Centre Val de Loire UNIVERSITÉ Spirit of the Loire Valley

UNIVERSITÉ D'ORLÉANS

UNIVERSITE D'ORLEANS

ÉCOLE DOCTORALE MATHÉMATIQUES, INFORMATIQUE, PHYSIQUE THÉORIQUE ET INGÉNIERIE DES SYSTÈMES

LABORATOIRE : PRISME

Thèse présentée par :

Félix-Bazin POLLA DE NDJAMPA

soutenue le : **[18 décembre 2020]**

pour obtenir le grade de : **Docteur de l'Université d'Orléans**

Discipline/ Spécialité : **Mathématique-Informatique**

Reconnaissance d'actions à l'aide d'un imageur infrarouge basse résolution

THÈSE DIRIGÉE PAR :

Bruno EMILE	Maître de conférence HDR, Université d'Orléans, Laboratoire PRISME
Hélène LAURENT	Maître de conférence HDR, INSA Centre Val de Loire, Laboratoire PRISME

RAPPORTEURS :

Choubeila MAAOUI	Professeur des Universités, Université de Lorraine, Laboratoire LCOMS
Ludovic MACAIRE	Professeur des Universités, Université de Lille, Laboratoire CRIStAL

JURY :

Ludovic MACAIRE	Professeur des Universités, Laboratoire CRIStAL, Université de Lille, Président du jury
Choubeila MAAOUI	Professeure des Universités, Laboratoire LCOMS, Université de Lorraine
Yannick BENEZETH	Maître de conférences, Université de Bourgogne, Laboratoire ImViA
Bruno EMILE	Maître de conférence HDR, Université d'Orléans, Laboratoire PRISME
Hélène LAURENT	Maître de conférence HDR, INSA Centre Val de Loire, Laboratoire PRISME

Remerciements

Ce travail de recherche n'aurait pas été possible sans le soutien d'un grand nombre de personnes. Ce mémoire est l'occasion pour moi de tous les remercier.

J'adresse mes remerciements à Monsieur Azeddine KOURTA, directeur Laboratoire Pluridisciplinaire de Recherche en Ingénierie des Systèmes, Mécanique, Énergétique (PRISME) pour m'avoir accueilli au laboratoire.

J'exprime ma profonde reconnaissance tout spécialement à mes deux directeurs de thèse Monsieur Bruno EMILE et Madame Hélène LAURENT pour m'avoir encadré, pour leurs conseils inestimables, pour leur disponibilité et enfin d'avoir fait tout leur possible pour que cette thèse se passe dans les meilleures conditions. Ça a été un honneur pour moi de vous avoir eu comme directeurs de thèse.

J'adresse mes profonds remerciements aux professeurs Choubeila MAAOUI et Ludovic MACAIRE pour l'honneur qu'ils m'ont fait en acceptant d'être rapporteur de mon travail et pour leurs remarques qui ont permis d'améliorer la qualité de ce mémoire. Je remercie Monsieur Yannick BENEZETH de me faire l'honneur d'être examinateur de ma thèse.

Je suis reconnaissant avec BPI France, les Conseils régionaux du Limousin et de Rhône-Alpes avec FEDER, le Conseil général de l'Isre et la ville de Bourges, Bourges Plus, pour avoir apporté leur soutien financier au projet FUI CoCAPs et donc le financement de ma thèse.

Je remercie les chercheurs et doctorants de l'Axe Image Vision du laboratoire PRISME pour leur gentillesse et les bons moments que j'ai partagés avec eux. Je remercie madame Laure SPINA pour son aide précieuse et pour les démarches administratives. Bien sûr, je n'oublie pas toute l'équipe enseignante et administrative de l'IUT de l'Indre pour leur amabilité.

Je dédie ce manuscrit à ma famille pour leur encouragements et leur soutien quotidien et sans faille.

Je remercie tous mes amis pour leur conseil et soutien moral.

Je remercie la société FittingBox pour m'avoir accordé du temps pour travailler sur mon mémoire malgré le contrat qui nous lie.

Sommaire

Liste des figures

Liste des tableaux

Chapitre 1

Introduction

Sommaire

De nos jours, tout un chacun dispose d'un ensemble d'appareils (capteurs, ordinateurs, tablettes, caméras vidéo, téléphones portables) capables d'enregistrer, de produire, de stocker et de partager des informations telles que des images, des vidéos ou toute sorte de données numériques. Ces informations sont de plus en plus disponibles dans nos vies et personne n'imagine le monde sans cela. Les capacités de traitement et de stockage augmentant, les vidéos sont de plus en plus couramment utilisées et remplacent les images. Elles ont l'avantage de permettre une meilleure compréhension de la scène observée là où l'utilisation d'une seule image ne donne qu'une information instantanée et peut même parfois induire l'observateur en erreur. Les caméras vidéos sont aujourd'hui utilisées presque partout : villes, lieux de travail, maisons, aéroports, commerces, métro, écoles, hôpitaux, banques... et touchent la sphère publique comme la sphère privée. Elles sont devenues des éléments indissociables de notre quotidien. Avec un nombre croissant de vidéos disponibles, un accès plus grand et plus facile à celles-ci, le besoin d'une compréhension automatique des contenus vidéos augmente également. Le développement de systèmes automatiques intelligents, capables d'analyser, de reconnaître ce qui se déroulent dans une vidéo et d'interpréter le contenu visuel d'une scène, s'affirme donc comme une nécessité afin de palier les limites d'une analyse humaine. L'axe de recherche capable de réaliser cela est la vision par ordinateur. L'un des objectifs majeurs de cette thématique est de reconnaître et de comprendre le mouvement humain et notamment de permettre la classification des activités humaines.

1.1 Contexte général

Bien que de multiples domaines de la reconnaissance d'activités aient été étudiés, les différents auteurs ne se sont pas toujours accordés sur une taxonomie générale et complète du domaine.

Néanmoins, toutes ou presque font des distinctions entre les actions en fonction de leurs complexités intrinsèques.

Une action est définie comme étant le mouvement élémentaire d'une personne, comme marcher, courir, s'asseoir, tourner sur soi, etc... On définira une activité comme étant une combinaison qui implique plusieurs actions se déroulant sur une longue durée. Par exemple, l'activité "se coucher" consistera à effectuer les actions s'asseoir sur le lit et s'allonger. L'activité peut être aussi définie en fonction du nombre de personnes agissant ensemble, par exemple "être en réunion" ou "défiler".

À partir des définitions précédentes nous plaçons nos travaux dans le cadre de la reconnaissance d'actions. Nous focaliserons donc la reconnaissance sur un individu indépendamment de ses interactions éventuelles.

La reconnaissance d'actions peut intervenir dans différents domaines, englobant les applications industrielles (surveillance, maintenance de site, formation), les applications médicales (maintien à domicile ou assistance aux personnes âgées), les jeux vidéos, l'analyse du mouvement sportif... Chacun de ses domaines possède ses propres contraintes, parfois incompatibles, souvent liées entre elles.

Plusieurs types de technologies ont été mises en œuvre pour la reconnaissance des activités des personnes. Ces capteurs diffèrent les uns des autres en termes du type de données en sortie, de prix, de facilité d'installation [20]. Parmi ces capteurs, on peut citer : des podomètres/accéléromètres, des capteurs sonores, des capteurs de mouvement, des capteurs vidéos... Beaucoup d'annonces d'innovations basées sur l'utilisation de caméras (de type RGB, RGBD, infrarouge ...) sont apparues ces dernières années. La perception « d'intrusion dans la vie privée », la limitation d'action dans le noir de certaines technologies et les coûts incompatibles avec les exigences du marché de masse font que ces technologies n'arrivent pas à percer et que les produits qui en dérivent ne touchent qu'une clientèle ciblée, sans arriver à s'adresser à un très grand nombre de consommateurs.

Dans les bâtiments, pour les réseaux électriques et les réseaux de communication, Legrand tient une place majeure en tant que fournisseur d'offres produits et de solutions complètes. Ces produits étaient historiquement raccordés aux réseaux mais indépendants en termes de fonctionnement. Le développement des systèmes domotiques et des systèmes de gestion du bâtiment tertiaire a apporté une nouvelle dimension en matière de fonctions et de connectivité.

Tous les produits de commande, les capteurs et les actionneurs qui sont raccordés aux réseaux des bâtiments peuvent avantageusement échanger des données via internet et apporter de nouveaux services au-delà de leur usage premier. L'entreprise "Legrand" dispose donc d'un potentiel très important d'objets connectables et certains produits ont déjà les ressources techniques pour devenir des objets connectés. Les capteurs qui informent sur la présence, donnent un état technique, mesurent une grandeur physique,... sont autant de produits qui peuvent être connectés et apporter des informations localement ou à distance sur l'état du bâtiment, sur son exploitation. Une possibilité nouvelle et importante est la capacité de faire communiquer et fonctionner des applications différentes entre elles qui échangent sur des langages différents. Cela permet de rendre

interopérables des systèmes jusqu'à présent fonctionnant séparément et d'apporter de nouvelles fonctions au niveau supérieur de gestion du bâtiment : gestion d'énergie, sécurité, confort... C'est donc ainsi qu'est née le projet "CoCAPS" (**C**omportement **CAP**teur**S**) l'objectif global consiste à utiliser les données provenant de différentes sources (microphone, caméra, détecteur PIR, détecteur d'alitement ou encore objet connecté) pour la reconnaissance de l'activité afin d'améliorer le confort et la sécurité des personnes. En d'autres termes le projet ambitionne de développer des capteurs faible coût permettant de fournir des informations enrichies sur le comportement de(s) personne(s) à l'intérieur d'un bâtiment et de fusionner ces données afin de garantir à l'utilisateur un meilleur cadre de vie. La fusion d'information est la clé de l'innovation dans l'optimisation énergétique, dans la sûreté et la sécurité des personnes et pour une meilleure efficience d'usage de nos systèmes et objets quotidiens.

Le consortuim construit autour du projet CoCAPS est composé d'organisations complémentaires dont un industriel, de trois PME dont une start-up innovante et trois laboratoires de recherche. Chaque partenaire a un rôle précis au sein du projet CoCAPS (voir tableau 1.1). Mon sujet de thèse est en lien direct avec le partenaire industriel IRLYNX

Partenaire	**Type**	**Rôle dans le projet**
LEGRAND	Industriel	- Leader du projet - Conception Mécanique - Logiciels embarqués fusion - Conception optique - Validation des algorithmes de fusion sur la plateforme embarquée
id3 Technologies	PME	- Conception d'un module de détection sur la base d'une matrice thermopile, intégration et packaging composant
Irlynx	PME	- Conception d'un module de détection sur la base d'un imageur CMOS IR - Logiciel embarqué analyse d'image - Intégration optique
EMKA ELECTRONIQUE SITE DE SOREC	PME	- Cartes électroniques plateforme - OS embarqué / API pour le développement logiciel - Marché de pré série sur capteurs son et imageurs thermopile ou bolomètre
Université d'Orléans - Pôle Capteurs - Laboratoire PRISME - CEDETE - VALLOREM	Recherche	- Coordination de projet avec le leader Legrand - Algorithmes de reconnaissance d'actions - Algorithme de fusion de données multimodale - Enquêtes d'acceptabilité amont
UTC Compiègne	Recherche	- Analyse de l'environnement sonore
Télécom SudParis	Recherche	- Algorithmes de prise de décision - Intégration logicielle

TABLE 1.1 – Rôle de chaque partenaire [19]

Dans ce mémoire, nos recherches portent principalement sur la reconnaissance d'actions à l'aide

d'un capteur infrarouge basse résolution développé dans le cadre du projet par la société Irlynx. Nos résultats seront utilisés et fusionnés avec des informations provenant des autres capteurs pour mieux caractériser l'activité humaine. La figure 1.1 représente l'articulation globale du projet en partant des capteurs, en passant par le traitement des données, la fusion d'information extraite de chaque capteur, la prise de décision et enfin une intégration du logiciel dans une plate forme embarquée.

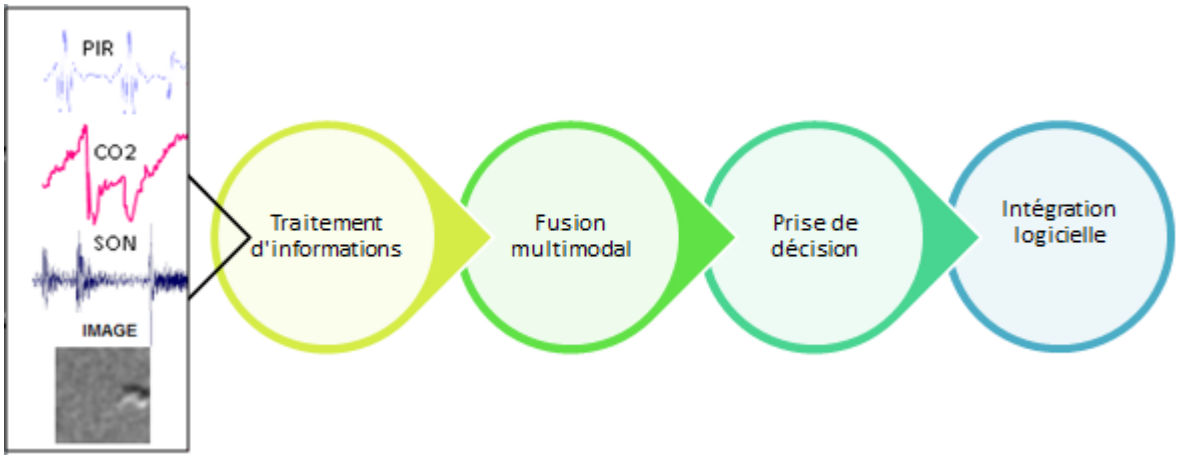


FIGURE 1.1 – Étapes du projet

Le capteur d'image infrarouge développé par la société Irlynx [1], et utilisé dans le cadre de nos travaux est un appareil capable de renvoyer des images d'objets en mouvement dans une pièce. Ce capteur a la particularité de détecter des corps chaud en mouvement dans une obscurité absolue et aussi celle de ne pas permettre la reconnaissance ou l'identification individuelle des acteurs de la scène. La technologie développée repose sur le principe de la détection pyroélectrique, c'est-à-dire la détection du déplacement d'un corps chaud présent dans le volume contrôlé. La présentation détaillée de ce capteur et des images sera faite dans le chapitre 3. On peut cependant voir quelques exemples d'images issues du capteur Irlynx sur la figure 1.2.

De nombreuses approches ont été développées ces dernières années, notamment dans le domaine de la vidéo surveillance [21] [22] pour la reconnaissance des activités ou des actions qui reste l'une des préoccupations majeures dans le domaine de la vision par ordinateur. Malgré des performances impressionnantes [23, 24, 25], le principal écueil auquel se heurtent les techniques développées pour la domotique est une réticence marquée des utilisateurs à être filmés. Même si les vidéos ne sont pas enregistrées et que le capteur joue un rôle limité, les informations sortantes ne portant que sur le comportement général (présence/absence, degré d'activité/immobilité...), la crainte d'une éventuelle utilisation abusive à des fins de surveillance reste très présente. Une enquête d'usage menée dans le cadre du projet CoCAPS [26] a confirmé que l'introduction d'une nouvelle technique ou de toutes solutions innovantes soulèvent des questions éthiques et sociales [27], principalement la question de l'acceptabilité d'une application automatisée. La préservation de l'anonymat assurée

1. www.irlynx.com

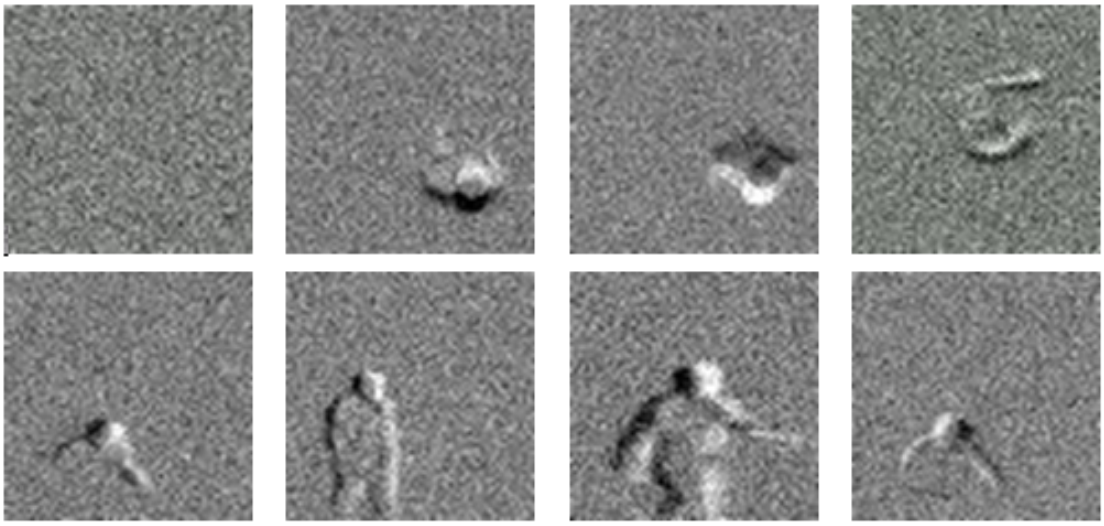


FIGURE 1.2 – Exemple d'images provenant du capteur Irlynx : la première ligne présente les images en vue de dessus et la seconde celles en vue de face

par le capteur Irlynx (voir figure 1.2) confère donc à cette technologie un potentiel d'acceptation certain tout en soulevant de nouveaux challenges pour l'objectif de reconnaissance d'actions visé. Dans la section suivante nous présentons les contraintes liées à notre application.

1.2 Problématique et Contraintes

Bien que le contenu vidéo soit assez informatif, la tâche de reconnaissance de l'action humaine reste difficile. Dans notre cas d'étude, la première problématique est la nature des données en entrée, qui sont des images très différentes des images du visible ou de l'infrarouge (base de données pour la reconnaissance d'actions NTU RGB+D [2]). Les données fournies par le capteur Irlynx sont assimilables à des images extrêmement bruitées ce qui met en défaut plusieurs approches de la littérature. La contrainte intrinsèque au projet qui est l'utilisation de capteur à faible coût garantissant le respect de l'intimité des personnes nous amène donc à travailler sur des images d'un nouveau type.

La seconde contrainte est le positionnement du capteur qui est en vue de dessus et au centre de la pièce surveillée. Ce choix est celui des industriels faisant partie du projet. Le produit final devant comporter d'autres capteurs (le capteur son notamment), il est judicieux que le boîtier soit au centre de la pièce pour une couverture maximale. Cette contrainte de positionnement complexifie également le problème car aucune application de reconnaissance d'activité n'utilise des bases d'images en vue de dessus.

De plus des difficultés additionnelles sont liées aux conditions d'enregistrement des séquences vidéo. Le changement des conditions de température et la distance de la caméra peuvent notamment influencer l'apparence des personnes détectées.

2. http ://rose1.ntu.edu.sg/datasets/actionrecognition.asp

D'autres contraintes dites de second niveau sont : la précision, la souplesse et la rapidité. Suivant l'application, nous sommes parfois amenés à faire un compromis entre ces différentes exigences. En médecine, les acteurs privilégient la précision, alors que dans des applications de surveillance on privilégiera en premier lieu la rapidité. Chaque solution doit trouver son propre équilibre entre ses contraintes, en fonction de son contexte d'application. Ces difficultés intrinsèques (bruits) et extrinsèques (contexte) ont de fortes répercutions sur la résolution du problème de la reconnaissance d'actions.

1.3 Plan de la thèse

Ce manuscrit est organisé en 5 chapitres :

Dans le chapitre introductif, nous décrivons le contexte général ainsi que la problématique à résoudre.

Dans le chapitre 2, nous présentons un état de l'art sur la reconnaissance d'actions à partir de séquences vidéos à l'intérieur de bâtiments. Nous présentons tout d'abord les méthodes basées sur les caractéristiques locales, ensuite les méthodes basées sur l'analyse des silhouettes et des membres du corps humains et enfin nous présentons des approches sur l'apprentissage profond.

Dans le chapitre 3, nous menons une étude sur les images provenant du capteur. Nous présentons d'abord le capteur utilisé, l'aspect physique des images puis justifions le choix effectué des approches qui sont développées dans ce mémoire. Enfin une analyse d'image est réalisée afin de proposer une chaîne de pré-traitement, segmentation et post-traitement adaptée pour le type d'images utilisées.

Dans le chapitre 4, nous nous concentrons sur la proposition de quelques méthodes de reconnaissance d'actions qui exploitent la méthode de segmentation proposée au chapitre 3. Dans la première partie, nous faisons la reconnaissance d'actions en utilisant et combinant plusieurs descripteurs de la littérature extraits sur une représentation des vidéos qui conserve la temporalité d'une action. Dans la seconde partie, nous proposons une méthode de reconnaissance d'actions qui utilise un vecteur descripteur basé sur des attributs statistiques pour l'identification d'un mouvement appris. Dans la troisième partie, nous exploitons les résultats obtenus pour proposer une méthode de reconnaissance d'actions avec plusieurs classifieurs (méthode en cascade). Dans la quatrième partie de ce chapitre, nous présentons quelques modèles du deep learning utilisés pour la reconnaissance d'actions à partir de séquences vidéos et faisons une étude comparative entre les approches classiques et des approches basées sur l'apprentissage profond. Pour terminer ce chapitre, nous décrivons et présentons l'approche retenue de reconnaissance d'actions sur des vidéos contenant plusieurs actions.

Dans le chapitre 5, nous résumons nos différentes contributions et présentons des perspectives d'amélioration ou d'autres aspects de recherche non traités dans ce manuscrit.

Chapitre 2

Etat de l'art

Sommaire

La reconnaissance d'actions peut cibler des aspects très diversifiés : des cas d'étude à l'intérieur du bâtiment (indoor), des cas d'étude à l'extérieur du bâtiment (outdoor) ou des cas d'étude mixte. Selon le cas considéré, les méthodes plus ou moins sophistiquées seront capables de s'adapter aux changements de luminosité, aux occultations, ou encore aux changements de point de vue, d'éclairage et d'apparence, ainsi qu'à la taille des images.

Ce chapitre a pour objectif de passer en revue les travaux de la littérature existante sur la reconnaissance d'actions indoor. Nous donnons un bref aperçu de la documentation, décrivons les techniques de recherche les plus pertinentes et les plus en vue liées à notre travail. Nous discutons également des points critiques où les méthodes existantes ne sont pas adaptées à tout type de vidéos.

2.1 Méthodes basées sur les caractéristiques locales (points d'intérêts)

Au cours des dernières années, diverses techniques ont été proposées pour la reconnaissance de l'action en utilisant des vidéos. Le schéma généralement adopté se compose d'une étape d'extraction des primitives et d'une étape de classification. L'extraction des primitives consiste à identifier des caractéristiques spatio-temporel distinctives à partir d'une séquence vidéo tout en étant robuste

au bruit. La même idée a été appliquée avec succès à la détection d'objets, à la reconnaissance de scènes, etc...

Les méthodes basées sur les caractéristiques locales décrivent les observations comme une série de descripteurs locaux ou patchs. Il n'est pas nécessaire de procéder à une étape de pré-traitement, par exemple la suppression de fonds et le suivi de silhouette. Ceci évite ainsi la propagation des erreurs produites en phase de pré-traitement, surtout dans le cas de scènes ayant des fonds dynamiques. Ces primitives sont généralement robustes, voire invariantes aux changements d'angle, à l'apparence des personnes et aux occlutations partielles.

Les points d'intérêt correspondent à un ensemble de pixels présentant une singularité que ce soit au niveau du gradient ou du contour. Parmi les premiers travaux proposés pour extraire les points d'intérêt spatio-temporels (STIPs) figure celui de Laptev et Lindeberg [1]. Les auteurs ont étendu le détecteur de coins de Harris [Harris and Stephens, 1988] en lui rajoutant la dimension temporelle (voir figure 2.1). Ce détecteur spatio-temporel, communément appelé Harris 3D, leur permet d'extraire des motifs de mouvement. Ces points d'intérêt spatio-temporels correspondent aux points dont le voisinage local est soumis à une variation spatiale et temporelle significative. Dans [28], leurs travaux ont été améliorés pour compenser les mouvements relatifs à la caméra.

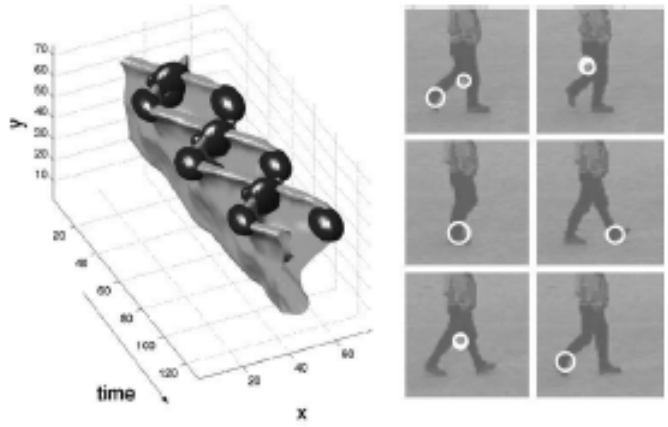


FIGURE 2.1 – Résultats de la détection des points d'intérêt spatio-temporels à partir du mouvement des jambes [1]

Dans [29], Dollar et al. ont proposé de nouveaux points d'intérêt spatio-temporels plus denses appelés cuboïde 3D ou simplement cuboïde. Un cuboïde est un cube construit à partir de pixels autour des points d'intérêt détectés. Ils estiment que le nombre de points issus du détecteur Harris 3D est relativement faible par rapport aux zones contenant des mouvements. Des études distinctes de Messing et al. [2] et Matikainen et al [30] ont remis en question le choix de cuboïdes comme représentation de séquence vidéo pour la reconnaissance d'actions et ont introduit la notion de trajectoires.

De nombreuses approches d'extraction de caractéristiques basées sur la trajectoire ont été proposées (voir figure 2.2), telles que le KLT-tracker[2], le SIFT matching [31], le DTF [32]. Cependant,

ces modèles présentent un certain nombre de faiblesses comme par exemple la présence de trajectoires non pertinentes ou redondantes, la présence de mouvements inutiles et la complexité des calculs.

Wang et Schmid [33] améliorent les performances des approches basées sur les trajectoires denses en prenant en compte les mouvements de la caméra et corrigent la présence des mouvements inutiles dus au mouvement de la caméra. Cela améliore considérablement la reconnaissance d'actions en utilisant des descripteurs basés sur le mouvement, tels que HOF et MBH.

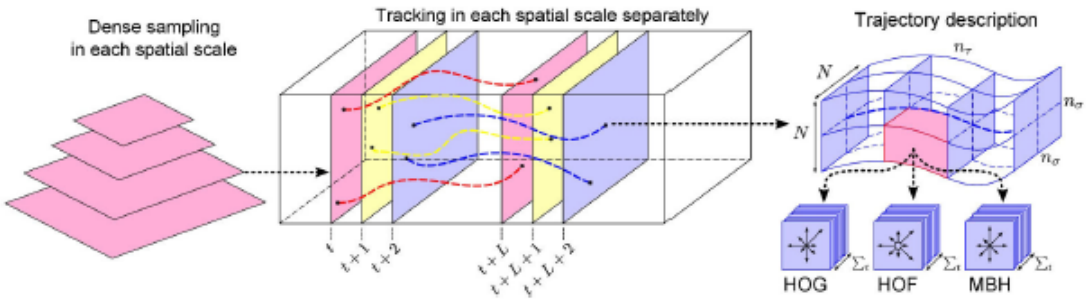


FIGURE 2.2 – Illustration des descripteurs de trajectoire de points denses [2].

Kaaniche et al. [34] ont développé une approche de reconnaissance de gestes en utilisant un système multi-caméras. Ils détectent les points d'intérêt dans la scène afin d'extraire les régions caractéristiques et les décrivent par des histogrammes de gradient orienté (HOG). Ensuite, ils effectuent un suivi temporel de ces descripteurs et classent les gestes en utilisant l'algorithme K-means à partir d'une base de gestes d'apprentissage. La reconnaissance est réalisée par un algorithme de plus proches voisins (K-Nearest Neighbors). Le caractère dynamique des gestes est décrit en temps réel par le suivi des points détectés par HOG dans la séquence. Cependant, l'utilisation de HOG comme caractéristique rend cette approche sensible aux changements d'échelle et rotations.

Willems et al. [35] ont proposé une extension du détecteur Hessian au domaine spatio-temporel. Le principe de cette méthode est de détecter des points d'intérêt denses, invariants à l'échelle. Zhen et Shao [36] ont introduit un détecteur compact décomposant une séquence vidéo avec une pyramide de Laplace, dans laquelle les caractéristiques spatio-temporelles sont localisées sur plusieurs échelles et orientations.

Karungaru et al. [3] utilisent un réseau de plusieurs caméras pour reconnaître des actions comme ("marcher", "s'arrêter", "courir", "s'asseoir", "travailler au bureau"). Ils proposent l'extraction et la détection de régions candidates humaines à l'aide de HOG [37] et d'AdaBoost [38] et associent cela à un ensemble de règles pour reconnaître des actions. Cette méthode garantit la détection des personnes et la reconnaissance des actions avec une grande précision en temps réel. Les règles sont définies sur la distance parcourue, la direction et l'inclinaison du corps. La figure 2.3 montre l'aperçu des étapes de reconnaissance basée sur les règles proposées dans cet article.

Scovanner et al [39] présentent un nouveau descripteur nommé SIFT 3D qui est une extension

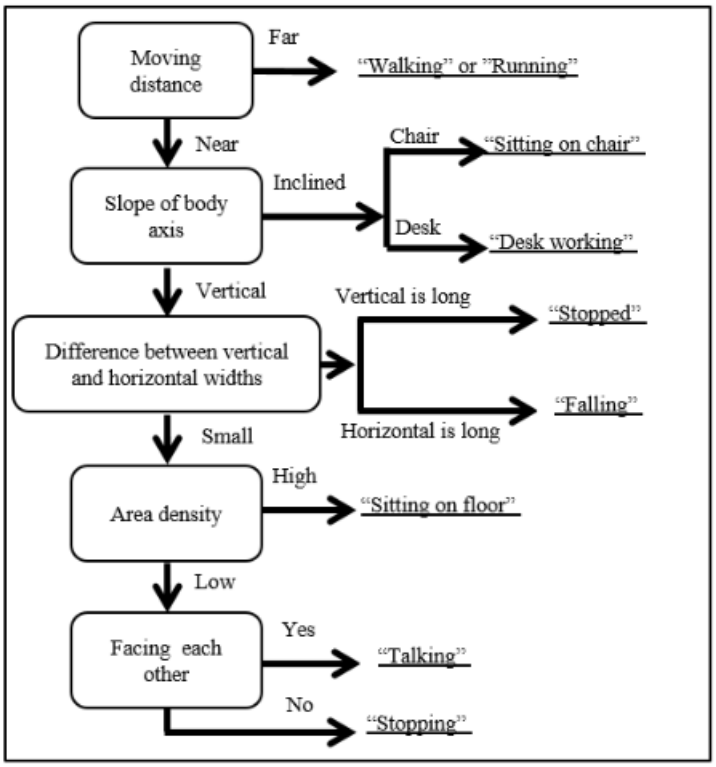


FIGURE 2.3 – Aperçu de la méthode de reconnaissance des actions humaines basée sur des règles [3]

du descripteur SIFT 2D [40]. Ils extraient des caractéristiques SIFT 3D sur des vidéos et utilisent les sacs de mots pour représenter le vecteur caractéristique. Avec cette approche, ils atteignent une performance de 82,6% sur la base Weizmann.

D'autres représentations spatio-temporelles comme les "local trinary patterns" (LTP) [41], et HOG3D[42], ESURF (Extended SURF) [35] ont également montré leur efficacité pour la reconnaissance d'actions, principalement grâce à leur robustesse face aux occlutations partielles et au bruit. Kläser et al. [42] suggèrent d'utiliser l'histogramme des orientations de gradient comme un descripteur de mouvement. Tout en s'inspirant de (HOG) [37], les auteurs l'étendent pour qu'il soit applicable au domaine spatio-temporel et propose donc "HOG3D" ceci en exploitant l'idée selon laquelle le flux optique modélise les mouvements de chaque pixel dans une vidéo. Laptev et al[22] proposent d'utiliser l'histogramme de flux optique (HOF) sur les régions locales comme descripteur spatio-temporel. Une extension plus robuste du descripteur HOF est le "Motion Boundary Histogram" (MBH) présenté dans Dalal et al. [43]. Bien qu'étant riche d'informations, le calcul du champ du flux optique est coûteux. Pour surmonter cette difficulté, Kantorov et Laptev [44] proposent d'utiliser les techniques de décompression vidéo. Plus précisément, au lieu de calculer les champs du flux optique pour obtenir MBH ou HOF les auteurs utilisent les champs de mouvement en compression MPEG qui peuvent être obtenus facilement par des méthodes de décodage vidéo.

Dans Zhao et Pietikainen [45], diverses extensions des descripteurs LBP (Local Binary Patterns) au domaine spatio-temporel sont présentées. Les informations sont codées par l'histogramme des motifs binaires. Le descripteur LBP est calculé en quantifiant le voisinage d'un pixel par rapport à son intensité.

2.2 Méthodes basées sur l'apparence

Les méthodes de reconnaissance d'actions basées sur l'apparence humaine utilisent des informations 2D ou 3D sur les parties du corps humain, telles que les positions et les mouvements des parties du corps. Ceci permet de se focaliser sur ces mouvements, indépendamment de ce qui peut se passer à l'arrière-plan. Ces méthodes peuvent être divisées en deux groupes portant sur : l'analyse de la forme du personnage, par le biais de sa silhouette et l'analyse du mouvement du personnage par le biais d'une identification de ses membres (mains, tête, jambes ...).

2.2.1 Analyse de la silhouette

Dans les analyses de la forme par des silhouettes, une première étape est nécessaire, celle de l'extraction des formes de la personne dans les images. Cela se fait en supprimant généralement l'arrière plan. Parfois après suppression d'arrière plan, l'image résultante comporte la personne et d'autres objet. Dans ce cas une modélisation plus complexe peut être mise en œuvre par exemple l'identification du personnage par un algorithme de détection d'humain. Une fois la silhouette extraite, une représentation de la séquence vidéo est nécessaire et enfin vient l'étape de la classification

Nous commençons par décrire les travaux influant de Bobick et Davis [4] dans lesquels sont présentés le Motion Energy Image (MEI) et le Motion History Image (MHI). L'idée principale est de coder l'information relative au mouvement par une seule image pouvant conserver l'information spatiale et temporelle. Ils procèdent à l'extraction des silhouettes à partir d'une seule vue et regroupent ensuite les différentes images segmentées. Le MEI construit est une image binaire décrivant l'endroit où le mouvement se produit. Il est défini par la formule 2.1.

$$MEI(x,y,t) = \cup_{i=0}^{\tau-1} D(x,y,t-i) \tag{2.1}$$

où $D(x,y,t)$ est une séquence d'images binaires représentant les pixels de l'objet détecté.

Le modèle MHI montre comment l'image animée se déplace. Chaque pixel du MHI est fonction de l'historique temporelle du mouvement en ce point (c'est à dire que les intensités plus élevées correspondent aux mouvements plus récents). Le MHI est défini par la formule 2.2

$$MHI(x,y,t) = \begin{cases} \tau & si \quad D(x,y,t) = 1 \\ max(0, MHI(x,y,t-1) - \delta) & sinon \end{cases} \tag{2.2}$$

où la durée τ contrôle la valeur temporelle du mouvement, et δ est un paramètre de décroissance qui permet de mettre à jour les valeurs des pixels de chaque nouvelle image analysée.

L'intensité des pixels de l'image de l'historique des mouvements représente la trace des mouvements de la silhouette (voir figure 2.4).

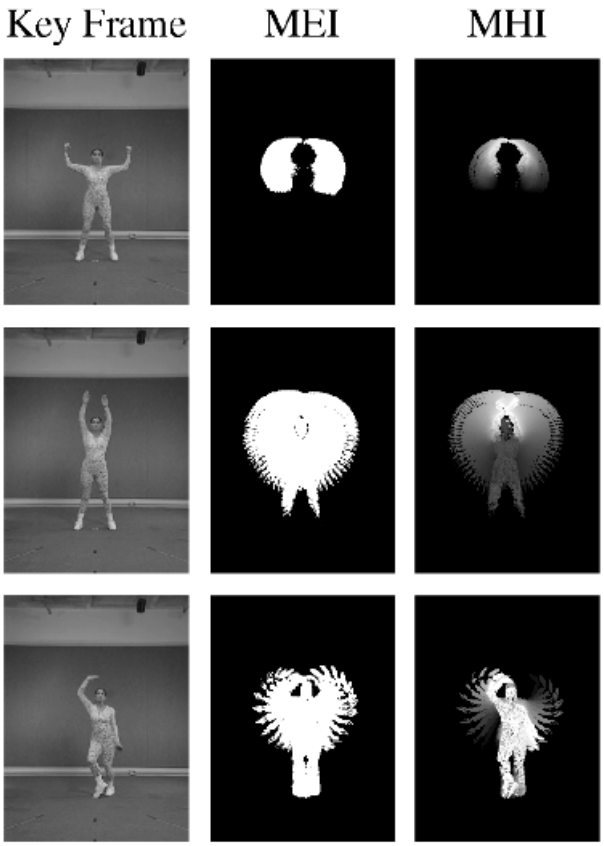


FIGURE 2.4 – Exemples de MEI et MHI [4]

Shechtman et Irani [46] ont appliqué un principe similaire à l'approche de Bobick et Davis [4] en utilisant des patchs au lieu de la silhouette complète. Leur méthode utilise des corrélations de petites parties du volume extrait de la vidéo. Ainsi l'apprentissage se fait seulement sur quelques patchs caractérisant l'action. De plus, il est possible de reconnaître plusieurs actions se produisant sur la même séquence vidéo en entraînant des patchs avec des étiquettes différentes sur une même image (voir figure 2.5). Le véritable problème dans cette approche est le regroupement de l'ensemble des patchs caractérisant une action.

Shao et Chen [47] ont proposé une méthode qui consiste à extraire les silhouettes puis les projeter dans un espace de dimension inférieure afin obtenir un histogramme des occurrences de chacun des mots visuels et de l'utiliser comme vecteur caractéristique d'une action. Avec cette approche les auteurs atteignent 100% de performance sur la base Weizmann.

Gorelick et al. [48] ont proposé un modèle basé sur des formes tridimensionnelles induites par les silhouettes dans le volume espace-temps. À chaque image, ils calculent une information de silhouette à l'aide d'une technique de soustraction d'arrière-plan. Ils empilent des silhouettes sur une séquence donnée pour former un volume spatio-temporel (voir Figure 2.6) et ils appliquent

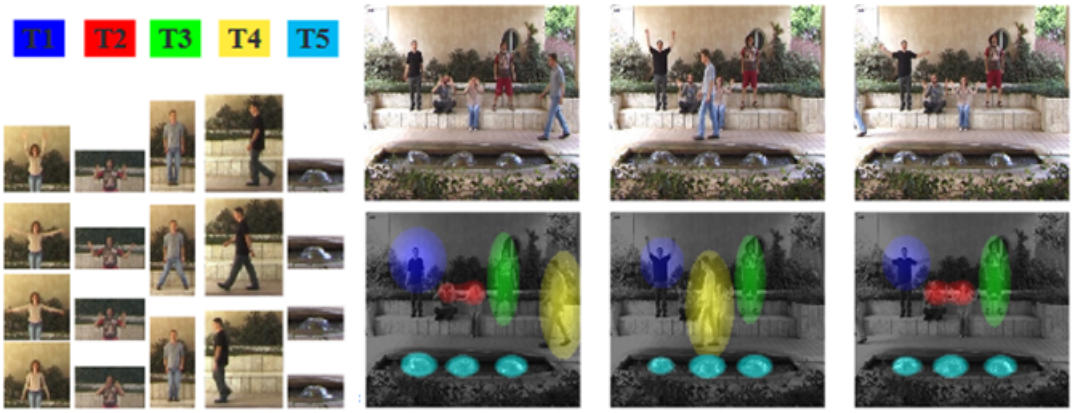


FIGURE 2.5 – Shechtman et Irani [4] : Exemples de multiples actions

l'équation de Poisson pour déduire les caractéristiques de saillance et d'orientation d'un pixel par rapport à son voisinage. Ils utilisent des trames d'une longueur de 10 frames et apparient ces trames en utilisant une approche à fenêtre coulissante. La classification est effectuée à l'aide d'un algorithme simple du plus proche voisin avec une distance euclidienne. Zhu et al. [49] ont utilisé une extension de la transformée de Radon sur les silhouettes binaires. Une telle représentation est invariante aux transformations géométriques telles que la mise à l'échelle et la translation.

Vili et al [50] proposent une approche pour l'analyse de l'action en décrivant les actions humaines avec des caractéristiques de texture. Il extraient des histogrammes LBP à partir des MHI et MEI et modélisent le comportement avec un modèle de Markov. Une performance de 97,8% est obtenue sur la base Weizemann.

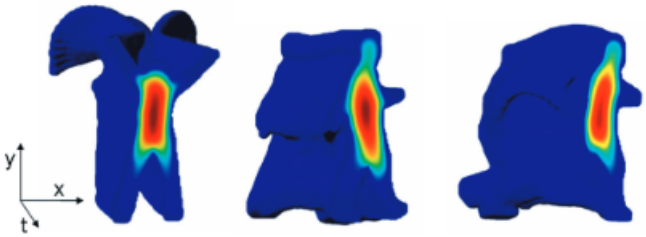
Zheng et al. [51] extraient la silhouette ensuite ils appliquent la transformée de $\Re$ ($\Re$ - Transform) pour extraire des caractéristiques et appliquent la carte de diffusion semi-supervisée (SSDM) pour la réduction de la dimensionnalité. Les cartes de diffusion semi-supervisées caractérisent la propriété spatio-temporelle de l'action, tout en préservant une grande partie de la structure géométrique locale et des informations sur les étiquettes. Ils obtiennent une performance de 98,8% sur la base Weizmann [52].

Le principal inconvénient de la méthode basée sur l'analyse des silhouettes est l'exigence d'extraction de la silhouette ce qui se fait typiquement par segmentation. La précision de ces techniques est fortement tributaire de la qualité d'extraction pour pouvoir obtenir de forts taux de classification.

2.2.2 Analyse du squelette ou des parties du corps humain

Certaines parties du corps humain peuvent être décrites dans l'espace 2D sous forme d'éléments d'image rectangulaires ou sous formes volumétriques dans l'espace 3D (voir Figure 2.7). Comme une silhouette humaine se compose de membres reliés entre eux, il est important d'obtenir des parties exactes du corps humain à partir de vidéos, ce problème est considéré comme faisant partie

(a) Formes spatio-temporelles des actions "jumping-jack", "walk" et "run" [48]



(b) l'équation de Poisson sur les formes spatio-temporelles

FIGURE 2.6 – Exemples de volumes spatio-temporels construits en empilant des silhouettes sur une séquence d'images.

du processus de reconnaissance de l'action. De nombreux algorithmes ont été développés pour résoudre ce problème.

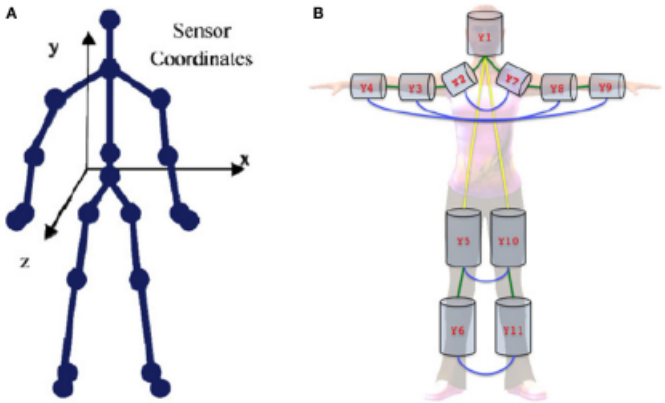


FIGURE 2.7 – Représentation du corps humain. (A) Modèle squelette 2D [5] et (B) Représentation de la silhouette en 3D [6]

Les méthodes basées sur le modèle de corps humain sont inspirées des travaux de Johansson [7] sur la perception visuelle des mouvements des parties du corps des organismes vivants. Il a montré que les actions humaines peuvent être reconnues en fixant quelques points lumineux au corps humain, décrivant ainsi les mouvements des principales articulations du corps. Il a constaté qu'entre 10 et 12 points lumineux sont suffisants pour reconnaître les mouvements de marche, de

course, de danse ... (voir la figure 2.8).

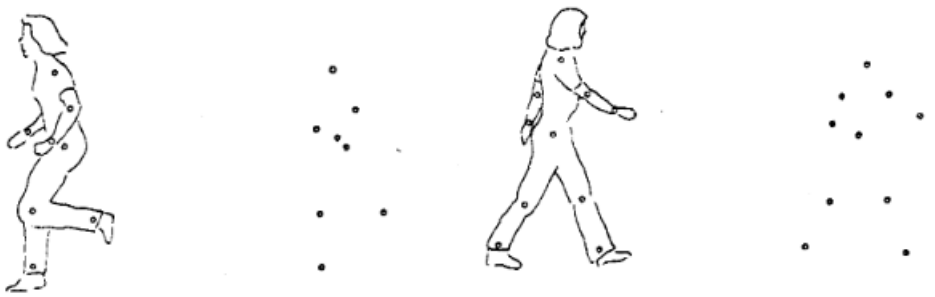


FIGURE 2.8 – Illustration d'un mouvement de course et d'une marche et leurs points lumineux attachés au corps humain Johansson [7]

Dans [53], les extrémités du corps humain comme la tête, les mains et les pieds sont utilisées pour modéliser l'action. Ces extrémités sont détectées à partir d'un contour du corps. Sheikh et al. [54] ont appliqué une projection affine aux trajectoires des articulations afin de reconnaître des actions, de façon invariante à l'angle de vue, par la mesure des angles entre les articulations dans l'espace projeté.

Dans [55], Xia et al. ont proposé une représentation compacte des postures appelées HOJ3D (histograms of 3D joint locations). Cette représentation correspond aux histogrammes des positions 3D des articulations. Ils encodent principalement l'occupation spatiale des articulations par rapport au centre de la silhouette. En effet, les articulations sont projetées dans un espace sphérique partitionné en n-bins. Ensuite, une quantification vectorielle est réalisée à l'aide de k-means pour construire les vecteurs de caractéristiques. Un modèle de Markov caché est utilisé pour la classification d'actions.

2.3 Méthodes basées sur l'apprentissage profond

Les méthodes d'apprentissage profond pour la reconnaissance d'actions sont étroitement liées à l'utilisation des réseaux neuronaux convolutifs (CNN) en classification d'images. En général, l'un des principaux avantages des CNN est qu'ils permettent de gérer de manière globale toutes les étapes de reconnaissance d'actions : extraction des caractéristiques (filtres) et la classification. Grâce à cela, nous pouvons obtenir des caractéristiques spécifiques qui décrivent mieux les données et facilitent la tâche de classification. Bien entendu, cette approche est susceptible de surajuster les données, de sorte que le processus d'entraînement des CNN nécessite une régularisation minutieuse et, en général, une quantité importante de données labellisées. Etant donné que les vidéos introduisent une dimension temporelle, la plupart des auteurs utilisent des CNN entraînés sur bloc d'images pour modéliser cet aspect temporel.

Ji et al. [8] proposent un modèle qui extrait des caractéristiques à la fois spatiales et temporelles en effectuant des convolutions 3D, capturant ainsi les informations de mouvement codées dans

plusieurs trames adjacentes. Pour améliorer encore les performances des modèles 3D-CNN, les auteurs régularisent le modèle avec des fonctionnalités de haut niveau en combinant les sorties de différents couches. Dans la figure 2.9 l'architecture se compose d'une couche d'entrée, de 3 couches de convolution, de 2 couches de sous-échantillonnage et d'une couche de sortie. Baldominos et

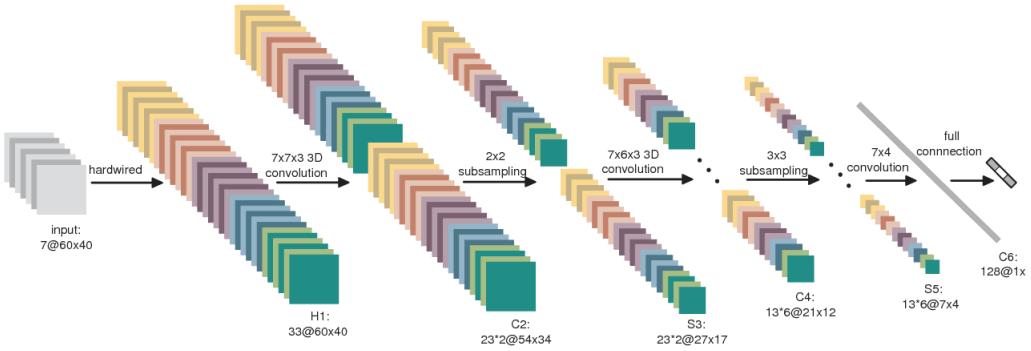


FIGURE 2.9 – Une architecture 3D-CNN pour la reconnaissance de l'action humaine [8]

al. [56] proposent l'utilisation de l'apprentissage en profondeur (réseau de neurone convolutif) pour effectuer la reconnaissance d'activité à l'aide d'un jeu de données multimodales (caméra, microphone ...). Ensuite, au lieu de choisir manuellement une topologie appropriée, un algorithme évolutif est en charge de déterminer la topologie optimale afin de maximiser le score de classification.

Jalal et al. [57] utilisent plusieurs descripteurs basées sur les régions d'intérêts (distance intra, inter région entre personne ...) ainsi qu'un réseau neuronal convolutionnel (CNN) pour la reconnaissance d'actions prédéfinies entre 2 personnes.

Delachaux et al. [58] proposent une stratégie visant à améliorer la robustesse de la reconnaissance des activités humaines en intérieur en combinant des capteurs portables et des caméras. Ils utilisent un ensemble de réseaux de neurones binaires un-contre-tous plus précisément le perceptron multicouche. C'est un type de réseaux de neurones formel organisé en plusieurs couches au sein desquelles une information circule de la couche d'entrée vers la couche de sortie uniquement. Chaque réseau neuronal individuel a été formé pour la reconnaissance d'une seule activité. La raison principale de ce choix est que différentes activités peuvent nécessiter des caractéristiques différentes.

D'autres auteurs montrent qu'on peut utiliser une architecture à deux flux pour la reconnaissance d'actions. Cela permet non seulement d'augmenter les nombres de données mais aussi la qualité des résultats du modèle. Le plus souvent les différents flux permettent de capturer des informations spatiales et des informations temporelles. Dans [9] Simonyan et Zisserman ont utilisé un réseau CNN à deux flux pour incorporer les deux types de caractéristiques, un flux prenant les images RGB en entrée et l'autre prenant les flux optiques superposés pré-calculés (voir figure 2.11). Comme le flux optique ne contient que des informations de mouvement bref, son ajout ne permet pas au modèle CNN d'apprendre les mouvements de longue durée. Le flux supplémentaire permet de considérablement augmenter la précision de la reconnaissance d'actions. Tran et al. [59] ont évité

d'avoir à pré-calculer les caractéristiques du flux optique. Ainsi ils ont proposé une architecture 3D-CNN, qui permet aux réseaux profonds d'apprendre également les caractéristiques temporelles.

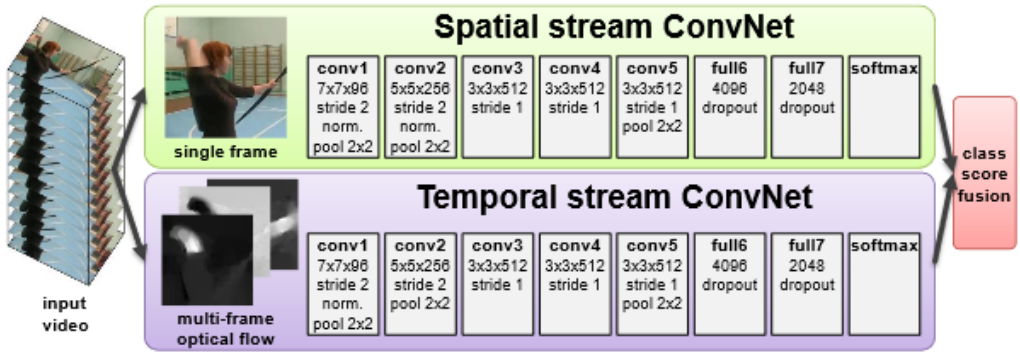


FIGURE 2.10 – Simonyan et Zisserman [9] : Architecture à deux flux pour la classification vidéo.

Dans [10], une représentation vidéo avec une approche d'extraction de trajectoire est tout d'abord mise en place puis des cartes de caractéristiques sont extraites en faisant des convolutions à multi-échelles. Pour renforcer la robustesse des descripteurs de trajectoire à convolution profonde (en anglais Trajectory-Pooled Deep-Convolutional Descriptors(TDD)), les auteurs conçoivent deux méthodes de normalisation pour transformer des cartes de caractéristiques convolutionnelles, à savoir la normalisation spatio-temporelle et la normalisation des canaux.

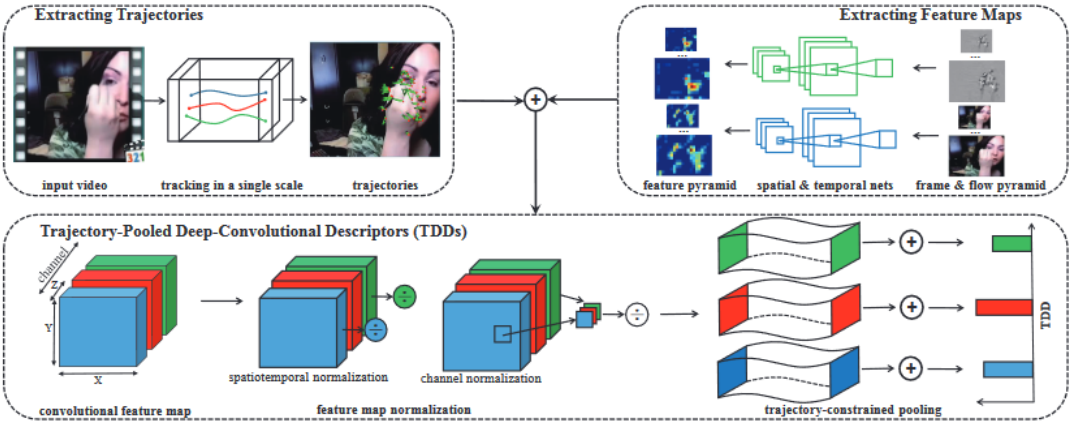


FIGURE 2.11 – Wang et al. [10] : Vue d'ensemble du TDD. Le calcul du descripteur se fait en 3 étapes : (i) extraction des trajectoires par la méthode des trajectoires denses, (ii) extraction de cartes de convolution à plusieurs échelles, et (iii) empilage des cartes de convolution obtenues à partir des points de la trajectoire.

Cependant, la méthode basée sur CNN n'extrait que les caractéristiques d'apparence visuelle et n'a pas les caractéristiques temporelles à longue portée. De plus, la méthode basée sur CNN ne tient pas compte de la différence intrinsèque entre les domaines spatial et temporel. C'est ainsi que

récemment, les réseaux neuronaux récurrents (RNNs) [60, 61], ont suscité beaucoup d'intérêt dans la résolution de nombreux problèmes complexes liés aux données de séries temporelles.

Afin de modéliser la dynamique temporelle entre les trames vidéo, les RNNs ont été considérés pour la reconnaissance d'actions humaines dans une vidéo. Dans ce type d'architecture une nouvelle donnée d'une couche cachée est obtenue en prenant en compte à l'étape courante une ou plusieurs informations prédites dans une étape précédente. La plupart des méthodes de pointes [62, 63, 64, 65, 8] ont proposé leurs propres réseaux en s'appuyant sur les CNNs et les RNNs. Ils présentent des modèles capables de coder plus d'informations visuelles en préservant les structures spatiales et temporelles de l'action humaine se produisant dans une séquence vidéo et permettent d'atteindre des performances impressionnantes. Par exemple, dans [64], les auteurs obtiennent une performance de 90,1% sur la base UCF101.

Cependant, en raison du grand nombre de calculs de paramètres et de l'entrée initiale dans chaque couche, apparaissent souvent des problèmes de disparition de gradient. La solution à ce problème est apportée dans les méthodes LSTM [66] [67], qui ont la capacité de capturer les dépendances à long terme et de préserver les informations de séquence dans le temps en intégrant des unités de mémoire. Le LSTM a été introduit pour la première fois dans [68, 69], il a été adapté avec succès à de nombreuses tâches de modélisation séquentielle telles que la reconnaissance vocale, la description visuelle et la traduction automatique, et a permis d'atteindre des performances encourageantes. Dans la plupart de ces réseaux, les entrées du LSTM sont les caractéristiques de haut niveau captées à partir d'une couche entièrement connectée de 3D-CNN (voir figure 2.12).

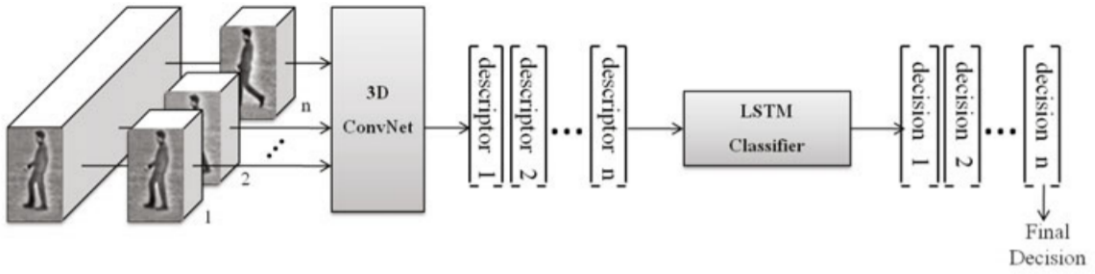


FIGURE 2.12 – Baccouche et al. [11] :Une architecture 3D-CNN pour la reconnaissance de l'action humaine.

Dans [70], Arif et al. proposent un nouveau framework de reconnaissance d'actions qui combine intelligemment 3D-CNN (réseau de neurone convolutif) pour générer une nouvelle carte de mouvement et le réseau LSTM (mémoire à court terme) qui est utilisé comme codeur-décodeur pour les prédictions finales. Les performances atteintes sont de 92,9 % sur la base UCF sport [71] et 72,1% sur la base HDMB [72].

Du et al. [12] proposent un RNN hiérarchique pour la reconnaissance d'actions basé sur le squelette du corps humain. Au lieu de prendre le squelette entier comme entrée, ils divisent le

squelette humain en cinq parties en fonction de la structure physique humaine (voir Figure 2.13), puis les relient séparément à cinq sous-réseaux.

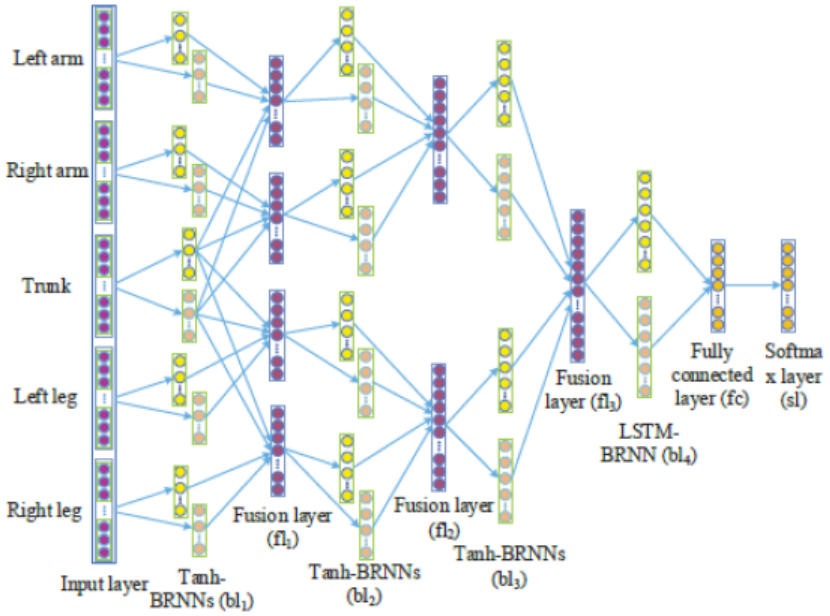


FIGURE 2.13 – Du et al.[12] : Une architecture RNN pour la reconnaissance de l'action humaine.

L'avantage des méthodes basées sur l'apprentissage profond est qu'elles ne nécessitent pratiquement aucun pré-traitement des données d'entrée. Cependant, le défi majeur réside dans la conception d'une topologie appropriée pour un ensemble de données pris en entrée.

Les principales couches rencontrées dans un réseau de neurone sont :

— La couche convolutive constituée d'un noyau 3D (noyau de convolution ou filtre de convolution). Ce noyau 3D de convolution permet de faire ressortir des informations dans les images par exemple, les lignes, les variations de coloration, etc.... Grâce à l'usage de noyau 3D, les aspects spatial et temporel sont pris en compte tandis que pour un noyau 2D c'est juste l'aspect spatial qui est considéré. Chaque convolution nous donne des cartes de caractéristiques.

— La couche Max pooling consiste à faire glisser une petite fenêtre pas à pas sur toutes les parties de l'image et à retenir la valeur maximale des pixels contenus dans cette fenêtre (voir image 2.14). Ce procédé permet de réduire la taille spatiale des images intermédiaires, réduisant ainsi la quantité de paramètres et de calcul dans le réseau tout en préservant les informations les plus importantes qu'elles contiennent. Il est donc fréquent d'insérer périodiquement une couche de pooling entre deux couches convolutives successives d'une architecture de réseau de neurones convolutifs pour réduire le sur-apprentissage.

— La couche dropout proposée dans [73] est une technique de régularisation des modèles de réseaux de neurones. Elle permet d'activer et désactiver les neurones aléatoirement dans

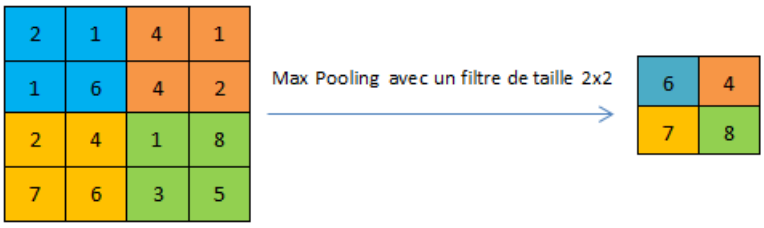


FIGURE 2.14 – Exemple de Max Pooling

le réseau, dans le cadre d'entraînements successifs. Une fois les séries d'entraînements terminées, tous les neurones sont ré-activés et le réseau peut fonctionner de façon classique. Cette technique a montré non seulement un gain dans la vitesse d'apprentissage, mais en déconnectant les neurones, on a aussi limité des problèmes de sur-apprentissage, rendant le réseau plus robuste et capable de mieux généraliser les classes apprises.

— La couche flatten va permettre d'aplatir la sortie du réseau de convolution en un vecteur. Ce vecteur est ensuite utilisé comme entrée dans un classifieur.

— la couche softmax donne une probabilité en sortie de chaque neurone. Le neurone de sortie avec la probabilité la plus grande permet alors de décider que sa classe associée est la classe prédite.

2.4 Comparaisons de l'état de l'art

De nombreux jeux de données allant des bases de données comportant des actions simples aux bases contenant des actions plus complexes ont été créés et utilisés pour des études comparatives de différentes méthodes. Le type d'images figurant généralement dans les bases de données pour la reconnaissance d'actions sont des images couleurs (RGB), des images de profondeur (D), les skeletons (S) et des images infrarouges. On peut citer quelques-unes de ces bases :

— la base de données KTH [74] composée de 6 actions effectuées par 25 personnes : marche, jogging, courir, boxe, agitation des mains et applaudissements. La base de données contient des vidéos en intérieur et en extérieur.

— La base Weizmann [52] qui est composée de 10 actions effectuées par 9 acteurs différents : marcher, courir, bondir, galoper latéralement, s'accroupir, lever une main, lever deux mains, sauter sur place.

— La base MSR Action3D [75] contient 20 types d'actions en vue de face (lever une main, lever 2 mains, applaudir, dessiner le cercle, service de tennis, swing de golf, coup de pied, boxe ...) effectués par 10 personnes, chaque personne effectuant chaque action 2 ou 3 fois. La résolution est de 640x240. Les données ont été enregistrées à l'aide d'un capteur de profondeur similaire à l'appareil Kinect.

— La base UCF101 [71] est l'extension de UCF50 ; elle contient 101 catégories d'actions différentes. Chaque action est constituée d'au moins 100 vidéos. Il y a 13320 clips vidéo au total.

La plus grande partie de ces vidéos sont des vidéos réalistes et comprennent des actions comme se coiffer, sufer, faire du sky, jouer au tambour, plonger...

— L'ensemble de données HDMB51 [72] comprend une variété de vidéos réalistes recueillies sur YouTube et Google. Il y a 6766 clips vidéo annotés manuellement de 51 classes d'action différentes, chaque action contenant environ 100 clips vidéo.

— L'ensemble de données UCF-sport comprend 150 vidéos provenant de 10 classes d'actions (plonger, jouer au golf, donner des coups de pied...). Ces vidéos sont enregistrées dans des environnements sportifs réels, extraits de différentes chaînes de télévision. Cet ensemble de données présentent des occlutations, des variations d'éclairage et d'arrière-plan, et cela en fait une base de données complexe.

— La base Hollywood2 [76] a été extraite de 69 films hollywoodiens différents et comprend 12 classes d'actions (répondre au téléphone, manger, se saluer..). Elle contient 1 707 vidéos divisées en un ensemble d'entraînement (823 vidéos) et un ensemble de test (884 vidéos). Les vidéos d'entrainement et de test proviennent de différents films.

— CAD-60 [77] est un jeu de données qui contient les images RGB, les séquences de profondeur et le squelette. Les données ont été saisies par le capteur Kinect de Microsoft. L'ensemble des données est constitué de 12 actions effectuées par 4 sujets. Les actions sont effectuées dans 5 environnements différents : bureau, cuisine, chambre, salle de bain et salon. L'ensemble des données contient 60 vidéos.

— MSRDailyActivity3D [78] est une base de données comprenant 16 actions (boire, manger, lire un livre, passer un appel, écrire sur un papier, rester assis, s'allonger sur un canapé, marcher, s'asseoir...). Chaque action est exécutée par 10 sujets et chaque sujet exécute le cas échéant chaque action en position debout et assise, ce qui ajoute une variation supplémentaire au sein de la classe. Au total, l'ensemble de données contient 320 vidéos enregistrées avec une résolution spatiale de 640×480 pixels. Des images RGB, des cartes de profondeur et des squelettes sont disponibles pour toutes les vidéos.

— UTD-MHAD [79] est une base de données dite multimodale. Elle a été collectée à l'aide d'un capteur Kinect de Microsoft et d'un capteur inertiel (le capteur inertiel retourne des signaux d'accélération et de rotation) dans un environnement intérieur. L'ensemble de données contient 27 actions effectuées par 8 personnes (4 femmes et 4 hommes). Pour cette base, quatre modalités de données sont enregistrées : les vidéos RGB, les vidéos de profondeur, les positions des articulations du squelette et les signaux des capteurs inertiels.

Nous avons synthétisé quelques approches de l'état de l'art dans le tableau 2.2. Celui-ci permet de résumer les caractéristiques principales et performances obtenues par les différentes approches en fonction de leur contexte d'étude (base de données).

Base de données	Année	Nombre de classe	Nombre de vidéo par classe	Type d'images / Type de vue	Résolution
KTH	2004	6	10	RGB, vue de face	160 × 120
Weizmann	2005	10	9	RGB, vue de face	180 × 144
Hollywood	2008	8	30 - 140	RGB	-
Hollywood2	2009	12	61 - 278	RGB	-
UCF Sport	2009	10	14 - 35	RGB, vue de face	720 × 480
UCF YouTube	2009	11	100	RGB, vue de face	-
MSR	2010	20	14-25	RGB-DS, vue de face	640 × 240
UCF50	2010	50	100 et plus	RGB, vue de face	-
HMDB51	2011	51	101 et plus	RGB, vue de face	-
CAD-60	2011	12	60	RGB-DS, vue de face	240 × 320
UTD-MHAD	2015	27	32	RGB-DS+Signal ; vue de face	640 × 480

TABLE 2.1 – Tableau récapitulatif de quelques bases de données utilisées dans la littérature pour la reconnaissance d'actions.

2.5 Conclusion

Nous avons présenté les techniques de la littérature qui ont donné des résultats intéressants pour la reconnaissance d'actions.

— Méthodes basées sur l'apparence : elles sont subdivisées en deux techniques. La première sur l'analyse du squelette ou des parties du corps humain et la seconde basée sur l'analyse de la silhouette. La première consiste à extraire des informations 2D ou 3D sur des parties du corps humain en d'autres termes à analyser le mouvement de la personne par le biais de l'identification de ses membres (mains, tête, jambes, ...) ou aussi par une reconstruction du modèle de son corps. La seconde consiste à localiser des personnes, à extraire la forme de ces personnes (ceci est souvent obtenu à l'aide d'algorithmes de segmentation) et à construire des descripteurs à partir de la représentation obtenue. Ce type de modélisation est suffisamment simple pour la mise en oeuvre des applications en temps réel mais reste très dépendant de la pertinence des résultats obtenus à l'étape d'extraction de la silhouette.

— Méthodes basées sur des caractéristiques locales : ces techniques nécessitent l'utilisation de détecteurs capables de construire un ensemble de descripteurs à partir de l'extraction de points d'intérêt. On s'intéressera dans la suite de ce manuscrit à leur applicabilité au type de données spécifiques traitées dans le cadre de notre projet.

— Méthodes basées sur des réseaux de neurones : ces techniques ont donné de très bons résultats sur de nombreux ensembles de données. Elles ont l'avantage de ne nécessiter généralement aucun pré-traitement sur les données. Dans la suite de ce manuscrit, nous considérerons quelques-unes de ces méthodes et les appliqueront au cadre spécifique de notre étude afin d'évaluer leurs performances.

Auteurs	Méthodes	Base de données	Performance
Shao et al. [47]	silhouette + sac de mots sur histogramme des occurrences	Weizmann	100%
Vili et al [50]	MHI + LBP	Weizmann	97,5%
Zheng et al. [51]	silhouette + R-transform	Weizmann	98,8%
Xia et al. [55]	HOJ3D + Modèle de Markov	MSR Action3D	96,2%
Scovanner et al [39]	SIFT3D + sac de mots	Weizmann	82,6%
Arif et al. [70]	LSTM–3D ConvNet	UCF sport	92,9%
Simonyan et Zisserman[9]	Réseau CNN à 2 flux	UCF101 ; HMDB51	91,5% ; 65,9%
Fernando et al. [80]	Modélisation de l'évolution temporelle dans la vidéo	HMDB51	61,5%
Wang et Schmid [33]	Trajectoires denses améliorées	UCF101 ; HMDB51	88,0% ; 57,2%
Wang et al. [10]	Trajectoire denses + CNN	UCF101 ; HMDB51	91,5% ; 65,9%
Wu et al. [64]	CNNs + RNNs	UCF101	90,1%
Martens et al. [61]	RNNs	MSRDailyActivity3D	42%

TABLE 2.2 – Tableau récapitulatif des performances obtenues par les méthodes de l'état de l'art mentionnées dans ce chapitre

Les résultats obtenus dans la littérature portent sur des bases de données toutes différentes de celles que nous exploitons pour nos expérimentations dans ce manuscrit. Les principales différences sont la résolution des images utilisées qui sont plus petites que toutes les autres bases de données, et la position de la caméra qui est en vue de dessus dans notre cas d'étude. Ces remarques importantes montrent toute la complexité et la difficulté qu'il y a de mener une étude comparative avec les bases de données présentes dans l'état de l'art.

Chapitre 3

Analyse d'images du capteur utilisé

Sommaire

3.1 Présentation du capteur

Les détecteurs thermiques sont des transducteurs dans lesquels le rayonnement infrarouge d'un objet est directement transformé en chaleur par absorption [13]. Les variations du flux thermique sur le transducteur permettent d'obtenir une information sous la forme d'un signal électrique. Ce signal électrique est d'autant plus intense que la température est élevée et ce, quelle que soit la gamme spectrale. Ces détecteurs sont composés d'un absorbeur, d'un thermomètre et d'une isolation thermique (voir figure 3.1), ceci afin de permettre la mesure de faibles variations de température moyennant l'un des effets, pyroélectrique, bolométrique, thermoélectrique ou bien encore thermomécanique. Aujourd'hui, si l'on voulait réaliser un système sur la base d'un capteur infrarouge thermique matriciel, il serait difficile d'obtenir des prix de revient bien inférieurs à plusieurs centaines voire milliers d'euros, prix d'une caméra thermique simple à ce jour. Cette équation technico-économique non résolue est à l'origine de la création de la start up Grenobloise

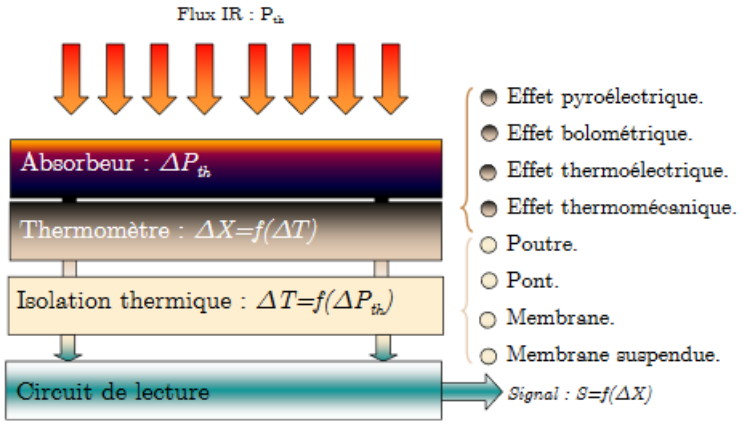


FIGURE 3.1 – Structure fondamentale d'un détecteur thermique [13]

IRLYNX. Elle développe et exploite une technologie Infrarouge dite « pyroélectrique CMOS / PVDF » pour caractériser l'activité humaine. La technologie pyroélectrique est bien connue, elle est utilisée au travers de composants à 1, 2 ou 4 pixels par la quasi-totalité des détecteurs de mouvement pour les automatismes d'éclairage ou les alarmes. Par contre, cette technologie intégrée dans un imageur (capteur capable de retourner des images) est unique au monde. Avec cette technologie de capteur apte à mesurer des variations de température, Irlynx est ainsi capable de délivrer de véritables images thermiques de taille 64 × 64 (voir figure 3.2).

Les potentialités de la détection infra-rouge intelligente offrent des perspectives très prometteuses et l'espoir d'éliminer les inconvénients des technologies actuellement utilisées, notamment le coût et l'intimité dévoilée.

Utiliser une telle technologie présente un certain nombre d'avantages et d'inconvénients. Les principaux avantages de cette technologie sont :

— un très faible coût (environ 30 euros),
— une très faible consommation, compatible avec les autonomies recherchées (le système visé devra permettre d'économiser entre 20% et 30% de l'énergie globale consommée par un système équipé de capteurs de mouvement ou de CO_2 basiques),
— un faible encombrement,
— une capacité intrinsèque à détecter un intrus dans l'obscurité la plus totale,
— une grande accessibilité (technologie éprouvée et répandue).

Quelques inconvénients demeurent cependant :

— l'impossibilité de discriminer entre sources chaudes en mouvement et sources chaudes immobiles mais variant dans le temps,
— la difficulté de discriminer entre hommes et animaux,
— l'impossibilité d'estimer la distance et la vitesse de la cible.

De ces inconvénients découle un certain nombre de contraintes et de limites profondes :

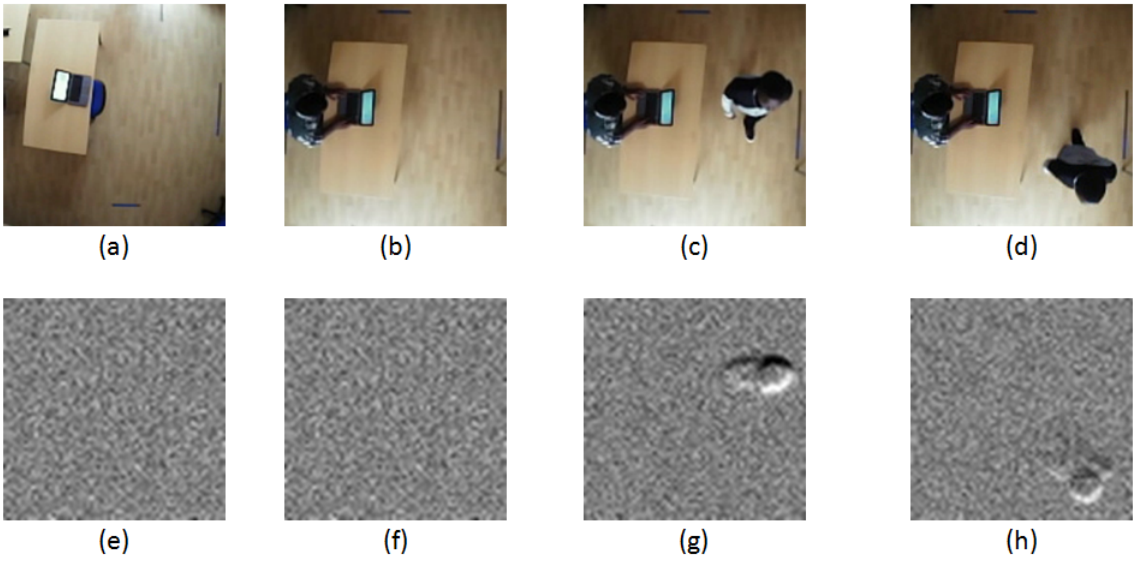


FIGURE 3.2 – La première ligne présente des images acquises avec une caméra visible respectivement sans et avec la présence d'une personne en mouvement dans la scène ; la deuxième ligne présente les images correspondantes acquises à l'aide du capteur infrarouge Irlynx.

- la nécessité de précautions, potentiellement contraignantes, à prendre lors de la pose/installation (présence de convecteurs électriques, baies vitrées,... créant des zones de non-détection),
- la nécessité d'avoir une pièce dont la température ambiante est inférieure à celle de l'objet ou du corps en mouvement,
- la nécessité d'interdire aux animaux domestiques l'accès aux zones protégées par ce type de détecteur,
- l'impossibilité de définir une distance de détection autrement que par les limites physiques du volume surveillé,
- l'interférence possible entre les différents détecteurs présents dans la pièce.

3.2 Rappel du principe physique

La capacité pyroélectrique PVDF présente une variation de charge électrique (tension) proportionnellement à une variation de température. Malheureusement il n'est pas possible de garder en permanence l'information de charge et l'information de la température à cause des courants de fuite du circuit intégré. Le détecteur thermique Irlynx, reposant sur la technologie pyroélectrique PVDF, possède deux éléments sensibles ce qui, lorsqu'une personne passe, va créer un pic positif (appelé abusivement "polarisation") et un pic négatif (appelé abusivement "dépolarisation"). Une lecture séquentielle de la tension aux bornes des détecteurs permettra de savoir quel signal avoir en sortie et donc la direction de la polarisation peut être modifiée suite à une variation de température. Notons que pour une polarisation à tension constante, le signal s'annule. Ce scénario s'observe par exemple pour un bruit de fond ou pour une personne immobile. Dans ces deux cas

le signal est presque nul. Dans la figure 3.3 pour une augmentation (diminution) progressive de la température, le signal de sortie est amplifié, ce qui permet de matérialiser l'apparition (ou la disparition) de la personne. Ce phénomène repose sur le principe de la détection pyro-électrique, c'est-à-dire la détection du déplacement d'un corps chaud présent dans le volume surveillé.

L'information électrique est donc remise à zéro par la charge d'offset à chaque trame de lecture, et l'amplification du signal est faite en différentiel de la charge d'offset.

Le changement de température est généré par le changement de flux infrarouge absorbé par chaque pixel.

Il est donc défini par les éléments suivants de la dynamique de scène :

— Largeur de scène en pixel (distance du capteur / champ de vue)

— Vitesse de déplacement de l'objet émetteur au travers de la scène / pixel

— Nombre de pixels éclairés = longueur (sens du déplacement) × largeur de l'objet émetteur

— Durée d'éclairement de chaque pixel = longueur / vitesse de déplacement de l'objet émetteur

Le changement de charge (signal) suit directement le changement de température, mais la lecture de charge est définie par la dynamique de trame, temps de reset et d'acquisition :

— Fréquence d'acquisition (temps de trame)

— Temps d'acquisition du signal électrique par le pixel = temps trame – temps de reset

Par conséquent l'acquisition de l'information est définie par l'interaction entre la dynamique de scène et la dynamique de lecture de charge. Pour la situation typique d'un déplacement dans une scène de 1 m/s, une fréquence de trame de 10Hz est correcte.

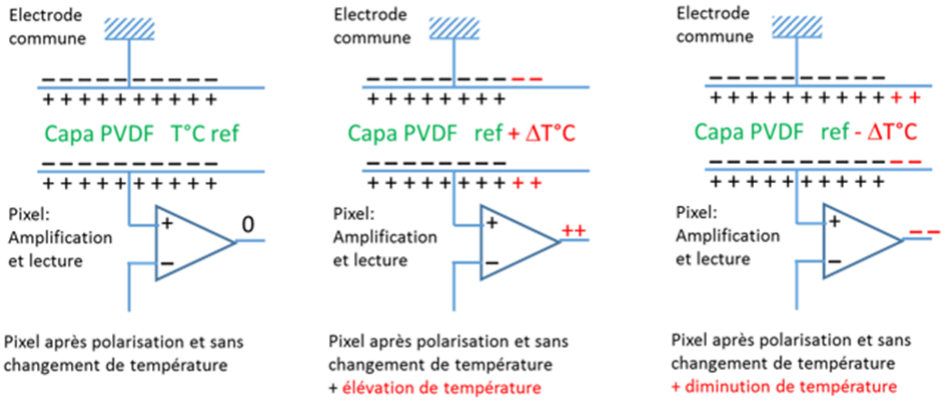


FIGURE 3.3 – Charge électrique en fonction de la température [14]

3.3 Particularités des images

Dans cette section, nous faisons une étude de l'aspect physique des images. Cette étude permettra de mieux comprendre les difficultés qui seront rencontrées lors de l'exploitation des images issues du capteur Irlynx. Elle permettra également de justifier le choix de différentes approches et méthodes qui seront adoptées dans ce chapitre ainsi que dans les chapitres suivants.

3.3.1 Aspect spatial et temporel des images

Selon le principe pyroélectrique, il existe une variation de la charge électrique proportionnelle à la variation de température. Ainsi, lors du déplacement d'un corps chaud, il y a une élévation de température dans la nouvelle région impactée, ce qui entraîne la formation d'une zone claire (visible par exemple sur les images 3.2-(g) et 3.2-(h)) et une chute de température dans l'ancienne région libérée, entraînant la création d'une zone sombre (visible par exemple sur la figure 3.2(g)).

La vitesse de déplacement de l'objet influence la forme des zones détectées (voir figure 3.4). Les patchs obtenus sur l'image réelle peuvent être plus ou moins larges voire couvrir toute l'image si la vitesse de déplacement est très grande.

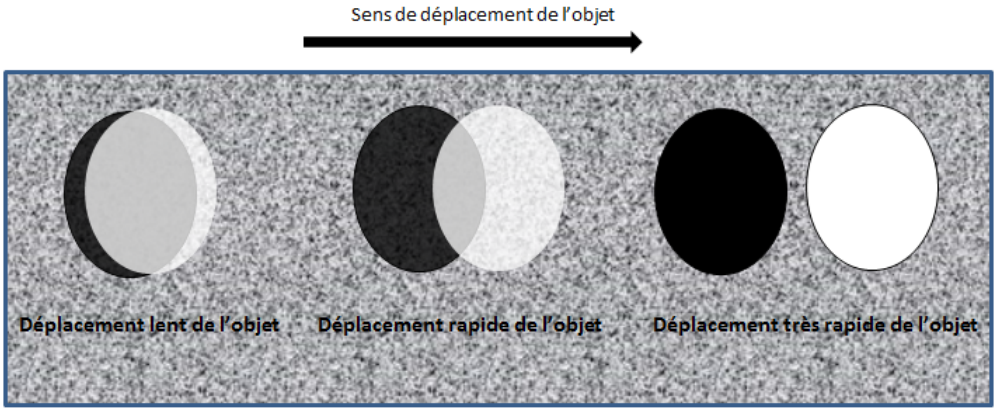


FIGURE 3.4 – Exemple d'images obtenues en fonction de la vitesse de déplacement de l'objet

Si l'on s'intéresse de manière globale aux informations contenues dans les images retournées par le capteur Irlynx, plusieurs constatations sont possibles :

- les histogrammes de niveaux de gris d'une image avec ou sans présence de personne sont similaires (voir figure 3.5-(c) et (d)).
- les attributs statistiques tels que l'écart type, le minimum, le maximum ne permettent pas de différencier une image avec ou sans présence de personne (voir figure 3.5).
- d'un point de vue visuel, on constate qu'une image est formée de trois zones
  - une zone claire correspondant au mouvement de la personne
  - une zone sombre correspondant à la traînée laissée par la personne
  - une zone grise correspondant à l'arrière plan ou plus généralement à toute zone subissant une faible variation de température

Considérons maintenant l'évolution temporelle des niveaux de gris pris par deux pixels issus d'une vidéo dans laquelle une personne traverse une pièce aller et retour en marchant. Le premier pixel considéré, en position (10,10), correspond à une zone de l'image qui n'est pas impactée par le passage de la personne. Le second pixel considéré, en position (45,45), correspond lui à une zone de passage de la personne.

On peut tout d'abord constater (voir figure 3.6) que les niveaux de gris pour un pixel se

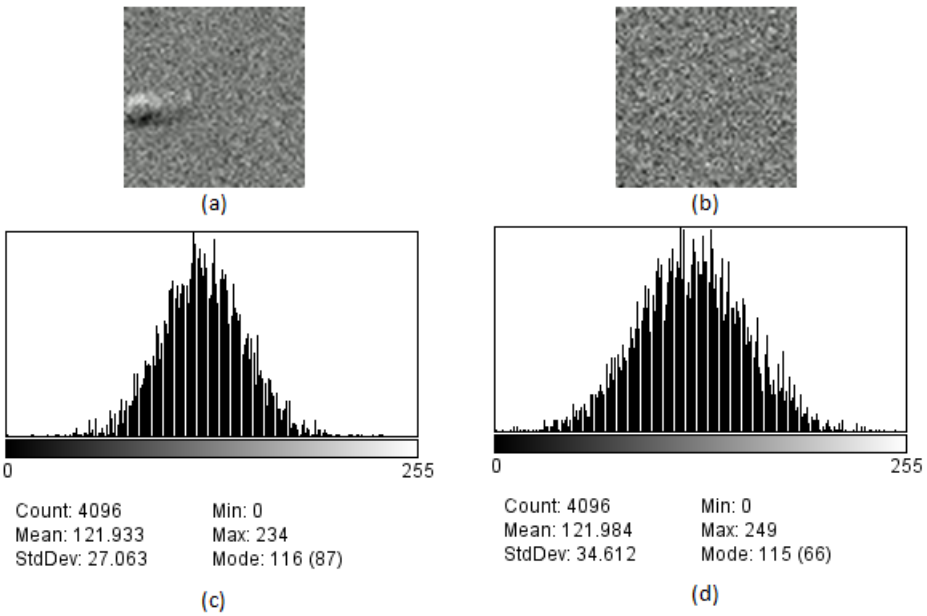


FIGURE 3.5 – Exemples d'histogrammes et de valeurs statistiques pour des images réelles avec et sans présence de personne

trouvant à une même position dans un ensemble d'images varient beaucoup. Sur la figure 3.6-(b), on a en rouge et en vert les pics qui correspondent au passage (entrée et sortie) de la personne au niveau du pixel à la position $(45, 45)$. Ces pics caractérisant la présence d'une personne ne sont pas particulièrement marqués et diffèrent peu de ceux apparaissant alors qu'aucun passage n'est observé. Cela est par exemple illustré au niveau des ellipses noires de la figure 3.6-(b) où les pics ne correspondent pas au passage d'une personne ou encore au niveau des différents pics dans la figure 3.6-(a) alors que le pixel (10,10) appartient à une zone au niveau de laquelle il n'y a aucun mouvement durant toute la vidéo.

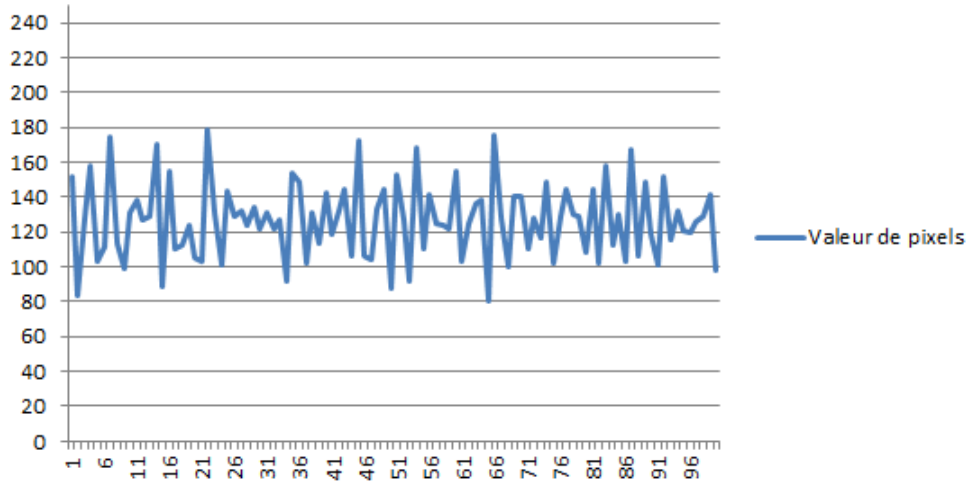
Si l'on applique un filtre sur la vidéo, on constate (voir figure 3.7) que les pics caractérisant le passage d'une personne apparaissent de façon beaucoup plus marquée.

D'après les constatations précédentes nous pouvons dire que la détection du mouvement d'une personne dans des vidéos issues du capteur Irlynx est difficilement réalisable par l'oeil humain et que les images produites sont caractérisées par un fort bruit aléatoire et nécessitent un prétraitement.

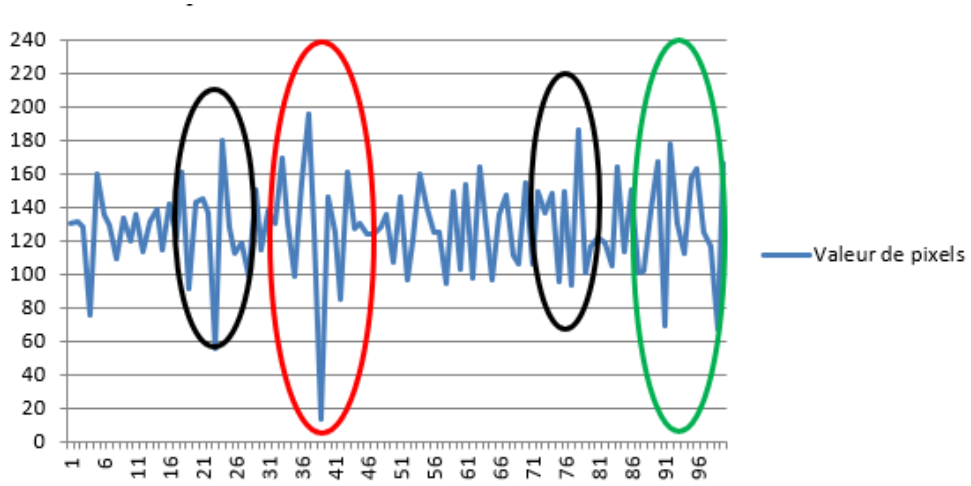
Après cette première analyse du contenu des images issues du capteur Irlynx, on peut également se poser la question de l'efficacité de méthodes par recherche de zones et points d'intérêts. L'étude qui suit permettra de guider les choix qui seront faits ultérieurement pour la proposition d'approches dédiées à la reconnaissance d'actions dans le contexte particulier qui est le nôtre.

3.3.2 Détection de caractéristiques locales dans les images

De nombreuses méthodes de reconnaissance d'actions de la littérature utilisent le flux optique ou encore des caractéristiques extraites autour de points d'intérêts comme entrées du modèle de



(a) Évolution des valeurs des pixels en position (10,10) dans une vidéo contenant des images originales

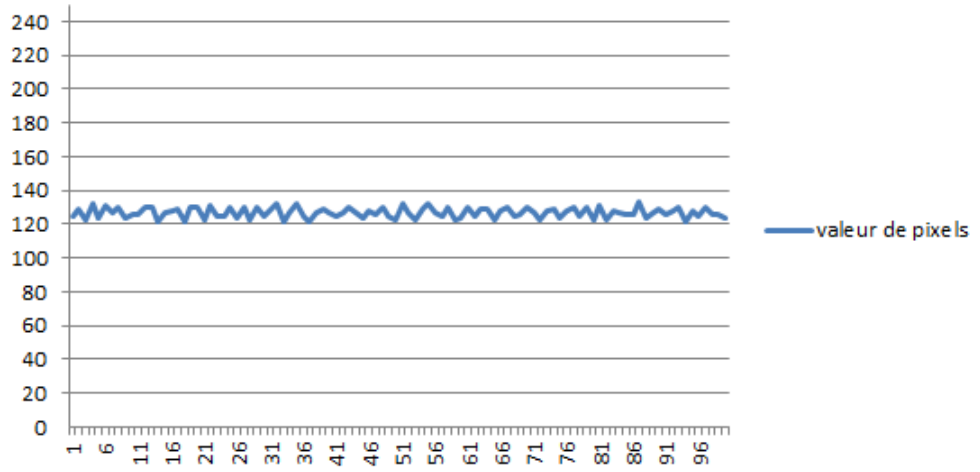


(b) Évolution des valeurs des pixels en position (45,45) dans une vidéo contenant des images originales

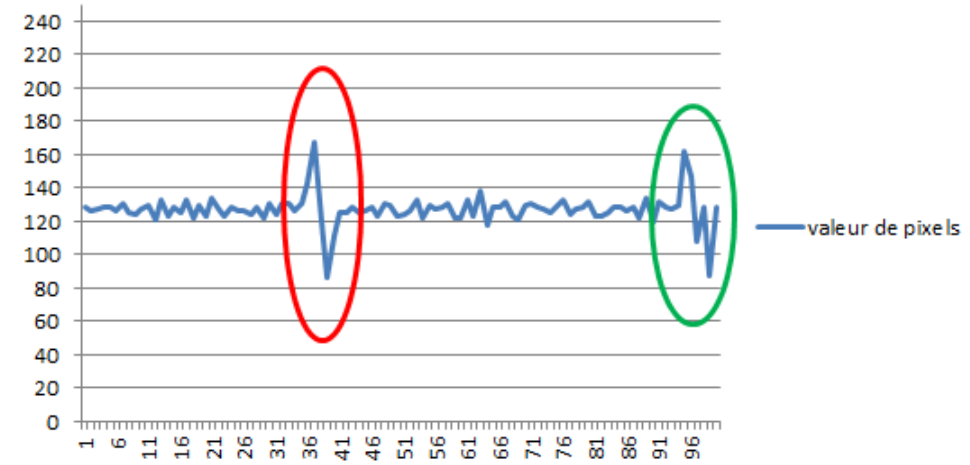
FIGURE 3.6 – Niveaux de gris pour deux pixels d'une vidéo originale sans (position (10,10)) et avec (position (45,45)) passage d'une personne

reconnaissance. Ces méthodes sont souvent utilisées car elles permettent de bien caractériser les mouvements et sont invariantes par rapport aux changements d'échelle et de vitesse. De plus, elles ne nécessitent pas la suppression du fond et ne requièrent pas un suivi et/ou un modèle explicite du corps humain.

Des méthodes de reconnaissance d'actions de la littérature basées sur des descripteurs comme SIFT 3D ou en encore Harris 3D ont donné de très mauvais résultats sur notre base de données. Dans l'optique d'expliquer ces mauvaises performances, des résultats obtenus par utilisation du flux optique ou par détection de points d'intérêts sur des images originales et filtrées sont présentés à titre d'exemple sur les figures 3.8 et 3.9. On constate que les points détectés se trouvent dans des endroits ne correspondant pas aux zones d'intérêts. Cela montre que les méthodes de reconnaissance



(a) Évolution des valeurs des pixels en position (10,10) dans une vidéo contenant des images filtrées

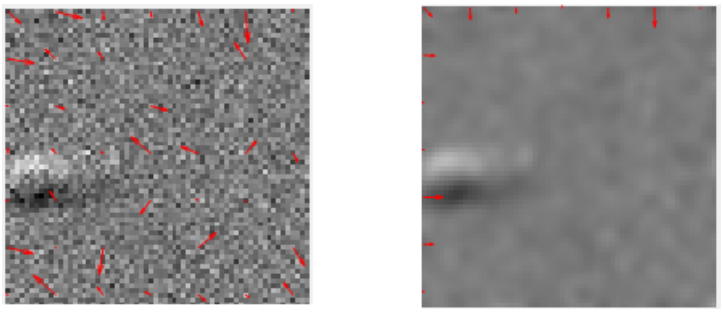


(b) Évolution des valeurs des pixels en position (45,45) dans une vidéo contenant des images filtrées

FIGURE 3.7 – Niveaux de gris pour deux pixels d'une vidéo filtrée sans (position (10,10)) et avec (position (45,45)) passage d'une personne

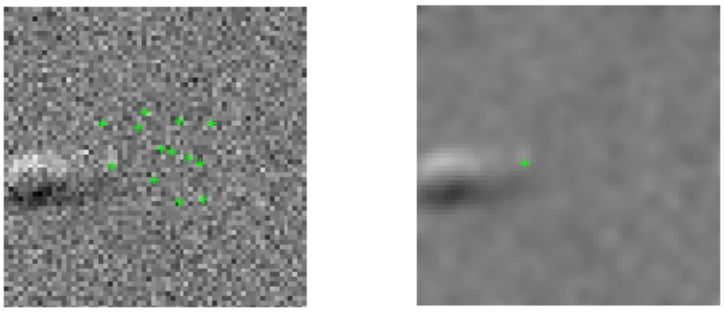
basées sur le flux optique ou sur les points d'intérêt spatiaux temporels ne sont pas de bonnes approches pour ce type de données. En effet, la texture au niveau de l'objet en mouvement est identique à celle de l'arrière plan. Or, ce qui caractérise le point d'intérêt c'est la variabilité de sa texture autour de ce point [81].

En plus des approches basées les caractéristiques locales, on trouve aussi dans l'état de l'art des approches basées sur l'apparence et plus précisément sur l'analyse des silhouettes. Ces approches nécessitent au préalable une bonne extraction de la silhouette de l'humain dans les images et ont recours à la segmentation. Pour obtenir une bonne extraction de la silhouette, un ensemble d'étapes est nécessaire.



(a) image originale (b) image filtrée

FIGURE 3.8 – Résultat du flux optique : à gauche sur une image originale et à droite sur une image filtrée



(a) image originale (b) image filtrée

FIGURE 3.9 – Visualisation des points d'intérêts détectés en utilisant l'algorithme SURF

3.4 Analyse d'images pour l'extraction de silhouettes

L'analyse d'images désigne l'ensemble des opérations qui consiste à extraire les informations pertinentes en regard de l'application concernée, les traiter puis les interpréter. Dans cette partie, nous allons appliquer successivement quelques méthodes de filtrage, segmentation et post-traitement pour déterminer les meilleures pour l'application souhaitée.

3.4.1 Filtrages

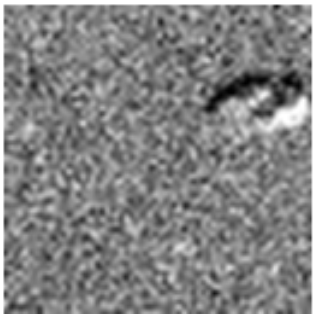
Le filtrage d'image est une technique de pré-traitement qui vise à diminuer les détails non significatifs et le fort bruit aléatoire qui peut caractériser les images afin de faciliter les étapes qui suivront pour la reconnaissance d'actions ou d'objets. Filtrer une image va consister à modifier la valeur des pixels d'une image, généralement dans le but d'améliorer son aspect. En pratique, il s'agit de créer une nouvelle image en se servant des valeurs des pixels de l'image d'origine afin que l'image filtrée soit adaptée pour une application spécifique.

Afin d'améliorer la qualité des images utilisées, nous nous sommes intéressés à quelques méthodes de filtrage de la littérature, ainsi qu'à l'impact de ces techniques vis à vis de l'objectif

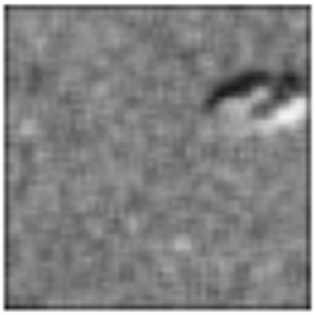
poursuivi d'extraction pertinente du motif de l'humain dans les images.

Filtre median

Le filtre médian consiste en une opération de lissage non linéaire réalisé en parcourant l'image avec une fenêtre d'observation et en remplaçant la valeur du pixel central par la valeur médiane de ses pixels voisins. Notons que plus la taille du voisinage (appelé aussi taille de la fenêtre) est grande, plus le lissage est fort et donc plus les détails de l'image sont atténués (voir figure 3.10-(d)). La figure 3.10 présente des résultats du filtre médian avec des tailles de fenêtre différentes.

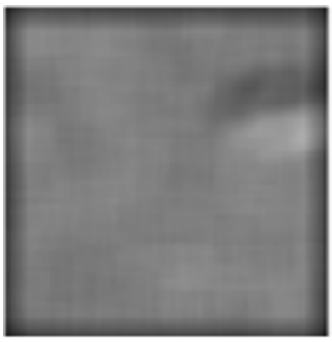


(a) Image originale



(b) Image filtrée : fenêtre 3×3

(c) Image filtrée : fenêtre 5×5



(d) Image filtrée : fenêtre 7×7

FIGURE 3.10 – Exemples de résultats obtenus par application du filtre médian

Filtre Gaussien

Le filtrage gaussien est efficace pour lisser les images (réduire les bruits) mais peut avoir l'inconvénient de ne pas préserver la luminosité de l'image. Il fonctionne en utilisant la distribution 2D comme une fonction d'étalement ponctuel. Il est obtenu en convoluant la fonction de distribution 2D gaussienne (voir équation 3.1) sur l'image. Le paramètre σ correspond à la largeur du filtre et à l'écart type de la fonction gaussienne utilisée. Un exemple de noyau de convolution de taille 3×3 est présenté dans la figure 3.11

$$G(x, y) = \frac{1}{2\pi\sigma} e^{-}(\frac{x^2 + y^2}{2\sigma^2}) \tag{3.1}$$

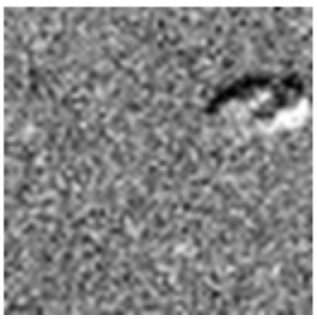
G (-1,-1)	G (0,-1)	G (1,-1)
G (-1,0)	G (0,0)	G (1,0)
G (-1,1)	G(0,1)	G (1,1)

$$\simeq \frac{1}{16}$$

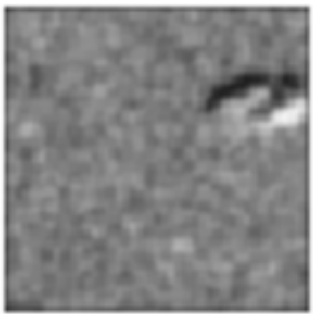
1	2	1
2	4	2
1	2	1

FIGURE 3.11 – Exemple de noyau de convolution du filtre gaussien de taille 3×3 avec $\sigma = 0.8$

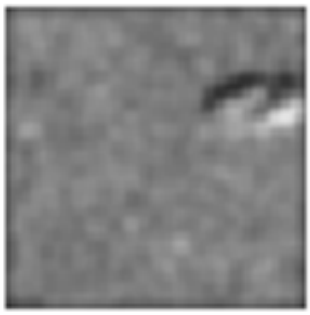
La figure 3.12 présente des résultats du filtre gaussien en fonction du paramètre σ. On constate que plus la valeur de σ est élevée plus l'image est floue. Par conséquent plus les détails sont atténués.



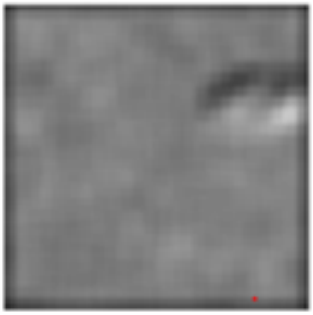
(a) Image originale



(b) Image filtrée : $\sigma = 0.9$



(c) Image filtrée : $\sigma = 1.4$



(d) Image filtrée : $\sigma = 2.7$

FIGURE 3.12 – Exemples de résultats par application du filtre gaussien

Filtre de Nagao

Le principe de ce filtre est de remplacer chaque pixel de l'image par la valeur moyenne des pixels contenus dans une fenêtre particulière. Dans une fenêtre 5x5 centrée sur le pixel à modifier, neuf sous-fenêtres sont définies comme illustré dans la figure 3.13. Le choix de la fenêtre particulière pour changer la valeur du pixel est celle qui présente la plus petite valeur de variance.

Sur la figure 3.14 est présenté un exemple de résultat du filtre de Nagao. On constate que pour le filtre de Nagao les contours sont bien conservés mais ce filtre pose un problème d'effet de bloc (apparition de discontinuités aux frontières entre les blocs adjacents).

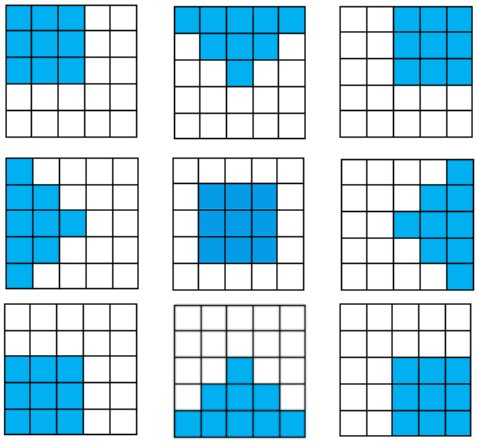
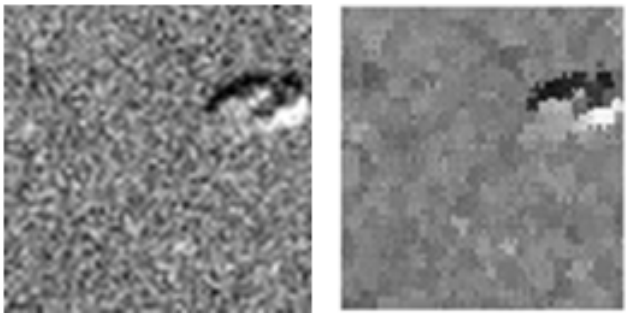


FIGURE 3.13 – Les 9 fenêtres utilisées pour le filtre de Nagao



(a) Image originale (b) Image filtrée

FIGURE 3.14 – Exemple de résultat obtenu par application du filtre de Nagao

Filtre de Perona & Malik

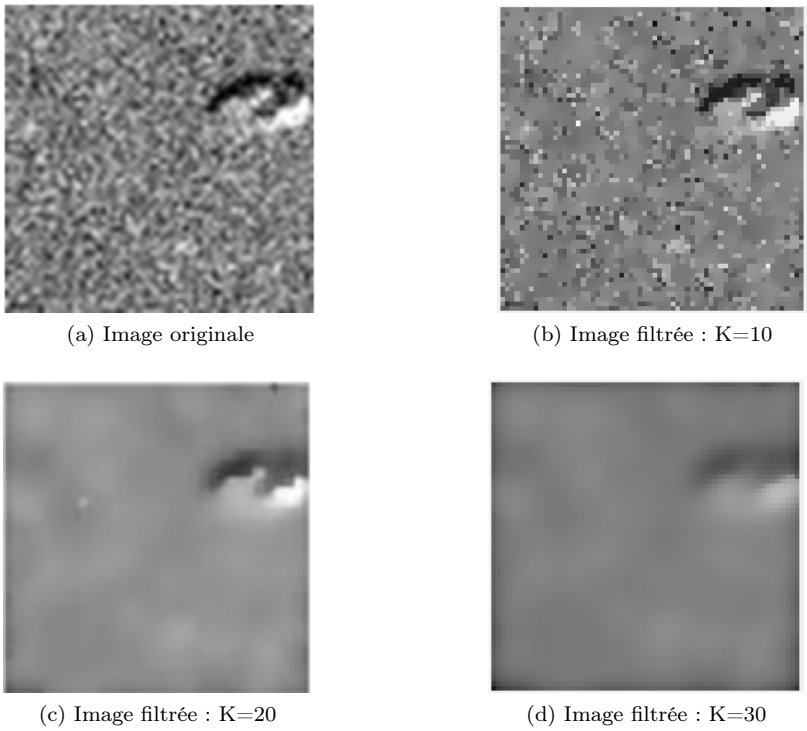
Pour préserver les discontinuités pertinentes, tout en continuant à réduire le bruit, il est nécessaire d'adapter l'intensité de diffusion en fonction de la zone. À cette fin, Perona et Malik ont proposé un modèle de diffusion non linéaire en utilisant des dérivées de premier ordre. Cela permet de lisser les zones homogènes de l'image tout en préservant les contours des objets. Ce modèle repose sur l'utilisation d'une diffusivité thermique variable et il a largement été étudié dans le cas bidimensionnel. Le coefficient c constant dans l'équation de chaleur (équation 3.2) est remplacé par une fonction de diffusion (fonction gaussienne (équation 3.3) ou fonction de Lorentz (équation 3.4)) afin de limiter la diffusion lorsque le gradient est élevé et inversement. Le modèle est donné par (voir équation 3.2) :

$$\frac{du}{dt} = div(c.\nabla u) \tag{3.2}$$

$$g(|\nabla u|) = e^{-(\frac{\nabla u}{K})^2}(fonction\ \ gaussienne) \tag{3.3}$$

$$g(|\nabla u|) = e^{\left(\frac{1}{(1+\frac{\nabla u}{K})^2}\right)} (fonction\ \ lorentzienne) \tag{3.4}$$

Le paramètre K est appelé barrière ou seuil de diffusion. Comme son nom l'indique, il sert à fixer la limite entre les forts gradients correspondant aux transitions à maintenir et les faibles gradients correspondant au bruit. Le choix de la barrière de diffusion K est crucial. En effet, c'est elle qui permet de définir quels contours seront rehaussés et donc maintenus. Un choix de barrière de diffusion K élevée provoquera un lissage de tous les contours. Dans le cas contraire, une valeur plus faible induira un rehaussement de contraste pour tous les contours, y compris ceux correspondant au bruit. Nous avons appliqué ce filtre en employant la fonction de diffusion lorentzienne. Les paramètres à fixer sont le nombre d'itérations, dt et K.



(a) Image originale

(b) Image filtrée : K=10

(c) Image filtrée : K=20

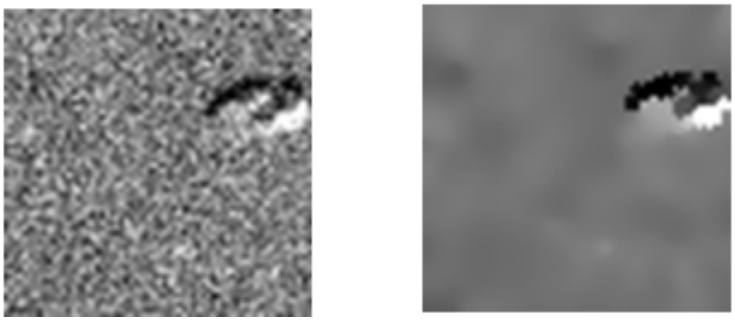
(d) Image filtrée : K=30

FIGURE 3.15 – Exemples de résultats obtenus par application du filtre Perrona & Malik en fixant : nombre d'itérations=15, dt=1/6 et en faisant varier K

Filtre Nagao + Malik

Partant du constat que le filtre Nagao a la particularité de préserver les contours mais qu'il y a apparition de discontinuités aux frontières entre les blocs adjacents, et que le filtre de Perona & Malik homogénéise les régions, nous avons combiné les deux filtres. Après avoir appliqué le filtre de Nagao, le résultat du filtre est pris comme entrée du filtre de Perona & Malik. Le filtre de Perona & Malik aura pour fonction de corriger les imperfections du filtre de Nagao. Ce nouveau filtre, qui

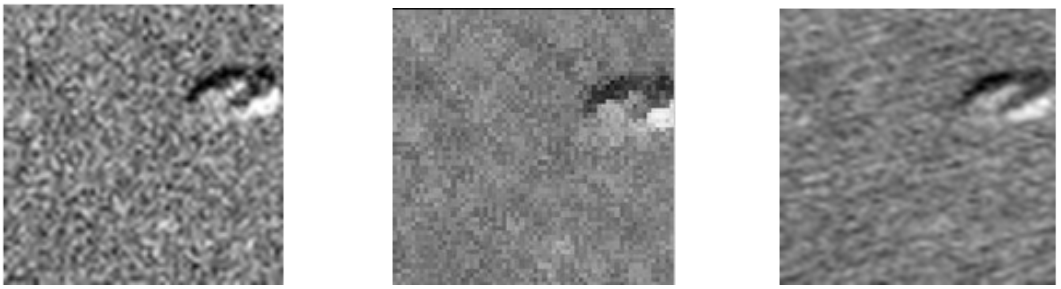
d'un point de vue visuel donne des résultats assez satisfaisants (cf figure 3.16), sera nommé filtre Nagao + Malik tout au long de ce chapitre.



(a) Image originale (b) Image filtrée

FIGURE 3.16 – Exemple de résultat obtenu par application du filtre Nagao + Malik

D'autres méthodes de filtrage ont été testées à savoir le filtre SNN (Symmetric nearest neighbors filter), le filtre Wiener. La figure 3.17 présente un exemple de résultat pour ces deux filtres. L'aspect visuel de ces résultats montre que les bruits se trouvant en arrière plan n'ont pas été clairement atténués et donc ils ne seront pas utilisés dans l'étude comparative.



(a) Image originale (b) Image filtrée par le filtre SNN (c) Image filtrée par le filtre Wiener

FIGURE 3.17 – Quelques autres résultats de filtrage

Un des problèmes les plus importants que pose le processus de filtrage est celui de l'évaluation de la qualité du résultat de filtrage. Nous rappelons que la qualité de l'image est une notion très liée à la perception visuelle humaine. Évaluer la qualité d'une image c'est lui associer un ou plusieurs critères permettant de situer sa position relative dans un référentiel défini et selon l'application envisagée. L'analyse des performances d'un filtre demeure une tâche difficile. La plupart du temps, on utilise des métriques d'évaluation pour le faire, métriques qui sont généralement regroupées selon les caractéristiques qu'elles mesurent : rapport signal bruit, préservation de détails, fidélité d'information et corrélation des pixels. Étant donné que les valeurs retournées par les métriques ne permettent pas toujours de justifier que l'information utile est conservée dans l'image filtrée, nous avons pris le parti dans ce mémoire d'évaluer les filtres en fin de chaîne d'extraction du motif

de l'humain. L'évaluation consistera donc à chercher le couplage filtre/segmentation qui permet de mieux extraire la silhouette de la personne dans les images.

3.4.2 Segmentation

Généralités

La segmentation d'images est une tâche importante présente dans de nombreuses applications de vision par ordinateur. Elle vient généralement après l'étape de débruitage. La segmentation, qui est un traitement bas-niveau, est définie comme étant un processus de partitionnement de l'image en régions homogènes où chacune d'elles regroupe un ensemble de pixels présentant des propriétés communes.

Dans le processus de reconnaissance d'actions basé sur l'analyse de silhouette, l'étape de segmentation joue un rôle crucial conditionnant les étapes ultérieures telles que la classification, le suivi des objets et la reconnaissance de l'action ou de la posture. Nous nous intéressons ici à une segmentation binaire. Cette segmentation consiste à subdiviser l'image en deux régions : la classe des pixels appartenant à la personne en mouvement et celle des pixels qui appartiennent à l'arrière-plan.

Proposer un état de l'art des méthodes de segmentation d'images est une tâche délicate, au vu du nombre d'articles parus dans la littérature durant ces quarante dernières années. La plupart des méthodes ont été développées dans le cadre d'applications spécifiques. En vertu de cette considération, nous nous intéressons ici aux méthodes de segmentation les plus couramment utilisées lorsqu'un problème de soustraction d'arrière-plan se pose.

Les caractéristiques des images provenant du capteur nous ont permis d'exclure les approches de segmentation utilisant des modèles simples de différence interframe, ou celles basées sur le flux optique car la valeur de tous les pixels varie d'une image à l'autre. Ainsi les approches de soustraction d'arrière plan semblent bien adaptées pour ce type de contexte. Le problème abordé par les techniques de soustraction d'arrière plan réside dans la construction de ce qu'on appelle le modèle d'arrière-plan, qui permettra de diviser l'image observée en deux ensembles complémentaires de pixels couvrant toute l'image : (1) les connaissances nouvelles qui contiennent les objets d'intérêt (la personne), et (2) l'arrière-plan. De nombreuses techniques de soustraction d'arrière-plan ont été proposées avec autant de modèles et de stratégies de segmentation, et plusieurs articles sont consacrés à ce sujet (voir par exemple [82] [83] [84]).

Nous classons les approches de soustraction d'arrière-plan en 3 grandes familles à savoir :

— Modèles basiques : ces approches consistent à construire une image dite d'arrière-plan et pour chaque image de la vidéo une différence pixel par pixel est faite entre l'image courante et l'image d'arrière plan afin d'obtenir l'image segmentée. La différence entre les approches relevant de cette famille réside dans les méthodes d'initialisation et de mise à jour de l'arrière-plan. Dans [85], les auteurs construisent l'arrière-plan en faisant la moyenne de l'ensemble des pixels à la même position. D'autres auteurs [86] utilisent des caractéristiques de texture

ou les histogrammes de couleurs et contours [87].

— Modèles statistiques : la plupart des modèles statistiques utilisent une distribution gaussienne pour séparer les différentes régions de l'image. Ceci est fait en déterminant les valeurs seuils optimales en estimant les probabilités a posteriori. En fonction de la complexité de la situation considérée, il existe des modèles de mélanges de simples gaussiennes [88] ainsi que des mélanges de gaussiennes généralisées (GGMM) [89]. Enfin des modèles statistiques non paramétriques [90, 91] peuvent également être considérés. .

— Modèles flous / Méthodes neuronales et neuro-floues : la plupart des méthodes qui ont été proposées dans cette catégorie souffrent d'une grande complexité et d'une faible vitesse de traitement par rapport à leurs homologues. Les modèles flous permettent d'utiliser un système basé sur des règles pour l'évaluation du critère d'homogénéité entre région et de considérer des connaissances a priori pour améliorer l'appartenance d'un pixel à une région [92] [93, 94]. Les modèles neuronaux permettent de produire des segmentations précises et détaillées [95, 96]. Ils sont construits en adaptant les réseaux de classification moderne (AlexNet [97], réseau VGG [98] et GoogLeNet [99]) en réseaux entièrement convolutifs et en ajustant leurs représentations apprises dans le but d'atteindre de meilleures performances pour la tâche de segmentation [100].

Un algorithme de segmentation doit remplir des exigences spécifiques pour être caractérisé d'idéal. Selon [101], une technique de soustraction d'arrière-plan doit s'adapter aux changements d'éclairage progressifs ou rapides (changement de moment de la journée, nuages, etc.), à l'apparition de différents mouvements (oscillations de la caméra, présence de feuilles ou de branches d'arbres) et aux changements du décor (voitures stationnées).

Dans les paragraphes qui suivent, nous décrivons 5 approches de segmentations récentes, robustes, ayant de bons résultats sur différentes bases et nécessitant peu de ressources (temps d'exécution, mémoire) [102] ainsi que l'approche de segmentation proposée pour le type de capteur infrarouge considéré dans cette étude. Pour chaque méthode de segmentation considérée, nous présentons brièvement son principe de fonctionnement ainsi que ses différents paramètres.

3.4.3 Quelques méthodes de segmentation

Algorithme de soustraction d'arrière plan adaptatif

Pour construire un modèle d'arrière-plan de la scène il suffit d'accumuler l'information contenue dans la séquence d'images, mettant ainsi progressivement en évidence les parties immobiles de la scène et effaçant ce qui bouge. Ici le modèle d'arrière plan est construit en calculant la valeur médiane prise par les pixels situés à une même position dans la séquence d'images considérée. Une fois l'arrière plan déterminé, une différence entre l'image courante et le modèle d'arrière plan construit est calculée pour obtenir l'image segmentée : $Dt = |Mt - It|$ (avec Dt : l'image segmentée, Mt : l'arrière plan construit et It : l'image courante). Un intérêt majeur de cette technique est sa capacité à détecter des cibles très petites ou qui bougent lentement.

Algorithme VIBE

La méthode VIBE [15] consiste à modéliser chaque pixel d'arrière-plan par un ensemble de N valeurs prélevées sur des images précédentes. Soit $v(x)$ la valeur dans un espace colorimétrique euclidien donné prise par le pixel situé à la position x dans l'image et v_i une valeur d'échantillon de fond. Chaque pixel de fond à la position x est modélisé par une collection de N valeurs constituant l'échantillon $M(x) = \{v_1, v_2, ..., v_N\}$. Ces N valeurs de pixel et celle de l'image courante sont placées dans un même espace colorimétrique (voir figure 3.18). Le pixel de l'image courante est classé comme pixel appartenant à l'arrière-plan si dans un cercle de rayon R centré sur ce pixel, le nombre d'éléments à l'intérieur est supérieur à un seuil défini. La méthode met à jour l'ensemble des N pixels en choisissant au hasard les valeurs à substituer dans le modèle de fond. Cette approche diffère des autres par la croyance classique selon laquelle les valeurs les plus anciennes doivent être remplacées en premier.

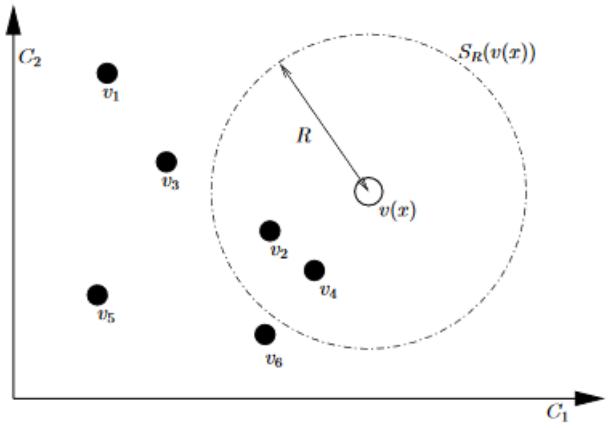


FIGURE 3.18 – Descriptif de décision d'appartenance d'un pixel se trouvant dans l'espace colorimétrique (C1,C2) à l'arrière-plan [15]

Algorithme W4*

La méthode W4 présentée dans [103] est une méthode qui utilise trois valeurs pour classer un pixel d'arrière-plan à savoir : la valeur du pixel d'intensité minimale, maximale et la différence d'intensité maximale entre les images consécutives de la séquence d'entraînement. Pour classer un pixel la formule 3.5 est utilisée. Dans [104], une amélioration est apportée au modèle W4 avec l'ajout d'un modèle de différence inter-frame. Nous appelons cette méthode W4* dans ce manuscrit. Les étapes de l'algorithme W4* sont présentées dans la figure 3.19.

$$B(x,y) = \begin{cases} 0 & si \ \ |P(x,y) - m(x,y)| < kd\mu \ \ ou \ \ |P(x,y) - n(x,y)| < kd\mu \\ 1 & sinon \end{cases} \tag{3.5}$$

$P(x,y)$ est la valeur de niveau de gris du pixel de l'image courante à la position (x,y), $n(x,y)$ et $m(x,y)$ sont respectivement les valeurs d'intensité minimale et maximale du pixel situé à cette

même position (x, y) dans la vidéo. $d\mu$ est la valeur moyenne de la différence d'intensité maximale et minimale entre un ensemble d'images consécutives.

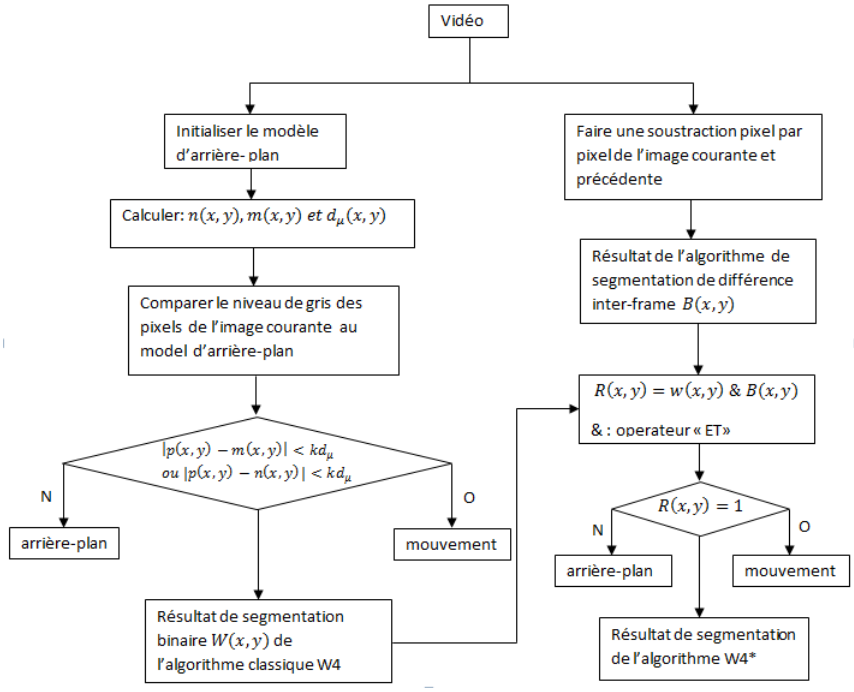


FIGURE 3.19 – Diagramme d'exécution de l'algorithme W4*

Algorithme KDE

L'algorithme KDE (Kernel Density Estimation)[91] est une technique de soustraction d'arrière-plan qui modélise l'arrière-plan en utilisant une fonction de densité et en exploitant les informations historiques récentes. En effet pour chaque pixel de l'image une probabilité est calculée et en fonction d'un seuil on classe le pixel comme mouvement ou non. Elle est appliquée en utilisant la formule 3.6. Si $Pr(x_t)$ est inférieur à un seuil, le pixel est un pixel d'arrière-plan.

$$Pr(x_t) = \frac{1}{N} \sum_{i=1}^{N} \prod_{j=1}^{d} \frac{1}{\sqrt{2\pi\sigma_j^2}} e^{\frac{(x_{tj} - x_{ij})^2}{\sigma_j^2}} \tag{3.6}$$

où x_t est la valeur d'intensité d'un pixel à l'instant t ; x_{tj} est la valeur du pixel à l'instant t dans le canal de couleur j ; N est le nombre d'échantillons considéré ; σ est la largeur de bande de la fonction noyau et d est le nombre de différents canaux de couleur dans l'image.

La fonction de densité noyau utilisée est une généralisation du modèle de mélange gaussien, où chaque échantillon unique parmi les N échantillons est considéré comme une distribution gaussienne.

Algorithme SUBSENCE

Dans la méthode SUBSENCE [16], les pixels sont modélisés à l'aide de caractéristiques spatio-temporelles plus exactement en utilisant le modèle de similarité binaire locale LBSP [105] (Local Binary Similarity Patterns). La méthode SUBSENCE est inspirée des méthodes VIBE [15] et PBAS [106]. PBAS est une méthode de segmentation qui utilise une stratégie permettant de contrôler la dynamique des valeurs des pixels d'arrière-plan. Son utilisation permet d'identifier et de traiter les régions instables (c'est à dire les régions que le modèle ne peut pas modéliser correctement) en réajustant les seuils de distance. Cette adaptation utilise des mécanismes de rétroaction à l'échelle locale. Concernant l'exploitation de la méthode VIBE, au lieu de projeter le pixel dans un espace colorimétrique, les caractéristiques LBSP de chaque pixel sont calculées, ce qui donne des échantillons portant à la fois l'intensité des couleurs locales et des informations spatio-temporelles. Pour une sensibilité accrue, la méthode comprend des modules qui permettent de mettre à jour les seuils et d'adapter dynamiquement les paramètres. Il s'agit d'offrir une solution adaptative s'affranchissant des longs réglages de paramètres communs à bon nombre d'algorithmes de segmentation. Une vue d'ensemble de la méthode est présentée sur la figure 3.20.

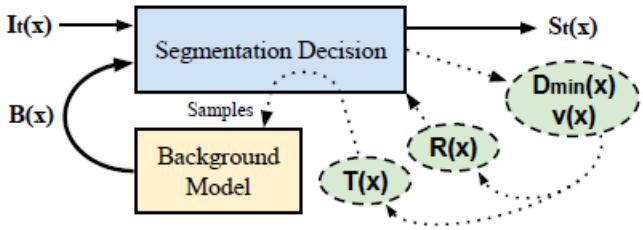


FIGURE 3.20 – Vue d'ensemble des principaux composants de SUBSENSE ; les lignes en pointillés indiquent les mécanismes de rétroaction. Dans ce contexte, $I_t(x)$ porte la représentation LBSP/RGB de x obtenue à partir de la trame de la séquence analysée, $B(x)$ contient N échantillons de fond récemment observés, $S_t(x)$ est la valeur de sortie de segmentation, $R(x)$ contrôle les seuils de décision de segmentation, $T(x)$ contrôle les seuils de mise à jour en arrière-plan, et enfin, à la fois $Dmin(x)$ et $v(x)$ contrôlent dynamiquement les variables précédentes en surveillant la dynamique de fond [16]

Algorithme Sigma-Delta

L'algorithme Sigma-delta [107] est une méthode de segmentation basée sur le filtre de détection de mouvement [108] $\Sigma - \Delta$ (sigma-delta). Comme dans le cas des convertisseurs analogique-numérique, un filtre de détection de mouvement sigma-delta consiste en une simple approximation récursive non linéaire de l'image de fond, qui est basée sur une comparaison et sur un incrément / décrément élémentaire (généralement -1, 0 et 1 sont les seules valeurs d'actualisation possibles). Le filtre de détection de mouvement sigma-delta convient parfaitement à de nombreux systèmes embarqués [109].

Algorithme de segmentation proposé

Une analyse détaillée des images nous a permis d'appréhender les facteurs qui pourraient permettre d'obtenir une bonne segmentation pour notre type de données. En analysant les histo-

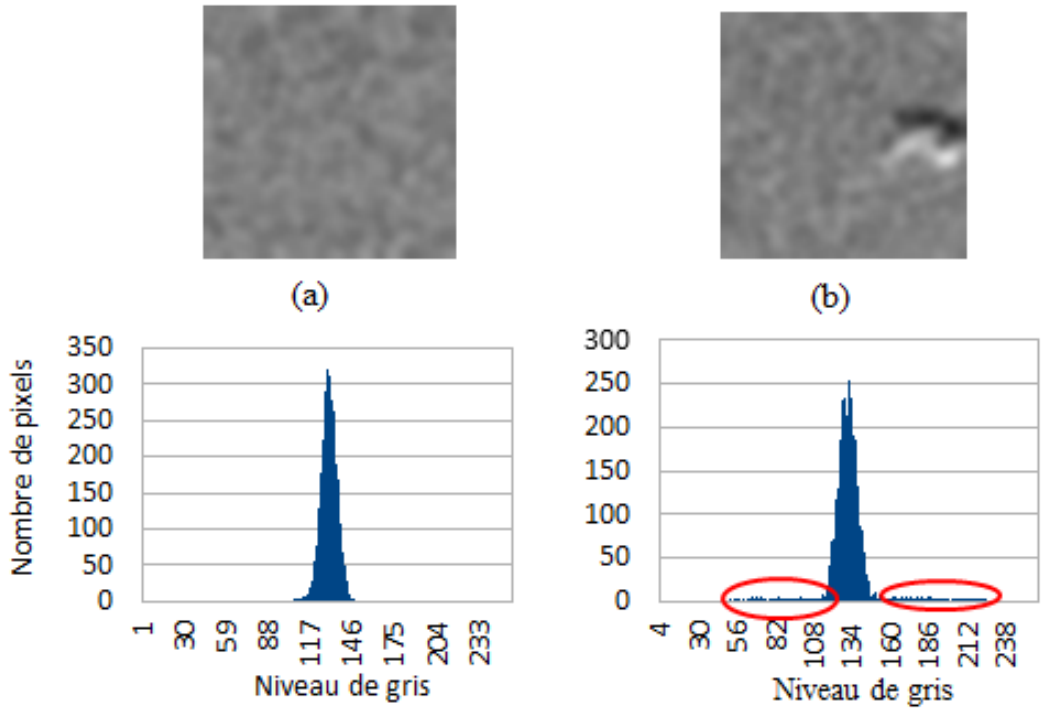


FIGURE 3.21 – Histogrammes des niveaux de gris

grammes des niveaux de gris des pixels (voir figure 3.21-a) pour une image sans présence d'objet ou de personne, on observe une concentration des niveaux de gris des pixels entre 100 à 150. Par contre il y a étalement des pixels sur les 255 niveaux de gris pour l'image contenant une personne (voir figure 3.21-b). Cela montre que l'approche classique de segmentation par seuillage ou de modélisation d'arrière-plan par une gaussienne peut permettre de séparer correctement l'objet et l'arrière-plan.

La méthode proposée est un algorithme de segmentation par seuillage. Elle consiste d'abord à rechercher la plage des pixels correspondant à l'arrière plan. Cela revient à trouver deux paramètres de seuils sur les niveaux de gris qui constitueront la borne inférieure et la borne supérieure. Ces seuils sont ensuite exploités pour segmenter l'image (voir Algorithme 1 et 2).

De plus, en considérant les informations spatiales des images, les attributs tels que l'écart type et l'étendue (écart entre l'intensité la plus élevée et l'intensité la moins élevée prises par les pixels de la région considérée) peuvent présenter des différences considérables. Ces caractéristiques peuvent être exploitées pour accroitre la capacité de discrimination des régions de mouvement et d'arrière-plan et mieux classer un pixel.

Algorithme 1 : Algorithme de segmentation avec seuil adaptatif

entrée : Image filtrée I_t ;
sortie : Image segmentée I_{seg} ;
calculer l'écart type des valeurs des pixels de l'image ;
si *écart type* < 4 **alors**
  pas d'objet dans l'image
sinon
  calculer le vecteur contenant l'histogramme des niveaux de gris (H) ;
  // recherche des paramètres de seuil ;
  $(borne\_inferieure, borne\_superieure) = rechercher\_parametre(H)$;
  pour *tous les pixels dans l'image* **faire**
    si $(I_t(i,j) < borne\_inferieure$ *ou* $I_t(i,j) > borne\_superieure)$ **alors**
      $(I_{seg}(i,j) = 255$;
    sinon
      $(I_{seg}(i,j) = 0$;
    fin
  fin
fin
retourner I_{seg} ;

La fonction "*rechercher_parametre*" permet de déterminer les paramètres de seuils (borne inférieure et supérieure) qui seront utilisés par l'algorithme de segmentation (voir Algorithme 1).

Algorithme 2 : Recherche des paramètres de seuils

entrée : vecteur (H) contenant le nombre de pixels correspondant à chaque niveau de gris dans l'image (l'histogramme) ;
sortie : borne inférieure et bonne supérieure ;
constante $seuil = 15$;
$i = 1$;
$j = taille(H)$;
tant que $(i < \frac{taille(H)}{2})$ *et* $(H(i) <= seuil)$ **faire**
  $i = i + 1$;
fin
si $H(i) >= seuil$ **alors**
  $borne\_inferieure = i$;
fin
tant que $(j > \frac{taille(H)}{2})$ *et* $(H(j) <= seuil)$ **faire**
  $j = j - 1$;
fin
si $H(j) >= seuil$ **alors**
  $borne\_superieure = j$;
fin

3.4.4 Post-traitement

L'approche de post-traitement que nous proposons (voir figure 3.23) utilise les résultats des algorithmes de segmentation pour améliorer la détection. L'amélioration consiste à essayer de reconstruire une forme, un contour proche de celui observé dans l'image initiale et aussi à éliminer les

bruits. Pour y parvenir, nous utilisons des informations spatiales d'une région d'intérêt pour mieux classer un pixel comme arrière plan ou non. Notre but est d'identifier les pixels mouvements qui ne sont pas détectés avec les algorithmes de segmentation. La figure 3.22 est un exemple de résultat de segmentation sans post-traitement. On constate en effet que les pixels se situant au centre de l'objet ont des valeurs similaires à celles de l'arrière-plan et il devient difficile algorithmiquement de caractériser ces pixels comme pixels de l'objet en mouvement.

FIGURE 3.22 – Exemple de résultat obtenu à l'issue de la segmentation sans post-traitement

Pour résoudre ce problème et compléter les pixels mouvement afin de retrouver l'objet dans sa globalité, les détails de la méthode de post-traitement proposée sont donnés ci-dessous et la figure 3.23 illustre ces étapes :

- le résultat de segmentation dans lequel la forme et le contour ne sont pas retrouvés de manière nette est utilisé pour récupérer les informations (la position, la taille) de la région où se trouve l'objet,
- on extrait de l'image initiale la sous fenêtre correspondant à cette région d'intérêt,
- la région sélectionnée est balayée par une fenêtre de taille $N \times N$ dans l'image initiale et pour chaque fenêtre nous calculons la valeur de son étendue. Cette valeur est comparée à l'écart-type σ des pixels de la fenêtre considérée multiplié par un facteur de pondération k. Si $etendue > k\sigma$, le pixel central de la fenêtre est marqué comme étant un pixel mouvement (couleur blanche) sur le résultat de l'image segmentée.

  Après ces étapes nous supprimons les différents bruits obtenus en remplaçant les pixels de toutes les petites régions détectées comme objet dans le résultat de segmentation initial par des pixels noirs.

3.4.5 Expérimentation : Évaluation du couplage filtrage/segmentation

Pour nos expérimentations, nous avons utilisé un ordinateur Intel Core i7-3210 M 2.5GHz. La chaîne d'analyse d'images que nous avons proposée (voir figure 3.24) consiste à filtrer les images, à les segmenter et enfin à faire le post-traitement des résultats de segmentation. Notre objectif est d'identifier la combinaison adéquate entre filtres et méthodes de segmentation pour l'application prévue. Dans les expérimentations, les méthodes de filtrage et de segmentation utilisées sont celles

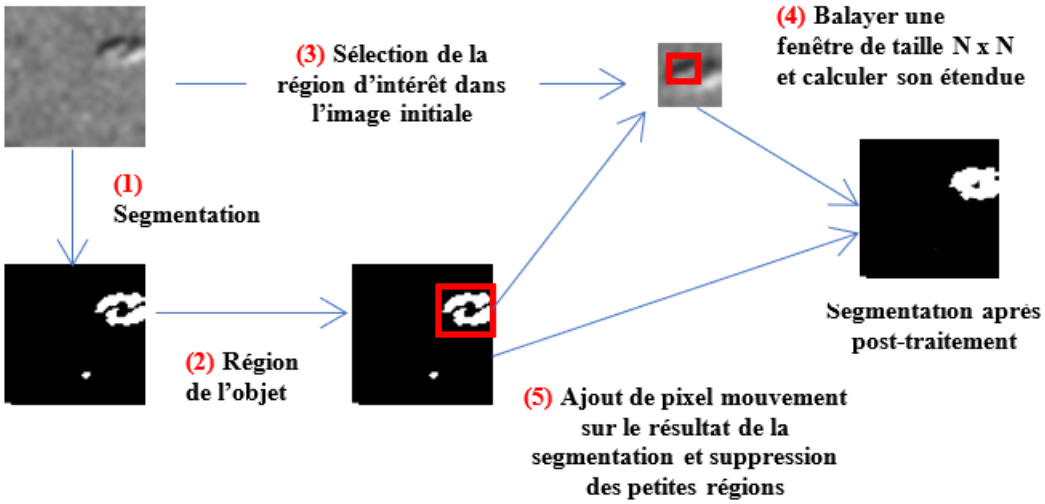


FIGURE 3.23 – Post-traitement proposé

présentées dans les sections 3.4.1 et 3.4.3. Le filtre de Perona & Malik a été supprimé des tests car les résultats de segmentation obtenus (visuellement) semblait très loin de la réalité. Les régions d'intérêts sont très homogènes et similaires à l'arrière plan et par conséquent, les méthodes de segmentation fonctionnent mal avec ce filtre.

Les méthodes d'évaluation de la segmentation d'images se décomposent en 2 catégories : les méthodes d'évaluation supervisée (avec vérité-terrain) et les méthodes d'évaluation non-supervisée (sans vérité-terrain). Ces dernières sont basées sur le calcul de différentes statistiques à partir du seul résultat de segmentation (homogénéité intra-régions, disparité inter-régions...). Si elles sont facilement automatisables, elles sont très dépendantes des caractéristiques choisies. Elles s'avèrent donc souvent moins fiables que les méthodes d'évaluation supervisée, certes difficiles à mettre en œuvre mais qui permettent de mieux prendre en compte les objectifs spécifiques à une application donnée. C'est ce type d'évaluation que nous avons mis en place.

Méthodes d'évaluation supervisée

Les méthodes d'évaluation supervisée mesurent la performance d'un algorithme de segmentation d'images en comparant l'image segmentée automatiquement avec une image de référence ou vérité-terrain (c'est-à-dire une segmentation réalisée manuellement par un/plusieurs experts). En effet, le degré de similarité entre la segmentation de référence et la segmentation automatique obtenue via l'algorithme de segmentation permet d'évaluer la qualité de celui-ci. L'intérêt potentiel d'une évaluation supervisée réside dans le fait que la comparaison entre la segmentation automatique et la segmentation de référence est censée fournir une évaluation très précise. En revanche, son inconvénient majeur réside dans la production de la segmentation de référence. En effet, la création (manuelle) d'une vérité-terrain reste une tâche difficile, subjective et lente. De plus, il est générale-

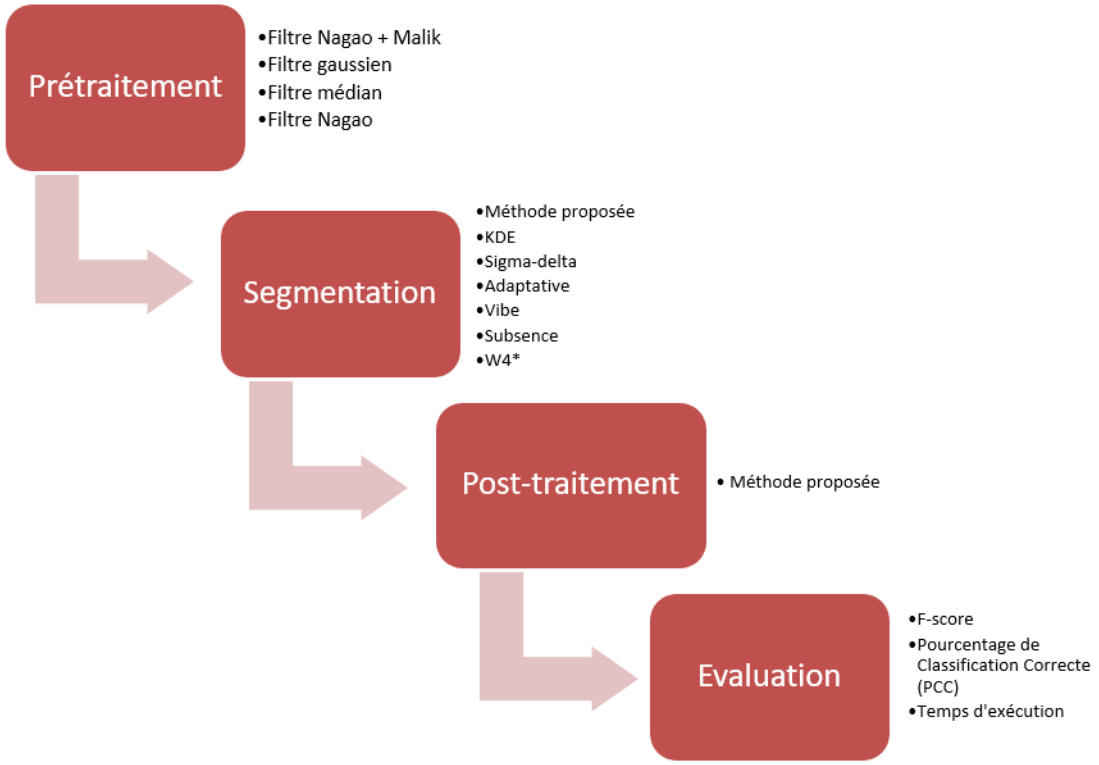


FIGURE 3.24 – Chaîne de traitements implémentée

ment difficile de juger si une segmentation de référence réalisée par un expert est meilleure qu'une autre segmentation réalisée par un autre expert. Ceci rend délicat le choix de la vérité-terrain à considérer pour comparer/valider une approche de segmentation d'images lorsque plusieurs experts ont fait l'exercice de créer une segmentation de référence.

Présentation de la base de données et des critères d'évaluation

Les méthodes de segmentation sont appliquées à 10 vidéos contenant chacune 100 images. Ces vidéos correspondent principalement aux scénarios de marche. Dans chaque vidéo, dix images ont été prises au hasard pour faire l'évaluation. Nous utilisons donc au total 100 images.

Avec l'aide et la disponibilité de 3 experts pour dessiner les vérités terrains, nous avons choisi de mettre en place un protocole d'évaluation supervisée. Pour réduire les écarts entre les vérités terrains établies par les 3 experts nous avons décidé de considérer un pixel comme arrière-plan dans la vérité terrain finale si le pixel a été étiqueté comme tel par au moins 2 des 3 experts.

Comme il est très important pour les applications temps réel de prendre en compte le temps d'exécution des algorithmes, nous utilisons ce critère en plus des deux métriques d'évaluation supervisée simples et bien connues que sont le F-score (voir équation 3.7) et le Pourcentage de Classification Correcte (PCC).

Pour définir les critères d'évaluation, nous utilisons les notions suivantes :

— Précision : $P = \frac{VP}{VP+FP}$

— Rappel : $R = \frac{VP}{VP+FN}$

avec VP : Vrai Positif, VN : Vrai Négatif, FP : Faux Positif, FN : Faux Négatif.

Les critères d'évaluation sont alors définis par :

— F-score :

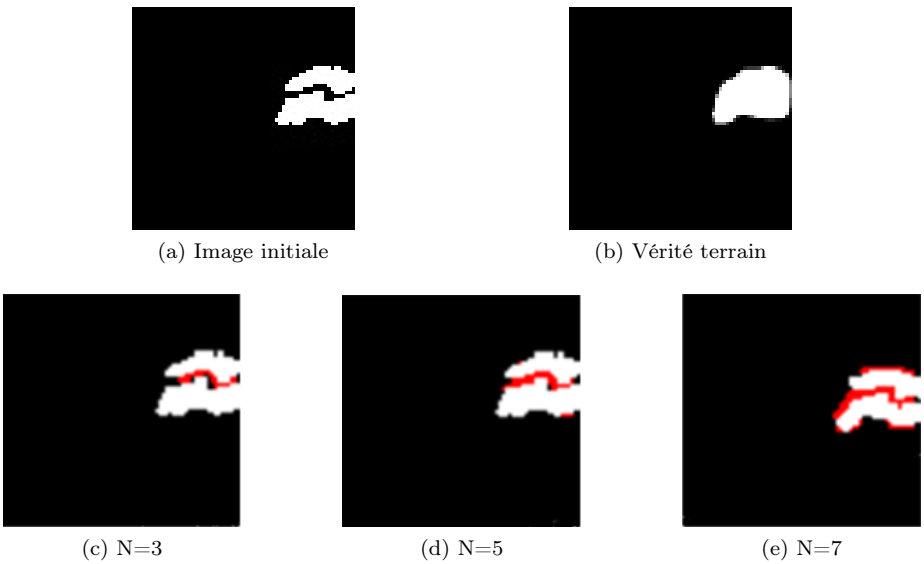
$$F = \frac{2 \times (P \times R)}{(P + R)} \tag{3.7}$$

— Pourcentage de Classification Correcte :

$$PCC = \frac{VP + VN}{VP + FN + FP + VN} \tag{3.8}$$

Paramètres utilisés pour les différents algorithmes

Concernant l'algorithme de post-traitement proposé, des tests empiriques nous ont amenés à choisir N=5 comme taille de fenêtre. Rappelons que le post-traitement a pour but de compléter les pixels mouvement afin de retrouver l'objet dans sa globalité et quasiment similaire à la vérité terrain (voir figure 3.25-(b)). Sur la figure 3.25, en rouge on a les pixels qui sont ajoutés lorsqu'on applique le post-traitement sur les résultats de segmentation. Nous constatons que plus nous augmentons la taille de la fenêtre de balayage, plus nous avons tendance à compléter l'ensemble des pixels de la région (voir figure 3.25-(e)). Plus nous la diminuons, moins nous réussissons à retrouver une forme similaire à celle de la vérité terrain (voir figure 3.25-(c)).



(a) Image initiale (b) Vérité terrain

(c) N=3 (d) N=5 (e) N=7

FIGURE 3.25 – Exemples de résultats obtenus pour différentes tailles de la fenêtre de balayage utilisée lors du post-traitement

Pour le choix du paramètre k, nous avons calculé le pourcentage de classification correcte (PCC) (voir équation 3.8) pour différentes valeurs de k. Les résultats sont présentés sur la figure 3.26. Ces

résultats montrent que pour k =4 nous obtenons le meilleur taux de classification correcte des pixels à savoir 97%.

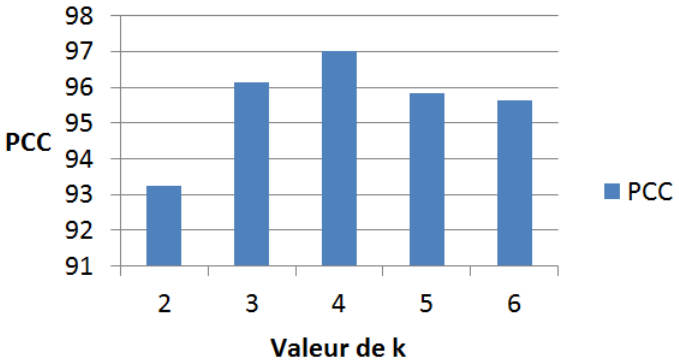


FIGURE 3.26 – Pourcentage de classification correcte pour différentes valeurs de k

La table 3.1 liste les différents paramètres utilisés pour chaque algorithme de filtrage et segmentation utilisé.

Numiter=Nombre d'itérations, dt=constante d'intégration, T=seuil, LF=nombre d'images pour l'apprentissage, α=alpha ou taux d'apprentissage, σ= écart-type.

	Méthodes	Paramètres
Filtrage	Filtre gaussien	σ=1.4
	Filtre médian	taille de la fenêtre=5 × 5
	Filtre Nagao	Pas de paramètre
	Nagao+malik	num_iter=15 ; dt=1/6 ; K=20
Segmentation	Proposée	T=15
	Adaptative	T=15 ; α=0.05 ; LF=15
	KDE	LF=10
	W4*	LF=10, T=3
	Sigma-delta	T=15 ; α=0.05 ; LF=15
	Subsense	taille de la fenêtre=3
Post-traitement		taille de la fenêtre=5 ; k=4

TABLE 3.1 – Différents paramètres des algorithmes de filtrage, segmentation et postraitement.

Interprétations des résultats

Les résultats d'évaluation du processus d'analyse d'images sont regroupés dans le tableau 3.2.

Pour une visualisation plus aisée des résultats comparatifs, nous les présentons sous forme de figure (voir figure 3.27). En ce qui concerne l'évaluation, les meilleures performances de F-score et PCC sont respectivement 0,89 et 0,97 et elles sont obtenues en utilisant le filtre gaussien suivi de la méthode de segmentation que nous avons proposée. La durée d'exécution minimale est obtenue en utilisant le filtre médian (2072 ms). Cependant, le temps d'exécution donnée par le filtre gaussien suivi de notre méthode de segmentation est de 2208 ms ce qui reste très proche du score précédent.

		Méthodes de segmentation							
Filtrage	Critère	PROP	KDE	$\Sigma - \Delta$	ADAPT	VIBE	SUBSENCE	W4*	Moyenne
Nagao	F-score	0.86	0.85	0.77	0.82	0.82	0.82	0.64	0.80
+	PCC	0.96	0.96	0.96	0.96	0.95	0.96	0.76	0.93
Malik	Temps(ms)	18093	20682	20148	20425	20511	26672	20103	-
	F-score	**0.89**	0.86	0.84	0.87	0.81	0.84	0.68	0.83
Gaussien	PCC	**0.97**	0.97	0.97	0.97	0.95	0.97	0.79	0.94
	Temps(ms)	2208	2437	2647	2771	2819	90870	2245	
	F-score	0.85	0.84	0.81	0.81	0.79	0.84	0.61	0.79
Médian	PCC	0.96	0.95	0.96	0.94	0.92	0.96	0.73	0.92
	Temps(ms)	**2072**	2856	2821	2743	2594	8345	2286	-
	F-score	0.86	0.83	0.82	0.84	0.50	0.78	0.63	0.75
Nagao	PCC	0.97	0.96	0.96	0.96	0.75	0.94	0.8	0.90
	Temps(ms)	2448	3077	2809	2378	3667	8446	2756	-

TABLE 3.2 – Résultats d'évaluation obtenus en termes de temps d'exécution, de F-score et de PCC. Nous indiquons en gras les meilleurs résultats obtenus. "PROP" : proposée ; "ADAPT" : Adaptative

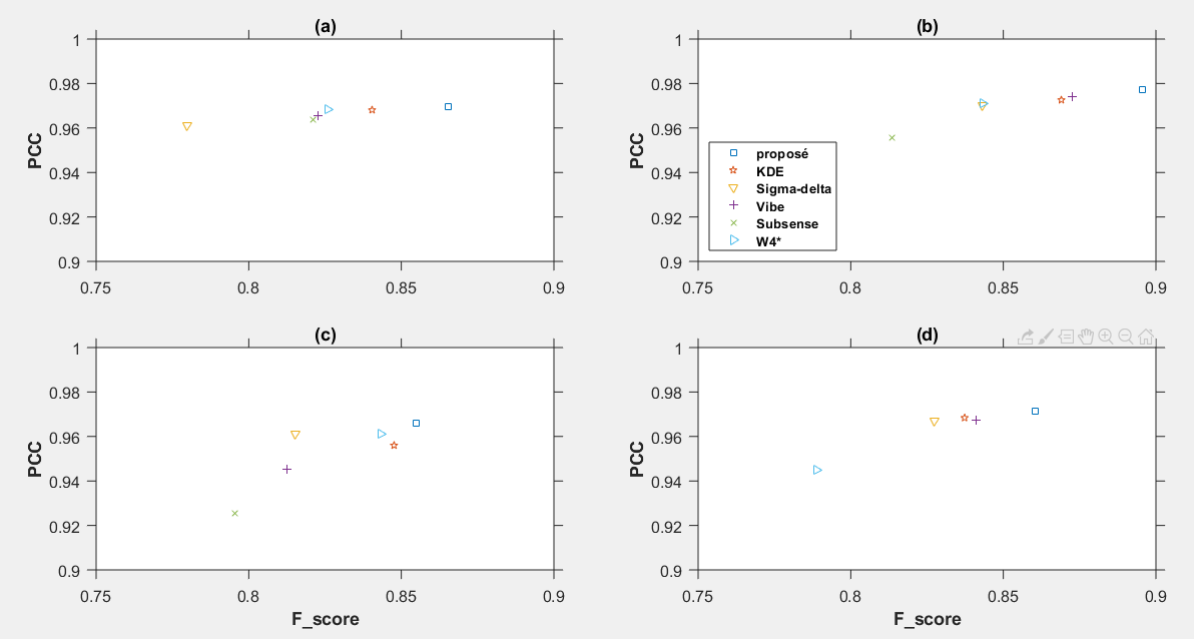


FIGURE 3.27 – Comparaison des méthodes de segmentation pour chaque filtre : (a) Segmentation avec résultat du filtre Nagao + Malik ; (b) Segmentation avec résultat du filtre gaussien, (c) Segmentation avec résultat du filtre médian, (c) Segmentation avec résultat of du filtre Nagao

Dans la figure 3.28, nous montrons les résultats des différentes étapes : pour une image originale, nous présentons son équivalence filtrée, puis son équivalence segmentée sans post-traitement et enfin avec post-traitement.

Du point de vue du filtrage, le filtre Nagao pose un problème d'effet de bloc qui est l'apparition de discontinuités aux frontières entre les blocs adjacents. Ce problème impacte les différents résultats de segmentation sans post-traitement. Ceci est observé par l'apparition de multiples petites régions sur les images segmentées.

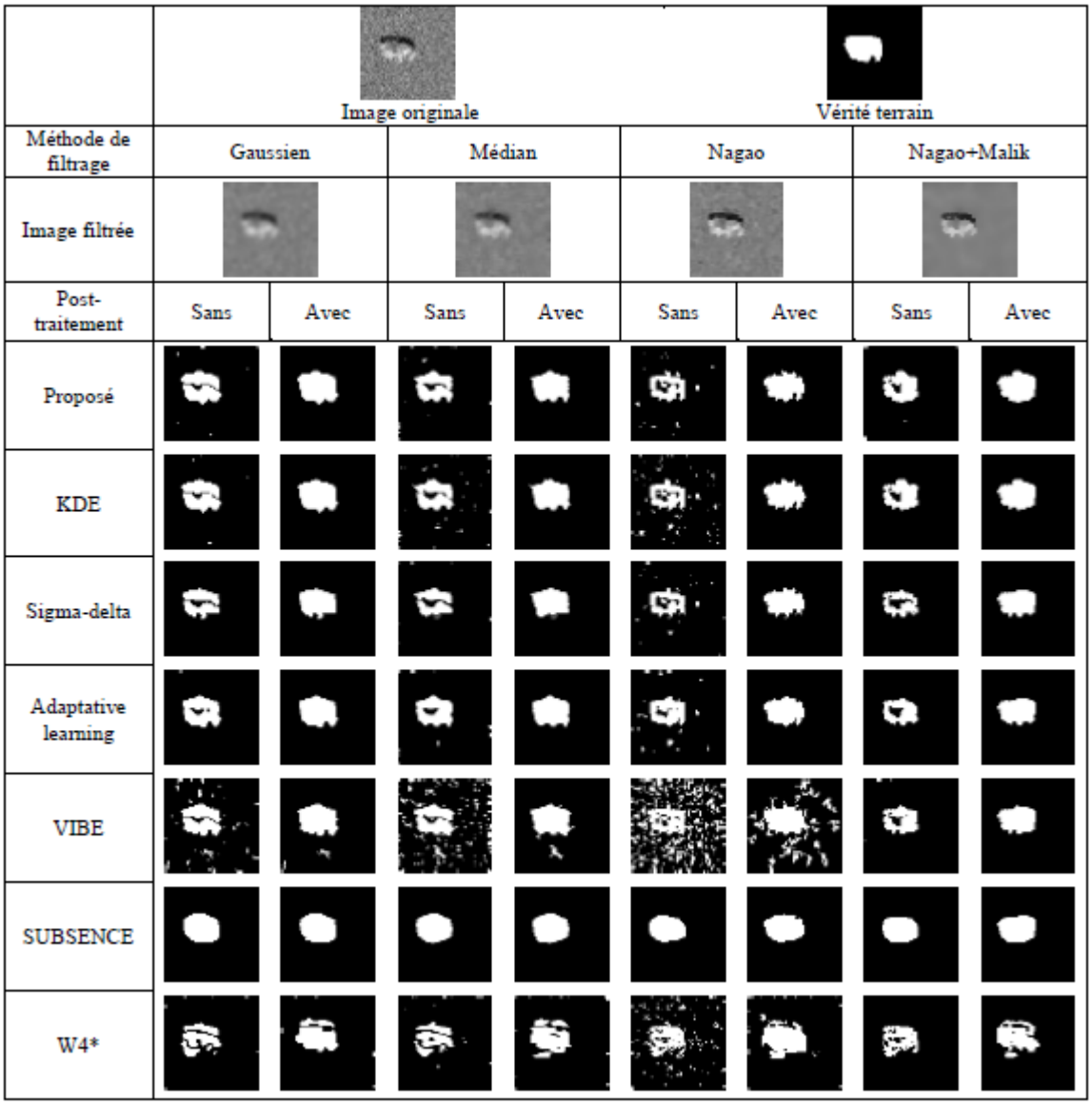


FIGURE 3.28 – Résultats visuels des différentes méthodes de segmentation sans et avec post-traitement, sur différentes méthodes de filtrage

Du point de vue de la segmentation sans post-traitement, les résultats VIBE sont aussi très bruités.

Les résultats de W4* montrent que les formes segmentées vont bien au-delà de l'analyse des 2 zones correspondant à l'apparition et à la disparition d'une partie mobile de la personne.

En ce qui concerne le post-traitement, il corrige bien la segmentation dans la plupart des images et permet d'avoir une forme de la personne à l'exception de W4* où la correction de la segmentation est partielle et aussi dans le cas où les sensibilités au bruit sont combinées comme dans l'utilisation du filtre Nagao suivi de la méthode de segmentation VIBE.

La méthode de segmentation SUBSENCE est la moins sensible au l'influence du filtrage : comme

on peut le voir sur la figure 3.28, cette méthode ne nécessite presque pas de post-traitement, mais prend beaucoup de temps d'exécution.

La segmentation KDE donne également de bons résultats. D'une manière globale on peut dire qu'elle vient en deuxième position après la méthode proposée.

On constate en regardant les résultats de segmentation, que la combinaison de filtre Nagao + Malik donne un bon résultat visuel. Les résultats moyens des critères montrent que cette combinaison de ces deux filtres peut être utilisée en second choix. Mais le filtre Nagao + Malik a besoin de plus de temps d'exécution que tous les autres filtres. Il est donc préférable d'utiliser le filtre Nagao + Malik au lieu d'utiliser uniquement le filtre Nagao pour améliorer la qualité d'image surtout si l'application à développer n'est pas une application temps réel. Le filtre Nagao avec le problème d'effet de bloc nécessite d'être combiné avec un autre filtre pour corriger ce problème.

En résumé, le filtrage gaussien couplé avec la méthode de segmentation et post-traitement que nous avons proposée reste le meilleur compromis entre la performance et le temps de calcul.

3.5 Conclusion

Dans ce chapitre, nous avons commencé par une étude détaillée de l'aspect spatial et temporel des images ce qui nous a permis d'avoir une piste pour répondre à la problématique de reconnaissance d'actions en utilisant les images provenant de l'imageur infrarouge que nous utilisons. Cette étude nous a permis de conclure que les approches basées sur les caractéristiques locales ne donnent pas des résultats intéressants. Nous avons ensuite présenté un procédé dédié à l'analyse d'images pour mieux extraire la silhouette des personnes dans les images. La chaîne proposée est composée d'étapes de prétraitement, de segmentation et de post-traitement. Elle permet de supprimer le bruit pour améliorer la segmentation de l'image, et de détecter la forme des personnes en mouvement dans l'image. Les résultats expérimentaux montrent que la combinaison du filtre gaussien avec la segmentation et le post-traitement que nous avons proposés donne les meilleures performances. Les formes extraites après cette chaîne seront exploitées pour la reconnaissance de d'actions dans le chapitre suivant.

Chapitre 4

Reconnaissance d'actions

Sommaire

4.1 Introduction

Ce chapitre est dédié à la présentation des approches proposées de reconnaissance d'actions et à une étude comparative de méthodes d'apprentissages classiques (ou simples) et profonds. Les méthodes de reconnaissance d'actions classiques se composent généralement d'une étape de représentation de la séquence vidéo, suivi d'une étape d'extraction de caractéristiques et enfin d'une étape de classification. La représentation de la séquence vidéo consiste à trouver un formalisme pouvant résumer l'ensemble des informations contenu dans la vidéo. L'extraction de caractéristiques consiste à identifier des descripteurs distinctifs entre différentes vidéos. Pour des approches de reconnaissance classiques, l'extraction des caractéristiques se fait manuellement c'est à dire que l'humain doit faire un choix de la méthode d'extraction, tandis que pour les approches apprentissage profond, elle se fait de manière automatique. La dernière étape, celle de classification s'intéresse à l'identification des actions à l'aide des méthodes d'apprentissage automatiques. La figure 4.1 présente les étapes de déroulement d'une approche de reconnaissance d'actions.

Dans la seconde section de ce chapitre, nous présentons une approche de représentation de séquence vidéo adaptée à notre cas d'étude ainsi que quelques uns des descripteurs qui se sont relevés pertinents pour décrire les actions présentes dans la base de données. Dans la troisième partie, nous présentons une méthode de représentation de séquences vidéos basée sur les distances des centres de gravité des zones d'intérêt et ensuite nous présentons le descripteur utilisé pour cette représentation. Dans la quatrième et cinquième partie, nous présentons un modèle de reconnaissance d'actions en cascade qui exploite les résultats des méthodes présentées dans les trois premières sections. Nous terminons le chapitre par des expérimentations en présentant les différents résultats comparatifs entre les approches classiques et les approches par apprentissage profond, en présentant l'impact que peut avoir les méthodes de sélection de caractéristiques sur le modèle et enfin en présentant les résultats de reconnaissance d'actions sur des flux vidéos contenant plusieurs actions.

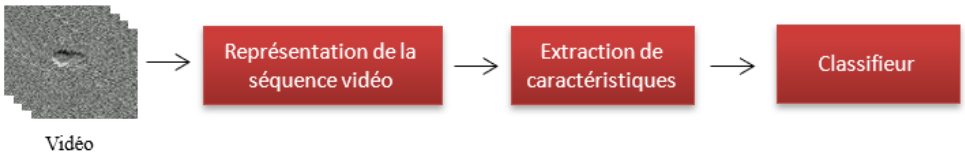


FIGURE 4.1 – Processus des approches classiques de reconnaissances d'actions

4.2 Représentation de la séquence vidéo par MHI et extraction de descripteurs de forme

4.2.1 MHI

Dans le chapitre précédent nous avons montré que les approches basées sur l'analyse de silhouette sont mieux adaptées que celles basées sur les points d'intérêt. Nous avons ainsi mis en place une segmentation robuste incluant pré-traitement et post-traitement permettant de récupérer une silhouette de la personne en mouvement dans la scène. La segmentation des images de la vidéo étant faite, il devient possible de construire une image assez représentative de l'action effectuée dans une vidéo. Dans [110] les auteurs présentent une méthode de construction d'une image binaire qui représente l'énergie de mouvement (MEI : Motion Energy Image). Dans [4], les mêmes auteurs présentent une autre représentation appelée MHI (Motion Historic Image en anglais) qui décrit à la fois la forme du mouvement effectué et la distribution spatiale d'un mouvement. On peut donc coder dans une seule image l'aspect temporel. Elle est plus représentative que le MEI. L'intensité de chaque pixel à une position dans le MHI est fonction de la densité de mouvement à cet endroit.

Pour construire le MHI, nous superposons dans une seule image les résultats de segmentation en changeant à chaque fois le niveau de gris des pixels pour matérialiser l'aspect temporel. Dans notre approche, nous redéfinissons et utilisons l'équation (équation 4.1) pour construire l'image de l'historique des mouvements. Cette équation est une autre formulation de l'équation 2.2. Son utilisation impose que toutes les m images faisant partie de la construction du MHI soient au préalable segmentées. Cette équation nous permet de regrouper les images consécutives de chaque séquence vidéo et ainsi de présenter clairement les différentes zones impactées par le mouvement. Le résultat du MHI est une image en niveaux de gris dans laquelle les pixels correspondant aux mouvements les plus récents sont plus lumineux (blancs) et ceux qui sont moins récents sont sombres (noirs).

$$MHI(x,y,t) = \underset{i\in 1,..,m}{\operatorname{argmax}} D(x,y,t-(m-i)) - 10(m-i) \tag{4.1}$$

où $D(x,y,t)$ est l'image binaire de détection de mouvement extraite de l'étape de segmentation à l'instant t, m représente l'intervalle de temps sélectionné ($m = 15$) et la valeur 10 est un paramètre permettant de varier le niveau de gris des pixels dans la construction de l'historique du mouvement.

Dans la figure 4.2, nous présentons quelques résultats du MHI pour quelques actions.

4.2.2 Extraction des caractéristiques

Le MHI est un modèle de représentation vidéo très simple et largement utilisé pour la reconnaissance d'actions [111]. A partir de cette représentation, il est possible d'extraire un grand nombre de caractéristiques. Les moments de Hu sont un exemple des caractéristiques qui ont été couramment utilisées pour la représentation des formes [112, 113, 114, 115]. D'autres méthodes

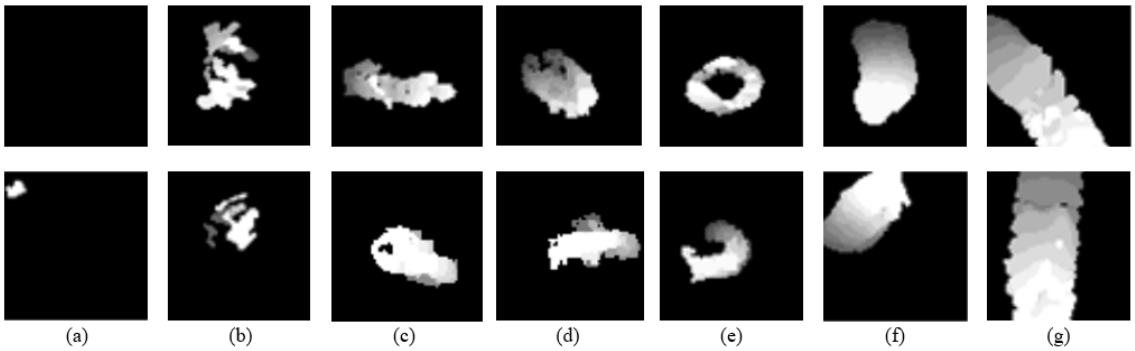


FIGURE 4.2 – Exemple d'images historiques des mouvements pour les différents classes d'actions considérées dans le cadre du projet CoCAPS : aucune action (a), agitation (b), s'asseoir (c), se lever (d), tourner sur le siège (e), marche lente (f) et marche rapide (g).

permettant d'extraire des caractéristiques à partir du MHI se sont avérées efficaces sur diverses bases de données. Dans la littérature, ces approches d'extraction de caractéristiques sont classées en deux familles [116], à savoir :

— les méthodes basées sur l'étude du contour.
— les méthodes basées sur la caractérisation de la forme globale (région).

La figure 4.3 présente une vue d'ensemble de différentes approches disponibles pour la représentation de formes. Cette partie décrit quelques unes des approches pour chacune des familles qui viennent d'être citées. Dans la partie expérimentation, ces méthodes sont testées sur la base de données créée dans le cadre du projet.

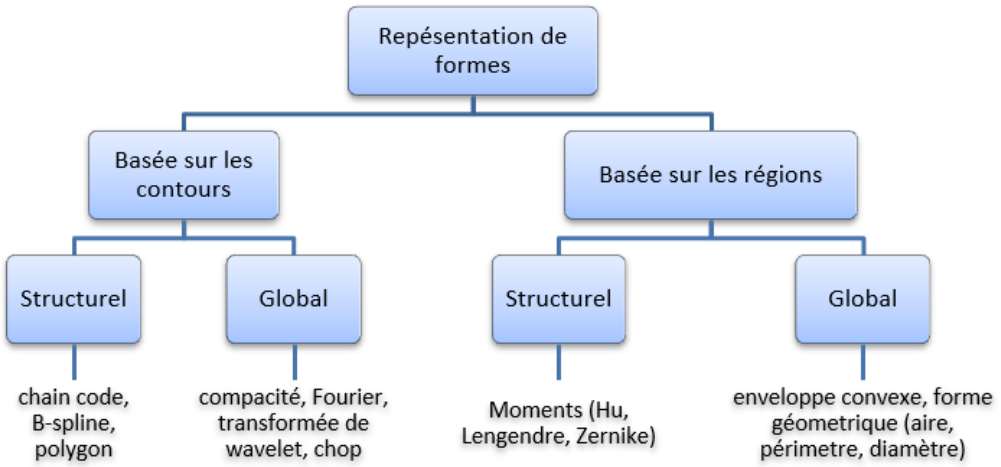


FIGURE 4.3 – Différentes méthodes d'extraction de caractéristiques pour la représentation de formes

4.2.3 Hu moments

Les premiers moments géométriques utilisés en vision par ordinateur ont été introduits par Hu [117]. La formulation générale des moments d'une image $I(x, y)$ est définie comme suit (voir équation 4.2) :

$$M_{m,n} = \sum_x \sum_y x^m y^n I(x, y) \qquad avec\ m, n = 0, 1, 2... \tag{4.2}$$

Pour normaliser la translation dans le plan de l'image, Hu a montré que les moments centraux $\mu_{m,n}$ doivent être utilisés. Ces moments sont définis par l'équation 4.3 :

$$\mu_{m,n} = \sum_x \sum_y (x - \bar{x})^m (y - \bar{y})^n I(x, y) \tag{4.3}$$

où $\bar{x} = \frac{M_{10}}{M_{00}}$; $\bar{y} = \frac{M_{01}}{M_{00}}$ sont les composantes du centroïde. Les moments centraux sont ensuite normalisés pour construire un descripteur invariant en translation et en échelle, voir l'équation 4.4.

$$\eta_{n,m} = \frac{\mu_{m,n}}{\mu_{0,0}^{\left(1+\frac{m+n}{2}\right)}} \tag{4.4}$$

où $\mu_{0,0} = M_{0,0}$ et $m + n \geq 2$.

Dans le domaine de la reconnaissance de formes, Hu a introduit sept moments invariants basés sur les moments centraux normalisés. Ce descripteur permet de coder une forme avec invariance de translation, d'échelle et de rotation. Ces 7 nouvelles mesures sont définies comme suit :

$\phi_1 = \eta_{20} + \eta_{02}$

$\phi_2 = (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2$

$\phi_3 = (\eta_{30} - 3\eta_{12})^2 + (3\eta_{21} - \eta_{03})^2$

$\phi_4 = (\eta_{30} + \eta_{12})^2 + (\eta_{21} + \eta_{03})^2$

$\phi_5 = (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] + (3\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2]$

$\phi_6 = (\eta_{20} - \eta_{02})[(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] + 4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{21} + \eta_{03})$

$\phi_7 = (3\eta_{21} - \eta_{03})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] - (\eta_{30} - 3\eta_{12})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2]$

Pour notre application, pour chaque image représentative d'une séquence vidéo, nous avons calculé le vecteur de caractéristiques composé des 7 moments invariants : $[\phi_1 ... \phi_7]$.

4.2.4 Color Histogram of Oriented Phase (CHOP)

Le descripteur CHOP [118] permet d'identifier et de localiser avec précision les caractéristiques de l'image grâce à des techniques reposant sur le gradient. Il est basé sur le calcul de congruence de phase locale de l'image. Il vise à résoudre le problème de variation d'éclairage et de contraste dans la reconnaissance d'objets. Pour construire ce descripteur, la congruence de phase (CP) est calculée sur chaque canal de couleur de l'image et les valeurs maximales pour chaque pixel correspondant

aux trois canaux sont sélectionnées. La carte d'orientation de phase qui en résulte est utilisée pour construire l'histogramme en divisant les images en blocs et en cellules. Le concept de congruence de phase repose sur le modèle de l'algorithme d'énergie local introduit par Morrone et Owens [119] qui postule que "les caractéristiques sont perçues aux points où les composantes de Fourier de l'image sont au maximum de phase".

Étant donné un signal d'entrée $I(x)$, la congruence de phase est définie par l'équation 4.5 suivante :

$$CP(x) = \frac{E(x)}{\epsilon + \sum_n A_n} \tag{4.5}$$

où $E(x)$ est l'énergie locale, A_n correspond à toutes les amplitudes de la composante de Fourier de $I(x)$ et ϵ est une petite quantité pour éviter la division par zéro.

Ragb et al[118] ont calculé la CP en convoluant l'image avec une paire de filtres log-Gabor en quadrature pour extraire les fréquences locales et les informations de phase. Nous proposons d'utiliser CHOP et de le tester combiné avec d'autres descripteurs car c'est un descripteur qui a apporté de meilleurs résultats pour la détection des humains par rapport au descripteur HOG.

4.2.5 Descripteurs géométriques

Plusieurs indices géométriques ont été proposés dans la littérature [120]. Un indice de forme géométrique peut être considéré comme tout paramètre, coefficient ou combinaison de coefficients permettant de donner des renseignements chiffrés sur la forme des objets. La figure 4.4 présente quelques exemples de mesures utilisées pour définir un descripteur d'indices géométriques : caractéristiques de l'ellipse (axe principal : R_{max}, axe secondaire : R_{min}), enveloppe convexe (C_H), périmètre (P) .

Pour construire notre descripteur géométrique, nous avons utilisé 9 indices géométriques qui sont :

- l'axe principal de l'ellipse englobante de la forme (R_{max}),
- l'axe secondaire de l'ellipse englobante de la forme (R_{min}),
- aire : nombre de pixels dans la région (A),
- aire convexe : nombre réel de pixels de l'enveloppe convexe ($A(C_H)$),
- périmètre : distance entre chaque paire de pixels adjacents autour de la bordure de la région (P),
- circularité : ($\frac{R_{min}}{R_{max}}$),
- convexité surfacique : ($\frac{A}{A(C_H)}$),
- convexité périmétrique : ($\frac{P(C_H)}{P}$),
- excentricité : rapport de la distance entre les foyers de l'ellipse englobante et la longueur de son axe principal.

Ces indices géométriques renseignent respectivement sur la taille et sur la forme de l'ensemble. Ils fournissent par conséquent des informations complémentaires sur la forme du mouvement de la personne.

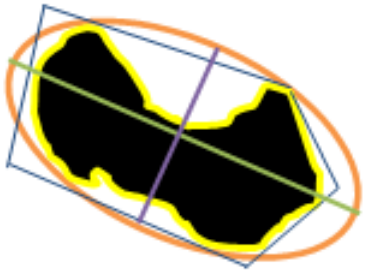


FIGURE 4.4 – Exemples de mesures : surface (noir), périmètre (jaune), axe principal (vert), axe secondaire (violet), enveloppe convexe (bleu)

Le choix de ces trois descripteurs (Hu, CHOP et géométriques) parmi tant d'autres résulte de différents tests réalisés sur notre base de données dans lesquels ces descripteurs se sont avérés plus pertinents que les autres (descripteur SIFT, HOG, HOF, Moment de Zernike, descripteurs de texture comme les attributs statistiques du second ordre, Local binary Patterns (LBP), transformée de Fourier). Parmi les descripteurs choisis, un est basé sur les contours (CHOP) et deux autres basés sur les régions (moment de Hu et descripteur de forme géométrique).

Nous avons également proposé de combiner ces 3 descripteurs entre eux pour évaluer l'impact d'une combinaison sur nos modèles car dans la littérature, la combinaison de descripteurs augmente parfois les performances des méthodes [121]. La combinaison de descripteurs ici consiste à concaténer les descripteurs les uns à la suite des autres pour avoir un nouveau vecteur descripteur. On utilisera la notation "$\oplus$" pour matérialiser la combinaison des descripteurs (par exemple $HU \oplus CHOP$).

4.3 Représentation de séquence vidéo basée sur les zones d'avancée-trainée et extraction d'attributs statistiques

Après l'étape de segmentation par seuillage (cf algorithme 2) (sans post-traitement), on extrait 2 zones d'avancée et de trainée dans une image (zone claire : partie correspondant au mouvement de la personne et zone sombre : partie correspondant à la trainée laissée par la personne). L'étape suivante consiste à trouver le centre de gravité des zones claires (respectivement des zones sombres) pour ensuite calculer les distances de déplacement des centres de gravité entre deux zones de même nature à savoir la zone claire et la zone sombre (voir figure 4.5). Si pour deux images consécutives il n'y a pas de forme segmentée, alors la distance est mise à "nulle". La représentation de la séquence vidéo est donc une matrice à 2 lignes : la première ligne correspond aux distances entre les centres de gravité des zones claires et la deuxième correspond aux distances entre centres de gravité des zones sombres.

Après avoir mis en place la représentation de la séquence vidéo via la matrice de distance des centres de gravité, nous construisons un descripteur basé sur des attributs statistiques. Pour

FIGURE 4.5 – Aperçu de représentation de la séquence vidéo

l'ensemble des distances des centres des zones claires (respectivement zones sombres), on compte et élimine du vecteur les distances nulles (ce nombre de distance nulle est considéré comme une caractéristique), ensuite on calcule les attributs statistiques du vecteur de distance et on retient principalement les quatre mesures suivantes : moyenne, minimum, maximum, écart-type. En plus de ces mesures on ajoute une sixième caractéristique qui caractérise le degré d'activité dans une vidéo. Ce degré d'activité peut être défini au travers du calcul de l'inertie entre toutes les images segmentées. Dans une vidéo, l'inertie des zones claires (respectivement sombres) est calculée en appliquant l'équation 4.6. Soient A_i la somme des aires des formes dans l'image segmentée i , G_i le centre de gravité des formes de l'image segmentée i, G le centre de gravité de tous les G_i, $d(G_i, G)$ la distance euclidienne entre G_i et G et m le nombre d'images dans la période considérée, l'inertie est définie par :

$$Inertie = \sum_{i=1}^{m} A_i \times d(G_i, G) \tag{4.6}$$

La figure 4.6 présente les grandeurs utilisées pour le calcul de l'inertie entre les différentes formes segmentées dans les images d'une vidéo.

Plus l'inertie est grande, plus l'action est composée de grands mouvements. Les vidéos n'ayant aucune action auront donc une inertie nulle. Pour une vidéo on a donc 12 descripteurs dont 6 sont extraits sur la trainée (zone sombre) et 6 sont extraits sur l'avancée (zone claire).

4.4 Expérimentations

Dans cette section, nous comparons les résultats obtenus pour les deux méthodes de représentation introduites précédemment et les différents descripteurs dans le cadre du protocole de test établi au sein du projet CoCAPS.

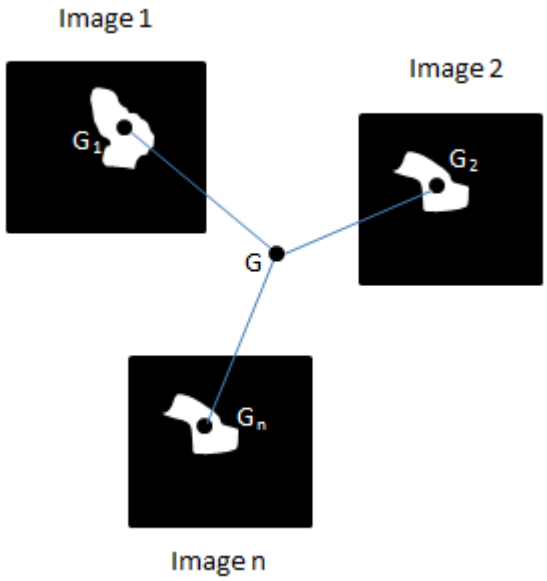


FIGURE 4.6 – Intertie inter-images

4.4.1 Base de données

Pour le projet CoCAPS, deux cas d'études ont été ciblés pour la reconnaissance d'actions. Le premier concerne les environnements de bureau, le second les institutions pour personnes âgées (maison EHPAD). Nous effectuons nos expériences sur les données acquises dans le cadre du projet. Conçue en collaboration avec des partenaires industriels, la base de données cible des situations de bureau et 7 classes d'actions ont été retenues : pas d'action, agitation, s'asseoir, se lever, tourner sur un siège, marcher lentement (vitesse inférieure à 1 mètre/seconde) et marcher rapidement (vitesse supérieure à 1 mètre/seconde). Pour l'action "agitation", il s'agit de tous les petits mouvements effectués autour d'un poste de travail tels que répondre au téléphone, déplacer des objets, effectuer une saisie au clavier... Au total, l'ensemble de données se compose de 700 vidéos échantillons (100 échantillons par action). Les vidéos ont été prises avec l'aide d'une trentaine d'étudiants de morphologie différente (taille, corpulence) et habillés différemment (casquettes, blouson, tee-shirt...). Notons aussi que les actions ont été effectuées suivant différentes directions, avec des éclairages différents et enfin elles ont été recueillies à différentes périodes de l'année de jour comme de nuit (sauf en cas de chaleur excessive de la pièce car aucune détection de mouvement n'est alors possible par le capteur). Une contrainte forte du projet impose le positionnement du capteur au plafond (en vue de dessus) et au centre de la pièce surveillée. Le champ de vision considéré est ainsi différent de ce qui est observé dans la plupart des bases de données de reconnaissance d'actions. Pour chaque vidéo, nous considérons des tailles de clips de 15 images (environ 1,5 secondes) pour la reconnaissance des actions. La figure 4.7 présente des exemples d'images consécutives extrait des clips vidéos correspondant à chaque action.

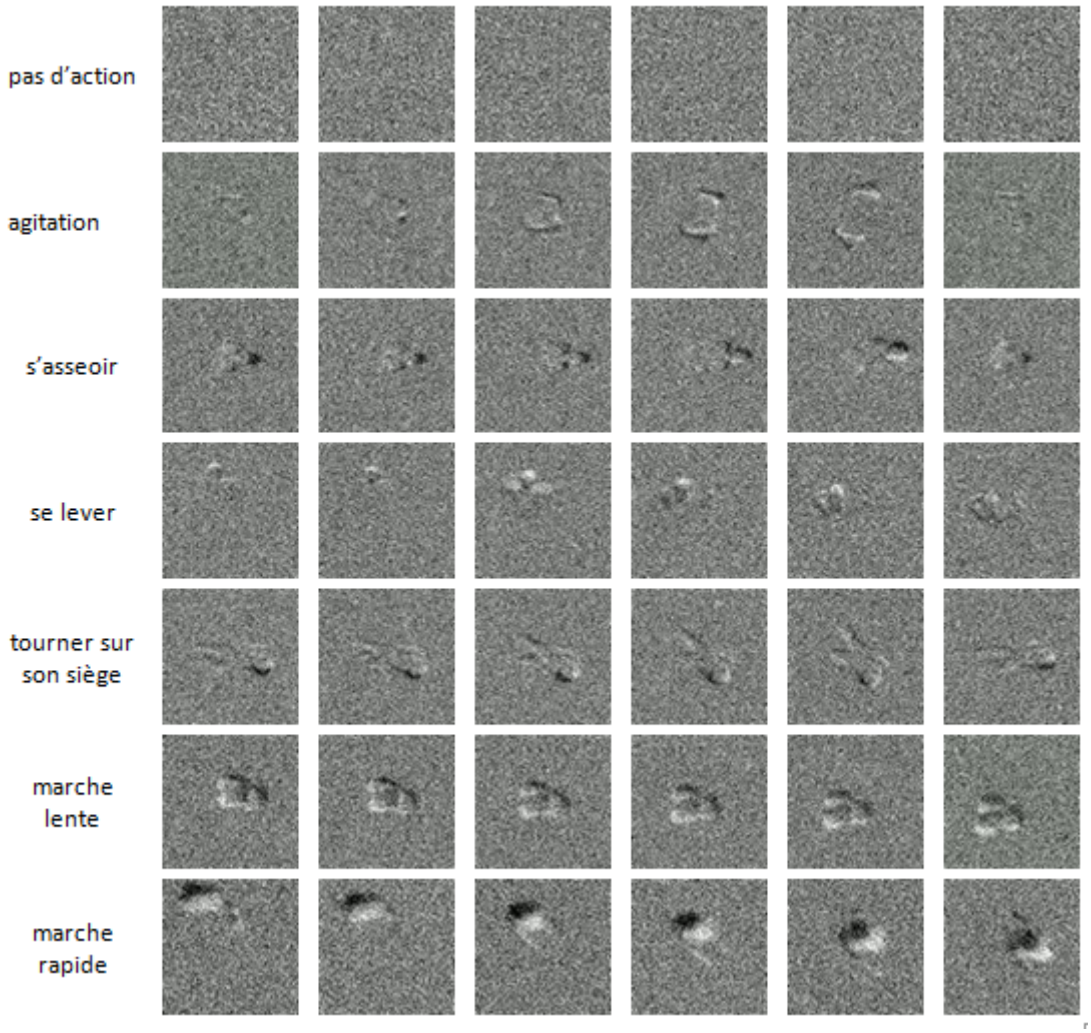


FIGURE 4.7 – Exemple d'images consécutives correspondant aux différentes actions

4.4.2 Classifieurs

On distingue deux types de classifieurs : les classifieurs supervisés et les classifieurs non supervisés. Nous nous intéressons ici au premier type puisque nous disposons d'une base de données labélisées. Il existe plusieurs méthodes de classification supervisée. Le choix d'un algorithme de classification n'est pas évident car cela dépend de la nature de nos données, de leur volume, des informations que nous souhaitons en extraire. Dans la première partie des expérimentations, nous exploiterons 4 types de classifieurs supervisés et nous retiendrons le plus discriminant pour la suite de notre étude.

Classifieur KNN

Le classifieur KNN est une approche non paramétrique qui a été utilisée au début des années 1970 dans des applications statistiques [122, 123]. Le classifier KNN permet de modéliser des

phénomènes non linéaires en se basant sur des informations locales. Plus les individus appartenant à la même classe sont proches (même s'ils sont séparés en plusieurs groupes dans l'espace des caractéristiques) plus la méthode modélise efficacement le problème. De plus cette méthode permet de travailler avec des bases de données contenant des classes déséquilibrées sans avoir à corriger ce déséquilibre. Pour classer une action, la méthode consiste à rechercher les k plus proches voisins en terme de distance et de prédire la classe de l'individu à partir de la classe des voisins [124]. Plusieurs distances existent (distance de Hamming, distance de Mahalanobis, distance euclidienne, distance de Chebychev, distance cosinus ....). Pour nos modèles, nous utiliserons la distance de Manhattan. Elle est obtenue en faisant la somme des différences absolues entre les coordonnées de 2 individus. Par exemple, entre deux points A et B, de coordonnées respectives (X_A, Y_A) et (X_B, Y_B), la distance de Manhattan est définie par :

$$d(A, B) = |X_B - X_A| + |Y_B - Y_A|$$

Cette distance produit des résultats proches de ceux obtenus par la simple distance euclidienne. Cependant, avec cette mesure, l'effet d'une seule différence significative (valeur aberrante) est atténué [125].

Pour ce classifieur, le paramètre k joue un rôle très important dans les performances du classifieur. Par exemple dans la figure 4.8, nous remarquons que pour une valeur de k différente, un nouvel individu peut être attribué à une autre classe. Pour $k = 3$, le nouvel individu sera mis dans la classe B tandis que pour $k = 6$, il est mis dans la classe A. Nous avons donc décidé de tester différentes valeurs de k et de retenir la meilleure (au sens du taux de prédiction). Après différents tests nous avons choisi d'utiliser K=3. La méthode KNN a généralement une bonne précision prédictive lorsqu'elle est utilisée avec de petites bases de données.

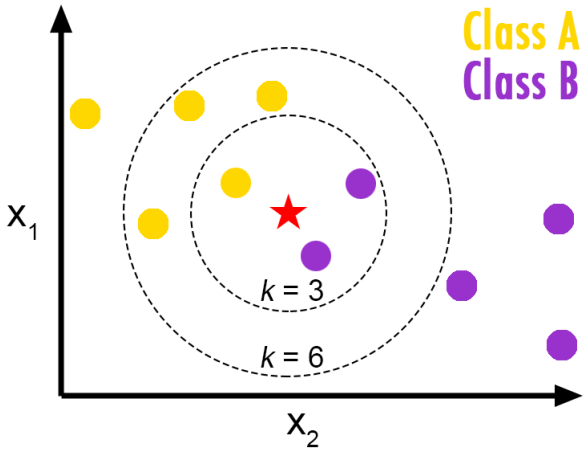


FIGURE 4.8 – Exemple de classification avec les KNN pour k= 3 et k=6 [17]

Classifieur bayesien

Le classifieur naïf bayésien [126] est un modèle probabiliste qui estime la probabilité qu'une image appartienne à une classe à l'aide du théorème de Bayes [127] : l'algorithme naïf de Bayes fait l'hypothèse que les prédicteurs sont conditionnellement indépendants, compte tenu de la classe. Un avantage de cette méthode est la simplicité de programmation, la facilité d'estimation des paramètres et sa rapidité (même sur de très grandes bases de données). Le classifieur bayesien modélise la densité de probabilité conditionnelle d'appartenance à une classe comme une distribution gaussienne avec une matrice de covariance diagonale [128]. Un motif de test est attribué à la classe C_i lorsque la probabilité a posteriori $P(C_i|x)$ est maximale parmi toutes les classes.

$P(C_i|x)$ est estimé par la règle de Bayes suivante :

$$P(C_i|x) = \frac{P(x|C_i).P(C_i)}{\sum_k P(x|C_k).P(C_k)}$$

où $P(x|C_i)$ est la probabilité conditionnelle qu'un élément de test x soit de classe C_i.

Classifieur LDA

L'analyse discriminante peut être prédictive ou descriptive. Il s'agit dans le premier cas de construire un ensemble de règle d'affectation qui permet de prédire le groupe d'appartenance d'un individu à partir des valeurs prises par les variables prédictives. C'est une méthode de discrimination basée sur une modélisation probabiliste des données. L'analyse discriminante linéaire est étroitement liée au classifieur bayesien en ce sens que les deux classifieurs supposent qu'il existe des distributions gaussiennes à l'intérieur de chaque classe [123, 129]. En LDA, deux hypothéses sont faites pour déterminer l'appartenance à une classe. L'hypothèse de normalité statue que la probabilité conditionnelle suit une loi normale multidimensionnelle. L'hypothèse d'homoscédasticité statue que les matrices de variance-covariance pour chaque classe sont identiques.

L'analyse discriminante est très séduisante dans la pratique car la fonction de classement s'exprime comme une combinaison linéaire des variables prédictives, facile à analyser et à interpréter.

Classifieur SVM

Les classifieurs SVM sont par nature des classifieurs binaires. Pour effectuer une classification multi-classe, la technique la plus courante en pratique a été de construire des classificateurs un contre tous et de choisir la classe qui prédit la donnée de test avec la plus grande marge. Une autre stratégie consiste à construire un ensemble de classifieur "un contre un" et à choisir la classe qui est sélectionnée par le plus grand nombre de classifieurs. Pour séparer les données, le SVM détermine l'hyperplan optimal possédant la marge maximale (Figure 4.9), c'est-à-dire l'hyperplan qui sépare les classes avec la distance de sécurité maximale aux données d'entraînement les plus proches de chaque côté [129] .

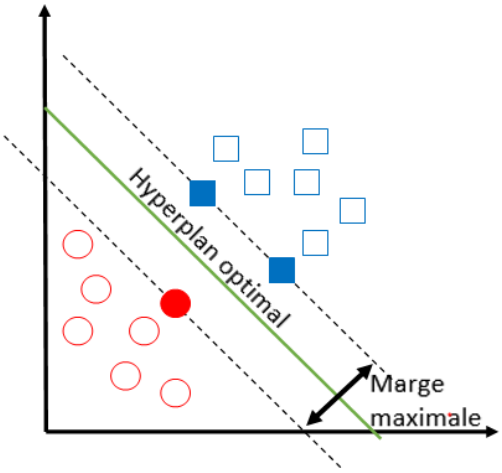


FIGURE 4.9 – Hyperplan optimal et marge maximale d'un SVM

4.4.3 Protocole de validation et critères d'évaluation

Dans la littérature, différents protocoles de validation sont classiquement rencontrés [130] :

— Le K-Fold ou cross validation est un protocole dans lequel les données sont séparées en K groupes de taille identique. L'apprentissage s'effectue sur $K-1$ groupes et la validation sur le groupe retiré. Cette opération est réitérée pour tous les groupes, puis la moyenne des taux de bonne prédiction est calculée.

— Le leave-one-out est un protocole de validation de type k-fold. Il fonctionne comme suit : à chaque itération, une donnée est retirée, puis l'apprentissage est effectué sur les données restantes et la validation est faite sur la donnée préalablement retirée. Ensuite, le processus est réitéré pour sur tous les individus et la moyenne des taux de bonne prédiction est calculée.

— Le hold-out encore appelé la validation non croisée consiste à séparer l'ensemble de données en 2 sous échantillons, appelés ensemble d'apprentissage et ensemble de tests. Dans la pratique, plus de 2/3 des données sont utilisés pour l'apprentissage et le reste pour le test.

Pour valider nos résultats, nous utilisons la validation K-fold (K=10) elle permet d'obtenir un bon compromis entre validation robuste (sur 10% des données) et problème de sur-apprentissage généralement observé dans la méthode leave-one-out (c'est à dire avoir autant de modèle que de données). La validation 10-fold est généralement conseillée pour de petites bases de données.

Le choix du protocole de validation étant fait, nous utilisons comme critère d'évaluation le F-score (équation (3.7)) qui fournit une mesure réaliste de la performance d'un test et est défini comme la moyenne harmonique pondérée de la précision et du rappel.

En plus de la valeur de F-score, nous utiliserons comme indice de comparaison la matrice de confusion. Une matrice de confusion est un résumé des résultats de prédictions sur un problème de classification. Les prédictions correctes et incorrectes sont mises en lumière et réparties par

classe. Les résultats peuvent ainsi être comparés avec les valeurs réelles. Cette matrice permet de comprendre de quelle façon le modèle de classification répartit les éléments de chaque classe lorsqu'il effectue des prédictions. Ceci permet non seulement de quantifier les erreurs commises, mais surtout de caractériser quels sont les types d'erreurs commises. A partir des résultats escomptés et des prédictions, la matrice de confusion indique le nombre de prédictions correctes pour chaque classe et le nombre de prédictions incorrectes pour chaque classe organisées en fonction de la classe prédite. Une matrice de confusion est une matrice carrée comportant autant de lignes que de classes présentes dans la base de données. En ligne on lit les labels réels et en colonne les labels prédits. A chaque fois que l'algorithme reconnaît l'objet de classe i comme étant un objet de classe j (j pouvant être égal à i), on incrémente la case (i,j). Les vrais positifs sont alors sur la diagonale de la matrice.

4.4.4 Résultats et interprétations

Sur la base de données que nous avons conçue pour le projet, plusieurs tests ont été effectués. Dans la table 4.1, nous présentons les résultats obtenus avec chaque descripteur, chaque combinaison de descripteur et chaque méthode de classification. La valeur de F-score maximale est 83%, obtenue lorsque nous utilisons le descripteur CHOP et le classifieur KNN. Nous remarquons que le modèle proposé basé sur les attributs statistiques a des résultats intéressants et très proches du meilleur modèle. Avec le classifieur LDA on atteint 81% de F-score. Ce résultat montre qu'en utilisant un descripteur simple (complexité de calcul moindre) basé sur des attributs statistiques, nous arrivons à bien caractériser les actions par rapport aux méthodes très utilisées de la littérature. La combinaison des descripteurs peut parfois dégrader les performances de classification d'un modèle. C'est le cas du descripteur $Geo \oplus Hu \oplus CHOP$ qui, utilisé avec le classifieur LDA donne un pourcentage très faible de 23% tandis qu'avec les autres méthodes de classification le pourcentage est supérieur à 50%. Ceci nous permet de dire que la méthode de classification LDA est très sensible au type de descripteurs utilisés.

	Descripteurs	Nombre de caractéristiques	KNN	BAYE-SIEN	LDA	SVM
Modèle basé sur le MHI et une famille de descripteurs	moments de Hu	7	49%	35%	36%	35%
	Géométrique(Geo)	9	64%	67%	70%	71%
	CHOP	128	**83**%	61%	71%	70%
Modèle basé sur le MHI et fusion de descripteurs	Geo⊕Hu	16	50%	73%	60%	64%
	Hu⊕CHOP	135	71%	64%	42%	60%
	Geo⊕CHOP	137	65%	70%	70%	82%
	Geo⊕Hu⊕CHOP	144	52%	70%	23%	71%
Modèle proposé	attributs statistiques	12	78%	77%	81%	80%

TABLE 4.1 – Valeurs de F-score pour différents modèles avec une classification en 7 classes (un seul classifieur)

Notre choix de classifieur s'est porté pour la suite de notre étude sur la méthode KNN parce

qu'elle permet d'obtenir en moyenne la plus grande valeur de F-score parmi tous les autres classifieurs. C'est l'une de méthodes de classification les plus simples et plus intuitives [131, 132]. Les résultats de classification présentés dans les paragraphes suivants sont obtenus avec cette méthode.

En observant la matrice de confusion (cf tableau 4.2) du meilleur modèle, celui obtenu en utilisant le modèle basé sur le MHI et le descripteur CHOP, nous constatons premièrement que la classe "pas action" est reconnue à 100% et qu'aucune autre classe ne se confond avec cette dernière. De plus, nous constatons que les erreurs de classification les plus fréquentes sont faites entre les classes "s'asseoir" et "se lever", entre les classes "agitation" et "tourner sur son siège" et enfin entre les classes "marche lente" et "marche rapide". Cela semble tout à fait normal vu le positionnement du capteur (vue de dessus) et aussi vu la méthode de représentation de vidéo utilisée (le MHI), il peut être difficile de séparer clairement ces classes.

	pas d'action	agitation	s'asseoir	se lever	tourner sur son siège	marche lente	marche rapide
pas d'action	**100**	0	0	0	0	0	0
agitation	0	**80**	2	5	11	2	0
s'asseoir	0	4	**76**	15	0	5	0
se lever	0	1	16	**81**	2	0	0
tourner sur son siège	0	11	2	2	**85**	0	0
marche lente	0	7	6	2	2	**72**	11
marche rapide	0	0	1	0	0	7	**92**

TABLE 4.2 – Matrice de confusion en utilisant CHOP comme descripteur (avec 83% de F-score)

En regardant aussi plus en détail (voir tableau 4.3) comment les classes sont prédites dans le modèle proposé basé sur les zones d'avancée-trainée et sur des attributs statistiques, nous constatons que les marches sont très bien classées par rapport à toutes les autres actions et aussi que ce modèle permet de reconnaître à 100% les vidéos ne comportant pas d'action. Comme dans la matrice de confusion du meilleur modèle (tableau 4.2), plusieurs classes se confondent entre elles, notamment entre la classe "agitation" et la classe "tourner sur son siège". Sur la ligne 6 (tourner sur son siège), le modèle a bien prédit 49 éléments de cette classe "tourner sur son siège" et le modèle a prédit 31 éléments comme étant de la classe "agitation", 12 éléments comme étant de la classe "s'asseoir" alors qu'en réalité ils étaient dans la classe "tourner sur son siège".

L'observation des matrices de confusion précédentes (tableau 4.2 et 4.3) montre que les approches considérées peuvent s'avérer complémentaires et qu'un modèle en cascade pourrait permettre d'améliorer les performances obtenues.

	pas d'action	agitation	s'asseoir	se lever	tourner sur son siège	marche lente	marche rapide
pas d'action	**100**	0	0	0	0	0	0
agitation	0	**74**	14	3	9	0	0
s'asseoir	0	15	**67**	12	4	1	1
se lever	0	9	15	**71**	3	1	1
tourner sur son siège	1	31	12	5	**49**	2	0
marche lente	0	0	1	1	1	**94**	3
marche rapide	0	2	0	0	0	1	**97**

TABLE 4.3 – Matrice de confusion du modèle proposé basé sur des attributs statistique (78% de F-score)

4.5 Proposition d'un modèle en cascade

4.5.1 Modèle en cascade proposé

Avant de présenter en détail le modèle en cascade nous définissons d'abord deux notions qui justifieront la différentiation de nos classifieurs à savoir la notion de grand mouvement et de petit mouvement.

Un mouvement est considéré comme grand mouvement s'il y a déplacement du centre de gravité de la personne et il est considéré comme petit mouvement dans le cas contraire. Ces notions nous permettent de classer les actions en 2 grandes catégories. Par exemple, dans les grands mouvements, nous avons la marche rapide ($>$ 1 mètre/seconde) et la marche lente ($<$ 1 mètre/seconde). Dans les petits mouvements nous avons les classes comme : s'asseoir, se lever, s'agiter et tourner sur un siège. Dans l'approche en cascade proposée, nous utiliserons donc trois classifieurs. Le premier (classifieur 1) pour séparer les vidéos sans mouvement, les grands mouvements et les petits mouvements. Le second classifieur (classifieur 2) est utilisé pour classifier les actions parmi les grands mouvements et le dernier (classifieur 3) est utilisé pour classifier les actions parmi les petits mouvements. Pour chaque classifieur un ensemble de caractéristiques appropriées lui est associé pour avoir de bonnes performances.

La figure 4.10 donne un aperçu de la méthode. Pour chaque classifieur, la définition d'un ensemble de fonctions dédiées est essentielle pour avoir de bonnes performances. La partie gauche de la figure 4.10 correspond à la phase d'apprentissage du modèle, à l'intérieur duquel nous avons testé différents ensembles de caractéristiques, y compris différentes combinaisons. Pendant la phase de test, nous exploitons les descripteurs identifiés comme étant les meilleurs pour chaque classifieur.

Pour choisir les descripteurs de chaque classifieur, nous avons regroupé dans le tableau 4.4 tous les résultats de F-score obtenus. Ces résultats nous ont permis de retenir les attributs statistiques (modèle proposé voir section (4.3)) pour le classifieur 1 et le classifieur 2, tandis que pour le

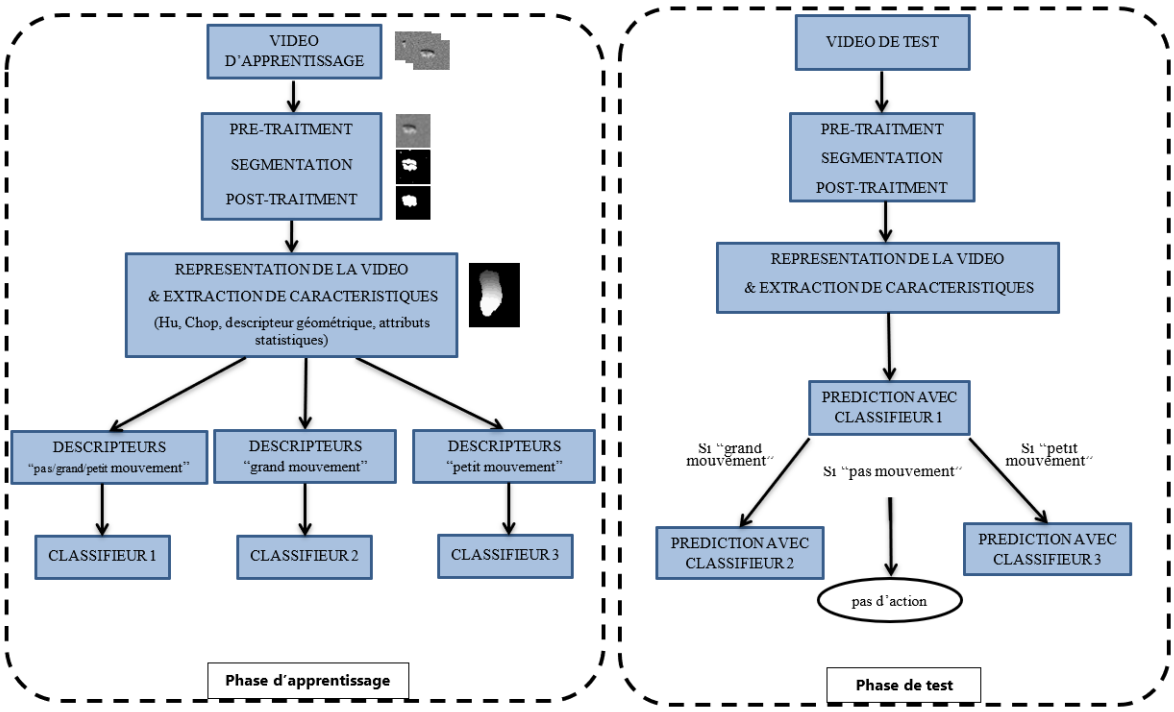


FIGURE 4.10 – Architecture du modèle en cascade

classifieur 3, nous retenons comme descripteur, la combinaison des moments de Hu et CHOP ($Hu \oplus CHOP$) qui permet d'avoir la plus grande valeur de F-score pour la catégorie de petit mouvement (83% de F-score).

Descripteurs	pas d'action/grand/petit mouvement	grand mouvement	petit mouvement
moment de Hu	69%	90%	61%
Descripteur géométrique(Geo)	97%	86%	47%
CHOP	97%	89%	81%
Geo⊕Hu	84%	86%	47%
Hu⊕CHOP	83%	93%	**83%**
Geo⊕CHOP	97%	87%	51%
Geo⊕Hu⊕CHOP	84%	87%	51%
attributs statistiques	**98%**	**97%**	65%

TABLE 4.4 – Valeurs de F-score pour chaque regroupement de classes

4.5.2 Résultat du modèle en cascade

En utilisant 3 classifieurs différents pour faire une classification en cascade, la valeur de F-score obtenue est de 89%. On a donc une augmentation de 6% par rapport à la meilleure approche de classification directe en 7 classes.

De plus, d'après le tableau 4.8, la matrice de confusion présente une uniformité au niveau des

taux de bonnes prédictions par classe. En comparant les résultats du modèle en cascade (tableau 4.8) à celui du modèle (MHI + CHOP) (tableau 4.2), on note nette diminution des erreurs de classification pour les classes qui étaient les moins bien reconnues, à savoir “s’asseoir” et “marche lente”.

	pas d’action	agitation	s’asseoir	se lever	tourner sur son siège	marche lente	marche rapide
pas d’action	**100**	0	0	0	0	0	0
agitation	0	**82**	2	5	11	0	0
s’asseoir	0	3	**82**	13	0	1	1
se lever	0	2	14	**80**	2	1	1
tourner sur son siège	1	10	2	1	**84**	2	0
marche lente	0	2	1	0	0	**94**	3
marche rapide	0	0	1	0	0	1	**98**

TABLE 4.5 – Matrice de confusion du modèle en cascade proposé avec 89% de F-score

4.6 Sélection de caractéristiques

Travailler avec un nombre élevé d’attributs augmente le risque de prendre en considération des attributs redondants ou corrélés ce qui rend ces algorithmes plus complexes (espace de stockage et temps d’apprentissage élevé) et parfois moins performants. La sélection de caractéristiques est considérée comme une étape de prétraitement importante dans l’apprentissage automatique. Dans la littérature on retrouve plusieurs approches de sélection de caractéristiques. Elles peuvent être divisées en 2 catégories :

— les algorithmes de classement des attributs (Feature ranking) [133] : ils consistent à ordonner l’ensemble des attributs de départ selon un critère d’évaluation et à sélectionner ensuite les k attributs les plus pertinents vis-à-vis du critère utilisé. Le choix du nombre optimal d’attribut à considérer après classement est crucial pour avoir des performances élevées.

— les algorithmes de recherche de sous-ensembles (subset selection) : ils consistent à rechercher le sous-ensemble d’attributs le plus pertinent selon un certain critère de sélection. Ces algorithmes doivent alors trouver le meilleur sous-ensemble d’attributs parmi les sous-ensembles candidats. C’est ainsi que différentes fonctions heuristiques peuvent être utilisées afin de réduire l’espace de recherche et le temps de calcul.

Étant donné que les algorithmes de recherche de sous-ensembles ne retournent pas toujours le sous ensemble optimal et qu’ils sont le plus souvent liés à un classifieur précis, notre choix s’est porté vers la famille des algorithmes de classement des attributs.

CFS(Correlation-based Feature selection)[134] est un algorithme qui classe des sous-

ensembles d'entités selon une fonction d'évaluation heuristique (voir équation 4.8). Le biais de la fonction d'évaluation est orienté vers des sous-ensembles contenant des entités fortement corrélées avec la classe et ayant une faible corrélation entre elles. Les entités non pertinentes doivent être ignorées car elles ont une faible corrélation avec la classe. Les fonctions redondantes doivent être masquées car elles seront fortement corrélées avec une ou plusieurs des fonctions restantes. Si la corrélation entre chacune des composantes et la sortie de la variable est connue (voir l'équation 4.7), et si la corrélation entre chaque paire de composantes est donnée, alors la corrélation entre l'ensemble des données et la variable externe peut être définie par :

$$correlation = \frac{\sum_i (x_i - \bar{x}_i)(y_i - \bar{y}_i)}{\sqrt{\sum_i (x_i - \bar{x}_i)^2} \sqrt{\sum_i (y_i - \bar{y}_i)^2}} \tag{4.7}$$

$$M_s = \frac{k\bar{r}_{cf}}{\sqrt{k + k(k-1)\bar{r}_{ff}}} \tag{4.8}$$

où M_s est l'heuristique d'un sous-ensemble de caractéristiques S contenant k caractéristiques, $\bar{r}_{cf}$ est la corrélation moyenne des classes de caractéristiques ($f \in S$) et $\bar{r}_{ff}$ est la corrélation moyenne entre les caractéristiques.

ILFS (Infinite Latent Feature Selection)[18] est un algorithme de sélection probabiliste robuste basé sur un graphe probabiliste qui considère tous les sous-ensembles possibles de caractéristiques, comme les chemins sur le graphe. Dans le graphe, les poids entre 2 noeuds représentent la probabilité que les caractéristiques soient de bons candidats pour l'ensemble final. La caractéristique la plus attrayante de la méthode ILFS est qu'elle modélise la pertinence d'un attribut en tant que variable latente dans un processus de génération inspiré des PLSA (Probabilistic latent semantic analysis) [135]. Ceci permet d'étudier l'importance d'une caractéristique quand elle est ajoutée à un ensemble d'attributs appelé tokens (dictionnaire de mots contenant des caractéristiques).

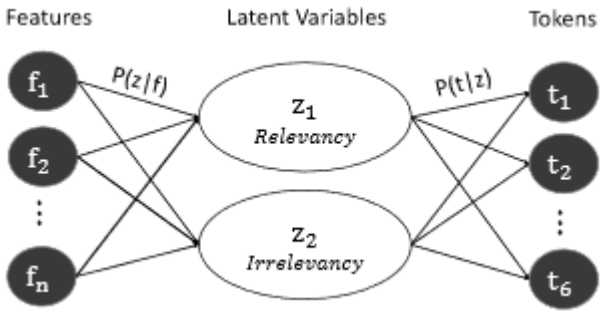


FIGURE 4.11 – Illustration de la structure générale de la méthode ILFS [18].

MRMR (Minimum Redundancy Maximum Relevance) [136] est une méthode de sélection de caractéristiques qui permet de sélectionner des caractéristiques qui sont mutuellement éloignées les unes des autres tout en ayant une corrélation élevée avec la variable de classification (c'est à dire les caractéristiques ayant une redondance minimale et une performance maximale). Cette méthode est basée sur des mesures statistiques classiques comme l'information mutuelle, la

corrélation, etc... Une étude de différentes méthodes permettant de comparer et d'évaluer la redondance minimale et de la pertinence maximale a été présentée dans [137]. Dans [136], les auteurs utilisent l'information mutuelle. La condition de la redondance minimale est donné par l' équation 4.9.

$$\min R_I, \qquad R_I = \frac{1}{|C|^2} \sum_{i,j \in C} I(i,j) \tag{4.9}$$

La condition de pertinence maximale consiste à maximiser la pertinence totale de toutes les caractéristique dans C. Elle est donnée par l'équation 4.10.

$$\max P_I, \qquad P_I = \frac{1}{|C|^2} \sum_{i \in C} I(i,Y) \tag{4.10}$$

Où C et $|C|$ représentent respectivement, l'ensemble et le nombre de caractéristiques. $I(i,j)$ est l'information mutuelle entre la ième et jème caractéristique ; $I(i,Y)$ est l'information mutuelle entre la ième caractéristique et l'ensemble des étiquettes de la classe Y. Le score d'une variable est la combinaison de ces deux facteurs tel que. Soient X et Y deux variables aléatoires dont les instances sont respectivement les valeurs de la ième caractéristique et les étiquettes des classes. $I(x_i)$ est l'information mutuelle entre la ième caractéristique et l'ensemble des étiquettes de la classe X. L'information mutuelle est estimée empiriquement par l'équation 4.11.

$$I(x_i) = \sum_{x,y} P(X = x_i, Y = y) log \frac{P(X = x_i, Y = y)}{P(X = x_i).P(Y = y)} \tag{4.11}$$

Où les $P(x_i)$, $P(y)$ et $P(x_i, y)$ sont estimées par les fréquences des différentes valeurs possibles.

L'ensemble des caractéristiques du MRMR est obtenu en optimisant les conditions des équations 4.9 et 4.10 simultanément. L'optimisation des deux conditions nécessite de les combiner en un seul critère. Deux critères simples de combinaison sont définis dans les équations 4.12 et 4.13 : Le score d'une caractéristique est le rapport entre le facteur pertinence et le facteur de redondance :

$$\max(\frac{P_I}{R_I}) \tag{4.12}$$

$$\max(P_I - R_I) \tag{4.13}$$

Résultat du modèle en cascade avec sélection de caractéristiques

Dans cette partie, nous utilisons la sélection de caractéristiques afin de voir premièrement laquelle des méthodes est meilleure pour le jeu de données utilisé et deuxièmement son impact sur le modèle en cascade. Rappelons que dans le modèle en cascade, on a 3 classifieurs. Les 2 premiers classifieurs n'ont que 12 attributs tandis que le classifieur 3 (celui de petit mouvement) en a 135. En plus de contenir un plus grand nombre d'attributs, il est formé de la combinaison de deux descripteus (Hu $\oplus$ CHOP). Notons qu'en combinant des descripteurs, on peut avoir des attributs redondants ou non pertinents. Ainsi les méthodes de sélection de caractéristiques concerneront uniquement ce classifieur. Le résultat de sélection de caractéristiques est présenté dans la figure 4.12.

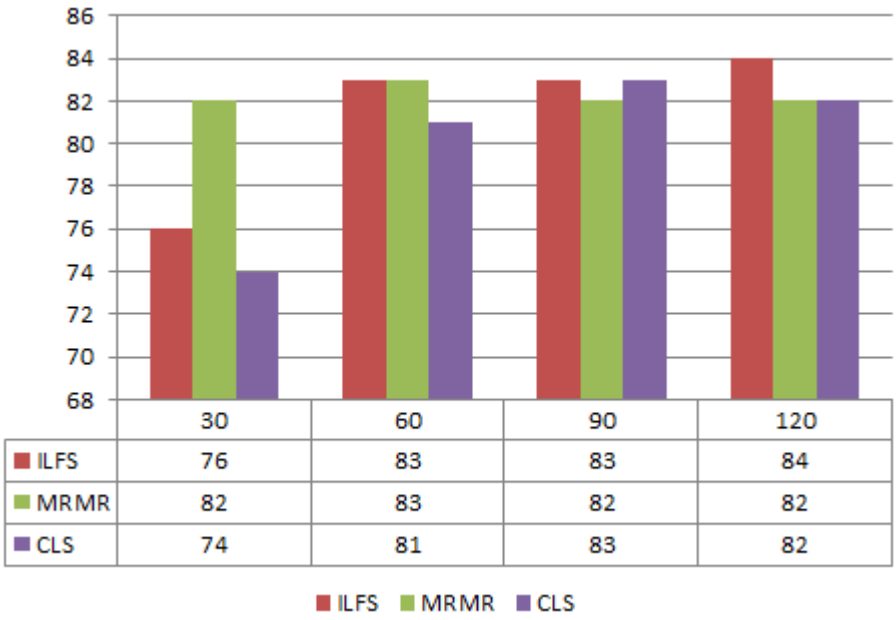


FIGURE 4.12 – Résultat de sélection de caractéristiques pour le classifieur 3 : petit mouvement.

Cette figure montre qu'au lieu des 135 descripteurs résultant de la combinaison de Hu et CHOP, nous pouvons utiliser les 120 premières caractéristiques sélectionnées par la méthode ILFS car elles permettent d'augmenter la performance du classifieur qui passe de 83% à 84%. Cette augmentation étant très faible, la performance globale du modèle en cascade reste inchangé et donc nous proposons de ne pas utiliser la sélection de caractéristiques dans notre système final car elle complexifie l'algorithme final avec l'adjonction d'une méthode de sélection de caractéristiques.

4.7 Étude comparative des approches classiques et des approches basées sur l'apprentissage profond (deep learning)

Dans cette section, nous explorons quelques approches du deep learning telles que : réseaux de neurones récurrents et réseaux de neurones à convolution, ceci dans le but de comparer les approches de l'apprentissage profond avec celles dites classiques présentées dans les paragraphes précédents en utilisant la même base de données. Le problème des approches classiques de reconnaissance d'actions est le suivant : pour avoir un bon modèle, il faut construire un bon extracteur de caractéristiques et ce dernier doit être repensé pour chaque nouvelle application. Les approches d'apprentissage profond offrent des alternatives intéressantes dans lesquelles la construction de l'ensemble de caractéristiques est faite de manière automatique par le modèle.

4.7.1 Quelques approches du deep learning

Nous présentons des architectures et décrivons quelques modèles du deep learning qui d'après l'état de l'art ont eu de bonnes performances sur d'autres bases de données. Chaque modèle a été adapté, modifié afin de produire de meilleurs résultats pour notre cas d'étude. L'implémentation des différents modèles est faite en python en utilisant la librairie Keras. Keras est une librairie Python qui encapsule l'accès aux fonctions proposées par plusieurs librairies de machine learning notamment tensorflow (une bibliothèque open-source développée par Google qui implémente des méthodes d'apprentissage automatique basées sur le principe des réseaux de neurones profonds).

Modèle 3D-CNN

Ce type de modèle est réalisé en convoluant un noyau 3D sur une séquence d'images. Le résultat de convolution donne des cartes de caractéristiques qui sont passées à un classifieur. Dans un réseau à convolution (CNN ou ConvNet) diverses architectures peuvent être conçues. Dans ce qui suit, nous décrivons l'architecture 3D-CNN que nous avons mise en place pour la reconnaissance d'actions humaines. L'architecture présentée est similaire à celle présentée dans [8]. Nous avons maintenu le nombre et différentes couches convolutives utilisées, changé la taille des noyaux de convolution pour tenir compte de la taille réduite des images sur lesquelles nous travaillons (64×64). Comme classifieur, nous utilisons un perceptron multi-couches. Notre modèle (voir figure 4.13) prend en entrée une séquence d'images (vidéo) de taille $64 \times 64 \times 15$.

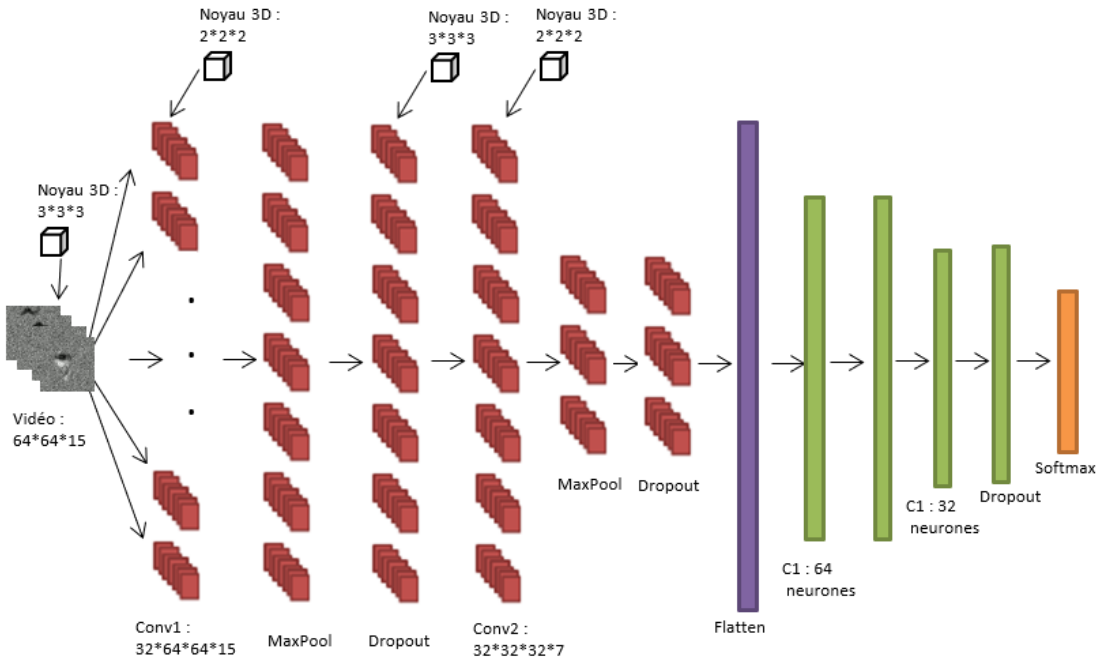


FIGURE 4.13 – Architecture 3D-CNN

Modèle LSTM

Le modèle LSTM présenté ici (voir figure 4.14) prend en entrée une vidéo. Cette vidéo est passée dans une couche appelée, "timedistributed" qui a la charge de repartir de manière temporelle les séquences d'images. Pour ce faire, la première image est passée à la première couche appelée cellule LSTM constituée de 50 neurones. Chaque unité (neurone) de chaque cellule LSTM fait la somme pondérée de tous les pixels de l'image à l'instant t plus les neurones de la cellule précédente. Ceci permet donc d'obtenir des informations dans les images précédentes avant de traiter l'image courante et de prendre une décision au niveau la dernière image.

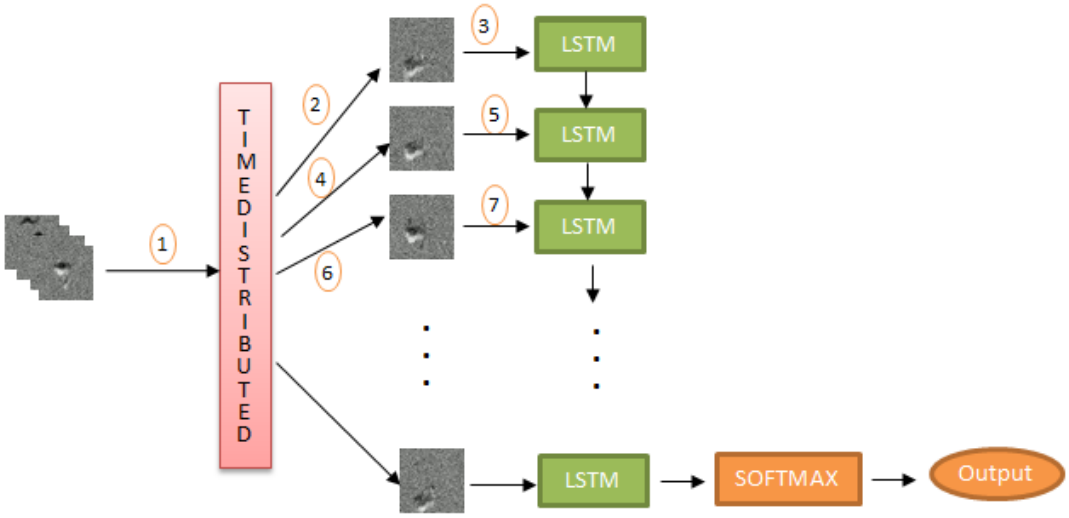


FIGURE 4.14 – Architecture LSTM

Modèle LRCN

Le LRCN (Long-term Recurrent Convolutional Network) [63] est un modèle de réseau convolutif récurrent à long terme (LRCN) combinant un extracteur visuel hiérarchique profond (tel qu'un CNN) avec un modèle qui peut apprendre à reconnaître et synthétiser la dynamique temporelle pour des tâches impliquant des données séquentielles (LSTM) (voir 4.15). Les modèles convolutifs récurrents sont "doublement profonds" dans la mesure où ils peuvent être composés de façon progressive, avec une grande précision aux niveaux spatial et temporel. De tels modèles peuvent avoir des avantages lorsque les concepts cibles sont complexes et/ou que les données d'entraînement sont limitées.

Modèle 3D-CNN+LSTM

Comme dans le modèle LRCN on utilise plutôt des convolutions 2D sur chaque image de la vidéo, nous proposons de tester un modèle 3D-CNN+LSTM qui est une combinaison du 3D-CNN avec le LSTM. Nous utilisons donc des convolutions 3D (les mêmes que dans la figure 4.13)

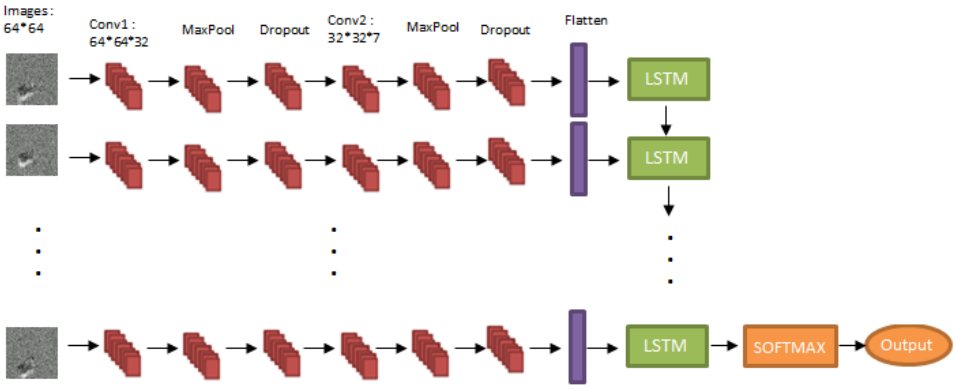


FIGURE 4.15 – Architecture LRCN

sur la vidéo pour extraire les cartes de caractéristiques spatio-temporel, puis nous utilisons ces caractéristiques en entrée du réseau de neurones récurrents (LSTM) pour la classification des données. L'architecture du dit modèle est présentée dans la figure 4.16.

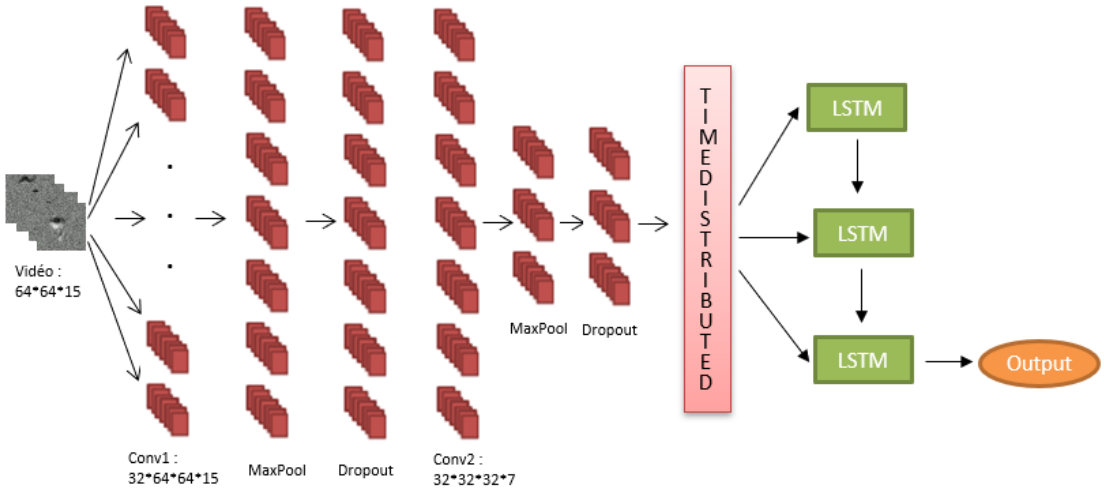


FIGURE 4.16 – Architecture 3D-CNN + LSTM

4.7.2 Résultats et interprétations

La base de données utilisée et les critères de comparaison sont identiques à ceux utilisés dans les sections précédentes.

Dans le tableau 4.6, nous présentons les différentes valeurs de F-score des modèles classiques et ceux du deep learning. Nous pouvons voir que le modèle en cascade proposé donne un meilleur résultat (89 % de F-score) que les autres modèles. Le second meilleur modèle est le 3D-CNN avec

une différence de 4% par rapport au meilleur. On conclut que malgré les performances excellentes que proposent les réseaux neurones par rapport aux approches classiques, l'apprentissage profond est moins performant dans notre contexte. Cela peut s'expliquer peut être par la quantité de données utilisées car nous savons que les modèles de la littérature basés sur les apprentissages profonds fonctionnent avec des milliers de données en entrée.

	Modèle	F-score *(7 classes)*
Approches classiques	meilleur modèle (CHOP)	83%
	modèle proposé (avec attribut statistique)	78%
	modèle en cascade proposé	**89%**
Approches deep learning	**3D-CNN**	**85%**
	LSTM	64%
	LRCN	81%
	3D-CNN+LSTM	65%

TABLE 4.6 – Récapitulatif des résultats des différents modèles

La matrice de confusion du meilleur modèle de l'approche par apprentissage profond est présentée dans le tableau 4.7. Dans cette matrice on note d'abord les faibles taux de prédiction obtenus pour la classe s'agiter. Ensuite on note une grande confusion entre la classe s'agiter et la classe pas d'action cela peut s'expliquer par le fait que les petits mouvements d'un membre du corps humain (le mouvement du bras par exemple pour répondre au téléphone) sont difficilement visibles sur les images provenant du capteur et donc ces images d'agitation peuvent être confondues avec les images ne correspondant à aucune action. Un autre constat est celui du faible taux au niveau de la classe marche lente qui est confondue avec de multiples actions, notamment s'asseoir, se lever et marche rapide.

Le tableau 4.8 correspond à la matrice de confusion du meilleur modèle de l'approche classique (modèle en cascade). Premièrement on peut noter que les taux de bonne prédiction sont plus uniformes que dans le tableau 4.7. Nous constatons quand même une légère confusion entre les classes "s'asseoir" et "se lever". En vue de dessus, le MHI construit pour ces deux classes se ressemblent fortement (voir 4.2 (c) et (d)).

4.7.3 Augmentation de la base données

Étant donnée que les méthodes d'apprentissages profonds ont généralement besoin d'un grand nombre de données pour fonctionner correctement, nous avons décidé d'augmenter la base de données pour tester les performances de ces méthodes. Plusieurs opérations peuvent être faites pour l'augmentation des données. Comme opération, on peut citer : la rotation, la translation, le changement d'intensités des valeurs des pixels, l'ajout ou la réduction des bruits,... Les choix des opérations pour l'augmentation d'une base de données peut entraîner une très grande baisse des performances. Par exemple lorsque nous avons ajouté des images filtrées dans la base, les performances des modèles ont considérablement chuté. Après de nombreux tests, nous avons choisi

	pas d'action	agitation	s'asseoir	se lever	tourner sur son siège	marche lente	marche rapide
pas d'action	**100**	0	0	0	0	0	0
agitation	22	**71**	1	0	6	0	0
s'asseoir	0	13	**84**	0	3	0	0
se lever	1	12	1	**84**	2	0	0
tourner sur son siège	1	12	0	1	**86**	0	0
marche lente	1	1	6	8	2	**77**	5
marche rapide	0	0	2	0	0	4	**94**

TABLE 4.7 – Matrice de confusion du modèle en 3D-CNN avec 85% de F-score

	pas d'action	agitation	s'asseoir	se lever	tourner sur son siège	marche lente	marche rapide
pas d'action	**100**	0	0	0	0	0	0
agitation	0	**82**	2	5	11	0	0
s'asseoir	0	3	**82**	13	0	1	1
se lever	0	2	14	**80**	2	1	1
tourner sur son siège	1	10	2	1	**84**	2	0
marche lente	0	2	1	0	0	**94**	3
marche rapide	0	0	1	0	0	1	**98**

TABLE 4.8 – Matrice de confusion du modèle en cascade proposé avec 89% de F-score

de considérer comme opérations la rotation et la translation. La base de données initiale comportait 700 vidéos. En modifiant la base, nous obtenons une base de 7000 vidéos soit 1000 vidéos par classe. Les méthodes de deep learning présentées à la section 4.7 ont été à nouveau testées sur cette nouvelle base. Nous utilisons la validation hold-out et considérons 80% des données pour l'apprentissage et 20% pour le test.

Le tableau 4.9 présente un récapitulatif des valeurs de F-score obtenues après augmentation de la base de données. Le détail de ces résultats est présenté en annexe. D'après ces résultats, le constat est que pour certaines méthodes, l'augmentation de la base a favorisé l'augmentation du score de prédiction (le cas du modèle 3D-CNN+LSTM) tandis que pour d'autres méthodes elle a entraîné une diminution (le cas du modèle 3D-CNN). Après l'augmentation de la base, le meilleur modèle obtenu (3DCNN +LSTM) atteint 86% de F-score. L'approche classique du modèle en cascade que nous avons proposé reste le meilleur modèle avec un F-score de 89% et c'est cette méthode qui sera utilisée dans notre système de reconnaissance d'actions.

Modèle	F-score
3D-CNN	66%
LSTM	72%
LRCN	78%
3D-CNN+LSTM	**86%**

TABLE 4.9 – Récapitulatif des résultats des différents modèles du deep learning avec augmentation de la base

4.8 Reconnaissance d'actions à partir des vidéos

Dans cette partie, nous présentons les différents choix mis en place pour obtenir un système de reconnaissance à partir d'un flux vidéo. Rappelons que la base de données sur laquelle nous avons travaillé et réalisé notre apprentissage pour la reconnaissance d'actions contient des extraits de séquences mono-action (ensemble d'images contenant une seule action) donc pour notre flux vidéo il sera nécessaire de trouver une méthode pour découper (segmenter) la vidéo.

La segmentation temporelle ou le découpage des séquences d'images pour la reconnaissance d'actions est une problématique souvent traitée dans la littérature. Le but est de retrouver les couples de positions (début, fin) dans une vidéo qui peuvent être utilisés de manière fiable à des fins de reconnaissance des actions humaines.

Des solutions utilisant des fenêtres glissantes de longueurs temporelles variables avec des pas de progression bien choisis peuvent être trouvées dans [138, 139, 140]. Dans [141] les auteurs ont proposé de calculer une fonction d'énergie depuis une fenêtre glissante de 1.8 secondes afin de décider du bloc d'images à considérer pour la reconnaissance d'actions.

Une autre famille de segmentation temporelle est la segmentation par apprentissage. Elle fait appel à des classifieurs binaires permettant d'extraire l'ensemble des images correspondant à l'action [142, 143].

Le découpage d'une séquence vidéo peut amener à considérer 5 cas de figure (voir figure 4.17) :

— le premier cas correspond à une séquence vidéo dans laquelle il n'y a aucune action
— le deuxième cas correspond à une séquence vidéo dans laquelle l'action effectuée est visible sur toutes les images
— le troisième cas correspond à une séquence vidéo dans laquelle l'action a lieu sur un ensemble d'images consécutives mais pas sur toutes les 15 images de la vidéo. Ce genre de scénario a le plus souvent été observé dans les actions correspondant aux petits mouvements car un petit mouvement s'effectue très souvent sur des durées brèves (s'asseoir, se lever...)
— le quatrième cas correspond à une séquence vidéo dans laquelle l'action est effectuée mais on perd ponctuellement la personne dans l'image. Ce cas s'obtient souvent lors d'un petit arrêt pendant l'exécution du mouvement
— le cinquième cas correspond à une séquence vidéo dans laquelle on observe 2 actions parfois différentes dans l'ensemble d'images. Par exemple, il s'agit d'un bloc d'images dans lequel

on retrouve la fin d'une action et ensuite le début d'une autre.

Les 3 premiers cas sont des cas idéaux pour la reconnaissance d'actions puisque ceux-ci correspondent à ce qui a été appris. Tandis que les 2 derniers sont des situations pour lesquelles il nous a fallu proposer une stratégie afin d'éviter trop d'erreur (faux positifs) dans le résultat final.

La stratégie de classification retenue nous a amené à définir une nouvelle classe appelée "indéfini" pour matérialiser le fait qu'on soit dans l'incapacité de classer l'action. Notons que la classe "indéfini" n'est pas apprise comme les autres classes et l'attribution d'une séquence vidéo à cette classe se fait après vérification de certaines conditions. A cette classe "indéfini", on associe donc les séquences de vidéos dans lesquelles le bloc d'images comporte un certain nombre d'images sans détection de mouvement. On ajoute également les blocs d'images n'ayant pas une grande similarité à un exemple se trouvant dans la base. Cette classe comportera donc aussi les blocs d'images qui correspondent aux zones de transition.

FIGURE 4.17 – Différents cas possibles dans une séquence vidéo : le gris correspond aux images dans lesquelles on observe la présence d'un mouvement ; le blanc aux images dans lesquelles il n'y a aucune observation

Pour la segmentation temporelle nous avons utilisé l'approche avec fenêtre glissante avec un pas de 1. Car elle est réaliste et applicable sur nos données.

Une fois 15 images du flux vidéo obtenues, nous détectons l'action correspondante à cette séquence requête en les passant au processus en cascade proposé. Les actions sont détectées en comparant le modèle d'une séquence requête avec les modèles associés aux séquences de référence en utilisant le classifieur KNN. L'algorithme 3 est l'algorithme final pour la reconnaissance d'actions sur des vidéos.

Nous avons effectué des tests sur 2 vidéos comportant plusieurs actions continues. Chacune des vidéos est faite par des acteurs différents. La consigne donnée aux acteurs était de s'arrêter après chaque action. Rappelons que lorsque la personne n'est pas en mouvement, il y a disparition de sa silhouette dans les images retournées par le capteur. Cette consigne nous a permis de repérer dans chaque vidéo le début et la fin de chaque action afin d'établir une vérité terrain pour la validation des résultats. Le dispositif expérimental est similaire à celui présenté à la figure 4.18. Ce dispositif

Algorithme 3 : Algorithme de reconnaissance d'actions

```
entrée : clip vidéo ;
sortie : Différentes actions effectuées;
paramètres;
construire le modèle pour le classifieur 1 ;
construire le modèle pour le classifieur 2 ;
construire le modèle pour le classifieur 3 ;
pour chaque 15 images de la vidéo faire
    Appliquer le filtre gaussien ;
    Extraire des caractéristiques pour le classifieur 1 (data_test_1);
    [resultat, score_prediction]=prédiction_classifieur_1(data_test_1) ;
    si score_prediction faible alors
        resultat="indéfini";
    sinon
        si data_test_1 == "pas_action" alors
            resultat = no_action;
        fin
        nb_image= compter le nombre d'images consécutives dans lesquelles on a un
         mouvement ;
        si nb_image > 5 alors
            si data_test_1 == "grand_mouvement" alors
                Extraire des caractéristiques pour le classifieur 2 (data_test_2);
                [resultat, score_prediction]=prédiction_classifieur_2 (data_test_2) ;
                si score_prediction faible alors
                    resultat="indéfini";
                fin
            sinon
                Extraire des caractéristiques pour le classifieur 3 (data_test_3);
                [resultat, score_prediction]=prédiction_classifieur_3 (data_test_3) ;
                si score_prediction faible alors
                    resultat="indéfini";
                fin
            fin
        sinon
            resultat="indéfini";
        fin
    fin
fin
retourner le graphe comportant la vérité terrain et les résultats prédits;
```

correspond à un environnement de bureau.

Scénario 1 :

Dans la première vidéo expérimentale, le scénario effectué est le suivant :

— traverser la zone en une marchant lentement,

— refaire le trajet inverse en marchant rapidement,

— re-entrer dans le zone en marchant lentement et s'arrêter à son poste de travail,

— s'asseoir sur la chaise,

— s'agiter en déplaçant son clavier puis s'arrêter,

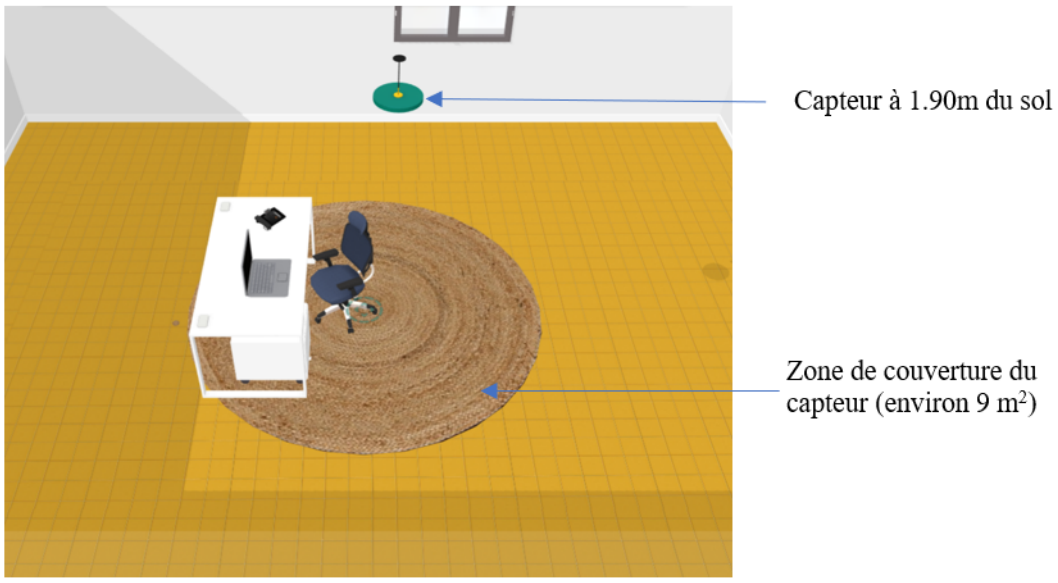


FIGURE 4.18 – Disposition de la salle

— se lever,
— marcher lentement et sortir de la zone couverte par le capteur.

Scénario 2 :

Dans la seconde vidéo expérimentale, le scénario effectué est le suivant :

— traverser la zone en une marchant lentement,
— refaire le trajet inverse en marchant rapidement,
— re-entrer dans le zone en marchant lentement et s'arrête à son poste de travail,
— s'asseoir sur la chaise,
— effectuer des mouvements de rotation étant assis puis s'arrêter,
— se lever,
— marcher lentement et sortir de la zone couverte par le capteur.

La figure 4.19 présente les résultats obtenus pour ces 2 scénarios. En rouge on présente la vérité terrain (donc les actions réellement menées par la personne) et en bleu les résultats provenant de l'algorithme 3. D'après ces résultats nous constatons que toutes les actions sont reconnues. Nous remarquons généralement des erreurs de classification en début ou en fin d'action. Cela est du au fait qu'en utilisant des fenêtres glissantes en début ou en fin d'action, nous avons des blocs d'images qui contiennent des images correspondant à aucune action et celle correspondant à un mouvement (soit celui de début, soit de fin d'action). La résolution de ce problème entraîne une étude approfondie sur le découpage de séquence vidéo pour la reconnaissance d'actions adapté à ce type de vidéo.

Dans un second temps, les industriels nous ont fait comprendre que la distinction entre certaines classes n'était pas si importante et qu'ils préfèrent un système comportant moins d'erreur de

classification. Ainsi nous avons refait l'apprentissage du modèle en regroupant un certain nombre de classes, à savoir : la classe s'asseoir et se lever, la classe agitation et tourner sur son siège et enfin la classe marche lente et marche rapide. En prenant en compte ce mélange de classes dans le modèle en cascade, on obtient de nouveaux résultats. En comparant les résultats du tableau 4.8 et ceux nouvellement obtenus en regroupant les classes (tableau 4.10), nous pouvons noter une augmentation de la valeur de F-score et moins d'erreur de classification entre les classes regroupées.

	pas d'action	agitation-tourner	s'asseoir- se lever	marche lente - marche rapide
pas d'action	**100**	0	0	0
agitation - tourner	0	**183**	16	1
s'asseoir - se lever	0	4	**196**	3
marche lente - rapide	0	2	2	**196**

TABLE 4.10 – Matrice de confusion du modèle en cascade proposé en regroupant certaines classes (96% de F-score)

Dans le cas où certaines classes ont été regroupées, l'algorithme final de reconnaissance d'actions sur des vidéos est également appliqué, et la figure 4.20 présente les résultats obtenus. En comparant ces résultats à ceux de la figure 4.19, le premier constat est l'obtention d'un résultat assez lisse au niveau du scénario 2. Comme on pouvait s'y attendre, plusieurs erreurs de classification observées précédemment dans la figure 4.19 ont disparues : par exemple, dans le scénario 2, autour du frame 380, l'algorithme de classification initial se trompe et classe un bloc d'image comme étant de la classe "tourner", tandis qu'il est de la classe "se lever". Cette erreur est corrigée après regroupement de certaines classes (voir figure 4.20 - scénario 2).

Dans le cadre du projet CoCaps ces résultats seront exploités par l'équipe qui travaille sur la fusion d'information. A chaque seconde l'équipe récupère des informations provenant de différentes sources afin de caractériser l'activité de la personne. Comme exemple d'activités, prenons le cas d'usage des maisons EHPADs qui est l'un des cas d'étude. On souhaite reconnaître par exemple l'activité "regarder la télé". Pour ce faire l'algorithme de prise de décision aura besoin d'information de localisation (résultat provenant des capteurs PIR confirmant la présence dans la zone où se trouve le téléviseur), de savoir si la personne s'est assise sur le siège (algorithme provenant du capteur image), et enfin de reconnaître le son provenant de l'appareil (résultat provenant du capteur son). Un ensemble de règles et conditions est donc défini par l'équipe de fusion d'information pour reconnaître un certain nombre d'activités effectuées par une personne dans la pièce.

4.9 Conclusion

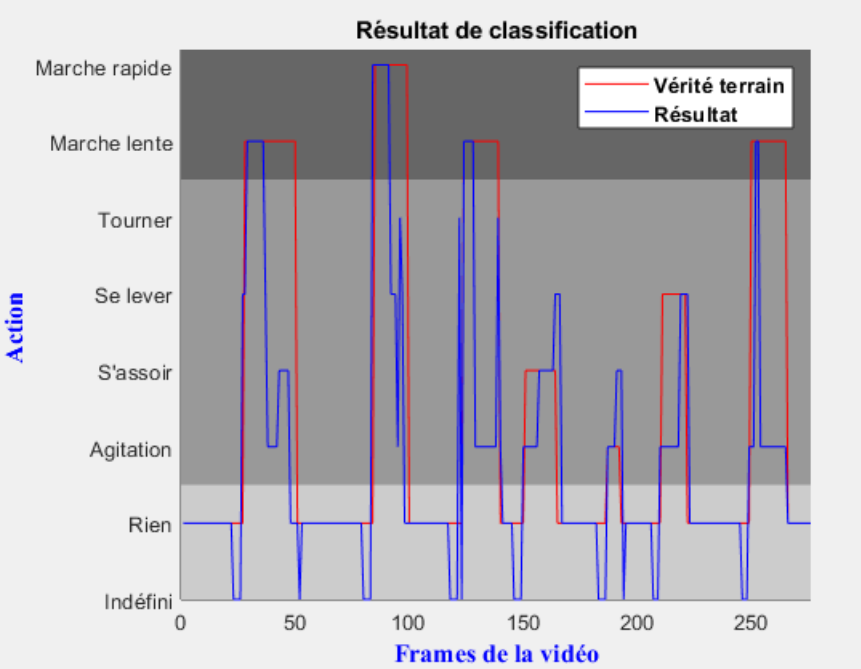
Dans le cadre de la reconnaissance d'actions 2 grandes catégories d'approches existent :

— L'approche classique qui consiste à extraire des caractéristiques directement des données. Ces caractéristiques sont extraites par un algorithme choisi par l'utilisateur. Les vecteurs de caractéristiques obtenus sont ensuite présentés en entrée d'un classifieur. Le point sensible

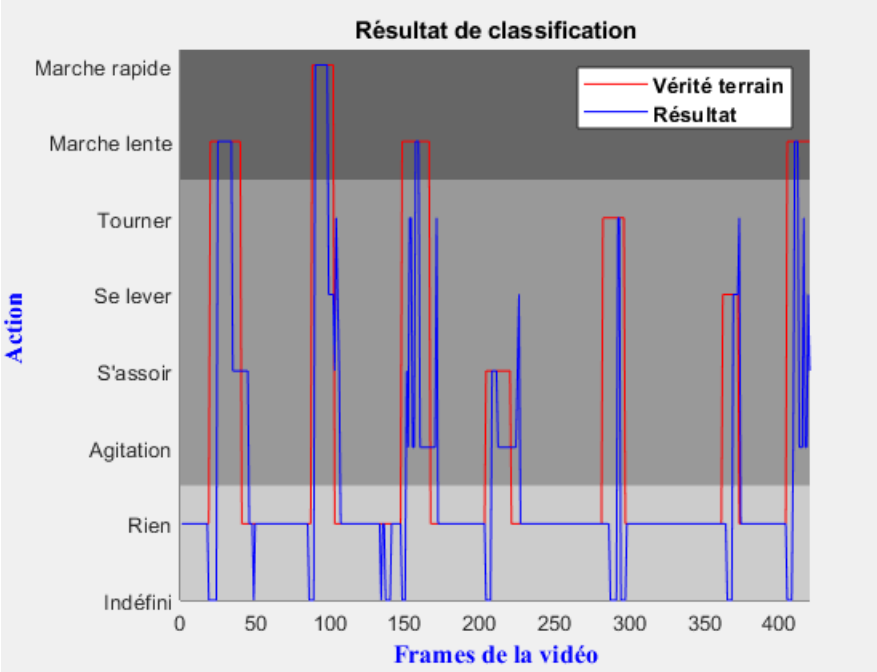
(l'extraction des caractéristiques) est laissé à la discrétion de l'utilisateur, et le choix de l'algorithme permettant l'extraction des caractéristiques est crucial et a motivé les études menées dans le chapitre précédent et dans ce présent chapitre pour proposer un modèle adapté aux images thermiques utilisées.

— L'approche par apprentissage profond qui prend en entrée d'un réseau de neurones la vidéo et se charge d'extraire automatiquement les caractéristiques pour construire le vecteur de caractéristiques lui même avant de la passer au classifieur.

Dans ce chapitre, nous avons proposé et comparé des méthodes classiques de reconnaissance d'actions basées sur l'extraction de silhouette du corps humain. En premier lieu nous avons extrait du MHI différents vecteurs descripteurs (Hu, Géométrique et CHOP) et testé leur performance. Ensuite nous avons présenté une modèle de reconnaissance d'actions qui exploite les distances entre centres de gravité des zones d'avancée-trainée pour représenter le mouvement dans une vidéo et un descripteur basé sur des attributs statistiques. Lors de l'interprétation des résultats, une idée d'amélioration du taux de performance nous a permis de proposer un modèle en cascade qui s'est avéré meilleur que les autres modèles présentés. Nous avons pu constater que la combinaison des descripteurs pouvait améliorer les performances d'un modèle ; par exemple dans notre cas d'étude $HU \oplus CHOP$ donne un meilleur taux pour le classifieur de petit mouvement. Pour une validation plus approfondie du résultat obtenu par le modèle en cascade, celui-ci a été comparé à certaines approches du deep learning.

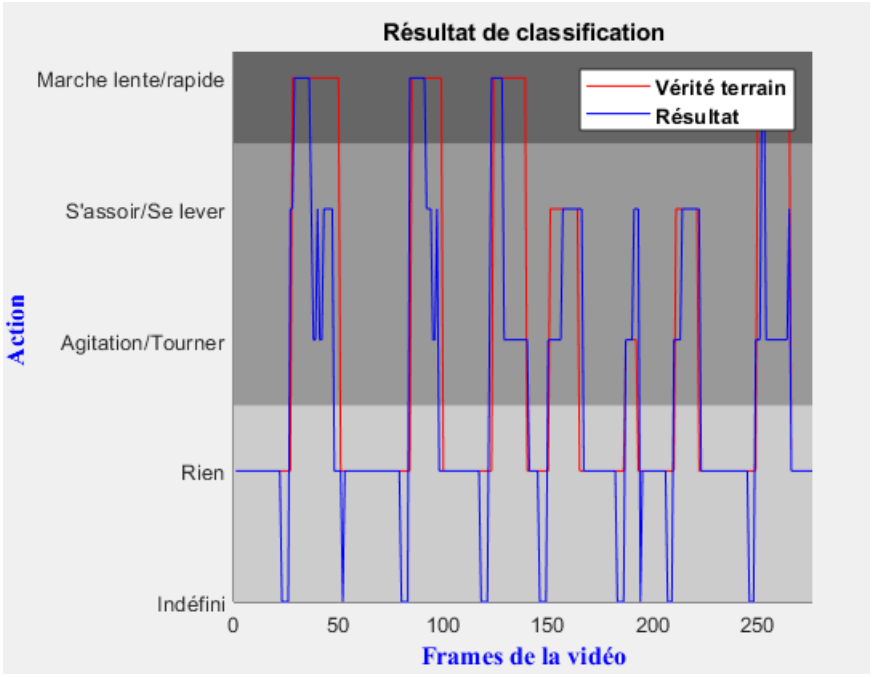


(a) Scénario 1

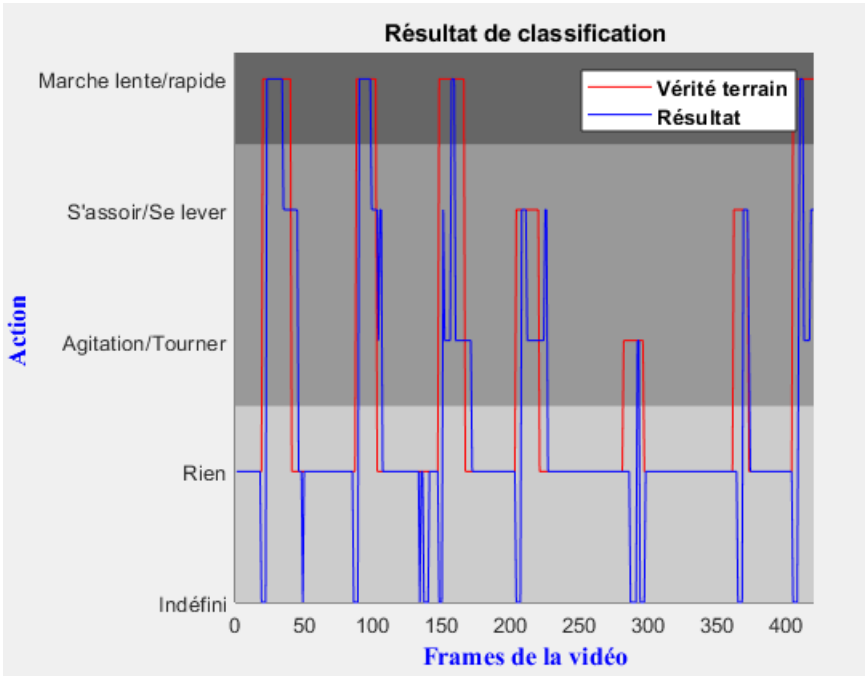


(b) Scénario 2

FIGURE 4.19 – Résultat de classification sur des clips vidéos : en rouge la vérité terrain et en bleu le résultat de classification



(a) Scénario 1



(b) Scénario 2

FIGURE 4.20 – Résultat de classification sur des clips vidéos avec réduction des classes : en rouge la vérité terrain et en bleu le résultat de classification

Chapitre 5

Conclusion

Sommaire

5.1 Synthèse de mes contributions

Dans ce travail, nous nous sommes intéressés à la reconnaissance d'actions à partir de séquences vidéo provenant d'un imageur infrarouge basse résolution ayant des propriétés différentes de celles souvent utilisées dans d'autres cas d'études. Pour traiter cette problématique, nous avons proposé des approches qui s'appuient sur l'extraction d'analyse des silhouettes. Afin d'extraire la silhouette des personnes dans des images, nous avons proposé un algorithme de segmentation par seuillage à seuil adaptatif, ensuite nous avons proposé une méthode de post-traitement qui consiste à ajouter des pixels sur les résultats de segmentation afin de retrouver des formes similaires à celles observées dans les images initiales. Dans le but de trouver la chaîne d'analyse d'images adaptée au capteur utilisé, nous avons effectué une étude comparative en couplant des méthodes de filtrage et de segmentation pour déterminer le meilleur couplage à exploiter pour l'application visée. De cette étude, nous avons conclu que pour extraire au mieux la silhouette des personnes dans les images, il faut appliquer comme prétraitement sur des images le filtre gaussien, ensuite utiliser l'algorithme de segmentation et de post-traitement proposé.

Pour la première approche de reconnaissance d'actions, nous avons exploité les silhouettes extraites (images segmentées) et construit le MHI comme représentation des vidéos, ensuite des caractéristiques pertinentes (les moments de Hu, les descripteurs géométriques, CHOP) ont été extraites du MHI pour être passées au classifieur KNN. Après comparaison des résultats le meilleur modèle est le MHI utilisé avec le descripteur CHOP.

Pour la deuxième approche toujours en exploitant les images segmentées, nous avons proposé un modèle qui consiste à représenter la séquence vidéo à l'aide des distances entre les centres de gravité des formes segmentées. De cette représentation des attributs statistiques sont extraits et

passés à un KNN. Ce modèle s'est montré plus efficace pour séparer les actions correspondant au déplacement du centre de gravité de la personne (ceux que nous avons défini comme étant les grands mouvements).

Dans l'optique d'améliorer les performances obtenues sur les deux modèles précédents, nous avons extrait leurs points forts afin de proposer une méthode en cascade avec 3 classifieurs différents.

Notre dernière contribution est la comparaison des approches classiques et des approches par apprentissage profond. Pour ce faire nous avons implémenté quelques modèles de deep learning et comparé leur performance. L'approche en cascade s'est avérée plus efficace que les autres modèles exploités ou proposés dans notre travail.

Nous avons terminé la rédaction de notre manuscrit en présentant la stratégie mise en place et en présentant les résultats obtenus de reconnaissance d'actions sur des clips vidéo contenant plusieurs actions successives. Ces résultats seront exploités par l'équipe travaillant sur la fusion d'informations pour caractériser l'activité humaine dans une pièce. Ils prennent en entrée des données provenant de plusieurs modalités à savoir : les données prétraitées (les résultats de localisation à l'aide des PIR, les résultats de reconnaissance d'actions et la parole) et les données brutes (détecteur d'alitement, eco-compteur).

5.2 Perspectives

A l'issu de ce travail, de nombreuses perspectives s'ouvrent sur les divers sujets traités.

— Au niveau de la reconnaissance d'actions dans une vidéo contenant plusieurs actions, une étude approfondie sur la découpe de manière automatique de la séquence vidéo est nécessaire, ceci dans le but de retrouver le début et la fin d'une action avant la passer dans le classifieur. Au lieu d'utiliser des fenêtres coulissantes qui se chevauchent et causent un problème de classification pour des zones de transition, deux approches peuvent être testées. Nous pourrions dans un premier temps apprendre dans le modèle des ensembles d'images qui correspondent aux zones de transition. Nous pourrions également utiliser une méthode qui génère un ensemble de propositions de blocs d'actions.

— Notre modèle en cascade utilise deux ensembles de descripteurs. L'un encode des informations spatio-temporelles (MHI + CHOP) et l'autre encode des informations sur d'apparence et de mouvement (attributs statistiques). Nous pourrions par exemple proposer un seul descripteur qui tient compte à la fois de ces deux informations dans le but d'économiser le temps de traitement.

— La question de la reconnaissance d'actions avec plus d'une personne dans la pièce surveillée n'a pas été traité dans nos travaux. Pour y parvenir, nous pourrions traquer tout au long de la vidéo les différentes régions où se trouve une personne et ensuite traiter chaque région de manière individuelle comme nous le faisons dans le cas d'une personne. L'étude des interactions entre les personnes nécessiterait également d'être approfondie.

— Au niveau de l'étude comparative avec des méthodes d'apprentissage profond, nous trouvons

qu'elle n'est pas assez conséquente car le jeu de données réelles utilisé est très petit par rapport à ceux qui sont souvent utilisés par ces méthodes. Ainsi, une augmentation de la base de données peut entraîner une amélioration des performances des modèles de deep learning. Nous avons exploré cette piste en augmentant la base de données (en faisant un choix judicieux des opérations d'augmentation de la base) et en optimisant les différents paramètres du modèle, mais les résultats ne sont pas restés cohérents par rapport au cas de petite base (700 données). Il serait donc intéressant d'avoir plus de données réelles pour valider notre étude comparative.

— A plus long terme, pour la sécurité des personnes, il serait intéressant d'étendre nos travaux afin de reconnaître par exemple les cas de chute des personnes. En d'autres termes ils s'agirait d'ajouter dans la base de données des vidéos de chute et tester si les modèles restent pertinents avec l'ajout d'autres actions.

— Les recherches montrent que l'homme est capable de prédire ce qui se passera dans le futur en se basant sur certaines observations. Nous aimerions aussi avoir un système qui, en fonction des actions effectuées précédemment, soit capable de prédire l'action qu'effectuera la personne.

ANNEXE : Résultats des modèles du deep learning après augmentation de la base de données

	precision	recall	f1-score	support
no_action	0.49	1.00	0.66	204
restless	0.14	0.14	0.14	188
sit_down	0.74	0.63	0.68	205
get_up	0.93	0.81	0.87	199
turn	0.64	0.18	0.28	178
walk_slow	1.00	0.80	0.89	208
walk_fast	0.99	0.99	0.99	218
avg / total	0.71	0.67	0.66	1400

FIGURE 5.1 – Modèle 3D-CNN

	precision	recall	f1-score	support
pas d'action	0.80	0.61	0.69	204
agitation	0.43	0.56	0.48	188
s'asseoir	0.72	0.87	0.79	404
se lever	0.66	0.37	0.47	178
tourner	0.88	0.83	0.85	208
marche lente	0.91	0.89	0.90	218
avg / total	0.74	0.72	0.72	1400

FIGURE 5.2 – Modèle LSTM

	precision	recall	f1-score	support
no_action	1.00	1.00	1.00	204
restless	0.99	0.99	0.99	393
sit_down	0.96	0.96	0.96	199
get_up	0.98	1.00	0.99	178
turn	0.48	0.98	0.65	208
walk_slow	0.00	0.00	0.00	218
avg / total	0.76	0.83	0.78	1400

final accuracy:83.2857142857

FIGURE 5.3 – Modèle LRCN

	precision	recall	f1-score	support
pas d'action	0.88	1.00	0.93	204
agitation	0.66	0.81	0.72	188
s'asseoir	0.84	0.95	0.89	205
se lever	0.88	0.89	0.89	199
tourner	0.86	0.41	0.56	178
marche lente	0.97	0.94	0.96	208
marche rapide	1.00	0.98	0.99	218
avg / total	0.87	0.86	0.86	1400

FIGURE 5.4 – Modèle 3D-CNN + LSTM

Publications

Les travaux menés au cours de cette thèse ont été publiés dans des conférences nationales et internationales. Ci dessous la liste des publications :

— Polla, F., Boudjelaba, K., Emile, B., Laurent, H. (2017). “Segmentation d’images infrarouge pour l’assistance aux personnes à domicile”. colloque JETSAN.

— Polla, F., Boudjelaba, K., Emile, B., et Laurent, H. (2017, September). “Proposal of segmentation method adapted to the infrared sensor”. In International Conference on Advanced Concepts for Intelligent Vision Systems (pp. 639-650). Springer.

— Polla, F., Laurent, H., Emile, B. (2019, June). “Action recognition from low-resolution infrared sensor for indoor use : a comparative study between deep learning and classical approaches“”. In 2019 20th IEEE International Conference on Mobile Data Management (MDM) (pp. 409-414). IEEE

— Polla, F., Laurent, H., Emile, B. (2020). “A Hierarchical Approach for Indoor Action Recognition from New Infrared Sensor Preserving Anonymity”. In VISAPP (pp. 229-236).

Bibliographie

[1] Ivan Laptev and Tony Lindeberg. Space-time interest points, in iccv. 2003.

[2] Ross Messing, Chris Pal, and Henry Kautz. Activity recognition using the velocity histories of tracked keypoints. In *2009 IEEE 12th international conference on computer vision*, pages 104–111. IEEE, 2009.

[3] Stephen Karungaru, Masayuki Daikoku, and Kenji Terada. Multi cameras based indoors human action recognition using fuzzy rules. *Journal of Pattern Recognition Research*, 1 :61–74, 2015.

[4] Aaron F Bobick and James W Davis. The recognition of human movement using temporal templates. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, (3) :257–267, 2001.

[5] Ilias Theodorakopoulos, Dimitris Kastaniotis, George Economou, and Spiros Fotopoulos. Pose-based human action recognition via sparse representation in dissimilarity space. *Journal of Visual Communication and Image Representation*, 25(1) :12–23, 2014.

[6] Vasileios Belagiannis, Sikandar Amin, Mykhaylo Andriluka, Bernt Schiele, Nassir Navab, and Slobodan Ilic. 3d pictorial structures for multiple human pose estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1669–1676, 2014.

[7] Gunnar Johansson. Visual perception of biological motion and a model for its analysis. *Perception & psychophysics*, 14(2) :201–211, 1973.

[8] Shuiwang Ji, Wei Xu, Ming Yang, and Kai Yu. 3d convolutional neural networks for human action recognition. *IEEE transactions on pattern analysis and machine intelligence*, 35(1) :221–231, 2013.

[9] Karen Simonyan and Andrew Zisserman. Two-stream convolutional networks for action recognition in videos. In *Advances in neural information processing systems*, pages 568–576, 2014.

[10] Limin Wang, Yu Qiao, and Xiaoou Tang. Action recognition with trajectory-pooled deep-convolutional descriptors. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4305–4314, 2015.

[11] Moez Baccouche, Franck Mamalet, Christian Wolf, Christophe Garcia, and Atilla Baskurt. Sequential deep learning for human action recognition. In *International workshop on human behavior understanding*, pages 29–39. Springer, 2011.

[12] Yong Du, Wei Wang, and Liang Wang. Hierarchical recurrent neural network for skeleton based action recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1110–1118, 2015.

[13] Christophe Escriba. *Conception, Réalisation et Caractérisation de capteurs infrarouges à thermopiles : Application à la détection de présence passive dans l'habitat.* PhD thesis, Université Paul Sabatier-Toulouse III, 2005.

[14] IRLYNX. Particularité d'une image thermique différentielle, 2017.

[15] O Barch and M Van Droogenbrock. Vibe : a universal background subtraction algorithm for video sequences. *IEEE Transactions on Image Processing*, 20(6) :1709–1724, 2011.

[16] Pierre-Luc St-Charles, Guillaume-Alexandre Bilodeau, and Robert Bergevin. Flexible background subtraction with self-balanced local sensitivity. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 408–413, 2014.

[17] Venkataraman B. Data science and analytics, https ://dslytics.wordpress.com/2017/11/16/classification-series-5-k-nearest-neighbors-knn/, 2020.

[18] Giorgio Roffo, Simone Melzi, Umberto Castellani, and Alessandro Vinciarelli. Infinite latent feature selection : A probabilistic latent graph-based ranking approach. In *Computer Vision and Pattern Recognition (CVPR)*, 2017.

[19] Pôle Capteurs. Annexe technique du projet, Réunion coordination, 2016.

[20] James Fogarty, Carolyn Au, and Scott E Hudson. Sensing from the basement : a feasibility study of unobtrusive and low-cost home activity recognition. In *Proceedings of the 19th annual ACM symposium on User interface software and technology*, pages 91–100. ACM, 2006.

[21] Ahmad Jalal, Yeon-Ho Kim, Yong-Joong Kim, Shaharyar Kamal, and Daijin Kim. Robust human activity recognition from depth video using spatiotemporal multi-fused features. *Pattern recognition*, 61 :295–308, 2017.

[22] Ivan Laptev, Marcin Marszałek, Cordelia Schmid, and Benjamin Rozenfeld. Learning realistic human actions from movies. In *CVPR 2008-IEEE Conference on Computer Vision & Pattern Recognition*, pages 1–8. IEEE Computer Society, 2008.

[23] Yannick Benezeth, Hélène Laurent, Bruno Emile, and Christophe Rosenberger. Towards a sensor for detecting human presence and characterizing activity. *Energy and Buildings*, 43(2-3) :305–314, 2011.

[24] Lei Han, Xinxiao Wu, Wei Liang, Guangming Hou, and Yunde Jia. Discriminative human action recognition in the learned hierarchical manifold space. *Image and Vision Computing*, 28(5) :836–849, 2010.

[25] Jun Liu, Amir Shahroudy, Dong Xu, and Gang Wang. Spatio-temporal lstm with trust gates for 3d human action recognition. In *European Conference on Computer Vision*, pages 816–833. Springer, 2016.

[26] F Barnier and R Chekkar. Building automation, an acceptable solution to dependence ? responses through an acceptability survey about a sensors platform. *IRBM*, 39(3) :167–179, 2018.

[27] Simeon Keates. Pragmatic research issues confronting hci practitioners when designing for universal access. *Universal Access in the Information Society*, 5(3) :269–278, 2006.

[28] Ivan Laptev, Barbara Caputo, Christian Schüldt, and Tony Lindeberg. Local velocity-adapted motion events for spatio-temporal recognition. *Computer vision and image understanding*, 108(3) :207–229, 2007.

[29] Piotr Dollár, Vincent Rabaud, Garrison Cottrell, and Serge Belongie. Behavior recognition via sparse spatio-temporal features. VS-PETS Beijing, China, 2005.

[30] Pyry Matikainen, Martial Hebert, and Rahul Sukthankar. Trajectons : Action recognition through the motion analysis of tracked features. In *2009 IEEE 12th international conference on computer vision workshops, ICCV workshops*, pages 514–521. IEEE, 2009.

[31] Ju Sun, Xiao Wu, Shuicheng Yan, Loong-Fah Cheong, Tat-Seng Chua, and Jintao Li. Hierarchical spatio-temporal context modeling for action recognition. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2004–2011. IEEE, 2009.

[32] Heng Wang, Alexander Kläser, Cordelia Schmid, and Cheng-Lin Liu. Dense trajectories and motion boundary descriptors for action recognition. *International journal of computer vision*, 103(1) :60–79, 2013.

[33] Heng Wang and Cordelia Schmid. Action recognition with improved trajectories. In *Proceedings of the IEEE international conference on computer vision*, pages 3551–3558, 2013.

[34] Mohamed Bécha Kaaniche and François Brémond. Tracking hog descriptors for gesture recognition. In *2009 Sixth IEEE International Conference on Advanced Video and Signal Based Surveillance*, pages 140–145. IEEE, 2009.

[35] Geert Willems, Tinne Tuytelaars, and Luc Van Gool. An efficient dense and scale-invariant spatio-temporal interest point detector. In *European conference on computer vision*, pages 650–663. Springer, 2008.

[36] Xiantong Zhen and Ling Shao. Spatio-temporal steerable pyramid for human action recognition. In *2013 10th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG)*, pages 1–6. IEEE, 2013.

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

[38] Trevor Hastie, Saharon Rosset, Ji Zhu, and Hui Zou. Multi-class adaboost. *Statistics and its Interface*, 2(3) :349–360, 2009.

[39] Paul Scovanner, Saad Ali, and Mubarak Shah. A 3-dimensional sift descriptor and its application to action recognition. In *Proceedings of the 15th ACM international conference on Multimedia*, pages 357–360. ACM, 2007.

[40] G Lowe. Sift-the scale invariant feature transform. *Int. J.*, 2 :91–110, 2004.

[41] Lahav Yeffet and Lior Wolf. Local trinary patterns for human action recognition. In *2009 IEEE 12th international conference on computer vision*, pages 492–497. IEEE, 2009.

[42] Alexander Klaser, Marcin Marszałek, and Cordelia Schmid. A spatio-temporal descriptor based on 3d-gradients. In *BMVC 2008-19th British Machine Vision Conference*, pages 275–1. British Machine Vision Association, 2008.

[43] Navneet Dalal, Bill Triggs, and Cordelia Schmid. Human detection using oriented histograms of flow and appearance. In *European conference on computer vision*, pages 428–441. Springer, 2006.

[44] Vadim Kantorov and Ivan Laptev. Efficient feature extraction, encoding and classification for action recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2593–2600, 2014.

[45] Guoying Zhao and Matti Pietikainen. Dynamic texture recognition using local binary patterns with an application to facial expressions. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, (6) :915–928, 2007.

[46] Eli Shechtman and Michal Irani. Space-time behavior based correlation. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pages 405–412. IEEE, 2005.

[47] Ling Shao and Xiuli Chen. Histogram of body poses and spectral regression discriminant analysis for human action categorization. In *BMVC*, pages 1–11, 2010.

[48] Lena Gorelick, Moshe Blank, Eli Shechtman, Michal Irani, and Ronen Basri. Actions as space-time shapes. *IEEE transactions on pattern analysis and machine intelligence*, 29(12) :2247–2253, 2007.

[49] Pengfei Zhu, Weiming Hu, Li Li, and Qingdi Wei. Human activity recognition based on r-transform and fourier mellin transform. In *International Symposium on Visual Computing*, pages 631–640. Springer, 2009.

[50] Kellokumpu Vili, Zhao Guoying, and Pietikäinen Matti. Texture based description of movements for activity analysis. In *Int. Conf. on Computer Vision Theory and Applications (VISAPP 2008)*, volume 1, pages 206–213, 2008.

[51] Feng Zheng, Ling Shao, and Zhan Song. Eigen-space learning using semi-supervised diffusion maps for human action recognition. In *Proceedings of the ACM International Conference on Image and Video Retrieval*, pages 151–157. ACM, 2010.

[52] Moshe Blank, Lena Gorelick, Eli Shechtman, Michal Irani, and Ronen Basri. Actions as space-time shapes. In *The Tenth IEEE International Conference on Computer Vision (ICCV'05)*, pages 1395–1402, 2005.

[53] Elden Yu and Jake K Aggarwal. Human action recognition with extremities as semantic posture representation. In *2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, pages 1–8. IEEE, 2009.

[54] Yaser Sheikh, Mumtaz Sheikh, and Mubarak Shah. Exploring the space of a human action. In *Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1*, volume 1, pages 144–149. IEEE, 2005.

[55] Lu Xia, Chia-Chih Chen, and Jake K Aggarwal. View invariant human action recognition using histograms of 3d joints. In *2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, pages 20–27. IEEE, 2012.

[56] Alejandro Baldominos, Yago Saez, and Pedro Isasi. Evolutionary design of convolutional neural networks for human activity recognition in sensor-rich environments. *Sensors*, 18(4) :1288, 2018.

[57] Ahmad Jalal, Maria Mahmood, and Abdul S Hasan. Multi-features descriptors for human activity tracking and recognition in indoor-outdoor environments. In *2019 16th International Bhurban Conference on Applied Sciences and Technology (IBCAST)*, pages 371–376. IEEE, 2019.

[58] Benoît Delachaux, Julien Rebetez, Andres Perez-Uribe, and Héctor Fabio Satizábal Mejia. Indoor activity recognition by combining one-vs.-all neural network classifiers exploiting wearable and depth sensors. In *International Work-Conference on Artificial Neural Networks*, pages 216–223. Springer, 2013.

[59] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 4489–4497, 2015.

[60] Mike Schuster and Kuldip K Paliwal. Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, 45(11) :2673–2681, 1997.

[61] James Martens and Ilya Sutskever. Learning recurrent neural networks with hessian-free optimization. In *Proceedings of the 28th international conference on machine learning (ICML-11)*, pages 1033–1040. Citeseer, 2011.

[62] Wojciech Zaremba, Ilya Sutskever, and Oriol Vinyals. Recurrent neural network regularization. *arXiv preprint arXiv :1409.2329*, 2014.

[63] Jeffrey Donahue, Lisa Anne Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Kate Saenko, and Trevor Darrell. Long-term recurrent convolutional networks for visual recognition and description. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2625–2634, 2015.

[64] Zuxuan Wu, Xi Wang, Yu-Gang Jiang, Hao Ye, and Xiangyang Xue. Modeling spatial-temporal clues in a hybrid deep learning framework for video classification. In *Proceedings of the 23rd ACM international conference on Multimedia*, pages 461–470. ACM, 2015.

[65] Ji-Hae Kim, Gwang-Soo Hong, Byung-Gyu Kim, and Debi P Dogra. deepgesture : Deep learning-based gesture recognition scheme using motion sensors. *Displays*, 55 :38–45, 2018.

[66] Nitish Srivastava, Elman Mansimov, and Ruslan Salakhudinov. Unsupervised learning of video representations using lstms. In *International conference on machine learning*, pages 843–852, 2015.

[67] Joe Yue-Hei Ng, Matthew Hausknecht, Sudheendra Vijayanarasimhan, Oriol Vinyals, Rajat Monga, and George Toderici. Beyond short snippets : Deep networks for video classification.

In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4694–4702, 2015.

[68] Jeffrey L Elman. Distributed representations, simple recurrent networks, and grammatical structure. *Machine learning*, 7(2-3) :195–225, 1991.

[69] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8) :1735–1780, 1997.

[70] Sheeraz Arif, Jing Wang, Tehseen Ul Hassan, and Zesong Fei. 3d-cnn-based fused feature maps with lstm applied to action recognition. *Future Internet*, 11(2) :42, 2019.

[71] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. Ucf101 : A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv :1212.0402*, 2012.

[72] H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, and T. Serre. HMDB : a large video database for human motion recognition. In *Proceedings of the International Conference on Computer Vision (ICCV)*, 2011.

[73] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout : a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1) :1929–1958, 2014.

[74] MultiMedia LLC. Recognition of human actions : action dataset, 2005.

[75] Wanqing Li, Zhengyou Zhang, and Zicheng Liu. Action recognition based on a bag of 3d points. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition-Workshops*, pages 9–14. IEEE, 2010.

[76] Marcin Marszałek, Ivan Laptev, and Cordelia Schmid. Actions in context. In *CVPR 2009-IEEE Conference on Computer Vision & Pattern Recognition*, pages 2929–2936. IEEE Computer Society, 2009.

[77] Jaeyong Sung, Colin Ponce, Bart Selman, and Ashutosh Saxena. Unstructured human activity detection from rgbd images. In *2012 IEEE international conference on robotics and automation*, pages 842–849. IEEE, 2012.

[78] Jiang Wang, Zicheng Liu, Ying Wu, and Junsong Yuan. Mining actionlet ensemble for action recognition with depth cameras. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1290–1297. IEEE, 2012.

[79] Chen Chen, Roozbeh Jafari, and Nasser Kehtarnavaz. Utd-mhad : A multimodal dataset for human action recognition utilizing a depth camera and a wearable inertial sensor. In *2015 IEEE International conference on image processing (ICIP)*, pages 168–172. IEEE, 2015.

[80] Basura Fernando, Efstratios Gavves, Jose M Oramas, Amir Ghodrati, and Tinne Tuytelaars. Modeling video evolution for action recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5378–5387, 2015.

[81] Cordelia Schmid, Roger Mohr, and Christian Bauckhage. Evaluation of interest point detectors. *International Journal of computer vision*, 37(2) :151–172, 2000.

[82] Shireen Y Elhabian, Khaled M El-Sayed, and Sumaya H Ahmed. Moving object detection in spatial domain using background removal techniques-state-of-art. *Recent patents on computer science*, 1(1) :32–54, 2008.

[83] Thierry Bouwmans, Fida El Baf, and Bertrand Vachon. Statistical background modeling for foreground detection : A survey. In *Handbook of pattern recognition and computer vision*, pages 181–199. World Scientific, 2010.

[84] Yannick Benezeth, Pierre-Marc Jodoin, Bruno Emile, Hélène Laurent, and Christophe Rosenberger. Review and evaluation of commonly-implemented background subtraction algorithms. In *2008 19th International Conference on Pattern Recognition*, pages 1–4. IEEE, 2008.

[85] Andrew HS Lai and Nelson HC Yung. A fast and accurate scoreboard algorithm for estimating stationary backgrounds in an image sequence. In *ISCAS'98. Proceedings of the 1998 IEEE International Symposium on Circuits and Systems (Cat. No. 98CH36187)*, volume 4, pages 241–244. IEEE, 1998.

[86] Marko Heikkila and Matti Pietikainen. A texture-based method for modeling the background and detecting moving objects. *IEEE transactions on pattern analysis and machine intelligence*, 28(4) :657–662, 2006.

[87] Maha M Azab, Howida A Shedeed, and Ashraf S Hussein. A new technique for background modeling and subtraction for motion detection in real-time videos. In *2010 IEEE International Conference on Image Processing*, pages 3453–3456. IEEE, 2010.

[88] Eric Hayman and Jan-Olof Eklundh. Statistical background subtraction for a mobile observer. In *null*, page 67. IEEE, 2003.

[89] Mohand Said Allili, Nizar Bouguila, and Djemel Ziou. Finite generalized gaussian mixture modeling and applications to image and video foreground segmentation. In *Fourth Canadian Conference on Computer and Robot Vision (CRV'07)*, pages 183–190. IEEE, 2007.

[90] Andrew B Godbehere, Akihiro Matsukawa, and Ken Goldberg. Visual tracking of human visitors under variable-lighting conditions for a responsive audio art installation. In *2012 American Control Conference (ACC)*, pages 4305–4312. IEEE, 2012.

[91] Ahmed Elgammal, David Harwood, and Larry Davis. Non-parametric model for background subtraction. In *European conference on computer vision*, pages 751–767. Springer, 2000.

[92] Alexander Steudel and Manfred Glesner. Fuzzy segmented image coding using orthonormal bases and derivative chain coding. *Pattern Recognition*, 32(11) :1827–1841, 1999.

[93] Thuy Xuan Pham, Patrick Siarry, and Hamouche Oulhadj. Integrating fuzzy entropy clustering with an improved pso for mri brain image segmentation. *Applied Soft Computing*, 65 :230–242, 2018.

[94] Yuhui Zheng, Byeungwoo Jeon, Danhua Xu, QM Wu, and Hui Zhang. Image segmentation by generalized hierarchical fuzzy c-means algorithm. *Journal of Intelligent & Fuzzy Systems*, 28(2) :961–973, 2015.

[95] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015.

[96] Yunzhi Wang, YU Haichao, Dashan Gao, and Jiao Wang. Image segmentation and object detection using fully convolutional neural network, May 28 2019. US Patent App. 10/304,193.

[97] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[98] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv :1409.1556*, 2014.

[99] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015.

[100] Jeff Donahue, Yangqing Jia, Oriol Vinyals, Judy Hoffman, Ning Zhang, Eric Tzeng, and Trevor Darrell. Decaf : A deep convolutional activation feature for generic visual recognition. In *International conference on machine learning*, pages 647–655, 2014.

[101] Massimo Piccardi. Background subtraction techniques : a review. In *2004 IEEE International Conference on Systems, Man and Cybernetics (IEEE Cat. No. 04CH37583)*, volume 4, pages 3099–3104. IEEE, 2004.

[102] Andrews Sobral and Antoine Vacavant. A comprehensive review of background subtraction algorithms evaluated with synthetic and real videos. *Computer Vision and Image Understanding*, 122 :4–21, 2014.

[103] Ismail Haritaoglu, David Harwood, and Larry S. Davis. W/sup 4 : real-time surveillance of people and their activities. *IEEE Transactions on pattern analysis and machine intelligence*, 22(8) :809–830, 2000.

[104] Jiale Yin, Lei Liu, He Li, and Qiankun Liu. The infrared moving object detection and security detection related algorithms based on w4 and frame difference. *Infrared Physics & Technology*, 77 :302–315, 2016.

[105] Pierre-Luc St-Charles and Guillaume-Alexandre Bilodeau. Improving background subtraction using local binary similarity patterns. In *IEEE Winter Conference on Applications of Computer Vision*, pages 509–515. IEEE, 2014.

[106] Martin Hofmann, Philipp Tiefenbacher, and Gerhard Rigoll. Background segmentation with feedback : The pixel-based adaptive segmenter. In *2012 IEEE computer society conference on computer vision and pattern recognition workshops*, pages 38–43. IEEE, 2012.

[107] Lionel Lacassagne, Antoine Manzanera, and Antoine Dupret. Motion detection : Fast and robust algorithms for embedded systems. In *2009 16th IEEE International Conference on Image Processing (ICIP)*, pages 3265–3268. IEEE, 2009.

[108] Antoine Manzanera. σ-δ background subtraction and the zipf law. In *Iberoamerican Congress on Pattern Recognition*, pages 42–51. Springer, 2007.

[109] Lionel Lacassagne, Antoine Manzanera, Julien Denoulet, and Alain Mérigot. High performance motion detection : some trends toward new embedded architectures for vision systems. *Journal of Real-Time Image Processing*, 4(2) :127–146, 2009.

[110] Aaron Bobick and James Davis. An appearance-based representation of action. In *Proceedings of 13th International Conference on Pattern Recognition*, volume 1, pages 307–312. IEEE, 1996.

[111] Md Atiqur Rahman Ahad, Joo Kooi Tan, Hyoungseop Kim, and Seiji Ishikawa. Motion history image : its variants and applications. *Machine Vision and Applications*, 23(2) :255–281, 2012.

[112] Md Atiqur Rahman Ahad, Joo Kooi Tan, HS Kim, and Seiji Ishikawa. Temporal motion recognition and segmentation approach. *International Journal of Imaging Systems and Technology*, 19(2) :91–99, 2009.

[113] Mohiuddin Ahmad and Seong-Whan Lee. Recognizing human actions based on silhouette energy image and global motion description. In *2008 8th IEEE International Conference on Automatic Face & Gesture Recognition*, pages 1–6. IEEE, 2008.

[114] Gary R Bradski and James W Davis. Motion segmentation and pose recognition with motion history gradients. *Machine Vision and Applications*, 13(3) :174–184, 2002.

[115] Xiaotao Zou and Bir Bhanu. Human activity classification based on gait energy image and coevolutionary genetic programming. In *18th International Conference on Pattern Recognition (ICPR'06)*, volume 3, pages 556–559. IEEE, 2006.

[116] Dengsheng Zhang and Guojun Lu. Review of shape representation and description techniques. *Pattern recognition*, 37(1) :1–19, 2004.

[117] Ming-Kuei Hu. Visual pattern recognition by moment invariants. *IRE transactions on information theory*, 8(2) :179–187, 1962.

[118] Hussin K Ragb and Vijayan K Asari. Color and local phase based descriptor for human detection. In *Aerospace and Electronics Conference (NAECON) and Ohio Innovation Summit (OIS)*, pages 68–73, 2016.

[119] M Concetta Morrone and Robyn A Owens. Feature detection from local energy. *Pattern recognition letters*, 6(5) :303–313, 1987.

[120] Michel Coster and Jean-Louis Chermant. Précis d'analyse d'images. Technical report, Presses du CNRS, 1989.

[121] Dieudonne Fabrice Atrevi, Damien Vivet, and Bruno Emile. Bayesian generative model based on color histogram of oriented phase and histogram of oriented optical flow for rare event detection in crowded scenes. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3126–3130. IEEE, 2018.

[122] Hector Franco-Lopez, Alan R Ek, and Marvin E Bauer. Estimation and mapping of forest stand density, volume, and cover type using the k-nearest neighbors method. *Remote sensing of Environment*, 77(3) :251–274, 2001.

[123] Richard O Duda, Peter E Hart, and David G Stork. *Pattern classification*. John Wiley & Sons, 1973.

[124] Yaman Akbulut, Abdulkadir Sengur, Yanhui Guo, and Florentin Smarandache. Ns-k-nn : Neutrosophic set-based k-nearest neighbors classifier. *Symmetry*, 9(9) :179, 2017.

[125] Md Mohibullah, Md Zakir Hossain, and Mahmudul Hasan. Comparison of euclidean distance function and manhattan distance function using k-mediods. *International Journal of Computer Science and Information Security*, 13(10) :61, 2015.

[126] Richard O. Duda, Peter E. Hart, et al. *Pattern Classification And Scene Analysis*, volume 3. John Wiley & Sons, New York, 1973.

[127] Sebastian Raschka. Naive Bayes and Text Classification I : Introduction and Theory, 2014. arXiv preprint arXiv :1410.5329 [cs.LG].

[128] Tom M Mitchell, Rebecca Hutchinson, Radu S Niculescu, Francisco Pereira, Xuerui Wang, Marcel Just, and Sharlene Newman. Learning to decode cognitive states from brain images. *Machine learning*, 57(1-2) :145–175, 2004.

[129] Christopher M Bishop. *Pattern recognition and machine learning*. springer, 2006.

[130] Thomas G Dietterich. Approximate statistical tests for comparing supervised classification learning algorithms. *Neural computation*, 10(7) :1895–1923, 1998.

[131] Evelyn Fix and Joseph L Hodges Jr. Discriminatory analysis-nonparametric discrimination : consistency properties. Technical report, California Univ Berkeley, 1951.

[132] G Shakhnarovich, T Darrell, and P Indyk. Nearest-neighbor methods in learning and vision : Theory and practice, chapter 3, 2006.

[133] Jason Van Hulse, Taghi M Khoshgoftaar, and Amri Napolitano. A comparative evaluation of feature ranking methods for high dimensional bioinformatics data. In *2011 IEEE International Conference on Information Reuse & Integration*, pages 315–320. IEEE, 2011.

[134] Mark A Hall. Correlation-based feature subset selection for machine learning. *Thesis submitted in partial fulfillment of the requirements of the degree of Doctor of Philosophy at the University of Waikato*, 1998.

[135] Thomas Hofmann. Probabilistic latent semantic analysis. In *Proceedings of the Fifteenth conference on Uncertainty in artificial intelligence*, pages 289–296, 1999.

[136] Hanchuan Peng, Fuhui Long, and Chris Ding. Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Transactions on pattern analysis and machine intelligence*, 27(8) :1226–1238, 2005.

[137] Benjamin Auffarth, Maite López, and Jesús Cerquides. Comparison of redundancy and relevance measures for feature selection in tissue classification of ct images. In *Industrial Conference on Data Mining*, pages 248–262, 2010.

[138] Ekaterina H Spriggs, Fernando De La Torre, and Martial Hebert. Temporal segmentation and activity classification from first-person sensing. In *2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, pages 17–24. IEEE, 2009.

[139] Georgios D Evangelidis, Gurkirt Singh, and Radu Horaud. Continuous gesture recognition from articulated poses. In *European Conference on Computer Vision*, pages 595–607. Springer, 2014.

[140] Javier Ortiz Laguna, Angel García Olaya, and Daniel Borrajo. A dynamic sliding window approach for activity recognition. In *International Conference on User Modeling, Adaptation, and Personalization*, pages 219–230. Springer, 2011.

[141] Sergio Escalera, Xavier Baró, Jordi Gonzalez, Miguel A Bautista, Meysam Madadi, Miguel Reyes, Víctor Ponce-López, Hugo J Escalante, Jamie Shotton, and Isabelle Guyon. Chalearn looking at people challenge 2014 : Dataset and results. In *European Conference on Computer Vision*, pages 459–473. Springer, 2014.

[142] Natalia Neverova, Christian Wolf, Graham W Taylor, and Florian Nebout. Multi-scale deep learning for gesture detection and localization. In *European Conference on Computer Vision*, pages 474–490. Springer, 2014.

[143] Bassem Seddik, Sami Gazzah, Thierry Chateau, and Najoua Essoukri Ben Amara. Augmented skeletal joints for temporal segmentation of sign language actions. In *International Image Processing, Applications and Systems Conference*, pages 1–6. IEEE, 2014.

Félix-Bazin POLLA DE NDJAMPA

Reconnaissance d'action à l'aide d'un imageur infrarouge basse résolution

Résumé :
L'identification des actions simples de la vie quotidienne (comme marcher, s'asseoir ...) à partir des vidéos est important pour un système de reconnaissance d'activités humaines dont le but est l'amélioration de la sécurité et le confort des personnes. Cette thèse étudie le problème de la reconnaissance d'actions humaines en utilisant des images infrarouges basse résolution qui garantissent l'intimité des personnes surveillées. L'objectif principale de cette thèse est de tester et proposer des approches algorithmiques pouvant exploiter un type d'images assez particulières pour la reconnaissance d'actions dans une pièce. Notre première contribution concerne la proposition d'une chaîne d'analyse d'images thermiques adaptée au capteur utilisé. Pour ce faire, une étude de quelques méthodes de filtrage est faite dans le but d'éliminer les bruits des images en conservant le maximum d'informations, puis une méthode de segmentation à seuil adaptatif est proposée. Une méthode de post-traitement est également proposée pour combler les vides et éliminer les bruits dans les résultats de segmentation. Enfin, une étude comparative entre les méthodes de filtrage et les méthodes de segmentation est faite dans le but de trouver le couplage (filtrage/segmentation) permettant de mieux extraire la silhouette de la personne dans les images. La seconde contribution est la proposition d'un modèle de reconnaissance d'actions qui représente une séquence vidéo en utilisant les distances entre les centres de gravité des formes segmentées, puis nous extrayons de cette représentation des attributs statistiques avant de les passer au classifier KNN. Une autre étude est faite en proposant plusieurs modèles de reconnaissance d'actions utilisant comme représentation de séquence vidéo le MHI, puis des descripteurs pertinents sont extraits et combinés (les moments de HU, CHOP et les descripteurs géométriques) puis passé au classifieur. De ces études sont nées notre troisième contribution qui est la proposition d'un modèle en cascade comportant 3 classifieurs, nous permettant d'atteindre 89% de F-score sur la base que nous avons conçu pour cette étude. Pour valider nos résultats, une étude comparative est faite entre les approches d'apprentissages classiques et les approches d'apprentissages profonds. Nous terminons nos travaux en présentant la stratégie mise en place pour un système de reconnaissances d'actions dans des vidéos contenant plusieurs actions.
Mots clés : reconnaissance d'actions humaines, extraction de caractéristiques, filtrage, image historique de mouvement (MHI), apprentissage automatique et profond.

Action recognition using a low-resolution infrared

Abstract :
The identification of actions of daily life (such as walking, sitting ...) from videos is important for a system of recognition of human activities whose goal is to improve the safety and comfort of people. This thesis studies the problem of the recognition of human actions using low-resolution infrared images that guarantee the privacy of the persons observed. The main objective of this thesis is to test and propose algorithmic approaches that can exploit a rather particular type of images for the recognition of actions in a piece. Our first contribution concerns the proposal of a thermal image analysis chain adapted to the sensor used. In order to do so, we first carry out a study of some filtering methods in order to eliminate noise from the images while keeping the maximum of information. Then a segmentation method with adaptive threshold is proposed. A post-processing method to fill the holes and eliminate noise in the segmentation results is also proposed. Finally, a comparative study between the filtering methods and the segmentation methods is made in order to find the coupling (filtering/segmentation) allowing to better extract the silhouette of the person in the images. The second contribution is the proposal of an action recognition model that represents a video sequence using the distances between the centers of gravity of the segmented shapes, then we extract statistical attributes from this representation. Finally, those attribute are used as the inputs of a KNN classifier. Another study is carried out by proposing several action recognition models using the MHI as a representation of a video sequence, then relevant descriptors are extracted and combined (HU moment, CHOP and geometric descriptors) and passed to the classifier. These studies inspired our third contribution which is a proposal of a cascade model with 3 classifiers, allowing us to reach 89% F-score on the dataset we develop for our thesis. To validate our results, we discuss in a comparative study the results we obtained using classical learning approaches and deep learning approaches. We conclude our work by presenting the strategy implemented for an action recognition system in videos containing several actions.
Keywords : human action recognition, feature extraction, filtering, motion history image (MHI), machine learning and deep learning.

Laboratoire PRISME, 2 Avenue François Mitterrand 36000 Châteauroux