

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Université des Sciences et de la Technologie Houari Boumediene
Faculté d'Electronique et d'Informatique



THÈSE

Présentée pour obtenir le grade de **Docteur en Sciences**

Filière : **Electronique**

Spécialité : **Traitement du Signal et des Images**

Par : Mr. **Nabil ZERROUKI**

Thème

**Localisation de personnes et identification de postures
dans une scène**

Soutenue publiquement, le 26/06/2018 devant le jury composé de :

Mme. Nouara ACHOUR
Mr. Amrane HOUACINE
Mme. Amina SERIR
Mme. Latifa HAMAMI
Mr. Mahmoud BELHOCINE
Mr. Nafaa Nacereddine

Professeur à l'USTHB
Professeur à l'USTHB
Professeur à l'USTHB
Professeur à l'ENP
Directeur de recherche au CDTA
Maitre de recherche A au CRTI

Présidente du jury
Directeur de Thèse
Examinatrice
Examinatrice
Examineur
Examineur

À la mémoire de ma très chère grand-mère

Remerciements

Mes plus profonds remerciements, toute ma gratitude et ma reconnaissance éternelle vont à mon directeur de thèse M Amrane HOUACINE, Professeur à l'USTHB. Merci à vous pour la confiance, le soutien permanent et la liberté nécessaire dont j'ai bénéficié de votre part durant ces travaux. Vos conseils avisés, votre disponibilité permanente à toutes mes sollicitations ont été un atout indéniable pour l'aboutissement de cette thèse. Merci encore pour votre encouragement continu et votre sens de l'humour sans lesquels je n'aurai jamais réussi à aller jusqu'au bout. Ce fut un plaisir et un grand honneur de travailler sous votre direction.

Je remercie vivement l'ensemble des membres du jury pour le temps qu'ils ont consacré à la lecture de mon document. Je remercie tout d'abord Mme Noura Achour, Professeur à l'USTHB, pour avoir accepté de présider ce jury de thèse. Je remercie également Mme Latifa HAMAMI, Professeur à l'ENP, Mme Amina SERIR, Professeur à l'USTHB, M Mahmoud BELHOCINE, Directeur de recherche au CDTA et M Nafaa NACEREDDINE Maître de recherche au CRTI pour avoir accepté d'être examinateurs de ma thèse. J'éprouve un profond respect pour vos travaux et parcours.

Un grand merci aussi à M Fouzi Harrou avec qui j'ai collaboré et qui a le don de présenter les mathématiques avec simplicité et clarté.

Un grand merci à tous les membres du Laboratoire Communication Parlée et Traitement de Signal (LCPTS), ainsi qu'à l'ensemble du personnel de la Faculté d'Électronique et d'Informatique de l'USTHB (enseignants, chercheurs, administrateurs, secrétaires) que j'ai pu côtoyer durant ces années passées pour leur bonne humeur et leurs compétences. Je ne saurais manquer non plus de remercier M Kamal Bouhadeb pour sa disponibilité, son sérieux et ses qualités au sein de l'administration.

Je voudrais aussi remercier mes collègues du Centre de Développement des Technologies Avancées (CDTA) que j'ai eu le privilège de côtoyer tout au long de mon doctorat pour les échanges passionnants et enrichissants dont j'ai pu bénéficier, l'atmosphère scientifique fertile que chacun entretient et leur bonne humeur invariable.

Enfin, je me dois de rendre hommage à toute la communauté scientifique pour les cours et les tutorats en ligne, ainsi qu'aux fondations des logiciels libres et open-source à qui nous devons tous beaucoup.

Afin d'être reconnaissant envers ceux qui m'ont appuyé et encouragé à effectuer ce travail de recherche, je dédie cette thèse:

À ma très chère mère et mon très cher père qui ont toujours été là pour moi et qui n'ont cessé de me combler par leur amour et leur tendresse!

À mon très cher frère Yacine et ma très chère sœur Hayet pour tous les sentiments d'affection et d'amour!

À mes très chers oncles et tantes pour tous leurs encouragements et leur foi en moi!

À ma très chère tante, Fawzia, pour son admirable soutien et sa constante affection!

À ma très chère nièce, Syrine, qui nous comble de joie!

À mes très chers cousins, Ryad, Younes et Chahinez pour leurs encouragements!

À mon très cher ami, Samir ABBA pour ses encouragements et son sens de l'humour!

À tous les membres de ma famille sans aucune exception. Et à tous ceux et celles pour qui ma réussite tient à coeur ...

Résumé

De nos jours, l'analyse de scène et plus particulièrement la reconnaissance d'activités de personnes, en utilisant les données de divers capteurs, représente un domaine de recherches très intéressant. D'autre part, la détection de chute est devenue une préoccupation majeure dans le domaine de santé, en particulier pour les personnes âgées. Il existe une demande de plus en plus croissante de la part des établissements de santé, (hôpitaux, maisons de retraites, etc.), sur les systèmes permettant la reconnaissance gestuelle et surtout la détection rapide d'anomalies ou de problèmes chez les patients.

Le but de ce travail est de concevoir un système de détection de chute basée sur les images des caméras de surveillance et en utilisant les algorithmes de machines d'apprentissages. Un système de détection de chutes devrait être robuste et rapide afin qu'il soit adapté aux conditions réelles. Ces deux critères essentiels sont pris en compte durant toutes les phases de traitement à savoir : la segmentation du corps humain dans l'image, l'extraction des caractéristiques, l'identification de posture et la détection de chutes.

Dans ce travail, nous avons en premier lieu mené plusieurs recherches sur les systèmes de détection de chutes proposés dans la littérature. Inspiré par certaines limitations rencontrées auparavant, les motivations principales de ce travail étaient de trouver une représentation adéquate et robuste à la silhouette humaine et d'exploiter les techniques de machine d'apprentissage (machine learning techniques) en considérant l'information du temps dans la détection des chutes.

Plusieurs attributs sont ainsi proposés pour caractériser la posture, à savoir la décomposition en valeurs singulières tronquée, l'utilisation des coefficients de curvelets et les rapports de surfaces de la silhouette humaine. L'avantage de ces caractéristiques est leur invariance par rapport à la translation et au changement d'échelle, cela fait qu'aucun calibrage de caméra préalable n'est nécessaire. Durant la phase de reconnaissance de posture, plusieurs techniques de machines d'apprentissage sont utilisées, telles que adaboost, les réseaux de neurones et les Séparateur à Vaste Marge. Les modèles de Markov cachés sont utilisés pour la détection de chutes ou les classes de postures sont définies comme des états du modèle de Markov ce qui a donné un sens physique aux états. Ce formalisme de détection de chute est aussi combiné sur des données multi-caméras, en utilisant la théorie de Dempster-Shafer. Enfin, les méthodes statistiques basées sur les exponentielles pondérées à moyennes mobiles sont appliquées sur les données de capteurs (accéléromètres) afin d'améliorer la prise de décision basée sur les images de caméras.

Abstract

Nowadays, camera based scene analysis, and more specifically people's behavior understanding using various sensor data has received an important consideration. On the other hand, fall detection has become a major concern in the field of health, especially for elderly. Where there is a growing request from health institutes (hospitals, retirement homes, etc.) on gesture recognition systems, and especially systems allowing detection of abnormalities or problems among patients.

The purpose of this work is to design a fall detection system based on video monitoring and machine learning algorithms. A fall detection system should be robust and fast so that it is adapted to real conditions. These two essential criteria are taken into account during all phases of treatment, namely: human body segmentation, feature extraction, posture identification and fall detection. In this work, we first conducted several researches on fall detection systems proposed in the literature. Inspired by certain limitations previously encountered, the main motivations of this work were to find an adequate and robust representation to the human silhouette and to exploit machine learning techniques by considering time information in fall detection.

Several attributes are thus proposed to characterize the posture, namely the truncated singular value decomposition, curvelet coefficients and human body area ratios. The advantage of these characteristics is their invariance with respect to translation and scaling, so that no prior camera calibration is needed. The next phase consists to recognize different postures according to their features. This classification is called static, since posture identification is based on each image separately (frame-by-frame classification). Several machine learning methods are used, such as adaboost, neural network, and support vector machine. The last step consists of detecting falls. This detection is also called dynamic classification, since the temporal or sequential aspect is considered. In this work, hidden Markov models (HMM) are used for the distinction between daily activities and falls, while posture classes are defined as Markov model states to provide states a physical meaning. The detection formalism is then combined with multi-view data using the Dempster-Shafer theory. Sensor data (accelerometers) are also exploited in a second approach, by the application of statistical methods based on exponentially weighted moving average.

Table des Matières

Remerciements	ii
Résumé	iv
Abstract	vi
Table des Matières	vii
Liste des Figures	x
Liste des Tableaux	xii
Liste des Acronymes	xiii
1. Chapitre 1 : Introduction et objectifs du travail	1
1.1. Introduction	1
1.2. Problématique	4
1.3. Motivation	7
1.3.1. Limitations des méthodes de détection basées sur l'approche capteur	8
1.3.2. Limitations des méthodes de détection basées sur l'approche vidéosurveillance	8
1.4. Structure de la thèse	9
1.5. Conclusion	10
2. Chapitre 2 : Modélisation de l'environnement et segmentation de personne (prétraitement)	12
2.1. Aperçu sur les méthodes de segmentations proposées dans l'état de l'art	12
2.1.1. Segmentation basée sur le flux optique	13
2.1.2. Segmentation par la différence temporelle	13
2.1.3. Segmentation par soustraction de l'arrière-plan	14
2.1.4. Segmentation par le modèle de mixtures de gaussiennes	15
2.2. Description de la méthode de segmentation utilisée (par la combinaison de la soustraction d'arrière-plan et la différence temporelle)	16
2.3. Prétraitement et élimination de bruits en utilisant les opérateurs morphologiques	17
2.3.1. Morphologie mathématique	17
2.4. Conclusion	20
3. Chapitre 3 : Extraction des caractéristiques du corps humain	21
3.1. Aperçu sur les méthodes existantes de caractérisation du corps humain	21
3.1.1. Caractérisation par Points	22
3.1.2. Caractérisation par formes géométriques primitives	23

3.1.3. Représentation par les contours	24
3.1.4. Représentation par les modèles de formes articulées	24
3.1.5. Représentation par les modèles squelettiques	25
3.1.6. Représentation par densités de probabilité	26
3.1.7. Représentation par modèles (template)	26
3.2. Utilisation de la décomposition en valeurs singulières tronquées	27
3.3. Utilisation des coefficients de curvelets comme caractéristiques	30
3.4. Proposition des rapports de surface comme caractéristiques	32
3.5. Réduction de dimension en utilisant l'algorithme de l'évolution différentielle (differential evolution algorithm)	35
3.6. Conclusion	37
4. Chapitre 4 : Classification de postures humaines	38
4.1. Classification statique : image par image	39
4.2. Classification de postures en utilisant les réseaux de neurones	39
4.3. Classification de postures en utilisant les Séparateurs à Vaste Marge (Support Vecteur Machines SVM)	43
4.3.1. Utilisation des SVMs pour des données linéairement séparables	43
4.3.2. Utilisation des SVMs pour des données non-linéairement séparables	45
4.3.3. SVM multi-classes en utilisant l'approche "un contre tous"	47
4.3.4. SVM multi-classes en utilisant l'approche "un contre un"	47
4.4. Identification de postures en utilisant la classification Adaboost	47
4.5. Conclusion	50
5. Chapitre 5 : Classification d'activités humaines	52
5.1. Classification dynamique : séquence vidéo	52
5.2. Classification de chutes en utilisant les modèles de Markov cachés	54
5.2.1. Etapes de construction du modèle HMM	55
5.2.2. Instanciation du classifieur HMM dans la détection de chutes humaines	56
5.3. Combinaison de données multi-cameras en utilisant Dempster-Shafer	59
5.4. Détection de chute en utilisant les exponentielles pondérées à moyennes mobiles (EWMA) appliquée aux données capteurs	62
5.5. Conclusion	68
6. Chapitre 6 : Etude expérimentale	69
6.1. Description des données	70
6.1.1. Description de la base de données UR Fall Detection Dataset (URFD)	70

6.1.2. Description de la base de données Fall Detection Database (FDD)	71
6.2. Fonctions : pertes et risques (Losses and Risks)	72
6.3. Critères d'évaluation	74
6.4. Résultats et discussion	77
6.4.1. Evaluation de l'utilisation des valeurs singulières tronquées comme caractéristiques de la silhouette humaine	78
6.4.2. Evaluation de l'utilisation des coefficients de Curvelets et les rapports de surfaces comme caractéristiques de la silhouette humaine	79
6.4.3. Evaluation de la classification de postures humaines	83
6.4.4. Evaluation de la détection de chutes humaines	85
6.4.5. Comparaisons des méthodes proposées avec des systèmes de détection existants	91
6.5. Conclusion	94
CONCLUSION GÉNÉRALE ET PERSPECTIVES	95
Références bibliographiques	98
Travaux scientifiques	103
• Publications scientifiques	103
• Communications scientifiques	104

Liste des Figures

1.1. Exemples de caméras de surveillance vidéo et un aperçu d'écran de visualisation du système de surveillance.	2
1.2. Exemples d'interaction homme-machine en utilisant la camera Kinect (image de gauche) et une image de la camera Kinect (image de droite).	3
1.3. Exemples de systèmes de caméras surveillance installés dans des hôpitaux et des maisons de retraites.	4
2.1 Exemple illustrant les deux opérateurs morphologiques : l'érosion (b) et la dilatation (c)	19
2.2. Exemple d'élimination de bruit en utilisant les opérateurs morphologiques [Nguyen, 2015].	19
3.1. Les différentes représentations de silhouette [Nguyen, 2015].	22
3.2. Exemples de représentations de silhouette en utilisant les formes géométriques (a) en position assise (b) en position accroupi, (c) en position penchée.	23
3.3. Exemples de représentations de silhouette en utilisant les contours présentant une confusion d'objets avec la silhouette humaine : (a) confusion avec la raquette (b) confusion avec la canne.	24
3.4. Exemples de représentations de silhouette en utilisant les formes articulées [Yang, 2013]	25
3.5 La variation entre le rapport d'énergie des deux images en fonction du rang tronqué r' .	29
3.6 La variation entre l'erreur quadratique des deux images en fonction du rang tronqué r' .	29
3.7 Comparaison entre les deux images : segmentée (a) et son approximation (b) en utilisant un rang tronqué $r' = 30$.	30
3.8 Exemple de décomposition en curvelet d'une image originale (image binaire représentant le corps humain dans la position debout) en une image d'approximation suivie par les images de détails dans les 8 directions [Zerrouki, 2017 (1)].	31
3.9 Partitionnement du corps humain en cinq zones d'occupation partielles	33
3.10 Exemple d'utilisation des rapports de surface comme caractéristiques de postures	34
3.11 Technique de sélection de caractéristiques en utilisant l'algorithme d'évolution différentielle.	37
4.1 Exemple d'architecture globale d'un réseau de neurones	40
4.2 Structure générale du perceptron	41
4.3 Schéma de l'algorithme de définition des hyperplans de séparation.	44
4.4 Schéma de l'algorithme adaboost en utilisant les arbres de décision.	47
4.5 Exemple de classification AdaBoost	49

5.1 Schéma de construction du vecteur caractéristique de la séquence vidéo par la concaténation des caractéristiques des frames.	53
5.2 Exemple de séquence d'images avec les états physique (classes de postures) du modèle HMM	57
5.3 Schéma de l'approche du système de détection de chute basée sur la combinaison Dempster-Shafer de multiple classifieurs.	60
5.4 Schéma de l'approche du système de détection de chute basée sur la combinaison Dempster-Shafer de multiple classifieurs.	63
5.5. Exemple sur les composantes d'accélération résultantes (rangée du haut) et leurs composantes d'accélération triaxiale correspondantes (rangée du bas) décrivant différents gestes : ramassant un objet (colonne de gauche), s'asseoir puis se lever (colonne de milieu) et tomber (colonne de droite).	64
5.6. L'organigramme de l'algorithme de détection de chute basé sur EWMA	65
5.7. L'organigramme de détection de chute basé sur le formalisme EWMA-SVM	67
6.1 Exemples de frames de la base de données URFD contenant des conditions d'éclairage variables,	70
6.2 Exemples de frames de la base de données FFD dans des différents environnements	71
6.3 Exemples de variation des rapports de surfaces pour deux classes de postures (debout et assise).	82
6.4 Evolution en fonction du temps des rapports de surface pour des activités quotidiennes (a) et pour une chute.	82
6.5 Exemples de mal-classification de postures et de chutes causées par la visualisation partielle du corps.	87
6.6 Exemples de séquence de chute en utilisant deux caméras.	88
6.7 Exemples de séquence de chute non-détectée à cause des conditions d'éclairage.	90

Liste des Tableaux

6.1. Evaluation de l'utilisation des valeurs singulières comme caractéristiques de la silhouette humaine [Zerrouki, 2014 (2)].	78
6.2. Comparaison de l'efficacité des valeurs singulières dans la caractérisation de la silhouette humaine [Zerrouki, 2014 (2)] avec des caractéristiques proposées dans la littérature.	79
6.3. Evaluation de l'utilisation des coefficients de curvelets pour caractériser la silhouette humaine [Zerrouki, 2014 (1)].	80
6.4. Evaluation de l'utilisation des coefficients de curvelets et les rapports de surfaces pour caractériser la silhouette humaine [Zerrouki, 2014 (1)].	81
6.5. Comparaison des performances entre les différents algorithmes de classification de posture [Zerrouki, 2016 (1)], [Zerrouki, 2017 (2)].	84
6.6. Comparaison des performances entre les différents algorithmes de détection de chutes humaines appliquée sur les bases de données FDD et URFD [Zerrouki, 2017 (1)].	85
6.7. Comparaison des méthodes de détection de chutes en utilisant les caméras séparément et leur combinaison avec DSTE.	88
6.8. Comparaison de l'approche proposée avec des méthodes de combinaisons.	89
6.9. Comparaison des performances de EWMA-SVM avec les différents algorithmes de détection [Zerrouki, 2016 (1)].	90
6.10. Comparaison des performances du système de détection de chute proposé appliqué sur la base de données FDD.	92
6.11. Comparaison des performances du système de détection de chute proposé appliqué sur la base de données URFD.	93

Liste des Acronymes

ANN	Artificial Neural Networks
AUC	Area Under Curve
CCTV	Closed Circuit Television
DE	Defferential Evolution
DNN	Deep Neural Network
DP	Decision Profile
DSTE	Dempster-Shafer Theory of Evidence
DT	Decision Template
EPF	European Profound Project
EWMA	Exponentially Weighted Moving Average
GMM	Gaussian Mixture Model
HMM	Hidden Markov Model
ICA	Independent Component Analysis
KNN	K Nearest Neighbor
MLP	Multi Layer Perceptron
MM	Morphologie Mathématique
NB	Naïve Bayes
OAA	One Against All
OAo	One Against One
PCA	Principle Component Analysis
PMC	Perceptron Multi-Couches
RBF	Radial Basis Function
ROC	Receiver Operation Characteristic
SIFT	Scale Invariant Feature Transform
SVD	Singular Value Decomposition
SVM	Support Vector Machines

Chapitre 1 :

Introduction et Objectifs du travail

1.1. Introduction	1
1.2. Problématique	4
1.3. Motivation	7
1.3.1. Limitations des méthodes de détection basées sur l'approche capteur	8
1.3.2. Limitations des méthodes de détection basées sur l'approche vidéosurveillance	8
1.4. Structure de la thèse	9
1.5. Conclusion	10

Dans ce chapitre, nous introduisons le contexte du sujet de cette thèse de doctorat, à savoir **la localisation de personnes et l'identification de postures dans une scène** (Section 1.1). Nous définissons tout d'abord la problématique de cet axe de recherche en citant quelques limitations et difficultés rencontrées par la communauté scientifique (section 1.2). Nous présentons la motivation essentielle de notre travail (section 1.3), qui sera utilisée tout au long de cette thèse. Nous présentons par la suite les principales contributions apportées à ce sujet (section 1.4). Finalement, nous clôturons ce chapitre par la présentation de la structure du document de thèse (section 1.5).

1.1. Introduction

Avec le développement actuel des différents capteurs électroniques, des systèmes visuels et des technologies de communication, la surveillance et la reconnaissance des activités humaines est devenu un thème de recherche important visant l'analyse et la compréhension du comportement humain. Surtout que les outils électronique (équipements, capteurs, cameras, etc.) sont de plus en plus puissant et disponible à des prix accessibles ou même modérés. Cette disponibilité et ce développement ont rendu les systèmes automatiques de suivi et de reconnaissance de comportement humain beaucoup plus performants et rapides. En effet, la reconnaissance de l'action humaine est très utile et appliquée dans de nombreux domaines tels que la sécurité et la vidéo surveillance, la recherche vidéo, l'interaction homme-machine et enfin la santé et la vigilance des patients [ZAIDENBERG, 2014].

• Sécurité et vidéo surveillance

De nos jours, les systèmes de surveillance vidéo reçoivent de plus en plus d'attention, surtout que leur utilisation est devenue courante et largement acceptée par la société.

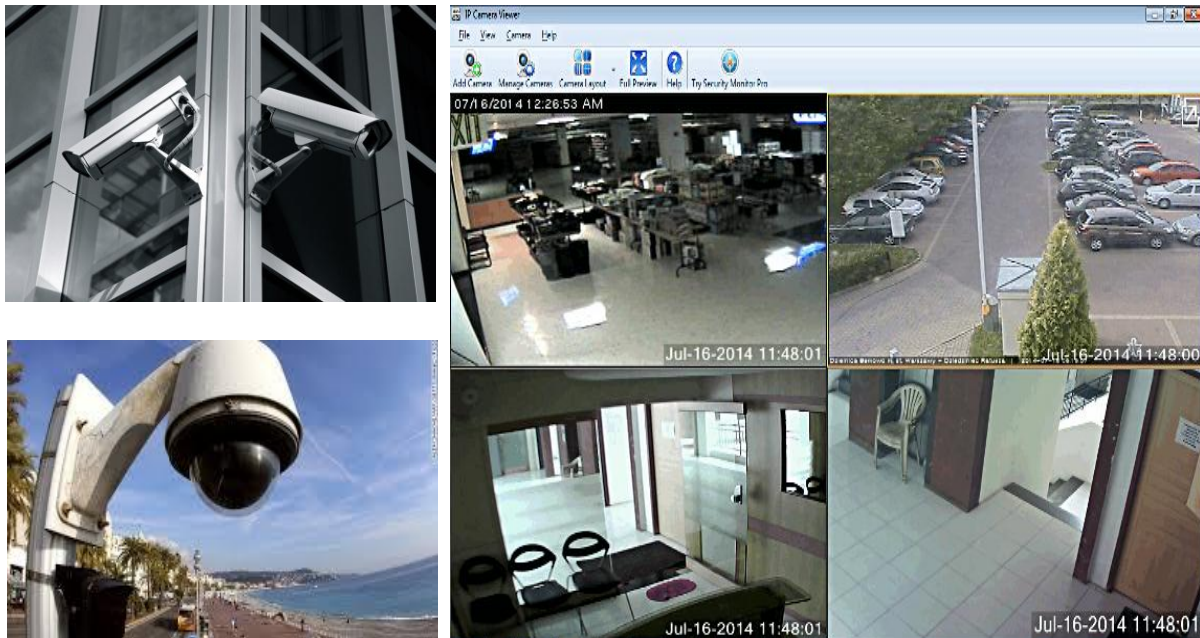


Figure.1.1. Exemples de caméras de surveillance vidéo et un aperçu d'écran de visualisation du système de surveillance [Deskshare, 2017].

Les systèmes de vidéosurveillance peuvent être trouvés dans différents lieux, tel que: les aéroports, les métros, les gares, les banques, les parkings, les magasins, etc. Ces systèmes de surveillance sont généralement équipés d'algorithmes d'exploitation permettant une gestion et un traitement automatiques des données telles que détection d'intrusion, détection de comportements dangereux, etc. [ZAIDENBERG, 2014].

• La recherche vidéo

En tant que deuxième domaine principal d'application, la recherche vidéo représente une tâche importante dans l'analyse vidéo et la compréhension, en particulier avec la quantité énorme de données vidéo enregistrées en permanence.

En plus de la diversité des modes de stockage et d'enregistrement de données vidéo; qui peut être (i) hors ligne (offline) en utilisant différents périphériques de stockage (disque dur, carte mémoire, bande de stockage, etc.) ou (ii) enregistrement en ligne (online) connu sous le nom de stockage en nuage (cloud storage); rend encore la recherche vidéo extrêmement importante et nécessaire pour les systèmes de reconnaissance de comportement humain.

- **Interaction homme-machine**

En plus de la surveillance et la recherche vidéo ; la reconnaissance de l'action humaine est également utilisée pour l'interaction homme-ordinateur et l'ingénierie d'exécution. L'exemple le plus populaire de l'interaction homme machine est l'utilisation de la camera Microsoft Kinect, qui est maintenant disponible dans des millions de foyers à travers le monde.



Figure.1.2. Exemples d'interaction homme-machine en utilisant la camera Kinect (image de gauche) et une image de la camera Kinect (image de droite) [Ashley, 2015].

La camera Kinect est un dispositif d'identification d'action qui permet aux utilisateurs de contrôler et d'interagir via une interface utilisateur, en stimulant des gestes et même des déplacements (évidemment, en restant dans le champ de vision de la Kinect). Par conséquent, un des aspects les plus importants pour ce dispositif est la reconnaissance des gestes et des actions courtes, ou peu de cas de confusions sont enregistrés [ZAIDENBERG, 2014].

- **Santé et vigilance des patients**

Selon Global Age Watch 15 [Gawi, 2015], dans le monde, il y avait environ 810 millions de personnes âgées de 60 ans et plus suivant les statistiques réalisées en 2012. On prévoit que, dans moins de dix ans, ce chiffre atteindra 1 milliard voir 1,375 milliard durant l'année 2030, ce qui signifie que 16% de la population mondiale sera composée de personnes âgées de 60 ans et plus. Par conséquent, la reconnaissance des actions et des comportements humains devient de plus en plus importante et fréquemment utilisée dans le domaine de santé, en particulier dans le but de surveiller ou d'assister les personnes âgées. Cependant, il existe une demande croissante de la part des centres de santé, hôpitaux, maisons de retraites, etc., sur les systèmes permettant la reconnaissance gestuelle et surtout la détection rapide d'anomalies ou de problèmes de santé chez les patients [Gawi, 2015].

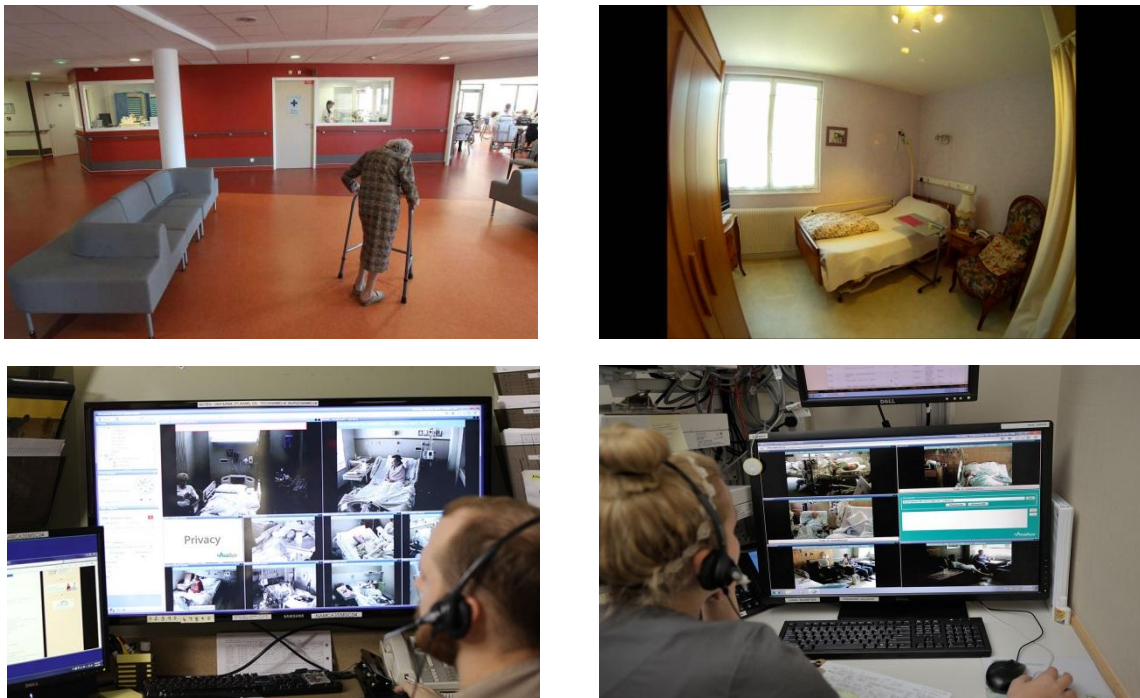


Figure.1.3. Exemples de systèmes de caméras surveillance installés dans des hôpitaux et des maisons de retraites [Ladde, 2017].

L'identification des activités quotidiennes du patient telles que : s'asseoir, marcher, faire de l'exercice ou dormir, permet aux scientifiques et spécialistes médicaux de mieux diagnostiquer et proposer des stratégies liées à leur hygiène de vie et leur adhésion aux prise des médicaments. En plus, un système de vidéosurveillance permet l'analyse du comportement de la personne (à l'état normal, sans présence d'anomalies), à savoir la prise des médicaments, les horaires de sommeil, etc. Cela est particulièrement important pour les personnes âgées, car ces systèmes les aident à conserver une part de leur autonomie [Ageing, 2008], [Gawi, 2015].

1.2. Problématique

Comme nous l'avons indiqué précédemment dans l'introduction (section 1.1), la reconnaissance gestuelle et d'activités humaines reçoit une attention croissante. Ces recherches sont surtout encouragées par le développement rapide de l'information, des technologies de la communication et des technologies intelligentes des systèmes de surveillance vidéo.

Dans ce travail nous nous sommes focalisés sur la détection de chutes humaines visant la vigilance sur les patients, les personnes âgées, etc. Le but est d'atténuer les problèmes de

santé causés par les chutes et éviter que la personne reste allongée sur le sol pendant une longue période de temps après la chute (plusieurs heures ou même plusieurs jours parfois). Plusieurs études ont défini les chutes comme la principale raison des hospitalisations des personnes âgées, handicapées, obèses et les personnes nécessitant une assistance en général [Ageing,2008], [Giannakouris, 2008]. Comme le montre les statistiques publiées par l'organisation mondiale de la santé (World Health Organization), 30% des personnes de plus de 65 ans tombent au moins une fois par an [Todd, 2004] et 47% de celles qui sont tombées ne peuvent plus se lever sans l'assistance d'autres personnes (ce qui implique généralement la perte d'autonomie) [Giannakouris, 2008], [Yu, 2012]. En outre, selon une étude réalisée par l'Eurostat, cette catégorie de population représentera 30% (en 2060) de la population globale au lieu de 17.1% (actuellement). Cela est reflété par une augmentation de 84,6 millions de personnes en 2008 à 151,1 millions de personnes en 2060 (ceci uniquement pour les pays de la communauté européenne) [Todd, 2004]. En outre, les effets sur la santé sont plus préjudiciables si la personne reste allongée pendant une longue période de temps après la chute, ces cas peuvent entraîner des préjudices graves et même la mort du patient [Yu, 2013]. Suivant l'étude américaine réalisée par Heinrich et autres, d'ici l'année 2020 les dépenses en soins médicaux causées par les chutes devraient augmenter de 43,8 milliards de dollars uniquement pour les États-Unis [Heinrich, 2010], [Soriano, 2007]. De ce fait, les systèmes fiables de détection et de classification des chutes amélioreront la qualité de vie, augmenteront les niveaux de sécurité et réduiront potentiellement les dépenses médicales.

La situation est identique en Algérie puisque la proportion de personnes âgées est en croissance permanente. Une étude réalisée en 2014 par le service de chirurgie orthopédique du CHU de Bab-El-Oued sur l'impact des accidents domestiques, causant annuellement des milliers de blessures et même d'handicaps. Selon Le prof. M. Yacoubi, chef de service de chirurgie orthopédique du CHU de Bab-El-Oued, 75% des causes des accidents domestiques sont les chutes dont 95% des victimes sont des personnes âgées de plus de 65 ans [Akkouche, 2014].

Afin de diminuer les risques de chute chez les personnes âgées, plusieurs réalisations sont proposées dans la littérature, en utilisant différent types de capteurs à savoir : capteurs intégrés au sol, Capteurs portés par la personne, ou la vidéo surveillance.

- **Capteurs intégrés à l'environnement**

Ces applications sont généralement liées aux projets d'environnements intelligents tels que la maison intelligente (Smart Home), ou l'hôpital intelligent (Smart Hospital), etc., où l'environnement est doté de différents types de capteurs : capteurs thermiques et infrarouges (pour la détection de présence des personnes), détecteurs de contact (pour le contrôle des accès: portes et fenêtres), etc. Néanmoins ce genre de capteurs ne permet pas la description ou l'identification des comportements humains. De ce fait pour la détection de chutes, l'environnement est doté de capteurs installés au sol afin de mesurer les vibrations (fournies par ces capteurs) qui sont proportionnelles à la chute. Plusieurs études ont adopté cette approche de capteurs au sol pour la détection de chute tel que : Alwan et al. [Alwan, 2006], ou Zigel et al. [Zigel, 2009] qui ont proposé leur combinaison avec des détecteurs de son et des microphones. Mais ces systèmes présentent encore un taux de d'erreur de classification considérable et ils ne sont pas suffisamment sensibles à certaines chutes comme les chutes lentes et les chutes à partir de position autre que debout (à partir des positions assise ou accroupi). En plus, ce type de systèmes dépend fortement de la nature du sol (dynamique différente pour un plancher en bois ou une moquette).

- **Capteurs portés par la personne**

Afin d'avoir des données plus précises pour identifier les différentes activités des personnes, plusieurs études ont proposé l'utilisation de capteurs portés par la personne. Il existe différents types de capteurs portables : capteurs de contrôle de paramètres physiologiques, boutons d'alerte et accéléromètres.

Les boutons d'alerte permettent à la personne âgée d'alerter une autre personne ou les services de secours en cas de problème. Cependant, ce type de capteur n'est pas souhaitable dans le cas des chutes, car une personne âgée perd généralement conscience après la chute ou elle est immobile d'où l'incapacité d'alerter les services de secours [Rougier, 2010]. Pour remédier à cela, les capteurs automatiques (ou détecteurs de chutes) sont plus utilisés par rapport à leur autonomie dans le fonctionnement ou aucune intervention humaine n'est nécessaire. Les détecteurs de chutes sont basés sur les accéléromètres et les gyroscopes. Dans le cas des accéléromètres, la donnée est issue de la magnitude et la direction de l'accélération; pour le cas des gyroscopes la donnée est basée sur l'orientation du corps. Cette technologie de capteurs portatifs est largement exploitée et même commercialisée par plusieurs compagnies de renommée mondiale, où l'actuel challenge est de pouvoir détecter une chute avant même l'impact au sol, ce qui permet le déclenchement de coussins gonflables de protection [Rougier, 2010].

Il est à noter que ces capteurs permettent de fournir une description de l'état et du comportement de la personne et peuvent aussi être utilisés pour le déclenchement d'une alarme. Cependant, le port des capteurs par les personnes âgées est parfois problématique, ou ils oublient et même trouvent l'inconvénient de les porter. De plus, ces capteurs nécessitent un entretien régulier tel que le changement de piles et le recalibrage [Cucchiara, 2005].

- **La vidéosurveillance**

Au cours des deux dernières décennies, beaucoup de travaux de recherche proposent des systèmes de détection de chutes basés sur la vidéosurveillance. Notamment plusieurs améliorations ont été apportées aux actuels systèmes de vidéosurveillance. Comparé aux systèmes de vidéosurveillance classiques connus sous le nom CCTV (Closed Circuit Television), les nouveaux systèmes de vidéosurveillance ne nécessitent plus le contrôle visuel des écrans assuré par un opérateur humain pour la détection des événements. Ces systèmes sont devenus semi-automatiques permettant la détection des événements suspects. Ces systèmes sont plus fiables et plus efficaces ou le nombre de séquences est réduit au nombre des événements suspects. Par conséquent, une demande croissante a été enregistrée de la part des centres de santé, hôpitaux et maisons de retraites sur ces systèmes de détection de chutes. La détection de chute basée sur l'approche vidéosurveillance a plusieurs avantages comparés aux deux précédentes approches où l'intervention humaine n'est pas nécessaire ou encore aucun port de capteur n'est requis. Par ailleurs, comparé aux capteurs, la caméra permet d'obtenir des informations plus complètes et détaillées sur la personne (posture, positionnement, localisation, etc.), ainsi que sur son environnement.

1.3. Motivation

La reconnaissance d'activité humaine dans une scène est un sujet de recherche très intéressant et difficile en même temps. L'un des principaux challenges liés à la reconnaissance d'activité humaine est que la même action peut être effectuée de différentes manières, même si elle est effectuée par la même personne (vitesse de déplacement, positionnement des membres du corps, etc.). De nombreux projets de recherche ont été lancés dans le but de concevoir des modèles autonomes et fiables de détection de chutes tels que : ENO FALLS, FATE project (Fall Detector For The Elderly), le projet européen EPF (European Profound Project), Fall-Watch et BIOTELEKINESY. Ces projets peuvent être distingués en deux approches principales à savoir: l'approche basée sur les capteurs et l'approche basée sur la vidéosurveillance. Même si plusieurs systèmes et algorithmes ont été proposés pour la détection des chutes, de nombreux problèmes de détection restent non résolus.

1.3.1. Limitations des méthodes de détection basées sur l'approche capteur :

Les méthodes de détection basées sur les capteurs présentent quelques limitations le fait que les détecteurs de chute basés sur les capteurs au sol ne sont pas suffisamment sensibles à certaines chutes comme les chutes lentes. De plus, l'installation de ces systèmes reste difficile à réaliser pour les particuliers, ou les frais d'aménagement dans les maisons et les appartements restent élevés. Autrement dit, même pour les centres de retraites et de santé, les surfaces équipées par les capteurs restent limitées, car le coût d'installation de ces capteurs sur des surfaces large est très élevé.

Pour le cas de détection basée sur les capteurs portés par la personne, telles que les méthodes basées sur les données d'accéléromètre, la plupart de ces méthodes comparent l'amplitude d'accélération avec un seuil défini manuellement et qui entraîne généralement un nombre élevé de fausses alarmes. En outre, certains capteurs (accéléromètres, capteurs de mesure inertielle) sont sensibles à la présence de bruit causé par les différents appareils présents dans l'environnement de la personne, ce qui génère des erreurs et fausses alarmes. De plus, les capteurs détectent uniquement l'instant de la chute, ainsi d'autres détails, telle que la durée de la chute, restent inconnus. Nous rappelons aussi que beaucoup de personnes âgées oublient ou même trouvent un inconvénient de porter constamment les accéléromètres.

1.3.2. Limitations des méthodes de détection basées sur l'approche vidéosurveillance :

En raison des grandes variations dans l'apparence visuelle du corps et du mouvement des personnes du point de vue de la caméra, ainsi que les occlusions, la présence des bruits, la gestion des grandes quantités de données vidéo ; la reconnaissance d'activités humaine reste une problématique difficile dans le domaine de la vision par ordinateur. Par conséquent, il existe quelques limitations liées aux différentes phases de traitement à savoir la segmentation des corps humains dans une scène, l'extraction des caractéristiques et la classification.

Dans la détection de chutes par vidéosurveillance, différents caractéristiques ont été proposées pour caractériser le corps humain. Cependant, la plupart de ces caractéristiques dépendent fortement de la variation de la taille et du positionnement du corps humain dans l'image (relative à la variation de distance entre les personnes et les caméras). Pour remédier aux problèmes de variations liés à l'échelle, d'autres travaux de recherche [Cucchiara, 2005], [Gao, 2004] utilisent la mise à l'échelle avec un facteur de distance. Mais cette opération nécessite un calibrage préalable de la caméra. La deuxième limitation qui a motivé notre

travail est associée à la phase de classification. Même si plusieurs algorithmes ont été utilisés pour la reconnaissance des activités des personnes, de nombreux algorithmes ne tiennent pas en compte de la variation du temps comme information à l'entrée de l'algorithme de classification. Ces méthodes sont plus appropriées lorsqu'il s'agit de problèmes statiques tels que la classification de postures, mais elles sont moins adaptées aux problèmes séquentiels (dynamiques) telle que la classification de vidéos. L'information du temps est très importante dans les applications séquentielles, en particulier, la durée de la transition entre une posture donnée (debout, accroupi, etc.) à la position couchée. Cette information aide à distinguer clairement les chutes réelles des activités quotidiennes similaires comme s'allonger par exemple.

Inspiré par l'ensemble des limitations existantes (citées précédemment) et rencontrées pendant la phase de la recherche bibliographique, la motivation principale de ce travail était de trouver une représentation adéquate au corps humain (extraite des images ou des vidéos). Ces caractéristiques doivent être à la fois discriminatives (afin de séparer les différentes actions) et globales (afin de classifier les séquences représentant les mêmes actions dans les mêmes classes). La deuxième motivation était d'exploiter les techniques des machines d'apprentissage (machine learning techniques) en considérant l'information du temps, afin de modéliser les différentes activités et bien distinguer les séquences représentant les chutes.

1.4. Principales Contributions

Comme nous l'avons déjà mentionné dans la section 1.3.2, certaines caractéristiques du corps ne considèrent pas les aspects de multiple résolution (l'échelle). Ainsi certaines erreurs de détection sont rencontrées lorsque la taille et la position de la silhouette du corps varient dans l'image. Afin d'éviter ces limitations, dans ce travail nous avons proposé des attributs basés sur les coefficients des curvelets pour caractériser le corps humain. Non seulement les coefficients des curvelets sont connus pour leurs invariabilités par rapport à la translation et à l'échelle, mais en plus aucun calibrage de caméra préalable n'est nécessaire. En outre, cinq rapports de surfaces sont également introduits pour permettre une caractérisation précise de la posture humaine et résoudre le problème de l'invariance par rapport à la rotation.

Une autre approche de représentation du corps humain a été aussi élaborée, à savoir l'utilisation de la décomposition en valeurs singulières tronquée. Cette décomposition était appliquée sur l'image segmentée (comportant uniquement la silhouette), ou les premières valeurs singulières sont sélectionnées comme caractéristiques. Cette décomposition en valeurs singulières est aussi invariante par rapport à la translation et multi résolution.

Pour la phase de détection et de classification, plusieurs algorithmes ont été appliqués afin de mieux classer les différentes postures et surtout bien détecter les chutes réelles avec un nombre réduit de fausses alarmes. Pour cela, nous avons appliqué pour la classification de posture : les réseaux de neurones, Adaboost et les Séparateur à Vaste Marge. Ensuite, pour la classification des chutes, les classes de postures sont définies comme des états des modèles de Markov cachés (HMM classification) ce qui a donné un sens physique aux états du modèle HMM.

Nous avons utilisé par la suite à la fois des données visuelles des caméras de surveillance avec les données de capteurs issues des accéléromètres. Les données de capteurs (signaux) sont utilisées pour (i) une première étape de détection, puisque ces données ne nécessitent pas de phase de prétraitement. Les signaux des accéléromètres sont exploités comme indicateur de séparation entre les activités quotidiennes et les chutes. Pour cela nous avons introduit les exponentielles pondérées à moyennes mobiles (exponentially weighted moving average EWMA) dans la phase de détection. La (ii) deuxième étape consiste en l'application d'une classification en utilisant les vidéos des caméras de surveillance, cette classification est appliquée uniquement sur les cas détectés auparavant pendant la première étape de détection (EWMA detection). Cette étape de classification permet la séparation des chutes réelles des activités similaires, d'où la réduction considérable du taux de fausses alarmes.

1.5. Structure de la thèse

Afin d'organiser cette thèse en parties logiques et cohérentes, nous avons structuré ce manuscrit en six chapitres. Une brève description des différents chapitres est présentée dans cette section. Dans le Chapitre 2, modélisation de l'environnement et segmentation de personne, nous présentons les méthodes de segmentation utilisées dans la littérature. Nous discutons par la suite le choix de la méthode de segmentation basée sur la soustraction de l'arrière-plan, ainsi que la phase des prétraitements et élimination des bruits. Dans le Chapitre 3, extraction des caractéristiques du corps humain, nous présentons un bref aperçu sur les caractéristiques les plus connues dans le domaine de reconnaissance des activités. Nous discutons également leurs avantages ainsi que leurs limitations. Par la suite, nous élaborons en détails les différentes caractéristiques proposées et utilisées dans ce travail à savoir : la décomposition en valeurs singulières tronquée, les coefficients de curvelets et les rapports de surfaces. Dans le chapitre 4, classification de postures humaines, nous présentons les aspects généraux de la classification de posture dans des images statiques. Nous détaillons également

Chapitre 1 : Introduction et objectifs du travail

les différentes méthodes de classification utilisées telles que : Adaboost, les réseaux de neurones et les Séparateurs à Vaste Marge. Le chapitre 5, Classification d'activités humaines, est consacré à la classification des séquences vidéo issues des caméras de surveillance en utilisant les modèles de Markov cachés. Nous élaborons aussi dans ce chapitre la classification des données capteur par l'introduction exponentielles pondérées à moyennes mobiles. Dans le chapitre 6, étude expérimentale, nous décrivons les données utilisées ainsi que une évaluation approfondie sur l'ensemble des approches proposées ou utilisées dans ce travail. Nous présentons par la suite une analyse globale des résultats obtenus en les comparant avec ceux des approches existantes dans l'état de l'art.

Pour finir, nous concluons notre thèse en résumant les principaux avantages et inconvénients des approches proposées et nous présentons alors des orientations futures et des perspectives.

Chapitre 2 :

Modélisation de l'environnement et segmentation de personne (prétraitement)

2.1. Aperçu sur les méthodes de segmentations proposées dans l'état de l'art	12
2.1.1. Segmentation basée sur le flux optique	13
2.1.2. Segmentation par la différence temporelle	13
2.1.3. Segmentation par soustraction de l'arrière-plan	14
2.1.4. Segmentation par le modèle de mixtures de gaussiennes	15
2.2. Description de la méthode de segmentation utilisée (par la combinaison de la soustraction d'arrière-plan et la différence temporelle)	16
2.3. Prétraitement et élimination de bruits en utilisant les opérateurs morphologiques	17
2.3.1. Morphologie mathématique	17
2.4. Conclusion	20

Comme déjà mentionné au premier chapitre, un système automatique de détection de chute est un outil très important pour les personnes âgées afin d'accélérer et améliorer les soins médicaux. En plus, il contribue à fournir des soins médicaux immédiats et résoudre le manque de main-d'œuvre dans le secteur de santé. De nombreux systèmes de détection de chute basés sur la vidéo surveillance ont été proposés. Cependant, la segmentation du corps humain reste une phase primordiale et joue un rôle important dans la détection ; car le résultat de la segmentation a un impact direct sur la robustesse du système global. Du point de vue pratique, le système devrait être en mesure de faire face aux variations des conditions d'éclairage, à la complexité de l'environnement, aux chevauchements du corps avec les objets de la scène et aux objets introduits ou enlevés de la scène. Même si les techniques de segmentation existent depuis plusieurs années, cette tâche reste difficile à réaliser, ce qui maintient toujours la segmentation parmi les axes de recherche d'actualité.

2.1. Aperçu sur les méthodes de segmentation proposées dans l'état de l'art

La segmentation consiste à extraire des régions appartenant à la silhouette du corps et à éliminer tous les pixels appartenant à l'arrière-plan. Plusieurs méthodes de segmentation de silhouette sont

proposées dans la littérature, ces méthodes peuvent être distinguées en quatre principales catégories à savoir : (i) segmentation par la différence temporelle, (ii) segmentation basée sur le flux optique, (iii) segmentation par soustraction de l'arrière-plan, (iv) segmentation par le modèle de mixtures de gaussiennes [Gao, 2004].

Au cours de ces dernières années, la segmentation par soustraction de l'arrière-plan est devenue plus populaire et beaucoup de travaux de recherche sont orientés vers le développement et l'amélioration de cette technique telle que l'optimisation dans l'estimation d'un fond dynamique et l'actualisation de l'arrière-plan [Wang, 2014]. Stauffer et Grimson [Stauffer, 1999] ont proposé un modèle de gaussiennes flexible et adapté aux objets en déplacement par rapport à l'arrière-plan.

2.1.1. Segmentation basée sur le flux optique :

Le principe de la segmentation basée sur le flux optique est de calculer les champs de vecteurs des objets en mouvement dans une séquence d'image. A partir de ces champs de vecteurs, il est possible de segmenter les régions en mouvement. Le calcul du flux optique est souvent complexe, d'où la principale contrainte de la segmentation basée sur le flux optique qui est le temps de calcul [Barron, 1994]. Ce qui fait que pour les applications en temps réel tel que la détection de chutes, le nombre de frames par seconde est diminué (une partie seulement des frames est prise en considération) afin de réduire le temps de calcul de la segmentation. Plusieurs travaux de recherche ont utilisé la segmentation basée sur le flux optique, mais la zone de recherche était limitée dans les frames afin d'augmenter la rapidité du traitement [Gao, 2004]. De plus, le flux optique est sensible au bruit présent dans l'image, ce qui le rend peu souhaitable dans les conditions d'éclairage variables. D'autres travaux de recherche ont utilisé la segmentation basée sur le flux optique pour la détection du mouvement des personnes, mais la plupart des scènes sont prises à l'intérieur et avec des conditions d'éclairage stables [Nguyen, 2015]. Gao et al. [Gao, 2004] ont appliqué le flux optique pour la détection des avant-bras et la tête d'une personne. Jean et al. [Jean, 2005] proposent le flux optique pour le suivi des pieds d'une personne mais en limitant leur recherche à une zone restreinte.

2.1.2. Segmentation par la différence temporelle :

Le principe de la segmentation par la différence temporelle reste simple comparé à celui basée sur le flux optique ou il consiste à détecter les pixels non stationnaires dans une séquence d'images. Cette segmentation est basé réellement sur la différence entre ces différentes images consécutives afin d'extraire les changements temporels. La principale contrainte de cette méthode de

segmentation est la vitesse de passage de la silhouette à segmenter dans l'image. Si le passage de la personne est rapide, la silhouette en mouvement à l'instant t et celle de l'instant $t-1$ seront détectées doublement. Par contre, si le passage de la personne est lent, les pixels formants la silhouette à l'instant t et $t-1$ seront pratiquement superposables. D'où, au bout d'un certain nombre de frames, la silhouette risque de ne pas être segmentée comme étant un objet en mouvement et sera ainsi confondue avec l'arrière-plan.

2.1.3. Segmentation par soustraction de l'arrière-plan:

La segmentation par soustraction de l'arrière-plan est devenue la méthode la plus utilisée dans de nombreux systèmes de traitement vidéo. Cette méthode est généralement choisie grâce à la simplicité de son implémentation d'une part et en plus de son efficacité pour les fonds de scènes non-complexes tels que les environnements intérieurs ou les scènes à fonds stationnaires. Le principe de la segmentation par la soustraction de l'arrière-plan est de définir une image comme référence afin d'éliminer les pixels inchangés dans les frames suivants. En effet, le modèle d'arrière-plan est construit en utilisant plusieurs images successives pour localiser les pixels stationnaires et réduire les variations dues à la variabilité des conditions d'éclairage, etc. Par la suite, un seuil est nécessaire pour décider si le pixel a changé ou il est stationnaire. D'où, le pixel est affecté soit, au premier plan ou à l'arrière-plan. Pour cela, chaque pixel (x, y) est associé la valeur absolue de la différence entre l'image actuelle $I(x, y)$ et le modèle d'arrière-plan $B(x, y)$. Un pixel $I(x, y)$ est marqué comme premier plan si seulement la condition sur la différence est vérifiée :

$$|I(x, y) - B(x, y)| > \text{Seuil} \quad (2.1)$$

La valeur du seuil est généralement choisie empiriquement, en tenant compte des conditions de la scène, des caractéristiques de la caméra, la vitesse du déplacement des personnes etc.

Afin de tenir en considération des éventuels changements dans le modèle d'arrière-plan, une mise à jour de l'arrière-plan est estimée comme suit :

$$B_{i+1}(x, y) = \alpha I_i(x, y) + (1 - \alpha) B_i(x, y) \quad (2.2)$$

Ou B_{i+1} et B_i représentent l'ancien et le nouvel arrière-plan respectivement et I_i représente l'actuelle image d'entrée. Le paramètre α est maintenu souvent petit pour éviter la considération des fausses détections telles que la présence de taches dans l'image d'entrée.

En différence d'image, si le pixel est considéré comme en mouvement, l'étiquette correspondante est mise à zéro. Dans le cas contraire, si le pixel est stationnaire pendant N frames, il est alors considéré comme une région d'arrière-plan. En général, la valeur de N est établie expérimentalement, en tenant compte de la vitesse de déplacement des personnes. Pour une image d'entrée, la valeur du seuil est définie par 2σ , où sigma σ est l'écart type du pixel I (x, y) dans l'arrière-plan enregistré [Nguyen, 2015].

2.1.4. Segmentation par le modèle de mixtures de gaussiennes:

La segmentation par un modèle adaptable de mixture de gaussienne [Blake, 1997] est l'une des solutions pour remédier aux problèmes de variation d'éclairage dans la scène observée. Les valeurs radiométriques d'un pixel particulier au fil du temps sont considérées comme une évolution « pixel process». Cette évolution est une série temporelle de valeurs radiométriques représentée comme suit:

$$\{X_1, \dots, X_t\} = \{I(x_0, y_0, k) : 1 \leq k \leq t\} \quad (2.3)$$

Si la plupart des pixels présentent une certaine homogénéité par rapport aux conditions d'éclairage, une simple gaussienne sera suffisante pour modéliser l'ensemble de ces pixels, tout en tenant compte du bruit d'acquisition. Par contre en pratique, souvent les pixels présentent une variabilité importante relative aux conditions d'éclairage changeantes. Ainsi, de multiples Gaussiennes adaptables seront nécessaires pour la modélisation des pixels [Blake, 1997]. La variabilité d'un même pixel dans plusieurs images successives, $\{X_1, \dots, X_t\}$, est modélisée par une mixture de K distributions gaussiennes. La probabilité d'observation du pixel est calculée comme suit:

$$P(X_t) = \sum \omega_{i,t} \eta(X_t, \mu_{i,t}, \Sigma_{i,t}) \quad (2.4)$$

Où $\omega_{i,t}$, est le paramètre de poids de la $i^{ème}$ distribution gaussienne; η est la fonction de densité de probabilité de la $i^{ème}$ distribution Gaussienne. Pour chaque image, la valeur radiométrique du pixel (x_0, y_0) est de X_t . Une distribution gaussienne est jugée correspondante si la valeur X_t est dans 2,5 fois la valeur de son écart type. En effet, si la répartition des pixels correspond à la Gaussienne, alors cette dernière sera marquée comme sélectionnée. Sinon, si aucune gaussienne ne correspond à sa répartition, le pixel X_t est alors détecté comme un pixel de l'avant plan.

Plusieurs travaux de recherches ont adopté cette approche dans la segmentation de personne ou d'objet en général. Dans [Zivkovic, 2004], les auteurs ont proposé une méthode de segmentation adaptative en utilisant la densité de probabilité des mixtures gaussiennes, ou il a utilisé les

équations récursives pour la mise à jour du nombre et des paramètres des mixtures suivant la complexité du voisinage du pixel. D'un autre côté, Genovese et Napoli [Genovese, 2014], ont proposé l'implémentation matérielle de la segmentation basée sur les mixtures gaussiennes sous ASIC et FPGA. Les circuits proposés permettent ainsi d'effectuer une identification d'arrière-plan en temps réel sur des séquences vidéo haute définition (HD).

2.2. Description de la méthode de segmentation utilisée (par la combinaison de la soustraction d'arrière-plan et la différence temporelle) :

Pour réaliser un système de détection de chute robuste et rapide, il faut trouver le bon compromis entre d'une part une segmentation efficace (en tenant compte les changements d'éclairages, les contraintes de l'environnement, .etc.) et d'une autre part la réduction de la complexité de l'algorithme (pour simplifier son implémentation). Dans ce travail, nous avons combiné les principes des deux approches de segmentation basés sur la soustraction d'arrière-plan et la différence temporelle.

Dans le système implémenté, nous avons en premier converti les images de l'espace de couleur rouge-vert-bleu RGB (Red, Green, Blue) en l'espace de couleur YUV, où Y correspond à la composante luminance, U et V correspondent aux composantes chrominances. Le calcul des nouvelles composantes YUV se fait comme suit :

$$Y = 0,299 R + 0,587 G + 0,114 B \quad (2.5)$$

$$U = 0,492 (B - Y) \quad (2.6)$$

$$V = 0,877 (R - Y) \quad (2.7)$$

La deuxième étape consiste à construire le modèle d'arrière-plan de la scène. Pour cela, nous avons appliqué le principe de la différence temporelle. La sélection des pixels stationnaires se base sur la différence de plusieurs images successives, pour remédier aux problèmes de changement de pixels causé par les variations d'éclairage, etc. Une valeur de seuil est nécessaire pour décider si le pixel appartient au premier plan ou à l'arrière-plan. Chaque différence entre deux images successives, est alors comparée au seuil défini. Soit le pixel est considéré comme stationnaire, d'où son étiquette est incrémentée. Par contre, si le pixel est considéré en mouvement, son étiquette garde sa valeur précédente. Après un certain nombre d'images successives, un pixel est considéré en mouvement si la valeur cumulée de l'étiquette dépasse la valeur d'un deuxième seuil N. Dans ce cas, ce pixel est classé comme pixel appartenant à l'avant-

plan ou à la silhouette. Dans le cas contraire, un pixel est considéré comme stationnaire, la valeur cumulée de l'étiquette étant inférieure au seuil N .

La valeur du seuil N est définie de manière expérimentale, en tenant compte de la vitesse de déplacement des personnes dans la scène ainsi qu'à la vitesse de la caméra c'est-à-dire le nombre d'images fournies par seconde. Après l'affectation des pixels soit d'arrière-plan, soit en pixels constituant la silhouette humaine ; l'image résultante de la segmentation est alors une image binaire. La prochaine étape consiste en l'amélioration du résultat de la segmentation, par l'élimination des pixels bruités, les prétraitements, etc. La section 2.3 est consacrée à la description des différentes techniques utilisées.

2.3. Prétraitement et élimination de bruits en utilisant les opérateurs morphologiques

Après la phase de segmentation, plusieurs régions de bruit peuvent être observées. Pour éliminer ce bruit, nous avons appliqué les opérateurs morphologiques tels que l'opérateur d'érosion et de dilatation. Dans cette section, nous décrivons brièvement le principe de ces techniques dans l'amélioration de la qualité de la silhouette segmentée.

2.3.1. Morphologie mathématique :

Le module de restauration des silhouettes (ou d'objets d'une manière générale) dans une image binaire segmentée consiste à supprimer les bruits et cela par l'application des différents filtres sur l'image. La morphologie mathématique (MM) est une théorie mathématique qui est souvent utilisée dans le traitement, l'analyse d'images et l'amélioration de la qualité des objets segmentés [Nguyen, 2015]. Cette théorie est basée généralement sur le concept de forme issu de la théorie des ensembles et non sur la modélisation mathématique traditionnelle [Veenman, 2001]. Dans la théorie de la morphologie mathématique, les images sont traitées comme des ensembles et les transformations morphologiques dérivées de l'addition et de la soustraction de Minkowski sont définies pour extraire les caractéristiques des images [Haralick, 1987]. Les opérateurs de morphologie mathématique les plus connus et qui sont utilisés dans ce travail sont la dilatation, l'érosion, l'ouverture et la fermeture.

Ces opérations morphologiques fonctionnent avec deux images, à savoir : l'image segmentée originale (bruitée) et un élément structurant. Chaque élément structurant a une forme qui peut être considérée comme un paramètre de l'opération. Pour les éléments structurants, il existe deux types de connexion les plus courants : à 4 ou 8 connexions (comme le montre la figure 2.2). Comme nous l'avons déjà mentionné précédemment, l'application des opérateurs morphologiques se fait

sur des images binaires, dans lesquelles les pixels constituant la silhouette sont codés par "1" et les pixels appartenant à l'arrière-plan sont codés par "0".

- **Dilatation**

La dilatation est une opération qui fait agrandir les objets détectés dans l'image binaire. Le résultat de la dilatation est directement dépendant à la fois à la taille et la forme de élément structurant. L'opération mathématique qui définit la dilatation d'une image binaire A par un élément structurant binaire B est définie comme suit:

$$A \oplus B = \{x | (\hat{B})_x \cap A \neq \emptyset\} \quad (2.8)$$

- **Erosion**

L'opération d'érosion est exactement l'inverse de la dilatation. C'est une opération qui fait réduire les objets détectés dans l'image binaire. L'érosion d'une image binaire A par l'élément structurant binaire B est définie comme suit:

$$A \ominus B = \{x | (B)_x \subseteq A\} \quad (2.9)$$

- **Ouverture**

Les opérateurs d'ouverture et de fermeture ne sont que des combinaisons de dilatation et d'érosion. L'ouverture est simplement l'érosion d'une image A par un élément structurant B suivi d'une dilatation par le même élément structurant. L'opération mathématique qui définit l'ouverture de A par un élément structurant binaire B est définie comme suit:

$$A \circ B = (A \ominus B) \oplus B \quad (2.10)$$

- **Fermeture**

La fermeture d'une image est également une combinaison d'érosion et de dilatation. La différence est située dans l'ordre d'application. C'est une dilatation suivie par une érosion. L'opération mathématique qui définit l'ouverture de A par un élément structurant binaire B est définie comme suit:

$$A \bullet B = (A \oplus B) \ominus B \quad (2.11)$$

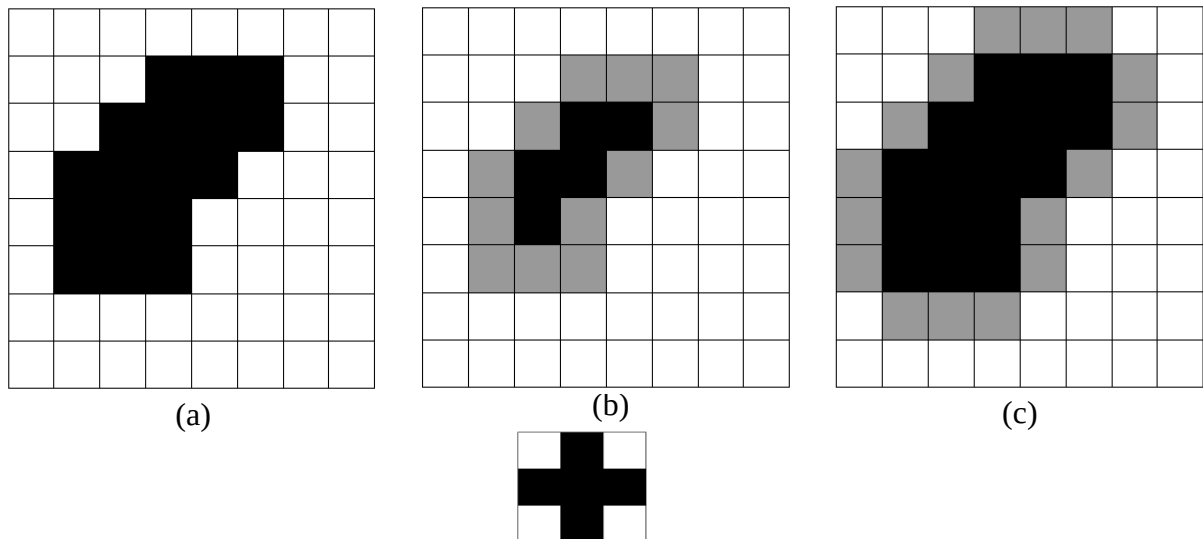


Figure 2.1 Exemple illustrant les deux opérateurs morphologiques : l'érosion (b) et la dilatation (c).

Un exemple illustrant les deux opérateurs morphologiques, à savoir la dilatation et l'érosion est représenté dans la figure 2.2. Afin de montrer la différence entre les résultats, la même image originale A et le même élément structurant B ont été utilisés pour les deux opérateurs.

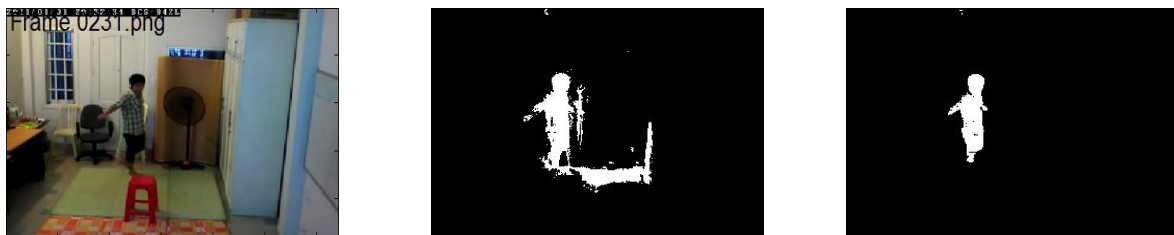


Figure 2.2. Exemple d'élimination de bruit en utilisant les opérateurs morphologiques [Nguyen, 2015].

Pour l'opérateur d'érosion (image b) les blocs en gris sont ceux qui disparaîtront par rapport à l'image originale. Et de même pour le cas de dilatation (image c) les blocs en gris sont ceux ajoutés par rapport à l'image l'original. Afin de mieux illustrer cette étape, un exemple d'élimination de bruit en utilisant les opérateurs morphologiques est représenté dans la figure 2.3. La première image représente une image d'entrée, la deuxième image représente le résultat de segmentation (ou plusieurs zones de bruit sont observées), tandis que la troisième image représente le résultat de l'image segmentée après élimination de bruits par les opérateurs morphologiques.

2.4. Conclusion

Dans ce chapitre, nous avons présenté, dans un premier lieu, un aperçu sur les différentes techniques de segmentation proposées dans la littérature. Par la suite, nous avons détaillé la méthode de segmentation utilisée dans ce travail à savoir la combinaison de la soustraction d'arrière-plan et la différence temporelle. La segmentation de la silhouette humaine reste une phase primordiale dans un processus de détection, puisque l'image segmentée représente le premier module du système et sa précision influe considérablement sur les modules de traitement suivants (extraction de caractéristiques, classification, etc.). Le chapitre suivant est consacré à la phase d'extraction des caractéristiques du corps humain.

Chapitre 3 :

Extraction des caractéristiques du corps humain

3.1. Aperçu sur les méthodes existantes de caractérisation du corps humain	21
3.1.1. Caractérisation par Points	22
3.1.2. Caractérisation par formes géométriques primitifs	23
3.1.3. Représentation par les contours	24
3.1.4. Représentation par les modèles de formes articulées	24
3.1.5. Représentation par les modèles squelettiques	25
3.1.6. Représentation par densités de probabilité	26
3.1.7. Représentation par modèles (template)	26
3.2. Utilisation de décomposition en valeurs singulières tronquées	27
3.3. Utilisation des coefficients de curvelets comme caractéristiques	30
3.4. Proposition des rapports de surface comme caractéristiques	32
3.5. Réduction de la dimension en utilisant l'algorithme de l'évolution différentielle (differential evolution algorithm)	35
3.6. Conclusion	37

L'extraction de caractéristiques est considérée comme une étape cruciale dans la classification des chutes. Ces caractéristiques ont un impact direct sur la performance du système de classification. Pour cela, les caractéristiques extraites de l'image doivent être invariantes (i) à la translation c.-à-d. lorsque la position de la personne change dans l'image et (ii) au facteur d'échelle c.-à-d. lorsque la dimension de la silhouette changent dans l'image. Cette dimension de silhouette dans l'image est proportionnelle à la distance entre la personne surveillée et l'emplacement de la caméra. Dans ce chapitre, nous apportons plus de détails sur l'ensemble des caractéristiques proposées dans ce travail.

3.1. Aperçu sur les méthodes de caractérisation du corps humain existantes:

Les objets ou les silhouettes en particulier sont généralement représentées par leurs formes ou leurs apparences. La représentation en utilisant les formes paraît la plus évidente [Nguyen, 2015]. Cependant, il existe de nombreuses façons de représenter ou caractériser la silhouette humaine dans la littérature. Il est à noter que pour plusieurs applications d'identification de comportements comme la détection de chutes, les représentations à base de formes sont également combinées avec d'autres caractéristiques [Yu, 2012].

Dans cette section, nous décrivons brièvement les différentes caractéristiques proposées dans l'état-de-l'art et dans le contexte de la représentation de la silhouette humaine; ces caractéristiques sont basées sur : les points, les formes géométriques primitives, les contours de l'objet et les modèles de forme articulés, etc.

3.1.1. Caractérisation par points : plusieurs travaux de recherche utilisent la représentation du corps humain par un ou un ensemble de points. Dans le cas d'une représentation par un seul point, ce point est généralement associé au centre de masse (centroïde) du corps comme le montre la Figure 3.1 (a) [Veenman, 2001]. Par contre pour la représentation par un ensemble de points, ceux-ci sont repartis sur les membres du corps comme l'illustre la Figure 3.1 (b) [Serby, 2004]. Cependant, cette représentation de points convient aux silhouettes qui occupent de petites régions dans une image et ne permet pas la caractérisation détaillée du corps. De plus, cette représentation de points est plus appropriée aux applications qui n'ont pas beaucoup de gestes à classifier, ou ces derniers sont facilement séparables et ne présentent pas une grande confusion.

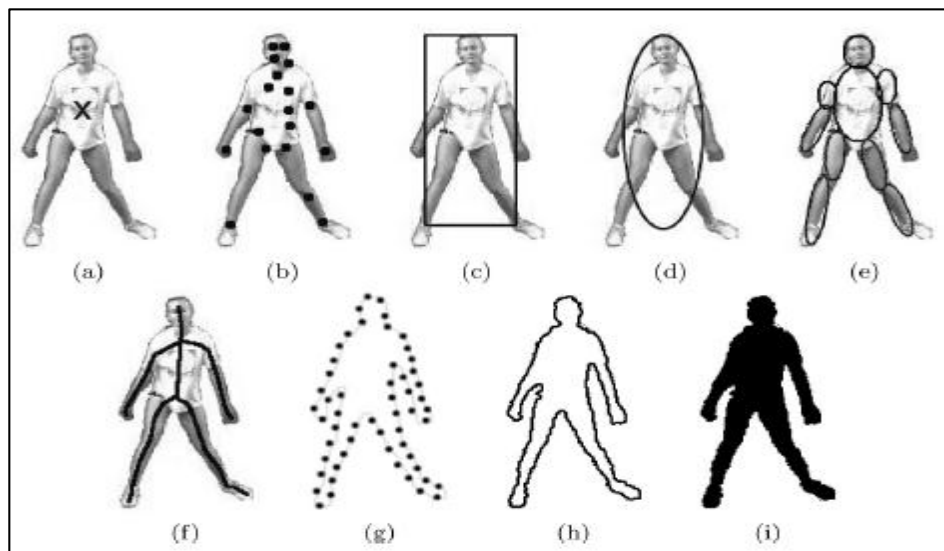


Figure 3.1. Les différentes représentations de silhouette [Nguyen, 2015].

Parmi Les représentations en points les plus connues, on peut citer celle des points d'intérêts (key points) issues de la transformation de caractéristiques visuelles invariante à l'échelle (Scale-invariant feature transform SIFT) [Nguyen, 2015]. Ces points sont des descripteurs locaux qui captent les caractéristiques dans une image. Ils sont très connus pour leur efficacité dans la détection et la reconnaissance d'objet. Néanmoins, les représentations en points d'intérêts ne prennent pas en compte toutes les actions du corps ce qui réduit son utilisation

dans les systèmes d'identification et de reconnaissance d'activité humaine [Nguyen, 2015] et [Miaou, 2006].

3.1.2. Caractérisation par formes géométriques primitives : La seconde représentation est celle basée sur les formes géométriques primitives telles que le rectangle ou l'ellipse comme le montre la Figure 3.1 (c) et (d) [Yilmaz, 2004]. Bien que les formes géométriques primitives soient souvent adaptées à la représentation d'objets rigides et simples, elles sont également utilisées pour les d'objets non rigides et courbés tel que la silhouette humaine. Cette représentation a été adoptée dans plusieurs travaux de recherches. Foroughi [Foroughi, 2008] et Yu [Yu, 2012] ont proposé des systèmes de soins à domicile avec une application de détection de chutes où ils ont utilisé des approximations de l'ellipse à la silhouette de corps en tant que caractéristiques. Miaou et al. [Miaou, 2006] ont utilisé le rectangle circonscrit (bounding box) englobant la silhouette segmentée. Par la suite, d'autres paramètres peuvent être extraits de ces formes tels que la hauteur et la largeur du rectangle circonscrit, les dimensions des grands et petits axes de l'ellipse, ainsi que leurs orientations.

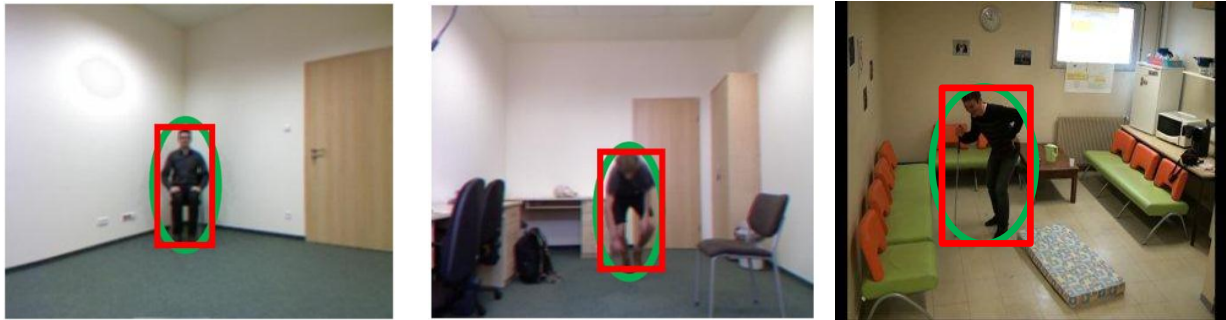


Figure 3.2. –Exemples de représentations de silhouette en utilisant les formes géométriques (a) en position assise (b) en position accroupi, (c) en position penchée.

Cependant, ces caractéristiques basées sur les formes géométriques présentent quelques limitations, ou elles ne peuvent pas toujours distinguer entre les diverses postures humaines, en particulier lorsqu'il existe un degré élevé de similitude entre les postures comme les positions assise et accroupie par exemple. La figure 3.2 illustre quelques contraintes rencontrées lors d'une représentation géométrique. Les dimensions et les orientations des rectangles circonscrits et des ellipses approximatives sont presque identiques pour les postures assise et accroupie (figures 3.2 (a) et (b)). La figure 3.2 (c) représente une personne avec une canne, ou le rectangle circonscrit et l'ellipse ont englobés la canne avec le corps. Cette confusion influence fortement les dimensions et les orientations de ces formes. Pour ces raisons, nous avons écarté l'idée de considérer les formes géométriques comme caractéristiques du corps.

3.1.3. Représentation par contours : La représentation de la silhouette ou d'objets dans une image peut être aussi définie par le contour délimitant cette silhouette ou cet objet (Figure 3.1 (g) et (h)). Cette représentation nécessite généralement une phase de binarisation afin d'extraire ces contours. L'avantage de cette représentation est que les contours s'adaptent exactement à la forme des silhouettes quel que soit la complexité de la forme. Cet avantage permet une meilleure description comparée aux représentations basées sur les formes géométrique primitives, où la représentation des formes complexes et non rigides est confondue [Nguyen, 2015]. Plusieurs travaux de recherche ont adopté cette représentation ; Yilmaz et al [Yilmaz, 2004] ont proposé une méthode de suivi de personnes basée sur l'évaluation des contours d'une image à une autre. Cette méthode peut s'adapter à la variation des conditions de vision et aux problèmes d'occlusions [Yilmaz, 2004].

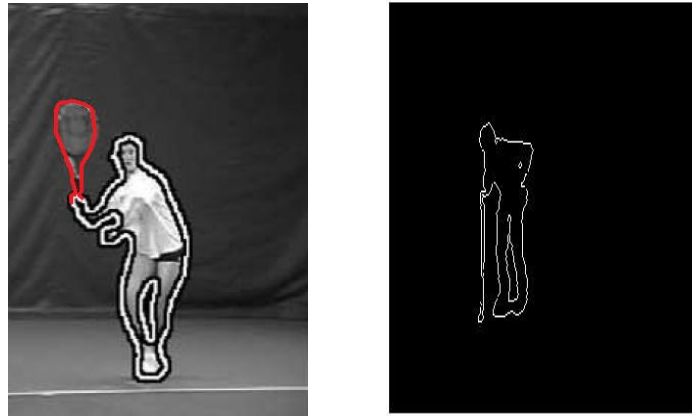


Figure 3.3. –Exemples de représentations de silhouette en utilisant les contours présentant une confusion d'objets avec la silhouette humaine : (a) confusion avec la raquette (b) confusion avec la canne.

Cependant, la représentation en contours présente certaines limites telles que la discontinuité des contours ce qui nécessite l'application d'autres algorithmes (comme les contours actifs 'snakes' [Marouf, 2017]) afin de connecter ces contours. L'utilisation de ces algorithmes peut être contraignante surtout pour les applications en temps-réel. La figure 3.3 présente une deuxième limitation de la représentation en contours, causée généralement par la confusion d'objet (une canne, une raquette, etc.) avec la silhouette humaine.

3.1.4. Représentation par les modèles de formes articulées : afin de remédier aux problèmes rencontrés lors d'une représentativité par une seule forme géométrique (carré ou ellipse), la représentation en formes articulées considère plusieurs formes correspondantes généralement aux différents membres du corps (figure 3.1(e)).

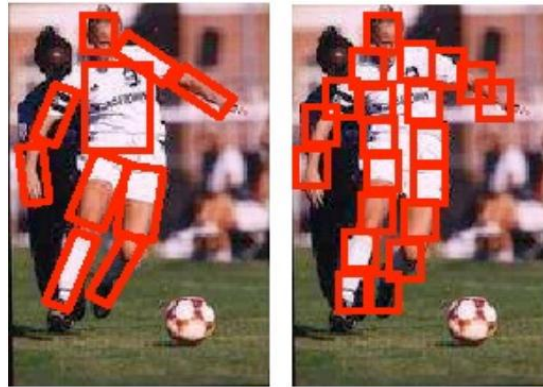


Figure 3.4. –Exemples de représentations de silhouette en utilisant les formes articulées [Yang, 2013].

Ces formes sont liées entre elles par des articulations, ou l'ensemble de ces formes représente les caractéristiques du corps. Par exemple, le corps humain est un objet articulé avec le torse, les jambes, les mains, la tête et les pieds reliés par les articulations. La représentation de la silhouette par des formes articulées peut être modélisée par des carrés, cylindres ou des ellipses comme le montre la Figure 3.4. Plusieurs travaux de recherche ont utilisé cette représentation ; Yang et autres ont utilisé des carrés articulés pour représenter la silhouette humaine [Yang, 2013]. Ces carrés sont de différents gabarits afin de permettre une meilleure couverture, d'où une meilleure représentativité du corps. Les articulations entre les différentes formes sont structurées en arborescence, ou le modèle proposé peut être efficacement optimisé avec une programmation dynamique [Yang, 2013].

3.1.5. Représentation par les modèles squelettiques : La représentation de la silhouette humaine par la squelettisation est aussi adoptée par plusieurs chercheurs dans les applications de reconnaissance de comportement (figure 3.1(f)). La squelettisation de la silhouette peut être extraite en appliquant plusieurs transformations comme l'amincissement, ou l'utilisation de l'axe médian de la silhouette [Yu, 2013]. Cette représentation permet la modélisation des objets à la fois articulés ou rigides. Le degré de similitude entre les squelettisations de deux postures peut être mesuré en comparant leurs cartes de distances (distance map) [Hsu, 2005].

3.1.6. Représentation par la densité de probabilité:

Cette représentation de silhouette est basée sur l'espace de caractéristiques texturées généré par le filtrage d'images texturées en utilisant des filtres isotropes et anisotropes et en analysant leurs réponses en tant que fonctions de densité de probabilités conditionnelles multi-composantes. Ces estimations de densité de probabilité sont soit (i) paramétriques, gaussienne ou une mixture de gaussiennes [Paragios, 2002] ou (ii) non paramétriques. Les densités de probabilité des caractéristiques d'apparence d'objet sont calculées à partir des régions d'image spécifiées par les modèles de forme (en utilisant l'ellipse, le contour, etc.) [Nguyen, 2015].

3.1.7. Représentation par gabarit (template): La représentation de la silhouette en utilisant les gabarits est de définir des échantillons de postures humaines et représenter par la suite le reste des postures en fonction des échantillons préalablement définis. Ces modèles de postures sont généralement représentés par des formes géométriques simples ou des images de silhouettes [Nguyen, 2015]. L'avantage de ces caractéristiques réside dans leur invariance par rapport aux facteurs d'échelle et de translation. Cependant, cette représentation ne convient que pour les silhouettes ou postures à faibles variations. Autrement dit ces caractéristiques ne prennent pas en compte de nouvelles variations dans l'apparence des silhouettes au cours du suivi, puisque les modèles sont déjà définis.

Dans cette section, nous avons abordé les caractéristiques de silhouettes les plus connues, c'est bien évident qu'il existe d'autres caractéristiques qu'on n'a pas décrits telles que : la représentation en repères [Nguyen, 2015], les caractéristiques de textures, les modèles d'appartenance active [Alhimale, 2014], modèles à vue multiple, génération de sous espaces (comme l'Analyse en Composantes Principales (PCA) et Analyse en Composantes Indépendantes (ICA)), etc.

Dans notre travail de recherche, nous avons utilisé plusieurs approches durant la phase d'extraction de caractéristiques, à savoir la décomposition en valeurs singulières tronquée, l'utilisation des coefficients de curvelets et enfin nous avons proposé des rapports de surface comme attributs de la silhouette humaine. L'avantage des caractéristiques utilisées et proposées dans ce travail est leurs invariances par rapport à la translation et à la l'échelle, de plus aucun calibrage de caméra préalable n'est nécessaire.

3.2. Utilisation de la décomposition en valeurs singulières (SVD) tronquées

Comme première exigence dans l'extraction de caractéristiques, nous voudrions que les caractéristiques soient aussi descriptives de la silhouette humaine que possible. D'autre part, seuls des vecteurs de caractéristiques de taille limitée seront souhaités, ceci afin de tenir compte des exigences pratiques et assurer une efficacité dans les calculs. Pour un premier travail, nous avons proposé l'utilisation des coefficients SVD comme caractéristiques. En étant motivé par le concept essentiel de la SVD tronquée qui est d'utiliser uniquement un nombre réduit de valeurs singulières pour se rapprocher de l'image originale [Aase, 1999] et [Zerrouki (1), 2014]. Supposant une image I de dimension $m \times n$, la décomposition SVD consiste en la factorisation de l'image I en trois matrices comme suit:

$$I = U * S * V^T, \quad (3.1)$$

Où V^T représente la transposée de la matrice V . La matrice S est une matrice diagonale contenant les valeurs singulières non négatives. Ces valeurs singulières sont classées dans un ordre décroissant. Admettant qu'il existe r valeurs singulières, où r est appelé le rang de l'image I . Les matrices U et V sont des matrices orthogonales. La matrice U est de dimension $m \times r$, alors que la matrice V est de dimension de $n \times r$. Dans notre cas, nous n'avons utilisé que les valeurs singulières fournies par la diagonale de la matrice S et constituant le vecteur de caractéristique. Ce choix de caractéristiques est justifié par la concentration de l'information utile dans les valeurs singulières, contrairement aux deux autres matrices (U et V) qui sont plutôt utilisées pour la reconstruction de l'image I .

Après la sélection des valeurs singulières en tant que caractéristiques, la prochaine étape consiste à réduire la dimension de vecteur caractéristique (les r valeurs) pour permettre un temps de calcul souhaitable (pendant l'extraction des caractéristiques ainsi que pendant les phases d'apprentissage et de tests des classifieurs). Pour remédier à ce problème, nous avons préféré représenter l'image I par son image d'approximation I' , qui correspond à un rang tronqué (ou réduit) r' , l'image d'approximation est définie comme suit:

$$I' = U * S' * V^T, \quad (3.2)$$

La matrice S' a les mêmes dimensions que celle de la matrice S , sauf qu'elle ne contient que les plus grandes valeurs singulières (qui représentent les premières valeurs et les plus importantes en même temps). Les valeurs singulières non sélectionnées (et qui sont jugées non importantes) sont alors remplacées par des zéros dans la matrice S' . Cette décomposition

en valeurs singulières tronquées est connue pour son efficacité surtout dans les applications de réduction de dimension. Cependant, le problème de cette décomposition réside dans la sélection des valeurs singulières les plus importantes (le critère d'arrêt). Plusieurs mesures ont été proposées dans la littérature. Dans notre cas, nous avons utilisé le rapport des énergies des deux images, à savoir l'image originale I et son approximation I' . Les énergies E et E' des deux images (I et I' respectivement) sont définies par les sommes des carrés des valeurs singulières correspondantes, comme le montrent les équations (3.4) et (3.5) :

$$E = s_{11}^2 + s_{11}^2 + \dots + s_{rr}^2 \quad (3.4)$$

$$E' = s_{11}^2 + s_{11}^2 + \dots + s_{r'r'}^2 \quad (3.5)$$

Où s_{ii} représentent les valeurs singulières des matrices S et S' :

$$S = \begin{bmatrix} s_{11} & 0 & \dots & 0 \\ 0 & s_{22} & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & s_{rr} \end{bmatrix} \quad \text{et} \quad S' = \begin{bmatrix} s_{11} & 0 & \dots & 0 \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & s_{r'r'} & \vdots \\ 0 & \dots & \dots & 0 \end{bmatrix} \quad (3.6)$$

En outre, le rapport des énergies (ER) des deux images I et I' est définie par la relation suivante:

$$ER (\%) = \left(E' / E \right) \times 100 \quad (3.7)$$

$$ER (\%) = \left(\sum_1^{r'} s_{ii}^2 / \sum_1^r s_{ii}^2 \right) \times 100 \quad (3.8)$$

La variation du rapport d'énergie est proportionnelle au nombre de valeurs singulières tronquées choisies, comme le montre la Figure 3.5. Cependant, nous pouvons constater que l'information essentielle de l'image (en termes d'énergie) est concentrée dans les premières valeurs singulières [Aase, 1999].

Chapitre 3 : Extraction des caractéristiques du corps humain

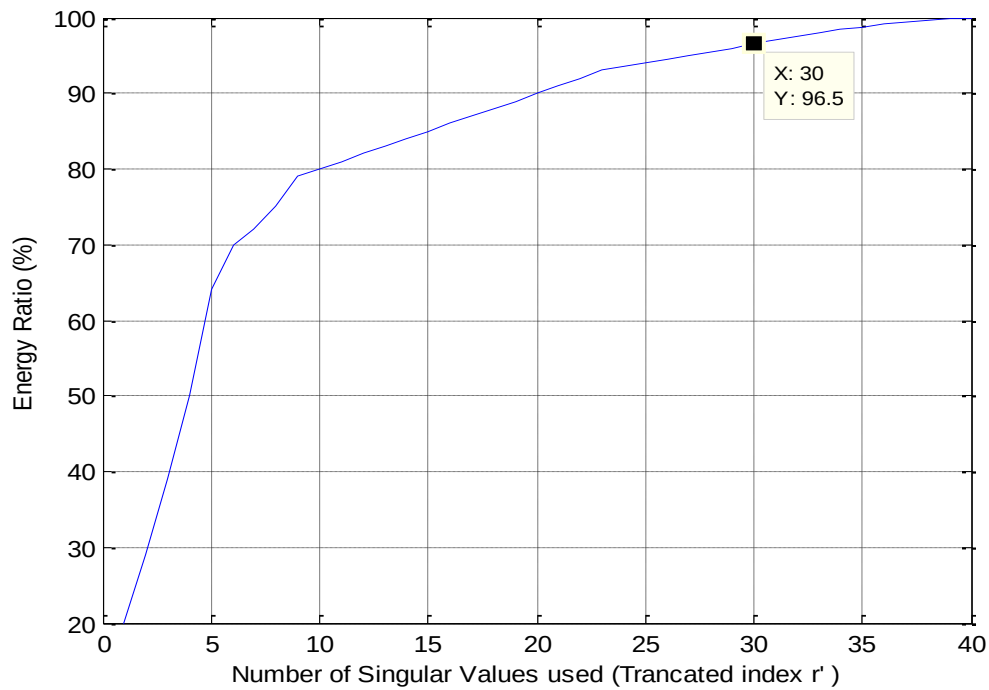


Figure. 3.5 La variation entre le rapport d'énergie des deux images en fonction du rang tronqué r' .

Cette décomposition permet l'élimination de l'information redondante ou inutile dans la représentativité des silhouettes. En d'autres termes, la représentation de la silhouette humaine dans l'image segmentée revient juste à considérer les premières valeurs singulières qui seront sauvegardées sous forme de vecteur.

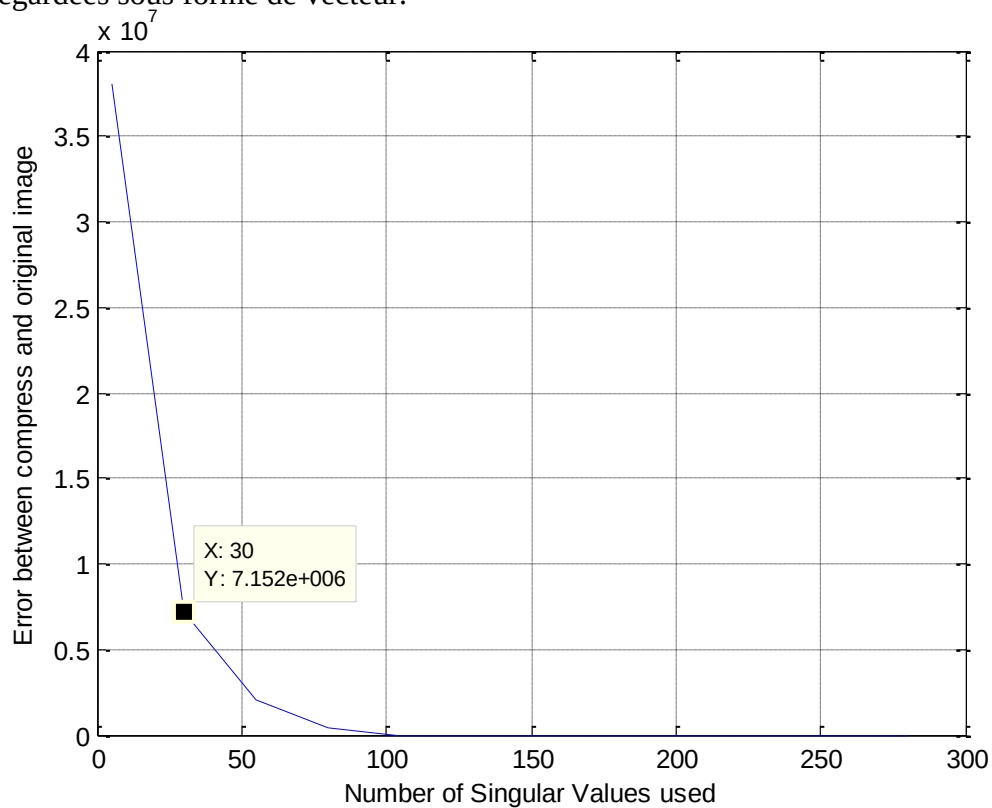


Figure. 3.6 La variation entre l'erreur quadratique des deux images en fonction du rang tronqué r' .

Le choix du nombre des valeurs singulières dépend essentiellement du compromis entre le rapport d'énergie et le taux de bonne classification. Dans notre travail, ce compromis a été trouvé après avoir testé plusieurs nombre de valeurs singulières, en considérant plusieurs conditions d'éclairage et dans différents environnements, etc. ou trente valeurs singulières ont été sélectionnées $r' = 30$.

Nous avons confirmé le choix du nombre des valeurs singulières par le calcul de l'erreur quadratique entre les deux images (segmentée et son approximation) comme le montre la figure. 3.6. On remarque que l'erreur a drastiquement chuté pour la valeur de $r' = 30$.

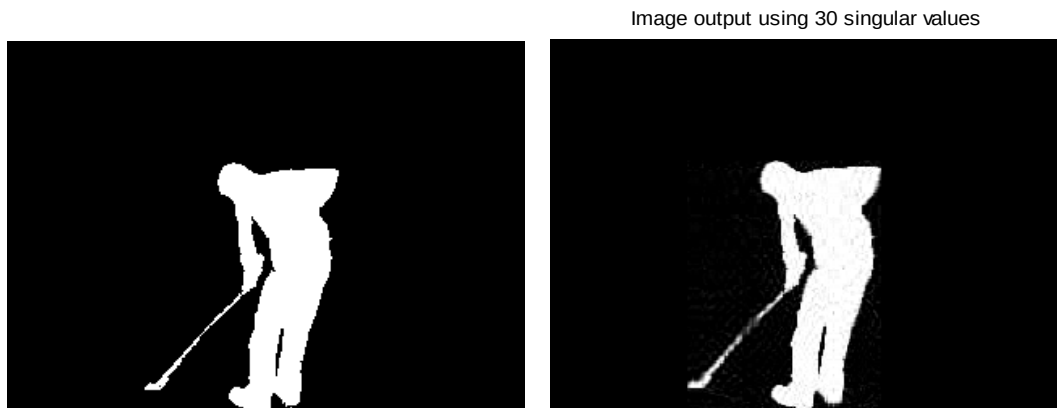


Figure. 3.7 Comparaison entre les deux images : segmentée (a) et son approximation (b) en utilisant un rang tronqué $r' = 30$.

Il est à noter que même après la valeur de $r' = 30$ l'erreur quadratique n'est pas nulle, mais cette erreur correspond principalement aux pixels de détails (qui sont considérés comme des pixels de bruit) comme l'illustre la figure 3.7 [Zerrouki, 2014 (2)].

3.3. Utilisation des coefficients de curvelets comme caractéristiques

Comme nous l'avons déjà indiqué précédemment, l'extraction des caractéristiques reste une phase cruciale dans la procédure de classification, puisque les attributs extraits ont un impact direct sur la robustesse de la reconnaissance. Plusieurs caractéristiques ont été proposés et utilisées jusqu'à maintenant [Nguyen, 2015] et [Zerrouki, 2016(1)]. Cependant, la plupart de ces caractéristiques dépendent fortement de la taille et de la position de la silhouette humaine. La taille de la silhouette est relative à la variation de la distance entre les personnes et la caméra. Pour cela, une opération de normalisation de l'échelle avec un facteur de distance est fréquemment utilisée, mais cette opération nécessite un étalonnage préalable de la caméra. Pour cette raison, les coefficients de curvelets sont introduits pour caractériser les différentes formes de posture. Essentiellement, les coefficients de curvelets sont considérés comme des transformés multi-résolution. De plus, ils peuvent extraire des informations qui sont

habituellement manquantes avec d'autres transformées comme la directionnalité et l'anisotropie [Starck, 2002]. Il est bien connu que les coefficients de curvelets sont plus populaires dans les tâches de prétraitement d'images telles que: réduction de bruits (*image enhancement*), compression et restauration, etc [Haralick, 2014]. Mais au cours des dernières années, les transformées de curvelets sont exploitées dans la résolution des problèmes de reconnaissance, tel que le caractère [Majumdar, 2007] ou la reconnaissance du visage [Ucar, 2016].

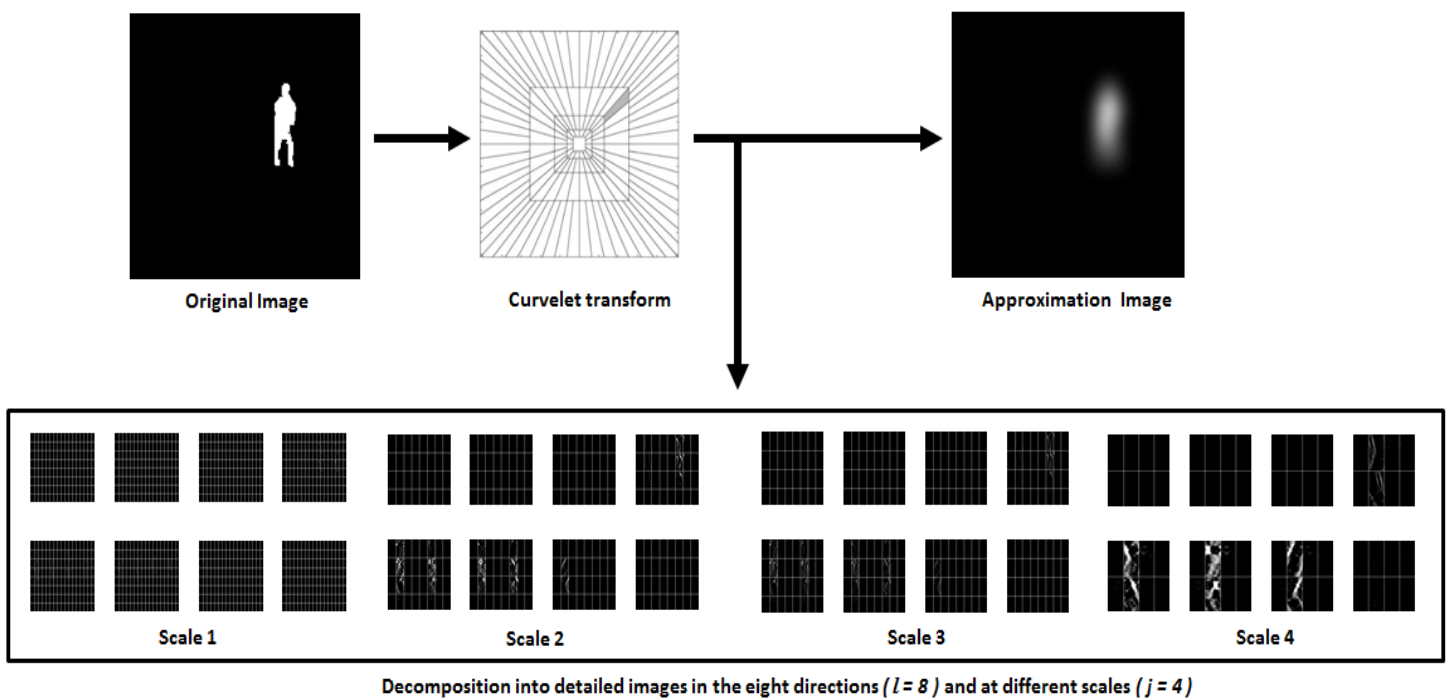


Figure. 3.8 Exemple de décomposition en curvelet d'une image originale (image binaire représentant le corps humain dans la position debout) en une image d'approximation suivie par les images de détails dans les 8 directions [Zerrouki, 2017 (1)].

Dans notre travail, la transformation de curvelets est appliquée sur les images binaires (représentant uniquement la silhouette du corps humain, extraite pendant la phase de segmentation). Contrairement aux transformations des ondelettes [Hiremath, 2006] ou Gabor [Malathi, 2016] qui sont plus adaptées aux images en niveaux de gris, les curvelets offrent une meilleure information directionnelle et constituent une représentation appropriée pour les contours et les singularités. Une autre motivation pour l'utilisation des coefficients de curvelets en tant qu'attributs est liée à leurs capacités à capturer des informations structurales (singularité) dans le domaine fréquentiel à l'aide de multiples directions radiales. Pour un facteur d'échelle j , les curvelets fournissent une représentation creuse (*sparse representation*)

$O(2^{j/2})$ des images par rapport aux ondelettes $O(2^j)$ [Starck, 2002]. La fonction curvelet appliquée sur une image I peut être définie comme suit:

$$C(I) = 2^{j/2} C(D_j R_l I - K_t) \quad (3.9)$$

Ou D_j , R_l et K_t représentent respectivement la matrice de mise à l'échelle, l'orientation et les paramètres de translation. L'image binaire de la silhouette humaine est tout d'abord décomposée puis divisée en fractions (*wedges*) dans le domaine fréquentiel [Starck, 2002].

La forme parabolique des fractions est le résultat du partitionnement en cercles concentriques et en divisions angulaires. Donc la décomposition de l'image est non seulement (i) en multiples échelles j ou appelés aussi sous-bandes, mais aussi (ii) en des divisions angulaires (en différents angles) ou appelées aussi orientations l . Dans ce travail, nous avons appliqué des transformées de curvelets à l'aide de quatre niveaux d'échelle ($j = 4$) et avec huit directions ($l = 8$).

Ceci produit d'une part l'image d'approximation qui contient les composantes basses fréquences et d'une autre part les coefficients de détails représentant les détails hautes fréquences selon les différentes orientations. La figure 3.8 montre un exemple de décomposition en curvelets d'une image binaire représentant le corps humain dans la position debout. Il est à noter que les coefficients de curvelets sont utilisés dans la reconnaissance d'objet dans une image, quel que soit sa position ou son orientation. D'où leurs invariance par rapport à la translation, l'échelle et la rotation. Cependant, l'invariance des transformations de curvelets à la rotation présente une certaine limitation par rapport à l'identification de postures. Car l'information liée à la rotation de la posture dans l'image est plus que nécessaire dans la phase d'identification, où l'invariance à la rotation peut provoquer une confusion parmi certaines postures qui sont principalement discriminées par des informations de rotation, telle que la position debout et allongé. Pour résoudre ce problème, un complément de cinq zones d'occupation partielles du corps a été proposé en tant que caractéristiques. La partie 3.4 sera consacrée à la description de ces rapports de surface et leur utilisation comme attributs de postures.

3.4. Proposition des rapports de surface comme caractéristiques

Nous rappelons que les caractéristiques extraites doivent décrire le mieux la silhouette humaine quel que soit sa position et sa taille dans l'image. Ces changements sont relatifs à la mobilité du corps dans la scène et à son positionnement par rapport à la caméra.

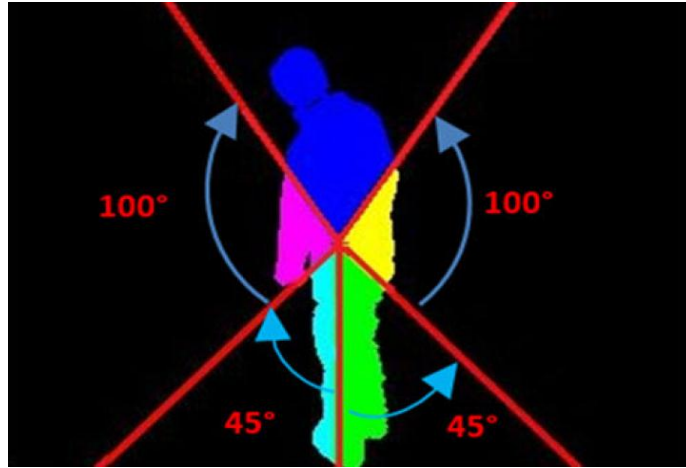


Figure. 3.9 Partitionnement du corps humain en cinq zones d'occupation partielles

Nous avons abordé dans la section 3.1 plusieurs caractéristiques (proposées dans la littérature) basées sur les informations de forme pour détecter et classer les chutes humaines telles que le centre de gravité, l'axe horizontal et vertical du rectangle circonscrit, etc. Cependant, ces caractéristiques présentent des limitations à séparer quelques cas de postures, en particulier lorsqu'il existe un degré élevé de similarité (confusions dans les dimensions et les orientations de l'ellipse des positions accroupi et assise).

Pour cette raison, au lieu d'utiliser la forme géométrique de la silhouette, nous proposons des caractéristiques extraites par rapport aux pixels constituant le corps humain (pixels non nuls après la phase de segmentation). Plus précisément, nous utilisons cinq zones d'occupation partielles (c'est-à-dire le nombre de pixels dans chaque zone) du corps pour détecter et classer les chutes. Ces zones correspondent spécifiquement aux différents membres du corps dans la position debout, à savoir la tête, les bras et les jambes comme le montre la figure 3.9. Ces zones ont été déterminées en utilisant le centre de gravité du corps (X_G, Y_G), qui est simplement le barycentre des pixels.

$$X_G = \frac{1}{N} \sum_{i=1}^N x_i, \quad Y_G = \frac{1}{N} \sum_{i=1}^N y_i \quad (3.10)$$

ou N est le nombre de pixels représentant le corps humain et x_i et y_i désignent respectivement les coordonnées horizontales et verticales des pixels associés au corps humain.

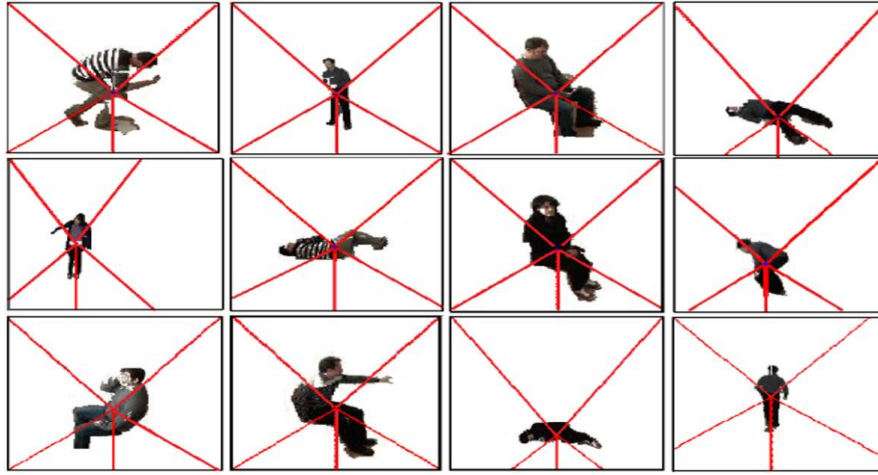


Figure. 3.10 Exemple d'utilisation des rapports de surface comme caractéristiques de postures

Le partitionnement s'effectue en traçant cinq segments à partir du centre de gravité du corps comme le montre la figure 3.9. Le premier segment est un segment vertical, les deuxième et troisième segments sont situés à 45° de part et d'autre du segment vertical tandis que le quatrième et le cinquième segment se situent à 100° de chaque côté du troisième et quatrième segment respectivement (voir figure 3.9). Le centre de gravité et les cinq zones sont calculés pour chaque image afin de représenter les caractéristiques correspondantes que nous estimons être une caractérisation précise pour la classification des gestes humains.

Pour maintenir la propriété de l'invariance par rapport à l'échelle en utilisant ces zones d'occupation, nous utilisons plutôt des surfaces normalisées en divisant chaque surface partielle par la surface globale du corps. Supposant la surface globale du corps humain A et les cinq surfaces partielles $A_i ; i = 1, \dots, 5$, les zones partielles normalisées sont alors définies par les rapports suivants :

$$R_i = \frac{A_i}{\sum_{i=1}^5 A_i} \quad (3.11)$$

Ces rapports de surface sont aussi utilisés comme caractéristiques (en plus des coefficients des curvelets) dans le processus de détection de chutes. Il est à noter que ces rapports sont non seulement invariants à la translation et le paramètre d'échelle, mais surtout prennent en considération l'information de rotation nécessaire à l'identification des postures (d'où l'utilité pour la détection des chutes). Par ailleurs, ces rapports de surfaces sont suffisamment représentatifs et descriptifs de la posture humaine et pas trop complexes afin d'assurer une simplicité et une rapidité dans les traitements et les calculs. Ces rapports de surface sont ensuite ajoutés aux coefficients de curvelets pour former l'ensemble du vecteur des

caractéristiques. La figure 3.10 illustre quelques exemples d'utilisation des rapports de surface comme caractéristiques de postures (assise, debout accroupi, etc.).

3.5. Réduction de la dimension en utilisant l'algorithme à évolution différentielle (differential evolution algorithm)

Comme nous l'avons déjà souligné dans la partie 3.4 les rapports de surface sont additionnés avec les coefficients des curvelets pour construire le vecteur des caractéristiques. Ce vecteur sera utilisé dans un premier lieu dans la classification de postures, ou chaque image sera classifiée individuellement et indépendamment de la séquence d'images. Dans un second lieu, puisque les images d'une vidéo sont assimilées à une séquence d'observations, l'ensemble des vecteurs caractéristiques calculés pour chaque image sont concaténés afin de former le vecteur caractéristique global correspondant à la séquence vidéo. D'ailleurs, la détection de chutes s'applique sur les séquences vidéo, comme la chute humaine est définie par un enchainement de postures. Cependant, le désagrément lié à cette caractérisation réside dans la taille importante du vecteur résultant, surtout quand il s'agit des séquences d'images et non pas chaque image séparément.

Pour remédier à cette limitation, une phase de sélection de caractéristiques est donc appliquée. L'objectif de cette phase est de réaliser un nouveau vecteur caractéristique avec une dimension réduite, en éliminant les caractéristiques inutiles, redondantes ou corrélées. Ce problème est abordé en utilisant l'algorithme d'évolution différentielle (Differential Evolution DE) pour la sélection de caractéristiques. Les caractéristiques sélectionnées permettront par la suite une réduction considérable du temps de calcul du système ce qui favorise son application en temps réel.

L'algorithme de DE a été initialement introduit par Storn et Price en 1995 [Storn, 1997] et il a été adapté dans divers domaines tel que l'optimisation et la réduction de la dimension [Das, 2011]. Dans ce travail, DE est utilisé comme un algorithme de sélection de caractéristiques appliqué aux coefficients de curvelets afin de réduire les coefficients redondants. Initialement, les coefficients de curvelets sont tirés ou classés d'une manière aléatoire. Le processus itératif utilise par la suite des opérations de mutation, de croisement et de sélection, dans la création du nouveau vecteur de caractéristiques.

- **Mutation**

Lors de l'opération de mutation, l'algorithme DE génère un nouvel élément (appelé élément mutant V_i) en combinant les caractéristiques originales X_i , ou la différence pondérée entre

deux caractéristiques à savoir X_1 et X_2 est ajouté à une troisième caractéristique originale X_3 :

$$V_i = X_3 + F (X_1 - X_2) \quad (3.12)$$

Où X_1 , X_2 et X_3 sont choisis aléatoirement parmi les caractéristiques d'origine pour créer l'élément mutant V_i ; et $F \in (0,1)$ représente l'élément d'échelle qui teste le taux de variations entre les caractéristiques.

- **Croisement**

Généralement dans les algorithmes évolutionnaires, l'opération de croisement combine des caractéristiques de différentes sources. Dans le cas de l'algorithme d'évolution différentielle, puisque la mutation est déjà basée sur une recombinaison des caractéristiques initiales X_i , le rôle du croisement est quelque peu différent. Ce croisement permet la sélection entre l'élément initial X_i et celui obtenu par la mutation V_i . Cette opération peut être assurée soit par une mixture des éléments, comme il est le cas pour le croisement binominal et exponentiel, ou par une recombinaison arithmétique entre les éléments initiaux et mutants. Dans notre cas nous avons utilisé le croisement binomial, les éléments d'essais U_i (Trial features) sont obtenus comme suit :

$$U_i = \begin{cases} V_i & \text{si } z_i(0,1) < CR \\ X_i & \text{sinon} \end{cases} \quad (3.13)$$

Où z_i est la valeur aléatoire uniformément distribuée dans $[0, 1]$ et $CR \in [0, 1]$ est un paramètre qui contrôle le nombre de composants pris dans le vecteur mutant V_i et il est appelé aussi le taux de croisement. Le nouveau vecteur des caractéristiques U_i correspond soit à la caractéristique mutante V_i (représentant la version corrigée par les mutations), soit à la caractéristique initiale X_i en fonction de la meilleure représentativité.

La phase suivante de l'algorithme DE consiste à classer les caractéristiques U_i suivant leurs ordre d'importance, cela en fonction des valeurs de poids attribuées. L'estimation de ces poids est basée principalement sur les performances des résultats de classification, puisque les caractéristiques sélectionnées dépendent essentiellement de l'application (classification de postures et détection de chutes). Autrement dit, le changement de l'application entraîne forcément un changement dans le classement des caractéristiques. Pour cela, nous avons utilisé un schéma de pondération de la roulette.

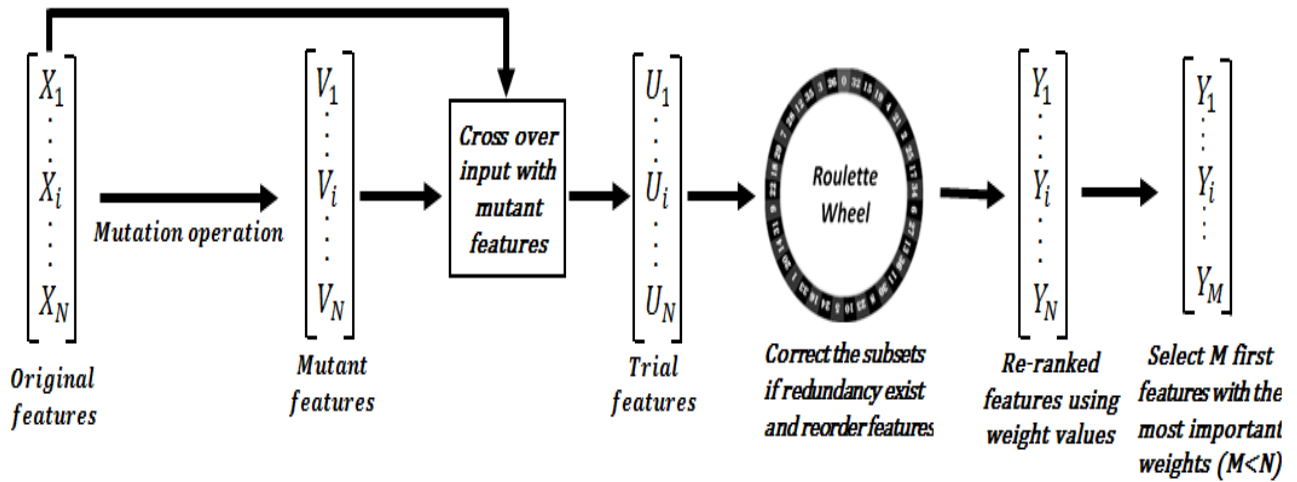


Figure. 3.11 Technique de sélection de caractéristiques en utilisant l'algorithme d'évolution différentielle

Le processus commence avec tous les éléments U_i , puis procède par des éliminations itératives des caractéristiques afin de sélectionner les caractéristiques les plus discriminantes. Les poids des caractéristiques sont alors calculés à partir des facteurs de distribution associés à chaque caractéristique (Obtenus par la roulette selon les facteurs de distribution les plus élevés). Après avoir classé l'ensemble des caractéristiques $Y_i \ i = \{1, 2, \dots, N\}$ suivant l'ordre d'importance, la dernière étape de l'algorithme DE est de conserver uniquement les M premier éléments ayant les poids les plus importants $Y_i \ i = \{1, 2, \dots, M\}$ avec $M < N$.

3.6. Conclusion

Nous avons présenté dans ce chapitre les différentes caractéristiques utilisées dans ce travail pour la représentation de la silhouette humaine, à savoir : les valeurs singulières tronquées, les coefficients de curvelets et les cinq rapports de surfaces. Nous avons aussi présenté la technique de sélection de caractéristiques ou de réduction de dimension en utilisant l'algorithme à évolution différentielle (differential evolution algorithm). Le chapitre suivant est consacré à la classification de postures humaines, où chaque image est représentée par les caractéristiques sélectionnées.

Chapitre 4 :

Classification de postures humaines

4.1. Classification statique : image par image	39
4.2. Classification de postures en utilisant les réseaux de neurones	39
4.3. Classification de postures en utilisant les Séparateurs à Vaste Marge (Support Vector Machines SVM)	43
4.3.1. Utilisation des SVMs pour des données linéairement séparables	43
4.3.2. Utilisation des SVMs pour des données non-linéairement séparables	45
4.3.3. SVM multi-classes en utilisant l'approche "un contre tous"	47
4.3.4. SVM multi-classes en utilisant l'approche "un contre un"	47
4.4. Identification de postures en utilisant la classification Adaboost	47
4.5. Conclusion	50

Après la phase d'extraction des caractéristiques, ce chapitre est consacré à la phase de classification de postures humaines dans une scène. Cette classification est appelée statique car l'identification de la posture est basée sur chaque image séparément et non pas la séquence d'images (classification image par image). Comparée à la détection ou au suivi de personnes dans une scène, la reconnaissance des comportements reste encore un axe de recherche à explorer et présente un intérêt très important dans les axes de recherches actuels. En effet, la reconnaissance de comportements est une tâche complexe. Les activités des personnes sont parfois si diverses qu'il n'est pas évident de séparer un événement parmi d'autres. Dans ce travail plusieurs algorithmes de reconnaissance de forme et de machine d'apprentissage ont été appliqués. Les machines d'apprentissage ou méthodes de classification consistent à distinguer les postures en fonction de leurs caractéristiques (ou le vecteur caractéristique représente les données d'entrées du classifieur). Il existe différentes catégories de classification, à savoir classification non-supervisée, supervisée et semi-supervisée, etc. Les méthodes de classification les plus utilisées dans la littérature sont les méthodes supervisées. L'idée de base de la classification supervisée est d'apprendre les règles de séparation à partir d'un ensemble de données dont le classement est déjà connu. Une fois les règles apprises, l'algorithme va l'appliquer pour des nouvelles données, dont les classes correspondantes sont inconnues.

Nous allons aborder en détails dans ce chapitre, les différentes méthodes de classification utilisées pour l'identification de postures, à savoir Adaboost, les réseaux de neurones et les machines à vecteurs supports SVM.

4.1. Classification statique: image par image

Les méthodes de reconnaissance statique (ou image par image) sont généralement basées sur des comparaisons entre l'image courante et des informations préalablement enregistrées. La reconnaissance de posture est une application type, où l'on cherche à reconnaître la posture d'une personne à partir d'éléments connus ou d'informations extraites des images précédentes. Plusieurs méthodes ont été proposées dans la littérature, en utilisant soit les approches classiques soit celles basées sur les machines d'apprentissage (apprentissage automatique). Les approches classiques utilisent généralement des méthodes de reconnaissances simples qui ne nécessitent pas forcément de modèle mathématique, parmi ces méthodes on peut citer la classification par seuillage ou par calcul de distances. Les approches de classification en utilisant les machines d'apprentissage sont des méthodes basées généralement sur un fondement mathématique, permettent la résolution des problèmes plus complexes tels que les problèmes non-linéaires. Il existe plusieurs catégories d'algorithmes de machine d'apprentissage, à savoir les méthodes non-supervisées, les méthodes supervisées, semi-supervisées, etc [Duda, 2012]. Dans le présent travail, nous avons utilisé plusieurs méthodes de classification supervisées pour la reconnaissance de posture, à savoir Adaboost, les réseaux de neurones et les SVMs. Ces méthodes supervisées comprennent deux principales étapes: (i) en premier lieu, affecter à l'algorithme certaines postures avec leurs classes correspondantes comme échantillons d'apprentissage et (ii) par la suite attribuer le reste des postures en fonction de leurs caractéristiques via le modèle construit dans la première phase. Dans les parties suivantes, nous évoquerons plus de détails sur chaque méthode de classification, la phase d'apprentissage, la construction du modèle et l'estimation de ces paramètres, l'affectation des nouveaux échantillons, etc. [Duda, 2012].

4.2. Classification de postures en utilisant les réseaux de neurones

Dans un premier lieu, nous avons appliqué les réseaux de neurones comme algorithme de classification afin de distinguer entre les différentes postures. La classification basée sur les réseaux de neurones, comme son nom l'indique, est une méthode qui s'inspire du fonctionnement du cerveau humain. Pour cela, un réseau est constitué généralement d'un nombre de cellules ou de neurones (nodes) interconnectées. De nombreuses variantes sont

définies dans la littérature, selon le choix de la cellule élémentaire, de l'architecture du réseau et de la dynamique du réseau, etc. Une cellule élémentaire peut manipuler des valeurs soit binaires ou réelles et différentes fonctions peuvent être utilisées dans le calcul des sorties du réseau de neurones. Plusieurs modèles de réseaux de neurones ont été proposés, tels que le model de Hopfield, le modèle du Perceptron multi-couches (PMC) qui est surtout connu par l'appellation Multi layer perceptron (MLP), etc.

La structure MLP dans les réseaux de neurones est la structure la plus utilisée durant ces dernières années, pour laquelle de nombreux algorithmes d'apprentissage ont été développés. Les réseaux MLP sont des modèles non linéaires composés d'un nombre de nœuds organisés en plusieurs couches. Ces réseaux comptent suffisamment de nœuds cachés (présentés dans des couches cachées) afin de gérer les quantités de données à classifier. Il a été rapporté par E. Alpaydin dans son célèbre livre [Alpaydin, 2014], que les réseaux MLP peuvent faire une approximation de n'importe quelle fonction à une certaine précision souhaitée. En d'autres termes, les réseaux MLP sont considérés comme des fonctions d'approximations universelles [Alpaydin, 2014], [Duda, 2012].

- **Perceptron multicouches :**

La structure MLP d'un réseau de neurone est composée d'un réseau de nœuds disposés en plusieurs couches comme l'illustre la figure 4. 1. En règle générale, un réseau MLP nécessite trois couches, ou plus, de nœuds de traitement, à savoir: une couche d'entrée qui reçoit les variables d'entrée (dans le cas présent, ce sont les caractéristiques extraites dans la phase précédente), une ou plusieurs couches cachées et une couche de sortie (en attribuant souvent chaque nœud de la couche de sortie à une classe).

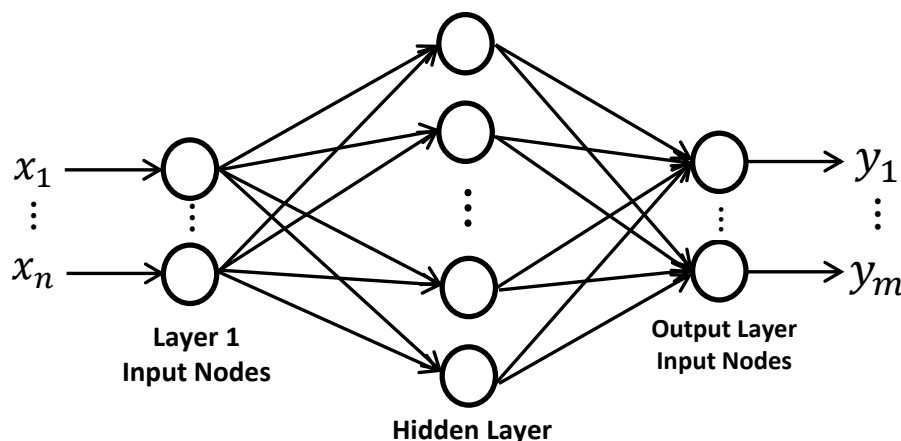


Figure 4.1 Exemple d'architecture globale d'un réseau de neurones

Le perceptron est l'élément de traitement de base. Il peut être utilisé en entrée, en sortie du réseau comme il peut aussi être un élément à l'intérieur du réseau (perceptron intermédiaire). Le fonctionnement d'un perceptron est principalement lié à sa fonction d'activation qui représente le seuil de stimulation du perceptron (inspiré du fonctionnement biologique). Il existe plusieurs types de fonctions d'activation, tel que la fonction sigmoïde, linéaire, gaussienne, etc. le perceptron est aussi caractérisé par ses liens soit en entrées x_i ou en sorties y_i .

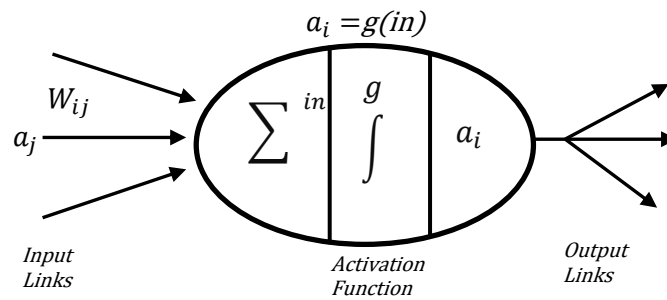


Figure 4.2 Structure générale du perceptron

Chaque entrée x_i , est pondérée par un poids de connexion, ou un poids synaptique $w_i \in R$. De la manière la plus simple, la sortie, y_i ne peut-être que la somme pondérée des entrées (voir figure 4.2):

$$y_j = \sum w_i x_i + w_0$$

w_0 est la valeur du biais, ou valeur d'interception afin de rendre le modèle plus général.

Comme nous l'avons déjà évoqué, le rôle d'un neurone ou perceptron est de fournir une réponse y_i , la valeur de cette sortie est comprise entre 0 et 1. Nous utilisons pour cela une fonction $g : R \rightarrow [0, 1]$, appelée fonction d'activation. Lorsque la réponse est proche de 1, alors le neurone est considéré comme actif et pour une réponse proche de 0, le neurone sera donc dit inactif.

D'une manière plus générale la sortie du neurone peut être alors définie par :

$$y_j = g(\sum w_i x_i + w_0) \quad (4.1)$$

Le principe du fonctionnement d'un réseau de neurones est que lorsque les données (ou les caractéristiques) sont présentées à la couche d'entrée (x_i), les nœuds de réseau effectuent des

calculs dans les couches successives jusqu'à ce qu'une valeur de sortie soit calculée à chacun des nœuds de sortie (y_i). Ce signal de sortie devrait en principe fournir la classe appropriée pour les données d'entrées, c'est-à-dire que nous prévoyons avoir une valeur de sortie élevée sur le nœud de la classe correspondante et des faibles valeurs de sortie sur les autres nœuds (qui correspondent au reste des classes). Pour parvenir à cette affectation correcte, le réseau de neurones (MLP) doit d'abord passer par une phase d'apprentissage supervisée en utilisant l'algorithme itératif de rétro-propagation.

Dans cette phase d'apprentissage, un ensemble d'échantillons de vecteurs caractéristiques, appelés ensembles d'entraînement est présenté au réseau de neurone, ou chaque échantillon d'entraînement lui est affecté la sortie désirée correspondante (situé au nœud correspondant). Par la suite, le processus commence par une affectation des poids. Les valeurs initiales des poids sont de faibles valeurs et surtout aléatoires. La phase cruciale durant l'entraînement d'un réseau de neurones consiste à fixer les poids et cela en fonction des échantillons d'entraînements et leurs résultats désirés. L'ajustement de ces poids se fait itérativement par un processus appelé rétro-propagation. Le principe consiste à ramener au maximum la valeur de sortie du réseau de neurones (obtenue) vers la valeur désirée, en modifiant les différents paramètres du réseau (dans le cas présent, c'est les poids). Pour cela, une fonction d'erreur E entre les sorties obtenue et désirée est définie ; ou les poids sont ajustés afin de minimiser cette mesure d'erreur :

$$E(w_{ij}) = 1/2 \sum_{i=1}^m (Y_{désirée} - Y_{obtenue})^2 \quad (4.2)$$

La règle d'apprentissage de la rétrogradation est généralement basée sur la descente en gradient :

$$\Delta w_{ij} = -\rho \frac{\partial E}{\partial w_{ij}} \quad (4.3)$$

Ou ρ est un facteur d'apprentissage qui indique la taille relative de la variation des poids. Les poids sont par la suite modifiés dans la direction qui réduira l'erreur:

$$w_{ij}(t + 1) = w_{ij}(t) + \Delta w_{ij}(t) \quad (4.4)$$

Ou t représente le nombre d'itération. La pertinence de l'algorithme de rétro-propagation réside dans la simplicité de fonctionnement du système itératif. Il est à noter que le cas idéal correspondant à une erreur nulle est quasi impossible dans les cas réels, d'où la définition d'un critère d'arrêt, appliqué soit sur l'erreur, ou sur le nombre d'itération.

4.3. Classification de postures en utilisant les Séparateur à Vaste Marge (Support Vectors Machines SVM)

Comme deuxième méthode de classification nous avons utilisé les Machines à vecteurs supports SVM afin de classer les différentes postures. Les SVMs correspondent à une méthode de classification supervisée relativement récente, introduite dans les débuts des années 90's par Vladimir Vapnik [Vapnik, 1995] et qui a donné de bonnes performances dans un large éventail d'applications pratiques.

Dans cette section, nous expliquons brièvement son principe de fonctionnement ; nous présentons d'abord le cas le plus simple où les données sont linéairement séparables, puis le cas général où les données peuvent être linéairement non séparables.

4.3.1. Utilisation des SVMs pour des données linéairement séparables

L'introduction des SVMs comme méthode de classification avait pour but de résoudre les problèmes de discrimination à deux classes. En réalité, il est difficile de se concentrer uniquement sur les problèmes de classifications bi-classes. Beaucoup de problématiques réelles présentent une multitude de classes telles que la classification de postures. Pour cela, plusieurs méthodes ont été suggérées pour étendre l'application des SVM aux problèmes de discrimination à plus de deux classes. Nous nous concentrerons dans un premier lieu sur le principe de classification des SVMs à deux classes. Supposant un vecteur caractéristique (représentant une posture donnée) X de dimension n qui est en même temps l'entrée du classifieur SVM. La classe à laquelle appartient l'individu est représentée par $y \in \{-1, 1\}$; ou une des classes possible est représentée soit par -1 soit par 1.

Comme toute méthode de classification supervisée, les SVMs utilisent un ensemble de données d'apprentissage pour lesquelles les classes sont déjà connues et s'en servent pour construire les règles de décision. Supposons que nous disposons d'un ensemble d'apprentissage de l données de la forme $(x_i; y_i) \in R^n \times \{-1, 1\}$ ($i = 1, \dots, l$).

L'idée la plus basique des SVMs dans la classification est d'utiliser un hyperplan de séparation afin de distinguer les échantillons d'entraînement des différentes classes. Pour le cas des données linéairement séparables, cela signifie qu'il existe un hyperplan dans R^n tel que toutes les données appartenant à la classe 1 et -1 se retrouvent de part et d'autre de l'hyperplan vérifiant l'équation : $w \cdot x + b = 0$; les données appartenant à la classe +1 doivent vérifier l'inégalité $w \cdot x + b > 0$, alors que les échantillons appartenant à la classe -1 doivent vérifier l'inégalité $w \cdot x + b < 0$.

Avec $w = (w_1, \dots, w_n) \in \mathbb{R}^n$ le vecteur des coefficients de l'hyperplan et $b \in \mathbb{R}$ un scalaire appelé le biais. La définition de l'hyperplan optimale, revient à résoudre la fonction indicatrice suivante : $y = classe(x)$

$$y = signe(w.x + b) \quad (4.5)$$

$$signe(w.x + b) = \begin{cases} -1 & \text{Si } w.x + b < 0 \\ 0 & \text{Si } w.x + b = 0 \\ +1 & \text{Si } w.x + b > 0 \end{cases} \quad (4.6)$$

Cependant, dans le cas des données linéairement séparables, la résolution de la fonction indicatrice dans la séparation des deux classes peut avoir une infinité d'hyperplans. L'idée des SVMs est de choisir le meilleur hyperplan de séparation, en vérifiant la condition de la marge de séparation. Cette marge M est définie par rapport à la distance euclidienne $dist$ entre l'échantillon x et l'hyperplan:

$$M = \min\{dist(x_i, w, b) : i = 1, \dots, l\} \quad (4.7)$$

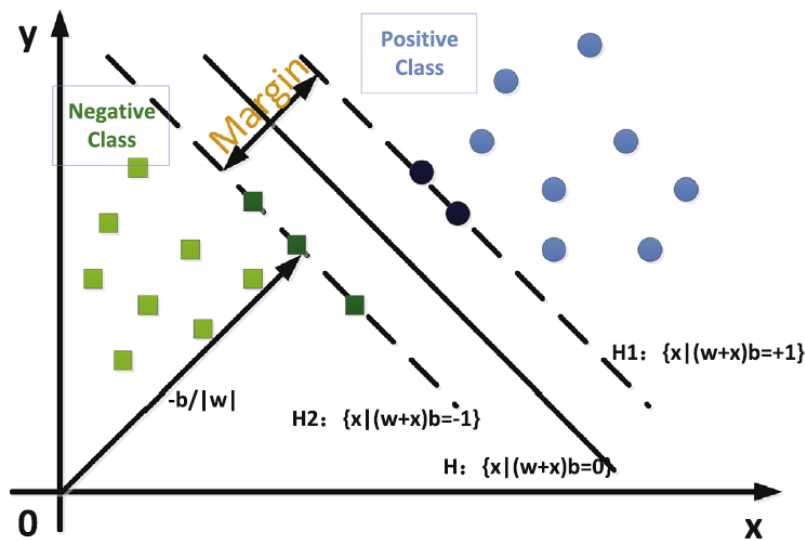


Figure 4.3 Schéma de l'algorithme de définition des hyperplans de séparation.

Un hyperplan optimal séparant les données doit s'écrire sous la forme canonique, ce qui va diminuer le nombre d'hyperplans (infinité d'hyperplan qui vérifient la condition de séparation). En considérant l'hyperplan canonique, l'hyperplan séparant les données devient unique correspondant à une marge maximale. Autrement dit, le vecteur des coefficients de l'hyperplan sera exprimé comme étant le point qui minimise une fonction strictement convexe

(rappelons que les fonctions strictement convexes n'ont qu'un unique minimum global). Pour cela, l'hyperplan optimal doit vérifier la condition suivante :

$$\min\{dist(w \cdot x_i + b) : i = 1, \dots, l\} = 1 \quad (4.8)$$

Les seuls points où $y_i(w \cdot x_i + b) = 1$ sont ceux qui sont sur les deux droites de la marge et correspondent à la distance maximal (c'est les éléments les plus proches, mais de classes différentes). Ces points sont appelés les vecteurs supports (voir figure 4.3).

4.3.2. Utilisation des SVMs pour des données non-linéairement séparables

Dans la section précédente (4.3.1), nous avons présenté le premier cas où les SVMs permettent la classification des données lorsque celles-ci sont linéairement séparables. Cependant, il existe de nombreux cas de données (telles que les caractéristiques extraites pour la classification de posture) pour lesquels il est impossible de les séparer avec un hyperplan. Pour remédier à ce problème, les SVMs appliquent une transformation sur les données de sorte qu'une fois transformées, elles soient linéairement ou quasi linéairement séparables. L'espace où se trouvent les données avant d'être transformées est appelé l'espace d'entrée (input space), alors qu'après transformation, les données se trouvent dans ce qu'on appelle l'espace de redescription (ou feature space). Il suffit alors de trouver l'hyperplan dans l'espace de redescription qui sépare le mieux ces données transformées. La fonction de transformation est définie comme suit :

$$\begin{aligned} \Phi : R^n &\rightarrow R^r \\ x &\rightarrow \Phi(x) \end{aligned}$$

Où r et n représentent les dimensions des espaces de redescription et d'entrée respectivement, avec $r > n$, ce qui signifie que la transformation est une projection dans un espace de dimension supérieure afin de pouvoir les séparer linéairement.

Afin de trouver l'hyperplan séparateur optimal, on procède de la même manière que précédemment (le cas linéairement séparable section 4.3.1) et cela revient à résoudre l'équation suivante:

$$y_i(w \cdot \Phi(x) + b) = 1 \quad (4.9)$$

La fonction indicatrice associée à ce problème est alors définie comme suit :

$$Classe(x) = \text{signe}(\sum_{i=1}^l (y_i \Phi(x_i) \cdot \Phi(x)) + b) \quad (4.10)$$

Si la transformation utilisée est appropriée, la résolution du problème permet de trouver un séparateur non linéaire avec la marge la plus grande possible, en utilisant ainsi les vecteurs du

support dans le cas des données non-linéairement séparables. Toutefois, l'utilisation des transformations pose certains problèmes car les calculs se font avec les données transformées, c'est-à-dire dans l'espace de redescription. Or, comme la dimension de cet espace est bien souvent beaucoup plus grande que celle de l'espace d'entrée, ces calculs peuvent devenir extrêmement longs et difficile à effectuer. Ce qui amène à l'utilisation des fonctions noyaux K (kernels) :

$$K: R^n \times R^n \rightarrow R$$

$$(x_i, x_j) \rightarrow \phi(x_i) \phi(x_j)$$

Le principe de base de l'utilisation des noyaux est de prendre en entrée deux points (représentant deux échantillons d'entraînement) dans l'espace d'entrée et calculer le produit scalaire dans l'espace de redescription. L'avantage des noyaux est que le calcul peut se faire directement à partir des données de l'espace d'entrée. Grâce au concept de noyau, la fonction indicatrice peut alors s'écrire de la manière suivante :

$$Classe(x) = \text{signe}(\sum_{i=1}^l (y_i K(x_i, x)) + b) \quad (4.11)$$

Lorsque la fonction noyau est connue, la transformation $\phi(x)$ n'apparaît nulle part, il n'est donc pas nécessaire d'effectuer la transformation sur les données dans le calcul ni se préoccuper de la dimension de l'espace de redescription.

Plusieurs types de noyaux sont utilisés, cela dépend essentiellement de l'application et des données. Les noyaux les plus fréquemment utilisés sont :

- Linéaire : $K(x_i, x_j) = (x_i \cdot x_j)$
- Polynomial : $K(x_i, x_j) = (x_i \cdot x_j + 1)^d$
- Gaussien (Radial Basis Function RBF) : $K(x_i, x_j) = e^{-\frac{\|x_i - x_j\|^2}{2\sigma^2}}$
- Multi-quadratique inverse : $K(x_i, x_j) = \frac{1}{\sqrt{(x_i \cdot x_j)(x_i \cdot x_j) + \beta}}$

Nous avons présenté jusqu'à présent le principe de séparation des SVMs entre deux classes. Il est à noter qu'initialement les SVMs ont été conçues pour résoudre des problèmes binaires ou bi-classes. Cependant, les problèmes du monde réel exigent souvent la discrimination entre plusieurs classes. Pour remédier à cette limitation, les SVMs décomposent les problèmes de classification multi-classes en une série de problèmes bi-classe, de sorte que les SVM standard puissent être directement appliquées. Deux approches de classification SVM multi-classes sont connues et utilisées, à savoir l'approche "un contre tous" ou "un contre un".

4.3.3. SVM multi-classes en utilisant l'approche "un contre tous"

L'approche un contre tous construit N classificateurs binaires distincts afin de classifier N classes. Le principe de cette approche est d'utiliser un classifieur binaire pour une classe donnée comme classe positive +1 et le reste des classes ($N - 1$ restantes) comme classes négatives -1. La même opération est alors reconduite avec chaque classe séparément jusqu'à balayage de toutes les classes.

4.3.4. SVM multi-classes en utilisant l'approche "un contre un"

La deuxième approche de séparation multi-classes est la classification un contre un ou par paire. Cette méthode construit $N(N - 1)/2$ classificateurs binaires afin de classifier les N classes et évalue toutes les combinaisons possibles. Durant la phase des tests, l'affectation de la classe correspondante à nouvel échantillon (une posture de test) est basée sur les estimations des différents SVMs, ou l'affectation la plus probable lui sera affectée.

4.4. Identification de postures en utilisant la classification Adaboost

En plus des réseaux de neurones et les SVMs, nous avons appliqué aussi une méthode de classification en cascade à savoir l'algorithme adaboost (en utilisant les arbres de décision comme méthodes basiques de classification ou faibles classifieurs). La classification adaboost est une classification supervisée, où les caractéristiques de postures sont nécessaires durant la phase d'entraînement du classifieur (afin de construire le modèle). Une fois le modèle construit, le classifieur est appliqué sur l'ensemble de postures de test.

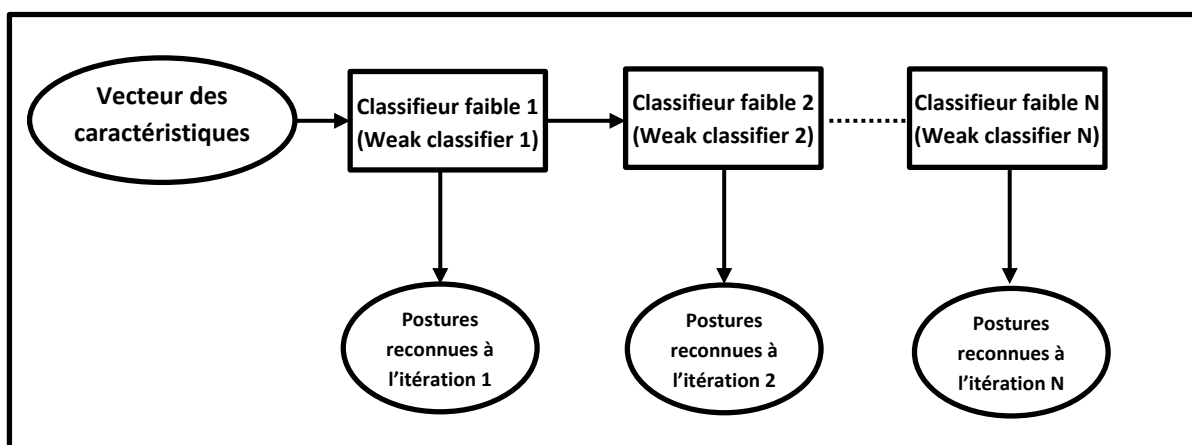


Figure 4.4 Schéma de l'algorithme adaboost en utilisant les arbres de décision.

La figure 4.4 illustre un exemple de classification par l'algorithme adaboost. Comme première réflexion (à partir du schéma du classifieur), une ressemblance avec les méthodes de classification par règles peut être observée. Cependant, il faut bien noter que la classification adaboost est très différente de celle basée sur des règles ; une phase d'apprentissage est appliquée pour chaque classifieur faible. La performance d'un classifieur faible est légèrement meilleure que celle obtenue par une classification aléatoire (correspondante à une performance de 0.5).

En d'autres termes, l'algorithme adaboost essaie de générer des classifieurs complémentaires en entraînant chaque nouveau faible classifieur uniquement sur les erreurs commises par le faible classifieur précédent. D'où le but principal de cette classification est de combiner plusieurs classificateurs faibles (weak classifiers) afin de construire au final un classifieur puissant. Les classifieurs faibles sont généralement sélectionnés en fonction de leur temps ou espace de complexité [Bourges, 2012]. La mise en cascade est un processus à plusieurs étapes, dans lequel le classifieur c_i est exécuté uniquement si **tous les classificateurs qui le précèdent**, c_k ($k < i$) ne sont pas suffisamment satisfaisants. Une valeur de poids w_i est alors incrémenté pour chaque échantillon mal classé, afin d'augmenter son influence durant la prochaine étape d'apprentissage. Le choix des faibles classifieurs c_i est basé sur la minimisation du taux d'erreurs de classification enregistré ε_i . Le résultat d'un faible classificateur doit être meilleur de celui obtenu par une classification aléatoire [Wang, 2004].

Les principales étapes de classification par l'algorithme AdaBoost sont résumées comme suit:

En entrée : soit un ensemble d'apprentissage $S = ((x_1, y_1), \dots, (x_m, y_m))$, ou x_i et y_i représente les échantillons de postures et leurs classes correspondantes ; m représente le nombre d'échantillons d'entraînement.

Initialisation : dans cette étape l'algorithme commence par

- Définir le nombre d'itérations T et,
- Initialiser tous les poids des échantillons de la première itération $w_1(i) = \frac{1}{m}$

Pour chaque itération ($t = 1, \dots, T$) :

- L'algorithme sélectionne un faible classifieur $C_t : R^d \rightarrow \{-1, +1\}$ et effectue une classification suivant la distribution des poids w_i .

- Calculer par la suite l'erreur de classification à chaque itération,

$$\varepsilon_t = \sum_{i=1}^N w_t(i) [Y_t \neq h_j(x)], \text{ et } C_t(x) = \text{Argmin } \varepsilon_t \quad (4.12)$$

Ou tout classifieur C_t correspondant à une erreur $\varepsilon_t > 0,5$ sera rejeté (le seuil de confiance est fixé à 0,5), ce qui rend le classifieur meilleur par rapport à une classification au hasard (correspondante à une erreur de 0.5).

- Augmenter les poids correspondant aux échantillons mal classés:

$$w_{t+1}(i) = \frac{w_t(i) \exp(-\alpha_t Y_t h_t(x_t))}{Z_t} \quad (4.13)$$

$$\text{Ou } Z_t \text{ représente le facteur de normalisation } Z_t = \sum_{i=1}^m w_t(i) \exp(-\alpha_t Y_t h_t(x_t)) \quad (4.14)$$

$$\text{et } \alpha_t = \frac{1}{2} \left(\ln \frac{1-\varepsilon_j}{\varepsilon_j} \right)$$

En sortie : l'algorithme combine les faibles classifieurs obtenus à chaque itération afin de construire le classifieur final H_t . Avec

$$H_T(x) = \text{sign}(\sum_{t=1}^T \alpha_i \times h_t(x)) \quad (4.15)$$

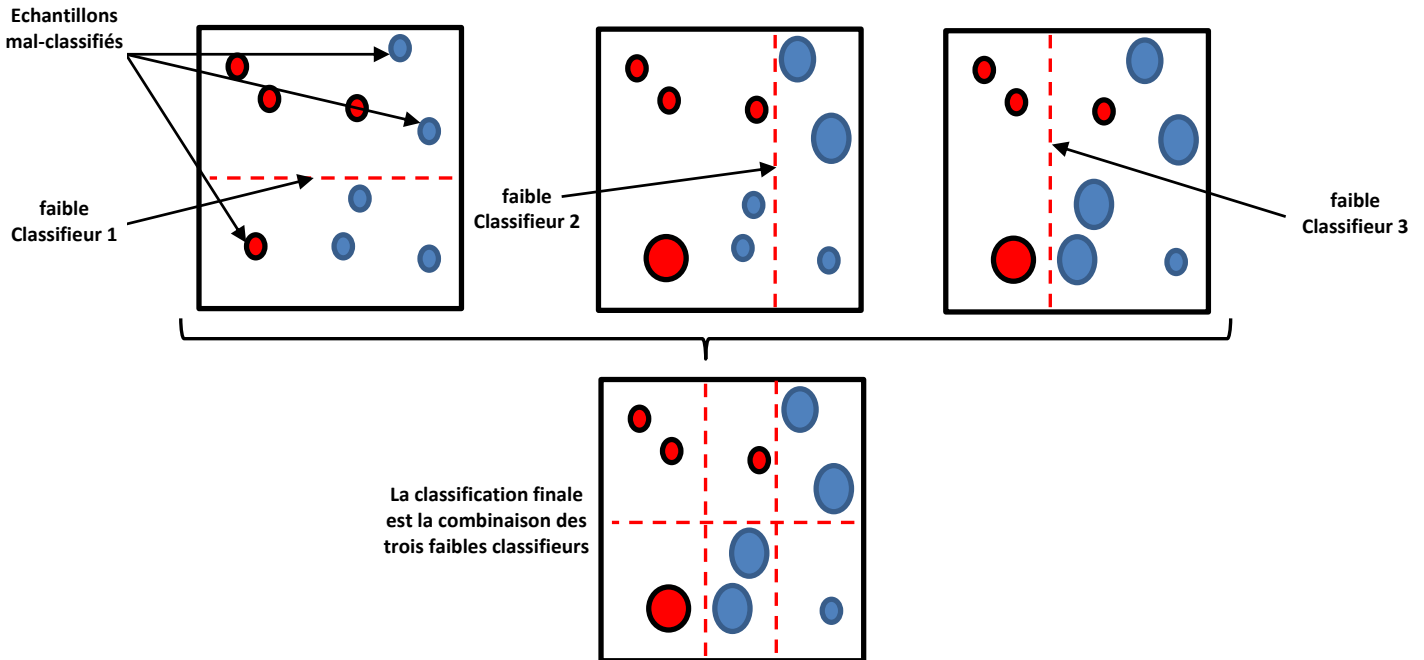


Figure 4.5 Exemple de classification AdaBoost

La figure 4.5 illustre un exemple de classification adaboost, ou le processus est répété trois fois correspondant à $T = 3$. On remarque que, les poids sont optimisés itérativement par

rapport aux échantillons mal classés et sont affectés de poids plus importants dans l'itération suivante. En plus, le faible classifieur est appliqué uniquement sur les échantillons mal-classifiés de l'itération précédente. Au final, on peut constater que les échantillons des deux classes sont correctement séparés.

Dans la procédure AdaBoost, de nombreux classifieurs faibles peuvent être combinés afin d'obtenir le classifieur final. Malgré l'utilisation des classifieurs faibles, la classification AdaBoost peut atteindre des performances élevées avec moins de paramètres à régler par rapport à d'autres méthodes de classification puissantes tels que les réseaux de neurones et les SVMs. L'implémentation d'un classifieur AdaBoost nécessite uniquement: (i) un classificateur faible; et (ii) le nombre de boosts utilisés durant la phase d'entraînement.

Nous avons appliqué le classifieur adaboost pour distinguer entre les différentes postures humaines, ou nous avons utilisé les arbres de décision comme faible algorithme de séparation. Il est à noter qu'initialement l'algorithme AdaBoost a été conçu pour résoudre des problèmes binaires. Afin de répondre aux problèmes multi-classes (classification de postures), nous nous sommes inspirés de l'approche "un contre tous" (One-Against-All) utilisée dans les SVM multi-classes (présentée dans la section 4.3.3).

Dans ce travail, les arbres de décision à un nœud, sont utilisés comme classifieurs faibles. L'idée principale est d'entraîner le premier (adaBoost) classifieur faible binaire sur les données de la première classe uniquement et les données des autres classes. Le deuxième classifieur s'applique sur les données de la deuxième classe (uniquement) et le reste des données à l'exception de la première classe (qui est déjà considérée dans le premier AdaBoost). De la même façon jusqu'à l'entraînement de toutes les classes. Les sorties successives des classifieurs faibles représentent une décision binaire pour chaque classe d'action, d'où l'appellation du classificateur en cascade [Wang, 2004].

4.5. Conclusion

Dans ce chapitre, nous avons présenté la phase de classification de postures humaines dans une scène. Nous avons aussi expliqué le principe de base de la classification en général, ainsi que le principe de fonctionnement des différents algorithmes de classification utilisés dans ce travail, à savoir les réseaux de neurones, les Séparateurs à Vaste Marge et adaboost en utilisant les arbres de décision. Comme nous l'avons déjà évoqué cette classification est considérée comme statique car l'identification s'applique sur chaque image séparément et non

Chapitre 4 : Classification de postures humaines

sur la séquence d'images comme ensemble ordonné (classification image par image). L'approche prenant en compte cet aspect permet de considérer l'aspect dynamique d'une activité et fait l'objet du chapitre suivant.

Chapitre 5 :

Classification d'activités humaines :

5.1. Classification dynamique : séquence vidéo	52
5.2. Classification de chutes en utilisant les modèles de Markov cachés	54
5.2.1. Etapes de construction du modèle HMM	55
5.2.2. Instanciation du classifieur HMM dans la détection de chutes humaines	56
5.3. Combinaison de données multi-cameras en utilisant Dempster-Shafer	59
5.4. Détection de chute en utilisant les exponentielles pondérées à moyennes mobiles (EWMA) appliquée aux données capteurs	62
5.5. Conclusion	68

Ce chapitre fait l'objet de la deuxième principale contribution de ce travail, à savoir la classification d'activité humaine dans une séquence vidéo ou la détection de chute plus précisément. Contrairement à la classification statique (présentée au chapitre précédent), la détection de chute nécessite de prendre en considération une séquence d'images constituant la vidéo. Dans ce chapitre, nous présentons en détails l'algorithme des modèles de Markov cachés utilisé pour la détection de chutes. Dans un deuxième volet, nous présentons la détection de chutes en utilisant des données multi-cameras. Pour cela deux classifieurs HMM ont été combinés en utilisant la théorie de Dempster-Shafer. Dans un troisième et dernier volet, nous présentons la détection de chutes en utilisant des données de capteurs (portés par les individus), ces données correspondent exactement aux mêmes mouvements décrits par les séquences vidéo. La détection de chute en utilisant les données capteurs a été élaborée en utilisant les méthodes statistiques à savoir les exponentielles pondérées à moyennes mobiles.

5.1. Classification dynamique : séquence vidéo

Comme nous l'avons abordé précédemment, il existe deux types de méthodes de reconnaissance de comportement: (i) la reconnaissance statique basée sur des caractéristiques obtenues à partir d'une seule image (dans notre cas, l'aspect statique réside dans la classification de postures) et (ii) la reconnaissance dynamique basée sur une analyse temporelle de la séquence d'images afin d'identifier des actions (dans le cas présent, l'aspect dynamique réside dans la détection de chute). Pour cela, nous allons procéder différemment par rapport à la classification des postures, car le vecteur caractéristique doit représenter une séquence d'images et non pas une image séparée.

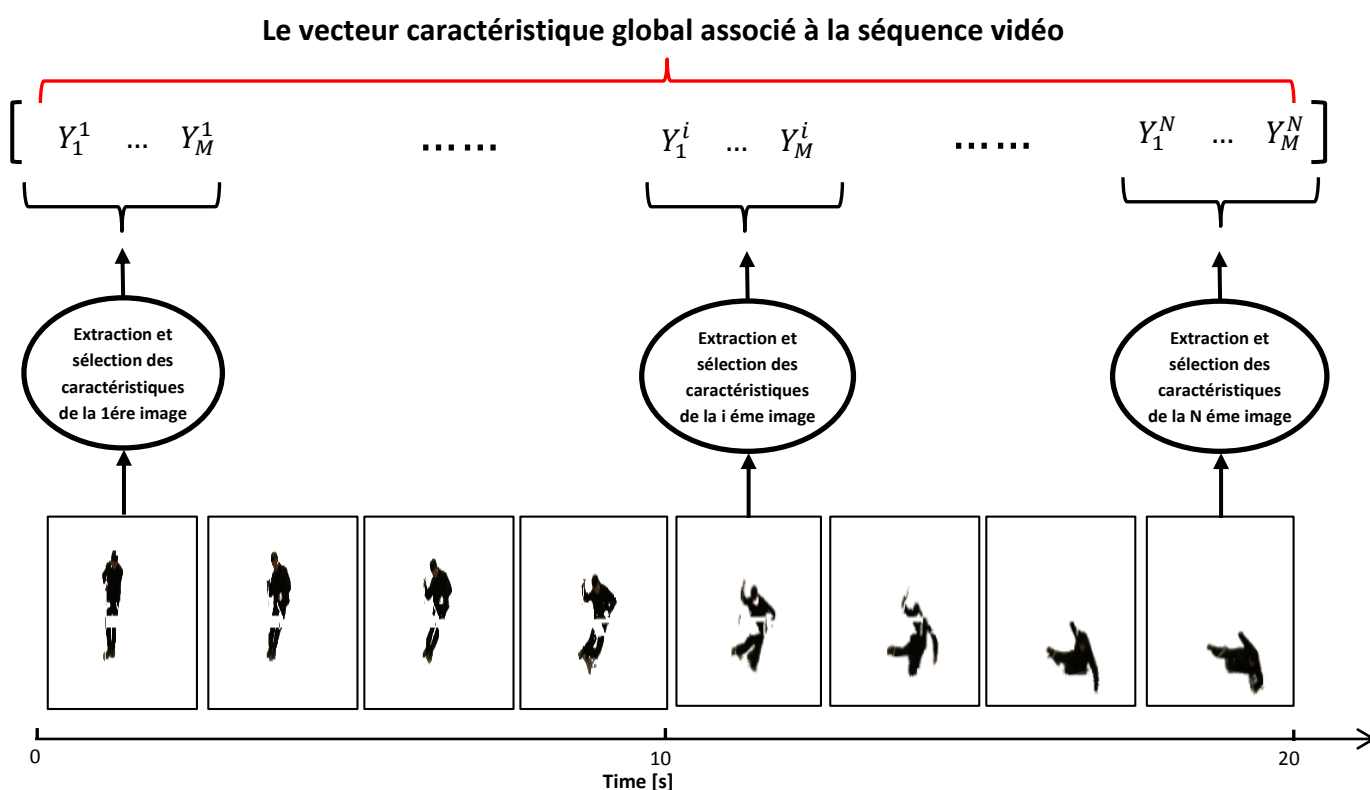


Figure 5.1 Schéma de construction du vecteur caractéristique de la séquence vidéo par la concaténation des caractéristiques des images.

Puisque les images d'une séquence vidéo sont assimilées à une séquence d'observations, les vecteurs caractéristiques correspondants à chaque image (constitués chacun par les coefficients des curvelets et les rapports de surfaces après avoir appliqué la sélection par l'évolution différentielle), sont concaténés pour définir le vecteur caractéristique global associé à la séquence vidéo.

Même si plusieurs algorithmes ont été proposés dans la classification de séquences d'images, surtout dans la reconnaissance des activités de personnes (behaviour understanding), de nombreux algorithmes ne prennent pas en compte l'information de temps ou plus particulièrement l'aspect séquentiel. Parmi ces méthodes, on peut citer : les réseaux de neurones, SVM, le plus proche voisin, le classifieur Bayésien, etc. Ces méthodes sont plus appropriées lorsqu'il s'agit de problèmes statiques comme la classification de postures, et présentent des limites avec les problèmes séquentiels. L'information du temps est primordiale pour un problème séquentiel, en particulier la durée de transition entre une position humaine à une autre. Dans le cas de la détection de chute, la durée de transition de n'importe quelle posture à la position couchée joue un rôle important dans la classification de la chute. Cette

information aide à distinguer clairement entre les chutes réelles et les activités similaires telle que s'allonger. Dans notre travail, nous avons utilisé la classification par les modèles de Markov cachés, afin de prendre en compte l'information séquentielle et surtout la durée de l'action (représentée par les transitions).

5.2. Classification de chutes en utilisant les modèles de Markov cachés

Basé sur l'apprentissage des modèles probabilistes, les HMMs jouent un rôle clé dans la résolution des problèmes de classification, essentiellement pour les données séquentielles. La pertinence de ces paradigmes statistiques, développés par Baum et Petrie [Baum, 1970] et démontrés par Rabiner et Juang [Rabiner, 1993], est très connue surtout dans le domaine de la reconnaissance de la parole. Par la suite plusieurs domaines voisins comme le traitement du signal, l'identification de texte ont également montré des résultats prometteurs en utilisant des modèles stochastiques. Pour la détection de chute, nous avons défini la classification comme un problème binaire à deux classes d'action à savoir "chute" et "non-chute". La classe "non-chute" englobe l'ensemble des actions quotidiennes (marcher, courir, etc.) et en incluant même les mouvements proches aux chutes (fausses chutes) tel que s'allongé ou se coucher, car ces derniers ne sont pas considérés comme des chutes réelles.

Le principe de la classification par les HMMs est de définir pour chaque classe un modèle de Markov caché $\lambda = \{\lambda_1, \lambda_2\}$. L'objectif des HMMs est de trouver la classe λ^* qui maximise la probabilité conditionnelle $P(\lambda/O)$, ou $O = \{O_1, \dots, O_T\}$ représente la séquence entière des observations par rapport aux frames séparées représentées par chaque O_i . Les deux modèles HMMs sont ensuite entraînés par rapport à chacune des actions (chute et non-chute) et la classification d'une séquence O est alors assignée au modèle présentant la probabilité la plus élevée. La notion de l'état (state) q dans un modèle de Markov n'est pas définie ou connue initialement telle que l'observation O . La définition des états peut changer d'un modèle à un autre même s'il s'agit d'une même application. Dans son fameux tutoriel, Rabiner a mentionné que les HMMs peuvent résoudre un problème de classification difficile sans avoir directement des états observables (d'où la désignation des états cachés) [Rabiner, 1993]. **La contribution apportée dans ce travail est de donner un sens physique aux états générés.** Pour cela, nous avons défini les classes de postures classifiées auparavant (debout, assis, accroupi, etc.) comme états du modèle de Markov. Le passage entre une posture à une autre dans un mouvement est alors traduit par des transitions entre les différents états. A chaque transition est associée une probabilité appelée probabilité de transition, l'ensemble de ces probabilités forme la matrice de transition "A". La liaison entre l'observation et l'état réside

dans une deuxième matrice appelée "B", ou elle représente les probabilités d'émission d'une observation O_i dans un état q_i .

Pour récapituler, la représentation la plus courante d'un modèle HMM est décrite par le triplet suivant: $\lambda = (A, B, \pi)$. Cependant, pour définir les composants constitutifs du modèle, un HMM est généralement caractérisé par:

- N : représente le nombre d'états cachés q_i dans le modèle. Les états sont généralement nommés cachés mais peuvent avoir une signification physique. Dans notre cas, ils sont décrits par les différentes postures du corps (debout, assis, couché, etc.).
- M : représente le nombre d'observations par état, ou les symboles d'observation O_i correspondent à la sortie du système en cours de modélisation.
- La distribution de l'état initial $\pi = \{\pi_i\}$, ou $\pi_i = P(q_i)$, car la première image représentant l'état initial peut avoir les différentes postures, avec $\sum_{i=1}^N \pi_i = 1$.
- La distribution de probabilité de transition entre les états (étant A une matrice carrée de dimension $N * N$): $A = \{a_{ij}\}$,
Ou $a_{ij} = P(q_i/q_j)$, et $i \geq 1 ; j \leq N ; \sum_j a_{ij} = 1$.
- La distribution de probabilité d'émission entre les états i et les observations k (étant B une matrice de dimension $N * M$): $B = b_i(k)$,
Où $b_i(k) = P(O_k/q_i)$ et $1 \leq k \leq M ; 1 \leq i \leq N ; \sum_k b_i(k) = 1$.

5.2.1. Etapes de construction du modèle HMM :

Pour la construction d'une classification HMM, il y a trois étapes principales qui le décrivent:

- **Evaluation:** Déterminer la probabilité $P(O_i)$ attribuée à une observation O_i par le modèle λ .
- **Décodage:** trouver la séquence la plus probable des états $q_1, q_2, \dots, q_T$ générant la séquence d'observation $O = \{O_1, O_2, \dots, O_T\}$.
- **Apprentissage:** cette phase consiste à déterminer les éléments HMM ($a_{ij}, b_i(k), \dots$) basés sur les séquences d'observation durant l'apprentissage.

Concentrons-nous maintenant sur le problème d'évaluation. Dans le concept HMM, les états générant les observations sont supposés être conditionnellement indépendants. L'étape d'évaluation consiste en effet à estimer la correspondance (matching) entre le modèle λ et la

séquence d'observation en calculant $P(O/\lambda)$. La description mathématique peut alors s'écrire comme suit:

$$P(O/\lambda) = \sum P(O, q/\lambda) \quad (5.1)$$

$$P(O, q/\lambda) = P(O/q, \lambda) P(q/\lambda) \quad (5.2)$$

En utilisant l'indépendance conditionnelle des états, nous pouvons alors écrire:

$$P(O, q/\lambda) = \prod_{t=1}^T P(O_t/q_t, \lambda) \quad (5.3)$$

Cette probabilité d'appartenance d'une séquence est calculée pour chaque modèle λ . Par la suite, la séquence est affectée à la classe λ^* qui maximise la probabilité conditionnelle $P(O, q/\lambda^*)$.

Cependant, si les deux classes ne sont pas suffisamment distinctes et que les probabilités d'appartenances sont proches, le système de détection de chutes fera appel à l'option du rejet que nous avons introduit afin de gérer les situations de conflit entre les modèles ou même les situations à risque élevé (cette condition de rejet sera évoquée en détails dans la section 6.2).

5.2.2. Instanciation du classifieur HMM dans la détection de chutes humaines :

Ce choix de classifieur est motivé par le formalisme des HMMs qui s'accorde parfaitement avec l'application puisque: (i) les échantillons représentent un aspect séquentiel (séquence d'images), (ii) et les états du modèle possèdent un sens physique car ils sont représentés par les classes de postures. D'où on peut effectuer l'instanciation suivante :

- La phase d'apprentissage est réalisée en utilisant séparément les séquences des deux classes, ou chaque séquence d'observation étant étiquetée 'chute' ou 'non-chute'.
- Par conséquent, deux modèles optimaux sont obtenus au cours de chaque phase d'apprentissage, à savoir: $\lambda_1 = (A_1, B_1, \pi_1)$ et $\lambda_2 = (A_2, B_2, \pi_2)$. Il est à noter que lors de l'apprentissage supervisé, les matrices de transition et d'émission sont estimées en utilisant des fréquences d'apparitions respectives.
- Une séquence d'observation O_i est représentée par une séquence de postures (définies comme les états de HMM).
- Pour les états générant les observations, un état q_i est associé à chaque classe de posture (déjà classifiée auparavant). Un état donné peut être: Allongé (Lying 'Ly'),

Debout (standing 'St'), Assis (Sitting 'Si'), Penché (Bending 'Be') ou Accroupi (Squatting 'Sq').

Afin d'apporter plus d'explication nous présentons l'exemple de classification suivant :

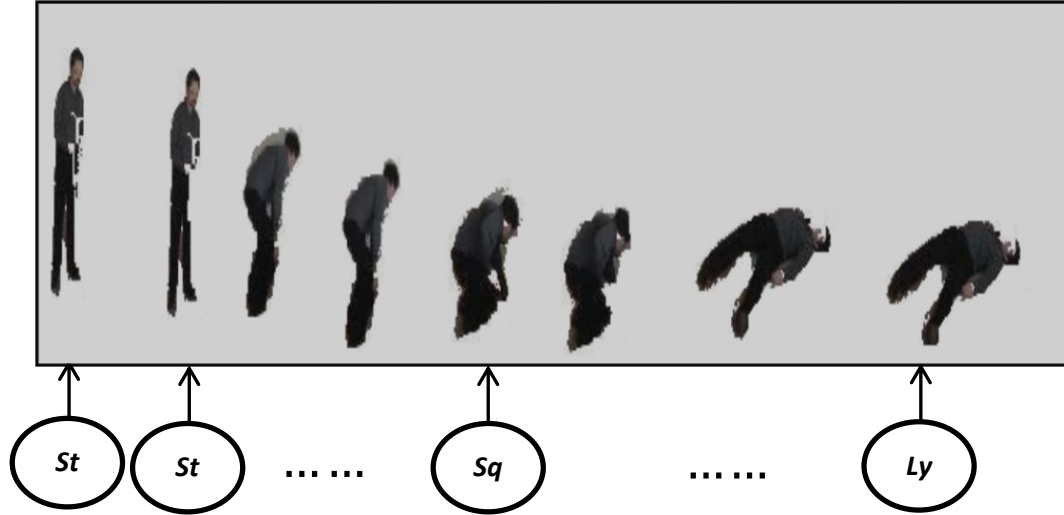


Figure 5.2 Exemple de séquence d'images avec les états physique (classes de postures) du modèle HMM

Nous reprenons les modèles déjà évoqués auparavant $\lambda_1 = (A_1, B_1, \pi_1)$ et $\lambda_2 = (A_2, B_2, \pi_2)$ décrivant les deux classes. A titre d'exemple, on définit uniquement trois séquences d'observations pour la première classe λ_1 (Tombée), ces observations sont définies comme suit :

$O_1 = \{St \ St \ St \ Be \ Be \ Be \ Sq \ Sq \ Ly \ Ly \ Ly \ Ly\}$;

$O_2 = \{St \ St \ St \ St \ St \ Be \ Be \ Sq \ Ly \ Ly \ Ly \ Ly\}$;

$O_3 = \{Si \ Si \ Si \ Sq \ Sq \ Sq \ Sq \ Sq \ Ly \ Ly \ Ly \ Ly\}$;

A partir de ces séquences d'observations, nous calculons les différents paramètres du modèle λ_1 , à savoir la matrice de transition A_1 , d'émission B_1 et le vecteur initial π_1 .

$$A_1 = \begin{bmatrix} \text{St} & \text{Si} & \text{Be} & \text{Sq} & \text{Ly} \\ 6/8 & 0 & 2/8 & 0 & 0 \\ 0 & 2/3 & 0 & 1/3 & 0 \\ 0 & 0 & 3/5 & 2/5 & 0 \\ 0 & 0 & 0 & 5/8 & 3/8 \\ 0 & 0 & 0 & 0 & 12/12 \end{bmatrix}$$

Le premier élément de la matrice A_1 (6/8) représente le nombre de transitions de l'état "debout St" au même état "St", alors que la valeur du dénominateur (8) représente le nombre d'apparition de l'état debout dans l'ensemble des observations (dans cet exemple nous avons défini 3 observations). De la même manière, le reste des éléments de la matrice A_1 sont calculés, ou on remarque que la somme des éléments de chaque ligne de la matrice A_1 est bien égale à 1. La matrice B_1 est obtenue par rapport aux émissions des observations (3 observations) dans les différents états (les 5 états possibles). Le premier élément de la matrice B_1 (3/8) représente les trois émissions de l'état "St" dans la première observation "O₁". Le vecteur initial π_1 représente le nombre d'apparition de chaque état uniquement dans la première observation "O₁".

$$B_1 = \begin{bmatrix} 3/8 & 5/8 & 0 \\ 0 & 0 & 3/3 \\ 3/5 & 2/5 & 0 \\ 2/8 & 1/8 & 5/8 \\ 4/12 & 4/12 & 4/12 \end{bmatrix}, \quad \pi_1 = \begin{bmatrix} 3/12 \\ 0 \\ 3/12 \\ 2/12 \\ 4/12 \end{bmatrix}$$

Pour cet exemple, nous avons expliqué le calcul des paramètres juste pour le premier modèle $\lambda_1 = (A_1, B_1, \pi_1)$ correspondant à la classe (Tombée). Le deuxième modèle $\lambda_2 = (A_2, B_2, \pi_2)$ correspondant à la classe (Non-chute) est défini exactement de la même manière que celui de λ_1 .

Une fois les matrices initiales du modèle HMM calculées, il reste le problème d'optimisation du HMM, c'est-à-dire trouver un modèle qui maximise la probabilité d'appartenance d'une séquence d'observations O à une classe λ (la ré-estimation des paramètres). La difficulté qui se pose est qu'il n'est pas possible de trouver un tel modèle de façon analytique. Plusieurs algorithmes ont été proposés mais l'ensemble de ces derniers nécessite une phase d'apprentissage ce qui veut dire la nécessité d'un nombre important d'observations pour chaque classe (Training samples $\{O_1 \dots O_n\}$, et $n \gg$). Dans notre travail, nous avons utilisé l'algorithme le plus connu à savoir l'algorithme de Viterbi et de Baum-Welch. L'algorithme de Baum-Welch est un algorithme itératif, qui permet d'estimer les paramètres du modèle en maximisant la probabilité d'une séquence d'observation. L'algorithme de Baum-Welch converge vers un maximum local suivant les échantillons d'apprentissage. Pour cela, un critère d'arrêt est nécessaire, tel que la tolérance de la convergence du processus, ou bien il suffit de fixer un nombre d'itérations de l'algorithme.

Autrement dit, le principe de Baum-Welch est de procéder à des réestimations des paramètres initiaux afin d'affiner le modèle à chaque nouvelle itération, c.à.d. calculer $\lambda_1^{itération 1}$ à partir de $\lambda_1^{initial}$, puis $\lambda_1^{itération 2}$ à partir de $\lambda_1^{itération 1}$, etc., ou chaque nouvelle itération doit satisfaire la condition suivante:

$$P(O/\lambda^{itération i}) > P(O/\lambda^{itération i+1}) \quad (5.4)$$

Pour cela, nous avons utilisé une procédure de type EM (Expectation-Maximisation), ce qui revient à définir les deux probabilités $\xi_l^{itération k}$ et $v_l^{itération k}$ [Baum, 1970]:

$$\xi_l^{itération k}(i, j) = P(q_t = \text{état } i, q_{t+1} = \text{état } j / O, \lambda) \quad (5.5)$$

$$\xi_l^{itération k}(i, j) = \frac{P(q_t = \text{état } i, q_{t+1} = \text{état } j, O / \lambda)}{P(O / \lambda)} \quad (5.6)$$

$$v_l^{itération k}(i) = P(q_t = \text{état } i / O, \lambda) \quad (5.7)$$

$$v_l^{itération k}(i) = \frac{\sum P(q_t = \text{état } i, q_{t+1} = \text{état } j, O / \lambda)}{P(O / \lambda)} \quad (5.8)$$

Les nouveaux paramètres se déduisent des anciens en réestimant les matrices A et B par comptage sur la base d'apprentissage de la nouvelle itération. Les éléments constituant les matrices A et B peuvent alors s'écrire comme suit [Baum, 1970]:

$$a_{ij} = \frac{\sum_{k=1}^{Nb\text{re itération}} \sum_{l=1}^N \xi_l^{itération k}(i, j)}{\sum_{k=1}^{Nb\text{re itération}} \sum_{l=1}^N v_l^{itération k}(i)} \quad (5.9)$$

$$b_j = \frac{\sum_{k=1}^{Nb\text{re itération}} \sum_{l=1}^N v_l^{itération k}(j)}{\sum_{k=1}^{Nb\text{re itération}} \sum_{l=1}^N v_l^{itération k}(j)} \quad (5.10)$$

5.3. Combinaison de données multi-cameras en utilisant Dempster-Shafer

Après la détection de chute en utilisant les SVM-HMMs comme méthode de classification, nous avons procédé par la suite à la détection de chute en utilisant plusieurs caméras. Cette approche était motivée en premier lieu par des cas de chutes non détectés, ou le corps humain n'était pas complètement couvert par la caméra, ou une partie du corps était cachée par un élément appartenant à l'arrière-plan. En plus, l'utilisation d'une seule caméra nécessite une optimisation du champ de vision, afin d'éliminer au maximum les coins morts. Par contre,

l'utilisation de plusieurs caméras permet d'avoir un champ de vision plus large d'où une couverture plus importante. La deuxième motivation de l'exploitation de plusieurs caméras est la disponibilité des données multi-vues (en terme de base de données ou benchmark). Parmi les méthodes proposées utilisant les données multi-caméras, beaucoup d'entre elles extraient les informations à partir de différentes caméras et les utilisent dans un seul vecteur caractéristique. Ensuite, ce vecteur de caractéristiques est destiné alors à la phase de classification (avec un seul classifieur). Cette approche présente plusieurs limites, ou la défaillance d'une caméra influe négativement sur le vecteur caractéristique. En plus, les situations de conflits entre les caractéristiques issues des différentes caméras ne peuvent pas être gérées par le classifieur unique. Dans ce travail, nous avons proposé une approche efficace pour la combinaison de plusieurs caméras en utilisant les HMMs en tant que classifieurs et la théorie de Dempster Shafer (Dempster-Shafer Theory of Evidence DSTE) comme méthode de combinaison comme l'illustre la figure 5.3.

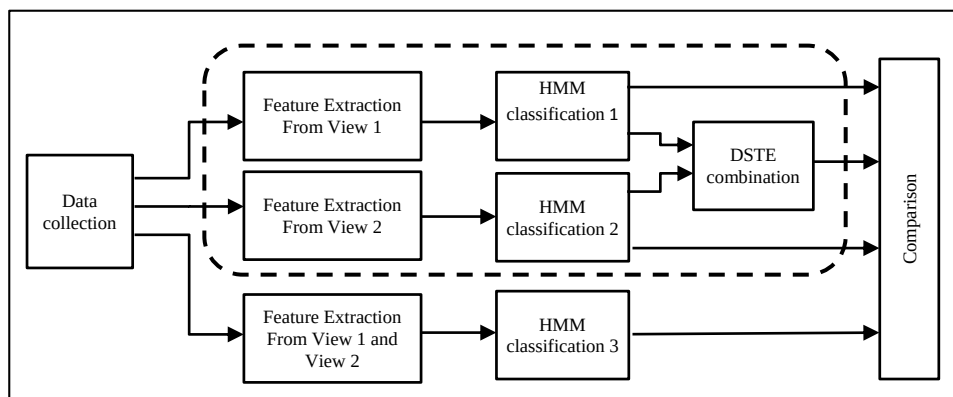


Figure 5.3 Schéma de l'approche du système de détection de chute basée sur la combinaison Dempster-Shafer de multiple classifieurs.

Le principe est d'extraire les caractéristiques de chaque caméra et les classifier séparément par les différents HMMs, afin d'affecter chaque séquence de frames à une classe. La décision finale est alors effectuée en utilisant la combinaison de DSTE basée sur les sorties probabilistes des classifieurs HMMs, afin de gérer les éventuels conflits entre les classifieurs. Cette méthodologie de combinaison s'est avérée plus efficace que d'utiliser plusieurs classifieurs séparés pour chaque caméra ou même d'utiliser les caractéristiques de façon globale.

A partir des classifieurs probabilistes HMM, les probabilités a posteriori sont converties puis combinées en utilisant les règles DSTE. Les sorties de combinaison peuvent être considérées comme des poids, définis pour chaque classe, la préférence est alors donnée à la classe ayant

le poids le plus important. En d'autres termes, DSTE établit un cadre de discernement, qui est défini comme un ensemble de descriptions. Il consiste à calculer le degré de croyance (plausibilité) à travers le calcul de la fonction de proximité ϕ . Ces deux termes, à savoir le degré de croyance ou fonction de proximité, sont dérivés de la matrice du modèle de décision. Cette dernière fonction est appliquée à tous les classifieurs via la matrice de profils de décision. Ce fait constitue une modélisation souple et riche couvrant l'ensemble des situations de surveillance du comportement humain. Supposons O une séquence d'observation (qui représente une séquence vidéo) constituée par le vecteur de caractéristiques $Y = \{Y_1, \dots, Y_M\}$ (les caractéristiques sélectionnées auparavant) et $\Omega = \{C_1, \dots, C_K\}$ l'ensemble des étiquettes des classes (pour notre cas nous avons que deux classes Tombée et Non-tombée). Chaque classifieur de l'ensemble $\lambda = \{\lambda_1, \dots, \lambda_K\}$ affiche le degré d'appartenance de la séquence d'observation à une classe. Plus le degré d'appartenance est grand, plus le label de la classe correspondante est probable. Comme nous disposons dans notre cas de plusieurs classifieurs, les sorties de ces classifieurs sont alors organisées dans un profil de décision $DP(O)$, comme le montre la matrice suivante :

$$DP(O) = \begin{pmatrix} d_{1,1}(O) & d_{1,2}(O) & \dots & d_{1,K}(O) \\ \vdots & \vdots & \ddots & \vdots \\ d_{K,1}(O) & d_{K,2}(O) & \dots & d_{K,K}(O) \end{pmatrix} \quad (5.11)$$

La méthode de fusion basée sur la théorie de Dempster-Shafer utilise le profil de décision de chaque classifieur $d_{i,j}$ afin de trouver le support global pour chaque classe i et affecte ensuite l'instance O à la classe ayant le plus grand support. Pour cela, les modèles de décision de Dempster-Shafer DT_i sont construits à partir des données d'apprentissage. Ces modèles de décision (DT_i) sont typique à chaque classe. Pour chaque séquence de test O , la méthode de Dempster-Shafer compare le profil de décision $DP(O)$ par rapport aux modèles de décision DT_i . Pour une meilleure représentation, le déroulement de cette fusion est récapitulé par les étapes suivantes:

a) Construction des modèles de décision:

Cette étape consiste à calculer la moyenne des profils de décision pour toutes les séquences d'entraînement appartenant à une classe C_i . Cette moyenne est appelée un modèle de décision (Decision Template DT) de la classe C_i :

$$DT_i = \frac{1}{N_i} \sum_{O \in C_i} DP(O) \quad (5.12)$$

Où N_i représente le nombre d'échantillons d'entraînement appartenant à la classe C_i .

b) Calcul de la fonction proximité :

La fonction proximité ϕ est calculée entre le modèle de décision d'une classe C_i et la sortie estimée du classifieur λ_j d'un échantillon test O :

$$\phi_{ij}(O) = \frac{(1 + \|DT_j^i - d_i(O)\|)^{-1}}{\sum_{i=1}^K (1 + \|DT_j^i - d_i(O)\|)^{-1}} \quad (5.13)$$

c) Calcul des degrés de croyances :

Les degrés de croyance (belief degrees) correspondant au classifieur λ_j et à la classe C_i peuvent être exprimés comme suit:

$$b_j(DT_i(O)) = \frac{\phi_{ij}(O) \prod_{i \neq j} (1 - \phi_{ij}(O))}{1 - \phi_{ij}(O) [1 - \prod_{i \neq j} (1 - \phi_{ij}(O))]} \quad (5.14)$$

d) Décision finale basée sur le support final de la classe :

Pour obtenir la décision finale de la combinaison, la règle de Dempster-Shafer combine les degrés de croyance obtenus de chaque classificateur; ou le support final relatif à chaque classe est obtenu en multipliant tous les degrés de croyance calculés auparavant :

$$\mu_j(O) = K \prod_{j=1} b_j(DT_i(O)) \quad (5.15)$$

K est une constante de normalisation assurant que le support total pour une classe C_i (représentée par DT_i) de tous les classificateurs est égal à 1. Enfin, la règle de Dempster-Shafer sélectionne la classe correspondante à la plus grande valeur de $\mu_j(O)$.

5.4. Détection de chute en utilisant les exponentielles pondérées à moyennes mobiles (exponentially weighted moving average EWMA) appliquée aux données capteurs :

Dans les sections précédentes, nous nous sommes concentrés sur l'identification du comportement humain en se basant sur les données caméras couleurs RGB (Red Green Blue), cependant, ce type de caméras est sensible aux conditions d'éclairage. Il existe quelques séquences, où la détection de personne dans l'image est quasi impossible suite au mauvais éclairage de l'environnement. Cette limitation nous a motivé à utiliser les données capteurs

dans la détection de chute en plus des données caméras. Il est possible d'utiliser encore des techniques de machines d'apprentissage et les appliquer sur les données capteurs pour la détection de chutes, mais cette dernière va générer un coût de calcul supplémentaire relatif aux estimations et réglages des paramètres. Pour garantir un système fiable et rapide permettant son adaptation pour des données en temps réel, nous avons sélectionné une méthode simple et efficace basée sur les exponentielles pondérées à moyennes mobiles EWMA. L'utilisation des EWMA comme méthode de détection va permettre une importante réduction dans les coûts de calcul. Les méthodes de surveillance basées sur les invariances statistiques, telles que EWMA [Montgomery, 2005], ne sont introduites que récemment afin de détecter les petites anomalies et améliorer les performances de la détection des chutes humaines. Dans notre cas, EWMA est appliquée sur les amplitudes des signaux fournis par les accéléromètres. Nous rappelons que ces accéléromètres (ou capteurs de mouvement) sont portés par les personnes (wearable sensors), ou l'accélération est automatiquement détectée quand la personne change de direction ou de vitesse, ce qui rend ces capteurs appropriés à la détection de chutes. Dans cette étude, l'accéléromètre est placé près de la région du bassin de la personne, comme l'illustre la figure 5.4.

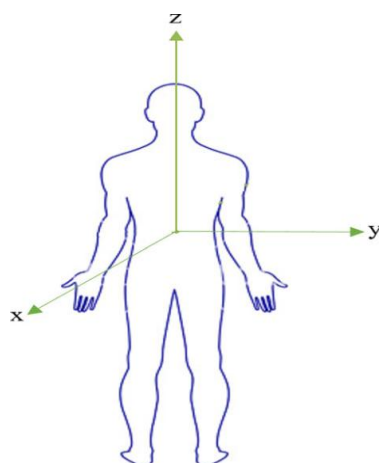


Figure 5.4 Schéma de l'approche du système de détection de chute basée sur la combinaison Dempster-Shafer de multiple classifieurs.

L'accéléromètre capte des données en trois dimensions: accélération longitudinale (axe des abscisses), accélération latérale (axe des y) et accélération verticale (axe z). La magnitude de l'accélération, a , dans l'espace tridimensionnel, qui est la norme vectorielle, est utilisée comme indicateur de chute et elle est définie comme suit :

$$a = \sqrt{a_x(t)^2 + a_y(t)^2 + a_z(t)^2} \quad (5.16)$$

Ou $a_x(t)$, $a_y(t)$ et $a_z(t)$ représentent les composantes d'accélération des axes x, y et z respectivement, à l'instant t. Un exemple sur les composantes d'accélération triaxiale observées et leurs composante résultante est illustrée dans la figure 5.5 décrivant différents gestes tels que marcher, s'asseoir, se pencher et tomber.

- **Principe de la détection à base des exponentielles pondérées à moyennes mobiles :**

La détection d'anomalies dans un système contrôlé est basée sur la vérification des différences statistiques entre les mesures courantes et les mesures sans erreur connues a priori (c'est-à-dire des mesures sans anomalies). Ces graphiques peuvent détecter de petits changements dans la moyenne des processus car ils sont pondérés dans le temps ; sachant que les graphiques EWMA pondèrent la dernière observation en fonction de son importance dans la caractérisation du processus [Montgomery, 2005].

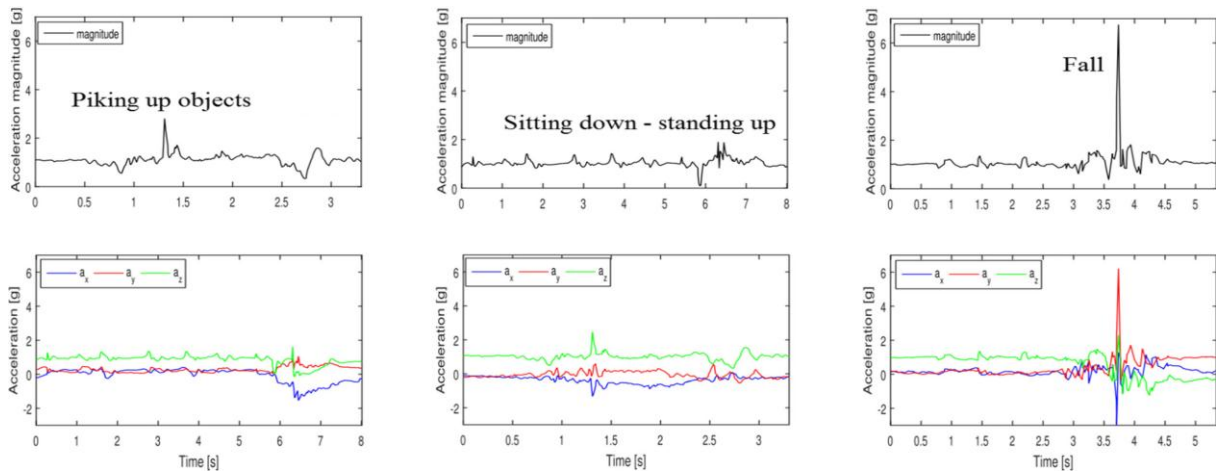


Figure 5.5. Exemple sur les composantes d'accélération résultantes (rangée du haut) et leurs composantes d'accélération triaxiale correspondantes (rangée du bas) décrivant différents gestes : ramassant un objet (colonne de gauche), s'asseoir puis se lever (colonne de milieu) et tomber (colonne de droite).

Le schéma de contrôle EWMA a été introduit par Roberts [Roberts, 1959] et il a depuis été largement utilisé dans l'analyse des séries chronologiques, spécialement dans le domaine industriel (dans le contrôle des chaînes de production). Selon plusieurs travaux proposés dans la littérature [Montgomery, 2005], [Zerrouki, 2016], [Shamsuzzaman, 2016], EWMA est l'une des méthodes de contrôle les plus utilisées en raison de sa flexibilité permettant la sélection

des paramètres appropriés pour atteindre les meilleures performances possibles ; en plus de sa sensibilité aux petits changements.

Supposons que $A = \{a_1, a_2, \dots, a_N\}$ sont des données collectées à partir des accéléromètres du processus de surveillance (sachant que ces données correspondent exactement aux séquences d'images enregistrées par les caméras). La statistique EWMA est alors calculée en utilisant l'équation suivante :

$$\begin{cases} z_t = \lambda a_t + (1 - \lambda) z_{t-1} & \text{if } t > 0 \\ z_0 = \mu_0 & \text{if } t = 0 \end{cases} \quad (5.17)$$

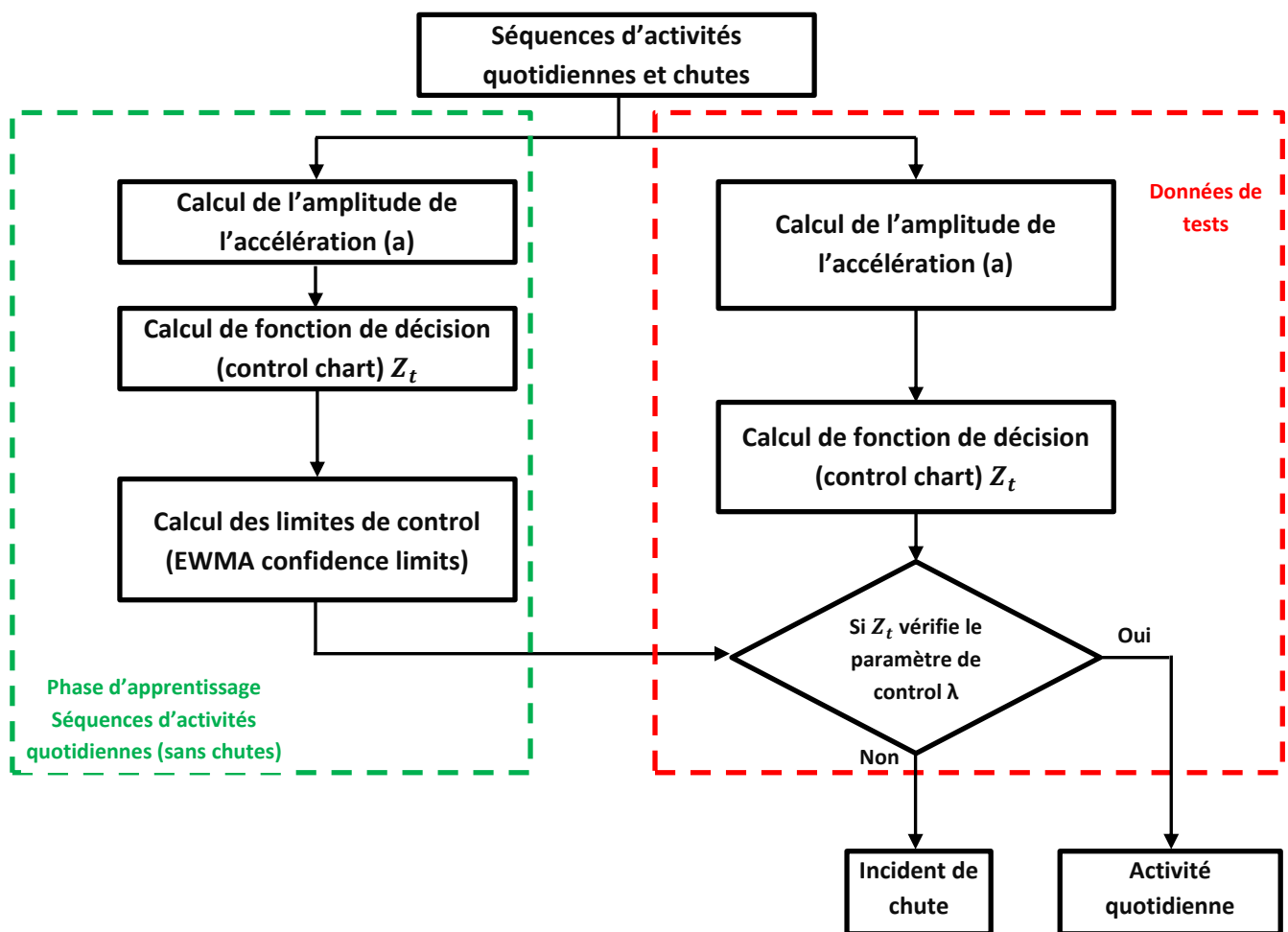


Figure 5.6. L'organigramme de l'algorithme de détection de chute basé sur EWMA

Ou μ_0 représente la moyenne des données sans erreur et $\lambda \in [0, 1]$ est un paramètre de pondération qui détermine la mémoire temporelle de la fonction de décision EWMA. Le diagramme de EWMA est généralement approprié pour détecter des changements relatifs au choix du paramètre λ . Autrement dit, lorsque le système est sensible aux petites anomalies

alors λ est choisi pour être petit, par contre lorsque la valeur de λ est choisi grande, le système est adapté pour détecter des changements relativement importants.

Plusieurs travaux proposés dans la littérature recommandent d'utiliser des valeurs du paramètre de pondération λ comprise entre 0,2 et 0,3 pour la plupart des objectifs de contrôle [Hunter, 1986], [Shamsuzzaman, 2016]. Cette charte basée sur EWMA permet de conclure avec évidence significative la présence d'anomalie à un instant t si la statistique de décision z_t dépasse les limites de contrôle qui sont exprimées en fonction de l'écart-type asymptotique de la statistique de décision :

$$z_t < \mu_0 - L\sigma_{z_t} \text{ or } z_t > \mu_0 + L\sigma_{z_t} \quad (5.18)$$

Où $\sigma_{z_t} = \sigma_0 \sqrt{\frac{\gamma}{(2-\gamma)} [1 - (1-\gamma)^{2t}]}$ représente l'écart-type de la statique z_t et σ_0 représente l'écart type de l'ensemble de données sans anomalie ou préliminaire. L est la largeur de la limite de contrôle, qui est généralement spécifiée en 3, ce qui correspond à un taux de fausses alarmes de 0,27%.

Dans notre travail, nous utilisons une version unilatérale d'EWMA pour la détection des chutes. Cette détection (appliquée sur les amplitudes des signaux fournies par les accéléromètres) comprend deux principales étapes à savoir: l'étape d'apprentissage et l'étape des tests. Cette approche est illustrée schématiquement par la figure 5.6 et décrite comme suit:

(i) La phase d'apprentissage implique le calcul des amplitudes des accélérations à partir des données sans chutes et de la limite de contrôle EWMA pour la détection (en utilisant l'équation 5.17).

(ii) La phase de test est une comparaison entre les amplitudes des accélérations des nouvelles mesures et le seuil précédemment calculé ; afin de détecter la présence d'anomalie d'où la prise de décision. On peut remarquer que cette méthode de détection diffère par rapport aux méthodes de classification dans la phase d'apprentissage ; ou cette dernière est appliquée uniquement sur les données correspondantes aux gestes quotidiens sans la présence des chutes. Autrement dit, il n'y a pas de phase d'apprentissage sur les données correspondantes aux chutes. Puisque dans les méthodes de détection, la chute est considérée comme anomalie et généralement les anomalies ne suivent pas une modélisation mathématique.

L'avantage principal de cette méthode de détection par rapport aux méthodes de seuillage (appelées classiques) est la définition automatique des paramètres de contrôle. Contrairement aux méthodes de seuillages classiques, où la détermination des valeurs limites nécessite une connaissance a priori des données ou même parfois ces valeurs sont déterminées empiriquement. Par ailleurs, la détection basée sur EWMA permet une actualisation simple et rapide des paramètres de control, ou il suffit de rajouter des séquences de données correspondantes aux types de gestes désirés pendant la phase d'apprentissage. Cependant, l'inconvénient de la détection de chute basée sur les EWMA est que cette dernière ne permet pas la distinction entre les types d'anomalies car aucun préalable apprentissage n'est appliqué sur les données comprenant les chutes. Cet inconvénient conduit à des confusions entre les chutes réelles avec quelques gestes quotidiens similaires tels que s'allonger d'une façon relativement rapide. La présence de ces confusions n'influe pas sur la détection des chutes réelles mais fait hausser considérablement le nombre de fausses alarmes. C'est pour cette raison que dans le présent travail, la prise de décision a été renforcée par des algorithmes de machines d'apprentissage appliquées sur les séquences images. Uniquement les séquences vidéo correspondantes au cas détectés par EWMA (données accéléromètres) seront considérés par l'étape de classification (comme le montre la figure 5.7). Ce qui réduit énormément le nombre de séquences utilisées dans la classification et réduit par la suite le temps de calcul global du système. Pour cette étape de classification, nous avons défini deux classes, à savoir : chutes réelles et fausses chutes (real falls and like-falls). L'algorithme SVM a été choisi comme méthode de classification. Ce choix est dû aux performances des SVMs dans la résolution des problèmes linéaire ou non linéaire par l'utilisation des fonctions noyaux.

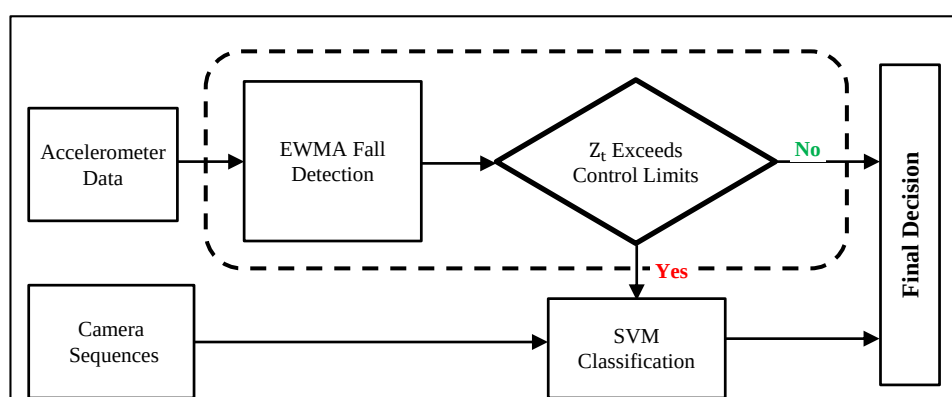


Figure 5.7. L'organigramme de détection de chute basé sur le formalisme EWMA-SVM

Pour récapituler, la détection de chute basée sur la technique EWMA identifie la chute suivant les données d'accéléromètres. Par la suite, le système extrait la séquence vidéo de la personne

surveillée et applique la classification SVM afin de confirmer l'événement de chute. Par contre, si la statistique de décision EWMA est inférieure au seuil, le système traite la séquence suivante où de nouvelles données d'accéléromètre sont collectées.

5.5. Conclusion

Dans ce chapitre, nous avons présenté la phase de classification dynamique à savoir la classification d'activité humaine dans une séquence vidéo ou la détection de chute plus précisément. Nous avons présenté en détails l'algorithme des modèles de Markov cachés utilisé dans la classification des chutes. Nous avons évoqué l'attribution d'une signification physique aux états cachés du HMM ce qui a fait l'objet d'une principale contribution de ce travail. Nous avons aussi présenté la détection de chutes en utilisant des données multi-cameras, ou deux classifieurs HMMs ont été combinés par la théorie de Dempster-Shafer. Nous avons présenté par la suite la détection de chutes en utilisant les amplitudes des signaux fournies par les accéléromètres. Cette dernière (détection) a été réalisée par l'introduction des méthodes statistiques basées sur les exponentielles pondérées à moyenne mobiles, ou la chute était considérée comme une anomalie au lieu que ça soit un phénomène à caractériser ou modéliser. Il est à noter aussi que pendant la phase de détection plusieurs confusions entre les chutes réelles et quelques gestes quotidiens similaires ont été enregistrées. Pour cette raison, la phase de détection de chute basée sur EWMA est suivie par une autre phase de classification SVM (appliquée sur les séquences d'images) afin de confirmer l'événement de chute et diminuer le nombre de fausses alarmes et permettre une reconnaissance fiable et précise.

En effet, plusieurs combinaisons de techniques peuvent mener à sélectionner la meilleure méthode de classification ou de détection de chutes. Le théorème de **“No Free Lunch”** indique qu'il n'y a pas d'algorithme unique qui, dans un domaine quelconque, induit toujours à la classification la plus précise [Alpaydin, 2014]. Pour remédier aux limitations de chaque approche (capteurs, imagerie, méthode de détection et classification, etc.), nous avons exploité dans ce travail des données issues de sources multiples à savoir accéléromètres, caméra, puis un système multi-caméras.

Chapitre 6 :

Etude expérimentale et validation de la méthodologie

6.1. Description des données	70
6.1.1. Description de la base de données UR Fall Detection Dataset (URFD)	70
6.1.2. Description de la base de données Fall Detection Database (FDD)	71
6.2. Fonctions : pertes et risques (Losses and Risks)	72
6.3. Critères d'évaluation	74
6.4. Résultats et discussion	77
6.4.1. Evaluation de l'utilisation des valeurs singulières tronquées comme caractéristiques de la silhouette humaine	78
6.4.2. Evaluation de l'utilisation des coefficients de Curvelets et les rapports de surfaces comme caractéristiques de la silhouette humaine	79
6.4.3. Evaluation de la classification de postures humaines	83
6.4.4. Evaluation de la détection de chutes humaines	85
6.4.5. Comparaisons des méthodes proposées avec des systèmes de détection existants	91
6.5. Conclusion	94

Le but de ce chapitre est d'évaluer l'ensemble des contributions apportées pour les différentes phases de traitement à savoir l'extraction des caractéristiques, la classification de postures et la détection de chutes. Pour cela, nous avons effectué plusieurs tests sur des données réelles collectées au sein de laboratoires internationaux ; ces données sont jugées motivantes ou plusieurs challenges sont présentés et considérés dans les séquences de vidéo-surveillance. Dans ce chapitre, nous décrivons en détails les caractéristiques des bases de données telles que le type de caméras utilisées, le flux des séquences, les environnements choisis, etc. Dans un deuxième volet, nous décrivons les mesures d'évaluation utilisées dans ce travail. Il est à noter que plusieurs critères de tests ont été évoqués afin d'assurer une évaluation significative du système. Dans un troisième et dernier volet, nous présentons les résultats obtenus de la classification de postures, la détection de chutes ainsi que les discussions et les interprétations correspondantes.

6.1. Description des données :

Pour valider les différentes contributions apportées dans ce travail (dans les deux applications à savoir classification de posture et détection de chutes), nous avons utilisé deux bases de données publiquement disponibles online : UR Fall detection Dataset (URFD) [Kwolek, 2014] et Fall Detection Database (FDD) [Charfi, 2012].

6.1.1. Description de la base de données UR Fall Detection Dataset (URFD):

Les séquences vidéo constituant cette base de données ont été collectées par les chercheurs du centre interdisciplinaire de modélisation informatique, Université de Rzeszow, Pologne (Interdisciplinary Centre for Computational Modelling, University of Rzeszow). En plus des séquences vidéo, cette base de données contient aussi les amplitudes des signaux fournies par les accéléromètres. Ces mesures d'accéléromètres correspondent exactement aux séquences vidéo; ce qui nous a permis, dans ce travail, de combiner les deux approches caméra et capteurs pour une meilleure caractérisation des chutes.

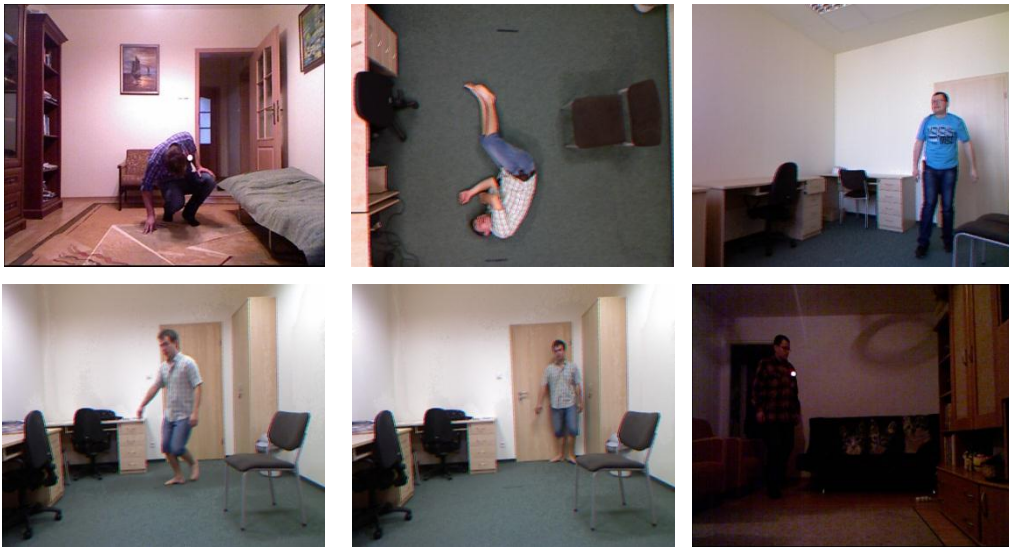


Figure 6.1 Exemples de frames de la base de données URFD contenant des conditions d'éclairage variables,

En effet, cette base de données contient 70 séquences, dont 30 séquences présentant des chutes et 40 séquences présentant différentes activités quotidiennes. Les séquences vidéos sont enregistrées par deux caméras couleurs RGB (caméra (0) et caméra (1)), fixées parallèles au plancher et au plafond, respectivement. Les enregistrements vidéo sont réalisés dans plusieurs environnements permettant une diversification des situations d'enregistrement à savoir : les conditions d'éclairage variables, présence d'objets, présence de reflets et

d'ombres, etc. (comme l'illustre la figure 6.1). Les données de capteurs ou d'accéléromètres ont été collectées à l'aide des dispositifs PS Move (60 Hz) et x-IMU (256 Hz). L'acquisition des données d'accéléromètre débute au même instant que la séquence vidéo, ces données sont stockées en unités de gravité (g). L'ensemble de données enregistré des capteurs et caméras est stocké dans un tableau en mentionnant les numéros de frames et les temps (en millisecondes) correspondant. Ces données sont aussi étiquetées soit par (-1) ce qui signifie un geste quotidien ou par (1) ce qui signifie que la personne est tombée. Il est à noter qu'après l'acquisition des séquences vidéo et les signaux d'accéléromètre, ces données ne sont pas strictement synchronisées. Pour une meilleure synchronisation, un arrangement basé sur la valeur d'horodatage la plus proche (synchronization based on nearest timestamp value) est alors appliqué [Kwolek, 2014].

6.1.2. Description de la base de données Fall Detection Database (FDD):

FDD est la deuxième base de données utilisée dans ce travail, cette base de données a été construite par les chercheurs du Laboratoire Electronique, Informatique et Image (LE2I), de l'Université de Bourgogne, avec la coopération du Centre National de la Recherche Scientifique (CNRS, France). Cette base de données contient uniquement des séquences vidéo, ces séquences ont été enregistrées en utilisant un système de surveillance réel doté d'une seule camera. La fréquence d'acquisition d'images est de 25 images / s et d'une résolution de 320 x 240 pixels. Ces données vidéo illustrent les principales difficultés et risques probables dans un environnement familial pour les personnes âgées.



Figure 6.2 Exemples de frames de la base de données FFD dans des différents environnements

Ces séquences sont aussi enregistrées dans plusieurs environnements tels que salle de séjour, chambre, bureau, etc. (comme le montre la figure 6.2). La base de données FDD contient aussi des séquences vidéo avec des illuminations variables et des difficultés typiques comme des occlusions ou des arrière-plans encombrés et texturés, afin de mesurer la robustesse des méthodes de détection de personne. Les chercheurs ont effectué diverses activités

quotidiennes normales et des chutes. La base de données FDD contient 191 vidéos, ces séquences sont toutes étiquetées. Un fichier représentant la vérité terrain, contenant les identifiants des séquences avec les étiquettes correspondantes, est disponible. Ce fichier contient aussi des informations supplémentaires telles que de la position de chute dans la séquence d'images [Charfi, 2012].

6.2. Fonctions pertes et risques (Losses and Risks)

Dans un problème de classification ou d'une prise de décision en général, les décisions prises sont caractérisées par deux facteurs essentiels à savoir la perte et le coût. Ces facteurs sont introduits initialement dans le domaine financier pour mesurer le facteur de risque dans la prise de décision [Alpaydin, 2014]. Ces fonctions sont appliquées dans le domaine de classification quand une erreur de classification peut engendrer une situation critique, tels que le diagnostic médical ou la prévision des séismes, etc. Dans notre travail, nous avons utilisé les fonctions pertes et risques afin d'activer l'option du rejet lors d'une détection de chute. Autrement dit, dans les problèmes de classification non-symétriques, telle que la détection de chute où le nombre de séquences présentant des activités quotidiennes normales dépasse le nombre de séquence représentant les chutes (unbalanced problems), le facteur de risque d'une erreur de classification n'est pas le même pour les deux classes [Hand, 2012]. Une erreur de classification d'une chute est beaucoup plus risquée qu'une erreur de classification d'une activité normale. Le but d'intégration du rejet est d'éliminer les situations tangentes, où le système de détection de chute choisira le rejet (le principe d'abstention) au lieu de l'attribution de la classe « non-chute ». Il est bien établi qu'une séquence rejetée (par lequel le système peut refuser la décision du classifieur) *entraîne un coût inférieur* par rapport à une erreur de classification. Par contre dans le calcul des performances, les cas rejetés sont considérés comme mal classifiés afin d'assurer une évaluation significative du système.

La question qui se pose est de savoir quel sont les cas concernés par le rejet dans la détection de chute? En effet, il existe quelques scénarios (en plus des cas tangents) dans lesquels une posture humaine est partiellement visible en raison de la présence d'ombres, ou d'occlusions de parties du corps avec des objets de l'environnement.

La condition de rejet est essentiellement basée sur la minimisation du risque associée à la matrice de pertes $\mathcal{L}$ (Loss Matrix) [Alpaydin, 2014]. Notons α_i ($i = 1, 2$) les deux actions d'assigner la séquence vidéo O à une des classes λ_i ("chute" ou "non-chute"). L'intégration du rejet mène à une troisième action α_3 qui correspond à cette décision. Chaque ligne de la

matrice de perte correspond à une action α_i à entreprendre. Un classifieur optimal préconise l'action α_i avec correspondante au risque minimum $\mathfrak{R}^*$:

$$\mathfrak{R}(\alpha_i|O) = \min_k \mathfrak{R}(\alpha_k|O) \quad (6.1)$$

Une classification correcte est correspond à une valeur de perte nulle :

$$\omega_{11} = \omega_{22} = 0 \quad (6.2)$$

Cependant, les erreurs de classification ne sont pas d'égal coût dans le cas de la détection de chute ($\omega_{12} \neq \omega_{21}$). Comme nous l'avons déjà évoqué, une erreur de classification d'une chute est plus critique qu'une classification erronée d'un geste quotidien. Par conséquent, pour discriminer les pertes entre les deux mauvaises décisions, différentes valeurs de ω_{12} et ω_{21} ont été utilisées pendant les phases de tests. La matrice de perte est définie comme suit:

$$\mathfrak{L} = \begin{matrix} \alpha_1 & \omega_{11} & \omega_{12} \\ \alpha_2 & \omega_{21} & \omega_{22} \\ \alpha_3 & \omega & \omega \end{matrix} \quad (6.3)$$

Le risque de choisir l'option rejet pour une séquence O est:

$$\mathfrak{R}(\alpha_3|O) = \sum_k^2 \omega P(O/\lambda_i) \quad (6.4)$$

Puisque les deux variables sont indépendantes, alors :

$$\mathfrak{R}(\alpha_3|O) = \omega \sum_k^2 P(O/\lambda_i) \quad (6.5)$$

d'où :

$$\mathfrak{R}(\alpha_3|O) = \omega \quad (6.6)$$

Les valeurs des risques de choisir une des classes $\lambda_i (i = 1, 2)$ tombée ou non-tombée sont définies par l'équation suivante :

$$\mathfrak{R}(\alpha_i|O) = \sum_{k \neq i} \omega_{ik} P(O/\lambda_k) \quad (6.7)$$

Les fonctions risques sont alors définies comme suit :

$$\begin{cases} \mathfrak{R}(\alpha_1|O) = \omega_{12} P(O/\lambda_2) \\ \mathfrak{R}(\alpha_2|O) = \omega_{21} P(O/\lambda_1) \\ \mathfrak{R}(\alpha_3|O) = \omega \end{cases} \quad (6.8)$$

Une fois les fonctions risques sont calculées, le choix de l'action optimale revient à choisir l'action correspondante au risque minimal d'où satisfaire les inégalités suivantes :

- Choisir λ_1 si : $\begin{cases} \mathfrak{R}(\alpha_1|O) < \mathfrak{R}(\alpha_2|O) \text{ et} \\ \mathfrak{R}(\alpha_1|O) < \mathfrak{R}(\alpha_3|O) \end{cases}$

$$\text{Donc choisir } \lambda_1 \text{ si: } \begin{cases} P(O/\lambda_1) > \frac{\omega_{12}}{\omega_{21}} \times P(O/\lambda_2) \text{ et} \\ P(O/\lambda_1) > 1 - \frac{\omega}{\omega_{12}} \end{cases}$$

- Choisir λ_2 si : $\begin{cases} \mathfrak{R}(\alpha_2|O) < R(\alpha_1|O) \text{ et} \\ \mathfrak{R}(\alpha_2|O) < R(\alpha_3|O) \end{cases}$

$$\text{Donc choisir } \lambda_1 \text{ si: } \begin{cases} P(O/\lambda_2) > \frac{\omega_{21}}{\omega_{12}} \times P(O/\lambda_1) \text{ et} \\ P(O/\lambda_2) > 1 - \frac{\omega}{\omega_{21}} \end{cases}$$

- Choisir l'option du rejet si : $\begin{cases} \mathfrak{R}(\alpha_3|O) < R(\alpha_1|O) \text{ et} \\ \mathfrak{R}(\alpha_3|O) < R(\alpha_2|O) \end{cases}$

$$\text{Donc choisir l'option du rejet si: } \begin{cases} P(O/\lambda_1) > \frac{\omega}{\omega_{21}} \text{ et} \\ P(O/\lambda_2) > \frac{\omega}{\omega_{12}} \end{cases}$$

6.3. Critères d'évaluation

Afin d'évaluer de manière significative la méthodologie proposée, plusieurs mesures statistiques ont été évoquées, telles que la performance de bonne reconnaissance (accuracy), le coefficient Kappa, F-measure, l'analyse par rapport à la courbe ROC, [Hand, 2012]. Ces critères d'évaluation sont appliqués dans un premier lieu sur les résultats de classification de postures (en incluant les phases d'extraction des caractéristiques et reconnaissance de postures, etc.). Par la suite, une deuxième évaluation a été appliquée sur les résultats de la détection de chutes (en considérant l'utilisation des données de capteurs, caméras, la combinaison, l'option de rejet, etc.).

- **Taux de reconnaissance (Accuracy) :**

Ce critère représente le pourcentage de postures ou d'actions qui sont correctement classifiées ou reconnues. Ce taux est calculé à partir des moyennes des éléments constituant la diagonale de la matrice de confusion. Cette matrice est dérivée d'une comparaison entre les données (postures ou séquences vidéo) de référence avec les données classifiées par le système. Il est à noter que le taux de reconnaissance est la mesure d'évaluation la plus largement utilisée. La précision ou le taux de reconnaissance est exprimé comme suit:

$$\text{Précision} = \frac{tp+tn}{tp+tn+fp+fn} \quad (6.9)$$

Ou tp (true positive ou vrai positif) et tn (true negative ou vrai négatif) représentent les données correctement classifiées (soit ces données représentent la classe ‘non-chute’ et elles ont été correctement classifiées dans cette classe ‘ tn ’ ; ou bien ils représentent la classe ‘chute’ et elles sont aussi correctement détectées ‘ tp ’). Pour fp (false positive, ou faux positif) représente les données mal classées, ce terme est également connu sous le nom de taux de fausses alarmes, ou une séquence représentant un geste quotidien est détectée comme chute et affectée à la classe ‘chute’. Le terme fn (false negative, ou faux négatif) est le taux de classification erronée ou perdue, tel qu’une séquence représentant une chute et elle est classée comme ‘non-chute’. Le taux d'erreur est défini comme suit :

$$Error = 1 - Précision.$$

Nous rappelons que certaines problématiques telles que la détection de chute présentent un déséquilibre dans les données de classes, ou le nombre de séquences présentant des activités quotidiennes normales dépasse le nombre de séquence représentant les chutes (unbalanced problems). Pour obtenir un taux de reconnaissance non biaisé, nous avons mené une procédure de validation croisée à trois niveaux (k-fold-cross-validation). L'ensemble des échantillons représentant chaque classe est partitionné au hasard en 3 sous-ensembles. Uniquement un seul sous-ensemble est réservé pour les tests et les deux autres sous-ensembles sont gardés pour l'apprentissage. Ce processus est répété trois fois, chacun des trois sous-ensembles étant utilisé exactement une fois. Finalement, la valeur finale du taux de reconnaissance est calculée en faisant la moyenne. Puisque dans ce travail, nous avons considéré l'option de rejet après la classification, le pourcentage des cas de données rejetés a été soustrait du taux de reconnaissance finale pour permettre une évaluation équitable. Cependant, la principale limitation du taux de reconnaissance est qu'il ne tient pas compte des classifications correctes obtenues par pure chance. Pour remédier à cette limitation, nous avons aussi calculé d'autres critères d'évaluation [Hand, 2012].

- **Le coefficient de Kappa :**

La deuxième mesure qui peut être extraite à partir de la matrice de confusion est le coefficient de Kappa. C'est une mesure statistique relative à l'accord entre classes. Cette mesure est plus robuste par rapport au taux de reconnaissance puisqu'elle soustrait les cas de classifications correctes obtenus par hasard. Ce coefficient est exprimé comme suit:

$$Kappa = \frac{P(a) - P(e)}{1 - P(e)} \quad (6.10)$$

Où $P(a)$ représente la probabilité d'accord entre classes cette probabilité peut être aussi considérée comme un taux de reconnaissance. La probabilité $P(e)$ représente l'accord entre classes dû au hasard. La plage du coefficient kappa est $[-1, 1]$. La valeur 1 représente une classification parfaite. La valeur 0 indique que l'accord n'est pas meilleur que le hasard, ce qui signifie que le classifieur n'est pas très performant. Un Kappa négatif indiquerait un accord pire que le hasard, ce qui reflète un mauvais résultat de classification (soit un faible algorithme ou la non-correspondance du classifieur avec les données).

- **Le coefficient F-measure :**

Le coefficient F-measure est une autre règle d'évaluation efficace dans le domaine de la classification. Cette mesure combine les deux mesures connues sous le nom de précision et rappel (precision and recall). La précision d'une classe est définie par le rapport entre les séquences correctement classées sur les séquences appartenant à cette classe. De même, le rappel pour une classe est défini comme le nombre de vrais positifs divisé par le nombre total d'éléments qui se rapportent réellement à la classe positive. Le coefficient f-measure représente réellement une moyenne harmonique de la précision (p) et le rappel (r); il est exprimé comme suit:

$$F = (1 + \beta^2) \times \frac{p \times r}{(\beta^2 \times p) + r} \quad (6.11)$$

$$\text{Ou} \quad p = \frac{t_p}{t_p + f_p} \quad (6.12)$$

$$\text{et} \quad r = \frac{t_p}{t_p + f_n} \quad (6.13)$$

Les valeurs de la précision, le rappel et la mesure F varient entre $[0, 1]$. La mesure du coefficient F atteint sa meilleure valeur à 1 lorsque p et r sont simultanément égaux à 1 et cela correspond à une classification parfaite. La constante β est un paramètre positif qui représente le poids assigné aux types d'erreurs. Dans ce travail, le paramètre β est fixé à 1.

- **La courbe ROC**

La fonction d'efficacité du récepteur, connue sous l'appellation anglaise Receiver Operating Characteristic (la courbe ROC), est initialement introduite pour mesurer la caractéristique de performance et de fonctionnement du récepteur. Par la suite, la courbe ROC a été appliquée dans la mesure des performances de méthodes de classifications binaires. La courbe ROC est un outil graphique simple pour illustrer la variation de la sensibilité en fonction de la

spécificité du système. Il est à noter que la sensibilité est relative au taux de vrai positif t_p alors que le terme (1-specificity) est relatif au taux de faux positif f_p . Dans de nombreux domaines d'applications, la courbe ROC reflète plus d'informations sur la qualité d'apprentissage du classifieur qu'un simple taux d'erreur. Afin de quantifier l'évaluation par la courbe ROC en utilisant une seule valeur, la surface sous la courbe ROC a été définie (Area Under Curve) AUC. La variation de l'aire sous la courbe ROC est proportionnelle à la performance du système et fournit un ordre partiel sur les données de tests. En plus, la valeur AUC résout le problème d'évaluation des systèmes présentant un croisement des courbes ROC. Une valeur AUC égale à 1 correspond à une classification idéale tandis que la première bissectrice ou la diagonale principale dans le graphique ROC symbolise une classification par hasard et correspond à un AUC de 0,5. Un classifieur pertinent possède un AUC supérieur à 0.8 [Hand, 2012].

6.4. Résultats et discussion

Comme nous l'avons déjà mentionné précédemment, dans ce travail nous avons évoqué quatre principaux volets à savoir : (i) L'extraction des caractéristiques, ou nous avons proposé l'utilisation des valeurs singulières tronquées, les coefficients de curvelets et les rapports de surface comme attributs pour la silhouette humaine. (ii) La classification de postures, ou plusieurs algorithmes de classification ont été appliqués. (iii) La détection de chutes par le classifieur HMM en utilisant d'abord une seule caméra, dans laquelle une signification physique (représentant les classes de postures humaines) a été attribuée aux états cachés du HMM. Par la suite, la détection de chutes a été appliquée en utilisant des données multi-cameras, ou deux classifieurs HMMs ont été combinés en utilisant la théorie de Dempster-Shafer. (iv) La détection de chutes a été appliquée en utilisant des données d'accéléromètres par la méthode des exponentielles pondérées à moyennes mobiles. Afin d'évaluer chaque étape du processus proposé, nous effectuons les changements et les ajustements uniquement sur la phase à évaluer et en gardant les paramètres des autres opérations fixes. Autrement dit, une évaluation spécifique consiste à choisir les meilleures techniques pour chaque phase de traitement, d'où les changements et les ajustements sont appliqués uniquement sur l'étape à évaluer (choisir les meilleures caractéristiques en fixant la méthode de classification, choisir le meilleur classifieur en utilisant les mêmes caractéristiques, etc.).

6.4.1. Evaluation de l'utilisation des valeurs singulières tronquées comme caractéristiques de la silhouette humaine :

Nous rappelons que l'application de la décomposition en valeurs singulières tronquées était pour sauvegarder les premières valeurs singulières correspondantes à l'information essentielle de l'image. Dans ce travail, le choix du nombre des valeurs singulières a été essentiellement basé sur (i) le compromis entre le rapport d'énergie et le taux de bonne classification et (ii) en considérant plusieurs conditions d'images, plusieurs postures, etc. [Zerrouki, 2014 (2)]. D'où trente valeurs singulières ont été sélectionnées $r' = 30$.

- **Protocoles de tests :**

Afin d'évaluer l'efficacité des valeurs singulières dans la représentation des silhouettes humaines, nous avons extrait près de 4000 images à partir des séquences vidéos constituant les bases de données FDD et URFD. Ces images ont été segmentées puis les bruits ont été éliminés par l'application des opérateurs morphologiques. Dans cette première évaluation, quatre postures ont été considérées à savoir : debout, accroupi, assis et couchée. Les images sélectionnées (4000) comportent l'une des postures mentionnées. On rappelle que l'affectation correcte des images à leurs postures correspondantes est le moyen choisi pour évaluer l'utilisation des valeurs singulières. Pour cela, nous avons utilisé l'algorithme des réseaux de neurones comme méthode de classification. Les images ont été donc partagées au hasard en trois ensembles ; le premier et le deuxième tiers d'images (environ 1300 chacun) sont assignés aux phases d'apprentissage et de validation pour obtenir les paramètres optimaux du réseau de neurones. Tandis que le dernier ensemble (1400 images) est alors assigné à la phase des tests.

TABLEAU 6.1: Evaluation de l'utilisation des valeurs singulières comme caractéristiques de la silhouette humaine [Zerrouki, 2014 (2)] :

Les données classifiées	Les données de références				
	Debout	Assis	accroupi	Couchée	
	Debout	314	0	0	0
	Assis	6	246	34	0
	accroupi	5	79	291	0
	Couchée	0	0	0	325
	Taux de reconnaissance	96.61 %	75.69 %	89.54 %	100 %
Le taux de reconnaissance global (Accuracy) = 90.46 %					

Le tableau 6.1 représente la matrice de confusion obtenue, ou on remarque que les postures debout et couchée ont été correctement classifiées. On peut constater aussi quelques erreurs de classification concernant les postures accroupi et assis, cela est principalement causé par le taux de ressemblance entre ces deux postures.

TABLEAU 6.2: Comparaison de l'efficacité des valeurs singulières dans la caractérisation de la silhouette humaine [Zerrouki, 2014 (2)] avec des caractéristiques proposées dans la littérature :

Méthodes	Les caractéristiques utilisées	Recognition Rate
[Zou, 2008]	Modèle de squelettisation	76.6 %
[Girondel, 2005]	Rectangle circonscrit	88.1 %
[Gkalelis, 2009]	Amplitudes de la transformée de Fourier discrète	90 %
[Zerrouki, 2014(2)]	Les valeurs singulières tronquées	90.46 %

Le tableau 6.2 présente une comparaison de l'efficacité des valeurs singulières dans la caractérisation de la silhouette humaine avec plusieurs caractéristiques proposées dans la littérature. L'idée de caractériser le corps humain avec les valeurs singulières a permis la distinction entre les différentes postures prédéfinies. Nous avons trouvé que le taux de reconnaissance de postures obtenu est acceptable et encourageant pour un premier travail [Zerrouki, 2014 (2)], ou il est supérieur à ceux des travaux présentés dans le tableau 6.2. Comme nous l'avons déjà mentionné, les caractéristiques basées sur les formes géométriques tel que le rectangle circonscrit [Girondel, 2005] présentent quelques limitations, ou elles ne peuvent pas distinguer entre les postures humaines ayant un degré élevé de similitude comme le cas des positions assises et accroupies. La caractérisation par squelettisation [Zou, 2008] présente certaines confusions entre les postures, surtout quand les membres (bras) sont collés au corps ou la silhouette est présentée comme un seul bloc.

6.4.2. Evaluation de l'utilisation des coefficients de Curvelets et les rapports de surfaces comme caractéristiques de la silhouette humaine :

a) utilisation des Coefficients de Curvelets uniquement:

Dans un deuxième lieu, nous avons proposé l'utilisation des curvelets dans la caractérisation du corps humain. Ces coefficients ont été proposés par rapport à leurs invariances (i) à la translation c.-à-d. lorsque la position de la silhouette change dans l'image et (ii) au facteur d'échelle (appelé aussi multi-résolution) c.-à-d. lorsque la dimension de la silhouette change

dans l'image. Ce changement est proportionnel à la distance entre la personne surveillée et l'emplacement de la caméra. Cette invariance nous permet d'éviter l'étape de calibrage préalable des caméras par rapport aux distances obtenues dans l'image de surveillance.

- **Protocoles de tests :**

Dans cette deuxième évaluation, nous avons utilisé pratiquement les mêmes conditions de tests qu'avec celle des valeurs singulières, ou les mêmes classes de postures ont été considérées à savoir : debout, accroupi, assis et couchée. Le nombre d'images sélectionnées est de (4000) comportant l'une des postures mentionnées. Cependant, nous avons utilisé dans cette évaluation l'algorithme des SVM comme méthode de classification avec un noyau Gaussien (Radial Basis Function RBF). Les images ont été aussi partagées au hasard en trois ensembles ; le premier et le deuxième tiers d'images (environ 1300 chacun) sont assignés aux phases d'apprentissage et de validation pour obtenir les paramètres optimaux du classifieur SVM à savoir "Cost" C et "Sigma" σ . Tandis que le dernier ensemble (1400 images) est alors assigné à la phase des tests.

Le tableau 6.3 représente la matrice de confusion obtenue lors de l'utilisation des coefficients de curvelets comme caractéristiques. Les résultats obtenus montrent une amélioration du taux de reconnaissance par rapport à celui présenté dans le tableau 6.1. Cette amélioration est principalement liée aux capacités des curvelets d'obtenir une meilleure information directionnelle et de représenter les contours et les singularités. Nous rappelons que nous avons appliqué des transformés de curvelets à l'aide de quatre niveaux d'échelle ($j = 4$) et dans les huit directions ($l = 8$).

TABLEAU 6.3: Evaluation de l'utilisation des coefficients de curvelets pour caractériser la silhouette humaine [Zerrouki, 2014 (1)] :

Les données classifiées	Les données de références (%)				
	Debout	Couchée	Accroupi	Assis	
	Debout	94.24	2	1.73	3.38
	Couchée	0	98	1.08	0
	accroupi	3.89	0	87.99	5.36
	Assis	1.87	0	9.20	91.26
Le taux de reconnaissance global (Accuracy) = 92.87 % Kappa= 0.9 ; F-measure= 0.89					

Cependant, on peut aussi constater à partir du tableau 6.3 quelques cas de confusion entre les postures tel que la position debout avec couchée ou même accroupi avec assis. Ces confusions

sont causées principalement par le fait que les transformés de curvelets sont aussi invariant par rapport à la rotation. L'information liée à la rotation ou à l'orientation de la posture dans l'image est plus que nécessaire dans la phase d'identification. Pour résoudre ce problème, un complément de cinq rapports de surfaces a été proposé en tant que caractéristiques.

b) Utilisation des Coefficients de Curvelets avec les rapports de surfaces:

Le tableau 6.4 représente la matrice de confusion obtenue lors de l'utilisation des coefficients de curvelets avec les rapports de surfaces. A partir de ces résultats, on remarque que la plupart des postures ont été correctement classifiées. Les images de la posture "couchée" ont été parfaitement identifiées. Autrement dit, les résultats obtenus montrent une nette amélioration du taux de reconnaissance global par rapport à celui présenté dans le tableau 6.3.

TABLEAU 6.4: Evaluation de l'utilisation des coefficients de curvelets et les rapports de surfaces pour caractériser la silhouette humaine [Zerrouki, 2014 (1)] :

		Les données de références (%)			
		Debout	Couchée	Accroupi	Assis
Les données classifiées	Debout	96.17	0	1.24	2.89
	Couchée	0	100	0.86	0
	accroupi	2.89	0	89.79	3.98
	Assis	0.92	0	8.11	93.13
	Le taux de reconnaissance global (Accuracy) = 94.77 % Kappa= 0.91 ; F-measure= 0.909				

Cette amélioration est due au rajout des rapports de surfaces dans le vecteur caractéristique. Puisque ces rapports de surfaces sont non seulement invariants par rapport à la translation et au paramètre d'échelle, mais surtout ils prennent en considération l'information liée à la rotation de la silhouette dans l'image [Zerrouki, 2014 (1)]. La figure 6.3 représente des exemples de rapports de surfaces calculés pour plusieurs postures. On remarque que les changements relatifs à la translation et l'échelle n'ont pas influencé les valeurs des rapports calculés. Ce résultat démontre encore que les rapports de surfaces sont suffisamment représentatifs et descriptifs des postures humaines. En plus, leurs simplicités d'utilisation permet la rapidité dans les traitements et les calculs. Comme nous l'avons évoqué à plusieurs reprises, l'avantage majeur des rapports de surfaces reste leurs indifférences par rapport à la translation et au changement d'échelle (multi-résolutions).

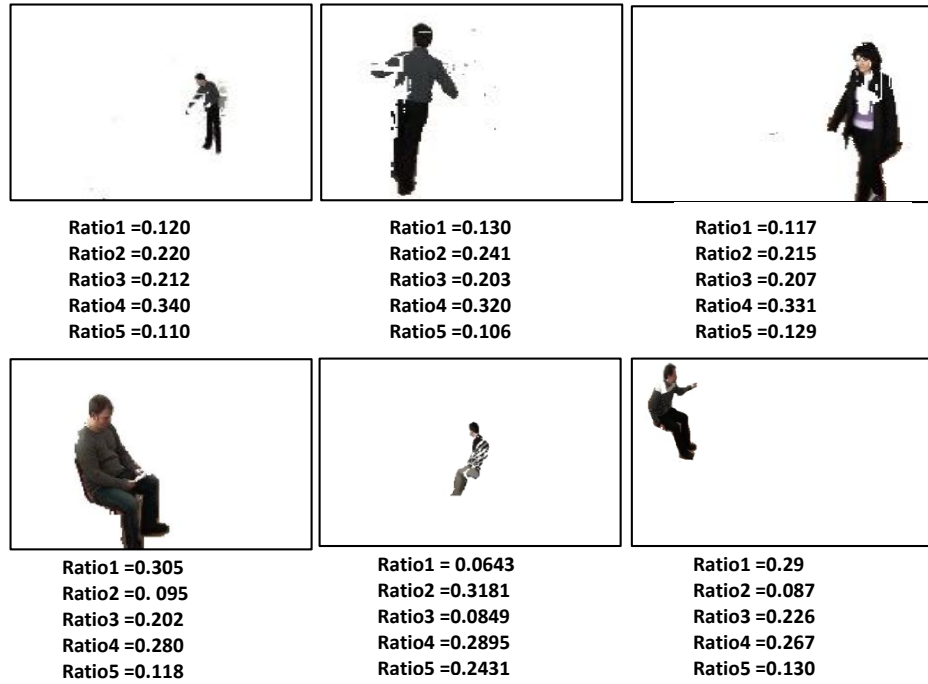


Figure 6.3 Exemples de variation des rapports de surfaces pour deux classes de postures (debout et assise).

La figure 6.3 offre une meilleure illustration de cet avantage, où les valeurs de rapports de surfaces des images représentant les mêmes postures sont proches (intra-classe) et d'une autre part les valeurs de rapports de surfaces des images représentant des postures différentes sont assez distinguées (inter classes).

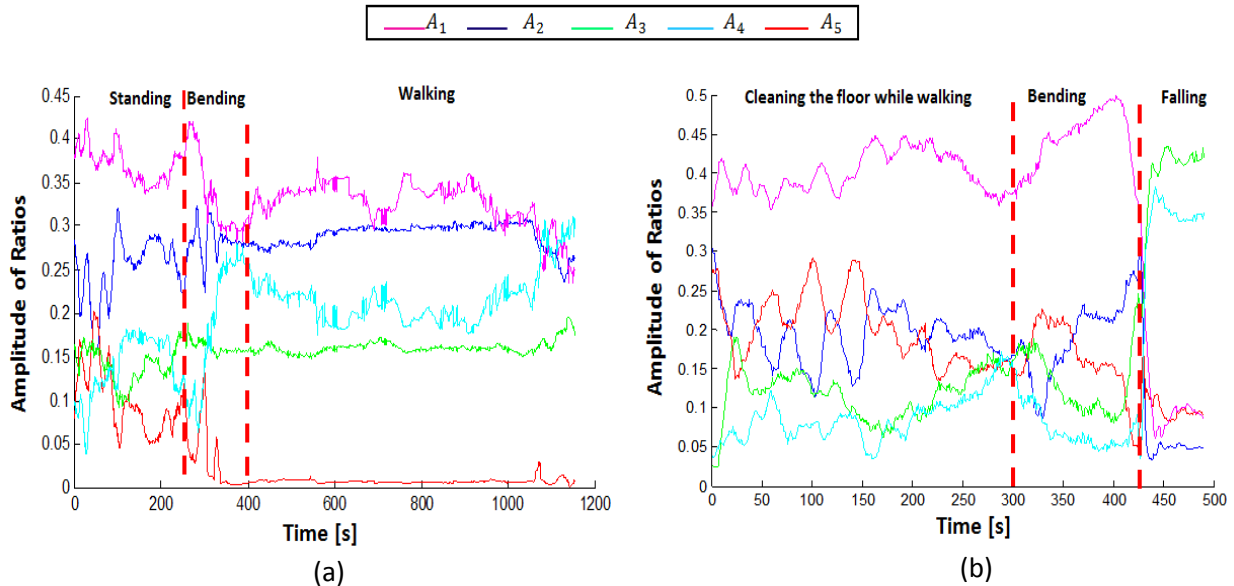


Figure 6.4 Evolution en fonction du temps des rapports de surface pour des activités quotidiennes (a) et pour une chute.

La figure 6.4 présente un exemple de l'évolution dans le temps des cinq caractéristiques ($R_1 - R_5$) tout en effectuant des différentes activités, à savoir des gestes quotidiens (marcher, se pencher, etc.) figure 6.4 (a). La figure 6.4 (b) montre l'évolution durant un incident de chute,

ou nous pouvons constater que ces rapports changent de manière instantanée juste avant et pendant le moment de la chute, signalant l'occurrence d'un événement qui peut être significativement séparé des activités quotidiennes.

6.4.3. Evaluation de la classification de postures humaines :

Afin d'obtenir une évaluation spécifique à chaque étape de traitement, les changements sont effectués uniquement sur les classifieurs et en gardant les mêmes caractéristiques pour toutes les méthodes de classification. Les caractéristiques optimales utilisées dans cette évaluation sont les coefficients de Curvelets et les rapports de surfaces. Comme nous les avons déjà mentionnés dans le chapitre 4, plusieurs algorithmes de classification ont été utilisés à savoir : le plus proche voisin (kNN), naïve Bayes, réseau de neurones, SVM et adaboost. Il est à noter que les performances d'un classifieur dépendent essentiellement du choix de valeurs de ses paramètres. Pour cela, les paramètres de chaque algorithme de classification (utilisé dans ce système ou même durant la comparaison) ont été optimisés pendant la phase d'apprentissage (phase de variation et d'ajustement des paramètres). Les paramètres optimaux doivent correspondre à une précision maximale de classification.

Pour la classification par réseau de neurones, nous avons varié plusieurs paramètres (nombre de neurones dans la couche cachée, etc.) afin de sélectionner l'architecture optimale du perceptron. Une seule couche cachée avec quinze (15) neurones a été sélectionnée, correspondant au meilleur taux de reconnaissance. Pour le classifieur KNN, le choix du nombre de voisins ou la valeur k dépend fortement du type de données. Dans ce travail, nous avons fait varier le paramètre k de 1 à 20 pour établir que $k = 3$ était la valeur optimale. Pour la classification basée sur Naïve Bayes, un modèle gaussien a été adapté comme méthode de distribution. De même, plusieurs valeurs de la moyenne et de l'écart-type ont été utilisées. Les valeurs de la moyenne et de la variance choisies sont égales à 3,43 et 0,38, respectivement. Enfin, nous avons testé itérativement les paramètres du noyau SVM, à savoir σ la largeur du noyau Gaussien et C le paramètre du coût (cost). La paire (σ, C) correspondante au taux de reconnaissance le plus grand a été sélectionnée ($\sigma=0.125$, $C=64$). Pour l'algorithme adaboost, les arbres de décision ont été sélectionnés comme faible classifieur tandis que le nombre d'itérations (ou le nombre de booste) est fixé à $T = 60$.

Les résultats de classification de posture (présentés dans le tableau 6.5) montrent la distinction de plusieurs méthodes de classification (telle que les réseaux de neurone, les supports vecteurs machines et l'algorithme adaboost) par rapport aux deux autres méthodes à savoir kNN et

Naive Bayes. La bonne séparation des postures revient principalement aux aspects d'apprentissage utilisés par chaque algorithme, tel que la considération de la non-linéarité et la non-structuration des données, utilisation des noyaux, etc.

A partir de tableau 6.5, on remarque que la classification par le plus proche voisin a présenté le taux de reconnaissance le plus faible. Ceci en raison de la complexité de l'algorithme kNN, qui calcule les plus proches voisins à chaque itération (c-a-d ré-estimation des distances pour chaque nouvelle posture). La classification basée sur le théorème de Bayes a présenté aussi quelques confusions entre les postures, ou elle a été également surpassé par les autres méthodes. Cela revient principalement à l'hypothèse d'indépendance de cette dernière, ce qui n'est pas toujours valable et même souvent inappropriée pour des données réelles. La classification par les réseaux de neurones a présenté un taux de reconnaissance acceptable.

TABLEAU 6.5: Comparaison des performances entre les différents algorithmes de classification de posture [Zerrouki, 2016 (1)], [Zerrouki, 2017 (2)] et [Zerrouki, 2018].

Méthodes	Taux de reconnaissance (%)	Coefficient Kappa	F-measure
KNN	91.09	0.895	0.91
Naive Bayes	92.21	0.904	0.92
Réseaux de neurones	94.67	0.938	0.94
SVM	95.26	0.908	0.9091
Adaboost	96.59	0.953	0.96

Cela est dû à l'estimation automatique des poids permettant (i) un apprentissage automatique et (ii) des résultats fiables même en présence de données bruitées en entrées du réseau. Le classifieur SVM a surpassé les réseaux de neurones (en termes de taux de bonne classification), car les SVMs utilisent une simple interprétation géométrique dans la séparation des données. Les SVMs définissent la solution en minimisant les risques structurels, contrairement aux réseaux de neurones qui utilisent une minimisation empirique des risques. En plus, les résultats de classification par SVM sont moins sensibles aux changements d'ajustement que les réseaux de neurones (nécessitant une ré-estimation des poids). En dernier, on peut constater que la classification en cascade ou adaboost présente le taux de reconnaissance le plus important. Malgré la simplicité du principe (utilisation des faibles classifieurs : arbre de décision), cette classification offre une bonne séparation de postures. La focalisation de chaque classifieur (arbre de décision) sur une partie des

caractéristiques uniquement, a permis une distinction acceptable entre les classes de postures. On remarque aussi que la classification adaboost est moins sensible au problème de sur-apprentissage, ce qui rend sa mise en œuvre souhaitable pour les applications en temps réel. Cependant, il est à mentionner que le classifieur adaBoost est beaucoup plus sensible aux caractéristiques bruitées et aux données manquantes comparées aux réseaux de neurones et SVMs. La performance de l'algorithme adaboost revient aussi à la caractérisation significative des postures par les coefficients de curvelets et les rapports de surfaces.

6.4.4. Evaluation de la détection de chutes humaines :

a) La détection de chutes en utilisant une seule caméra :

Comme première phase dans la détection de chutes, nous avons utilisé l'algorithme de classification HMM appliqué sur les données issues d'une seule caméra. Comme il a été mentionné dans le chapitre 5, le choix du classifieur HMM a été fortement motivé par son adaptation aux données séquentielles puisque les vidéos sont considérées comme des séquences d'images. En plus, une signification physique (représentant les classes de postures humaines) a été attribuée aux états cachés du HMM.

TABLEAU 6.6: Comparaison des performances entre les différents algorithmes de détection de chutes humaines appliquée sur les bases de données FDD et URFD [Zerrouki, 2017 (1)].

Méthode de détection	Taux de reconnaissance (%)	F-measure	Temps de test (s)
KNN	91.95	0.91	0.22
Naive Bayes	93.56	0.93	0.31
Réseaux de neurones	94.92	0.94	0.15
SVM	95.29	0.95	0.062
SVM-HMM	96.88	0.96	0.12

Pour la définition des états du HMM, nous avons utilisé les supports vecteurs machines pour la classification de postures. Pour la phase de détection, deux classes ont été définies : "chute" et "Non-chute". La classe "chute" représente les incidents de chutes, tandis que la classe "Non-chute" englobe l'ensemble des gestes quotidiens (marché, courir, s'asseoir, etc.). Les tests effectués dans ce travail ont été appliqués sur les deux bases de données à savoir FDD et URFD.

Le point crucial de la classification avec option de rejet reste de trouver le bon compromis entre le taux d'erreur et le taux de rejet. Pour discriminer les pertes entre les décisions de mal-classifications et l'option du rejet, nous avons testé au cours des évaluations différentes valeurs de pertes ω_{ij} . Nous avons fixé $\omega = 0,3$, $\omega_{12} = 0,7$ et $\omega_{21} = 0,5$, les taux d'erreur et de rejet correspondants sont de 0,02% et 0,016%, respectivement. Le pourcentage des cas rejetés est soustrait du taux de reconnaissance globale, pour permettre une évaluation significative.

Le tableau 6.6 représente une comparaison entre la méthode proposée dans ce travail **SVM-HMM** avec plusieurs méthodes de classifications, utilisées cette fois-ci pour la détection de chutes. Afin de permettre une comparaison équitable, les mêmes séquences vidéo ont été utilisées durant l'évaluation. Cette comparaison comprend le taux de reconnaissance (%), le coefficient F-measure, ainsi que le temps moyen de test [Zerrouki, 2017 (1)].

En termes de précision (taux de détection ou coefficient de F-measure), on peut remarquer à partir du tableau 6.6 que la combinaison SVM-HMM présente le meilleurs taux de détection comparé aux autres méthodes utilisées. Ce résultat reflète la forte adéquation de la méthode proposée avec l'application de détection de chutes. Non seulement l'aspect séquentiel présenté dans l'algorithme HMM a été considéré, en plus une caractérisation significative des chutes a été proposée (par l'enchaînement des postures représentant les transitions entre les états du modèle HMM). Cet enchaînement a été utile également dans la réduction du taux de fausses alarmes. Par exemple, se lever en position debout après avoir raté une marche n'est pas considéré comme une chute réelle. Cette distinction est due aux probabilités d'émissions qui diffèrent entre les deux scénarios. Par contre, le cas d'une chute réelle est caractérisé par une probabilité d'émission importante car la personne reste dans la position couchée.

En termes de temps de calcul, dans ce type d'application le temps d'apprentissage n'est pas très important par rapport au temps de test. Puisque l'apprentissage n'est pas effectué en temps réel; contrairement à la phase de test qui se fait en temps réel. Pour cela le temps de calcul moyen (durant la phase de test) de chaque méthode de détection est calculé et rapporté dans le tableau 6.6. A l'exception de l'algorithme kNN, qui ne possède pas de phase d'entraînement indépendante, d'où le temps calculé peut être considéré comme le temps de calcul global pour les deux phases : apprentissage et de test. Selon le tableau 6.6, le classifieur réseaux de neurones présente un temps de traitement inférieur à celui basé sur Naive Bayes et kNN, tandis que le classifieur SVM est considéré comme l'algorithme le plus rapide pour cette application. Cependant, comparé à toutes ces méthodes de détection, la

combinaison SVM-HMM reste nettement meilleure, car elle présente le meilleurs compromis avec la plus grande précision et un temps de calcul acceptable.

b) La détection de chutes en combinant deux caméras par Dempster Shafer Theory of Evidence (DSTE):

Après la détection de chute en utilisant les SVM-HMMs comme méthode de classification, nous avons procédé par la suite, à la détection de chute en utilisant plusieurs caméras. Cette approche était motivée par des cas de postures non reconnues et de chutes non détectées, ou le corps humain n'était pas complètement couvert par la caméra comme le montre la figure 6.5.



Figure 6.5 Exemples de mauvaise-classification de postures et de chutes causées par la visualisation partielle du corps.

L'approche adoptée dans beaucoup de travaux de recherches, représente les données multi-cameras par un seul vecteur caractéristique classifié ensuite par un seul algorithme. Cette approche présente plusieurs limites, ou la défaillance d'une caméra influe négativement sur le vecteur caractéristique (comme le montre la figure 6.6 : première et deuxième colonnes). En plus, les situations de conflits entre les caractéristiques issues des différentes caméras ne peuvent pas être gérées par le classifieur unique.

Comme nous l'avons déjà évoqué dans le chapitre 5, nous avons proposé une combinaison de deux caméras en utilisant deux classifieurs HMMs séparés et la théorie de l'évidence de Dempster Shafer (DSTE) comme méthode de combinaison.

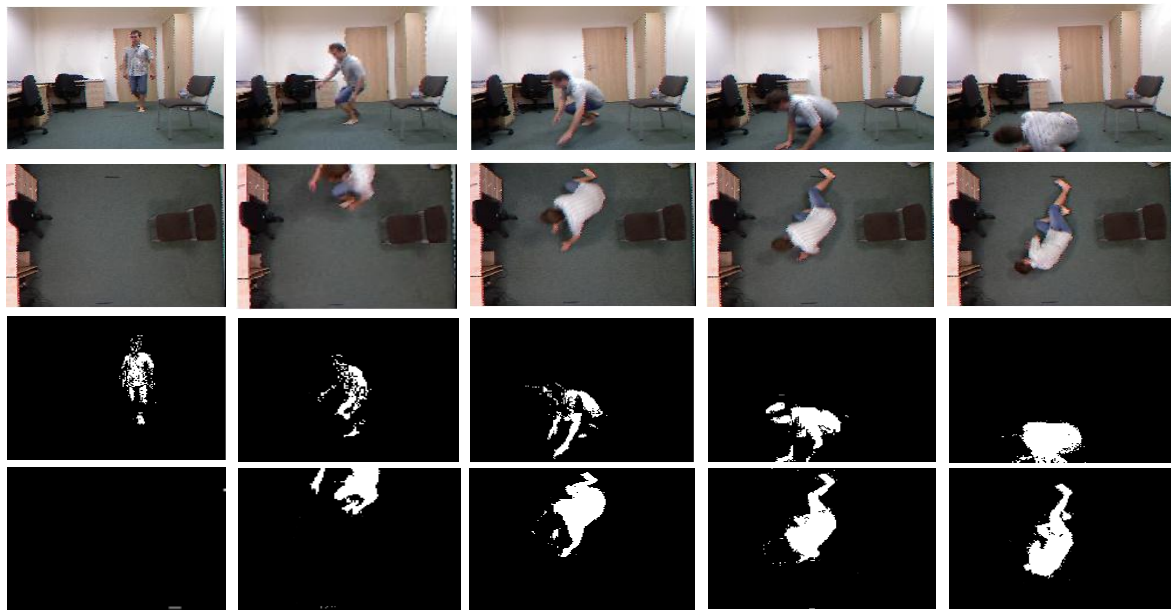


Figure 6.6 Exemples de séquence de chute en utilisant deux caméras.

Le Tableau 6.7 présente les performances des quatre méthodes de détection : utilisation de chaque caméra séparément, utilisation des deux caméras avec un classificateur HMM unique, puis la combinaison des deux HMMs liés aux deux caméras par DSTE.

TABLEAU 6.7: Comparaison des méthodes de détection de chutes en utilisant les caméras séparément et leur combinaison avec DSTE.

Méthode de détection	Taux de reconnaissance (%)
Classification en utilisant la caméra placée au plafond seulement	89.52
Classification en utilisant la caméra placée parallèles au plancher seulement	96.88
Classification en utilisant les données des deux caméras sans combinaison (un seul HMM)	92.38
Classification en combinant les deux cameras avec DSTE	97.29

Les résultats obtenus démontrent la robustesse du formalisme de la combinaison DSTE par rapport aux autres classifications séparées, en particulier pour les situations présentant des données manquantes ou des situations de désaccords de vues. Cela révèle encore comment les

informations supplémentaires complétées par la fonction proximité ϕ_{ij} et le degré de croyance (ou plausibilité) b_j permettent de distinguer le classifieur le plus fiable avant la prise de décision. On remarque également que la classification en utilisant la caméra placée parallèles au plancher seulement (96.88%) est plus robuste que celle qui utilise les deux caméras sans combinaison (92.38%). Ce résultat confirme la limite de (i) la représentation des données multi-caméras par un seul vecteur caractéristique (ou la défaillance d'une caméra influe directement sur la détection) et (ii) l'utilisation d'un seul HMM pendant la détection (ou les situations de conflits ne sont pas gérées par le classifieur).

Une seconde évaluation consiste à comparer la méthode de détection en utilisant la combinaison par DSTE avec d'autres méthodes de combinaison telles que le vote de majorité (majority vote), le modèle de décision (Template decision), la moyenne (Average) et Bayes. Afin d'obtenir une comparaison équitable, les mêmes sorties de classifieurs HMMs ont été utilisées pour toutes les méthodes de combinaison. On peut constater (suivant le Tableau 6.8) que le résultat obtenu en utilisant DSTE est meilleur par rapport aux autres méthodes de combinaison. Le fondement de vote de majorité est le moins précis dans la détection de chute, cela revient au nombre de classifieurs utilisés (uniquement deux) ce qui est inapproprié pour cette combinaison. Plus il y a de classifieurs, plus le vote de majorité converge vers la classe correcte. La combinaison basée sur la moyenne a enregistré aussi des confusions entre les deux classes.

TABLEAU 6.8 Comparaison de l'approche proposée avec des méthodes de combinaisons

	Méthodes de combinaison				
	Vote de majorité	Moyenne	Bayes	Modèle de décision	DSTE
Taux de détection (%)	90.48	92.05	93.10	92.24	97.29

Ces confusions sont causées principalement par les cas de conflit entre les classifieurs HMMs. A l'encontre de la théorie de Dempster-Shafer qui permet de spécifier un degré d'ignorance dans les situations de conflit au lieu de fournir des probabilités antérieures qui influent la décision finale.

c) La détection de chutes en utilisant les données capteurs avec la caméra :

La troisième partie de ce travail consiste à la détection de chutes en utilisant les données d'accéléromètres avec les séquences vidéo. Comme il a été mentionné dans le chapitre 5, cette démarche a été fortement motivée par l'incapacité de détection dans les environnements nocturnes en utilisant les images de caméras couleurs RGB. D'une autre part, la disponibilité de données de capteurs (correspondantes aux séquences vidéo) nous a aussi encouragés à les exploiter.

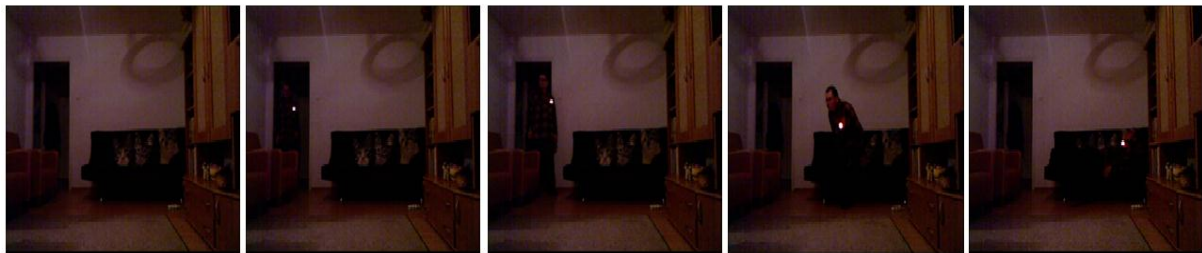


Figure 6.7 Exemples de séquence de chute non-détectée à cause des conditions d'éclairage.

Nous rappelons que la technique basée sur les exponentielles pondérées à moyennes mobiles EWMA identifie la chute suivant les amplitudes des signaux fournis par les accéléromètres. Par la suite, le système classe la séquence vidéo correspondante afin de confirmer l'événement de chute. Par contre, les gestes quotidiens sont directement affectés à la classe 'non-chute' pendant la phase de détection EWMA.

TABEAU 6.9: Comparaison des performances de EWMA-SVM avec les différents algorithmes de détection [Zerrouki, 2016 (1)].

Méthode de détection	Taux de reconnaissance (%)	F-measure	Temps de test (s)	AUC
KNN	91.95	0.91	0.22	0.93
Naive Bayes	93.56	0.93	0.31	0.94
Réseaux de neurones	94.92	0.94	0.15	0.95
SVM	95.29	0.95	0.062	0.95
EWMA-SVM	96.37	0.96	0.0207	0.96

Le tableau 6.9 présente une comparaison entre la méthode **EWMA-SVM** proposée dans ce travail avec plusieurs méthodes de détection de chutes. Cette comparaison comprend le taux

de reconnaissance (%), le coefficient F-measure, le temps moyen de test, ainsi que l'aire sous la courbe ROC (AUC) [Zerrouki, 2016 (1)].

En termes de précision (taux de détection, coefficient de F-measure, ou AUC), on peut constater que EWMA-SVM présente un meilleurs taux de détection comparé aux autres classifieurs (appliqués seuls, sans phase de détection) en incluant l'algorithme SVM. Ce résultat indique que la phase de détection a permis la distinction entre les activités quotidiennes et les chutes, en réduisant le nombre de cas concernés par la classification SVM. Pour la méthode EWMA-SVM la classification SVM est appliquée uniquement pour distinguer entre les chutes réelles et les fausses détections. Par contre, pour les autres méthodes de classification, le classifieur est destiné à l'ensemble des cas. Cette réduction d'espace (pour la méthode EWMA-SVM) a permis l'obtention d'une meilleure séparabilité pendant la phase de classification SVM. Cela se reflète clairement sur les temps de calcul, ou on constate que EWMA-SVM est considérée comme la détection la plus rapide pour cette application. Bien que les classifications SVM ou par réseaux de neurones présentent des temps de traitement acceptables comparés à celui des classifications Bayésienne ou KNN, la détection EWMA-SVM reste nettement plus rapide avec un degré de précision plus élevé.

6.4.5. Comparaisons des méthodes proposées avec des systèmes de détection existants :

Pour finir et afin de mesurer la contribution réelle du présent travail, des comparaisons avec des systèmes existants conduits sur les mêmes bases de données FDD et URFD sont rapportés dans les tableaux 6.10 et 6.11, respectivement. Plusieurs algorithmes ont été utilisés, à savoir: les réseaux de neurones [Alhimale, 2014], [Anderson, 2006] et [Ma, 2014], les méthodes de seuillage [Bourke, 2007], KNN [Kwolek, 2015], SVM [Kwolek, 2014], [Kwolek, 2015] et [Yun, 2015], DAGSVM [Yu, 2013]. A partir des résultats présentés dans les deux tableaux 6.10 et 6.11, on peut remarquer l'efficacité des systèmes proposés dans la détection de chutes même dans les situations compliquées et les scénarios critiques. De plus, on peut noter aussi que la plupart des événements similaires aux chutes (comme s'allonger intentionnellement) sont correctement classés comme non-chute.

La combinaison SVM-HMM a surpassé la plupart des méthodes de classification, puisque l'information du temps est considérée dans le formalisme HMM. En particulier, la durée de la transition entre une posture donnée (debout, accroupi, etc.) et la position couchée traduite par les probabilités de transition dans le classifieur HMM. Ces probabilités aident aussi à distinguer les chutes réelles des activités quotidiennes similaires d'où réduction du nombre de

fausses alarmes. Contrairement à la plupart des méthodes utilisées dans la comparaison (NN, KNN et SVM), ou l'aspect séquentiel n'est pas considéré. Notons par ailleurs que la classification HMM permet une modélisation plus puissante lorsque les états du modèle sont partiellement observables [Alpaydin, 2014]. Dans ce travail, la phase de classification de postures (en utilisant SVM) utilisée dans la définition des états du modèle HMM a joué un rôle clé dans l'amélioration du taux de détection.

TABEAU 6.10: Comparaison des performances du système de détection de chute proposé appliqué sur la base de données FDD.

	Les caractéristiques utilisées	L'approche utilisée	Taux de reconnaissance (%)
[Alhimale, 2014]	Largeur et hauteur du rectangle circonscrit (Bounding box)	Réseau de neurones	94.27
[Ma, 2014]	Caractéristiques du flux de silhouette	Réseau de neurones (VPSO-NEURAL NETWORK)	92.17
[Bourke, 2007]	Valeurs de crête supérieures et inférieures	Seuil de chute supérieure (Upper fall threshold UFT)	95
		Seuil de chute inférieure (Lower fall threshold UFT)	86.67
[Yu, 2013]	Histogramme de projection et les axes de l'ellipse	DAGSVM	96.09
[Zerrouki, 2017 (1)]	Coefficients de Curvelets et rapports de surfaces	SVM-HMM	97.02

D'autre part, l'approche EWMA-SVM a présenté des taux de reconnaissance appréciables malgré que l'aspect séquentiel ne soit pas pris en considération. Cependant, l'application d'une phase de détection EWMA sur les données accéléromètres a surpassé cette limitation. La classification SVM est utilisée uniquement pour les cas détectés, ce qui a justifié la réduction du nombre de fausses détections.

En outre, les taux de reconnaissances obtenus, soit en utilisant SVM-HMM ou bien EWMA-SVM, démontrent encore une fois l'efficacité des caractéristiques utilisées dans ce travail à savoir les coefficients de curvelets et les rapports de surfaces. Surtout que ces caractéristiques s'adaptent avec les variabilités par rapport à la translation et l'échelle (multi-résolutions). Ces caractéristiques offrent aussi une description significative des incidents de chutes, facilitant leurs distinctions des activités quotidiennes.

On remarque aux deux tableaux (6.10 et 6.11), que la méthode de détection SVM-HMM a présenté un meilleurs taux de reconnaissance (97.02 %) pour la base de données FDD, par rapport à la base URFD (96.88 %). Cela est dû à la présence (dans la base URFD) de (i)

plusieurs séquences d'allongement intentionnel et (ii) des acquisitions dans des conditions d'éclairage dégradées. Puisque l'approche SVM-HMM est entièrement basée sur les images couleurs, ou il sera difficile d'extraire la silhouette humaine dans un environnement sombre. Par ailleurs, il est à mentionner que la méthode proposée dans [Kwolek, 2014] (en utilisant SVM) a permis la meilleurs détection avec un taux de reconnaissance de 98.33%. Ce meilleur résultat n'est pas dû uniquement aux méthodes choisies mais aussi aux types de données utilisées. Puisque dans ce travail, plusieurs sources de données ont été utilisées, à savoir : les images de profondeur (depth images), les données d'accéléromètres ainsi que les images couleurs. Cependant, le traitement avec toutes ces données à la fois peut être plus intensif en termes de calcul (d'où le temps de calcul), ce qui constitue une limitation pour les applications en temps réel.

TABLEAU 6.11: Comparaison des performances du système de détection de chute proposé appliqué sur la base de données URFD.

	Les caractéristiques utilisées	L'approche utilisée	Les données utilisées	Taux de reconnaissance (%)
[Kwolek, 2014]	Coefficients de la transformée de Hough	SVM	Les séquences vidéo couleurs (RGB)	90.00
		SVM	Les séquences vidéo couleurs (RGB), les données accéléromètre et les images de profondeur	98.33
[Kwolek, 2015]	Caractéristiques de profondeur et le descripteur P40	SVM	Les séquences vidéo couleurs (RGB), les données accéléromètre et les images de profondeur	94.28
		KNN	Les séquences vidéo couleurs (RGB), les données accéléromètre et les images de profondeur	95.71
[Yun, 2015]	Statistiques de vitesse et les distances géodésiques	SVM	Les séquences vidéo couleurs (RGB) et les images de profondeur (depth image)	96.77
[Zerrouki, 2016 (1)]	Rapports de surfaces	EWMA-SVM	Les séquences vidéo couleurs (RGB), les données accéléromètre,	96.37
[Zerrouki, 2017 (1)]	Coefficients de Curvettes et rapports de surfaces	SVM-HMM	Uniquement les séquences couleurs (RGB)	96.88

Au final, on peut dire que les résultats obtenus (par SVM-HMM ou EWMA-SVM) sont satisfaisants, ou ils ont surpassé la plupart des méthodes de détection de chute existantes et appliquées sur les même base de données (FDD et URFD) :[Alhimale, 2014], [Anderson, 2006], [Ma, 2014], [Bourke, 2007], [Kwolek, 2015], [Yun, 2015] et [Yu, 2013]. Cette

observation confirme que les formalismes proposés (SVM-HMM ou EWMA-SVM) présentent des compromis entre les taux de reconnaissance et les temps de calcul très intéressants ce qui les révèlent très adaptés à l'application de détection de chute que d'autres méthodes de classification.

6.5. Conclusion

Dans ce chapitre, nous avons présenté les différents résultats obtenus dans ce travail. Durant la phase des évaluations nous avons essayé de prendre en considération l'ensemble des facteurs influant sur la robustesse du système tels que les conditions d'éclairage, les images dégradées, les erreurs de segmentation, les paramètres des modèles mathématiques, les taux de reconnaissances, les temps de calcul, les conditions du rejet, etc.

Nous avons aussi analysé les différentes évaluations, en discutant et en justifiant les résultats obtenus ou nous avons montré (i) l'apport et l'utilité de chaque source de données utilisée (données mono-caméra, multi-caméras et capteurs), (ii) les limites et les avantages de chaque modèles utilisé pour la reconnaissance de postures et la détection de chutes, (iii) l'apport des contributions proposées, (caractéristiques et méthodes de détection), dans l'amélioration de la détection des incidents de chutes.

Conclusion générale

Un système automatique de détection de chute est un outil très important pour les personnes âgées afin d'accélérer et améliorer les soins médicaux. En plus, il contribue à fournir des soins médicaux immédiats, d'où résoudre le manque de main-d'œuvre dans le secteur de santé. Dans ce travail, nous nous sommes intéressés à la reconnaissance d'actions et à la détection de chutes à partir de séquences vidéo. Pour traiter ce problème, nous avons développé une approche de reconnaissance séquentielle fondée sur les techniques de machines d'apprentissage. Le système de reconnaissance se résume en quatre phases principales. La première concerne la segmentation de la silhouette humaine à partir des séquences d'images. La seconde phase permet l'extraction des caractéristiques offrant une description significative de la silhouette. La troisième phase a pour but de classifier les postures humaines sur chaque frame séparément. Enfin, la dernière phase permet la distinction entre les chutes et les différentes activités quotidiennes.

Les principales contributions de ce travail de thèse consistaient à aborder certains points essentiels, dans la description de la posture dans les images et sa classification ainsi que la détection des incidents de chutes dans les séquences vidéo.

En premier lieu, nous avons proposé l'utilisation de plusieurs paramètres dans la caractérisation du corps humain à savoir les valeurs singulières tronquées, les coefficients de curvelets et les rapports de surfaces. Les meilleurs résultats de reconnaissance ont été obtenus en combinant les coefficients de curvelets sélectionnés par la technique d'évolution différentielle et les rapports de surfaces. Ces caractéristiques ont été choisies pour leurs invariabilités par rapport à la translation et à la l'échelle dans l'image. Ceci rend le système proposé adapté à n'importe quel type de vidéo surveillance couleur (RGB camera), en évitant le recalibrage préalable des caméras.

Pour la phase de classification de posture, nous avons appliqué plusieurs algorithmes de machines d'apprentissage, réseaux de neurones, Adaboost, séparateur à vaste marge, afin de mieux distinguer les différentes postures. Cette classification de posture est vue comme une étape intermédiaire qui permet par la suite la définition des états cachés des modèles de Markov (HMM classification) utilisés dans la détection de chutes. En plus de l'aspect séquentiel des HMMs, l'attribution des classes de postures aux états du modèle a permis de donner un sens physique aux états HMM et d'obtenir de meilleurs résultats de détection comparés aux méthodes classiques dans lesquels les états des HMMs étaient cachés.

Conclusion et perspectives

Nous avons aussi introduit les méthodes statistiques basées sur les exponentielles pondérées à moyennes mobiles (EWMA) dans la phase de détection, ou elles étaient appliquées sur les signaux des accéléromètres.

Il est important de noter que suivant le principe du ‘‘No Free Lunch’’, il n’y a pas en réalité d’algorithme unique qui, dans un domaine quelconque, induit toujours à la détection la plus précise. Pour cette raison, nous avons utilisé des données de source multiples à savoir accéléromètres, caméra, puis un système de multi-caméras. Les approches proposées dans ce travail ont été évaluées significativement, ou des tests sur des bases de données réelles ont été effectués. Ces données sont jugées motivantes car plusieurs challenges sont présentés et considérés dans ces séquences de vidéo-surveillance. Les approches proposées ont bien permis la séparation entre les chutes réelles et les autres activités même les activités proches ou similaires (s’allonger), ce qui a permis une réduction considérable du taux de fausses alarmes.

Malgré les résultats prometteurs obtenus par les travaux menés dans cette thèse, il y a encore des cas qui soulèvent un certain nombre de questions et fournissent plusieurs orientations pour les travaux futurs. En particulier, les points suivants méritent d’être pris en compte comme perspectives à ce travail:

- Dans ce travail, la détection de chute est effectuée sur la base de données d’accéléromètres et les séquences vidéo des caméras couleurs (RGB cameras). Dans certaines conditions d’illumination tels que les environnements mal éclairés et sombres, les caméras couleurs (RGB) ne sont pas capables d’extraire la silhouette humaine durant la phase de segmentation. En plus, beaucoup de personnes âgées oublient ou trouvent un inconvénient de porter continuellement des accéléromètres. Pour cela, nous prévoyons l’intégration d’une caméra infrarouge afin de reconnaître les activités même dans les environnements obscurs. Ceci permettrait d’enrichir l’information en entrée du système de classification et d’améliorer significativement la précision de reconnaissance. Notant aussi que l’intégration des caméras infrarouges ne va pas changer notre modélisation. Puisque les images infrarouge seront considérées comme des images en niveaux de gris et les phases de traitement tel que l’extraction des caractéristiques et la classification resteront pratiquement les mêmes mais avec des données en entrée plus informatives.
- Dans l’étape de la reconnaissance d’activités, nous avons considéré uniquement les deux classes disponibles et proposées dans les bases de données, ou on s’est limité sur deux

classes d'activité à savoir "chute" ou "non-chute". Nous proposons d'augmenter le nombre de classes d'activités et de définir des classes de sous-activités associées, afin d'avoir une description plus fine des comportements humain et, en conséquence, améliorer encore la précision du classifieur.

- Il serait aussi intéressant d'étendre la détection de chute effectué dans ce travail à la prévention de l'incident de chute. Même si les systèmes de prédiction du moment de l'incident restent difficiles encore à modéliser, mais on peut ramener ce problème à la prédiction de la dépendance de la personne. Ou il s'agit de détecter les dégradations dans les activités quotidiennes de l'individu, en considérant l'historique des activités du même individu. Ceci nécessite donc la reconnaissance de l'individu à chaque apparition afin de faire la correspondance avec les états de ces activités stockés auparavant sur une période de temps. Ce type d'application est jugé encore difficile car il nécessite l'analyse de quantité de données très importantes ainsi qu'une très grande capacité de calcul pour garantir son fonctionnement en temps réel.
- Nous proposons d'explorer d'autres approches basées sur les Réseaux de Neurones Artificiels Profonds (Deep Neural Networks- DNN). Ces approches font partie des grandes tendances actuelles. Ou ils ont pris de l'ampleur et de l'importance dans le domaine des machines d'apprentissage, surtout dans l'analyse des données volumineuses (Big Data Analysis).
- Concernant les bases de données dédiées à la détection de chutes, la plupart des bases disponibles sont construites par des jeunes chercheurs ou des jeunes bénévoles. Cette catégorie de personnes ne reflète pas significativement les personnes âgées. Pour cela, il serait utile d'incorporer davantage de systèmes de surveillance dans les établissements de santé et les maisons de retraite afin d'utiliser des bases de données réelles et représentant des personnes âgées.

Références

1. [Aase, 1999] S. O. Aase, J. H. Husøy, and P. Waldemar, "A critique of SVD-based image coding systems," in Proc. IEEE International Symposium on Circuits and Systems, 1999, pp. 13-16.
2. [Ageing, 2008] Ageing, W.H.O., and Unit, L.C., WHO global report on falls prevention in older age. World Health Organization, 2008.
3. [Akkouche, 2014] Salima Akkouché, Accidents domestiques Et la politique de prévention ?, Le soir d'algérie. <http://www.lesoirdalgerie.com/articles/2014/05/19/article.php?sid=163664&cid=2>
4. [Alhimale, 2014] Alhimale L, Zedan H, Al-Bayatti A (2014) The ation of an intelligent and video-based fall detection system using a neural network. Appl Soft Comput 18:59–69.
5. [Alpaydin, 2014] Alpaydin, Ethem. Introduction to machine learning. MIT press, 2014.
6. [Alwan, 2006] M. Alwan, P.J. Rajendran, S. Kell, D. Mack, S. Dalal, M. Wolfe et R. Felder. A smart and passive floor-vibration based fall detector for elderly. Dans 2nd Information and Communication Technologies, volume 1, pages 1003–1007, 2006.
7. [Anderson, 2006] Anderson D, Keller JM, Skubic M, Chen X, He Z (2006) Recognizing falls from silhouettes. In: IEEE international conference of engineering in medicine and biology society, pp 6388–6391.
8. [Ashley, 2015] Ashley, J. The Imaginative Universal, Authentically Virtual, 2015.
9. [Barron, 1994] Barron, J. L., Fleet, D. J., & Beauchemin, S. S. (1994). Performance of optical flow techniques. International journal of computer vision, 12(1), 43-77.
10. [Baum, 1970] Baum LE, Petrie T, Soules G, Weiss N (1970) A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains. Ann Math Stat 41(1):164–171.
11. [Blake, 1997] Blake, A., & Isard, M. (1997). The CONDENSATION algorithm-conditional density propagation and applications to visual tracking. In Advances in Neural Information Processing Systems(pp. 361-367).
12. [Bourges, 2012] Pierre Bouges. Gestion de données manquantes dans des cascades de boosting : application à la détection de visages. Université Blaise Pascal - Clermont-Ferrand II, 2012. Français.
13. [Bourke, 2007] Bourke AK, O'Brien JV, Lyons GM (2007) Evaluation of a threshold-based tri-axial accelerometer fall detection algorithm. Gait Posture 26(2):194–199.
14. [Charfi, 2012] Charfi I, Miteran J, Dubois J, Atri M, Tourki R (2012) Definition and performance evaluation of a robust SVM based fall detection solution. In: International conference on signal image technology and internet based systems, pp 218–224.
15. [Cucchiara, 2005] Cucchiara, Rita et al. "Probabilistic posture classification for human-behavior analysis." *IEEE Transactions on systems, man, and cybernetics-Part A: Systems and Humans* 35.1, 42-54, 2005.
16. [Das, 2011] Das, S., Suganthan, P. N., Di_erential evolution: a survey of the state-of-the-art. IEEE transactions on evolutionary computation, 15(1), 4-31, 2011.
17. [Delahoz, 2014] Delahoz, Y., and Labrador, M., Survey on fall detection and fall prevention using wearable and external sensors. Sensors 14(10):19806–19842, 2014.
18. [Deskshare, 2017] Video Monitoring Software for IP Camera, IP Camera Viewer, 2017.
19. [Duda, 2012] Duda, R. O., Hart, P. E., & Stork, D. G. Pattern classification. John Wiley & Sons, 2012.

20. [Foroughi, 2008] Foroughi, H., Aski, B. S., Pourreza, H., Intelligent video surveillance for monitoring fall detection of elderly in home environments, IEEE International Conference on Computer and Information Technology, 219-224, 2008.
21. [Gao, 2004] Gao, J., Hauptmann, A. G., Bharucha, A., & Wactlar, H. D. (2004, August). Dining activity analysis using a hidden markov model. In Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on (Vol. 2, pp. 915-918). IEEE.
22. [Gawi, 2015] Global Age Watch Index 2015: Insight report Published by HelpAge International, PO Box 70156, London WC1A 9GB, UK www.helpage.org info@helpage.org, ISBN 978-1-910743-00-3.
23. [Genovese, 2014] GENOVESE, Mariangela et NAPOLI, Ettore. ASIC and FPGA implementation of the gaussian mixture model algorithm for real-time segmentation of high definition video. IEEE Transactions on very large scale integration (VLSI) systems, 2014, vol. 22, no 3, p. 537-547.
24. [Giannakouris, 2008] Giannakouris, K., Ageing characterises the demographic perspectives of the european societies. Stat. Focus 72: 2008.
25. [Girondel, 2005] Girondel, V., Bonnaud, L., Caplier, A., Rombaut, M. "Belief Theory-Based Classifiers Comparison for Static Human Body Postures Recognition in Video," in Proc. WEC, 2005, pp.237-240.
26. [Gkalelis, 2009] N. Gkalelis , N. Nikolaidis , I. Pitas, "View independent human movement recognition from multi-view video exploiting a circular invariant posture representation," in Proc. IEEE international conference on Multimedia and Expo, 2009, pp. 394-397.
27. [Hand, 2012] Hand DJ (2012) Assessing the performance of classification methods. Int Stat Rev 80(3):400–414.
28. [Haralick, 1987] Haralick, R. M., Sternberg, S. R., & Zhuang, X. (1987). Image analysis using mathematical morphology. IEEE transactions on pattern analysis and machine intelligence, (4), 532-550.
29. [Heinrich, 2010] Heinrich, S., Rapp, K., Rissmann, U., Becker, C., and K"onig, H.-H., Cost of falls in old age: a systematic review. Osteop. Int. 21(6):891–902, 2010.
30. [Hiremath, 2006] Hiremath PS, Shivashankar S, Pujari J (2006)Wavelet based features for color texture classification with application to CBIR. Int J Comput Sci Netw Secur 6(9A):124–133.
31. [Hsu, 2005] Hsu, Y. T., Hsieh, J. W., Kao, H. F., & Liao, H. Y. M. (2005, October). Human behavior analysis using deformable triangulations. In *Multimedia Signal Processing, 2005 IEEE 7th Workshop on* (pp. 1-4). IEEE.
32. [Hunter, 1986] Hunter, J.S., The exponentially weighted moving average. J. Qual. Technol. 18(4):203–210, 1986.
33. [Jean, 2005] F. Jean, R. Bergevin et A.B. Albu. Body tracking in human walk from monocular video sequences. Dans The 2nd Canadian Conference on Computer and Robot Vision, CRV'05, pages 144–151, Mai 2005.
34. [Kwolek, 2014] Kwolek, B., and Kepski, M., Human fall detection on embedded platform using depth maps and wireless accelerometer. Comput. Methods Programs Biomed. 117(3):489–501, 2014.
35. [Kwolek, 2015] Kwolek B., Kepski M. (2015) Improving fall detection by the use of depth sensor and accelerometer. Neurocomputing 168:637–645.
36. [Ladde, 2017] Ladde N et Gillibert B, des familles dénoncent des dysfonctionnements à la maison de retraite. Sud-Ouest Magazine, 24, 2017.

37. [Ma, 2014] Ma X, Wang H, Xue B, Zhou M, Ji B, Li Y (2014) Depth-based human fall detection via shape features and improved extreme learning machine. *IEEE J Biomed Health Inform* 18(6):1915–1922.
38. [Majumdar, 2007] Majumdar, A (2007) Bangla basic character recognition using digital curvelet transform. *J Pattern Recognit Res* 2(1):17–26.
39. [Malathi, 2016] Malathi T, Bhuyan MK, Local Gabor wavelet-based feature extraction and evaluation (2016) International conference on advanced computing, networking and informatics, pp 181–189.
40. [Marouf, 2017] Marouf, A., & Houacine, A. (2017). Improving parametric active contours by using attracting point distance map. *Multimedia Tools and Applications*, 76(10), 12583-12595.
41. [Miaou, 2006] Miaou, S.G., Sung, P.H, Huang, C.Y. A Customized Human Fall Detection System Using Omni-Camera Images and Personal Information, 1st Trans-Disciplinary Conference on Distributed Diagnosis and Home Healthcare (D2H2), pp. 39–42, 2006.
42. [Montgomery, 2005] Montgomery, D.C., Introduction to Statistical Quality Control. New York: Wiley, 2005.
43. [Nguyen, 2015] Nguyen, T. K. H. (2015). *Low power architecture for fall detection system* (Doctoral dissertation, Université Nice Sophia Antipolis).
44. [Rabiner, 1993] Rabiner L, Juang BH (1993) Fundamentals of speech recognition.
45. [Roberts, 1959] Roberts, S.W., Control chart tests based on geometric moving averages. *Technometrics* 1(3):239–250, 1959.
46. [Rougier, 2010] Rougier, C. (2010). Vidéosurveillance intelligente pour la détection de chutes chez les personnes âgées.
47. [Serby, 2004] Serby, D., Meier, E. K., & Van Gool, L. (2004, August). Probabilistic object tracking using multiple features. In *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on* (Vol. 2, pp. 184-187). IEEE.
48. [Shamsuzzaman, 2016] Shamsuzzaman, M., Khoo, M. B. C., Haridy, S. et al. An optimization design of the combined Shewhart-EWMA control chart. *The International Journal of Advanced Manufacturing Technology*, vol. 86, no 5-8, p. 1627-1637, 2016.
49. [Soriano, 2007] Soriano, T., DeCherrie, L., and Thomas, D., Falls in the community-dwelling older adult: a review for primary-care providers. *Clin. Inter. Aging* 2(4):545, 2007.
50. [Starck, 2002] Starck J, Candes EJ, Donoho L (2002) The Curvelet transform for image denoising. *IEEE Trans Image Process* 11:670–684.
51. [Stauffer, 1999] Stauffer, C., & Grimson, W. E. L. (1999). Adaptive background mixture models for real-time tracking. In *Computer Vision and Pattern Recognition, 1999. IEEE Computer Society Conference on*. (Vol. 2, pp. 246-252). IEEE.
52. [Storn, 1997] Storn, R., Price, K., Differential evolution: a simple and efficient heuristic for global optimization over continuous spaces. *Journal of global optimization*, 11(4), 341-359, 1997.
53. [Todd, 2004] Todd, C., and Skelton, D., What are the main risk factors for falls among older people and what are the most effective interventions to prevent these falls? Copenhagen, WHO Regional Office for Europe, 2004.
54. [Ucar, 2016] Ucar A, Demir Y, Guzelis C (2016) A new facial expression recognition based on curvelet transform and online sequential extreme learning machine initialized with spherical clustering. *Neural Comput Appl* 27(1):131–142.
55. [Vapnik, 1995] V. Vapnik, *The Nature of Statistical Learning Theory*. New York: Springer-Verlag, 1995.

56. [Veenman, 2001] Veenman, C. J., Reinders, M. J., & Backer, E. (2001). Resolving motion correspondence for densely moving points. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(1), 54-72.
57. [Wang, 2004] Wang, Yubo et al. "Real time facial expression recognition with adaboost." *Proceedings of the 17th Int Conf on Pattern Recognition*, Vol. 3., 2004.
58. [Wang, 2014] Wang, S., Wu, M., & Xie, Y. (2014, September). An effective segmentation cue for moving object segmentation from a moving camera. In *7th International Symposium on Advanced Optical Manufacturing and Testing Technologies (AOMATT 2014)* (pp. 92841I-92841I). International Society for Optics and Photonics.
59. [Yang, 2013] Yang, Y., & Ramanan, D. (2013). Articulated human detection with flexible mixtures of parts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(12), 2878-2890.
60. [Yilmaz, 2004] Yilmaz, A., Li, X., & Shah, M. (2004). Contour-based object tracking with occlusion handling in video acquired using mobile cameras. *IEEE transactions on pattern analysis and machine intelligence*, 26(11), 1531-1536.
61. [Yu, 2012] Yu, M., Miao, R., Adel, N., Mohsen, S., Wang, L., and Chambers, J., A posture recognition-based fall detection system for monitoring an elderly person in a smart home environment. *IEEE Trans. Inf. Technol. Biomed.* 16(6):1274–1286, 2012.
62. [Yu, 2013] Yu, M., Yu, Y., Rhuma, A., Naqvi, S. M. R., Wang, L., & Chambers, J. A. (2013). An online one class support vector machine-based person-specific fall detection system for monitoring an elderly individual in a room environment. *IEEE journal of biomedical and health informatics*, 17(6), 1002-1014, 2013.
63. [Yun, 2015] Yun Y, Gu IYH (2015) Human fall detection via shape analysis on Riemannian manifolds with applications to elderly care. In: *IEEE international conference on image processing*, pp 3280–3284.
64. [ZAIDENBERG, 2014] ZAIDENBERG, Sofia, BILINSKI, Piotr, et BRÉMOND, François. Towards unsupervised sudden group movement discovery for video surveillance. In : *Computer Vision Theory and Applications (VISAPP)*, 2014 International Conference on. IEEE, 2014. p. 388-395.
65. [Zerrouki, 2013] Zerrouki, N. et Houacine, A. Automatic classification of human body postures based on the truncated SVD. *6th International Conference on Computer Science and Information Technology*, 2013, p. 58-62.
66. [Zerrouki, 2014 (1)] Zerrouki N, Houacine A (2014) Automatic classification of human body postures based on curvelet transform. In: *International conference image analysis and recognition*. Springer International Publishing, pp 329–337.
67. [Zerrouki, 2014 (2)] Zerrouki, N. et Houacine, A. Automatic classification of human body postures based on the truncated SVD. *Journal of Advances in Computer Networks*, 2014, vol. 2, no 1, p. 58-62.
68. [Zerrouki, 2016 (1)] Zerrouki, N., Harrou, F., Sun, Y., & Houacine, A. Accelerometer and camera-based strategy for improved human fall detection. *Journal of medical systems*, 40(12), 284, 2016.
69. [Zerrouki, 2016 (2)] Zerrouki, N, Harrou, F, Sun, Y et Houacine, A. A data-driven monitoring technique for enhanced fall events detection. *IFAC-PapersOnLine*, Elsevier, 2016, vol. 49, no 5, p. 333-338.
70. [Zerrouki, 2016 (3)] Zerrouki, N, Harrou, F, Houacine, A et Sun, Y. Fall detection using supervised machine learning algorithms: A comparative study. In *Modelling, Identification and Control (ICMIC)*, 2016 8th International Conference on. IEEE, 2016. p. 665-670.

71. [Zerrouki, 2017 (1)] Zerrouki, N et Houacine, A. Combined curvelets and hidden Markov models for human fall detection. *Multimedia Tools and Applications*, 2017, p. 1-20.
72. [Zerrouki, 2017 (2)] Zerrouki, N, Harrou, F, Sun, Y, Houacine, A, Adaboost-based algorithm for human action recognition. In *15 th IEEE International Conference on Industrial Informatics (INDIN 2017)*: p. 189-193.
73. [Zerrouki, 2018] Zerrouki, N., Harrou, F., Sun, Y., & Houacine, A. (2018). Vision-based Human Action Classification Using Adaptive Boosting Algorithm. *IEEE Sensors Journal*, 18(12), 5115-5121
74. [Zigel, 2009] Y. Zigel, D. Litvak et I. Gannot. A method for automatic fall detection of elderly people using floor vibrations and sound - proof of concept on human mimicking doll falls. *IEEE Transactions on Biomedical Engineering*, 56(12):2858–2867, 2009.
75. [Zivkovic, 2004] Zivkovic, Zoran. Improved adaptive Gaussian mixture model for background subtraction. In *Proceedings of the 17th IEEE International Conference on Pattern Recognition, 2004. ICPR 2004*. p. 28-31, 2004.
76. [Zou, 2008] C. Zou, Z Liu, “Behavior Classification Method Based on Skeleton Model from Video Images,” in *Proc. IEEE international Conference on Computer Science and Information Technology*, 2008, pp. 309-312.

Travaux scientifiques

Les contributions présentées précédemment dans la section 1.4 et d'autres parties de ce travail de thèse ont fait l'objet des publications et des communications suivantes :

Publications scientifiques :

- Zerrouki, N et Houacine, A. Combined curvelets and hidden Markov models for human fall detection. *Multimedia Tools and Applications*, 2017, p. 1-20.
- Zerrouki, N, Harrou, F, Sun, Y et Houacine, A. Accelerometer and camera-based strategy for improved human fall detection. *Journal of medical systems*, 2016, vol. 40, no 12, p. 284.
- Zerrouki, N., Harrou, F., Sun, Y., & Houacine, A. (2018). Vision-based Human Action Classification Using Adaptive Boosting Algorithm. *IEEE Sensors Journal*, 18(12), 5115-5121.
- Harrou, F., Zerrouki, N., Sun, Y., & Houacine, A. (2017). Vision-based fall detection system for improving safety of elderly people. *IEEE Instrumentation & Measurement Magazine*, 20(6), 49-55.
- Zerrouki, N. et Houacine, A. Automatic classification of human body postures based on the truncated SVD. *Journal of Advances in Computer Networks*, 2014, vol. 2, no 1, p. 58-62.

Communications scientifiques :

Communications scientifiques internationales :

- Zerrouki, N. et Houacine, A. Automatic classification of human body postures based on the truncated SVD. *6th International Conference on Computer Science and Information Technology*, 2013, p. 58-62.
- Zerrouki, N. et Houacine, A. Automatic classification of human body postures based on curvelet transform. In: *International Conference Image Analysis and Recognition*. Springer International Publishing, 2014. p. 329-337.
- Zerrouki, N, Harrou, F, Sun, Y et Houacine, A. A data-driven monitoring technique for enhanced fall events detection. *IFAC-PapersOnLine, Elsevier*, 2016, vol. 49, no 5, p. 333-338.

- Zerrouki, N, Harrou, F, Houacine, A et Sun, Y. Fall detection using supervised machine learning algorithms: A comparative study. In *Modelling, Identification and Control (ICMIC), 2016 8th International Conference on*. IEEE, 2016. p. 665-670.
- Zerrouki, N, Harrou, F, Sun, Y, Houacine, A, Adaboost-based algorithm for human action recognition. In *15 th IEEE International Conference on Industrial Informatics (INDIN 2017)*: p. 189-193.
- Harrou, F., Zerrouki, N., Sun, Y., & Houacine, A. (2016, July). A simple strategy for fall events detection. In *14th IEEE International Conference on Industrial Informatics (INDIN 2016)*: pp. 332-336.

Communications scientifiques nationales :

Zerrouki, N. et Houacine, A. Human Action Recognition using a Multi-class Adaboost Algorithm, Mostaganem, *4th Conference on Image and Signal Processing and their Applications (ISPA'2015) University of Abdelhamid Ibn Badis, Mostaganem, Algeria , September 6,7 2015.*