, chaque image est représentée par un vecteur de deux niveaux multiplié par K (taille du BoF) (figure 21). Pour évaluer les performances de classification, nous utilisons les deux niveaux SPM (0,1) avec les tailles de BoF suivantes : 100, 300, 500. Pour des performances optimales, la taille du BoF est différente lors de l'utilisation d'un seul niveau et de la pyramide spatiale. Par exemple, dans un seul niveau, L = 0 et BoF = 300, la taille de BoF dans ce niveau conservera la même taille de BoF utilisée.

En effet, si nous prenons L = 1 et BoF = 300, la taille du BoF dans ce niveau est de 1200. Elle représente la taille du BoF résultant de la concaténation du BoF des 4 partitions (300 * 4). En revanche, la taille du BoF devient 1500 (300 + 300 * 4) lorsque l'on applique la pyramide spatiale. Cette taille est calculée par la concaténation entre les deux tailles de niveaux L = 0 et L = 1.

5.1.5 Classification des images en utilisant Random Forest

Les BoF de chaque niveau SPM seront concaténés pour construire un vecteur global qui représente l'image. Ce vecteur sera exploité par le classificateur Random Forest (500 arbres standards) pour classifier les images. Les expériences de classification ont été réalisées sur différents types de bases de données pour évaluer la robustesse du descripteur proposé.

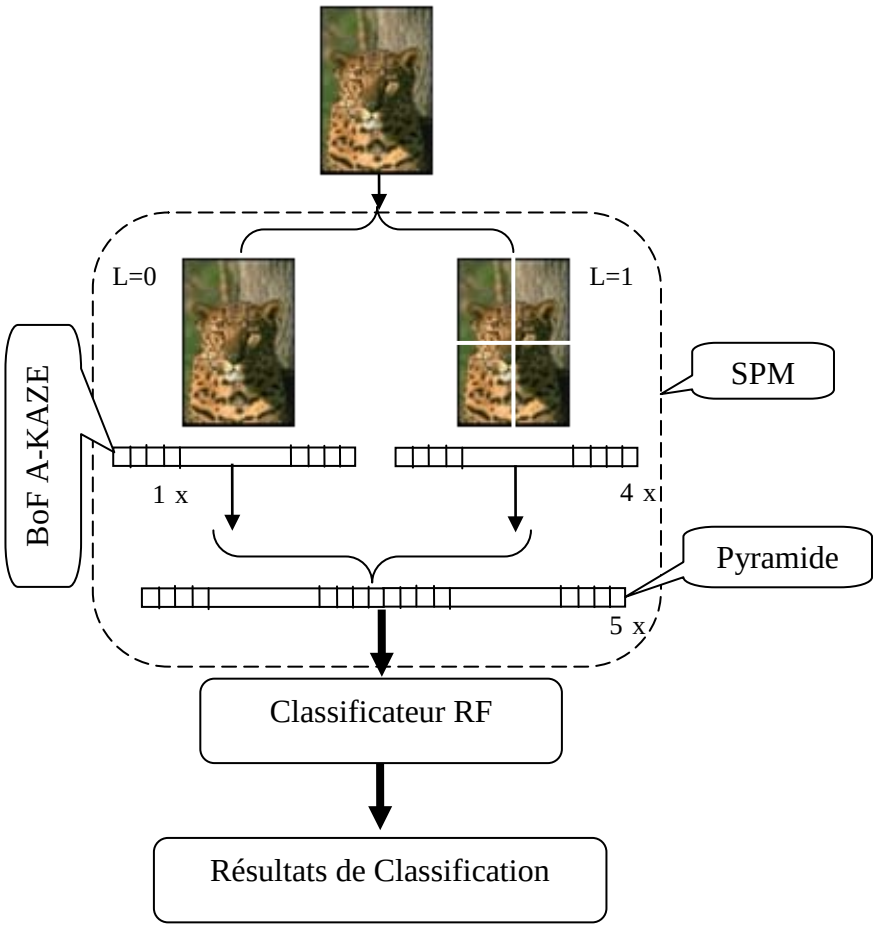


Figure 21. Architecture du système de classification proposé

5.2 Expériences et résultats

Pour évaluer notre approche, le processus d'expérimentation est effectué sur les quatres bases de données C**altech 101** (Fei-Fei, Fergu et Perona, 2007), **Caltech 256** (Griffin, Holub, Perona, 2007), **15 catégories de scènes** (Lazebnik et al. , 2006) et **Pascal VOC 2007** (Everingham, Gool, Williams, Winn et Zisserman, 2007). Ces bases de données ont été choisies pour les expériences en raison de leur grande variabilité inter-classe. La plupart des images de ces bases de données ont une résolution moyenne. Elles sont utilisées par plusieurs chercheurs pour évaluer leurs travaux.

V. Nouvelle approche pour améliorer le processus de classification des objets

Nous commençons nos expériences avec la base de données Caltech 101 à partir duquel nous choisissons au hasard 10 catégories (classes) des images. Ensuite, nous commençons le processus de classification sur ces images. Le résultat obtenu est enregistré comme résultat intermédiaire. Cette expérience est répétée 9 fois sur la même base de données en sélectionnant à chaque fois 10 catégories différentes choisies au hasard. Les résultats de ces 9 expériences sont également enregistrés comme résultats intermédiaires. Le résultat final est obtenu en calculant la moyenne des résultats intermédiaires des expériences. Pour mieux évaluer les performances de notre approche, nous avons répété toutes ces expériences sur les bases de données Caltech 256 et 15 catégories de scènes. Enfin, nous avons pensé à évaluer différemment notre approche en utilisant la base de données Pascal VOC 2007. En effet, au lieu de choisir aléatoirement 10 catégories, nous utilisons directement les 2 ensembles "trainval" et "test" fournis avec cette base de données. La précision de la classification est mesurée à l'aide de la précision moyenne ou en aglais est *Average Precision* (AP) (équation 5.1). Nous prenons la moyenne donnée par 20 classes.

$$\text{précision} = \frac{TP}{TP + FP} \qquad (5.1)$$

$TP : True\ \ Positive$
$FP : False\ Positive$

5.2.1 Caltech 101

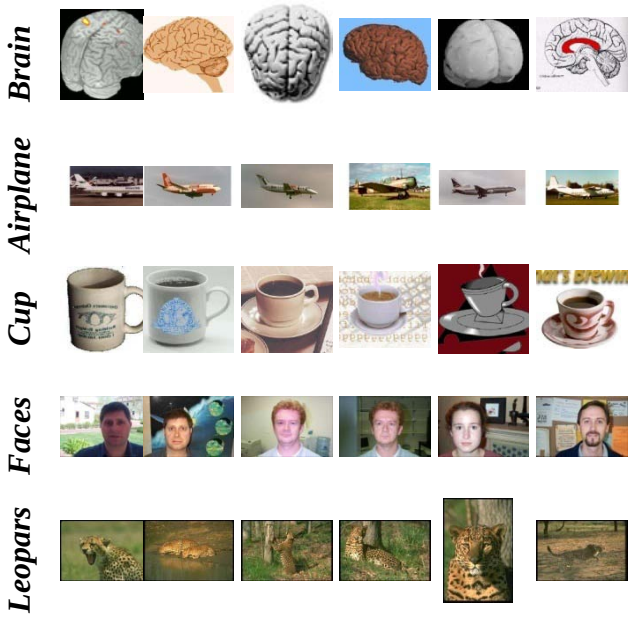


Figure 22. Quelques images de l'ensemble de données Caltech 101.

V. Nouvelle approche pour améliorer le processus de classification des objets

Cette base de données est collectée par Fei-Fei et al. (2007) se compose des images de 101 catégories et contient de 31 à 800 images par catégorie. La figure 22 présente quelques images de quelques catégories différentes dans cette base de données.

Pour les expériences, nous avons utilisé le classificateur RF. En effet, dans la phase d'apprentissage, nous avons choisi pour la première fois 15 images et une autre fois 30 images de chaque catégorie. Les images sont choisies au hasard. Pour la phase de test, ce sont 30 images qui sont choisies au hasard dans chaque catégorie. Cette procédure est répétée chaque fois que la taille du BoF du descripteur ou le niveau SPM est modifié.

Pour calculer les performances de notre approche, nous construisons la matrice de confusion (présentée dans le tableau 4) à chaque fin de l'expérience. Cette matrice contient le taux moyen de la classification des 10 catégories choisies et le taux global de l'expérience. Les valeurs dans la diagonale représentent les taux de classification moyens pour les classes individuelles. Ici, le taux de classement moyen est de 72%.

	Brain	***Airplanes***	***Cup***	***Face***	***Leopards***	***Motobiques***	***Sissors***	***Headphone***	***Menorah***	***Barrel***
Brain	0,63	0,07	0,07	0,00	0,00	0,03	0,00	0,00	0,03	0,17
Airplanes	0,03	0,50	0,10	0,03	0,00	0,07	0,03	0,07	0,07	0,10
Cup	0,00	0,03	0,73	0,03	0,07	0,00	0,03	0,03	0,00	0,07
Face	0,00	0,07	0,17	0,67	0,00	0,00	0,00	0,03	0,00	0,07
Leopards	0,03	0,03	0,07	0,03	0,57	0,00	0,07	0,03	0,00	0,17
Motobiques	0,00	0,17	0,00	0,00	0,00	0,70	0,00	0,13	0,00	0,00
Sissors	0,00	0,00	0,03	0,00	0,07	0,00	0,90	0,00	0,00	0,00
Headphone	0,00	0,00	0,00	0,00	0,03	0,00	0,00	0,97	0,00	0,00
Menorah	0,00	0,00	0,07	0,00	0,00	0,00	0,00	0,00	0,90	0,03
Barrel	0,00	0,13	0,07	0,00	0,10	0,00	0,03	0,00	0,03	0,63

Tableau 4. Matrice de confusion pour les 10 catégories choisies de la base de données Caltech 101.

Le tableau 5 montre les résultats de la classification de notre approche en utilisant 30 images dans l'apprentissage avec différents niveaux de SPM (0, 1) et différentes tailles de BoF (100, 300, 500). Les expériences montrent que le meilleur résultat est donné au niveau de la pyramide L = 1 avec une taille BoF égale à 300. Avec ces paramètres, un vecteur global

d'une taille égale à 1500 est généré par la concaténation d'un vecteur d'une taille égale à 300 généré au niveau L = 0 et un vecteur de taille égale à 1200 (4 * 300) généré au niveau L = 1 toujours avec une taille de BoF égale à 300.

Niveau (L)		**0 (1x1)**	**1 (2x2)**
BoF=100	***Un seul niveau***	57.63±0.25	61.2±0.72
	Pyramide	//	62.55±0.35
BoF=300	***Un seul niveau***	66.62±0.78	74.5±0.3
	Pyramide	//	76.73±0.42
BoF=500	***Un seul niveau***	75.48 ±0.33	64.44±0.39
	Pyramide	//	63.15±0.53

Tableau 5. Résultats d'expériences utilisant 30 images de chaque catégorie de la base de données Caltech 101 pour l'apprentissage.

Nous avons comparé notre approche à d'autres travaux en utilisant la même base de données. Les résultats de ces travaux sont présentés dans le tableau 6.

Méthodes	**Caltech 101**	
	Apprentissage de 15 images	**Apprentissage de 30 images**
Lazebnik et al. (2006)	56.40	64.6 ±0.8
(Zhang, Berg, Maire, Malik, 2006)	59.1	66.20
(Lou, Huang, Fan, & Xu, 2014)	73.3%	75.4%
(Gemert, Geusebroek, Veenman, & Smeulders, 2008)	//	64.5
(Wang, Yang, Yu, Lv, Huang, Gong, 2010) (LLC)	65.43	75.44
(Yang et al., 2009) (ScSPM)	70.8%	73.2%
(Jiang, Zhang, & Davis, 2012)	67.50	75.30
(Yang, Li, Tian, Duan, & Gao, 2009) (MKL)	73.2	84.3
(Gemert, Veenman, Smeulders, & Geusebroek, 2010)	//	64.1±1.5
(Zhu, Q. et al., 2017)	68.20±0.041	74.30±0.036
(Chen, Li, Peng, Wong, 2015)	62.09	69.18
Rahman, Rahman, Rahman, Hossain, & Shoyaib, 2017) (DTCTH + LSVM)	63.66	72.26
Ours approach RF+SPM (A-KAZE)	67.92±0.79	76.73±0.42

Tableau 6. Précisions de classification (%) sur Caltech 101

Le meilleur résultat est donné par les travaux de (Jiang et al., 2012) qui ont atteint plus de 75%. Ces auteurs proposent un *Submodular Dictionary Learning* (SDL) en extrayant des caractéristiques de pyramide spatiale pour chaque image, puis les réduisons à 3000 dimensions par ACP. Les caractéristiques de *Sparse Coding* du descripteur SIFT sont calculées à partir des caractéristiques de la pyramide spatiale. Nous avons obtenu le même résultat avec notre approche au niveau L = 0 avec une taille de BoF égale à 500. De plus, nous notons une légère amélioration de 1% dans la classification lorsque nous utilisons la pyramide spatiale au niveau L = 1 avec une taille de BoF égale à 300. Cependant, les résultats de notre approche commencent à se détériorer toujours au niveau L = 1 mais avec une taille de BoF égale à 500. Cependant, les résultats de notre approche commencent à se détériorer lorsque la taille de BoF est égale à 500, même au niveau L = 1 (64,44% pour le niveau unique et 63,15% pour le niveau pyramidal). Cette dégradation est due à la taille du BoF (2000 en un seul niveau, 2500 dans la pyramide spatiale) utilisée.

5.2.2 Caltech 256

La base de données Caltech 256 a été collectée en sélectionnant un ensemble de catégories d'objets téléchargées depuis Google Images, puis en excluant toutes les images qui ne correspondaient pas à la catégorie. Le nombre total de catégories d'objets est de 256 contenant un total de 30607 images. En effet, ce nombre est plus que doublé par rapport à Caltech 101. Dans chaque catégorie, il y a entre 31 et 80 images par catégorie.

Pour évaluer notre approche avec cette base de données, nous suivons la même procédure que les expériences utilisées dans la section précédente. Au cours de cette expérience, nous avons fixé la taille du BoF à 300 car avec cette taille, nous avons fait plusieurs expérimentations et nous avons trouvés que 300 est la meilleur taille du BoF donnant des bons résultats.

Dans le tableau 7, nous présentons une comparaison détaillée des résultats de certaines approches sur l'ensemble de données Caltech 256. À partir de ce tableau, nous remarquons que notre approche donne des résultats comparables lorsque nous utilisons 15 images dans l'apprentissage. Nous constatons qu'il y a une amélioration des résultats lorsque nous utilisons 30 images dans l'apprentissage. Les résultats de notre approche dépassent d'environ 1% les résultats de Zhu et al. (2017). Dans ce dernier, ils ont utilisé une stratégie de codage des caractéristiques spatiales préservant la localité. Nous constatons que les résultats obtenus sont inférieurs à 50% en raison des caractéristiques des images contenues dans cette base de

données. Les images de cette base de données sont très variées et complexes par rapport à la base de données caltech 101.

Méthodes	Caltech 256	
	Apprentissage de 15 images	Apprentissage de 30 images
(Perronnin et al., 2010) IFK (SIFT+Color)	34.7 ±0.2	40.8± 0.1
(Gemert et al., 2008)	//	26.6
(Wang et al., 2010) (LLC)	34.36	41.19
(Yang et al., 2009) (ScSPM)	27.73	34.02
(Gemert et al., 2010)	//	27.2±0.4
(Zhu et al., 2017)	34.38±0.117	42.42±0.124
(Chen et al., 2015)	28.58±0.35	35.20±0.36
(Rahman et al., 2017) (DTCTH + LSVM)	27.43±0.37	33.57±0.43
Ours approach RF +SPM (A-KAZE)	34.36±0.25	43.2±0.56

Tableau 7. Précisions de classification (%) sur Caltech 256.

5.2.3 15 catégories de scènes

Cette base de données a été développée progressivement Elle a été collectée par (Oliva, & Torralba, 2001) en commençant initialement par 8 catégories. Ensuite, Fei-Fei et Perona (2005) proposent d'ajouter 5 catégories supplémentaires à cette base de données. Enfin, Lazebnik et al. (2006) ont introduit 2 catégories supplémentaires. Les 15 catégories de scènes sont forêt, cuisine, chambre, banlieue, industriel, bureau, immeuble de grande hauteur, cité intérieure, chambre, rue, autoroute, campagne, salon, magasin et montagne. Les images mesurent environ 300 × 250 en taille moyenne, avec 210 à 410 images dans chaque catégorie. La base de données de 15 catégories de scènes contient au total 4485 images. Quelques images de cette base sont montrées dans la figure 23.

Pour évaluer notre approche sur la base de données 15 catégories de scènes, nous suivons la même procédure expérimentale de Lazebnik et al., (2006). Nous prenons 100 images par classe pour l'apprentissage et le reste des images pour la comparaison. Dans notre évaluation, sur les 10 catégories sélectionnées au hasard, seules 07 catégories atteignent plus de 75%.

V. Nouvelle approche pour améliorer le processus de classification des objets

Les expériences montrent que le meilleur résultat de la classification de notre approche sur cette base de données est donné au niveau de la pyramide L = 1 avec une taille de BoF égale à 500. Les résultats de comparaison détaillés sont présentés dans le tableau 8.

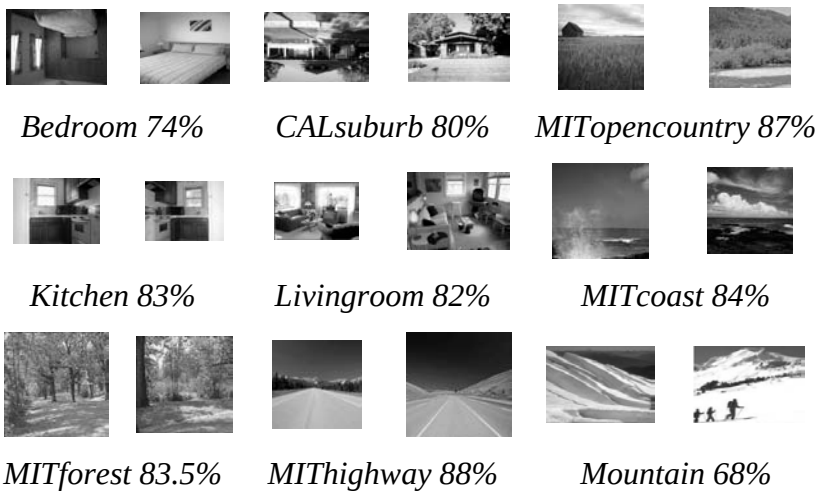


Figure 23. Exemples des images de classes avec la plus grande précision de classification de l'ensemble de données 15 catégories de scènes.

Les résultats de classification des images évalués sur la base de données 15 catégories de scènes sont très satisfaisants par rapport aux résultats obtenus des bases de données Caltech 101 et Caltech 256.

Cette augmentation est attribuée au nombre d'images utilisées dans d'apprentissage, qui est estimé à 100. La classification par la méthode RF à l'aide du descripteur A-KAZE sous forme pyramidale donne une augmentation de 1% par rapport au résultat donnée par Zhu et al. (2017).

Méthodes	**15scene catégories** **Apprentissage de 100 images**
(Lazebnik et al., 2006)	81.4 ±0.5
(Gemert et al., 2008)	78.5
(Yang et al., 2009)(ScSPM)	80.28 ± 0.93
(Gemert et al., 2010)	76.7±0.4
(Zhu et al., 2017)	83.62±0.64
(Rahman et al., 2017) (DTCTH + LSVM)	82.66 ± 0.50
Ours approach RF +SPM (A-KAZE)	84.56±0.24

Tableau 8. Précisions de classification (%) sur les 15 catégories de scènes.

5.2.4 Pascal VOC 2007

La base de données Pascal VOC 2007 est considérée comme une bonne base de données pour la création et l'évaluation des algorithmes de classification des images et de détection des objets. Elle est composée de 20 catégories, Quelques images de cette base sont montrées dans la figure 24.

Figure 24. Exemples des images de la base de données Pascal VOC 2007.

Méthodes	Pascal VOC 2007 AP (en %)
(Perronnin et al., 2010) IFK (SIFT+Color)	60.3
(Perronnin et al., 2010) IFK (SIFT)	58.3
(Wang et al., 2010) (LLC)	59.3
(Yang et al., 2009) (MKL)	62.2
(Gemert et al., 2010)	60.5
(Cevikalp & Triggs, 2017)	53.2
Notre approche RF +SPM (A-KAZE)	60.1

Tableau 9. Précisions de classification (%) sur Pascal VOC 2007

Dans cette expérience, nous utilisons 5011 images pour l'apprentissage et 4952 pour le test. Le tableau 6 donne une comparaison entre les performances de notre approche et celles d'autres approches récemment publiées (Cevikalp & Triggs, 2017). Il est à noter que l'approche Multiple Kernel Learning (MKL) (Yang et al., 2009) a donné de meilleures

performances dans le challenge Pascal VOC 2007 (62,2%). Notre approche atteint des performances similaires aux méthodes de (Perronnin et al., 2010; Gemert et al., 2010) citées dans le tableau 6. En fait, Perronnin et al. (2010) ont utilisé SVM linéaire et le descripteur SIFT. Ils ont atteint une précision de 60,3%

6 Conclusion

Dans ce chapitre, nous avons tenté d'amélioration les performances des systèmes de classification par l'utilisation de la pyramide spatiale. Nous avons utilisé le descripteur A-KAZE et SPM pour la description des informations spatiales dans l'image. Ce descripteur est utilisé en combinaison avec le modèle BoF pour construire un vecteur qui représente l'image à l'aide du modèle SPM qui peut améliorer les performances finales de la classification des images. Les expériences réalisées sur Caltech 101, Caltech 256, 15 catégories de scènes, Pascal VOC 2007 en utilisant la méthode Random forest démontrent l'efficacité de l'approche proposée. Ainsi, les résultats de la classification des images obtenus en utilisant ces bases de données sont très satisfaisants.

VI. Extraction automatique des caractéristiques pour la reconnaissance faciale

1 Introduction

La biométrie est un ensemble de technologies appelées "technologies biométriques" qui exploitent des caractéristiques physiques ou comportementales humaines telles que les empreintes digitales, la signature, l'iris, la voix, le visage et les gestes de la main pour identifier ou reconnaître des personnes. En termes de sécurité et surtout d'authentification des personnes, la biométrie offre plus d'avantages que les autres méthodes existantes telles que les clés, les numéros d'identification (ID), les mots de passe et les cartes magnétiques. En effet, les technologies biométriques sont plus sûres que les cartes de contrôle d'accès automatique et leurs codes secrets qui peuvent être volés ou égarés très facilement.

La reconnaissance faciale utilise des paramètres biométriques pour reconnaître ou vérifier l'identité d'une personne. Un système de reconnaissance faciale pour la sécurité et la surveillance doit être robuste, en particulier pour les variations de posture et les changements de luminance. Ces dernières années, la reconnaissance faciale a attiré beaucoup d'attention et elle est largement développée. Bien que, ses performances soient bien atteintes dans un environnement contrôlé. Mais, elles sont encore loin d'être satisfaisantes dans des applications réelles. Il existe de nombreuses approches efficaces pour résoudre les problèmes de reconnaissance faciale de manière satisfaisante, mais les variations d'expression, de pose, d'occlusion et d'éclairage restent des problèmes critiques qui affectent les performances de la reconnaissance faciale. En fait, changer la pose et la luminance d'un même visage peut changer radicalement l'apparence de la personne. Ainsi, ces changements rendent le

processus de reconnaissance faciale difficile car ces changements peuvent affecter la précision de ce processus.

Notre principale contribution dans cette thèse est de développer une approche robuste, notamment en ce qui concerne la variation des poses et les changements de luminance. En effet, nous avons proposé un nouveau descripteur que nous avons nommé *Honeycomb-Local Binary Pattern* (Ho-LBP) (Gafour, Berrabah. & Gafour, 2020 ; Gafour, Berrabah & Mahmoudi, 2018) qui est basé sur le descripteur *Local Binary Pattern* (LBP) (Ojala, Pietikäinen & Harwood, 1996 ; Ojala, Pietikäinen & Mäenpää, 2001).

Il s'agit d'un descripteur adapté à la structure en nid d'abeille pour décrire un pixel en niveaux de gris dans une image. Cette structure permet de réduire les effets du changement de luminosité et de la variation des poses pour améliorer la précision et augmenter les chances de reconnaître les visages.

Afin de vérifier les performances de notre approche et montrer l'efficacité des variantes de notre descripteur. Nous avons utilisé les trois classificateurs les plus utilisés, qui sont : *Support Vector Machine* (SVM), *Random Forest* (RF) et *K-Nearest Neighbors* (K-NN) et des bases de données de références.

2 Systèmes biométriques

La biométrie est actuellement un domaine de recherche très actif couvrant plusieurs disciplines telles que le traitement d'images et la reconnaissance de formes en vision par ordinateur. L'objectif principal de la biométrie est de construire des systèmes capables d'identifier les personnes à partir de certaines caractéristiques observables qui seraient puissants et alternatifs aux schémas d'authentification traditionnels. Les systèmes biométriques offrent plusieurs avantages par rapport aux modèles d'authentification traditionnels comme par exemple les mots de passe. Ces systèmes sont plus fiables que les modèles d'authentification traditionnels car les caractéristiques biométriques ne peuvent pas être perdus ou oubliés. Les caractéristiques biométriques sont difficiles à copier, à partager et à distribuer ; et ces caractéristiques exigent la présence de la personne authentifiée au moment de l'authentification. L'authentification biométrique fait référence à l'établissement d'une identité basée sur les caractéristiques physiologiques et comportementales montrées par la figure 25 d’une personne, telles que le visage, les empreintes digitales, l'iris, la géométrie de la main, la voix, la signature, etc.

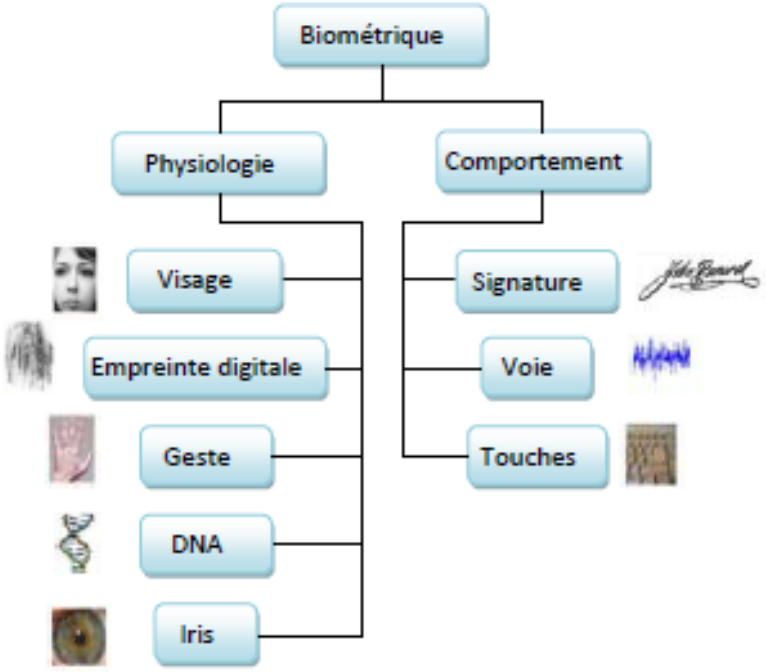


Figure 25. Classification des caractéristiques biométriques

3 Caractéristiques biométriques

Les caractéristiques physiologiques uniques pour identifier une personne sont caractérisées par les propriétés suivantes (Anot, Singh, 2016) :

Universel : Les caractéristiques de chaque personne doivent être universelles c.-à-d. que ces caractéristiques sont inchangeables même dans des situations particulières, par exemple un accident ou bien une maladie.

Mesurabilité : Les caractéristiques doivent s'adapter à la capture dans un laps de temps et doivent être collectées facilement.

Singularité : Les caractéristiques doivent avoir des propriétés distinctives et doivent être propres à une personne afin de la distinguer d'une autre. Le poids, la taille, la couleur des yeux et les cheveux sont tous des composants uniques mais n'offrent pas suffisamment de distinction pour être utiles pour la classification.

Acceptation : La capture des caractéristiques devrai être satisfaite de la majorité des citoyens. Il faut éviter les innovations particulièrement persistantes. Ces innovations qui nécessitent de prendre une partie du corps humain ou qui (apparemment) altèrent le corps humain.

Réductibilité : les données capturées doivent être sauvegardé dans un fichier facile à manipuler.

Confidentialité : Cette procédure ne doit pas briser la sécurité de la personne. La comparaison des caractéristiques à moins de probabilités de similitude et elles sont plus fiable en fonction de l'identification.

Inimitable : les caractéristiques doivent être iréproductibles autrement. Moins les caractéristiques sont reproductibles, plus elles seront probables.

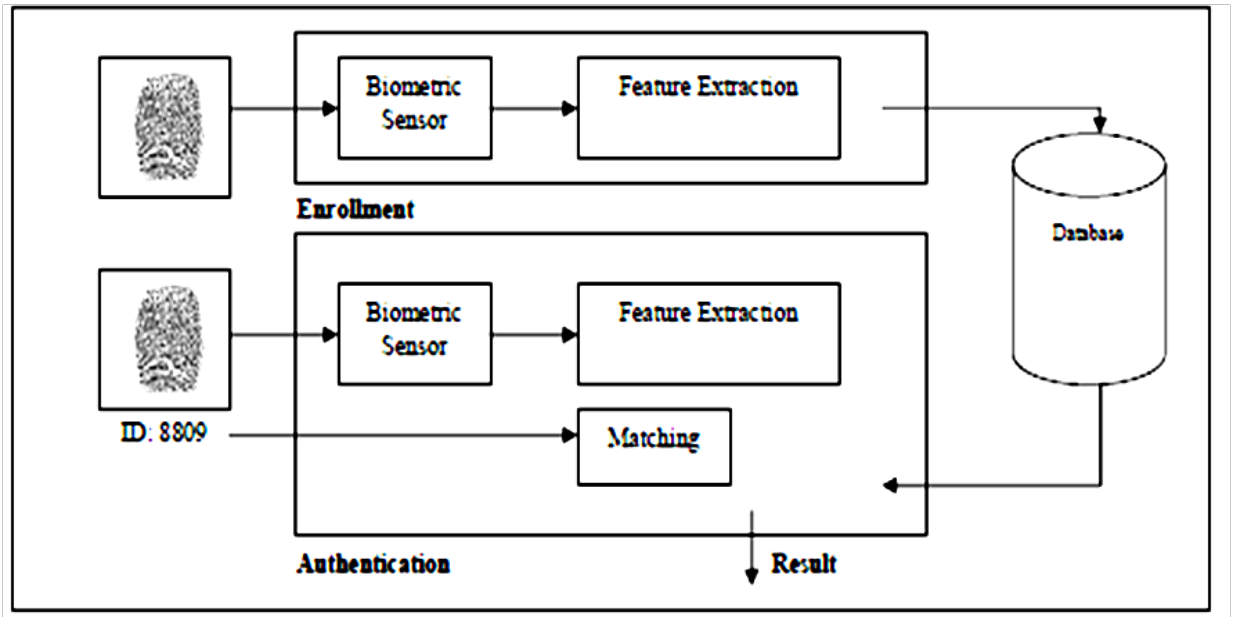


Figure 26. Système biométrique général (Tripathi, 2011).

Un système biométrique (figure 26) peut être soit un système **d'identification** ou bien un système de **vérification** (authentification). Les deux systèmes sont définis ci-dessous (Anot, Singh, 2016).

Identification (1 : n) - un-à-plusieurs : la biométrie peut être utilisée pour déterminer l'identité d'une personne, même sans son acceptante à l'avance. Par exemple, en examinant un groupe de personne à l'aide d'une caméra et l'utilisation de la technologie de reconnaissance faciale, nous pouvons vérifier les correspondances qui sont déjà stockées dans notre base de données.

Vérification (1: 1) individuelle : la biométrie peut également être utilisée pour vérifier l'identité d'une personne. Par exemple, nous pouvons autoriser l'accès d'une personne

physique à un espace spécifique en utilisant ces empreintes digitales ou bien en utilisant l'examen de la rétine.

4 Systèmes de reconnaissance faciale

La technologie biométrique devient non seulement de plus en plus importante, mais aussi de plus en plus étudiée par de nombreux chercheurs, car la reconnaissance faciale est également considérée comme un élément important des environnements intelligents (Ekenel & Sankur, 2004). La reconnaissance faciale devient l'une des techniques d'authentification les plus biométriques de ces dernières années. C'est une application intéressante et réussie de la reconnaissance des formes et de l'analyse de l'image. Elle permet l'identification des humains par les caractéristiques uniques de leurs visages. La reconnaissance faciale est devenue une branche spécialisée dans le vaste domaine de la vision par ordinateur. Il s'agit d'un système biométrique qui utilise divers algorithmes ou techniques à des fins d'identification et de sécurité.

Grâce à ses performances incomparables, la reconnaissance faciale englobe à la fois les technologies utilisées pour mesurer et celles appliquées pour analyser les caractéristiques uniques d'une personne. Elle a attiré une grande attention (Gafour & Berrabah, 2018 ; Liu, Fieguth, Zhao, Pietikäinen & Hud, 2016 ; Yuan, Shi, Xia, Zhang & Li, 2019) en raison des progrès des descripteurs locaux (El merabet, Ruichek & El idrissi, 2019) et des méthodes de classification qui répondent aux exigences croissantes des applications du monde réel. Son principal problème est dû aux variations d'expression, d'éclairage, de vieillissement, de posture et d'occlusion. Un processus de reconnaissance faciale doit être robuste aux changements de visage, en particulier pour les systèmes qui gérent la sécurité et la surveillance.

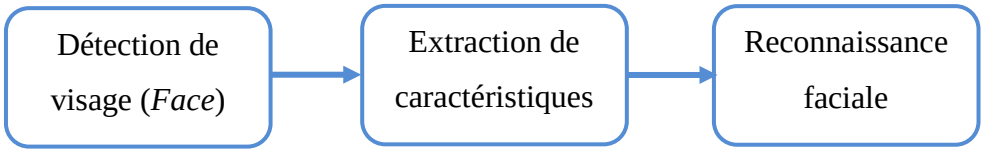


Figure 27. Processus de la reconnaissance faciale

Comme le montre la figure 27, un système de reconnaissance faciale peut être divisé en trois étapes : la détection des visages, l'extraction de caractéristiques et la reconnaissance faciale.

VI. Extraction automatique des caractéristiques pour la reconnaissance faciale

Le système de reconnaissance faciale commence par détecter l'existence d'un visage dans une image. Si le système de détection des visages a décidé que l'image contient un visage donc son rôle est de localiser la position d'un ou plusieurs visages dans l'image. En effet, cette étape devient difficile si des variations de position, d'expression faciale (sourire, surprise, etc.), d'illumination, d'orientation et de critères morphologiques (moustaches, lunettes, etc.) se produisent. Toutes ces variations peuvent empêcher la détection correcte du visage et par conséquent diminuer le taux de détection du visage.

Après avoir détecté un visage dans une image, nous procédons à l'extraction des caractéristiques du visage (Mejdoub, Amar, 2013; Borgi, Labate, El'Arbi, Amar, 2013). Cette étape est importante pour la reconnaissance faciale. Cette étape consiste à extraire un vecteur de caractéristique appelé signature qui va représenter le visage détecté. Il doit vérifier l'unicité du visage, ainsi que la propriété de discriminer entre deux individus différents.

Enfin, la reconnaissance faciale implique l'authentification et l'identification. L'authentification consiste à comparer un visage avec un autre afin d'approuver l'identité demandée. L'identification, compare un visage à plusieurs autres visages donnés pour trouver l'identité du visage parmi plusieurs visages stockés. La reconnaissance faciale existe essentiellement trois approches pour la reconnaissance faciale (Pandya, Rathod, Jadav, 2013):

A/ Approche basée sur les caractéristiques : dans l'approche basée sur les caractéristiques telles que le nez, les lèvres, les yeux sont segmentées et peuvent être utilisées comme données d'entrées dans la détection des visages pour faciliter la tâche de reconnaissance des visages.

B/ Approche holistique : dans l'approche holistique, le visage entier est pris comme entrée dans le système de détection de visage pour effectuer la reconnaissance faciale.

C/ Approche hybride : l'approche hybride est une combinaison entre les deux approches précédentes. Dans cette approche, des parties locales du visage et le visage entier sont utilisés comme entrée pour le système de détection de visage.

5 Approches de la reconnaissance faciale

Dans les approches qui abordent le problème de la reconnaissance faciale, les chercheurs se sont concentrés sur deux catégories : la première est le *Deep learning* (Sun, Liang, Wang et Tang, 2015 ; Parkhi, Vedaldi et Zisserman, 2015 ; Schroff, Kalenichenko et Philbin, 2015 ;

Wen, Zhang & Qiao, 2016 ; Taigman, Yang, Ranzato & Wolf, 2014 ; Ding & Tao, 2015). Tandis que, la seconde est basée sur l'extraction des caractéristiques des images afin qu'elles puissent être exploitées par des classificateurs (Lei, Pietikainen & Li, 2014 ; Zhang, Wang, Zhu, Chen et Chen, 2015). Les deux catégories ont fait l'objet d'études importantes.

Les approches qui utilisent le *Deep learning* donnent d'excellents résultats. Malheureusement, leurs performances dépendent directement de la taille et de la qualité des échantillons utilisés dans l'apprentissage ; par exemple 4 millions d'échantillons (Taigman et al., 2014) qui ne sont pas évidemment disponibles dans la pratique. La grande taille et la qualité des échantillons utilisés en apprentissage profond permettent de générer un modèle de *Deep Learning*. Ce modèle nécessite des millions de paramètres à ajuster, ce qui limite l'application de ces approches dans de nombreux cas pratiques.

Les approches de la deuxième catégorie sont basées sur l'extraction des caractéristiques de bas niveau à partir de l'image. Ces caractéristiques sont représentées comme des descripteurs exploités par les classificateurs. Cette catégorie est très robuste, non seulement elle est facile à déployer, mais elle donne également des taux de reconnaissance très élevés et en particulier pour les variations intra-classe, telles que l'expression, la luminance, l'occlusion et les variations de pose. En conséquence, les approches basées sur l'extraction des caractéristiques ont été largement étudié et son utilisation est très avantageuse, notamment pour la reconnaissance faciale (Lei et al., 2014; Zhang et al., 2015). Les descripteurs qui représentent les caractéristiques locales sont moins affectés par les changements d'apparence faciale, ce qui nous a encouragés à les utiliser dans notre approche. L'utilisation de ces descripteurs ne nécessite pas une de base de données de visages de grande taille ni une machine de calcul haute performance aussi appelé *High Performance Computing* (HPC) consommant trop d'énergie. Nous pouvons également mieux analyser et interpréter les résultats lorsque nous choisissons nous-mêmes les descripteurs.

LBP est un descripteur utilisé pour décrire la texture d'une image. Ce descripteur donne de meilleurs résultats en termes de vitesse, de performances et de discrimination. Le principe de ce descripteur est d'étudier la relation entre un pixel et ses voisins, ce qui permet de construire une description binaire autour de ce pixel. L'un des avantages de l'utilisation de ce descripteur est qu'il est peu sensible aux variations d'éclairage et d'échelle. Les applications de vision par ordinateur qui utilisent les techniques du descripteur LBP et ses variantes ont

montré de meilleures performances de classification de texture (Liu, Fieguth, Guo, Wang & Pietikäinen, 2017).

L'exploitation de ces applications a prouvé que les modèles binaires locaux sont simples, en termes d'informatique (mise en œuvre), et se sont révélés très efficaces. Les descripteurs sont capables de représenter des images d'une manière très puissante afin de les exploiter dans plusieurs tâches en vision par ordinateur (Shahidoleslam, 2014), y compris la reconnaissance faciale (Zhang, Chu, Xiang, Liao, & Li, 2007 ; Tan & Triggs, 2010 ; Ahonen, Hadid & Pietikäinen, 2006 ; Lee, Ho & Kriegman, 2005 ; Pillai et al.2018 ; Liu et al. 2016 ; Chakraborty, Singh et Chakraborty, 2018).

6 Approche proposée pour la reconnaissance faciale

6.1 Descripteur LBP

Comme indiqué dans la section précédente, le descripteur LBP peut être utilisé pour représenter efficacement une image. En effet, c'est une description de la texture qui est invariante aux changements de luminance. La description de la texture d'une image au moyen du LBP se fait à l'aide d'un histogramme. Ce dernier est basé sur une représentation binaire calculée sur une région de cette image.

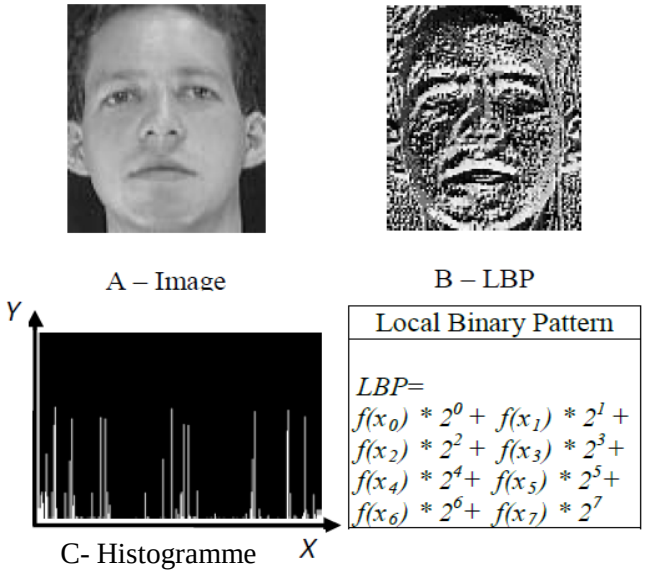


Figure 28. Une image, sa description en LBP et son histogramme

La figure 28-B montre le résultat de l'application du descripteur LBP de l'image représentée sur la figure 28-A. Chaque pixel de la figure 28-B est associé à une valeur comprise entre 0 et 255. La figure 28-C montre l'histogramme qui représente la distribution

des intensités de l'image (figure 28-B). Il s'agit d'une fonction discrète qui représente le nombre de pixels (axe Y) en fonction des valeurs comprises entre 0 et 255 (axe X).

La version originale du LBP (voir la figure 28) est calculée selon la formule suivante :

$$LBP(a,b) = \sum_{i=0}^{N-1} f(p_i - p_c) * 2^i \qquad (6.1)$$

Où *(a, b)* sont les coordonnées d'un pixel dans une image. p_c et p_i représentent les valeurs de gris du pixel central et de chacun de ses pixels voisins. *N* est le nombre de pixels autour du p_c. La fonction f (x) est définie comme suit:

$$f(x) = \begin{cases} 1 & x \geq 0 \\ 0 & Sinon \end{cases} \qquad (6.2)$$

6.2 Descripteur Honeycomb-LBP

6.2.1 Hexagones réguliers de nids d'abeilles

Les abeilles ouvrières construisent des ruches pour stocker le miel, le pollen, les œufs et les larves. Cette construction est fabriquée à partir de leur propre cire. La forme d'une cellule est détaillée dans la figure 29. Seules trois formes : triangle, carré ou hexadécimal, créent un tableau géométrique régulier de cellules identiques avec de simples sections transversales polygonales. Les auteurs Karihaloo, Zhang, & Wang, (2013) ont découvert que parmi ces formes, seule la structure hexagonale qui divise l'espace en utilisant un espace mural plus petit pour former une cellule en nid d'abeille. En effet, ils consomment moins de cire, cette structure hexagonale parfaite de nids d'abeilles, admirée depuis des millénaires comme un exemple de forme naturelle de motifs qui peut être exploitée dans différents domaines.

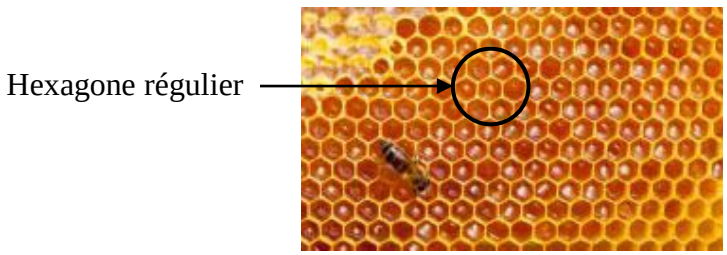
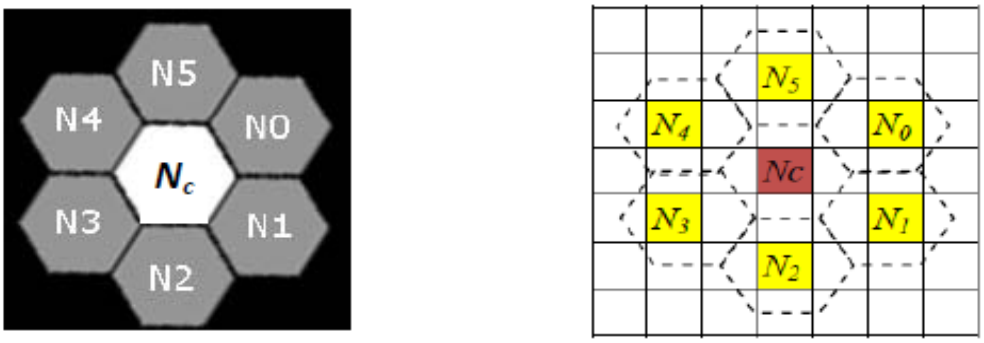


Figure 29. Hexagones réguliers de nids d'abeilles.

La topologie spatiale des cellules du nid d'abeilles est assez fascinante. Leur répartition spatiale des cellules permet de renforcer la résistance d'un élément tout en garantissant une

légèreté maximale qui forme une structure hexagonale. Aujourd'hui, cette structure en nid d'abeille est utilisée dans de nombreux secteurs industriels pour créer des structures résistantes, légères et optimales. Nous les trouvons dans l'automobile, la navigation aéronautique et d'autres secteurs. Cette structure est généralement en plastique, en aluminium ou en un autre matériau. Dans ce chapitre, nous montrons comment améliorer les performances de classification en choisissant une solution inspirée des cellules en nid d'abeilles, tout en adaptant le descripteur LBP à la représentation hexagonale des cellules d'une ruche.



A - Structure en nid d'abeille B - Description hexagonale des pixels

Figure 30. Présentation en nid d'abeille

Le descripteur que nous avons proposé est appellé *Honeycomb-Local Binary Pattern* (Ho-LBP) (Gafour, Berrabah & Mahmoudi, 2018). Il est défini par analogie avec la structure en nid d'abeille, ce qui signifie qu'il est basé sur une représentation hexagonale (figure 30-A). En d'autres termes, un pixel n'est affecté qu'à six voisins (figure 30-B) contrairement à sa représentation réelle dans une image (huit voisins). Il est à noter que l'image est convertie en niveau de gris avant de la représenter avec le descripteur Ho-LBP. La définition du descripteur Ho-LBP repose principalement sur deux étapes. La première consiste à déterminer les voisins d'un pixel et à calculer leurs valeurs respectives, et la seconde à associer à ce pixel une représentation binaire qui sera utilisée pour déterminer sa valeur descriptive. Pour désigner les voisins d'un pixel, nous utilisons le motif de structure hexagonale inspiré des cellules en nid d'abeille. En fait, un pixel n'a que six voisins.

6.2.2 Descripteurs Vertical Honeycomb –LBP et Horizontal Honeycomb –LBP

L'originalité de notre proposition réside dans le fait que notre descripteur prend en compte un maximum d'informations pour décrire un pixel. En effet, au lieu de ne prendre que les pixels en voisinage direct (ou en contact direct) avec le pixel central, notre descripteur prend

en compte un voisinage de trois niveaux pour décrire le pixel central (figure 31); ce qui signifie qu'un voisin est défini par l'ensemble de pixels P_{ij} d'une grille 3x3. Il est à noter que toutes les valeurs des pixels de cette grille sont utilisées pour calculer la valeur du pixel voisin ce qui prouve la richesse des informations utilisées pour caractériser un pixel. Nous utilisons l'équation 6.4 pour calculer la moyenne de cette grille.

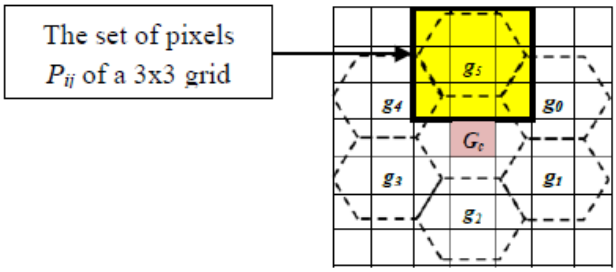


Figure 31. Voisinage du pixel (g5) dans la structure en nid d'abeille

Deux types de voisinage peuvent être envisagés : un voisinage vertical dénommé en anglais *Vertical Honeycomb-LBP* (VHo-LBP) (figure 32) et un voisinage horizontal dénommé en anglais *Horizontal Honeycomb-LBP* (HHo-LBP) (figure 33).

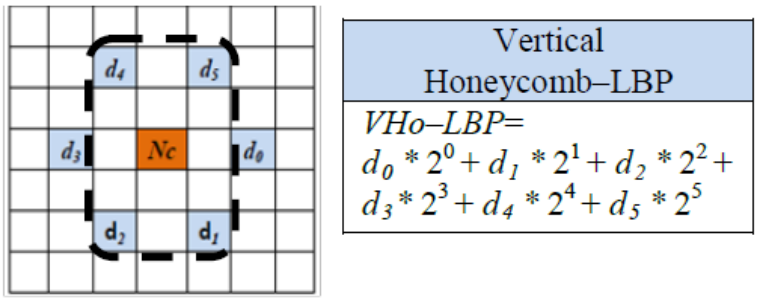


Figure 32. Descripteur Vertical Honeycomb–LBP

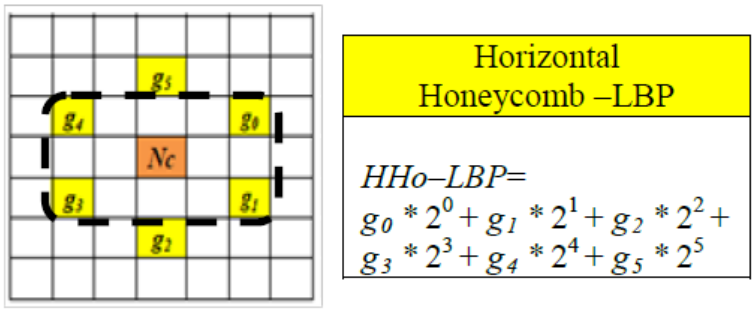
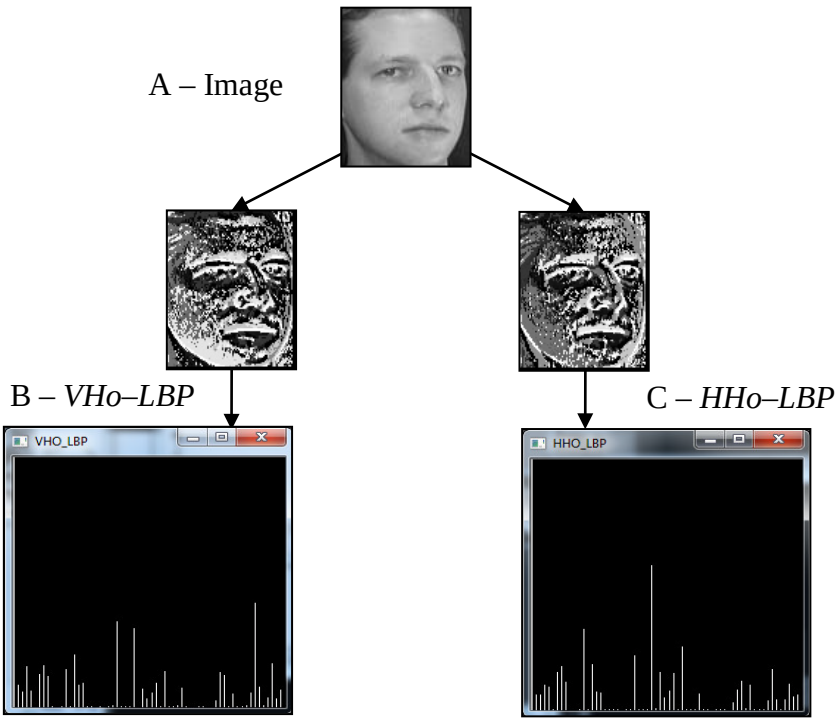


Figure 33. Descripteur Horizontal Honeycomb–LBP



(D) Histogramme de l'image représentant *VHo - LBP* (64 bins)

(E) Histogramme de l'image représentant *HHo-LBP* (64 bins)

Figure 34. Processus de calcul de l'histogramme de l'image associée à VHo-LBP et HHo-LBP

VHo-LBP est inspiré de deux paires de pixels (d_1, d_5) et (d_2, d_4) qui se présentent sous la forme de deux lignes verticales de l'image et qui sont symétriques par rapport au pixel central N_c. De plus, il n'y a qu'une seule paire de pixels (d_0, d_3) dans la ligne horizontale qui est symétrique par rapport au centre N_c. La figure 34-B montre l'application du descripteur VHo-LBP sur une image originale.

Nous pensons que l'application du descripteur VHo-LBP seul sur l'image produit une perte d'informations locales car la description est basée uniquement sur les six pixels qui forment la structure hexagonale voisine. Les six pixels voisins sont calculés par les huit pixels qui tournent autour d'eux. C'est pourquoi nous avons pensé d'utiliser le modèle HHo-LBP qui est complémentaire au descripteur VHo-LBP pour acquérir des caractéristiques beaucoup plus locales. HHo-LBP s'inspire des deux paires de pixels (g_0, g_4) et (g_1, g_3) qui se présentent sous la forme de deux lignes horizontales dans l'image et qui sont symétriques par rapport au pixel central N_c. De plus, il n'y a qu'une seule paire de pixels (g_2, g_5) dans la ligne verticale qui est symétrique par rapport au centre N_c. La figure 34-C montre l'application du descripteur HHo-LBP sur une image originale.

L'application des deux descripteurs VHo-LBP et HHo-LBP sur l'image originale donne deux images différentes (figure 34-B et figure 34-C). Il en résulte deux histogrammes différents (voir figure 34-D et figure 34-E). Nous pensons que cette approche enrichit la description sémantique des informations visuelles, capture les caractéristiques locales des visages et offre plus de pouvoir de discrimination par rapport à d'autres descripteurs de la littérature. Les descripteurs VHo – LBP et HHo – LBP sont proposés avec une taille réduite qui est de 64 cases. Ces descripteurs semblent plus efficaces et robustes pour la reconnaissance des visages dans des images prises à différents moments, en variant l'éclairage, les expressions faciales et les poses.

Pour représenter le voisinage d'un pixel central N_c, nous utilisons une relation d'ordre (0, 1, .., 5). Ainsi, les voisins d'un pixel central seront notés N_0, N_1, .., N_5. La valeur d'un pixel voisin N_k, k = 0 .. 5 est donnée en calculant la moyenne des valeurs des pixels P_{ij} de la grille 3x3 désignant ce voisin selon la formule de l'équation (6.3). Le deuxième point fort de notre proposition est de prendre les deux types de voisinage pour la description de l'image ce qui renforce l'étape de collecte d'informations sur les pixels de l'image.

$$d_k = \left(\left(\sum_{i-1}^{i+1} \sum_{j-1}^{j+1} (p_{ij}) \right) Div\ 9 \right) \tag{6.3}$$

$$k = 0..5,\ \ (i,j)\ \text{Coordonnées des pixels}\ \ d_k$$

$$g_k = \left(\left(\sum_{i-1}^{i+1} \sum_{j-1}^{j+1} (p_{ij}) \right) Div\ 9 \right) \tag{6.4}$$

$$k = 0..5,\ \ (i,j)\ \text{Coordonnées des pixels}\ \ g_k$$

Une fois les voisins d'un pixel déterminés et leurs valeurs respectives calculées, une valeur binaire (0 ou 1) est associée à chacun d'eux en fonction de sa valeur calculée respective pour VHo-LBP. En fait, ces valeurs sont transformées en valeurs binaires en utilisant la fonction des équations (6.5) et (6.6). Ainsi, une représentation binaire de six bits est donnée à chaque pixel de l'image.

$$f(D_k) = \begin{cases} 1, & d_k \geq D_c \\ 0 & sinon \end{cases} \tag{6.5}$$

$$VHo - LBP\ (a,b)_2 = \bigoplus_{k=0}^{k=5} f(D_k) \qquad (6.6)$$

Où VHo-LBP (a, b) est la valeur du pixel central (a, b) et f (D_k) sont les valeurs de ses six pixels voisins qui sont égales (0 ou 1).

⊕ Spécifie l'opérateur de concaténation des chaînes binaires.

Pour calculer les valeurs binaires du descripteur HHo-LBP, nous utilisons les équations (6.7) et (6.8).

$$f(G_k) = \begin{cases} 1, & g_k \geq G_c \\ 0 & sinon \end{cases} \qquad (6.7)$$

$$VHo - LBP\ (a,b)_2 = \bigoplus_{k=0}^{k=5} f(G_k) \qquad (6.8)$$

où HHo-LBP (a, b) est la valeur du pixel central (a, b) et f (G_k) sont les valeurs de ses six pixels voisins qui sont égales (0 ou 1).

⊕ Spécifie l'opérateur de concaténation des chaînes binaires.

Chaque représentation binaire associée à un pixel est transformée en une valeur décimale selon l'équation (6.9) pour VHo-LBP et l'équation (6.10) pour HHo-LBP. En fait, c'est la valeur descriptive VHo-LBP ou HHo-LBP du pixel (figure 35). Enfin, un histogramme de 64 (2^6) bins (cases) des étiquettes résultantes est utilisé comme descripteur de caractéristiques de l'image.

$$VHo - LBP(a,b)_{decimale} = \sum_{k=0}^{N-1} f(d_k) * 2^k \qquad (6.9)$$

$$k = 0..5,\ (a,b)\ \text{Coordonnées des pixelss}\ d_k$$

$$HHo - LBP(a,b)_{decimal} = \sum_{k=1}^{N-1} f(g_k) * 2^k \qquad (6.10)$$

$$k = 0..5,\ (a,b)\ \text{Coordonnées des pixels}\ g_k$$

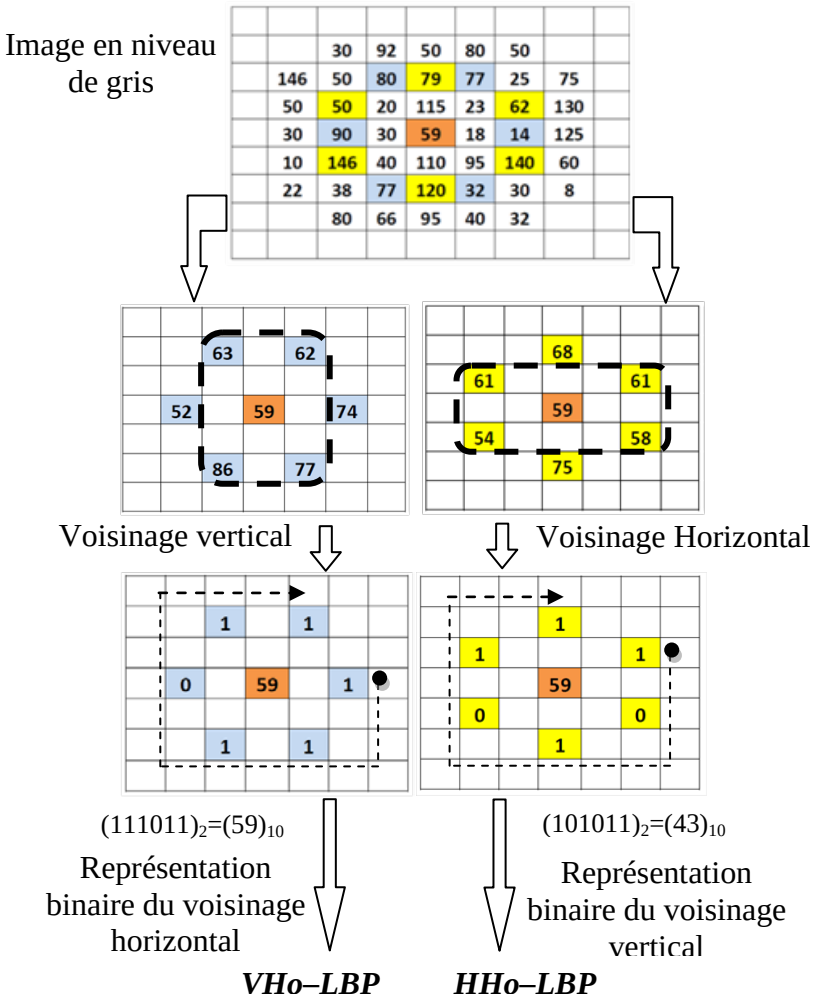


Figure 35. Illustration du schéma des codages VHo – LBP et HHo –LBP.

6.3 Méthodes de classification supervisées

Les méthodes de classification supervisées apprennent à effectuer une tâche particulière. L'apprentissage se fait par le biais de vecteurs saisis basés sur des détecteurs ou des descripteurs. L'objectif de la classification est principalement de définir des règles de classification des objets en classes à partir de variables qualitatives ou quantitatives caractérisant ces objets.

Parmi ces différentes méthodes d'apprentissage automatique, nous nous sommes concentrés sur les méthodes SVM, RF et K-NN que nous utiliserons dans notre approche, car elles sont reconnues par leurs performances de classification des images (Wang, Chen, Wu & Liu, 2017).

6.4 Amélioration de la précision de la reconnaissance faciale

Un descripteur joue un rôle très important dans la description d'une image. Il est exploité par des classificateurs pour classer ou reconnaître les images décrites par ce descripteur. Afin d'améliorer les performances de notre descripteur pour la classification des images, notamment pour la reconnaissance faciale nous avons proposé deux approches.

Avant de décrire ces approches, nous introduisons d'abord trois autres variantes de notre descripteur Ho – LBP qui sont : *Add Honeycomb – LBP* (AHo – LBP), *Or Honeycomb* – LBP (OHo – LBP) et *Xor Honeycomb – LBP* (XHo – LBP). Ces variantes sont le résultat de l'application respective des opérateurs logiques AND, OR et XOR composés sur les deux variantes (représentations binaires horizontales et verticales) HHo – LBP et VHo – LBP.

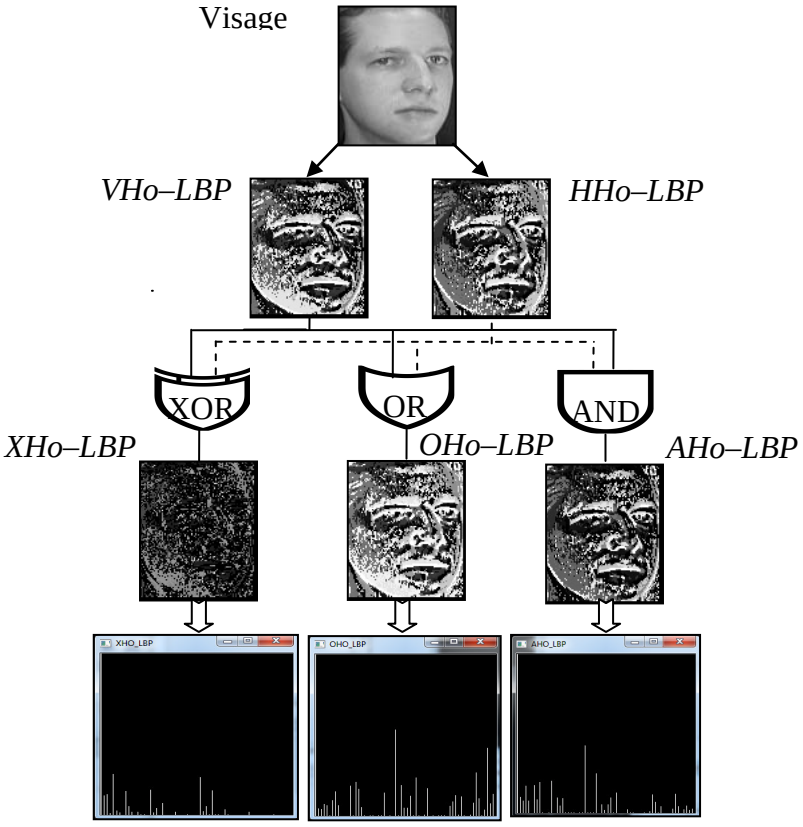


Figure 36. Opérations arithmétiques des bits XOR, OR et AND entre les deux images (VHo-LBP et HHo-LBP)

Le choix de ces opérateurs logiques n'est pas arbitraire. En effet, l'application de ces opérateurs sur les deux représentations binaires d'un pixel permet de donner une description distinctive à l'image (figure 36). La première approche consiste à utiliser une seule variante du descripteur Ho – LBP et un classificateur à la fois. En revanche, et dans la seconde

approche, toutes les variantes du Ho – LBP sont combinées pour être exploitées par chacun des classificateurs.

6.4.1 Première approche

Dans cette approche, toutes les images dans une base de données sont représentées à l'aide des variantes Ho-LBP proposées dans ce chapitre. Ensuite, chaque variante est exploitée par les trois classificateurs SVM, RF ou K-NN indépendamment pour l'apprentissage et les tests. Les résultats sont enregistrés pour décider de la meilleure variante du Ho-LBP. Le but principal de cette approche est d'évaluer la performance de chaque variante de notre descripteur avec chacun des classificateurs sur trois bases de données différentes (figure 37).

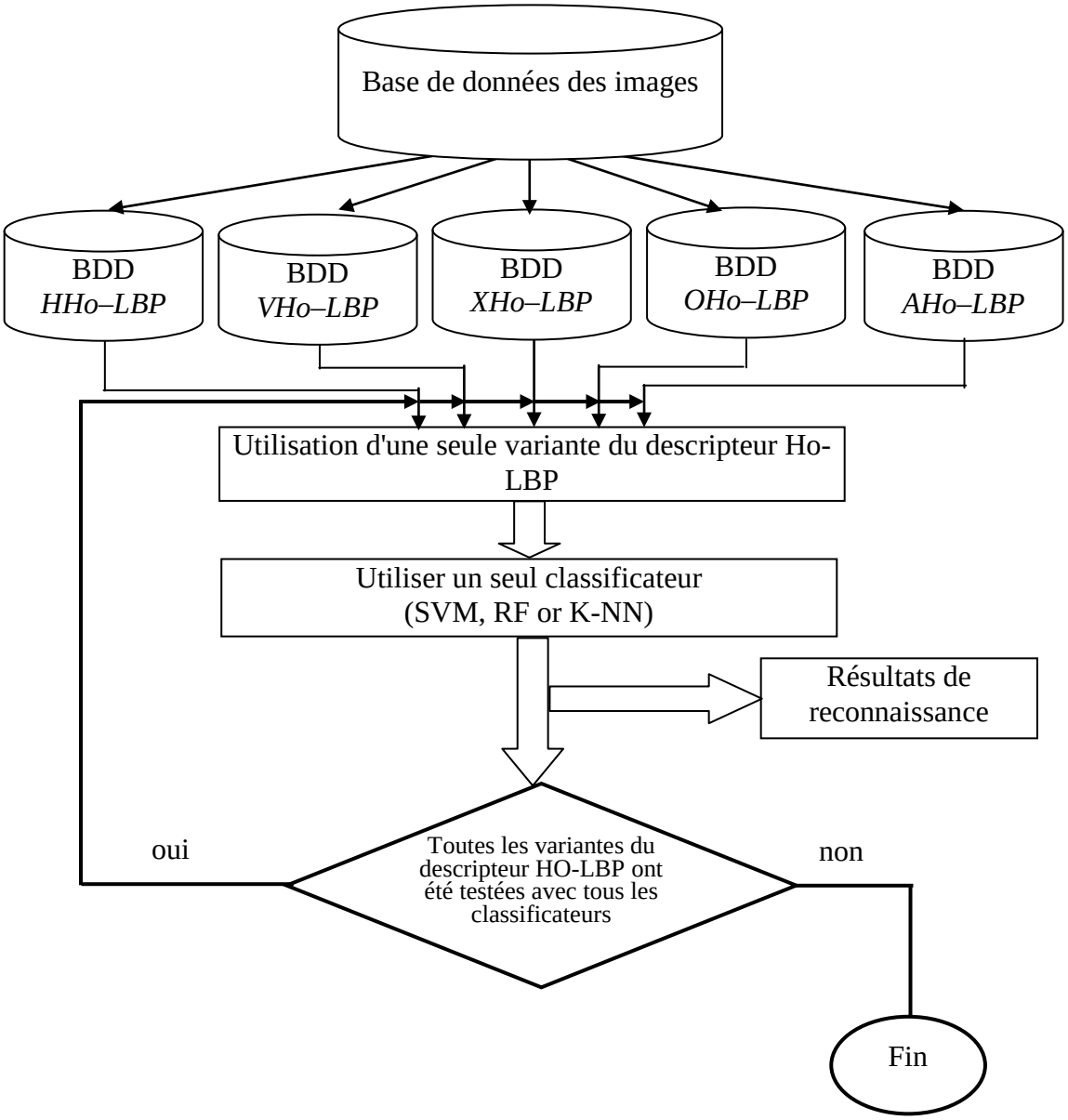


Figure 37. Modèle proposé de la première méthode

6.4.2 Deuxième approche

Dans cette approche, les résultats de la première approche sont récupérés. Ces résultats sont interprétés de telle manière que si l'image est correctement affectée par le classificateur à la classe correcte, la valeur "1" est affectée à la variante, du Ho-LBP, sinon "0".

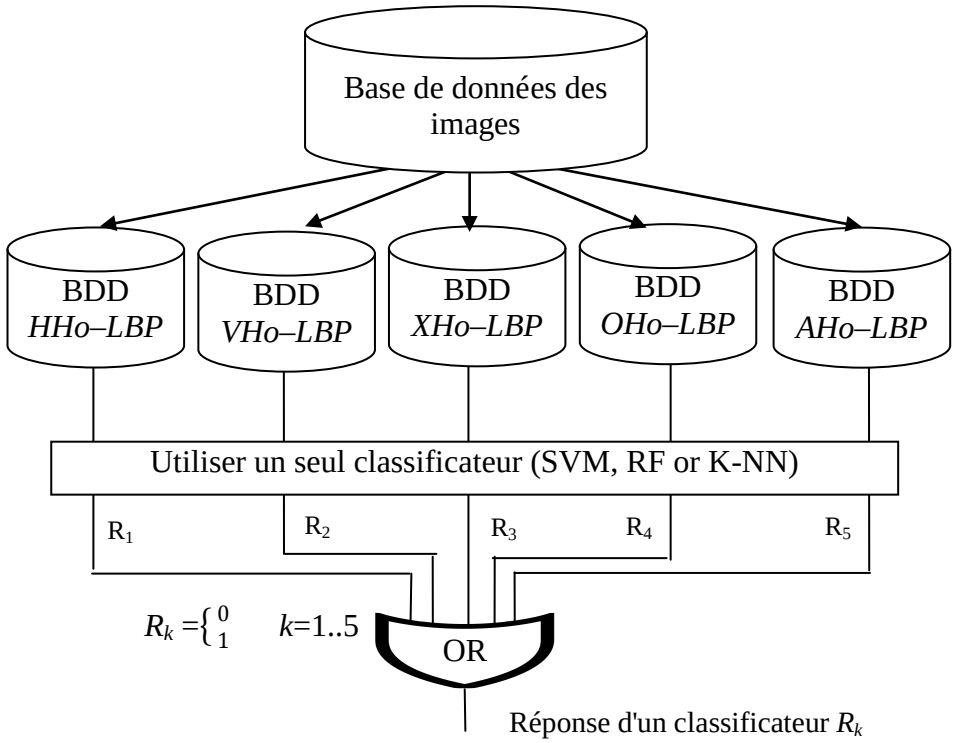


Figure 38. Modèle proposé pour la deuxième approche

Ainsi, les résultats de ce processus seront notés respectivement : R_1, R_2, R_3, R_4 et R_5 nombre de variantes du descripteur Ho-LBP (figure 38). La phase finale consiste à combiner les résultats de ces cinq variantes en appliquant l'opérateur logique OU selon l'équation (6.11). Cela améliorerait le taux de précision puisque les résultats du processus de reconnaissance des cinq variantes par classificateur sont pris en considération. Ce processus sera répété pour les trois classificateurs et les résultats seront enregistrés.

$$\underbrace{Final\ Result}_{\text{Résultat de la reconnaissance}} = \{R_1\ or\ R_2\ or\ R_3\ or R_4\ or\ R_5\} \qquad (6.11)$$

6.5 Expériences et résultats

Pour évaluer notre approche, nous avons utilisé des bases de données de référence. Nous rappelons que notre objectif principal dans ce chapitre est de parvenir à faire la reconnaissance faciale. ORL, Extended Yale-B et FERET sont les bases de données les plus

utilisés pour évaluer les travaux traitant des problèmes de la reconnaissance faciale. Ils contiennent des images représentant des visages. La base de données ORL contient des images dans différentes poses, obtenues par rotation verticale et horizontale et variantes à l'éclairage. Les bases de données Yale B et Extended Yale-B incluent de nombreux changements de luminance. Ces bases de données sont transformées à l'aide des variantes du descripteur Ho-LBP que nous avons proposées et qui sont ensuite exploitées par les trois classificateurs (RF, SVM et K-NN) pour l'apprentissage et les tests. Nous avons d'abord évalué les résultats expérimentaux de nos variantes, puis les avons comparées à d'autres variantes avancées de LBP qui sont BGLBP (Davarpanah et al., 2015), CSLBP (Heikkilä et al, 2006), CSLDP (Xue et al., 2011), XCSLBP (Silva et al., 2015) et CSSILTP (Wu et al., 2014). Les résultats sont évalués en calculant le taux de précision de reconnaissance à l'aide de l'équation (6.12).

$$\text{Précision } = \frac{TP}{TP + FP} \tag{6.12}$$

$$TP : True\ \ Positive$$
$$FP : False\ Positive$$

6.5.1 Bases de données

6.5.1.1 Base de données ORL

La base de données Olivetti Research Laboratory (ORL) stocke un ensemble des images représentant des visages. Cette base a été conçue par les laboratoires *AT & T* de l'Université de Cambridge en Angleterre. Il s'agit d'une base de données de référence pour les systèmes de reconnaissance automatique des visages. Il comprend 400 visages de 40 personnes différentes afin que chaque personne ait 10 images différentes de son visage. En d'autres termes, les images correspondant à la même personne sont prises à des moments différents, provoquant ainsi des différences d'expression lumineuse et faciale, pose des changements, porte des lunettes, etc. Pour se faire une idée de ces différences, des images du sujet S1 de cette base de données sont montrées dans la figure 39. Toutes les images sont redimensionnées à 100 × 100 pixels. Pour expérimenter nos approches sur cette base de données, 10 sujets sont choisis de manière aléatoire. Ensuite, sept images sont également prises au hasard de chaque sujet pour la phase d'apprentissage. Les trois images restantes sont utilisées pour la phase de test. À la fin de chaque expérience, les taux d'apprentissage et de reconnaissance des tests sont calculés selon les équations (6.13) et (6.14) respectivement.

Figure 39. Exemples des images du sujet S1 de la base de données ORL

$$Taux\ de\ reconnaissance\ (apprentissage) = \frac{nombre\ d'images\ correctement\ attribuées\ \text{à}\ la\ classe}{70} \quad (6.13)$$

$$Taux\ de\ reconnaissance\ (teste) = \frac{nombre\ d'images\ correctement\ attribuées\ \text{à}\ la\ classe}{30} \quad (6.14)$$

6.5.1.2 Base de données Extended Yale-B

Cette base de données d'images est une extension de la base de données Yale Face B créée par l'Université de Yale. Elle comprend 16128 images avec 28 sujets différents. Chaque sujet a 9 poses avec 64 éclairements différents par pose, selon l'angle entre la direction de la source lumineuse et l'axe central de la caméra. Les images de cette base de données ont été capturées dans différentes conditions d'éclairage. La figure 40 présente des images du sujet Yale B16 de la base de données Extended Yale B.

Figure 40. Quelques images du sujet Yale B16 avec plusieurs poses et différents changements de lumière.

Avant de faire nos expériences, tous les visages de cette base de données ont été réduits à la taille de 100 × 120 pixels. Il est à noter qu'à chaque expérience, 10 sujets sont choisis au hasard. Pour expérimenter notre approche, nous sélectionnons 70% des images de chaque sujet pour l'apprentissage. Pour faire le test, nous utilisons le reste des images de l'apprentissage de chaque sujet. À la fin de chaque expérience, les taux de reconnaissance (*train*) et (*test*) sont calculés par les équations (6.15) et (6.16).

$$Taux\ de\ reconnaissance\ (apprentissage) = \frac{nombre\ d'images\ correctement\ attribuées\ à\ la\ classe}{4032} \quad (6.15)$$

$$Taux\ de\ reconnaissance\ (teste) = \frac{nombre\ d'images\ correctement\ attribuées\ à\ la\ classe}{1728} \quad (6.16)$$

6.5.1.3 Base de données Feret

La base de données FERET est l'une des plus grandes bases de données pouvant être utilisées par les chercheurs pour effectuer leurs expériences. Elle contient 14051 images de 1199 sujets. Les images de cette base de données se caractérisent par des variations de poses, d'expressions faciales et d'éclairage. Cette base de données est composée de plusieurs ensembles tels que Fa (expression faciale régulière), Fb (alternative pour évaluer notre approche ; nous avons choisi l'ensemble Fb car dans cet ensemble il y a suffisamment d'images par sujet "plus de onze images par sujet pour faire l'apprentissage"), Ba (série frontale "b"), Bj (expression alternative à Ba), Bk (éclairage différent de Ba) etc. Pour tester notre approche sur cette base de données, nous sélectionnons 10 sujets aléatoires de l'ensemble Fb. Toutes les images sont recadrées à 128 × 128 pixels. Des exemples des images sont présentés à la figure 41. À chaque fin d'une expérience, nous calculons le taux de reconnaissance (*train*) et (*test*) à l'aide des équations (6.16) et (6.17).

$$Taux\ de\ reconnaissance\ (apprentissage) = \frac{nombre\ d'images\ correctement\ attribuées\ à\ la\ classe}{80} \quad (6.16)$$

$$Taux\ de\ reconnaissance\ (teste) = \frac{nombre\ d'images\ correctement\ attribuées\ à\ la\ classe}{30} \quad (6.17)$$

Figure 41. Exemples des images de la base de données FERET

6.5.2 Première approche

6.5.2.1 Base de données ORL

Comme expliqué dans la section 6.4.1, la première approche consiste à tester chaque variante du descripteur Ho-LBP en utilisant les trois classificateurs RF, SVM et K-NN. Nous commençons notre évaluation des performances de ces variantes sur la base de données ORL. Les résultats expérimentaux des taux de précision dans le processus de reconnaissance faciale des phases d'apprentissage et de test sont présentés dans le tableau 10.

Les résultats du classificateur RF montrent que les meilleurs taux de reconnaissance faciale pendant la phase d'apprentissage sont donnés par les deux variantes VHo–LBP et AHo–LBP avec une précision de 92,9%, sachant que les autres variantes ne sont pas loin, avec un taux inférieur ou égal à 84,3% donné par les deux variantes HHo–LBP et XHo–LBP. De plus, le meilleur taux de reconnaissance faciale pendant la phase de test est donnée par la variante HHo–LBP avec une précision de 63,3%. Sinon, le faible taux de reconnaissance a été marqué par la variante XHo–LBP avec une précision de 20% sur l'ensemble de test.

Concernant le classificateur SVM, les résultats sont prometteurs notamment ceux donnés lors de la phase d'apprentissage avec un taux égal à 100%. Quant à la phase de test, le meilleur taux de reconnaissance est donné par la variante VHo–LBP avec une précision égale à 96,7% et le taux le plus bas est toujours donné par la variante XHo–LBP.

Enfin, et pour le classificateur K-NN, c'est la variante AHo–LBP qui a enregistré le meilleur taux de reconnaissance lors de la phase d'apprentissage avec une précision de 80% et

c'est la variante AHo–LBP qui donne le meilleur taux de reconnaissance lors de la phase de test avec une précision égale à 70%.

Pour conclure, il faut dire que les meilleurs résultats ont été obtenus avec la variante VHo–LBP. Sauf que, dans la phase de test, la variante HHo–LBP en utilisant le classificateur RF dépasse la variante VHo–LBP mais seulement de 3,3%. Pendant la phase d'apprentissage, les résultats obtenus avec la variante AHo–LBP en utilisant le classificateur K–NN dépassent d'environ 6% la variante VHo–LBP.

Nous avons comparé nos résultats à ceux des variantes du descripteur LBP. Il est à noter que la variante XCSLBP est la seule concurrente des variantes de notre descripteur Ho–LBP mais uniquement avec le classificateur RF. En effet, XCSLBP a enregistré de bons taux de reconnaissance.

	RF		**SVM**		**K-NN**	
descripteur	**Taux de reconnaissance (apprentissage)**	**Taux de reconnaissance (teste)**	**Taux de reconnaissance (apprentissage)**	**Taux de reconnaissance (teste)**	**Taux de reconnaissance (apprentissage)**	**Taux de reconnaissance (teste)**
BGLBP	92.9	33.3	100	73.3	64.3	40
CSLBP	78.6	63.7	95.7	73.3	48.6	53.3
CSLDP	91.4	56.7	100	76.7	61.4	50
XCSLBP	94.3	73.3	100	96.7	52.9	43.3
CSSILTP	87.1	50	100	66.7	61.4	50
VHo–LBP	92.9	60	100	96.7	75.7	70
HHo–LBP	84.3	63.3	100	90	74.3	63.3
XHo–LBP	84.3	20	100	56.7	44.3	26.7
OHo–LBP	88.6	53.3	100	90	75.7	60
AHo–LBP	92.9	60	100	90	80	60

Tableau 10. Taux de reconnaissance faciale (RR) sur différentes variantes de Ho – LBP sur la base de données ORL

Cependant, il n'y a pas de grand écart entre ces taux et ceux de nos variantes (1,4% pour la phase d'apprentissage par exemple). Sinon, en utilisant le classificateur SVM, les résultats donnés avec cette variante sont presque identiques aux nôtres. Quant au classificateur K-NN, toutes nos variantes (sauf XHo–LBP) dépassent la variante XCSLBP en phase d'apprentissage et de test.

6.5.2.2 Base de données Extended Yale B

Les résultats expérimentaux de notre approche de reconnaissance faciale pour les phases d'apprentissage et de test sur la base de données Extended Yale B sont présentés dans le tableau 11. En effet, ceux obtenus par les variantes du Ho–LBP à l'aide du classificateur RF montrent que la variante HHo–LBP donne le meilleur résultat par rapport aux autres variantes en atteignant 85,7% pendant la phase d'apprentissage. Pendant la phase de test, nous avons constaté que c'est toujours la variante HHo–LBP qui a enregistré la meilleure précision. Quant au taux de reconnaissance le plus bas, il est enregistré par la variante AHo–LBP avec une valeur de 67,1% pour la phase d'apprentissage et une valeur de 49% pour la phase de test. En ce qui concerne les variantes LBP, nous avons remarqué qu'elles enregistraient de bons taux de reconnaissance dans les deux phases d'apprentissage et de test avec le classificateur RF.

En ce qui concerne le classificateur SVM, le meilleur résultat a été donné par la variante OHo–LBP avec une précision de 93,6% pendant la phase d'apprentissage. On note une légère amélioration de 0,3% de la précision au cours de la même phase par rapport au résultat obtenu par le classificateur RF dans le cas de la variante XCSLBP. Quant à la phase de test, le meilleur taux de reconnaissance est donné par la variante HHo–LBP avec une précision de 73,7% et qui dépasse largement celle obtenue par la variante XCSLBP avec le classificateur RF durant cette même phase.

Enfin, et avec le classificateur K-NN, la variante OHo–LBP a enregistré le meilleur taux de reconnaissance pendant la phase d'apprentissage avec une précision de 76,9% qui dépasse toutes les variantes utilisées dans cette approche. En ce qui concerne la phase de test, c'est la variante OHo–LBP et cette fois CSLDP qui donnent toutes les deux les meilleurs taux de reconnaissance avec une précision de 56%.

	RF		SVM		K-NN	
Descripteur	**Taux de reconnaissance (apprentissage)**	**Taux de reconnaissance (teste)**	**Taux de reconnaissance (apprentissage)**	**Taux de reconnaissance (teste)**	**Taux de reconnaissance (apprentissage)**	**Taux de reconnaissance (teste)**
BGLBP	91.4	50.3	70.6	68.7	72.7	51
CSLBP	92.7	60.3	53.7	48.3	66.6	50.3
CSLDP	91.7	60	52.1	45.7	75.1	56
XCSLBP	93.3	64.7	84.1	64.3	75.4	49.7
CSSILTP	93.1	61.3	66.1	49	73.6	54
VHo–LBP	70.9	53	89	73	68.6	49.7
HHo–LBP	85.7	59	82.4	73.7	71.7	44.7
XHo–LBP	73.4	51.3	59.9	46.3	64	40
OHo–LBP	70.9	58	93.6	73.3	76.9	56
AHo–LBP	67.1	49	84	68.7	71.1	43

Tableau 11. Taux de reconnaissance faciale des classificateurs avec des descripteurs sur la base de données Extended Yale B

6.5.2.3 Base de données FERET

Les résultats expérimentaux et les taux de précision du processus de reconnaissance faciale pendant les phases d'apprentissage et de test sur la base de données Feret sont présentées dans le tableau 12.

Les résultats fournis par le classificateur RF sont motivants. En effet, le meilleur résultat est donné par la variante XHo–LBP avec une précision de 85,7% pendant la phase d'apprentissage, et en le comparant avec la précision de la variante XCSLBP, il reste un peu loin. Concernant la phase de test, les meilleures précisions sont obtenues par les variantes HHo–LBP et XCSLBP avec une précision de 63,3%.

Quant au classificateur SVM, les résultats sont très intéressants. En effet, pendant la phase d'apprentissage, la variante OHo–LBP a atteint un taux de précision de 100%. Pendant la phase de test, nous avons enregistré une amélioration de la précision donnée par nos variantes

de près de 3,4% par rapport aux résultats fournis par la variante XCSLBP avec le classificateur RF.

	RF		SVM		K-NN	
Descripteur	**Taux de reconnaissance (apprentissage)**	**Taux de reconnaissance (teste)**	**Taux de reconnaissance (apprentissage)**	**Taux de reconnaissance (teste)**	**Taux de reconnaissance (apprentissage)**	**Taux de reconnaissance (teste)**
BGLBP	92.9	56.7	68.6	66.7	41.4	46.7
CSLBP	84.3	50	72.9	50	41.4	40
CSLDP	90	53.3	98.6	60	50	43.3
XCSLBP	97.1	63.3	94.3	60	44.3	40
CSSILTP	88.6	50	100	43.3	54.3	26.7
VHo–LBP	78.6	50	97.1	76.7	48.6	50
HHo–LBP	82.9	63.3	94.3	70	50	46.7
XHo–LBP	85.7	40	90	56.7	45.7	43.3
OHo–LBP	80	50	100	70	54.3	50
AHo–LBP	72.9	50	98.6	73.3	50	50

Tableau 12. Taux de reconnaissance faciale des classificateurs avec des descripteurs sur la base de données Feret

Enfin, les résultats enregistrés par le classificateur K-NN sont modestes car le meilleur résultat obtenu par les variantes Ho-LBP ne dépasse pas 54,3% en phase d'apprentissage. Quant à la phase de test, les meilleurs taux de reconnaissance sont donnés par les variantes VHo–LBP, OHo–LBP et AHo–LBP avec une précision de 50% et le plus faible est donné par la variante CSSILTP.

6.5.3 Deuxième approche

6.5.3.1 Base de données ORL

Les résultats obtenus avec la deuxième approche sont très satisfaisants car les taux de reconnaissance enregistrés en combinant les résultats des cinq variantes sont meilleurs que ceux enregistrés pour chaque variante (Tableau 13).

	Taux de reconnaissance	Les variantes combinées Ho-LBP
RF	**apprentissage**	100
	teste	90
SVM	**apprentissage**	100
	teste	**96.7**
K-NN	**apprentissage**	90
	teste	70

Tableau 13. Taux de reconnaissance faciale des variantes de Ho-LBP combinées sur la base de données ORL

En effet, la combinaison des résultats des cinq variantes en appliquant entre elles l'opérateur logique OU permet de ne prendre en considération que les meilleurs résultats, c'est-à-dire les résultats où l'image est correctement attribuée à la bonne classe. Par exemple, le taux de reconnaissance a été considérablement amélioré pendant la phase d'apprentissage pour les classificateurs RF et K-NN avec des précisions égales à 100% et 90% respectivement. Il convient également de noter que le taux de reconnaissance pendant la phase de test pour le classificateur RF a également été considérablement amélioré avec une précision de 90%. De plus, nous n'avons pas enregistré d'amélioration du taux de reconnaissance pendant la phase d'apprentissage pour le classificateur SVM et pendant la phase de test pour les classificateurs SVM et K-NN, sachant que ces taux étaient déjà significatifs. Pour revenir à la variante XCSLBP dont les résultats concurrençaient avec les nôtres avec le classificateur RF, nous avons remarqué que les résultats de nos variantes combinées dépassent de loin ceux de cette variante.

6.5.3.2 Base de données Extended Yale B

Les résultats obtenus avec la seconde approche par RF sont très intéressants (Tableau 14). Les taux de reconnaissance faciale pour les deux phases, apprentissage et test, sont respectivement de 95% et 86,7%, ce qui montre une amélioration de la précision des deux phases de 10% et 27,7% respectivement, par rapport à la première approche. En ce qui concerne le classificateur SVM, nous constatons qu'il y a une amélioration de 5,7% et 15,3% dans les phases d'apprentissage et de test respectivement par rapport à la première approche.

	Taux de reconnaissance	Les variantes combinées Ho-LBP
RF	**apprentissage**	95
	teste	86.7
SVM	**apprentissage**	99.3
	teste	**89**
K-NN	**apprentissage**	86.9
	teste	65

Tableau 14. Comparaison du taux de reconnaissance faciale des trois classificateurs de la base Extended Yale B

Enfin, les résultats obtenus par le classificateur K-NN avec la deuxième approche ont donné 86,9% sur la phase d'apprentissage et 65% sur la phase de test. Ces résultats montrent clairement qu'il y a une amélioration significative par rapport à tous les résultats obtenus par les différentes variantes dans la première méthode.

Nous pouvons conclure que les résultats obtenus par la seconde approche sont très compétitifs par rapport aux meilleurs résultats donnés par la première approche, y compris ceux des variantes du descripteur LBP.

6.5.3.3 Base de données FERET

Les résultats, obtenus à l'aide du classificateur RF avec la deuxième approche sur la base de données Feret sont très satisfaisants (tableau 15).

	Taux de reconnaissance	Les variantes combinées Ho-LBP
RF	**apprentissage**	94.3
	teste	80
SVM	**apprentissage**	100
	teste	**90**
K-NN	**apprentissage**	65.7
	teste	63.3

Tableau 15. Comparaison du taux de reconnaissance faciale des trois classificateurs de la base de données Feret

Les taux de reconnaissance faciale durant les phases d'apprentissage et de test ont atteint respectivement 94,3% et 80%. Ces résultats dépassent ceux de la première approche, ce qui

montre qu'il y a une nette amélioration des précisions de 8,6% et 16,7% dans les deux phases. En ce qui concerne le classificateur SVM, nous constatons qu'il n'y a pas vraiment d'amélioration pendant la phase d'apprentissage puisque le taux de reconnaissance a atteint 100% dans la première approche. Cependant, nous avons enregistré une amélioration de 13,3% par rapport à la première approche pendant la phase de test. Enfin, les résultats obtenus par le classificateur K-NN dans la deuxième approche ont atteint respectivement 65,7% et 63,3% pendant la phase d'apprentissage et de test. Ces résultats ont montré une nette amélioration par rapport à tous les résultats obtenus par les différentes variantes dans la première approche.

7 Discussion

L'utilisation de descripteurs permet d'extraire des caractéristiques pour représenter les images. Dans le cas réel, ces descripteurs sont utilisés pour décrire des caractéristiques de visages de manière efficace pour la tâche de vérification et de reconnaissance de ces visages. Ces caractéristiques, optimisées sans perte d'informations, sont exploitées par les classificateurs pour apprendre et reconnaître des visages.

Dans notre première approche, nous n'avons pas besoin de bases de données de grande taille pour apprendre notre modèle. En d'autres termes, notre approche ne nécessite pas énormément de données de formation. L'exploitation du descripteur Ho – LBP à l'aide des trois classificateurs permet de donner des résultats très prometteurs pour le processus de reconnaissance dans plusieurs cas. Les meilleurs résultats sont donnés par le classificateur SVM et dans certains cas par RF. La deuxième approche proposée a clairement amélioré les résultats de la reconnaissance. Son avantage est de combiner les résultats des trois classificateurs sans avoir besoin de calculs supplémentaires. En fait, nous prenons simplement les résultats obtenus par nos variantes et les combinons pour améliorer les taux de reconnaissance faciale. Comme indiqué dans la section précédente, certaines variantes du descripteur LBP ont concurrencé les variantes du descripteur Ho – LBP dans quelques cas.

De plus, les résultats fournis par la deuxième approche ont largement dépassé ces variantes compétitives. Nous avons également remarqué que nos résultats sont meilleurs par rapport aux approches qui utilisent l'apprentissage profond pour la reconnaissance faciale telles que DeepID (Sun, Chen, Wang et Tang, 2014) et DeepFace (Taigman, Yang, Ranzato et Wolf, 2014). Ces approches fonctionnent bien, sauf qu'elles doivent utiliser de grands volumes de données (des millions d'images) pour apprendre un réseau profond. Ainsi, ce type

de formation doté d'un réseau qui nécessite une grande quantité de données d'entraînement ainsi que des millions de paramètres à régler, ce qui nécessite parfois l'utilisation d'un GPU, et cela n'est pas évident sur les applications pratiques car les appareils destinés à faire la reconnaissance d'image ne sont pas tous équipés de cette technologie.

8 Conclusion

Dans ce chapitre, nous avons présenté des nouvelles approches pour la reconnaissance faciale basée sur les variantes VHo – LBP et HHo-LBP du descripteur Ho – LBP. Nous avons exploité ces variantes à l'aide de trois classificateurs, SVM, RF et K-NN. Afin d'améliorer les résultats de la reconnaissance faciale, nous avons utilisé deux approches : la première utilise une seule variante du descripteur Ho – LBP et un seul classificateur. La seconde combine plusieurs variantes du descripteur Ho – LBP et un seul classificateur. Nous avons évalué notre approche sur les trois bases de données : ORL, Extended Yale B et Feret.

Les expériences de nos approches sur la base de données ORL montrent que le meilleur score était de 96,7%, qui a été obtenu en exploitant le classificateur SVM et les variantes descripteurs du Ho-LBP. Les résultats obtenus des expériences sur la base de données Extended Yale B avec le classificateur SVM ont obtenu une meilleure précision de 89% par rapport autres descripteurs testés. Les expériences sur la dernière base de données Feret montrent que le taux de reconnaissance obtenu est de 90% en utilisant le même classificateur SVM.

Les deux dernières bases de données (Extended Yale B et Feret) sont considérées comme très difficiles car les bases de données Extended Yale B diffèrent par le degré de variation de la pose, de l'éclairage et de l'expression présents dans leurs images de visage, car la base de données FERET comprend des milliers des images qui varient selon l'âge, le sexe et l'ethnicité. Les résultats expérimentaux obtenus par notre approche sont très prometteurs, en les comparants à ceux de nombreuses approches. Ainsi, nous montrons que notre approche est robuste pour résoudre le problème de la variation des poses et des changements de luminance des faces.

Notre approche prouve ses performances et surtout lorsque les changements d'éclairage sont rapides et importants. Elle permet d'augmenter la puissance de la description pour représenter les visages et en particulier dans les conditions réelles. Les résultats obtenus ont montré que les variantes verticale et horizontale de notre descripteur Ho-LBP sont

complémentaires. En effet, toute information perdue par l'une pourrait être récupérée par l'autre.

VII. Conclusion générale

La reconnaissance des formes est l'une des branches très importantes et activement recherchées de l'intelligence artificielle. C'est la science qui essaie de rendre des machines aussi intelligentes que les humaines pour reconnaître les objets et les classer dans les catégories souhaitées de manière simple et fiable. La reconnaissance des formes offre la solution à de nombreux problèmes de la classificaton et de la reconnaissance, tels que la reconnaissance vocale, la reconnaissance faciale, la classification des caractères manuscrits, le diagnostic médical, etc. La conception d'un système de reconnaissance de formes comprend essentiellement les deux étapes suivantes: la représentation des images et la classification. Dans la première étape, les données sont acquises et prétraitées. Cette étape est suivie par l'extraction des caractéristiques, la réduction des caractéristiques et le regroupement des caractéristiques. Dans l'étape de classification, le classificateur entraîné attribue l'objet en entré à l'une des classes des objets en fonction des caractéristiques calculées.

De nombreuses caractéristiques telles que les différences de texture, les différences de forme, les fluctuations de lumière et les changements de couleur dans l'image fournissent des informations utiles pour les algorithmes de classification. Le point le plus important pour la classification des images est de déterminer des caractéristiques correctes qui représentent au mieux l'image et contiennent moins de paramètres et de sélectionner l'algorithme de classification approprié pour ces caractéristiques. Avec les caractéristique spécifiées, l'image peut être exprimée de manière significative en utilisant moins de paramètres et en utilisant un descripteur de caractéristiques qui doit etre robustes, offrant un caractère distinctif et invariant aux transformations d'image courantes telles que l'occlusion, le point de vue de la caméra, l'échelle, la rotation et les changements d'éclairage.

L'objectif principal de cette thèse est de proposer des nouveaux descripteurs pour extraire efficacement les caractéristiques des images afin de les classifier ou de faire de la reconnaissance faciale. En effet, dans la première approche proposée, nous tentons de

participer à l'amélioration de la performance des systèmes de classification en'utilisant le SPM. SPM employé avec le BoF améliore nettement les résultats de la classification des images. Le principe de l'utilisation du SPM est de partitionner l'image en un ensemble de nivaux de même taille. Ce partitionnement peut aller jusqu'à trois niveaux dont le but est de capturer des informations spatiales dans l'image. Pour améliorer le système de classification des images utilisent SPM, nous proposons d'utiliser le descripteur A-KAZE qui permet de calculer les caractéristiques locales et globales de l'image. Le descripteur A-KAZE est le premier algorithmes ayant utilisé la diffusion non linéaire dans la détection de caractéristiques multi-échelles. Ce descripteur montre une meilleure répétabilité et de meilleures performances que d'autres algorithmes tels que BRISK, SIFT et SURF. Les BoF de chaque niveau de SPM seront concaténés pour construire un descripteur global qui représente l'image qui peut améliorer les performances finales de la classification des images. Les expériences réalisées sur Caltech 101, Caltech 256, 15 catégories de scènes, Pascal VOC 2007 avec la classfieur *Random forest* montrent l'efficacité de l'approche proposée. Bien que, les résultats experiementaux obtenus en utilisant les bases de données de taille modeste sont très encourageants.

En ce qui concerne la deuxième approche, nous avons présenté une nouvelle approche pour la reconnaissance faciale basée sur le descripteur LBP modifié. Nous avons proposé un descripteur d'image original qui prend en charge les pixels voisins sous la forme d'une structure en nid d'abeille par rapport au pixel central. Cette structure permet de construire une chaîne binaire de six bits pour calculer le vecteur de caractéristiques. Cette approche est basée sur les variantes VHo – LBP et HHo-LBP du descripteur Ho – LBP. Nous avons exploité ces variantes à l'aide de trois classificateurs, SVM, RF et K-NN. Afin d'améliorer les résultats de la reconnaissance faciale, nous avons utilisé deux méthodes: la première utilise une seule variante du descripteur Ho – LBP avec un seul classifieur et la seconde combine plusieurs variantes du descripteur Ho – LBP avec un seul classifieur. Nous avons évalué notre approche sur les trois ensembles de données: ORL, Extended Yale et Feret.

Les résultats expérimentaux obtenus par notre approche sont très prometteurs, en les comparant à ceux de nombreuses approches avancées. Ainsi, nous montrons que notre approche est robuste pour résoudre le problème de la variation des poses et des changements de luminance des faces. Notre approche prouve ses performances et surtout lorsque les changements d'éclairage sont rapides et importants. Elle permet d'augmenter la puissance de la description pour représenter les visages et, en particulier, dans des conditions réelles. Les résultats obtenus ont montré que les variantes verticale et horizontale de notre descripteur

Ho-LBP sont complémentaires. En effet, toute information perdue par l'une des variantes pourrait être récupérée par l'autre.

Nous prévoyons de tester nos descripteurs sur une grande base de données en exploitant des architectures de calcul hautes performances (locales ou cloud) (clusters, multi-CPU, multi-GPU, etc.) afin de garantir des temps de calcul réduits et une plus grande précision. Nous prévoyons également d'exploiter les algorithmes d'apprentissage profond pour fournir une comparaison complète entre les différents modèles de formation en termes de précision, de temps d'exécution et de taille des données etiquetées requises.

Références

Abdi. H. & Williams, L.J. (2010). Principal component analysis. *Wiley Interdisciplinary Reviews: Computational Statistics*. 2 (4), pp. 433–459.

Ahmed, H.O.A. Wong, M.L.D. Nandi, A.K. (2018). Intelligent condition monitoring method for bearing faults from highly compressed measurements using sparse over-complete features, *Mech. Syst. Sig. Process*. 99 pp. 459–477.

Ahonen, T., Hadid, A., & Pietikäinen, M. (2006). Face decription with local binary patterns: application to face recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 28 (12), pp. 2037–2041.

Alahi, A., Ortiz, R., & Vandergheynst, P. (2012). Freak: Fast retina keypoint, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 510–517.

Turing A.,(1950). Computing machinery and intelligence. Mind , *Oxford University Press*, 59(236), pp. 433-460.

Alcantarilla, P. F., Bartoli, A., & Davison, A. J. (2012). KAZE features. *European Conference on Computer Vision (ECCV*), 6 , pp. 214-227.

Alcantarilla, P. F., Nuevo, J. & Bartoli, A. (2013). Fast explicit diffusion for accelerated features in nonlinear scale spaces, *in British Machine Vision Conference (BMVC).*

Anderson, J.A., & Rosenfeld, E. (1988). Neurocomputing: Foundations of Research. Cambridge: *MIT Press*.

Andreopoulos, A., Tsotsos, J. K. (2013). 50 years of object recognition: directions forward. *Computer Vision and Image Understanding*. 117(8), pp. 827–891.

Anot, N., Singh, K.K. (2016). A Review On Biometrics And Face Recognition Techniques, *International Journal of Advanced Research*, 4(5), pp. 783-786.

Bahi, M., & Batouche, M. (2018). Deep Learning for Ligand-Based Virtual Screening in Drug Discovery, *3rd International Conference on Pattern Analysis and Intelligent Systems (PAIS)*, DOI: 10.1109/PAIS.2018.8598488

Banerjee, P., Bhunia, A. K., Bhattacharyya, A., Roy, P. P., & Murala, S. (2018). Local Neighborhood Intensity Pattern – A new texture feature descriptor for image retrieval. *Expert Systems with Applications*, 113, pp. 100-115.

Baum, L.E., & Petrie, T. (1966). Statistical Inference for Probabilistic Functions of Finite State Markov Chains, *The Annals of Mathematical Statistics*, 37(6), pp. 1554-1563. doi:10.1214/aoms/1177699147

Baum, L.E., and Petrie, T. (1966). Statistical inference for probabilistic functions of finite state Markov chains, *Ann. Math. Statist*, 37(6), pp. 1554-1563.

Bay, H., Tuytelaars, T., & Van Gool, L. (2006). Surf: Speeded up robust features, *European Conference on Computer Vision (ECCV)*, pp. 404– 417.

Beaudet, P., (1978). Rotationally invariant image operators. *In Proceding of 4th International. Joint Conference Pattern Recogntion*, pp. 579–583.

Benjelloun, F., & Ait Lahcen, A. (2015). Big data security: challenges, recommendations and solutions Handbook of Research on Security Considerations in Cloud Computing. *IGI Global*, pp. 301-313. DOI: 10.4018/978-1-4666-8387-7.ch014.

Borgi, M.A., Labate, D., El'Arbi, M., Amar, C.B. (2013). Shearlet network-based sparse coding augmented by facial texture features for face recognition. *In Image Analysis and Processing—ICIAP 2013, Proceedings of the 17th International Conference, Naples, Italy, 9–13 September 2013; Petrosino, A., Ed.; Lecture Notes in Computer Science; Springer: Berlin/Heidelberg, Germany, 8157*, pp. 611–620.

Bosch, A., Zisserman, A., & Munoz, X. (2007). Representing shape with a spatial pyramid kernel, *Proceedings of the 6th ACM International Conference on IMAGE and Video Retrieval*, pp.401-408.

Boser, B., Guyon, I., & Vapnik, V. (1992). A training algorithm for optimal margin classifiers. *Proceedings of the fifth annual workshop on Computational learning theory COLT '92*, pp. 144–152. doi.org/10.1145/130385.130401

Breiman, L. (2001). Random Forests. *Machine Learning*. 45, pp. 5–32.

Breiman, L., Friedman, J.H., Olshen, R.A., Stone, C.J. (1984). Classification and Regression Trees, *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* , 1(1), pp. 14 -23. Doi: 10.1002/widm.8

Bridle, J. S. (1990). Alpha-nets: a recurrent 'neural'network architecture with a hidden Markov model interpretation. *Speech Communication*, 9(1), pp. 83-92.

Burges, C. J. (1998). A Tutorial on Support Vector Machines for Pattern Recognition. Data Mining and Knowledge Discoveryn, 2, pp. 121–167. https://doi.org/10.1023/A: 1009715923555

Cai, J., Luo, J., Wang, S., & Yang, S. (2018). Feature selection in machine learning: A new perspective. *Neurocomputing*, 300, pp. 70–79. doi:10.1016/j.neucom.2017.11.077

Calonder, M., Lepetit, V., Strecha, C., & Fua, P. (2010). Brief: Binary robust independent elementary features. *European Conference on Computer Vision (ECCV),* 4, pp. 778–792.

Cevikalp, H., & Triggs, B. (2017). Visual Object Detection Using Cascades of Binary and One-Class Classifiers. *International Journal of Computer Vision*, 123(3), pp. 334-349.

Chakraborty, S., Singh, S. K., & Chakraborty, P. (2018). Local Gradient Hexa Pattern: A Descriptor for Face Recognition and Retrieval. *IEEE Transactions on Circuits and Systems for Video Technology*, 28 (1), pp.171 –180.

Chakraborty, S., Singh, S. K., & Chakraborty, P. (2017). Local directional gradient pattern: a local descriptor for face recognition. *Multimedia Tools and Applications*, 76(1), pp.1201–1216.

Chatterjee, S., Dey, N., Shi, F., Ashour A. S., Fong, S. J., & Sen, S. (2018). Clinical application of modified bag-of-features coupled with hybrid neural-based classifier in dengue

fever classification using gene expression data. *Medical & Biological Engineering & Computing*, 56(4), pp. 709–720.

Chen, J., Li, Q., Peng, Q., Wong, KH. (2015). Csift based locality-constrained linear coding for image classification, *Pattern Analysis and Applications*, 18(2), pp. 441–450.

Chen, J., Shan, S., He, C., Zhao, G., Pietikäinen, M., Chen, X., Gao, W. (2010). WLD: a robust local image descriptor. *IEEE Transaction Pattern Anal. Mach. Intell*. 32(9), pp. 1705–1720.

Choi, J., & Lee, J. (2019). EmbraceNet: A robust deep learning architecture for multimodal classification, *Information Fusion*, 51, pp. 259-270.

Cohen, P., Feigenbaum, E. (1982). The Handbook of Artificial Intelligence, 3, *Heuristech Press.*

Cortes, C., Vapnik, V. (1995). Support-vector networks. *Machine Learning*, 20, pp. 273–297. doi.org/10.1007/BF00994018.

Cover, T.M., Hart, P.E., (1967). Nearest neighbor pattern classification, *IEEE Transaction Information Theory*, 13, pp. 21–27.

Cristianini, N., & Shawe-Taylor, J. (2000). Frontmatter. In An Introduction to Support Vector Machines and Other Kernel-based Learning Methods. Cambridge: *Cambridge University Press*, (pp. I-Iv).

Csurka, G., Dance, C., Fan, L., Willamowski, J., & Bray, C. (2004). Visual categorization with bags of keypoints, *In Workshop on Statistical Learning in Computer Vision.*

Dalal, N., Triggs, B. (2005). Histograms of oriented gradients for human detection, *IEEE Computer Society Conference on Computer Vision and Pattern Recognition,*1, pp. 886-893, 2005.

Davarpanah, S. H., Khalid, F., Abdullah, L. N., & Golchin, M. (2015). A texture descriptor: BackGround Local Binary Pattern (BGLBP). *Multimedia Tools & Applications*, 75(11), pp. 6549-6568.

Davies, E. (2005). Machine Vision, third ed., Signal Processing and its Applications, *Morgan Kaufmann, Burlington*. doi:https://doi.org/10.1016/B978-0-12-206093-9.

Ding, C., & Tao, D. (2015). Robust face recognition via multimodal deep face representation. *IEEE Transactions on Multimedia*. 17 (11), pp. 2049– 2058. doi:10.1109/TMM.2015.2477042.

Donoho, D., Elad, M., & Temlyakov, V. (2006). Stable recovery of sparse over complete representations in the presence of noise, *IEEE Transactions on Information Theory* , 52(1), pp. 6–18.

Drif, N., & Ameur, Z. (2016). Detection of Facial Feature Points Using an Autoumatic Method, *International Journal of Imaging and Robotics*. 16, (3), pp. 64-70.

Drucker, H., Burges, C.J., Kaufman, L., Smola, A.J., & Vapnik, V. (1996). Support Vector Regression Machines. *Advances in neural information processing systems,* 28(7), pp. 779-784.

Druzhkov, P. N., & Kustikova,V. D. (2016) A survey of deep learning methods and software tools for image classification and object detection. *Pattern Recognition and Image Analysis*, 26(1), pp. 9–15.

Duda, R.O., & Hart, P.E. (1973). Pattern classification and scene analysis. *A Wiley-Interscience publication.*

Dy J.G. (2012) Feature Selection in Unsupervised Learning. *In: Seel N.M. (eds) Encyclopedia of the Sciences of Learning*. Springer, Boston, MA.

Dy, J.G. (2008).Unsupervised feature selection. *Computational Methods of Feature Selection*, pp. 19-39.

Ekenel, H. & Sankur, B. (2004). Feature selection in the independent component subspace for face recognition, *Pattern Recognition Letters*, 25, 12, pp. 1377-1388.

El merabet, Y., Ruichek, Y., & El idrissi, A. (2019). Attractive-and-repulsive center-symmetric local binary patterns for texture classification. *Engineering Applications of Artificial Intelligence*, 78, pp.158-172.

Everingham, M., Gool, L., Williams, C., Winn, J., & Zisserman, A. (2007). The PASCAL Visual Object Classes Challenge 2007 (*VOC2007*) Results.

Fawcett, T. (2003). ROC graphs: Notes and practical considerations for data mining researchers. *Technical Report HPL-2003-4, HP Labs.*

Fei-Fei, L., & Perona, P. (2005). A Bayesian hierarchical model for learning natural scene categories, *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, 2, pp. 524-531, doi: 10.1109/CVPR.2005.16.

Fei-Fei, L., Fergu, R., & Perona, P. (2007). Learning generative visual models from few training examples: an incremental Bayesian approach tested on 101 object categories. *Computer Vision and Image Understanding*, 106(1), pp. 59-70.

Francois-Lavet, Vincent; et al. (2018). "An Introduction to Deep Reinforcement Learning". *Foundations and Trends in Machine Learning*. 11 (3–4): pp. 219–354. arXiv:1811.12560. Bibcode:2018arXiv181112560F. doi:10.1561/2200000071

Fukunaga, K. (1990). Introduction to Statistical Pattern Recognition. second ed., *New York: Academic Press.*

Fukunaga, K., Hostetler, L.D. (1975). k-nearest-neighbor Bayesrisk estimation, *IEEE Transaction Information Theory* ,21, pp. 285–293.

Gafour Y., Berrabah D. & Gafour A., (2020). A Novel Approach to Improve Face Recognition Process Using Automatic Learning. *Int. J. Comput. Vis. Image Process*. 10(1), pp. 42-66.

Gafour Y. & Berrabah D., (2020). New Approach to Improve the Classification Process of Multi-Class Objects. *Int. J. Organ. Collect. Intell.* 10(2). pp. 1-19.

Gafour Y. & Berrabah D., Benaissa M., (2019). Supervised multi-class object of image database based on a-kaze descriptor, International *Conference on Networking and Advanced Systems (ICNAS 2019).*

Gafour, Y., Berrabah, D., & Mahmoudi, S. A., (2018). Robust Facial Recognition Using Extended Local Binary Patterns. *International Conference on Cloud Computing Technologies and Applications,* Cloudtech 2018– Belgium.

Gafour Y. & Berrabah D., (2018). Optimize Features to Better Recognize Faces, *Artificial Intelligence and its Applications* (AIAP'18).

Geert J. S. Litjens, Thijs Kooi, Babak Ehteshami Bejnordi, Arnaud Arindra Adiyoso Setio, Francesco Ciompi, Mohsen Ghafoorian, Jeroen A. W. M. van der Laak, Bram van Ginneken, Clara I. Sánchez. (2017). A survey on deep learning in medical image analysis. *Medical Image Analysis*. 42, pp. 60 – 88. doi:10.1016/ j.media.2017 .07.005.

Gemert, J., Geusebroek, J., Veenman, C., & Smeulders, A. (2008). Kernel codebooks for scene categorization. *European Conference on Computer Vision (ECCV)*, 3, pp. 696-709.

Gemert, J., Veenman, C., Smeulders, A., & Geusebroek, J. (2010). Visual word ambiguity. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 32(7), pp. 1271-1283.

Gokhale, M., Cohen, J., Yoo, A., Miller, W. M., Jacob A. C., Ulmer, C., & Pearce, R. A. (2008). Hardware technologies for high-performance data-intensive computing. *IEEE Computer*. 41(4), pp. 60-68. DOI:10.1109/MC.2008.125.

Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., C. Bengio,Y. (2014). Generative Adversarial Networks. *NIPS'14: Proceedings of the 27th International Conference on Neural Information Processing Systems.*

Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep Learning. *MIT Press.*

Gottfredson, L. S. (1998). The general intelligence factor. *Scientific American Presents*, 9, pp. 24–30.

Graves, A., Mohamed, A., & Hinton, G. (2013). Speech recognition with deep recurrent neural networks. In Proc. *IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 6645-6649.

Graves, A., & Jaitly, N. (2014). Towards end-to-end speech recognition with recurrent neural networks. *In Proc. International Conference on Machine Learning,* pp. 1764-1772,

Griffin, G., Holub, A., Perona, P. (2007). Caltech-256 object category dataset, *Technical Report 7694, California Institute of Technology,* Retrieved from. http://authors.library.caltech. edu/7694.

Han, J., & Kamber, M. (2005). Data Mining: Concepts and Techniques, *2nd ed. San Mateo, CA, USA: Morgan Kaufmann,* pp. 285–378.

Harris, C., & Stephens, M. (1988). A Combined Corner and Edge Detector. *Alvey Vision Conference*, pp. 1–6.

Hashem, I. A. T., Anuar, N. B., Gani, A., Yaqoob, I., Xia, F., & Khan, S. U. (2016). MapReduce: Review and open challenges. *Scientometrics*, 109(1), pp. 389–422.

Hashem, I. A. T., Anuar, N. B., Marjani, M., Gani, A., Sangaiah, A. K., & Sakariyah, A. K. (2017). Multi-objectives cheduling of MapReduce jobs in big data processing. *Multimedia Tools and Applications*, 77(8),pp. 9979–9994.

Hassaballah, M., Ali, A. A., Alshazly, H., (2016). Image Features Detection, Description and Matching. Image Feature Detectors and Descriptors. *Foundations and Applications: Springer International Publishing (Verlag)*. DOI: 10.1007/978-3-319-28854-3_2.

Hastie, T., Tibshirani, R., Friedman, J., Franklin, J. (2001). The elements of statistical learning; data mining, *inference and prediction.* Springer Verlag, New York.

Heikkilä, M., Pietikäinen, M. , & Schmid, C. (2006). Description of interest regions with center-symmetric local binary patterns Indian Conference on Computer Vision. *Graphics & Image Processing*, pp. 58–69.

Hidaka, Akinori & Kurita, Takio. (2017). Consecutive Dimensionality Reduction by Canonical Correlation Analysis for Visualization of Convolutional Neural Networks. Proceedings of the ISCIE International Symposium on Stochastic Systems Theory and its Applications. 2017. pp. 160-167. 10.5687/sss.2017.160.

Hinton, G.E., & Salakhutdinov, R. (2006). Reducing the Dimensionality of Data with Neural Networks. *Science*, 313(5786), pp. 504-507.doi: 10.1126/science.1127647.

Hinton, G.E., Krizhevsky, A., Wang, S.D. (2011) Transforming Auto-Encoders. In: Honkela T., Duch W., Girolami M., Kaski S. (eds) Artificial Neural Networks and Machine Learning – ICANN 2011. ICANN 2011. *Lecture Notes in Computer Science*, 6791. Springer, Berlin, Heidelberg

Hodgkin A., Huxley A. (1952). A quantitative description of membrane current and its application to conduction and excitation in nerve. *The Journal of Physiology*. 117(4), pp. 500–44.

Hu, K., Niu, X., Liu, S., Zhang, Y., Cao, C., Xiao, F., Yang, W., Gao, X. (2019). Classification of melanoma based on feature similarity measurement for codebook learning in the bag-of-features model. *Biomedical Signal Processing and Control*, 51, pp. 200-209. DOI.org/10.1016/j.bspc.2019.02.018.

Jain, A. K., Dubes, R. C. (1988). Algorithms for clustering data. *Upper Saddle River, NJ, USA: Prentice-Hall, Inc.*

Jiang, Y., Ngo, C., & Yang, J. (2007). Towards optimal bag-of-features for object categorization and semantic video retrieval, *Proceedings of the 6th ACM international conference on Image and video retrieval, ACM,* pp. 494-501. DOI: 10.1145/1282280.1282352.

Jiang, Z., Zhang, G., & Davis, L. S. (2012). Submodular dictionary learning for sparse coding. *Computer Vision and Pattern Recognition*, pp. 3418-3425.

Kaifeng G., Gang M., Francesco P., Salvatore C.., Jingzhi T., Zenan H. (2020). Julia language in machine learning: Algorithms, applications, and open issues. Computer Science Review, 37, doi.org/10.1016/j.cosrev.2020.100254.

Karihaloo, B. L., Zhang, K., & Wang,J. (2013). Honeybee combs: how the circular cells transform into rounded hexagons. *J R Soc Interface 10:* 20130299.doi.org/10.1098 /rsif.2013.0299

Kaufman, L., & Rousseeuw, P. (1990). Finding Groups in Data: An Introduction to Cluster Analysis. *John Wiley & Sons.*

Ke, Y., & Sukthankar, R. (2004). Pca-sift: A more distinctive representation for local image descriptors. *Proceedings of IEEE computer society conference on Computer vision and pattern recognition*, pp. 506–513.

Kumar Y, & Jain, Y. (2013), Research Aspects Of Expert System. *International Journal Of Computing & Business Research*, pp. 23-29.

Kurfess, T. R. (Ed.). (2005). Robotics and automation handbook. *New York, NY: CRC Press.*

Lake, B., Ullman, T., Tenenbaum, J., & Gershman, S. (2017). Building machines that learn and think like people. *Behavioral and Brain Sciences*, 40, E253. doi:10.1017/S0140525X16001837

Lazebnik, S., Schmid, C., & Ponce, J. (2006). Beyond Bags of Features: Spatial Pyramid Matching For Recognizing Natural Scene Categories. *Computer Vision and Pattern Recognition*, 2, pp. 2169-2178.

Lazebnik, S., Schmid, C., Ponce, J. (2005). A sparse texture representation using local affine regions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 27(8), pp. 1265–1278.

Lecun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11), pp. 2278 – 2324.DOI: 10.1109/5.726791.

Lee, J., Kong, T., & Lee, K. (2019). Ensemble patch sparse coding: A feature learning method for classification of images with ambiguous edges. *Expert Systems with Application*, 124, pp. 1-12.

Lee, K. C., Ho, J., & Kriegman, D. (2005). Acquiring linear subspaces for face recognition under variable lighting. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 27 (5), pp. 684–698.

Lei, Z., Pietikainen, M., & Li, S. Z. (2014). Learning discriminant face descriptor, *Pattern Analysis and Machine Intelligence. IEEE Transactions*, 36 (2), pp. 289–302.

Leutenegger, S., Chli, M., & Siegwart, R. (2011). BRISK: binary robust invariant scalable keypoints. *In Proceedings of IEEE International Conference on Computer Vision*, pp. 2548–2555.

Leutenegger, S., Chli, M., & Siegwart, R. (2011). BRISK: binary robust invariant scalable keypoints. *In Proceedings of IEEE International Conference on Computer Vision*, pp. 2548–2555.

Levi, G. & Hassner, T. (2015). LATCH: Learned Arrangements of Three Patch Codes, arXiv preprint arXiv:1501.03719.

Liao, S., Zhao, G., Kellokumpu, V., Pietikainen, M., & Li, S. Z. (2010). Modeling pixel process with scale invariant local patterns for background subtraction in complex scenes. *IEEE Conference on Computer Vision and Pattern Recognitio*, pp. 1301-1306.

Lindeberg, T., & Gårding, J. (1997). Shape-adapted smoothing in estimation of 3-D shape cues from affine deformations of local 2-D brightness structure. *Image and Vision Computing*. 15 (6), pp. 415–434.

Lindeberg, T., (1998). Feature detection with automatic scale selection. *International Journal of Computer Vision*, 30(2), pp. 79–116.

Liou, C., Huang, J., Yang, W. (2008). Modeling word perception using the Elman network. *Neurocomputing*. 71 (16–18), pp. 3150-3157. doi:10.1016/j.neucom.2008.04.030.

Liu, L., Fieguth, P., Guo, Y., Wang, X, Pietikäinen. M., (2017). Local binary features for texture classification: Taxonomy and experimental study. *Pattern Recognition*, 62, pp. 135-160.

Liu, L., Fieguth, P., Zhao, G., Pietikäinen, M., & Hud, D. (2016). Extended local binary patterns for face recognition. *Information Sciences*, pp. 358–359, 56-72.

Liu, P., Guo, J., Chamnongthai, Ko., & Prasetyo, H. (2017). Fusion of color histogram and LBP-based features for texture image retrieval and classification. *Information Sciences*, 390, pp. 95-111.

Liu, S., Bai, X. (2012). Discriminative features for image classification and retrieval. *Pattern Recogn. Letters.* 33(6), pp. 744–751. DOI: 10.1109/ICIG.2011.149

Loncomilla, P., Ruiz-del-Solar, J., Martínez L. (2016). Object recognition using local invariant features for robotic applications: Asurvey, *Pattern Recognition*, 60, pp. 499–514.

Lou, X., Huang, D., Fan, L., & Xu, A. (2014). An image classification algorithm based on bag of visual words and multi-kernel learning, *Journal of Multimedia*, 9(2), pp. 269-277.

Lowe, D., (2004). Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 2(60), pp. 91–110.

Lowe, D. G., (1999). Object recognition from local scale-invariant features, *Proceedings of the International Conference on Computer Vision*, 2, pp. 1150–1157.

Mair, E., Hager, G. D., Burschka, D., Suppa, M., & Hirzinger, G. (2010). Adaptive and generic corner detection based on the accelerated segment test. *Proceedings of the 11th European Conference on Computer Vision: Part II, ECCV'10, Berlin, Heidelberg: Springer-Verlag*. pp. 183– 196.

Mairal, J., Bach, F., Ponce, J., Sapiro, G., & Zisserman, A. (2009). Supervised dictionary learning. *Advances in Neural Information Processing Systems*. 21, pp. 1033–1040.

Matas, J. Chum, O., Urban, M., and Pajdla, T. (2004). Robust wide-baseline stereo from maximally stable extremal regions. *Image and Vision Computing*. 22(10), pp. 761-767.

McCarthy, J. (2007).What is Artificial Intelligence?. *Stanford University.*

McLachlan, G.L., Basford, K.E., (1988). Mixture Models: Inference and Applications to Clustering. *Applied Statistics*. 38(2), *Marcel Dekker,* New York. DOI: 10.2307/2348072

Mejdoub, M.; Amar, C.B. (2013). Classification improvement of local feature vectors over the KNN algorithm. Multimedia. *Tools Application*., 64, pp. 197–218.

Mikolajczyk, K. & Schmid, C. (2005). A performance evaluation of local descriptors, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(10), pp. 1615–1630.

Mikolajczyk, K., Schmid, C. (2001). Indexing based on scale invariant interest points. *Proceedings. Eighth IEEE International Conference on Computer Vision*. DOI: 10.1109/ICCV.2001.937561

Mikolajczyk, K. & Schmid, C. (2002). An affine invariant interest point detector. *In Proceedings of European Conference on Computer Vision,* pp 128-142,Denmark.

Mikolajczyk, K., Schmid, C. (2004). Scale and Affine Invariant Interest Point Detectors. *International Journal of Computer Vision*, 60(1), pp. 63-86.

Mikolajczyk, K., Tuytelaars, T., Schmid, C., Zisserman, A., Matas, J., Schaffalitzky, F., Kadir T., & Van Gool L. (2005). A comparison of affine region detectors. *International Journal of Computer Vision*. 65, pp. 43–72.

Mokeev, A.V., Mokeev, V.V. (2015). Pattern recognition by means of linear discriminant analysis and the principal components analysis. *Pattern Recognit. Image Anal*. 25, pp. 685–691. doi.org/10.1134/S1054661815040185.

Moreels, P., Perona, P. (2005). Evaluation of features detectors and descriptors based on 3D objects. *IEEE International Conference on Computer Vision*. 1. pp. 800 – 807. DOI: 10.1109/ICCV.2005.89.

Murillo, A. C., Guerrero, J. J., Sagues, C. (2007). SURF features for efficient robot localization with omnidirectional images. *Proceedings - IEEE International Conference on Robotics and Automation. In: International Conference on Robotics and Automation*, pp. 3901–3907.

Nigam, S., Singh, R. & Misra, A.K. (2018). Efficient facial expression recognition using histogram of oriented gradients in wavelet domain. *Multimed Tools Application*, 77, pp. 28725–28747. doi.org/10.1007/s11042-018-6040-3

Noord, N. v., & Postma, E. (2017). Learning scale-variant and scale-invariant features for deep image classification, *Pattern Recognition,* 61, pp. 583-592.

Ojala, T., Pietikäinen, M., & Harwood, D. (1996). A comparative study of texture 420 measures with classification based on featured distributions. *Pattern recognition,* 29 (1), 51–59.

Ojala, T., Pietikäinen, M., & Maenpaa , T. (2002). Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI),* 24 (7),pp. 971–987.

Ojala, T., Pietikäinen, M., & Mäenpää, T. (2001). A Generalized Local Binary Pattern Operator for Multiresolution Gray Scale and Rotation Invariant Texture Classification. *Advances in Pattern Recognition (ICAPR),* pp. 397-406.

Oliva, A., & Torralba, A. (2001). Modeling the shape of the scene: A holistic representation of the spatial envelope, *International Journal of Computer Vision*, 42(3), pp.145-175.

Olshausen B. A., Field D. J. (1996), Emergence of Simple-Cell Receptive Field Properties by Learning a Sparse Code for Natural Images, *Nature*, 381,pp. 607-609.

Olshausen, B. A., Field, D. J. (1997), Sparse Coding with an Overcomplete Basis Set: A Strategy Employed by V1?, *Vision Research,* 37, pp. 3311-3325.

Pandya, J. M., Rathod, D., Jadav, J. J. (2013). A Survey of Face Recognition approach. *International Journal of Engineering Research and Applications (IJERA),* 3(1), pp.632-635

Parkhi, O. M., Vedaldi, A., & Zisserman, A. (2015). Deep face recognition. *British Machine Vision Conference.* 41, pp. 1-41.

Patel, T. S., Gurjwar, R. (2016). A Study on Feature Extraction Techniques for Image Mosaicing System. *International Journal of Advance Research and Innovative Ideas in Education,* 2(3), pp. 2395-4396.

Kaushik, N., Rawat, R., & Bhalla, A., (2016). A Brief Study of Different Feature Detector and Descriptor, *International Journal of Advanced Research in Computer and Communication Engineering,* 5(4), pp. 506-510.

Peng, Y., Li, L., Liu, S., Wang, X., & Li, J., (2018). Weighted constraint dictionary learning algorithm for image classification, *Pattern Recognition Letters*, DOI: 10.1016/j.patrec.2018.09.008.

Pérez, D. S., Bromberg, F., & Diaz, C. A. (2017). Image classification for detection of winter grapevine buds in natural conditions using scale-invariant features transform, bag of features and support vector machines. *Computers and Electronics in Agriculture*, 135, pp. 81-95.

Perona, P., Malik, J. (1990). Scale space and edge detection using anisotropic diffusion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12, pp. 629–639

Perronnin, F., Sánchez, J., & Mensink, T. (2010). Improving the Fisher Kernel for Large-Scale Image Classification. *European Conference on Computer Vision (ECCV),* 4, pp. 143-156.

Pietikäinen, M., Ojala, T., & Xu, Z. (2000). Rotation-invariant texture classification using feature distributions. *Pattern Recognition*, 33 (1), pp. 43–52.

Pillai, A., Soundrapandiyan, R., Satapathy, S., Satapathy, S. C., Jung, K., & Krishnan, R. (2018). Local diagonal extrema number pattern: A new feature descriptor for face recognition Future Generation. *Computer Systems*, 81, pp. 297-306.

Rahman, M. M., Rahman, S., Rahman, R., Hossain, B. M. M., & Shoyaib, M. (2017). DTCTH: a discriminative local pattern descriptor for image classification, *EURASIP Journal on Image and Video Processing*, 30.

Raina, R., Battle, A., Lee, H., Packer, B., & Ng, A.Y. (2007). Self-taught learning: transfer learning from unlabeled data. *In Proceedings of the 24 the International Conference on Machine Learning , ICML '07,* pp. 759–766.

Rosenblatt, F. (1958). The Perceptron: A Probabilistic Model for Information Storage and Organization in The Brain. *Psychological Review*, pp.65–386.

Rosenblatt, F.F. (1963). Principles of neurodynamics. perceptrons and the theory of brain mechanisms. *American Journal of Psychology*, 76, 705.

Rosten, E., Drummond, T. (2006). Machine learning for high speed corner detection. *European Conference on Computer Vision*, 1, pp. 430–443.

Rosten, E., Porter, R., Drummond, T. (2010). Faster and better: a machine learning approach to corner detection. *IEEE Transaction. Pattern Analysis and Machine Intelligence*, 32, pp. 105-119.

Roy, S. K., Chanda, B., Chaudhuri, B. B., Banerjee, S., Ghosh, D. K., & Dubey, S. R. (2018). Local directional ZigZag pattern: A rotation invariant descriptor for texture classification. *Pattern Recognition Letters*, 108, pp. 23-30.

Rumelhart, D. E., & McClelland, J. L. (1987). Information Processing in Dynamical Systems: Foundations of Harmony Theory, in Parallel Distributed Processing: Explorations in the Microstructure of Cognition: Foundations , *MITP*, pp.194-281.

Salleh, K. A., & Janczewski, L. (2016). Technological, Organizational And Environmental Security And Privacy Issues Of Big Data: A Literature Review, *Procedia Computer Science*, 100, 19-28. DOI.org/10.1016/j.procs.2016.09.119.

Sanin, A., Sanderson, C., Harandi, M. T. & Lovell, B. C. (2013). Spatiotemporal covariance descriptors for action and gesture recognition. *In IEEE Workshop on Applications of Computer Vision.*

Schaffalitzky F., Zisserman A. (2002) Multi-view Matching for Unordered Image Sets, or "How Do I Organize My Holiday Snaps*?". In: Heyden A., Sparr G., Nielsen M., Johansen P. (eds) Computer Vision — ECCV 2002. Lecture Notes in Computer Science*, 2350. Springer, Berlin, Heidelberg.

Schroff, F., Kalenichenko, D., & Philbin, J. (2015). Facenet: A unified embedding for face recognition and clustering. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 815–823.

Schwenker, F., Abbas, H., Gayar, N., Trentin, E. (eds) (2016) Artificial neural networks in pattern recognition. *In: 7th IAPR TC3 Workshop, ANNPR 2016, proceedings, Lecture Notes in Computer Science,* 9896. Springer, Berlin.

Shabbir, J., & Anwer, T. (2015). Artificial Intelligence and its Role in Near Future. *journal of latex class files.* 14(8), pp. 1-10.

Shahidoleslam, M. (2014). Local gradient pattern-A novel feature representation for facial expression recognition. *Journal of AI and Data Mining*. 2 (1), pp. 33-38.

Shao, H.D., Jiang, H.K., Wang, F.A., Wang, Y.N. (2017). Rolling bearing fault diagnosis using adaptive deep belief network with dual-tree complex wavelet packet, *Computers in Industry*, 96, pp. 27-39. doi.org/10.1016/j.compind.2018.01.005.

Shen, X., Liao, W., Choudhary, A., Memik, G., & Kandemir, M. (2003). A high-performance application data environment for large-scale scientific computations. *IEEE Transactions on Parallel and Distributed Systems*.14(12), pp. 1262–1274.

Silva,C., Bouwmans, T. & Frelicot, C. (2015). An eXtended Center-Symmetric Local Binary Pattern for Background Modeling and Subtraction in Videos. *Proceedings of the 10th International Conference on Computer Vision Theory and Applications*, 1, pp. 395-402.

Sluzek, A.S., Kozera, R. (2014). Improving Performances of BoW-based Image Retrieval by Using Contextual Keypoint Descriptors. *Proceedings of the International Conference on Machine Vision and Machine Learning Prague*, 139, pp. 14-15, Czech Republic.

Smith, S.M., Brady, J.M. (1997). A new approach to low level image processing. *International Journal of Computer Vision*, 23(1), pp. 45–78.

Sun, Y., Chen, Y., Wang, X., & Tang, X. (2014). Deep learning face representation by joint identification-verification. *In Neural Information Processing Systems (NeurIPS)*, pp. 1988– 1996.

Sun, Y., Liang, D., Wang, X., & Tang, X. (2015). Deepid3: Face recognition with very deep neural networks. *The Computing Research Repository (CoRR)* , arXiv:abs/1502.00873.

Sutton, R.S. (1988). Learning to predict by the method of temporal differences. *Machine Learning*, 3, pp. 9-44.

Taigman, Y., Yang, M., Ranzato, M., & Wolf, L. (2014). Deepface: Closing the gap to human-level performance in face verification. *Proceedings of the IEEE 715 Conference on Computer Vision and Pattern Recognition*, pp. 1701– 1708.

Takacs,G., Chandrasekhar,V., Tsai, S., Chen, D., Grzeszczuk, R.,Girod, B. (2013). Rotation-invariant fast features for large-scale recognition and real-time tracking. *Signal Processing Image Communication*. 28(4), pp. 334–344. DOI: 10.1016/j.image.2012.11.004

Tan, X., & Triggs, B. (2010). Enhanced local texture feature sets for face recognition under difficult lighting conditions. *IEEE Transactions on Image Processing*, 19 (6), pp. 1635–1650.

Tan, S. Y., Arshad, H., & Abdullah, A. (2019). Distinctive accuracy measurement of binary descriptors in mobile augmented reality. *PLOS ONE*, 14(1), DOI: 10.1371/journal.pone. 0207191.

Tiwari V., & Jain, S. C. (2019). Histopathological Image Classification Using Efficient Bag-of-Features and Whale Optimization Algorithm. *Proceedings of International Conference on Sustainable Computing in Science, Technology and Management (SUSCOM).*

Tripathi, K. P. (2011). A Comparative Study of Biometric Technologies with Reference to Human Interface. *International Journal of Computer Applications*, 14(5), pp. 10-15.

Tuytelaars, T., & Gool, L., V. (2004). Matching widely separated views based on affine invariant regions. *International Journal of Computer Vision*, 1(59), pp. 61-85.

Tuytelaars, T., Mikolajczyk, K. (2007). Local Invariant Feature Detectors: A Survey. *Computer Graphics and Vision*. 3(3), pp. 177–280. DOI: 10.1561/0600000017

Uchida, Y. (2016). Local feature detectors descriptors and image representations: A survey, *arXiv:1607.08368*.

van Otterlo, M., Wiering, M. (2012). Reinforcement learning and markov decision processes. Reinforcement Learning. Adaptation, *Learning, and Optimization*. 12. pp. 3–42. doi:10.1007/978-3-642-27645-3_1.

Vapnik, V. (1998). Statistical Learning Theory. *Wiley, New York*

Vapnik, V. N. (2000). The Nature of Statistical Learning Theory. *2nd ed., New York: Springer Press* doi:10.1007/978-1-4757-3264-1

Vinciarelli, A., Esposito, A., André, E., Bonin, F., Chetouani, M., Cohn, J. F., Cristani, M., Fuhrmann, F., Gilmartin, E., Hammal, Z., Heylen, D., Kaiser, R., Koutsombogera, M., Potamianos, A., Renals, S., Riccardi, G., & Ali Salah A. (2015). Open challenges in modelling, analysis and synthesis of human behaviour in human–human and human–machine interactions, *Cognitive Computation*, 7(4), pp. 397–413.

Viola, P., Jones, M.J. (2004). Robust Real-Time Face Detection. *International Journal of Computer Vision*, 57, pp. 137–154. doi.org/10.1023/B:VISI.0000013087.49260.fb

Wang, J., Yang, J., Yu, K., Lv, F., Huang, T., Gong, Y. (2010). Locality-Constrained Linear Coding For Image Classification. *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3360-3367.

Wang, K., Chen, Z., Wu, Q.M. J., & Liu, C. (2017). Illumination and pose variable face recognition via adaptively weighted ULBP_MHOG and WSRC. *Signal Processing: Image Commu*nication, 58, pp. 175–186.

Wen, Y., Zhang, K., Li, Z., & Qiao, Y. (2016). A discriminative feature learning approach for deep face recognition. *European Conference on Computer Vision*, pp. 499–515.

Wold, S., Esbensen, K., & Geladi, P. (1987). Principal component analysis. *Chemometrics and Intelligent Laboratory Systems*, 2(1-3), pp. 37–52. doi:10.1016/0169-7439(87)80084-9

Wright, J., Yang, A., Ganesh, A., Sastry, S., & Ma, Y. (2009). Robust face recognition via sparse representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(2), pp. 210–227.

Wu, H., Liu, N., Luo, X., Su, J., & Chen, L., (2014). Real-time background subtraction-based video surveillance of people by integrating local texture patterns. *Signal, Image and Video Processing*, 8(4), pp. 665-676.

Xu, Y., Li, Z., Zhang, B., Yang, J., & You, J. (2017). Sample diversity, representation effectiveness and robust dictionary learning for face recognition, *Information Sciences*, 375, pp. 171-182.

Xu, Y., Zhu, Q., Fan, Z., Zhang, D., Mi, J., & Lai, Z. (2013). Using the idea of the sparse representation to perform coarse-to-fine face recognition, *Information Sciences*, 238(20), pp. 138-148.

Xue, G., Song, L., Sun, J., & Wu, M. (2011). Hybrid center-symmetric local pattern for dynamic background subtraction. *International Conference on Multimedia and Expo (ICME)*, pp. 1–6.

Xue, G., Sun, J., & Song, L. (2010). Dynamic background subtraction based on spatial extended center-symmetric local binary pattern. *In IEEE International Conference on Multimedia*. DOI:10.1109/ICME.2010.5582601.

Yali, P., Shiganga, L., Xili, W., & Xiaojun, W., (2019). Joint local constraint and fisher discrimination based dictionary learning for image classification, *Neuro computing*, DOI.org/10.1016/j.neucom. 2019.05.103.

Yan, X. (2009), Linear Regression Analysis. *Theory and Computing, World Scientific*. doi.org/10.1142/6986

Yang, J., Li, Y., Tian, Y., Duan, L., & Gao, W. (2009). Group sensitive multiple kernel learning for object categorization. *IEEE International Conference on Computer Vision (ICCV)*, pp. 436-443.

Yang, J., Yu, K., Gong, Y., & Huang, T. S. (2009). Linear spatial pyramid matching using sparse coding for image classification, *Computer Vision and Pattern Recognition*, pp. 1794-1801.

Yang, X., & K.-T. Cheng. (2014). Local Difference Binary for Ultra-fast and Distinctive Feature Description. *IEEE Transactions on Software Engineering*. 36(1), pp. 188-194

Yap, P., Jiang, X., Kot, A. C. (2010). Two-dimensional polar harmonic transforms for invariant image representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 32(7), pp. 1259–1270.

Yuan, F., Shi, J., Xia, X., Zhang, L. , & Li, S. (2019). Encoding pairwise Hamming distances of Local Binary Patterns for visual smoke recognition. *Computer Vision and Image Understanding*, 178, pp. 43-53.

Zerdoumi, S., Sabri, A. Q. M., Kamsin, A., Hashem, I. A. T., Gani, A., Hakak, S. Al-garadi, M. A., & Chang, V. (2018). Image pattern recognition in big data: taxonomy and open challenges: survey. *Multimedia Tools Application*,77(8), pp. 10091-10121.

Zhang, H. (2004). The Optimality of Naive Bayes. *FLAIRS Conference.*

Zhang, H., Berg, A. C., Maire, M., Malik, J., (2006). Svm-knn: Discriminative nearest neighbor classification for visual category recognition. *In IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2126-2136.

Zhang, L., Chu, R., Xiang, S., Liao, S., & Li, S. (2007). Face detection based on multi-block lbp representation. *Proceedings Advances in Biometrics, International Conference, ICB 2007, Lecture Notes in Computer Science 4642*, pp. 11–18.

Zhang, Z., Wang, L., Zhu, Q., Chen, S.-K., & Chen, Y. (2015). Pose-invariant face 730 recognition using facial landmarks and weber local descriptor. *Knowledge- Based Systems*, 84, pp. 78–88.

Zheng, P., Zhao, Z., Gao, J., & Wu, X. (2018). A set-level joint sparse representation for image set classification. *Information Sciences,* 448-449, pp. 75-90. Doi.org/10.1016/j.ins. 2018.02.062.

Zhu, Q., Wang, Z., Mao, X., & Yang, Y. (2017). Spatial locality-preserving feature coding for image classification. *Applied Intelligence,* 47(1), pp. 148–157.