



Université Mohammed V
Rabat



Faculté des Sciences
Rabat

THESE DE DOCTORAT

Présentée par

Abderrazak IAZZI

Discipline: Sciences de l'ingénieur

Spécialité: Informatique

Contribution à la détection de chute des personnes âgées par vidéosurveillance intelligente: L'analyse de la posture et de l'activité humaine par Machine Learning

Soutenue le 30/10/2021 devant la commission d'examen :

Président :

Mr. Abderrahmane SBIHI PES, Faculté des sciences, Université Ibn Tofail, Kenitra, Maroc

Rapporteurs :

Mr. Ahmed TAMTAOUI PES, Institut National des Postes et Télécommunications, Rabat, Maroc

Mr. Moulay Driss RAHMANI PES, Faculté des science, Université Mohammed 5, Rabat, Maroc

Mr. Youness TABII PH, Ecole Nationale Supérieure d'Informatique et d'Analyse des Systèmes, Université Mohammed 5, Rabat, Maroc

Examineurs :

Mr. Mohammed ELHAS-SOUNI PES, FLSH, Université Mohammed 5, Rabat, Maroc

Mr. Hammadi NAIT-CHARIF PES, Université de Bournemouth, Bournemouth, Royaume-Uni.

Directeur de thèse :

Mr. Rachid OULAD HAJ THAMI PES, Ecole Nationale Supérieure d'Informatique et d'Analyse des Systèmes, Université Mohammed 5, Rabat, Maroc

Mr. Mohammed RZIZA PES, Faculté des sciences, Université Mohammed 5, Rabat, Maroc

Travaux préparés au laboratoire : LRIT-CNRST (URAC-29) de la Faculté des Sciences de Rabat.



REMERCIEMENTS

En préambule à ce mémoire, je souhaite adresser mes remerciements les plus sincères aux personnes qui m'ont apporté leur aide et qui ont contribué à l'élaboration de ce rapport ainsi qu'à la réussite de ce formidable parcours universitaire.

Un travail de thèse ne s'accomplit jamais de manière individuelle mais grâce à l'aide précieuse de plusieurs personnes, qui, chacune dans son domaine, chacune à sa manière, contribue à l'aboutissement de ce projet. Que ce soit par leur savoir-faire, leur aide technique, leurs conseils, leur expérience, leur vision de la recherche, ou, tout simplement, leur soutien moral, leur amour ou leur amitié, toutes ces personnes méritent d'être chaleureusement remerciées.

Tout d'abord, je remercie **Mr. Abderrahmane SBIHI**, Professeur d'Enseignement Supérieur à la Faculté des Sciences, Université Ibn Tofail, Kenitra. Maroc, pour avoir accepté la présidence de ma thèse.

je tiens à remercier également **Mr. Ahmed TAMTAOUI**, Professeur d'Enseignement Supérieur à l'Institut National des Postes et Télécommunications à Rabat, d'avoir accepté d'évaluer ce travail de thèse en tant que rapporteur.

Je tiens à remercier **Mr. Moulay Driss RAHMANI** Professeur d'Enseignement Supérieur à la Faculté des Sciences, Université Mohammed V-Rabat, d'avoir accepté d'évaluer ce travail de thèse en tant que rapporteur.

Je tiens à remercier **Mr. Youness TABII**, Professeur Habilité à l'École Nationale Supérieure d'Informatique et d'Analyse des Systèmes, Université Mohammed V-Rabat, d'avoir accepté d'évaluer ce travail de thèse en tant que rapporteur.

Je remercie **Mr. Mohammed ELHASSOUNI** Professeur d'Enseignement Supérieur à la Faculté des Lettres et des Sciences Humaines, Université Mohammed V – Rabat d'avoir accepté d'examiner ce travail et de bien vouloir me faire part de ses remarques lors de ma soutenance.

Je remercie **Mr. Hammadi NAIT-CHARIF**, Professeur d'Enseignement Supérieur à l'Université de Bournemouth, Bournemouth, Royaume-Uni, d'avoir accepté

d'examiner ce travail et de bien vouloir me faire part de ses remarques lors de ma soutenance.

Je tiens à porter mes remerciements les plus sincères à **Mr. Rachid OULAD HAJ THAMI**, Professeur d'Enseignement Supérieur à l'École Nationale Supérieure d'Informatique et d'Analyse des Systèmes, Université Mohammed V-Rabat, pour son encadrement et son précieux conseils. Je le remercie pour la pertinence et la qualité de ses remarques et commentaires, pour son réactivité et pour son implication tout au long de ces années.

J'adresse également mes remerciements à **Mr. Mohammed RZIZA**, Professeur d'Enseignement Supérieur à la Faculté des Sciences, Université Mohammed V-Rabat, pour sa direction de mes travaux de recherche. Ses connaissances du monde de la recherche et des sciences de l'ingénieur ainsi que son expérience m'ont énormément aidé pour accomplir ce travail. Grâce à ses encouragements, sa pédagogie et ses précieux conseils, il a pu me guider dans le parcours de mes recherches pour la réalisation de ce mémoire. Ce travail n'aurait jamais vu le jour sans ses directions, le suivi qu'il m'a accordé et tout le temps qu'il m'a consacré tout au long de cette thèse. Je lui exprime ma profonde gratitude et le respect que je lui porte.

Je tiens à rendre hommage à feu **Pr. Driss ABOUTAJEDINE** que dieu lui fasse miséricorde, de m'avoir accueilli au sein du Laboratoire de Recherche en Informatique et Télécommunications de la Faculté des sciences de Rabat. Je remercie également **Pr. Mohammed OUADOU**, nouveau directeur du Laboratoire de recherche en Informatique et télécommunications, pour sa direction du laboratoire durant ces dernières années et sa disponibilité.

Mes remerciements vont également à ma famille. Je souhaite exprimer ma plus profonde gratitude à mes chers parents pour leur amour et leurs encouragements continus, à nous, à mes frères, mes soeurs et tout membre de ma famille pour leur soutien et leurs encouragements.

Je remercie aussi mes meilleurs amis pour leurs encouragements, mes collègues doctorants du laboratoire LRIT et toute personne ayant prié pour moi et mon succès.



RÉSUMÉ

Depuis longtemps, la majorité des personnes âgées, vivant à domicile et souvent seules, font face à des situations à risques telles que des chutes. Dans ce contexte, la vidéo-surveillance est une solution innovante qui peut leur permettre de vivre normalement dans un environnement sécurisé. L'idée serait de placer un réseau de caméras dans l'appartement de la personne pour détecter automatiquement une chute. Un système de vidéo-surveillance est en général composé de trois modules.

Le premier consiste qu'à chaque nouvelle image, la personne en mouvement doit être détectée, le deuxième permet de suivre une personne ou une trajectoire tout au long de la séquence vidéo. Enfin, la reconnaissance de comportement permet de détecter un événement suspect et fournit en sortie un résultat correspondant à une prise de décision à savoir si le comportement est normal ou anormal. Parmi les grands problèmes des nouvelles techniques de détection de chute des personnes âgées, le risque d'une fausse alarme lors d'un comportement normal qui peut survenir d'une mauvaise interprétation de ces techniques, d'où le besoin de proposer une nouvelle approche plus cohérente dans notre projet de thèse.

Notre première proposition consiste à utiliser une seule caméra RGB en analysant la forme et le mouvement de la personne durant les activités quotidiennes. La reconnaissance de la chute se fait en première par la reconnaissance de type de l'activité de la personne (s'asseoir, marcher, courir, se coucher, debout, s'accroupir, chuter) en utilisant l'apprentissage automatique, puis la confirmation par l'utilisation de la stratégie de vote et aussi des règles de gestion. Une évaluation de la performance du système proposé est faite à travers en deux scénarios. Le premier scénario consiste à tester l'application sur trois bases de données différentes. Ensuite, en deuxième scénario, nous avons mélangé ces trois bases de données pour créer une confusion. Le résultat obtenu nous montre la capacité de notre système d'être stable et de garder des bons résultats par rapport à l'état de l'art.

La deuxième contribution de notre thèse consiste en premier à la proposition d'un descripteur pour la reconnaissance de la posture de la personne. Ensuite, un algorithme pour la détection de chute est proposé en exploitant l'information de la posture. La chute est détectée quand la posture de la personne est identifiée comme allongée et la personne est totalement ou une grande partie de son corps est dans le sol pendant 2 secondes. La performance du système proposé atteint des bons résultats par rapport à l'état de l'art.

Mots-clés : *Détection de chute ; Personne âgée ; Vidéo-surveillance ; Vision par ordinateur ; Reconnaissance de postures et d'activités ; Machine Learning ; Système intelligent*



ABSTRACT

For a long time, the majority of elderly, living at home and often alone, have been facing risky situations such as falls. In this context, video surveillance is an innovative solution that can allow them to live normally in a secure environment. The idea is to place a network of cameras in the person's apartment to automatically detect a fall. A video surveillance system is usually composed of three modules.

The first one is that at each new image, the moving person must be detected, the second one allows to track a person throughout the video sequence. Finally, the behavior recognition allows to detect a suspicious event and provides an output corresponding to a decision whether the behavior is normal or abnormal. One of the major problems of the new techniques for detecting falls in the elderly is the risk of a false alarm during a normal behavior that can arise from a misinterpretation of these techniques, hence the need to propose a new and more consistent approach in our thesis project.

Our first proposal is to use a single RGB camera by analyzing the shape and movement of the person during daily activities. The recognition of the fall is done by recognizing the type of activity of the person (sitting, walking, running, lying down, standing, crouching, falling) by using machine learnings, then the confirmation by using the voting strategy with the management rules.

The second contribution of our thesis consists first in the proposal of a posture recognition descriptor of the person. Then, an algorithm for fall detection is proposed by exploiting the posture information. The fall is detected when the posture of the person is identified as lying down and the person is totally or a large part of his body is inside the ground for more than 2 seconds. The performance of our two proposals achieves good results compared to the state of the art.

Keywords : *Fall detection ; Elderly ; videosurveillance ; computer vision ; Posture and human activity recognition ; Machine learning ; Intelligent system*

TABLE DES MATIÈRES

Remerciements	i
Résumé	iii
Abstract	v
Liste des notations et abréviations	ix
Liste des figures	xiv
Liste des tableaux	xvi
Liste des algorithmes	xvi
Introduction générale	1
Chapitre 1 : Etat de l'art : généralités sur la vidéosurveillance des personnes âgées	7
1.1 Introduction	7
1.2 Les technologies de la surveillance des personnes âgées	8
1.2.1 Capteurs	8
1.2.1.1 Positionnement des capteurs	9
1.2.2 Caméra vidéo	9
1.2.2.1 Types des caméras	10
1.2.2.2 Catégories des systèmes de vidéo-surveillance	11
1.3 Les challenges et les critères d'évaluation	11
1.3.1 L'environnement et la vie privée de la personne	11
1.3.2 Les critères d'évaluation	13
1.4 Structure générale d'un système de vidéo-surveillance	14
1.4.1 L'extraction de la personne	14
1.4.2 Extraction des caractéristiques	15
1.4.3 Les caractéristiques de la chute	19
1.4.4 Les techniques d'apprentissage (Machine learning)	19
1.4.5 Les techniques d'apprentissage profondi (Deep Learning)	25

Chapitre 2 : Les catégories des systèmes de détection de la chute	30
2.1 Les approches de détection de chute dans la littérature	30
2.1.1 Approches basées sur le mouvement et la forme	32
2.1.2 Approches basées sur la posture	38
2.1.3 Approches basées sur le mouvement de la tête	40
2.1.4 Approches basées sur la variation spatio-Temporel	42
2.1.5 Les méthodes à base de l'apprentissage profondi (deep learning)	44
2.2 Comparaison quantitative des méthodes de détection de chute	46
2.3 Bases de données	50
2.3.1 Introduction	50
2.3.2 Bases de données réalisées à base de la caméra RGB	51
2.3.2.1 Base de données Multi caméra de Rougier (2010)	51
2.3.2.2 Base de données de Chua (2013)	51
2.3.2.3 Base de données de Charfi (2012)	52
2.3.2.4 Base de données de Baldewijns (2016)	53
2.3.3 Bases de données réalisées à base de la caméra de profondeur RGBD	54
2.3.3.1 Base de données SDUFall de Haibo Wang (2017)	54
2.3.3.2 Base de données URFDS de Kwolek (2014)	54
2.3.3.3 Base de données EDF de Zhong Zhang et Vassilis Athitsos (2012)	54
2.3.3.4 Base de données de Planinc (2012)	55
2.3.4 Synthèse	55
2.4 Synthèse et conclusion	56
2.4.1 Synthèse	56
2.4.2 Conclusion	57
Chapitre 3 : Proposition d'un système de détection de chute à base de la forme et de mouvement	59
3.1 Introduction	59
3.2 Problématique et la littérature	60
3.3 L'algorithme du système proposé	61
3.4 Structure du système proposé	63
3.4.1 Détection de la personne et la segmentation du sol	63
3.4.1.1 Extraction du sol	65
3.4.2 Extraction des caractéristiques	66
3.4.2.1 Les caractéristiques de la forme	67
3.4.2.2 Les caractéristiques de mouvement	68
3.4.2.3 Notre descripteur proposé	70
3.4.3 La structure proposée pour la reconnaissance de chute	70
3.4.4 Méthode de segmentation des activités dans la vidéo	71
3.4.5 Méthode proposée pour la reconnaissance de la chute	72
3.5 Expérimentation et analyse de performance	73

3.5.1	Analyse des caractéristiques proposées	74
3.5.2	Évaluation quantitative des caractéristiques proposées	76
3.5.3	Évaluation par scénario de notre approche et la classification	77
3.5.3.1	Évaluation par Scénario	77
3.5.3.2	Comparaison avec les approches dans la littérature	78
3.5.3.3	Résultat de notre système proposé	79
3.5.3.4	Temps d'exécution de notre système	81
3.5.4	Discussion	81
3.6	Conclusion	82
Chapitre 4 : Proposition d'un système de détection de chute à base de la posture		84
4.1	Introduction	84
4.2	L'algorithme du système proposé	87
4.3	Structure du système proposé	88
4.3.1	Extraction de la silhouette de la personne	88
4.3.1.1	Détection des objets mobiles	89
4.3.1.2	Détection de l'ombre	89
4.3.1.3	Extraction de la silhouette de la personne	91
4.3.2	Extraction des caractéristiques de forme proposées	93
4.3.3	Méthode de reconnaissance de la posture de la personne	97
4.3.4	L'analyse et la détection de chute	97
4.4	Expérimentation et discussion	99
4.4.1	l'extraction de la silhouette de la personne	100
4.4.2	Classification des postures	100
4.4.2.1	La base de données des postures	101
4.4.2.2	Evaluation des caractéristiques Locales et Globales	102
4.4.2.3	Évaluation du descripteur de la posture proposée	103
4.4.2.4	Comparaison des méthodes d'extraction des caractéristiques	105
4.4.2.5	Evaluation des classifieurs pour la reconnaissance des postures	107
4.4.3	Résultats et la performance de notre système	107
4.4.3.1	Détection de chute de notre système	107
4.4.3.2	Comparaison avec l'état de l'art	108
4.4.4	Discussion	109
4.5	Conclusion	111
Chapitre 5 : Conclusion et les perspectives		113
5.1	Conclusion	113
5.2	Limites	114
5.3	Perspectives	115
Bibliographie		116

LISTE DES NOTATIONS ET ABRÉVIATIONS

TB	A base de seuillage (Thresholding Based)	47
NA	Pas disponible(Not Available)	47
DL	Apprentissage Approfondi (Deep Learning)	14
ADL	Activités de vie quotidienne (Activity Daily Life	53
CB	CodeBook	xii
AMF	Filtre médian approximatif	16
SGM	Modèle gaussien unique (Single Gaussian model)	16
SVM	Machine à vecteurs de support (Support Vector Machine)	xi
DAG-SVM	Graphe acyclique dirigé de SVM	61
L-SVM	Linear SVM	xiii
NN	Réseau de neurones (Neural Network)	23
DBN	Réseau de croyances profondes (Deep Belief Network)	28
BN	Bayessian Native	39
NB	naïve bayésienne (Naive Bayes)	31
ELM	Extreme Learning Machine	31
CNN	Réseau neuronal convolutif (Convolution Neural Network)	26
RNN	Réseau de neurones récurrents (Recurrent Neural Networks)	26
KNN	K-plus proches voisins (K-Nearest Neighbour)	22
HMM	Modèle de Markov caché (Hidden Markov Model)	31
MCMC	Chaîne de Markov Monte Carlo	72
ANN	Réseau de neurones artificiels (Artificial Neural Network)	31
LSTM	Mémoire à long court-terme (Long Short Time memory)	27
RF	Forêts Aléatoires (Random Forest)	25
DT	Arbre de décision (Decision Tree)	107
BB	Rectangle (Bounding Box)	59
BS	Soustraction d'arrière-plan (Background Subtraction)	xii
GMM	Modèle de mélange gaussien (Gaussian Mixture Model)	16
RBF	Fonction de base radiale (Radial Base Function)	21
MEI	Mouvement-Énergie de l'image (Motion-Energie of Image)	18
INME	Ellipse ajusté et la normalisation d'énergie	42
AG	Algorithme Génétique	33

HOG	Histogramme de gradient orienté	34
HOF	Histogramme de flot optique	34
CSS	Curvature Scale Space	34
FV	Fischer Vector	34
SOV	Orientation de volume de la silhouette	39
RA	Rapport d'Aspect	36
HS	Vitesse de la tête (Head Speed)	36
MPGD	Descripteur géométrique du mouvement de la pose	35
ISTE	Energie Spatio-Temporelle Intégrée	37
MHI	Historique de mouvement des images (Motion History Images)	18
SFA	Analyse des caractéristiques lentes (Slow Features Analysis)	61



LISTE DES FIGURES

LISTE DES TABLEAUX

1.1	La comparaison entre les caméras couleur (RGB) et les caméras de profondeur (RGB-D)	12
1.2	Tableau de Confusion	13
1.3	Les algorithmes de la soustraction de l'arrière-plan pour les systèmes de détection de chute	16
1.4	Résumé des caractéristiques 2D de base pour détecter la chute	17
1.5	Résumé des caractéristiques 3D de base	19
2.1	La comparaison et la performances des systèmes de détection de chutes	46
2.2	Les bases de données et les approches qui ont adopté ces base de données pour évaluer leurs systèmes de la détection de chute	55
3.1	La description des caractéristiques	66
3.2	Les difficultés et les challenges de chaque base de données	74
3.3	Les informations quantitative de la base de données de l'expérimentation	74
3.4	Résultat de l'évaluation des caractéristiques	76
3.5	La performance de la méthode proposé en utilisant le premier scénario	77
3.6	La performance de la méthode proposé en utilisant le 2ème scénario .	77
3.7	Comparaison des différents classifieurs pour deux classes, chute et non-chute	78
3.8	Comparaison avec les méthodes de l'état de l'art	79
3.9	Comparaison du système proposé et des méthodes de l'état de l'art en termes de temps d'exécution	81
4.1	La description et les valeurs des paramètres des seuils	92
4.2	Caractéristiques de chaque base de données	102
4.3	Résultat de la comparaison des caractéristiques locales et globales pour les deux bases de données D1 et D2	103
4.4	Classification des postures pour la base de données D1	103
4.5	Classification des postures pour la base de données D2	103
4.6	Évaluation de nombre de partition pour l'extraction des caractéristiques	105
4.7	Comparaison de différentes méthodes d'extraction des caractéristiques pour la base de données D1	105

4.8	Comparaison de différentes méthodes d'extraction de caractéristiques pour la base de données D2	106
4.9	Comparaison des méthodes d'extraction des caractéristiques	106
4.10	Résultats de notre système de détection de chute proposé	109
4.11	Comparaison avec l'état de l'arts	110



LISTE DES ALGORITHMES

3.1	La stratégie pour détecter la chute	73
4.1	Algorithme proposé pour la projection de l'histogramme	96
4.2	La stratégie de la détection de chute	99

INTRODUCTION GÉNÉRALE

Contexte général

Selon les statistiques mondiales [5, 6] sur l'évolution de la population, la vieillesse a extrêmement augmenté en 21^{ème} siècle. D'après ce rapport, 1 sur 8 était âgée plus que 65 ans en 2001. Grâce aux conditions et des lois appliquées aux familles de ne pas avoir plus que 2 enfants au maximum, en 2026, la proportion deviendra 1 sur 5 qui vont avoir plus que 65 ans. Dans la Figure 1, nous présentons l'évolution de la vieillesse de la population japonaise.

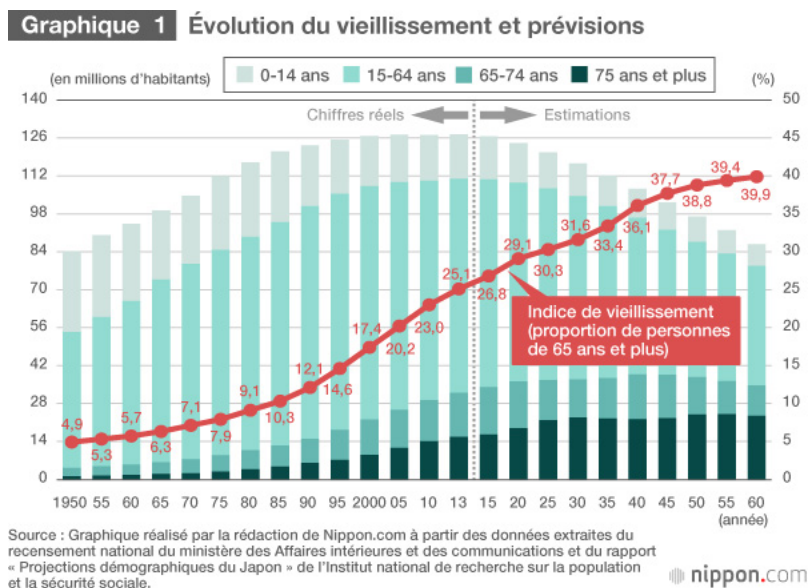


FIGURE 1 – L'évolution de vieillesse de la population mondiale

L'étude réalisée par l'organisation mondiale de la santé (OMS) [6] montre que l'augmentation de l'espérance de vie a un impact sur l'augmentation d'avoir différentes générations vivant en même temps au sein d'une famille. Bien que le nombre de générations vivantes dans une même famille puisse avoir augmenté, aujourd'hui, ces générations sont plus susceptibles de vivre séparément que dans le passé. En effet, dans de nombreux pays, la proportion des personnes âgées vivant seules augmente

de façon spectaculaire. Par exemple, dans certains pays européens, plus de 40% des femmes âgées de 65 ans et plus, vivent seules (figure 2).

En plus, ces personnes âgées sont exposées à des risques comme la chute à cause de leurs conditions de santé. En conséquence, ils ont besoin de les surveiller afin de sauver leur vie dans le cas où la personne âgée est chutée à cause d'une perte de conscience ou avoir une crise cardiaque et personne n'est là pour l'aider. Après la chute, la personne reste sur le sol plusieurs heures et il n'a aucun moyen pour appeler le secours ou quelqu'un qui va l'aider. La croissance des personnes âgées ainsi le nombre de chutes chez eux nous incite à développer des systèmes qui seront un moyen effectif pour les aider à vivre mieux dans un environnement sécurisé.

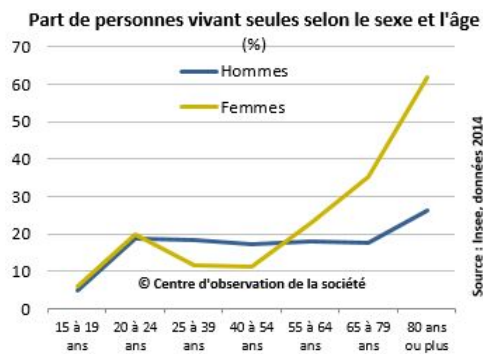


FIGURE 2 – La proportion des personnes vivant seules

Plusieurs études [7, 8, 9] ont montré l'intérêt de maintenir les personnes âgées à domicile d'un point de vue humain, et aussi d'un point de vue financier.

Motivations du travail

La chute reste une problématique de premier plan en raison de sa forte prévalence et incidence couplée aux multiples conséquences délétères qu'elle entraîne, et cela, quelle que soit la personne considérée, qu'il s'agisse du chuteur lui-même ou du décideur de santé publique [10]. Aujourd'hui, l'événement chute est devenu pandémie dans les pays développés, au sens où il affecte toutes les catégories de personnes âgées de 65 ans et plus, quel que soit leur lieu de vie [11].

Au fur et à mesure que le nombre de facteurs de risque augmente, le risque de rechuter encore une fois est devenu complètement important dont il est multiplié par 20 après la première chute, il est donc important d'identifier et définir une chute avant de commencer à la détecter. Dans la littérature, nous trouvons différentes définitions de la chute. Les composants de comportement, biomécanique et typographique sont la combinaison utilisée par la plupart des définitions. La vraie définition de chute est le fait de passer à un niveau inférieur par rapport au niveau de départ d'une manière volontaire [10].

Parmi les plus fréquentes définitions, nous citons, la définition d'un groupe de recherche "Kellogg international working group , Gibson et al., 1987", la chute est bien définie comme suivante : Une chute peut être définie comme involontairement à venir au sol ou d'un niveau inférieur et autre que comme conséquence de maintenir un violent coup, la perte de conscience, l'apparition soudaine de la paralysie que dans un accident vasculaire cérébral ou une crise d'épilepsie. Une autre définition de chute dite "accidentelle" qui est un terme d'origine anglophone « accidental fall ». Cette terminologie a été décrite par le descripteur MeSH dans Medline où il définit la chute comme « tout glissement ou trébuchement aboutissant à une blessure » (*"Falls due to slipping or tripping which result in injury"*). La chute a été définie dans le travail [12] par le fait « tout événement au cours duquel une personne se retrouve, par inadvertance ou intentionnellement, sur le sol ou sur un autre niveau inférieur tel qu'une chaise, toilette ou un lit » (*"any event in which a person inadvertently or intentionally comes to rest on the ground or another lower level such as a chair, toilet or bed"*).

Toutes les définitions commencent d'une base commune dont le fait d'arriver à un niveau inférieur par rapport à la position initiale peu importe le niveau d'arriver. Le sol peut être considéré la référence. En général, il existe plusieurs types de chutes selon plusieurs causes comme suivants :

- La chute dite lourde : perte de verticalité rapide associée à un choc (Figure 3).
- La chute molle : lorsque la personne se retient à un meuble par exemple.
- La chute syncopale (maladie) : lors de la perte de conscience.

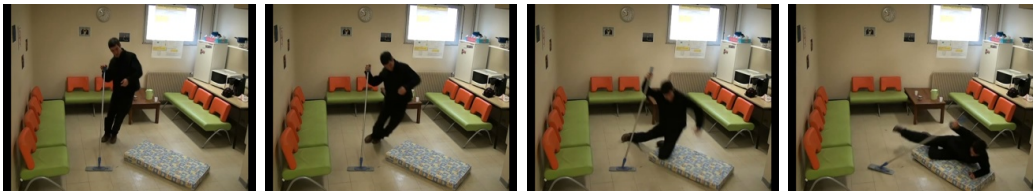


FIGURE 3 – Simulation de la chute lourde

Chaque type parmi ces derniers types de chute peut être divisé en plusieurs catégories en se basant sur différents critères : l'orientation, l'amplitude du mouvement et aussi la transition de la posture. Plus le temps passé au sol est long, plus les conséquences de la chute seront graves. Il est donc primordial de pouvoir détecter une chute pour le maintien de l'autonomie des personnes âgées. Cependant, il est difficile de définir avec précision toutes les chutes, d'autant qu'il convient d'identifier également les fausses alertes. Afin de minimiser les fausses d'alarme, les chercheurs ont essayé d'analyser le comportement de la personne au cours du temps afin de distinguer entre deux types d'activités, les activités anormales qui sont considérées comme des activités non-contrôlées et des activités normales qui sont considérés comme des activités contrôlées. Parmi les caractéristiques qui en distinguent est le mouvement et/ou aussi la durée d'un changement de la posture. Ces caractéristiques sont les plus importantes dans n'importe quel système d'analyse des comportements de la personne.

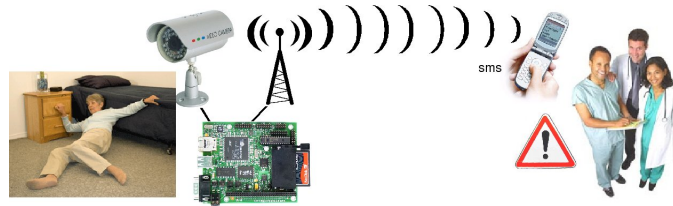


FIGURE 4 – La surveillance des personnes âgées

Structure de thèse

Contributions de la thèse

Les contributions de cette thèse portent sur trois principaux axes de recherche :

Benchmark des systèmes de détection de chute : nous avons proposé une taxonomie des approches de détection de chute en se basant sur le type des caractéristiques ainsi la méthode utilisée pour reconnaître la chute. Nous avons commencé par la discussion des méthodes utilisées pour détecter la personne dans la vidéo, puis, les types des caractéristiques qui décrivent les différents types d'activités humaines, ensuite les techniques utilisées pour la vérification et la confirmation de la chute.

Système de détection de chute à base de la variation de mouvement et de la forme : cette contribution consiste d'abord à la proposition des caractéristiques utilisées pour analyser la forme et le mouvement de la personne pour détecter la chute. Nous avons divisé la forme de la personne en trois régions pour identifier la tête, le centre du corps et les jambes. À travers chaque région, nous avons extrait plusieurs caractéristiques de forme et de mouvement afin d'analyser la variation de ces régions durant la chute et les activités quotidiennes. La classification des activités est utilisée à base de la méthode de Support Vecteur Machine (SVM) pour une reconnaissance automatique.

Système de détection de chute à base de la posture de la personne : Cette contribution consiste à la proposition de la reconnaissance de posture de la personne, puis l'exploitation de l'information de la posture pour la reconnaissance de la chute. Nous avons proposé un nouveau descripteur de forme pour reconnaître la posture de la personne d'une façon automatique en utilisant la classification par SVM. Ensuite, nous proposons un algorithme pour identifier si l'activité de la personne est une chute ou une activité normale.

Publications

Publications internationales indexées

- IAZZI, Abderrazak, RZIZA, Mohammed, et THAMI, Rachid Oulad Haj. **Fall Detection System-Based Posture-Recognition for Indoor Environments**. Journal of Imaging, 2021, vol. 7, no 3, p. 42.
- IAZZI, Abderrazak, RZIZA, Mohammed, et THAMI, Rachid Oulad Haj. **Efficient fall activity recognition by combining shape and motion features**. Computational Visual Media, 2020, vol. 6, no 3, p. 247-263.
- Brahim Alibouch, Abderrazak Iazzi, Amina Radgui and Mohammed Rziza. **"An Adapted Block Thresholding Method for Omnidirectional Image Denoising"**, RJASET, 2014.

Conférences internationales indexées

- Iazzi Abderrazak, Rziza Mohammed, Rachid Oulad HajThami. **Human posture recognition based on projection histogram and Support Vector Machine** . ISIVC 2018, Rabat, Morocco.
- Iazzi Abderrazak, Rziza Mohammed, Rachid Oulad HajThami. **"Fall detection based on posture analysis and support vector machine"**, ATSIP 2018, Sousa, Tunis.
- Iazzi Abderrazak, Rziza Mohammed, Rachid Oulad HajThami. **"A New Method for Fall Detection of Elderly Based on Human Shape and Motion Variation"**, ISVC 2016, Las Vegas, Nevada, USA.
- IAZZI, Abderrazak, THAMI, Rachid Oulad Haj, et RZIZA, Mohammed. **A novel approach to improve background subtraction method for fall detection system**. In : 2015 IEEE/ACS 12th International Conference of Computer Systems and Applications (AICCSA). IEEE, 2015. p. 1-2.Marrakech, Morocco. (poster)

Plan de thèse

Le reste de la thèse est organisé comme suivant :

Dans le **chapitre 1** et **chapitre 2**, nous présentons une étude sur la problématique et l'état de l'art des systèmes de détection de chute pour les personnes âgées. Les différentes techniques utilisées pour analyser et reconnaître la chute. Nous discutons aussi les différentes catégories des approches de détection de chute proposées dans la littérature. Par la suite, nous discutons les bases de données publiques utilisées pour évaluer la performance des systèmes de détection de chute.

Chapitre 3 présente notre contribution à la détection de chute à base de la forme et de mouvement. La déformation de la forme et le changement de la vitesse de la personne caractérisent la chute d'une façon unique. Les caractéristiques de forme et de mouvement sont combinées et utilisées pour distinguer entre les activités normales et les activités anormales. Ensuite, nous présentons les expérimentations effectuées pour évaluer la performance de notre système.

Chapitre 4 présente notre proposition à la détection de chute à base de la

posture de la personne. Nous présentons notre algorithme proposé pour extraire le descripteur de la posture de la personne. Les différentes phases de notre système sont présentées en détail depuis la détection et l'extraction de la personne jusqu'à la détection de chute. Les expérimentations réalisées sur différentes bases de données sont introduites afin d'évaluer la performance de notre système de reconnaissance de la posture et la détection de chute.

Finalement, dans le **chapitre 5**, nous concluons notre rapport par une discussion sur les différentes contributions proposées. Les limites et les perspectives qui seront considérées pour un travail de futur.

ETAT DE L'ART : GÉNÉRALITÉS SUR LA VIDÉOSURVEILLANCE DES PERSONNES ÂGÉES

Sommaire

1.1	Introduction	7
1.2	Les technologies de la surveillance des personnes âgées	8
1.2.1	Capteurs	8
1.2.2	Caméra vidéo	9
1.3	Les challenges et les critères d'évaluation	11
1.3.1	L'environnement et la vie privée de la personne	11
1.3.2	Les critères d'évaluation	13
1.4	Structure générale d'un système de vidéo-surveillance	14
1.4.1	L'extraction de la personne	14
1.4.2	Extraction des caractéristiques	15
1.4.3	Les caractéristiques de la chute	19
1.4.4	Les techniques d'apprentissage (Machine learning)	19
1.4.5	Les techniques d'apprentissage approfondi (Deep Learning)	25

1.1 Introduction

Vu la perte d'autonomie des personnes âgées qui vivent seules à la maison, ils sont face aux nombreux risques. Dans ce chapitre, nous présentons une étude sur les différents types des systèmes existants pour surveiller ces personnes durant leurs activités quotidiennes à la maison ou à l'extérieur. Ces systèmes sont divisés en plusieurs catégories selon le domaine d'application. Nous donnons aussi plus de détails sur l'impact de ces systèmes sur la vie d'une personne âgée de point de vue avantages et négatives.

Durant notre étude, nous avons analysé plusieurs articles des journaux et des conférences. Nous avons basé sur plusieurs bases de données des éditeurs suivants : IEEE Explore, Elsevier, Springer, google Scholar, Pub Med, SPIE, etc. Nous avons suivi une stratégie pour trouver un nombre important d'articles qui contiennent notre sujet où nous avons fixé la date de publication à partir de 2005 jusqu'à 2020. Puis, nous avons commencé notre recherche en utilisant des clés de recherche (Anglais et français) bien précises comme "fall detection", "fall detecter", "fall activities", "fall

event”, ”camera based fall detection”, ” recognition of activities”, ”fall methods”, ” abnormal activities detection”, ”fall recognition”, etc. La majorité des références des articles ont été trouvées à base de la recherche et d’autres en utilisant les citations. Comme résultat, nous avons trouvé un nombre important d’articles qui dépasse 1000 articles et qui proposent des approches à propos de la détection de chute. Pour ne pas avoir des redondances, nous avons analysé et sélectionné seulement les articles qui ont satisfait quelques critères comme suivant :

- Premièrement, nous sélectionnons que les papiers qui sont indexés dans des bases de données de qualité et connues.
- Deuxièmement, nous considérons que les travaux qui présentent des résultats acceptables et aussi des enquêtes pionnières.

Nous avons constaté que les contributions dans ce sujet sont bien remarquables dans ces dernières années. Au cours des années, le nombre des papiers publiés chaque année s’augmente. Nous pouvons justifier cette évolution par l’augmentation des demandes des systèmes performants pour surveiller des personnes âgées et aussi l’augmentation des personnes âgées vivant seules.

1.2 Les technologies de la surveillance des personnes âgées

Dans la littérature, le domaine de la surveillance des personnes âgées est devenu indispensable dans le monde. Les chercheurs ont bénéficié de cette pandémie mondiale pour développer des systèmes de surveillance robustes et faisables pour sauver la vie des personnes âgées. La Figure 1.1 montre les trois catégories des systèmes de détection de chute dont on trouve les systèmes à base des capteurs portables, les systèmes à base de caméra et les systèmes qui regroupent les deux derniers systèmes. Tous ces systèmes utilisent une méthode de la reconnaissance de chute soit à base d’une méthode de seuillage ou d’une méthode à base de l’apprentissage automatique.

1.2.1 Capteurs

Parmi les premiers systèmes proposés pour surveiller les personnes âgées sont les systèmes qui se basent sur les capteurs portés par la personne, soit sont portés directement à son corps soit sont intégrés dans ses vêtements ou sont installés à la maison. Ces capteurs sont utilisés pour capturer plusieurs types d’informations de la personne. Nous avons des capteurs qui permettent de calculer la fréquence cardiaque, la pression artérielle et la température. Aussi, les capteurs qui permettent de calculer la vitesse de la personne. Des capteurs pour surveiller sa respiration. Des capteurs pour savoir sa position dans la maison. Tous ces types de capteurs nous offrent des informations importantes sur l’état physique de la personne ainsi sa position.

Dans la littérature, nous trouvons plusieurs travaux qui se basent sur ces capteurs afin d’offrir un système automatique pour surveiller les personnes âgées. L’objectif est de diminuer le risque et sécuriser leur environnement. Un tel exemple est présenté

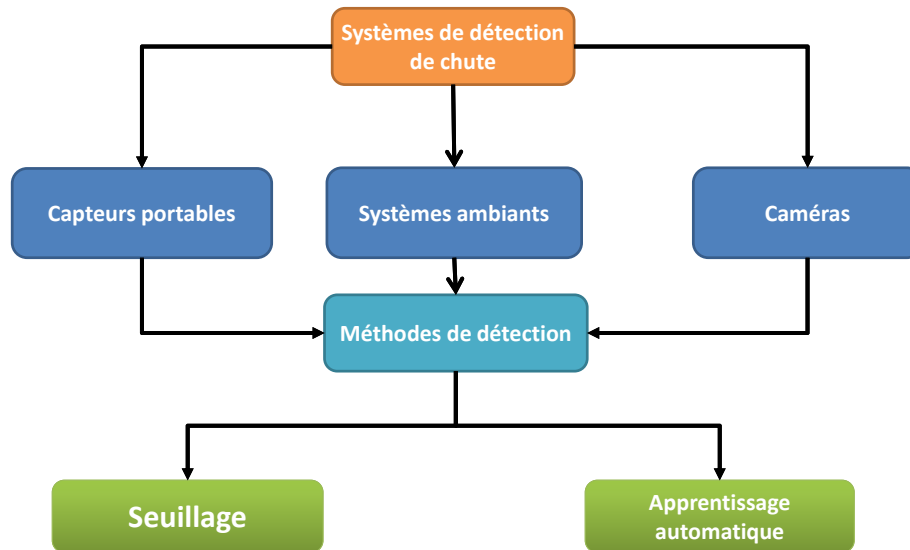


FIGURE 1.1 – Les technologies de la détection de chute.

dans le travail [13] où les auteurs ont utilisé plusieurs capteurs pour collecter plusieurs données pour savoir l'état de la personne en temps réel.

1.2.1.1 Positionnement des capteurs

Dans la littérature, nous trouvons énorme de capteurs qui sont utilisés pour surveiller des personnes à la maison. Le choix d'un capteur dépend généralement de la personne âgée et la position où il sera installé selon l'infrastructure de la maison. Généralement, ces capteurs peuvent être divisés en plusieurs groupes selon leurs positionnement où ils sont installés. On trouve des capteurs qui sont portés par la personne sous forme de bracelet, avec les vêtements ou à travers les chaussures. Il y a aussi des capteurs qui sont installés dans la maison comme le sol.

1.2.2 Caméra vidéo

Dans n'importe quel système de vidéosurveillance et spécialement de détection de chute, le type de caméra joue un rôle important dans le résultat final. Il existe différents types de caméras de vidéosurveillance, à domicile, dans une entreprise ou un commerce. Elles se distinguent par leurs fonctionnalités mais aussi par leur niveau de gamme. Plusieurs travaux ont été proposés pour la vidéosurveillance des personnes âgées en utilisant différents types de caméra où le choix d'un type de caméra est dépendu de plusieurs critères ; la méthode à utiliser, l'environnement à surveiller et la position de la caméra, le champ de vue et n'oubliant pas le cout du système.

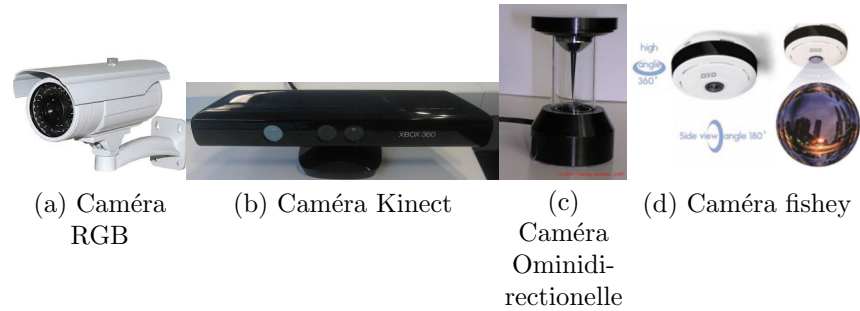


FIGURE 1.2 – Différents types de caméras de vidéosurveillance. a) Caméra RGB, b) Microsoft Kinet caméra, c) caméra catadioptrique (Omnidirectionnelle)

1.2.2.1 Types des caméras

Dans le marché, on trouve des caméras omnidirectionnelles (Fig. 1.2) qui offrent un champ de vue jusqu'à 360 degrés [14], Malheureusement, ce type de caméra est rarement utilisé dans le domaine de détection de chute car elle pose un problème au niveau de la forme de l'image résultante et aussi le coût est très élevé par rapport aux autres caméras. Un autre type de caméra offre aussi un champ de vue égale à 360 degrés est la caméra fish-eye qui se base sure une seule lentille. Cependant, ce type de caméra produit des images de moins de qualité. Le troisième type est la caméra Microsoft Kinect [15] qui est particulièrement utilisé au début dans les jeux et elles deviennent de plus en plus importantes dans le domaine de la vidéosurveillance. Parmi les avantages d'utiliser ce type de caméra dans la vidéosurveillance est qu'elles sont insensibles au changement de la luminosité car elles utilisent un capteur infrarouge et elles offrent aussi 3 flux d'informations qui aident à la construction des modèles d'objets 3D. Le dernier type est la caméra RGB. Ce type de caméra est le plus utilisé dans la littérature car elle offre un champ de vue plus que 120 degrés, une meilleure qualité d'image et elle est moins chère. Cependant, les chercheurs doivent faire un effort pour étudier le contenu des images RGB où ils sont obligés de développer des algorithmes de prétraitement robustes aux changements de l'éclairage et les problèmes de bruit d'acquisition.

Dans le tableau 1.1, nous présentons une comparaison entre les types de caméras en décrivant leurs caractéristiques, ainsi leurs avantages et inconvénients. Sachant que le type de caméra joue un rôle important dans le système où elle offre différents types de données en sortie. Les chercheurs doivent faire attention de ne pas utiliser uniquement la même caméra pour enregistrer la totalité de la base de données et pour évaluer leur système. Pour plus de détails, nous demandons de faire un coup d'œil sur le travail de [16] qui présente un résumé.

Des images capturés par ces types de caméras sont présentés dans la Figure 1.3.

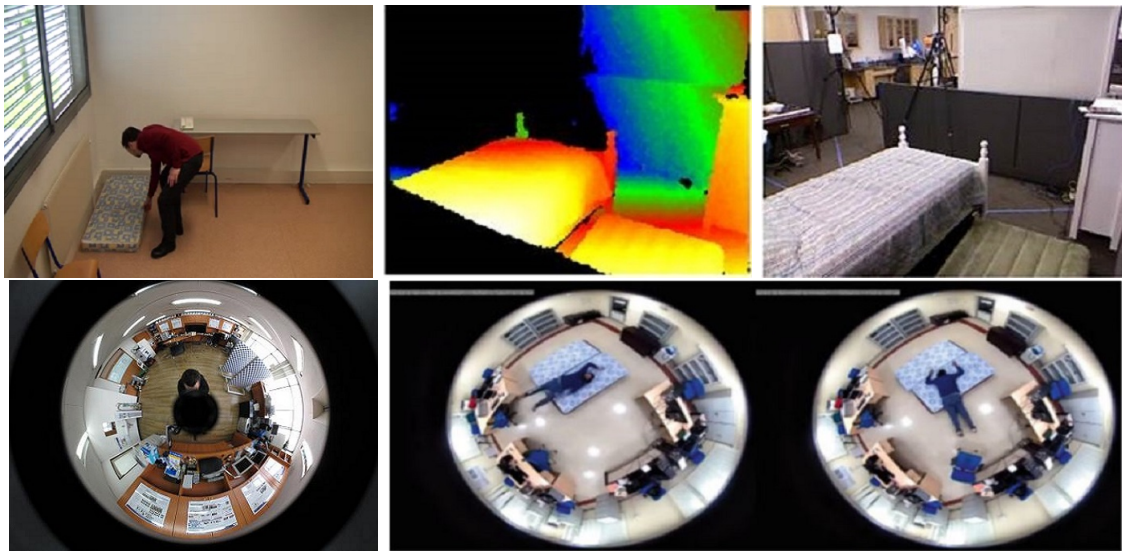


FIGURE 1.3 – Quelques images capturées par des caméras RGB, Kinect, Omnidirectionnelle et Fish-eye

1.2.2.2 Catégories des systèmes de vidéo-surveillance

L'utilisation des caméras dans un système de détection de chute offre un avantage pour analyser le comportement de la personne afin d'identifier la nature de la chute, et aussi pour informer les gens de secours avec un extrait de la vidéo de la chute. Plusieurs approches ont été proposées pour la détection de chute en utilisant la caméra où ils se divisent en deux classes. Les approches qui se basent sur une seule caméra comme [18, 3, 19, 20, 21, 22, 23, 4, 24, 25]. Les autres approches utilisent multicaméras comme dans [26, 15, 27, 28, 29, 30, 31].

1.3 Les challenges et les critères d'évaluation

1.3.1 L'environnement et la vie privée de la personne

Dans cette section, nous présentons différents types d'environnement possibles pour surveiller la personne et aussi comment faire pour respecter la vie privée de la personne. La surveillance d'une personne âgée à domicile demande plus de critère à savoir avant de commencer à développer un système de reconnaissance de la chute. Plusieurs informations nécessaires sont collectées à travers les capteurs utilisés afin de prendre une décision plus précise sans faute.

Les systèmes qui se basent sur la vision par ordinateur sont divisés en deux catégories. La première catégorie exige que la personne soit totalement visible et il est dans le champ de vision de caméra. Puis, extraire précisément les différentes caractéristiques y compris la forme et le mouvement de la personne afin de les utiliser pour la partie de décision. Pour pouvoir surveiller la personne dans toute la maison,

TABLEAU 1.1 – La comparaison entre les caméras couleur (RGB) et les caméras de profondeur (RGB-D)

Type de caméra	Caractéristiques	Remarques
Caméra RGB	Avantages : Offre l'apparence et l'information visuel, Champs de vue jusqu'à 170° , l'utilisation et l'installation facile dans tous les endroits. Inconvénients : La difficulté au niveau de l'arrière-plan et le fond lors de la présence de confusions, le changement de luminosité	Elle est utilisée dans les endroits ouverts et des endroits fermés. Cependant, ce type de caméra, il doit avoir un algorithme robuste pour la soustraction de l'arrière-plan et elle pose le problème d'intimité aux personnes
Caméra Kinect/ Caméra de profondeur (RGBD) [17]	Avantages : Offre deux informations, image RGB et image de profondeur, facile à séparer l'arrière-plan et le fond par image de profondeur, robuste au changement de la luminosité, les données peuvent être capturés en différent conditions de la lumière sans changement sur l'information capturé. Offre aussi l'information 3D. Limites : L'information capturée ne doit pas dépasser 5 mètres. Le champs de vue ne dépasse pas 90°	Elle est préférable de l'utiliser dans les environnements fermés. Préserve la vie privé de la personne.

un réseau de caméras qui sont connectées entre eux est utilisé. Le système sélectionne la caméra où la personne est visible avec un pourcentage important par rapport aux autres caméras. La deuxième catégorie utilise un groupe de caméra qui sont connectées et synchronisées entre eux dans la même chambre. Le système collecte les images à travers chaque caméra et les utilise pour les analyser afin de traiter l'objectif demandé.

Cependant, de nombreuses études portant sur les conditions de santé des personnes âgées à la maison ont été en mesure d'évaluer en profondeur les menaces environnementales des logements des personnes âgées. Par exemple, le bruit de l'environnement, les conditions de l'éclairage, vibrations, la température, humidité et d'autres critères. Souvent, les personnes âgées n'acceptent pas les caméras, car ils ont peur de leur vie privée, par contre, les capteurs portables sont acceptables pour les portés par eux. Vu l'évolution des caméras et leurs importances dans la surveillance, ils vont être plus utilisés et acceptés chez les personnes âgées.

		Les instances		Total
		Chute	Non-chute	
Système	Positive	TP	FN	$TP + FN$
	Négative	FP	TN	$FP + TN$
Total		$TP + FP$	$FN + TN$	

TABLEAU 1.2 – Tableau de Confusion

1.3.2 Les critères d'évaluation

Dans cette section, nous présentons les différents critères de performance souvent utilisés pour mesurer la capacité d'un système. Ces critères se basent sur le nombre d'instances à utiliser dans la classification. Dans notre cas, nous avons deux instances, des activités normales et l'activité de chute qui sont considérées comme deux instances pour tester l'efficacité du système. Le système prédit la classe de chaque instance : si la classe est correctement prédite, le résultat est identifié par *True*. Sinon, il est identifié par *False*.

Pour évaluer le système, nous présentons plusieurs indicateurs qui se basent sur TP, FP, TN et FN qui sont définis comme suivant :

- True Positives (TP) : Le nombre des instances positives qui sont classifiées comme positives.
- True Negatives (TN) : Le nombre des instances négatives qui sont classifiées comme négatives .
- False Positives (FP) : Le nombre des instances négatives qui sont classifiées comme positive.
- False Negatives (FN) : Le nombre des instances positives qui sont classifiées comme négatives .

Après la classification, la matrice de confusion est utilisée pour décrire la performance du classificateur sur les données de test. Le tableau 1.2 présente la matrice de confusion pour deux instances.

Le premier critère est **Rapport Erreur** qui se base sur le rapport entre le nombre d'instance identifié par *False*(Positive, Négative) sur l'ensemble des instances.

Exactitude (Accuracy) du système est le nombre d'instances identifiées par *True*(Positive, Négative) sur l'ensemble des instances.

La **Précision** du système est le nombre d'instances identifiées par *True*(Positive) sur la somme des instances positives et négatives identifiées par *True*.

La **sensibilité** ou la **recall** est le nombre d'instances identifiées par *True* (positive) sur l'ensemble des instances positives.

La **spécificité** est le nombre d'instances identifiées par *True* (Négative) sur l'ensemble des instances négatives.

$$Erreur = \frac{FP + FN}{TP + TN + FP + FN}, \quad Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (1.1)$$

$$Sensibilite = \frac{TP}{TP + FN}, \quad Specificite = \frac{TN}{FP + TN}, \quad Precision = \frac{TP}{TP + FP} \quad (1.2)$$

1.4 Structure générale d'un système de vidéo-surveillance

La plupart des approches de détection de chute dans la littérature qui se basent sur la vidéo suivent la même ligne de processus. Comme présenté dans la Figure 1.4 : La vidéo donnée en entrée est utilisée pour l'extraction et suivre la trace de la personne, puis l'extraction des caractéristiques. Ces dernières sont utilisées pour analyser et reconnaître le comportement de la personne afin d'envoyer une alerte si la chute est détectée.

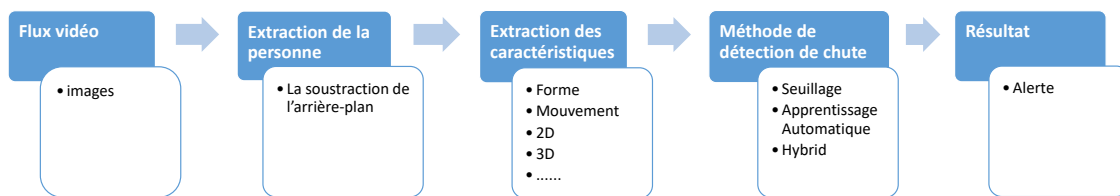


FIGURE 1.4 – La charte d'un système de détection de chute standard

Dans les récentes années, l'apprentissage approfondi a eu un succès important dans le domaine de la vidéosurveillance. Vu l'évolution de la technologie des caméras ainsi la performance des ordinateurs et l'Internet, l'utilisation de l'Apprentissage Approfondi (Deep Learning) (DL) est plus facile et n'est plus un problème aujourd'hui. L'architecture générale d'un système de vidéosurveillance à base de DL est un peu différente de l'architecture standard. La figure 1.5 illustre le processus utilisé par la plupart des chercheurs pour appliquer le DL pour la vidéosurveillance. Comme montré sur la Figure, l'extraction des caractéristiques et aussi la classification pour la reconnaissance de la chute se font d'un seul coup par le DL.

Dans la suite de cette section, nous présentons le détail des étapes de l'architecture de chaque système de vidéosurveillance appliqué pour la reconnaissance de la chute.

1.4.1 L'extraction de la personne

Comme présentée dans la Figure 1.4, la première étape dans le système de la détection de chute est la détection et l'extraction de la personne à partir de flux de

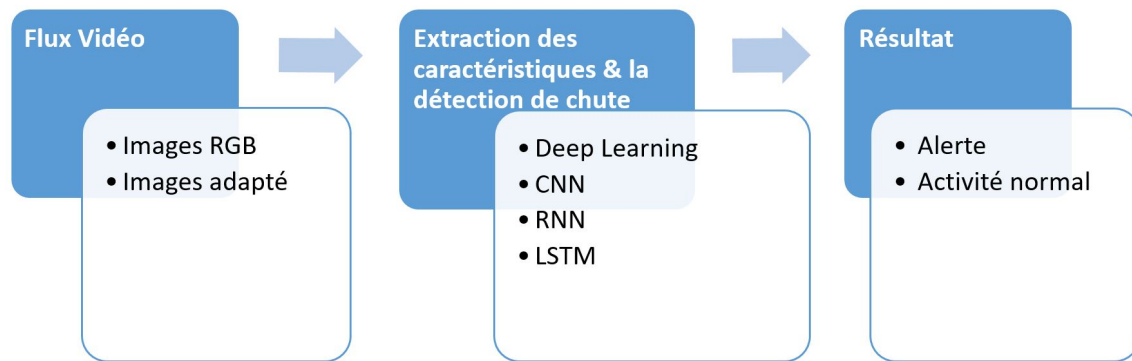


FIGURE 1.5 – La charte d'un système de détection de chute à base de Deep Learning

vidéo donnée comme entrée. La détection de la personne est généralement réalisée par la soustraction de fond Soustraction d'arrière-plan (Background Subtraction) (BS). L'algorithme de la soustraction de l'arrière BS plan est le plus utilisé pour détecter les objets mobiles par rapport à l'image de référence. Puis, une étape de prétraitement est utilisée pour déterminer la silhouette de la personne parmi les autres objets détectés. Afin de faire cette étape, dans la littérature, nous avons plusieurs techniques qui sont divisées en trois types : méthodes à base de la forme, les méthodes à base de mouvement et les méthodes à base de la texture. Pour plus d'illustration, l'objet de taille max et le nombre de pixels qui sont en mouvement dépassent un seuil sont utilisés pour sélectionner l'objet souhaité [32, 33]. Suivre la trace de la personne est généralement basé sur la texture [34].

Dans la littérature, nous avons trouvé plusieurs algorithmes de la détection des objets mobiles à base de la soustraction de l'arrière-plan qui sont divisés en deux catégories principales. Les algorithmes non-récurrents et algorithmes récurrents. Les techniques non-récurrentes stockent un nombre d'images (N) dans la mémoire. Ces images sont utilisées pour calculer l'estimation entre ces images et l'image courante. Comme exemple, nous avons la méthode de différence entre images qui est la plus connue, la différence temporelle (DT) et la méthode deCB[35]. Les techniques récurrentes mettent à jours les images stockées dans la mémoire régulièrement. L'avantage est que nous trouvons uniquement une seule image dans la mémoire. Comme exemple, nous avons l'algorithme de GMM [36], AMF [37].

Dans le Tableau 1.3, nous présentons la comparaison et la description de chaque algorithme. Nous pouvons constater que chaque algorithme contient des avantages et de limites par rapport aux bases de données utilisées. Le choix d'un algorithme dépend du contexte et des données à traiter.

TABLEAU 1.3 – Les algorithmes de la soustraction de l'arrière-plan pour les systèmes de détection de chute

Algorithme	Principe	Remarques	Approches basées
Différence entre image	Cet algorithme se base sur une image de référence statique ou des images consécutives. Les pixels en mouvements sont détectés si la différence entre l'image courante et l'image de référence est supérieure au seuil défini. Avantages : il détecte tous les pixels en mouvements selon le seuil choisi. Négatives : il est adressé aux travaux qui se basent sur une référence statique. Il ne permet pas de détecter l'ombre.	il est préférable de l'utiliser pour les bases de données de disparité. Il permet d'extraire l'objet souhaité	[38, 39, 14, 24, 40, 25, 41, 42, 34, 43, 44]
Modèle de mélange gaussien (Gaussian Mixture Model) (GMM)/ Modèle gaussien unique (Single Gaussian model) (SGM) [36]	Cette méthode permet d'extraire les objets en mouvement en se basant sur la probabilité pour comparer les pixels de l'image courante et les pixels de modèle. Il permet de détecter l'ombre.	il est adapté pour les approches qui se basent sur le mouvement parce qu'il ne permet pas de détecter des objets qui sont immobiles pendant une durée précise dont il les considère comme des objets de référence. Il est appliqué sur tous les types de bases données des images ou des vidéos	[45, 46, 47, 48, 49, 50, 24, 51, 52, 53, 54, 55, 56, 57, 58, 23, 21, 59, 60]
CodeBook (CB) [35, 61]	Cet algorithme se base sur un dictionnaire qui contient les pixels de références afin de faire la soustraction de l'image courante avec les pixels de dictionnaire. Cette soustraction se fait par la comparaison en utilisant des règles dont 6 caractéristiques sont utilisées. Avantages : Cet algorithme est capable de détecter l'ombre, applicable dans des environnements complexes, résistance au bruit de l'acquisition, appliquer aux systèmes de temps réel.	Il n'est appliqué que sur les bases de données où les images de références ne contiennent pas des objets en mouvement.	[62, 63, 33, 64, 65, 66, 67, 31, 68, 69, 70]
Différence temporelle ([71])	Cet algorithme se base sur la mise à jour du modèle de base par des pixels qui sont statique pendant une durée en utilisant la méthode de Bayésienne. L'algorithme est composé de 4 parties : détection de changement, classification des changements, la segmentation de l'objet détecter et la mise à jour de modèle et la maintenance. Avantages : il est appliqué pour les objets en mouvement ou statique. Négatives : il souffre de plusieurs problèmes où il détecte quelques parties de l'objet si l'objet reste immobile pendant une longue durée.	Appliqué sur tous les types de base de données.	[72, 73, 4, 52, 74]
Filtre médian approximatif (AMF) ([37])	Avantages : consommation faible de la mémoire, robuste et rapide en calcul. Négatives : plutôt lent en réponse à de grands changements d'éclairage, et si l'objet reste immobile plus longtemps, cet objet disparaîtra dans le modèle d'arrière-plan, cela pourrait alors influencer l'extraction des caractéristiques utilisées pour détecter une chute	Souffre de graves problèmes d'ouverture, car seules certaines parties des objets en mouvement sont détectées. Il est utilisé pour analyser des vidéos basées uniquement sur le mouvement, car cette méthode ne permet pas de détecter les objets immobiles. Il peut être utilisé dans n'importe quelle base de données	[75, 3, 76, 77, 19, 32, 78, 79]

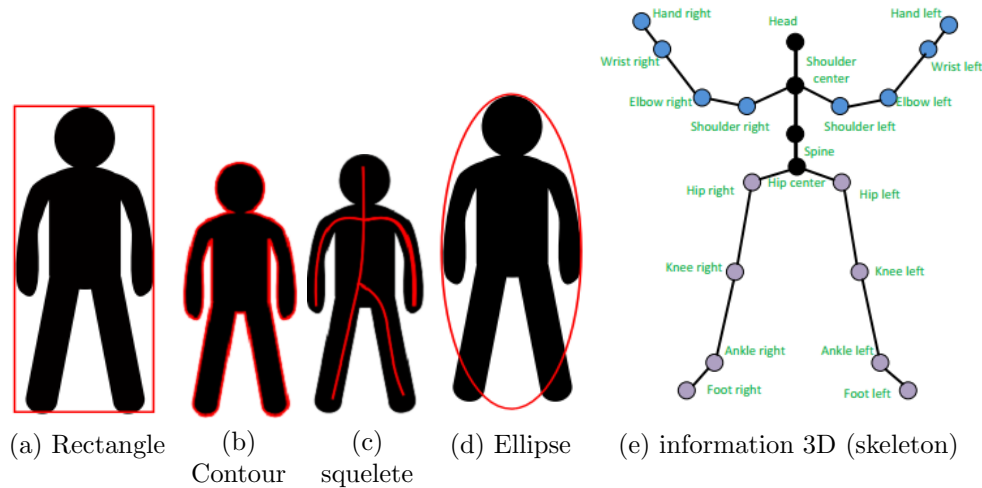


FIGURE 1.6 – Différentes formes qui englobent la silhouette de la personne

1.4.2 Extraction des caractéristiques

Après l'extraction de la silhouette de la personne, l'étape d'extraction des caractéristiques nécessaires pour analyser le comportement de la personne est appliquée. Généralement, les caractéristiques sont divisées en deux types, les caractéristiques 2D et 3D. Les informations 2D sont extraites à travers la silhouette 2D de la personne et le flux optique. La Figure 1.6 montre les différents forme qu'on peut utiliser pour extraire les informations 2D. Les informations 3D sont fournis par la caméra de la profondeur (Kinect) ou en utilisant plusieurs caméras calibrées. Ces informations sont illustrées dans la figure 1.6.

Ces caractéristiques sont utilisées pour analyser la déformation de la posture de la personne lorsqu'il pratique ces activités quotidiennes. En plus, afin de distinguer entre les activités similaires, les caractéristiques de mouvement sont ajoutées. Dans les deux Tableaux 1.4 et 1.5, nous présentons les différentes caractéristiques 2D et 3D nécessaires pour analyser les activités humaines et surtout pour détecter la chute.

TABLEAU 1.4 – Résumé des caractéristiques 2D de base pour détecter la chute

No.	Description
F1	Rapport entre Largeur et Longueur de la rectangle
F2	La variation de F1
F3	Largeur et Longueur de la rectangle
F4	Le centre de la rectangle
F5	Orientation de l'ellipse
F6	Rapport l'abscisse a et l'abscisse b de l'ellipse
F7	La vitesse de la personne à base de son centre de gravité
F8	La vitesse de la personne à base de flux optique
F9	La direction et l'orientation du mouvement
F10	Histogramme de la contexte de la personne

F11	Position de la personne à base de son centre de gravité
F12	La vitesse de la tête
F13	La distance entre le sol et la tête
F14	La taille de la personne (Largeur, longueur)
F15	Histogramme de la silhouette de la personne, Largeur et Longueur
F16	La durée de la chute
F18	La densité de la forme
F19	Le centre de la région de tête et des pieds
F20	Le coût moyenne de l'histogramme
F21	La distance (Full Procrustes Distance)
F23	Le mouvement à base de Mouvement-Énergie de l'image (Motion-Energie of Image) (MEI)
F24	Le mouvement à base de Historique de mouvement des images (Motion History Images) (MHI)
F25	Les coordonnées des points contour de la silhouette 2D
F26	Curvature Scale Space(CSS)
F27	La surface de la silhouette
F28	La moyenne de flux optique
F29	Histogramme des gradient orientés de flux optique (Histogram of oriented gradients of optical flow) (HOGOF)
F30	La distance des gradients
F31	La vitesse des points extrêmes
F32	Integrated normalized motion energy image (INMEI)
F33	La surface de la région
F34	La distance Géodésique
F35	Rapport des triple caractéristiques normaux
F36	Orientation des courbes
F37	Image de l'orientation de la silhouette (Silhouette Orientation image)
F38	Histogram of oriented gradients Difference(HOGD)
F39	La matrice de covariance de la région
F40	Histogramme de flux optique
F41	La distance entre le centre de gravité de la personne et le sol
F43	Moment of order 0,1,2
F44	Histogramme d'information de contexte
F45	L'histogramme des points clés
F46	Integrated spatio-temporal energy map (ISTE)
F47	Hu-moments of image
F48	Largeur de la tête
F49	La map de la distance
F51	Integrated Time Motion Image (ITMI)
F52	Le mouvement à base de flux optique

F53	La vitesse vertical de la tête
F54	Histogramme de la région
F55	Histogram Oriented Gradient (HOG)
F56	Grayscale Motion Map (GMM)
F57	Les images de profondeur

TABLEAU 1.5 – Résumé des caractéristiques 3D de base

No.	La description
F17	Les coordonnées 3D des points de squelette
F22	La distribution de volume (Volume 3D de la personne)
F25	Les coordonnées des points contour de la silhouette 3D
F42	Le centre 3D de la tête
F50	Les voxels (Voxels)

1.4.3 Les caractéristiques de la chute

Dans cette section, nous présentons les différentes phases pour analyser et la reconnaissance de la chute. Avant de commencer l'analyse de la chute, les chercheurs ont essayé de définir une chute et de classifier les différentes méthodes de détection de chute. L'événement de chute est découpé en 4 phases comme montre la figure 1.7.

Phase avant la chute (Pré-chute) : La personne exerce ses activités quotidiennes, avec occasionnellement des mouvements soudains tels que « s'asseoir » ou « s'accroupir ». Ces activités ne doivent pas générer d'alarme par le détecteur de chutes.

Phase critique : Cette phase, très courte, correspond à un mouvement soudain du corps en direction du sol, terminé par un impact au sol.

⇒ *Méthodes « directes » de détection de chutes* :

1. Détection précoce de la chute (vitesse du corps par exemple)
2. Détection de l'impact au sol

Phase après la chute (post-chute) : Juste après la chute, la personne reste le plus souvent immobile, incapable de bouger et allonger sur le sol.

⇒ *Méthodes « indirectes » de détection de chutes* :

1. Personne allongée
2. Absence de mouvement

Phase de rétablissement : Éventuellement, la personne peut réussir à se relever toute seule ou avec l'aide de quelqu'un.

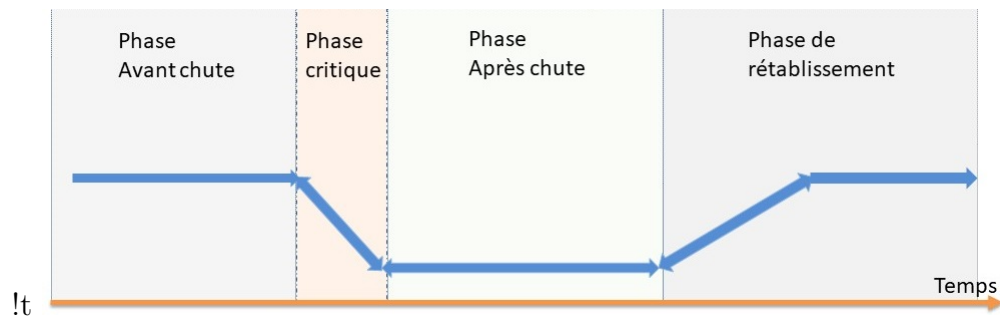


FIGURE 1.7 – Les différentes phases de la chute [1]

1.4.4 Les techniques d'apprentissage (Machine learning)

Dans cette section, nous discutons les algorithmes de l'apprentissage les plus utilisés pour faire la reconnaissance de la chute. Les algorithmes d'apprentissage se basent sur deux phases. La phase d'apprentissage et la phase de test. Durant la 1^{ère} phase, l'algorithme est entraîné par des données de chute et des données non-chute libellés/non-libellés pour reconstruire un modèle. Dans la 2^{ème} phase, le modèle est testé par des données de test qui ne sont pas identiques aux données de l'apprentissage. L'algorithme répète ces deux phases plusieurs fois jusqu'à avoir un meilleur taux de reconnaissance (voir 1.3.2).

Machine à vecteurs de support (Support Vector Machine) (SVM) Le SVM est un algorithme d'apprentissage automatique supervisé qui aide à résoudre les problèmes de classification ou de régression. Il vise à trouver une frontière optimale entre les sorties possibles.

En termes simples, le SVM effectue des transformations complexes des données en fonction de la fonction de noyau sélectionnée et, sur la base de ces transformations, il tente de maximiser les limites de séparation entre vos points de données en fonction des étiquettes ou des classes que vous avez définies.

Dans sa forme de base, la séparation linéaire, le SVM essaie de trouver une ligne qui maximise la séparation entre un ensemble de données à deux classes de points dans un espace à deux dimensions. Pour généraliser, l'objectif est de trouver un hyperplan qui maximise la séparation des points de données vers leurs classes potentielles dans un espace à N dimensions. Les points de données présentant la distance minimale à l'hyperplan (points les plus proches) sont appelés vecteurs de support.

Dans la Figure 1.8, les vecteurs de support sont les 3 points (2 bleus et 1 vert) situés sur les lignes dispersées, et l'hyperplan de séparation est la ligne rouge pleine :

Le calcul de la séparation des points de données dépend d'une fonction noyau. Il existe différentes fonctions noyau : Linéaire, polynomiale, gaussienne, Fonction de base radiale (Radial Base Function) (RBF) et sigmoïde. En termes simples, ces fonctions déterminent la régularité et l'efficacité de la séparation des classes, et jouer

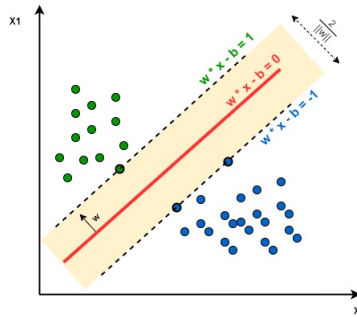


FIGURE 1.8 – Principe de SVM

avec leurs hyperparamètres peut conduire à un ajustement excessif ou insuffisant.

Dans son type le plus simple, le SVM ne supporte pas nativement la classification multi-classes. Il supporte la classification binaire et la séparation des points de données en deux classes. Pour la classification multi-classe, le même principe est utilisé après avoir décomposé le problème de multi-classification en plusieurs problèmes de classification binaire.

L'idée est de faire correspondre les points de données à un espace de grande dimension pour obtenir une séparation linéaire mutuelle entre deux classes. C'est ce qu'on appelle l'approche One-to-One, qui décompose le problème de multi-classification en plusieurs problèmes de classification binaire. Un classificateur binaire pour chaque paire de classes. Une autre approche que l'on peut utiliser est l'approche One-to-Rest. Dans cette approche, la décomposition est fixée à un classificateur binaire pour chaque classe. Un seul SVM fait de la classification binaire et peut faire la différence entre deux classes. Ainsi, selon les deux approches de classification, pour classer les points de données de l'ensemble de données de m classes :

- Dans l'approche One-to-Rest, le classificateur peut utiliser m SVMs. Chaque SVM prédit l'appartenance à l'une des m classes.
- Dans l'approche Un-à-Un, le classificateur peut utiliser $\frac{m(m-1)}{2}$ SVMs.

Prenons l'exemple d'un problème de classification de 3 classes : vert, rouge et bleu, comme dans la figure 1.9 :

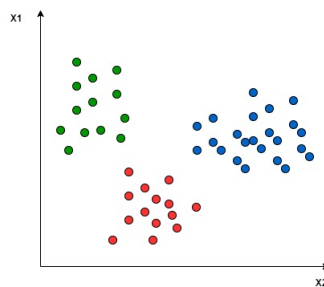


FIGURE 1.9 – Classification multi-class SVM

L'application des deux approches à cet ensemble de données donne les résultats suivants :

Dans l'approche One-to-One, nous avons besoin d'un hyperplan pour séparer entre chaque deux classes, en négligeant les points de la troisième classe. Cela signifie que la séparation ne prend en compte que les points des deux classes dans la séparation actuelle. Par exemple, la ligne rouge-bleu essaie de maximiser la séparation uniquement entre les points bleus et rouges. Elle ne tient pas compte des points verts Figure 1.10

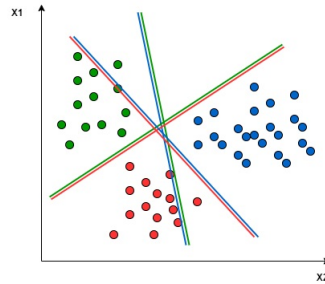


FIGURE 1.10 – Classification SVM One to One

Dans l'approche One-to-Rest, nous avons besoin d'un hyperplan pour séparer une classe de toutes les autres à la fois. Cela signifie que la séparation prend en compte tous les points, en les divisant en deux groupes : un groupe pour les points de la classe et un groupe pour tous les autres points. Par exemple, la ligne verte essaie de maximiser la séparation entre les points verts et tous les autres points à la fois Figure 1.11.

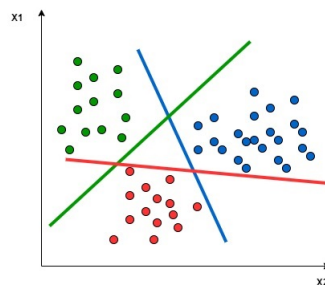


FIGURE 1.11 – Classification SVM One to Rest

K-plus proches voisins (K-Nearest Neighbour) (KNN) Cet algorithme est l'une des techniques les plus simples utilisées en apprentissage automatique. C'est une méthode préférée par de nombreux acteurs du secteur en raison de sa facilité d'utilisation et de son faible temps de calcul. Il est utilisé pour résoudre des problèmes de classification et de régression. L'algorithme KNN assume que les choses similaires existent à proximité les unes des autres. En d'autres termes, les choses similaires sont proches les unes des autres.

KNN est un modèle qui classe les points de données sur la base des points qui leur sont les plus similaires. Il utilise des données de test pour faire une "supposition entraînée" sur la classification d'un point non classé.

KNN est un algorithme qui est considéré à la fois comme non paramétrique et comme un exemple d'apprentissage paresseux. Que signifient exactement ces deux termes ?

- Non paramétrique signifie qu'elle ne fait aucune hypothèse. Le modèle est entièrement constitué à partir des données qui lui sont fournies plutôt que de supposer que sa structure est normale.
- L'apprentissage paresseux signifie que l'algorithme ne fait aucune généralisation. Cela signifie que l'utilisation de cette méthode nécessite peu d'entraînement. Pour cette raison, toutes les données d'apprentissage sont également utilisées pour les tests lors de l'utilisation de KNN.

Le fonctionnement de KNN se base sur une distance euclidienne 1.3. L'objectif est de trouver les distances entre une requête et tous les exemples dans les données, en sélectionnant le nombre spécifié d'exemples (K) les plus proches de la requête, puis en votant pour l'étiquette la plus fréquente (dans le cas de la classification) ou en faisant la moyenne des étiquettes (dans le cas de la régression).

$$d(p, q) = d(q, d) = \sqrt{(q_1 - p_1)^2 + (q_2 - p_2)^2 + \dots + (q_n - p_n)^2} = \sqrt{\sum_{i=1}^n (q_i - p_i)^2} \quad (1.3)$$

Réseau de neurones (Neural Network) (NN) Un algorithme d'apprentissage de réseau neuronal artificiel, ou réseau neuronal, ou simplement réseau neuronal, est un système d'apprentissage informatique qui utilise un réseau de fonctions pour comprendre et traduire une entrée de données d'une certaine forme en une sortie souhaitée, généralement sous une autre forme. Le concept de réseau neuronal artificiel s'inspire de la biologie humaine et de la façon dont les neurones du cerveau humain fonctionnent ensemble pour comprendre les entrées des sens humains.

Les réseaux neuronaux ne sont qu'un des nombreux outils et approches utilisés dans les algorithmes d'apprentissage automatique. Le réseau neuronal lui-même peut être utilisé comme une pièce dans de nombreux algorithmes d'apprentissage automatique différent pour traiter des entrées de données complexes dans un espace que les ordinateurs peuvent comprendre. Les réseaux neuronaux sont aujourd'hui appliqués à de nombreux problèmes de la vie réelle, notamment la reconnaissance de la parole et des images, le filtrage du courrier indésirable, la finance et le diagnostic médical, pour n'en citer que quelques-uns.

Les algorithmes d'apprentissage automatique qui utilisent des réseaux neuronaux n'ont généralement pas besoin d'être programmés avec des règles spécifiques qui définissent ce qu'il faut attendre de l'entrée. L'algorithme d'apprentissage du réseau neuronal apprend plutôt en traitant de nombreux exemples étiquetés (c'est-à-dire des données avec des "réponses") qui sont fournis pendant la formation et en utilisant cette clé de réponse pour apprendre quelles caractéristiques de l'entrée sont nécessaires pour construire la sortie correcte. Une fois qu'un nombre suffisant d'exemples ont été traités, le réseau de neurones peut commencer à traiter des entrées nouvelles

et inédites et obtenir des résultats précis. Plus le programme voit d'exemples et de variétés d'entrées, plus les résultats deviennent précis, car le programme apprend avec l'expérience.

Ce concept peut être mieux compris à l'aide d'un exemple. Imaginez le problème "simple" consistant à essayer de déterminer si une image contient ou non un chat. Si ce problème est plutôt facile à résoudre pour un humain, il est beaucoup plus difficile d'entraîner un ordinateur à identifier un chat dans une image en utilisant des méthodes classiques. Compte tenu des diverses possibilités d'apparence d'un chat sur une image, il est presque impossible d'écrire un code qui tienne compte de tous les scénarios. Mais en utilisant l'apprentissage automatique, et plus précisément les réseaux neuronaux, le programme peut utiliser une approche généralisée pour comprendre le contenu d'une image. En utilisant plusieurs couches de fonctions pour décomposer l'image en points de données et en informations exploitables par un ordinateur, le réseau neuronal peut commencer à identifier les tendances qui existent dans les très nombreux exemples qu'il traite et classer les images en fonction de leurs similitudes.

Après avoir traité de nombreux exemples d'entraînement d'images de chats, l'algorithme dispose d'un modèle des éléments d'une image, et de leurs relations respectives, qui sont importantes à prendre en compte pour décider si un chat est présent ou non dans l'image. Lorsqu'il évalue une nouvelle image, le réseau neuronal compare les points de données de la nouvelle image à son modèle, qui est basé sur toutes les évaluations précédentes. Il utilise ensuite des statistiques simples pour décider si l'image contient un chat ou non, en fonction du degré de correspondance avec le modèle.

Dans cet exemple, ce sont les couches de fonctions entre l'entrée et la sortie qui constituent le réseau de neurones. En pratique, le réseau de neurones est légèrement plus compliqué que ne le montre l'image ci-dessus. L'image 1.12 illustre un peu mieux l'interaction entre les couches, mais n'oubliez pas qu'il existe de nombreuses variations des relations entre les nœuds, ou neurones artificiels :

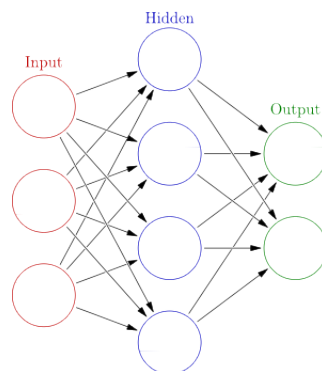


FIGURE 1.12 – Les couches de réseau de neurones

Les réseaux de neurones peuvent être appliqués à un large éventail de problèmes et peuvent évaluer de nombreux types d'entrée différents, notamment des images, des vidéos, des fichiers, des bases de données, etc. Ils ne nécessitent pas non plus de

programmation explicite pour interpréter le contenu de ces entrées.

En raison de l'approche généralisée de la résolution de problèmes qu'offrent les réseaux neuronaux, il n'y a pratiquement aucune limite aux domaines d'application de cette technique. Parmi les applications courantes des réseaux neuronaux aujourd'hui, citons la reconnaissance d'images et de formes, la prédiction de la trajectoire des véhicules de conduite autonome, la reconnaissance faciale, l'exploration de données, le filtrage des spams, le diagnostic médical et la recherche sur le cancer. Les réseaux neuronaux sont utilisés de bien d'autres façons aujourd'hui, et leur adoption augmente rapidement.

Forêts Aléatoires (Random Forest) (RF) La forêt aléatoire [80] est un algorithme d'apprentissage supervisé. La "forêt" qu'il construit est un ensemble d'arbres de décision, généralement formé avec la méthode "bagging". L'idée générale de la méthode de bagging est que la combinaison de modèles d'apprentissage augmente le résultat global. En d'autres termes, la forêt aléatoire construit plusieurs arbres de décision et les fusionne pour obtenir une prédiction plus précise et plus stable.

L'un des grands avantages de la forêt aléatoire est qu'elle peut être utilisée pour les problèmes de classification et de régression, qui constituent la majorité des systèmes actuels d'apprentissage automatique. Examinons la forêt aléatoire dans le cadre de la classification, puisque la classification est parfois considérée comme la pierre angulaire de l'apprentissage automatique. Vous pouvez voir la figure 1.13 à quoi ressemblerait une forêt aléatoire avec deux arbres.

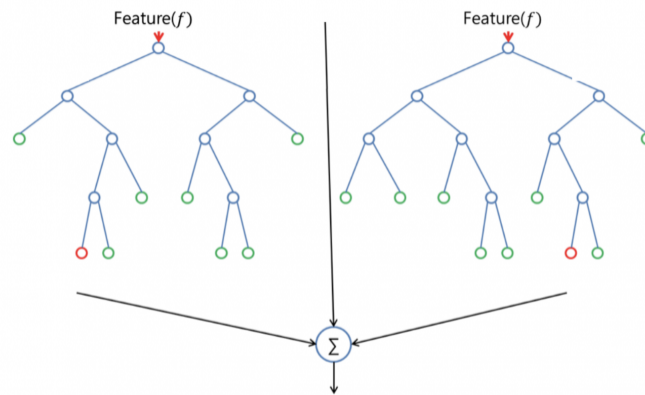


FIGURE 1.13 – Combinaison de deux arbres de Random Forest

Le plus grand nombre d'arbres dans la forêt conduit à une plus grande précision et évite le problème de l'overfitting.

Le diagramme ci-dessous dans la figure 1.14 explique le fonctionnement de l'algorithme Random Forest :

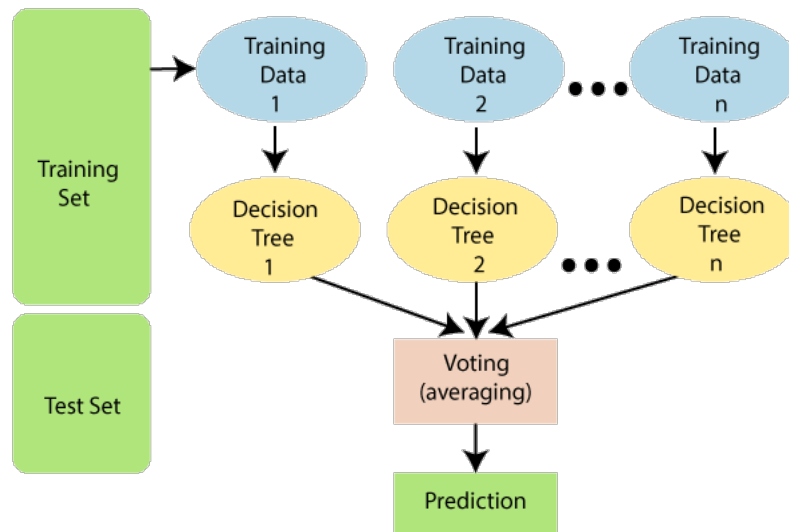


FIGURE 1.14 – Le fonctionnement de l’algorithme de Random Forest

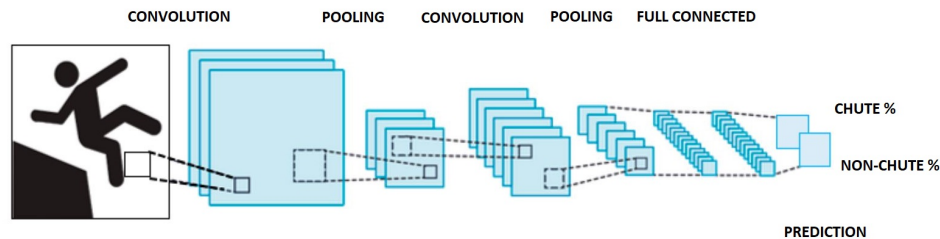


FIGURE 1.15 – Architecture de réseau de neurones convolutif (CNN)

1.4.5 Les techniques d’apprentissage profondi (Deep Learning)

Ces dernières années, un intérêt croissant a été porté aux approches, appelées modèles profonds, capables d’apprendre plusieurs couches de hiérarchies de caractéristiques et de construire automatiquement des représentations de haut niveau de l’entrée brute. Ils sont donc plus génériques puisque le processus de construction des caractéristiques est entièrement automatisé. L’un des modèles profonds les plus utilisés est l’architecture de Réseau neuronal convolutif (Convolution Neural Network) (CNN).

Convolution neural network (CNN) Dans les réseaux de neurones, le CNN est parmi les meilleurs catégories des réseaux de neurones pour l’extraction et la classification. La reconnaissance et la classification des images, la détection et la reconnaissance des objets, la reconnaissance des visages, etc., sont parmi les domaines dont CNN est le plus connu. Techniquement, durant l’apprentissage et le test du modèle CNN, chaque image en entrée doit passer par une série des couches de convolution avec des filtres (noyaux), la couche de pooling, la couche de correction ReLU et la couche de fully-connected. La figure 1.15 représente l’architecture de CNN dont nous illustrons les différentes couches et le processus utilisé pour la classification d’une image.

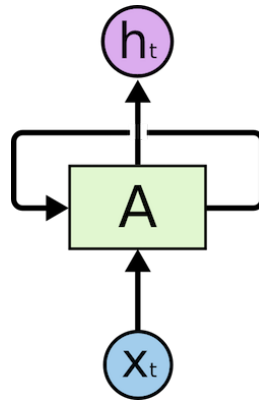


FIGURE 1.16 – Architecture de Réseaux de Neurones Récurrent Sous forme de boucle

Réseau de neurones récurrents (Recurrent Neural Networks) (RNN) est une généralisation du réseau neurones à action directe qui possède une mémoire interne. Le RNN est récurrent par nature car il exécute la même fonction pour chaque entrée de données, tandis que la sortie de l'entrée actuelle dépend du calcul précédent. Après avoir produit la sortie, celle-ci est copiée et renvoyée dans le réseau récurrent. Pour prendre une décision, il considère l'entrée actuelle et la sortie qu'il a apprise de l'entrée précédente.

Contrairement aux réseaux de neurones à action directe, les RNN peuvent utiliser leur état interne (mémoire) pour traiter des séquences d'entrées. Cela les rend applicables à des tâches telles que la reconnaissance de l'écriture manuscrite non segmentée et connectée ou la reconnaissance vocale. Dans les autres réseaux de neurones, toutes les entrées sont indépendantes les unes des autres. Mais dans un RNN, toutes les entrées sont liées les unes aux autres. Pour ce faire, le RNN va créer des réseaux comportant des boucles, ce qui lui permet de conserver l'information Figure 1.16.

Cette structure en boucle permet au réseau neuronal de prendre la séquence d'entrée. Si vous voyez la version déroulée dans la Figure 1.17, vous la comprendrez mieux. RNN est appliqué sur plusieurs domaines dont on cite quelques uns comme

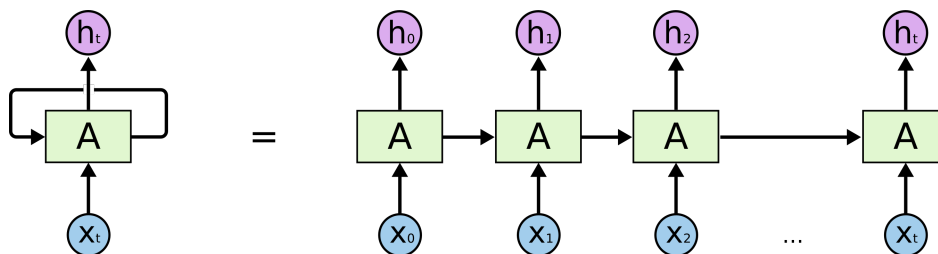


FIGURE 1.17 – Structure de RNN

la reconnaissance des actions humain [81, 82], La reconnaissance de parole [83], la détection des anomalies des séries temporelles [84].

Mémoire à long court-terme (Long Short Time memory) (LSTM) Les réseaux LSTM sont une version modifiée des RNN qui facilite la mémorisation des données passées. Le problème de gradient évanescant des RNN est résolu ici. Les réseaux LSTM sont bien adaptés à la classification, au traitement et à la prédiction de séries temporelles avec des décalages temporels de durée inconnue. Il entraîne le modèle en utilisant la rétro-propagation. Dans un réseau LSTM, trois portes sont présentes :

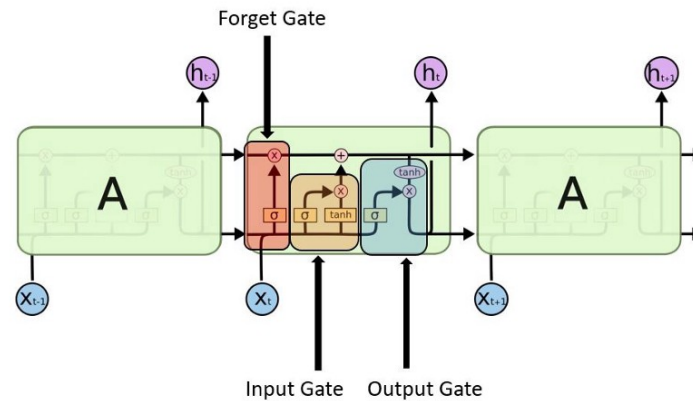


FIGURE 1.18 – Architecture LSTM

- 1 **Porte d'entrée : "input Gate"** découvre quelle valeur de l'entrée doit être utilisée pour modifier la mémoire. La fonction sigmoïde σ décide des valeurs à laisser passer (0,1), et la fonction \tanh donne un poids aux valeurs qui sont passées en décidant de leur niveau d'importance allant de -1 à 1.

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \quad (1.4)$$

$$C_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \quad (1.5)$$

- 2 **Porte d'oubli "Forget Gate"** : découvre quels détails doivent être écartés du bloc. Il est décidé par la fonction sigmoïde. Il regarde l'état précédent (h_{t-1}) et l'entrée du contenu (x_t) et sort un nombre entre 0 (omettre ceci) et 1 (garder ceci) pour chaque nombre dans l'état de la cellule C_{t-1} .

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \quad (1.6)$$

- 3 **Porte de sortie "Output Gate"** : l'entrée et la mémoire du bloc sont utilisées pour décider de la sortie. La fonction Sigmoïde décide des valeurs à laisser passer 0, 1 et la fonction \tanh donne un poids aux valeurs qui sont passées en décidant de leur niveau d'importance allant de 1 à 1 et multiplié par la sortie de Sigmoïde.

$$O_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \quad (1.7)$$

$$h_t = O_t * \tanh(C_t) \quad (1.8)$$

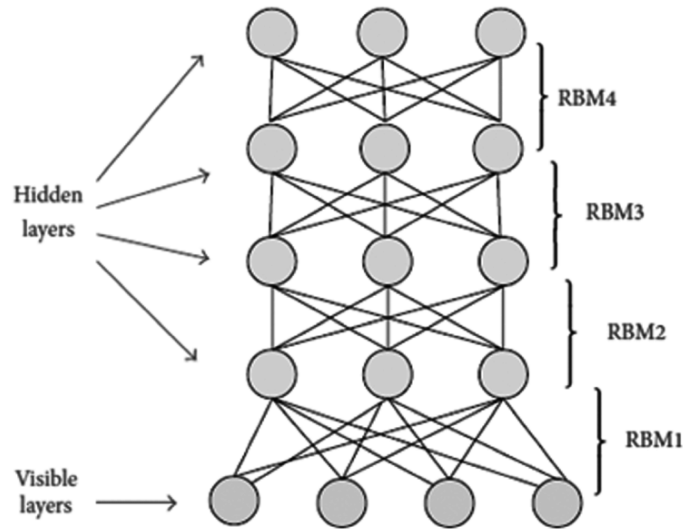


FIGURE 1.19 – Architecture de Deep Belief Network

Réseau de croyances profondes (Deep Belief Network) (DBN) est un modèle graphique génératif multicouche composé de modules empilés de machines de Boltzmann restreintes (RBM). Les DBN ont des connexions bidirectionnelles sur la couche supérieure, tandis que les couches inférieures n'ont que des connexions descendantes. Ils sont formés à l'aide d'un pré-entraînement par couche. Le pré-entraînement consiste à entraîner le réseau composant par composant, du bas vers le haut : traiter les deux premières couches comme un RBM et les entraîner, puis traiter la deuxième couche et la troisième couche comme un autre RBM et les entraîner pour ces paramètres. Dans La figure 1.19, nous illustrons l'architecture de DBN dont il est composé de 4 RBM connectés entre eux.

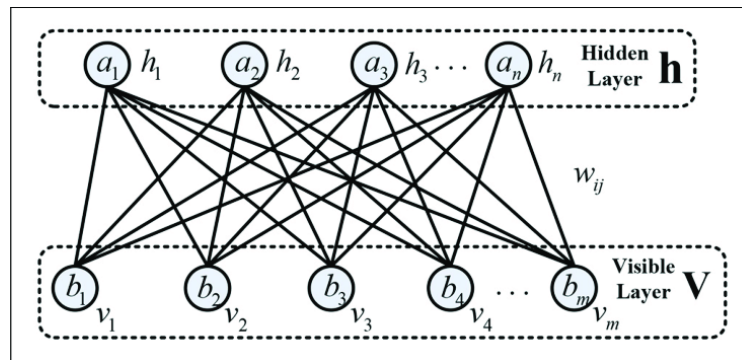


FIGURE 1.20 – Architecture de RBM

RBM est un réseau stochastique à deux couches. Ses deux couches sont la couche visible v et la couche cachée h . Comme le montre la figure 1.20, il s'agit d'un réseau RBM dont la couche visible comporte n nœuds et la couche cachée m nœuds. Il est différent du réseau de Boltzmann car il n'y a pas d'effets entre les nœuds dans une seule couche du RBM [85].

LES CATÉGORIES DES SYSTÈMES DE DÉTECTION DE LA CHUTE

Sommaire

2.1	Les approches de détection de chute dans la littérature	30
2.1.1	Approches basées sur le mouvement et la forme	32
2.1.2	Approches basées sur la posture	38
2.1.3	Approches basées sur le mouvement de la tête	40
2.1.4	Approches basées sur la variation spatio-Temporel	42
2.1.5	Les méthodes à base de l'apprentissage profondi (deep learning)	44
2.2	Comparaison quantitative des méthodes de détection de chute	46
2.3	Bases de données	50
2.3.1	Introduction	50
2.3.2	Bases de données réalisées à base de la caméra RGB	51
2.3.3	Bases de données réalisées à base de la caméra de profondeur RGBD	54
2.3.4	Synthèse	55
2.4	Synthèse et conclusion	56
2.4.1	Synthèse	56
2.4.2	Conclusion	57

2.1 Les approches de détection de chute dans la littérature

Récemment, la détection de chute est devenue un domaine actif pour les chercheurs et plusieurs systèmes qui se basent sur la vision par ordinateur sont proposés. Généralement, la plupart de ces systèmes suivent le processus présenté dans la Figure 1.4. Chaque système se base sur une caméra spécifique pour réaliser une base de données afin de développer une méthode qui dépend sur le type d'information capturé par la caméra. Au moment de l'acquisition des images, l'algorithme de l'extraction de la silhouette de la personne est appliqué, suivi par une phase de pré-traitement pour extraire correctement la silhouette de la personne. Le choix de l'algorithme de l'extraction de l'arrière-plan est important, car il aide à la décision finale. Ensuite, les meilleures caractéristiques sont extraites à travers la silhouette. Finalement, Analyser l'activité de la personne en analysant la variation des caractéristiques extraient

précédemment par une méthode à base des règles ou une méthode à base de l'apprentissage.

Dans la littérature, les méthodes de détection de chute sont divisées en deux catégories : les méthodes à base de seuillage et les méthodes à base de l'apprentissage. La première catégorie essaye de détecter la chute via un algorithme qui se base sur des règles comme dans les travaux suivants [38] [14] [49] [50] [51] [3] [86] [87] [30]. Le principe est de construire un algorithme séquentiel à base des seuils bien définie pour chaque caractéristique afin de reconnaître le comportement de la personne. L'inconvénient de ces approches est que l'algorithme doit être adapté à chaque changement de l'environnement comme le changement de la position de la caméra et le type de la caméra.

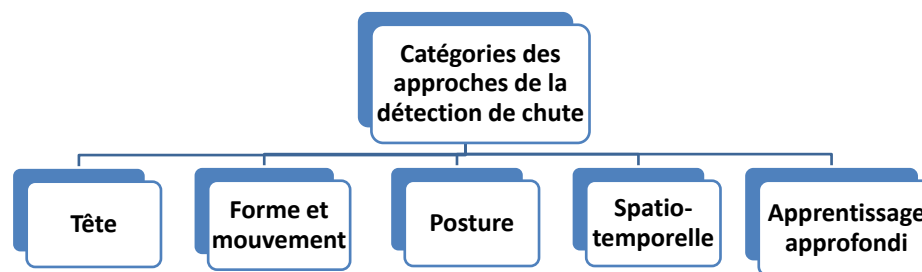


FIGURE 2.1 – Catégories des approches de détection de chute à base de la vision par ordinateur

Pour éviter toutes ces contraintes de la 1ère catégorie, les chercheurs ont adopté les méthodes à base des techniques de l'apprentissage y compris des techniques à base de GMM [67], SVM [32] [33] [40] [25] [41] [52] [66] [4] [76] [88] [89], KNN [34] [41] [90] [91], Modèle de Markov caché (Hidden Markov Model) (HMM) [56] [21] [26], naïve bayésienne (Naive Bayes) (NB) [92], Réseau de neurones artificiels (Artificial Neural Network) (ANN), Extreme Learning Machine (ELM) [23]. Cependant, récemment, les méthodes à base de l'apprentissage approfondi (Deep Learning) ont connu une forte évolution et de succès dans plusieurs domaines. Les chercheurs ont adapté et appliqué cette technique pour la détection de chute [93, 94].

Afin de faire la différence entre les approches de la détection de chute, nous avons classifié les différentes méthodes selon les caractéristiques et l'algorithme utilisé comme montre la figure 2.1. Il y a des techniques qui se basent sur **la forme et le mouvement** de la personne [32][39][41][4][3]. Les techniques à base de **Spatio-temporelle**[49][52][76], les approches à base de **la tête** [41] [67] et les méthodes à base de **la posture**[33][86][88] [64] [95].

Nous allons maintenant discuter les différentes techniques mentionnées précédemment dans les sections suivantes. Pour chaque technique, nous allons expliquer les approches dont les caractéristiques et le nombre de caméra utilisé. Ainsi, les résultats obtenus et les limitations par rapport à l'existant.

2.1.1 Approches basées sur le mouvement et la forme

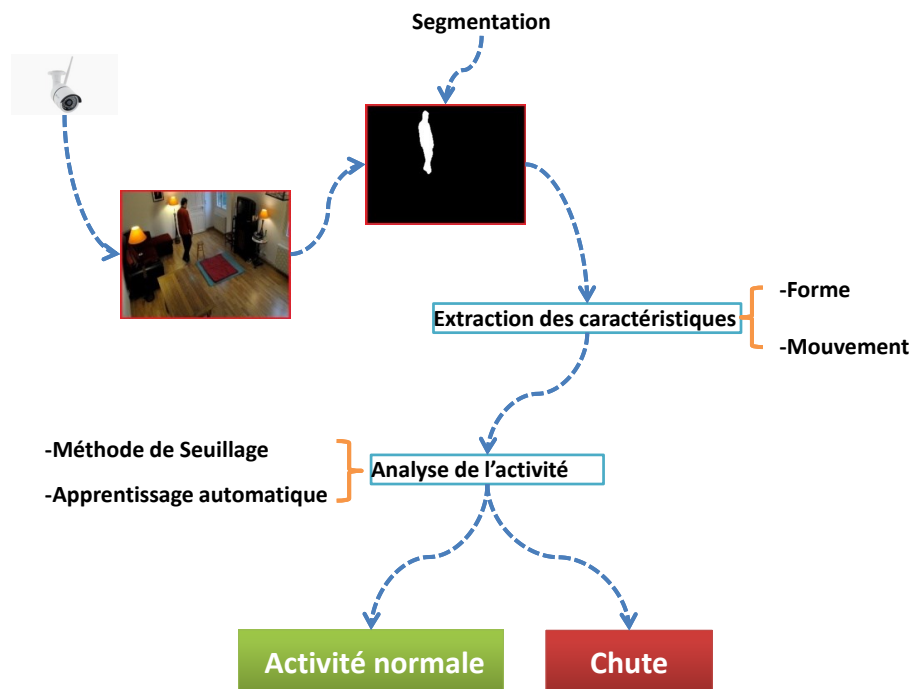


FIGURE 2.2 – Système de détection de chute à base de la forme et de mouvement

L'analyse de la variation de la forme et le mouvement au cours du temps permet de détecter la chute. Plusieurs travaux ont été proposés d'extraire des caractéristiques à base de la forme et le mouvement qui peuvent décrire correctement la chute et surtout la différence entre les activités normales et anormales. Dans le travail [51], les auteurs ont proposé un système de détection de chute en combinant les caractéristiques y compris le rapport du rectangle, gradients horizontal et vertical de la personne et l'angle de chute. Le système est composé de deux états. D'abord, le système détecte si l'événement est une chute ou un événement normal. Ensuite, il vérifie que pendant 7 images après la détection de l'événement que l'angle calculé est inférieur à 45 degrés ou les caractéristiques ont satisfait les conditions du modèle de détection de chute. Les auteurs ont montré que leur méthode est robuste sur plusieurs critères y compris le changement de l'environnement (intérieur, extérieur), le nombre de personnes (single, multiple) et le type de caméra (omnidirectionnelle, normal). Malgré ces avantages, la méthode reste limitée parce qu'ils ont fait des tests uniquement avec leurs bases de données privées.

L'approche dans [96] propose un système de détection de chute en se basant sur la méthode de seuillage. D'abord, l'extraction des joints de la personne à partir d'une caméra Kinect est déployée. La distance et l'angle de la posture de la personne par rapport au sol sont calculés à base des joints de la personne. La chute est détectée si la distance et l'angle sont inférieur et supérieur à un seuil défini respectivement. Pour montrer la performance de l'approche. Les auteurs ont utilisé leur base de données pour leur expérimentation. Leur base de données se compose de 37 vidéos de chute,

26 vidéos de marcher et 31 vidéos de s'asseoir. Le système arrive à distinguer entre la chute et les deux autres activités normales dont il atteint 100% en précision et 2.85% en taux d'erreur (1/37). Cependant, ce travail limité par rapport à la réalité, car il arrive à faire seulement la différence entre l'activité de chute et deux types d'activités normales qui ne ressemblent pas aux activités anormales.

La détection de chute avec une caméra installée au plafond de la chambre a été utilisée dans l'article [42]. Les auteurs proposent un algorithme qui se base sur l'analyse des données fournies par la caméra de profondeur. L'algorithme se compose de 4 étapes. D'abord, le pré-traitement et la segmentation pour la détection des objets. Après, l'identification des objets en classifiant les pixels sous forme des classes. Ensuite, identifier la silhouette de la personne passant par plusieurs relations anthropométriques. Finalement, le suivi de la personne et l'analyse des caractéristiques extraites pour détecter la chute. La chute est détectée si la personne est totalement proche au sol en utilisant la distance entre la caméra et la personne.

Les auteurs dans [87] utilisent des informations 3D pour analyser la forme de la personne en utilisant une caméra de profondeur. La position 3D et l'orientation de la personne sont deux caractéristiques utilisées pour détecter la chute. Les auteurs exploitent la caméra de profondeur pour pouvoir extraire les informations du plan afin de calculer la distance entre le centre de la personne et le sol. Quand les deux caractéristiques sont inférieures à des seuils, l'activité courante est détectée comme chute. Les expérimentations montrent que cette approche arrive à détecter des chutes même avec différentes directions de posture.

Les auteurs dans [38] présentent un système de détection de chute qui se base sur la combinaison de plusieurs caractéristiques. La caméra de profondeur a été exploitée pour extraire un nuage de points qui représente la forme de la personne. Puis, plusieurs informations ont été extraites et construire un vecteur qui va caractériser chaque activité de la personne. Une étape de sélection automatique des meilleures caractéristiques en utilisant l'Algorithme Génétique (AG) est précédée. Cette technique permet d'avoir minimisé la dimension de vecteur de caractéristiques pour respecter la notion de temps réel. Le système calcule la vitesse de vecteur de caractéristiques au cours du temps, si cette vitesse dépasse un seuil défini, une alarme de détection de chute se lance, sinon, aucune alarme est lancée si la personne pratique une activité normale comme marcher, s'asseoir, etc. Les auteurs testent la performance de leur système où ils ont abouti à 97.13% et 95.45% en sensibilité et en spécificité respectivement.

Autre que la détection de chute à la maison, Une autre approche a été proposé pour la détection de chute à l'extérieur de la maison dans [39]. Les auteurs extraient la silhouette de la personne à partir de l'arrière-plan, puis calculer la position de la personne dans la scène. Plusieurs informations ont été extraites comme le centre de gravité, la trajectoire, l'aspect ratio du rectangle et le temps passé pour analyser une activité. La détection de chute se fait par l'analyse de ses caractéristiques passant par la technique de seuillage. L'expérimentation sur leurs bases de données privées présente une meilleure précision.

Un système de détection de chute à base d'une caméra de profondeur RGB-D a été présenté dans [24]. Les auteurs proposent la combinaison entre les caractéristiques 2D et les caractéristiques 3D. L'identification de la chute se base sur la détection de l'événement anormal en utilisant la variation du rapport du rectangle et la vitesse verticale du centre de la silhouette, puis utiliser la variation de la profondeur pour la confirmation de vraie chute. Afin de montrer la performance de la méthode, les auteurs ont appliqué l'algorithme sur plusieurs vidéos qui contiennent des chutes et des activités anormales. Ils ont obtenu 97% pour la précision et 2% pour le taux d'erreur. L'avantage de ce système est qu'il utilise seulement un algorithme de seuillage pour détecter les événements anormaux.

L'utilisation de plusieurs caméras a été introduite par [30]. D'abord, l'utilisation de plusieurs caméras permet de la construction 3D de la forme de la personne. Par la suite, l'analyse de la distribution de volume sur l'axe vertical au cours du temps. Une alarme se déclenche lorsque la plus grande partie de cette distribution est normalement près du sol pendant une période prédéfinie, ce qui implique que la personne est tombée sur le sol. Le système atteint 99,7% de sensibilité et de spécificité avec 2 ou 3 caméras. De plus, il atteint mieux avec quatre caméras ou plus. L'avantage de ce système est qu'il arrive à détecter la chute même s'il y a des occlusions dans la scène. Cependant, il présente un inconvénient au niveau du coût d'exécution. Le travail présenté dans [55] propose un système de détection de chute en utilisant une seule caméra avec des angles de vue arbitraire. Le descripteur de l'apparence est basé sur la région de covariance, car l'apparence de l'objet peut être décrite de manière efficace par la covariance des caractéristiques des pixels. Le descripteur de la forme est basé sur l'Histogramme de gradient orienté (HOG), où la forme de l'objet est décrite par la distribution des gradients d'intensité par le vote de la direction de bord dominants. Le descripteur de mouvement est basé sur l'Histogramme de flot optique (HOF). L'apparence, la forme et le mouvement d'une personne cible sont représentés chacun comme un point qui se déplace sur une variété riemannienne différente. L'analyse de la variation dynamique de ses caractéristiques se base sur le calcul des statistiques de la vitesse des points de mouvement sur la variété correspondant à base de la distance géodésique. Les résultats obtenus sont 98.55% en sensibilité et 95.84% en spécificité.

Dans [57], les auteurs proposent un système de détection de chute à base de la reconnaissance de la forme en utilisant le classificateur SVM. Curvature Scale Space (CSS) et Fischer Vector (FV) sont utilisés pour caractériser la forme de la personne afin d'entraîner le SVM pour classifier entre la chute et d'autres types d'activités. Le système proposé atteint un taux de détection de 88.83% en précision avec 2% mieux que la méthode comparée. L'avantage de ce système est qu'il utilise seulement la forme de la personne pour détecter la chute. Cependant, cette méthode est moins robuste au niveau de précision par rapport à la littérature.

L'approche [50] propose un système qui détecte la chute en se basant sur la matrice des caractéristiques de la personne et le classificateur SVM. D'abord, l'extraction de la silhouette de la personne via l'extraction de l'arrière-plan. Puis, un traitement morphologie est suivi pour obtenir une meilleure forme de la silhouette. Les deux matrices des caractéristiques sont construites en se basant sur l'information extraite

du Hu-Moment invariant et la posture de la personne. Ensuite, Le classificateur SVM est utilisé pour détecter la chute. Le système proposé atteint 93.3% pour la sensibilité et 96.2% pour la spécificité. En outre, ces résultats obtenus avec cette approche sont moins par rapport aux approches comparées dans l'article. Une autre faiblesse est que le système est très lent en temps d'exécution.

Les auteurs dans[23] proposent une approche de détection de chute où ils combinent deux techniques à base de la vision par ordinateur. La caractérisation de la chute est basée sur la forme. Un classificateur est utilisé pour la classification afin de distinguer entre les chutes et les autres actions quotidiennes. Étant donné un clip vidéo de chute, extraire les caractéristiques de l'espace de l'échelle de la courbure (CSS) des silhouettes humaines à chaque cadre et représenter l'action par un dictionnaire de mots CSS (BoCSS). Ensuite, utiliser le classificateur ELM pour identifier la représentation BoCSS d'une chute par rapport aux autres actions.

Avec l'utilisation d'une caméra de profondeur Kinect à faible coût, ils ont construit un ensemble de données d'action qui se compose de 6 types d'actions (chuter, se pencher, s'asseoir, accroupir, marcher, et se coucher) à partir de 10 sujets. Les expérimentations sur l'ensemble de données montrent que la performance de l'approche proposée peut atteindre à une sensibilité allant jusqu'à 91,15%, une spécificité de 77,14% et une précision de 86,83%.

La majorité des approches proposées dans la littérature s'intéressent à détecter et confirmer l'existence de la chute. et pourtant, il existe des approches qui proposent des méthodes de prédiction de la chute avant que ce là arrive afin de réduire les risques associés à ces chutes. Dans l'article [97], les auteurs développent un système de prédiction de l'activité humaine dans une séquence vidéo à ses débuts par l'utilise de la représentation anatomique plane. Cette dernière se base sur un Descripteur géométrique du mouvement de la pose (MPGD) et elle utilise une seule caméra de profondeur(capteur Kinect de Microsoft) qui permet de capturer la position 3D des 20 joints du squelette de la personne. Deux profils, le mouvement et la pose, sont construits et concaténés pour construire le descripteur MPPGD pour chaque image de la séquence vidéo. La prédiction de la chute se base sur la construction d'un dictionnaire de codewords par la classification des MPPGDs des images dans la phase de l'apprentissage des vidéos en utilisant l'algorithme de K-means. Le centre de chaque classe trouvé représente le codeword. ce qui permet de représenter la vidéo d'une activité sous forme de vecteur des histogrammes accumulatives dont chaque bin est le nombre de MPPGD dans chaque classe. Ce vecteur est donné comme entrée pour le classifieur SVM dans la phase de test pour prédire la chute. Avec cet méthode, les auteurs sont capables de prédire la chute avec une précision de 93.8%.

Avec un autre de type caméra RGB-D, l'approche proposée dans [25] est d'analyser la forme et le mouvement de la personne afin de détecter la chute. Un ensemble de caractéristiques sont extraits comme la vitesse du centre de la personne, la distance entre le centre et la tête et les caractéristiques de HOG et de HOF. Pour une séquence d'image, la différence entre les caractéristiques de HOG pour deux images successives est calculée pour construire un vecteur de différence de HOG. La moyenne et la va-

riance sont extraits de ce vecteur et les utilisés comme deux caractéristiques pour la classification . Le flot optique est utilisé aussi pour extraire deux histogrammes de la moyenne et la variance de flot optique qui représentent deux vecteurs de caractéristiques. Tous ces caractéristiques sont fusionnées pour construire un vecteur de caractéristiques. Ce vecteur est utilisé pour entraîner le classifieur SVM. Les expérimentations ont été appliqués sur leur base de données construit par une caméra Kinect par 12 volontaires. Les vidéos utilisées sont composées seulement de deux activités, se coucher et la chute. Le résultat obtenu avec cette approche est 93.07% en precision.

Le travail dans[98] présente une nouvelle approche de détection de chute à partir des angles de vus aléatoires en se basant sur l'analyse de la forme sur une variété riemannienne unifiée pour les champs de caméra différentes. La forme de la personne est représentée dynamiquement sous forme d'un mouvement de point sur l'unité de n-sphère où cette dernière est parmi les unités simples des variétés riemanniennes. L'analyse de la variation de ces points sur la variété riemannienne se fait par le calcul de la distance géodésique qui représente une vitesse correspondant aux déplacements des points sur l'unité n-sphère. Les auteurs ont utilisé deux bases de données A et B qui contiennent juste des données RGB et des vidéos enregistrées par des caméras avec des champs de vus différents pour évaluer leur approche. En comparant avec les méthodes citées dans l'article, le système proposé ne donne pas un meilleur taux de performance don le résultat obtenu avec la base de données A : 91.30% et 91.67% en sensibilité et spécificité respectivement. Pour la deuxième base de données B, le résultat obtenu est : 96.77% et 89.74% en sensibilité et spécificité respectivement.

Le travail présenté dans[32] propose un système de détection de chute et leur propre base de données. D'abord, des caractéristiques simples ont été extrait comme Rapport d'Aspect (RA) du rectangle, la variation de RA, orientation de la chute(FA), la vitesse du centre(CS) et la vitesse de la tête Vitesse de la tête (Head Speed) (HS). Puis, ces caractéristiques sont normalisées pour avoir une moyenne nulle et l'écart type de l'unité. Ensuite, ces caractéristiques sont utilisées pour construire un vecteur de caractéristiques qui est la base pour détecter la chute. Les auteurs ont testé leur algorithme sur leur base de données où il atteint 90.9% en sensibilité et 100% en spécificité.

Pour améliorer les systèmes qui se basent uniquement sur la caméra ou les capteurs portables, des auteurs ont proposé des systèmes ambiants où ils combinent la caméra avec des capteurs portables pour détecter la chute.[66] proposent d'utiliser une caméra RGB avec un accéléromètre tri-axial. Premièrement, l'accéléromètre est utilisé pour détecter des activités anormales par la détection des mouvements anormaux. Puis, les auteurs utilisent la caméra pour vérifier si cette activité est une chute ou non. Les auteurs extraient la silhouette de la personne. La silhouette est divisée en 5 régions où le nombre de pixels dans chaque région est utilisé comme information pour classifier entre la chute et les autres activités quotidienne. Le classifieur SVM est choisi pour la décision finale. Le SVM est choisi vu il donne le meilleur taux de classification par rapport aux d'autres classifieurs existants. Les expérimentations de l'approche montrent que la performance du système atteint 96.77% en précision.

Les auteurs dans [40] ont utilisé la même technique présentée dans [25] pour la détection de chute. Ils fusionnent plusieurs caractéristiques, y compris l'histogramme de gradients orienté HOG, l'histogramme de gradients orienté de flux optique (HOGOF) et les caractéristiques extraites de la silhouette de la personne. Le classificateur SVM est appliqué pour la reconnaissance de la chute. La performance de l'algorithme atteint 97.50% pour la précision et 2.50% pour le taux d'erreur.

L'approche dans [25] présente une méthode de détection de chute qui se base sur une caméra Kinect. Ce type de caméra offre 2 flux d'informations qui vont être exploités en les combinant pour extraire la silhouette de la personne et utiliser le classificateur SVM pour distinguer entre une chute et une personne couchée. Le classificateur SVM est entraîné par la fusion de plusieurs caractéristiques, y compris la vitesse du centre de gravité, la distance entre la tête et le centre de gravité, HOG et la variance et la moyenne de flux optique. Les auteurs ont testé la performance de leur algorithme sur une base de données qui se compose de 1008 vidéos dont 505 vidéos contiennent les activités de chute et 503 vidéos contiennent la personne entrain de coucher. Le système atteint 93.07% pour la précision et 6.93% pour le taux d'erreur.

Charfi et al. [4] ont proposé un système de détection de chute en se basant sur une seule caméra. Ils commencent par la construction du vecteur des caractéristiques en combinant plusieurs caractéristiques afin d'utiliser ce vecteur comme entrée pour le classifieur SVM. Le système proposé classe les activités en deux classes, soit une chute ou soit une activité normale. Les auteurs ont testé leur approche sur leur propre base de données qui est disponible et publique. Le taux d'erreur atteint est de 0.38%. Pour montrer la robustesse de la méthode, les auteurs ont utilisé la base de données citée dans l'article [31]. En gardant les mêmes paramètres de l'algorithme, ils ont obtenu un taux d'erreur de 4% qui est inférieur à la méthode proposée dans [31].

Les auteurs dans [68] utilisent l'analyse du mouvement humain et les variations de forme de la silhouette humaine pour détecter et distinguer entre les événements de glissement et de chute. La mesure de mouvement est obtenue en analysant l'énergie de la zone active de mouvement (MA) dans la carte d'Energie Spatio-Temporelle Intégrée (ISTE).

Vu les limitations connues pour une seule caméra au niveau de l'angle de vue et aussi de l'occlusion, les chercheurs ont essayé d'utiliser plusieurs caméras comme solution. Les auteurs dans l'article [31] ont proposé un système de détection de chute en utilisant 4 caméras positionnées sur différents angles. Ils sont basés sur l'analyse du changement de déformation de la forme de la personne au cours de la séquence vidéo. L'analyse de la forme se base sur le context-shape dont Full Procrustes Distance et MeanCost sont deux caractéristiques extraites. Puis, ces deux dernières informations sont utilisées par le classificateur GMM afin de classifier entre une chute et une activité normale. La décision est prise par une méthode de vote afin de sélectionner la bonne caméra parmi les 4 caméras utilisées. Ce système de détection de chute atteint à un taux d'erreur de 3.8%.

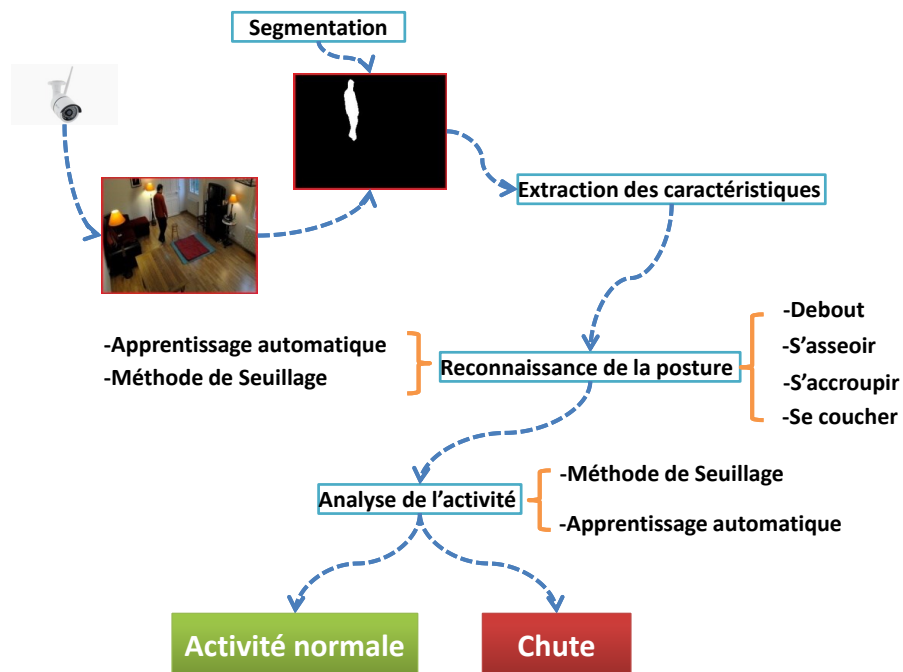


FIGURE 2.3 – Système de détection de chute à base de la posture

Une autre catégorie de la méthode basée les informations 3D à l'aide de plusieurs caméras. Les auteurs dans [26] et [30] ont utilisé des caméras calibrées qui permettent la reconstruction 3D de l'objet qui nécessite une addition de temps de calcul dans le processus d'étalonnage. Dans [30], les auteurs détectent les événements de chute en analysant la répartition du volume le long de l'axe vertical. Une alarme est déclenchée lorsque la majeure partie de cette distribution est proche du plancher pendant une période prédéfinie. L'approche [26] présente la classification entre la pose debout et la pose couchée basée sur l'histogramme de projection. L'HMM a été utilisé pour rendre la reconnaissance de pose plus robuste. Cependant, aucune modélisation explicite du mouvement en terme de séquence de pose n'est proposée.

2.1.2 Approches basées sur la posture

Dans cette section, nous allons décrire les différentes approches pour détecter la chute à base de l'analyse de la posture de la personne comme montre la figure 2.3.

Le système commence par la détection de la posture anormale de la personne. Ensuite, étape de la vérification de l'existence de la chute. Les postures anormales sont fixées par : se baisser, s'asseoir et se coucher. Ces postures sont classifiées anormales par ce qu'ils se ressemblent en terme de la forme et la surface occupée. Cette méthode est dite une méthode indirecte par ce qu'il appartient à phase de pré-traitement.

Les auteurs dans [86] proposent un modèle de bruit dépendant de la distance qui permet de détecter les personnes tombées de manière fiable. En utilisant une

caméra calibrée. L'extraction de la silhouette de la personne et le suivi la trace de la personne qui se basent sur la méthode de la soustraction de l'arrière-plan et le filtre de kalman respectivement. Un modèle de l'arrière-plan est ajouté pour ne pas détecter des objets non-requis au cours du temps. Une étape de classification est utilisée après la détection des objets en mouvement afin de distinguer entre les objets détectés. Ensuite, les auteurs utilisent un vecteur de quatre caractéristiques qui représente la géométrie de la silhouette : la surface occupée, la largeur de la silhouette, la densité et la forme de la silhouette en se basant sur le rectangle qui englobe la silhouette de la personne. La phase de la détection de chute comprise deux états : la prédiction de l'événement et la vérification de la chute. La prédiction de l'événement se base sur une méthode probabiliste afin de prédire la pose de la personne. Si cette dernière est classifiée comme coucher ou s'asseoir et la personne passe plusieurs secondes sans mouvement, alors le système lance une alarme de détection de chute. De plus, une phase de vérification est introduite pour minimiser les fausses d'alarme. Cette phase comprise trois conditions à vérifier qui sont : si la personne a pu se rétablir seul avant une minute, s'il y a une autre personne dans la scène ou si plusieurs déclencheurs d'alarme sont lancés successivement plusieurs fois, cet événement sera définitivement ignoré. Le système a été évalué sur une base de données expérimentale et sur des données réelles enregistrées pendant 6 mois. Les résultats obtenus pour les données expérimentales atteignent à 97.5% en sensibilité et 100% en spécificité. Cependant avec des données réelles, le système atteint 58% en précision.

Cependant, dans les deux approches [59] et [99], deux caméras sont utilisées pour détecter la chute. Deux caractéristiques, la taille de la personne et la surface de la silhouette, sont estimées et utilisées pour distinguer entre trois postures y compris debout, s'asseoir et se coucher. En se basant sur des seuils bien définis, la chute est confirmée par l'analyse du temps de transition de la posture. Les auteurs ont testé la performance de leur système sur la base de données [2] où ils ont choisi les simulations enregistrées par la caméra 2 et la caméra 5. La segmentation par partie du corps (BPS) a été utilisée dans [100] pour la reconnaissance des activités anormales dans un environnement bruité. La similarité entre la posture courante et les postures dans la base de données utilisées est estimée par l'utilisation de BPS. Pour tester leur algorithme, ils ont considéré deux activités normales y compris debout et s'asseoir, puis la chute pour les activités anormales. L'algorithme BPS utilise la modélisation avec HMM et GMM pour faire la classification. Les deux classifieurs sont déjà entraînés par une phase d'apprentissage avec les caractéristiques du contexte de la forme (Shape-Context).

Afin d'améliorer les résultats obtenus par les méthodes à base de seuillage, les chercheurs ont basculé vers les méthodes à base de l'apprentissage comme [92], [33] et [88]. Dans le travail [92], les auteurs ont proposé une approche intéressante à base de l'Orientation de volume de la silhouette (SOV). SOV est un descripteur de forme qui a été présenté pour représenter les actions humaines et aussi pour classifier les chutes. Il offre une précision de classification importante et il peut aboutir des meilleurs résultats si on combine avec des outils de modélisation comme Bag-of-Words (BoF) et le classifieur Bayessian Native (BN). Les auteurs ont testé leur algorithme sur la base de données présentées dans [101]. Le résultat obtenu atteint 91.89% en précision qui reste

moins en performance par rapport aux systèmes récents. Par contre, dans le travail [33], les auteurs proposent un système de détection de chute en utilisant la méthode de semi-supervisé pour détecter un événement anormal à base de la posture et la méthode de seuillage pour la décision finale qui se base sur l'analyse du mouvement. L'analyse de la posture de la personne se base sur l'information de l'angle, le rapport de l'axis-a et l'axis-b de l'ellipse, histogramme de Shape-Context et la position de la personne. L'information MEI est utilisée pour analyser le mouvement de la personne. L'étape de la vérification de la chute est lancée lorsque le classifieur SVM sort avec une posture de coucher ou de se baisser. L'alarme est lancée lorsque le mouvement de la personne dépasse un seuil prédéfini. Les auteurs ont testé leur algorithme sur leur base de données privée et ils ont obtenu un taux d'erreur de 3% qui est inférieur aux résultats présentés dans [31]. Une autre méthode est utilisée dans [88]. Les auteurs proposent un algorithme de reconnaissance de la chute par l'extraction de squelette de la personne en se basant sur une caméra de profondeur. Ils utilisent l'algorithme de classifieur X-moyennes pour la reconnaissance des poses de la personne afin que le système arrive à décrire plus précisément l'activité humaine. Le classifieur multi-classe SVM a été introduit avec l'utilisation de séquentiel minimal optimisation dans la phase de l'apprentissage. Afin de montrer la robustesse de l'approche, les auteurs ont évalué le système sur deux bases de données disponibles en littérature [102, 103] et ils ont obtenu sur la 1ère base de données 99.8% en précision et sur la 2ème base de données 97.2%.

Alban et al. [21] ont proposé un système de détection de chute à base d'HMM où ils classifient les événements en deux classes, chute et non-chute. Plusieurs caractéristiques ont été utilisées pour détecter la chute. L'orientation de l'ellipse, rapport de la hauteur et largeur du rectangle sont utilisées pour identifier la posture de la personne. La vitesse de la tête et la durée d'un changement de la posture sont utilisés pour confirmer l'existence de la chute. La performance du système proposé obtenu atteint jusqu'à 90% en sensibilité et 100% en spécificité.

2.1.3 Approches basées sur le mouvement de la tête

La tête représente un élément important dans l'analyse de l'activité humaine. L'élément le plus haut dans le corps humain. Lors de la chute, le mouvement de la tête et sa position permettent de distinguer entre les activités classiques. La complexité de cette méthode est que l'approche doit développer une méthode performant pour suivre la trace de la tête et sa position. La figure 2.4 montre le schéma global à suivre pour détecter la chute à base de la tête.

La détection de chute par le suivi la trace de la tête de la personne est une technique qui a été utilisé dans les travaux [67, 34]. Les auteurs dans [67] ont développé une méthode pour extraire les informations 3D de la tête d'une personne à suivre à base d'un filtre particulier en utilisant une seule caméra calibrée. La tête est représentée par un ellipsoïde 3D. Cependant, à base des caméras monoculaires, il n'est pas facile à réaliser le suivi de la tête à long terme [34].

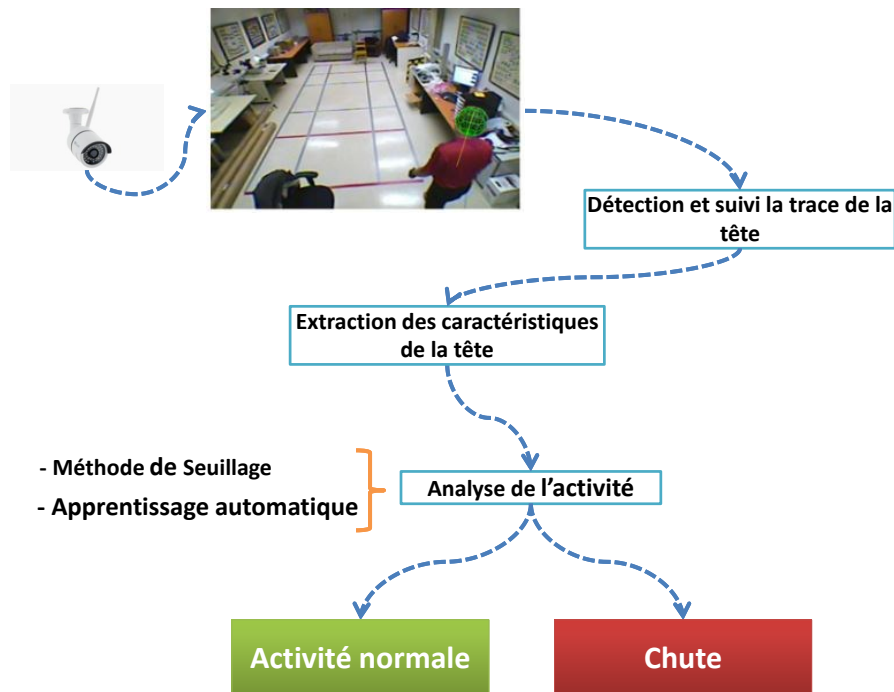


FIGURE 2.4 – Système de détection de chute à base de la tête

Le travail [58] présente une nouvelle méthode de détection de chute à base d'une caméra de profondeur. La position de la tête et les coefficients de l'équation du plan de sol sont extraits en premier. Puis, un algorithme pour suivre la position de la tête est utilisé afin de calculer la distance entre la tête et le sol. En plus, la distance entre le centre et le sol est calculée comme la deuxième caractéristique. La détection de chute est validée si et seulement si les deux informations sont inférieures à un seuil défini. Le système proposé est testé seulement sur des cas simples car l'avantage de ce système respecte la notion de temps réel où il propose un taux de 43 images par seconde qui est plus rapide par rapport aux méthodes existantes.

Kepski et al. [34] montrent que le suivi de la tête soutenue par les caméras de profondeur est beaucoup plus fiable. Les auteurs utilisent le classifieur KNN pour distinguer entre la personne allongée au sol et les activités quotidiennes en se basant sur des caractéristiques y compris la distance entre de la tête et le sol, la surface de la personne et la plus grande longueur à la largeur de forme. L'analyse du mouvement de la personne à base des informations 3D de la tête est appliquée pour déterminer si la personne est couchée à cause d'une activité normale ou d'une vraie chute. L'algorithme est évalué sur la base de données¹ qui contient plusieurs simulations des activités normales (marchant, assis, accroupi, couché) et activités anormales incluent les chutes. Ils ont obtenu 99.67% pour l'exactitude et 100% pour la précision.

Dans un autre travail, les auteurs dans [104] présentent une nouvelle approche de détection de chute à base de "body parts" de la personne. D'abord, une approche

1. <http://fenix.univ.rzeszow.pl/mkepski/ds/uf.html>

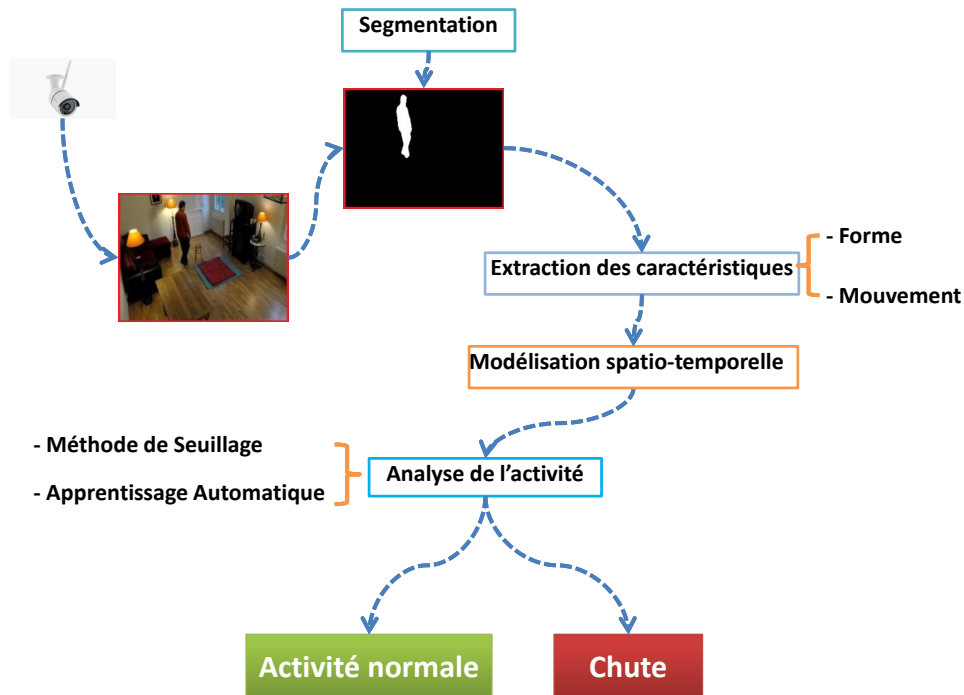


FIGURE 2.5 – Système de détection de chute à base de la variation spatio-temporelle

est proposée pour extraire les joints de la personne avec l'utilisation de l'algorithme "pose-invariant Randomized Decision Tree (RDT) algorithm". Puis, l'extraction des caractéristiques via de ces joints de la personne comme l'information 3D de la tête et la pose de la personne. Ces informations sont données comme entrées pour le classificateur SVM pour détecter une activité anormale. Puis, une étape de confirmation de la détection de chute est requise en se basant sur l'analyse de mouvement de la personne pendant un temps précis. Le système atteint une précision de la détection de chute à 97.6% et un taux d'erreur de 2.4%.

2.1.4 Approches basées sur la variation spatio-Temporel

Dans cette section, nous allons voir les différentes approches qui se basent sur l'analyse de la variation temporelle des caractéristiques. Un descripteur spatio-temporel est construit à partir des caractéristiques dites des informations spatiales et leurs évolutions temporelles. La figure 2.5 montre le processus général des approches à base de la variation spatio-temporelle.

L'approche présentée dans [76] propose la détection de chute par l'analyse de la variation des caractéristiques de la forme et le mouvement de la personne. La forme et le mouvement de la personne sont représentés et modélisés par l'Ellipse ajusté et la normalisation d'énergie (INME) dans l'image respectivement. INME est calculé à partir d'une séquence vidéo de courte durée. Ensuite la déformation de forme quantifiée des silhouettes ajustées est utilisée comme des caractéristiques pour

distinguer les différentes poses de l'être humain. Le classificateur SVM est utilisé pour classer les différentes poses en se basant sur ces caractéristiques pour détecter la chute. Les auteurs ont testé la performance de la méthode sur la base de données citées dans [3]. Les résultats obtenus sont 95.5% pour la sensibilité et 100% pour la spécificité.

Une nouvelle technique a été proposée dans [52] où les auteurs proposent une étude qui présente des nouvelles caractéristiques pour la détection de chute. Dans cette étude, ils examinent la capacité de la technique Historique des Traits Triple basée sur la Trace transform, qui permet d'avoir un système qui est robuste au bruit et invariant à des variations de différentes caractéristiques pour la représentation spatio-temporelle des occurrences de chute. Chaque séquence d'activité est représentée par un vecteur composé par l'ensemble des rapports de caractéristique triplé calculé pour chaque image de la séquence d'activité. Le vecteur de caractéristique est donné comme entrée pour le classifieur SVM afin de distinguer entre les différentes actions. Les auteurs ont testé leur travail sur deux bases de données disponibles dans la littérature [41, 4]. Les expérimentations montrent que la performance de l'approche proposée atteint à 100% en précision pour la détection de chute. Cependant, ce travail n'a pas été testé sur des cas compliqués.

On trouve aussi autres méthodes qui se basent sur la variation temporelle (spatio-Temporal) appliqué sur les points d'intérêt pour analyser les actions humaines y compris la chute. Kai Wang et al.[22] proposent un système de reconnaissance de chute des personnes âgées en se basant sur des caractéristiques extraits à partir du résultat de spatio-temporel des points d'intérêts détectés par l'utilisation de filtre linéaire séparé. Afin de discriminer efficacement entre la chute et l'activité normale, un descripteur ou un espace vectoriel caractéristique qui peut représenter le modèle local de l'activité des chutes devrait être extrait en premier lieu pour chaque patch cuboïde ou vidéo. Le classifieur SVM est utilisé pour classifier entre deux classes, les activités de chute et des activités normales. Pour valider le système proposé, les auteurs ont présenté des vidéos sous forme de patch (ou cuboïde) qui contiennent des activités normales comme marcher, s'asseoir, coucher, debout, courir, et des activités de chute comme chuter en latéral, en arrière, vers l'avant, faux pas, etc. Les résultats obtenus avec cet algorithme atteignent 93.1% en sensibilité et 94.4% en précision.

Un autre type de système qui combine les capteurs portables et les caméras a été proposée par [41]. Les auteurs utilisent une caméra de profondeur installée sur le plafond de la chambre et un accéléromètre pour distinguer entre la chute et d'autres types d'activités normales. Tout d'abord, le système utilise seulement la caméra de profondeur pour distinguer entre l'activité de se coucher et d'autres types d'activités. La séparation se fait par le classificateur KNN qui a été entraîné par des caractéristiques comme la distance entre la tête et le sol, la surface de la silhouette et le rapport entre la hauteur et largeur de la personne. De plus, pour distinguer entre couché et une chute accidentelle, l'analyse de mouvement entre les postures statiques est employée. Les expérimentations montrent que l'algorithme arrive à détecter la chute avec 100% de précision et sans fausse d'alarme. Afin de réduire le temps d'exécution, les auteurs utilisent un accéléromètre pour indiquer un

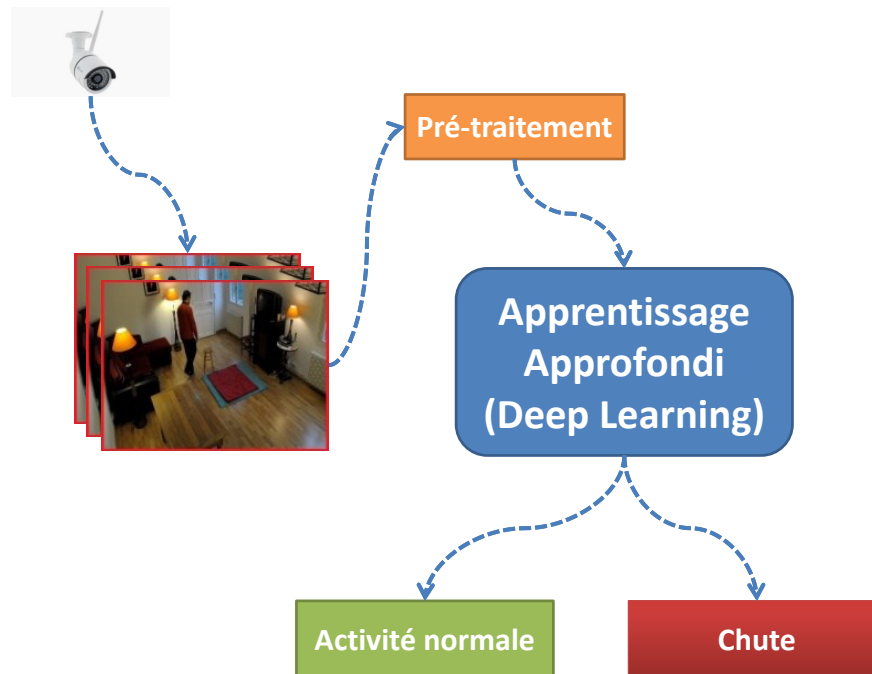


FIGURE 2.6 – Système de détection de chute à base de deep learning

mouvement inattendu pour commencer l'analyse avec des images de profondeur.

Les auteurs dans[90] présentent un système de détection de chute un peu différent par rapport à d'autres systèmes dont ils détectent la chute dans un environnement sombre. L'extraction de la silhouette est basée sur une caméra thermique au lieu d'utiliser des caméras CCD qui sont habituellement imparfaites en raison du changement abrupt de l'illumination. Tout d'abord, les actions anormales qui ressemblent aux chutes sont identifiées par l'analyse des caractéristiques de flot optique descendant. Puis, la projection horizontale de l'MHI extraite des actions anormales détectées est alors conçue pour vérifier l'existence de la chute par l'intégration de la ligne d'entité la plus proche proposée (NNFLE). Les expériences montrent que la performance de cette approche atteint 99% en précision et 1% en taux d'erreur.

2.1.5 Les méthodes à base de l'apprentissage profondi (deep learning)

Ces récentes années, l'apprentissage profondi a apporté des avantages dans plusieurs domaines. Le succès de l'apprentissage profondi a changé la vie des chercheurs pour étudier et détecter la chute. La figure 2.6 montre le processus général d'un système de la reconnaissance de chute à base de deeplearning.

L'apprentissage profondi à base de réseaux de neurones comme CNN [105] et RNN [106] sont devenus le plus utilisés pour analyser des images et des vidéos, inclut la représentation des vidéos d'activités. Dans la vision par ordinateur, la reconnaissance des activités peut être considérée aussi un problème de classification

des images.

L'approche[107] propose un système de reconnaissance de la posture en utilisant CNN pour la classification des activités en utilisant des images RGB et des images de profondeur. À partir des images de profondeur, 18 clés points sont récupérées à partir de squelette de la personne. Ces points sont divisés en trois régions pour identifier la partie de haut qui représente la tête, la partie du centre qui représente les bras et le ventre du corps de la personne et la partie bas qui représente les pieds. La position de ces trois points donne des clés significatives pour distinguer la chute avec d'autres types d'activités. Le modèle d'extraction développé de ces points arrive à 96.7% en exactitude (accuracy) par rapport au travail inspiré [3]. Le LSTM de RNN est utilisé pour la classification des activités de chute et des activités normales.

Le travail présenté dans[108] utilise le CNN et le flux optique pour la détection de chute dont ils montrent l'importance d'information du mouvement dans les systèmes de détection de la chute. Les auteurs [109] ont développé un système de détection de chute à base de CNN en 3 étapes. La première étape consiste à passer des images de flux optique aux réseaux de neurones pour éviter les informations d'apparence. Puis, le 3D-CNNest entraîné par différentes base de données pour acquérir des informations globales. Enfin, pour surmonter le problème des bases de données de petite taille, le système proposé a appliqué l'apprentissage par transfert. L'expérimentation a été réalisée avec le système développé en utilisant trois jeux de données : URFD [43], Multicam² et FDD³. Sur ces bases de données, une répartition standard (80-20)% a été effectuée. Le système a atteint une sensibilité et une précision moyennes de 94% pour les deux. Le système proposé a obtenu les meilleurs résultats avec la base de données URFD, avec une précision de 99 % et une sensibilité de 100%. Le système ne fonctionne peut-être bien que pour la détection des chutes d'une seule personne, alors que la détection des chutes de plusieurs personnes est également importante. Le travail [110] se base sur CNN pour la reconnaissance de la posture à travers des images de la silhouette de la personne obtenues via l'étape de la soustraction de l'arrière-plan. Leur système atteint le taux de classification de posture 99% pour la posture de coucher et 2% pour la posture de se baisser. Pour la reconnaissance des activités, ils proposent d'utiliser des caractéristiques comme la vitesse de la variation de la posture, Rapport hauteur/largeur et l'orientation pour distinguer entre une chute et une activité normale. Cependant, il y a des méthodes qui ne nécessitent pas l'extraction de la silhouette de la personne. Les auteurs dans [111] appliquent l'apprentissage approfondi directement à travers la séquence de vidéo. Le R-CNN est utilisé pour analyser la scène pour détecter la personne et les meubles. La reconnaissance de la chute se fait par l'analyse le changement des caractéristiques comme Rapport du rectangle de la forme de la personne et les meubles. En jugeant les relations entre les personnes et les meubles à proximité, les chutes sur les meubles peuvent être efficacement détectées. L'approche [112] transform les séquences de chute d'image RGB en images dynamiques. Ces images sont utilisées pour entraîner le réseau Deep ConvNet [113]. La stratégie d'une fenêtre de plusieurs

2. <http://www.iro.umontreal.ca/labimage/Dataset/>

3. <https://imvia.u-bourgogne.fr/basededonnees/fall-detection-dataset.html>

images a été utilisé pour déterminer le début et la fin d'une activité dans la vidéo. La base de données [4] a été utilisé pour évaluer l'approche dont leur système arrive à 98.2% pour la spécificité.

2.2 Comparaison quantitative des méthodes de détection de chute

Dans cette section, nous présentons un benchmark des performances des approches de détection de la chute. Dans le Tableau 2.1, nous présentons une comparaison des approches selon plusieurs critères comme, l'exactitude, le nombre de caméra et le type de caméra, le type de la méthode utilisée si elle est la méthode de seuillage ou la méthode de l'apprentissage. Les caractéristiques utilisées pour chaque approche. Cette comparaison nous permet d'avoir une idée sur l'évolution des systèmes durant les années, ainsi, de classer les approches aux systèmes performants en terme de taux de reconnaissance et aussi en terme de complexités.

TABLEAU 2.1 – La comparaison et la performances des systèmes de détection de chutes

Approches	Caméra	Caractéristiques	Méthodes	Performance
Chadia Khraief (2020)[114]	Mono-Camera RGB	F9, F23, F27, F33, F57	CNN	ACCU : 99.72%
Wang et al. (2020) [115]	Mono-Camera RGB	F5, F7, F11, F17	CNN	ACCU : 96.91%, SEN : 97.64% and SPE : 96.51%
Fouzi Harrou et al. (2019) [116]	Mono camera RGB	F33, F27	SVM	Accu : 96.66%, Sen : 94.00%, Spe :100%
Gracewell et al. (2019)[89]	Mono camera RGB	F5, F8, F9, F11	SVM	Accu : 97.14%, Sen : 93.75%, Spe :100%
Kaibo Fan et al. (2018) [117]	Mono-Camera RGB	F5, F6, F7, F11, F27	SVM	Accu : 97.10%
Ahmed Lotfi et al. (2018) [118]	mono-camera RGB	F3, f4, F5, F6, F7, F12, F15, F24	MLP	Acc : 99.24%, Spec : 97.38%, Sen : 99.52%
Nanez Marco et al. (2017) [108]	Mono-Camera RGB	F52	CNN, FC- NN	Accu : 97.00%
Kaibo Fan et al. (2017) [119]	Mono-Camera RGB	F5, F9, F11	SVM	Accu : 95.2%
Akagunduze et al. (2017) [92]	Mono-Camera de profondeur	F25, F36, F37	NB	Accu : 91.89%

la suite est dans la pages suivante

Tableau 2.1 – la suite du tableau précédent

Approches	Caméra	Caractéristiques	Méthodes	Performance
Glen Dardet al. (2016) [32]	Mono-Camera RGB	F1, F2, F5, F7, F12	SVM	SEN : 90.9% and SPE : 100%
Zerrouki et al. (2016)[66]	Mono-Camera RGB & Accéléromètre (Système ambiant)	F33	A base de seuillage (Thresholding Based) (TB) and SVM	Accu : 96.77%
Carolina Maldonado et al. (2016) [38]	Mono-Camera Kinect		TB	SEN : 97.13%, SPE : 95.45%, Accu : 96.10%, Er : 3.90%
Manzi et al. (2016) [88]	Mono-Camera Kinect	F17	SVM	Accu : Base de données (CAD-60) : 99.8% Base de données (TST) : 97.2%
Wang et al. (2016) [39]	Mono-Camera RGB	F1, F7, F16	TB	Pas disponible(Not Available) (NA)
Cristopher et al (2016) [86]	Mono-Camera de profondeur calibré	F1, F3, F18, F33	TB	SEN : 97.5 SPE : 100%
Kumar et al. (2016) [40]	Mono-Camera Kinect	F7, F29, F30, F31	SVM	Accu : 97.5%, Er : 2.50%
Chen et al. (2016) [90]	Mono-Camera thermique	F8, F24	TB and KNN	Accu : 99%, Er : 1%
Yun Yixiao et al. (2016) [55]	Mono-Camera Kinect	F38, F39, F40	AdabBoost	Dataset-A : SEN : 98.55%, SPE : 95.84%, Dataset-B : SEN : 100% , SPE : 97.25%
Nguyen et al. (2016) [49]	Mono-Camera RGB	F5, F6, F7, F9, F24	TB	Accu : 93.25%
Wang Kai et al. (2016)[22]	Single camera RGB	F1, F3, F7, F11	TB	Accu : 94.4%, SEN : 93.1%
Slim Abdelhedi et al. (2016) [53]	Single camera RGB	F3, F5	TB	Accu : 91.80%

la suite est dans la pages suivante

Tableau 2.1 – la suite du tableau précédent

Approches	Caméra	Caractéristiques	Méthodes	Performance
Yun Yixiao et al. (2015) [98]	Mono-Camera RGB	F33, F34	SVM	Accu : Dataset-A : 91.30%, Dataset-B : 96.77%
Goudelis et al. (2015) [52]	Mono-Camera Kinect (RGB, Profondeur)	F35	SVM	Accu : 100%
Muzafer aslan et al. (2015) [57]	Single/multiple depth camera	F26	SVM	Single Camera : Accu = 88.83%, multiple camera : Accu = 97.92%
Lei yang et al. (2015) [58]	Single depth camera	F13, F41	SVM	NA
Martha Magali et al. (2015) [96]	Single Kinetc camera	F17	TB	Accu : 100% and Er : 2.85%
Rami Alazrai et al. (2015) [97]	Single Kinect Camera	F17	SVM	Accu : 93.8%
Bian Zheng-Peng et al. (2015) [104]	single camera Depth	F42	SVM	SE :95.3%, SP :100%, Acc :97.6%, Er :2.4%
Michael Kepski et al. (2014) [41]	Single Kinect camera and accelerometer(system anmbient)	F1, F13, F14, F27	SVM, KNN, and KStar	Accu : 100%
Kepski et al. (2014) [34]	Kinect camera (Depth)	F1, F3, F41	KNN	Accu : 99.67%, Pre : 100%
Weiguo Feng et al. (2014) [76]	Single camera RGB	F1, F5, F7, F32	SVM	SEN : 95.2% , SPE : 100%
Xin Ma et al. (2014) [23]	Single Kinect camera	F25, F26	ELM	Accu : 97.20%, SEN : 99.93% , SPE : 91.97(4.25)%
Bogdan Kwolek et al. (2014) [43]	Single Kinect camera & accelerometer	F1, F3, F41	SVM	only camera : Accu = 90% & camera + accelerometer : Accu = 98%

la suite est dans la pages suivante

Tableau 2.1 – la suite du tableau précédent

Approches	Caméra	Caractéristiques	Méthodes	Performance
Benaoumeur Senouci et al. (2014)[20]	Single camera RGB	F1, F3, F4, F5, F15, F43	SVM	Accu : 99.25%
Banerjee Tanvi et al. (2014) [120]	3 type of camera	F47	GK	Accu : 94.86%
Miao Yu et al. (2013) [33]	Single camera RGB	F5, F5, F10 , F23	OC-SVM	SEN = 100% and Er = 3%
Resae Khosro et al. (2013) [54]	single camera RGB	F5, F9, F24	TB	SEN : 90% and SPE : 95%
Jia-Luen Chua et al. (2013) [3]	Single camera RGB	F1, F7, F19	TB	SEN : 90.5%, SPE : 93.3%, Er : 6.7%
Mei Jiang et al. (2013)[121]	single camera RGB	F1, F2, F14, F44	SVM and RVM	Accu : 100%
Tang et al. (2013) [14]	Single camera omnidirectional	F1, F3, F23, F24, F27	TB	Accu : 95.56%
Sen Wang et al. (2013)[122]	Two cameras RGB	F40, F45	SVM	Accu : 90.46%
Hung et al. (2013) [99]	Two cameras RGB	F2	TB	SEN : 95.8%, SPE : 100%
Zhong Zhnag et al. (2012) [15]	Single kinect camera	F16, F12, F48	TB	26 t - 2 er
Young Sook et al. (2012) [24]	Single kinect camera	F1, F4, F11	TB	SEN : 94%, SPE : 98% , Er : 2%
Ting Liao et al. (2012) [68]	Single camera RGB	F3, F4, F5, F46	BBN	slip only : Accu = 90%, fall : Accu = 95%
Rainer Planinc et al. (2012)[123]	Single Kinect camera	F17	TB	Accu : 98.6%
Alban MEFFER et al. (2012)[21]	Single camera RGB	F1, F5, F12	HMM	SEN : 90% , SPE : 100%
Chenyang Zhan et al. (2012) [91]	Single kinect camera	F14, F17	KNN, SVM	Accu = 76% 98%
Caroline Rougier et al. (2011) [74]	Single Kinect camera	F7, F41	TB	Accu : 98.7%
Edouard Auvinet et al. (2011) [30]	Multiple camera RGB	F22	TB	Accu : 97.5% and Er : 2.5%

la suite est dans la pages suivante

Tableau 2.1 – la suite du tableau précédent

Approches	Caméra	Caractéristiques	Méthodes	Performance
Rougier Caroline et al. (2011) [31]	Multiple camera RGB(4)	F6, F20, F21	GMM	Er : 3.8%
Giovani Deraco et al. (2010)[60]	Single kinect camera	F17, F41	TB	SEN : 80% and SPE : 97.3%
Chen et al. (2010) [124]	Single camera RGB	F5, F6, F49	TB	SEN : 90%
Derek Anderson et al. (2009) [125]	Two Calibrated camera RGB	F50	TB	SEN : 100% and SPE : 93.75%
Jared Willems et al. (2009) [78]	Single camera RGB	F1, F5, F15	TB	Accu : 82.5%
Foroughi et al. (2008) [69]	single camera RGB	F5, F12, F15	SVM	SEN : 90.27% and SPE : 95.16%
Homa Foroughi et al. (2008)[126]	Single camera	F51	SVM	Accu : 89.49%
Nicolas Thome et al. (2008)[56]	Multiple camera RGB	F5	HMM	Accu :82%
Huang et al. (2007)[79]	Single camera RGB	F52	NN and SVM	Accu : 85.12%
Rougier et al. (2007)[70]	Single camera RGB	F5, F6, F24	TB	SEN : 88% and SPE : 87.5%
Vinay Vishwakarma et al. (2007) [51]	Arbitrary cameras	F1, F5	TB	Accu : 95%
Cucchiara et al. (2007) [26]	Multiple camera RGB	F15	HMM	NA

2.3 Les bases de données de chute

2.3.1 Introduction

Des chercheurs dans le domaine de vision par ordinateur et la science de données nécessitent une grande base de données significative pour développer et valider leurs algorithmes.

Dans cette section, nous présentons la plupart des différentes bases de données publiques des chutes existant dans la littérature, en discutant leurs caractéristiques et leurs limitations par rapport aux systèmes de reconnaissance de la chute. Une

base de données se classifie comme une base de données proche de la réalité si elle contient un nombre important d'exemples qui représente une variabilité d'activité possible, différentes personnes, changement de la scène et l'arrière-plan, variation de la lumière, etc. Les systèmes et les algorithmes performants et robustes proposés doivent prendre en considération toutes ces contraintes.

2.3.2 Bases de données réalisées à base de la caméra RGB

Nous présentons dans cette section les différentes bases de données réalisées avec des caméras RGB. Dans la littérature, nous avons trouvé quatre bases de données publiques.

2.3.2.1 Base de données Multi caméra de Rougier (2010)

Rougier and al.⁴ [2] : cette base de données a été réalisée dans une seule salle équipée de huit caméras RGB calibrées avec différents points de vue. Les caméras sont positionnées en haut avec 2.8 m par rapport au sol. 1 seul personne (âge n'est pas marqué) a participé pour faire les simulations possibles. Elle contient 24 scénarios où les 20 premiers scénarios contiennent les chutes et le reste ne contient que les activités normales et activités similaires aux chutes. Elle se compose de 192 vidéos qui contiennent différentes simulations de chute et des activités quotidiennes normales. Les activités normales sont du type marchant, debout, s'asseoir, se coucher, se baisser. Les activités qui ressemblent aux chutes incluent la personne prend un objet, déplacer un objet, s'asseoir rapidement et marchant et se baisser pour prendre un objet.

Bien que la scène contient des objets qui pourraient cacher partiellement la personne lors d'une chute et parfois globalement pour certaines caméras. Le sol est de couleur blanche qui pose un problème aux algorithmes d'extraction de l'arrière-plan au niveau de la réflexion de la lumière. Une autre contrainte est que la personne change ces vêtements où parfois leurs couleurs sont presque identiques aux objets ou au sol de la scène. La figure 2.7 montre quelques images de la base de données.

2.3.2.2 Base de données de Chua (2013)

Base de données⁵ [127, 3] : Toutes les vidéos ont été acquises à partir d'un appareil non calibré (Dlink DCS-920) au format MJPEG de 320 * 240 pixels. Ces vidéos peuvent être divisées en deux catégories : (1) 30 activités quotidiennes normales, y compris 6 accroupi, 7 assis, 3 accroupi vers le bas et etc., (2) 21 chutes simulées, dont 8 chutes latéralement, 7 chutes en arrière, 3 chutes en avant et 3 chutes en raison de la perte d'équilibre. Le nombre de personne qui simule ces activités sont 2 personnes qui s'habillent différentes couleurs de vêtements pour mettre plus de contraintes. Dans chaque vidéo, la personne simule en répétant au moins une seule fois la même activité

4. <http://www.iro.umontreal.ca/labimage/Dataset/>

5. <http://foe.mmu.edu.my/digitalhome/FallVideo.zip>



FIGURE 2.7 – Quelques images à partir de la base de données [2]

avec différents directions. Les enregistrements commencent où la personne n'est pas dans la scène, qui est un avantage aux algorithmes de l'extraction de l'arrière-plan pour faire l'apprentissage au début. La scène généralement reste stable dans plusieurs vidéos sauf dans quelques un où quelques objets se déplacent comme la chaise, ainsi où la personne marche avec un robot. La figure 2.8 montre quelques images de la base de données.

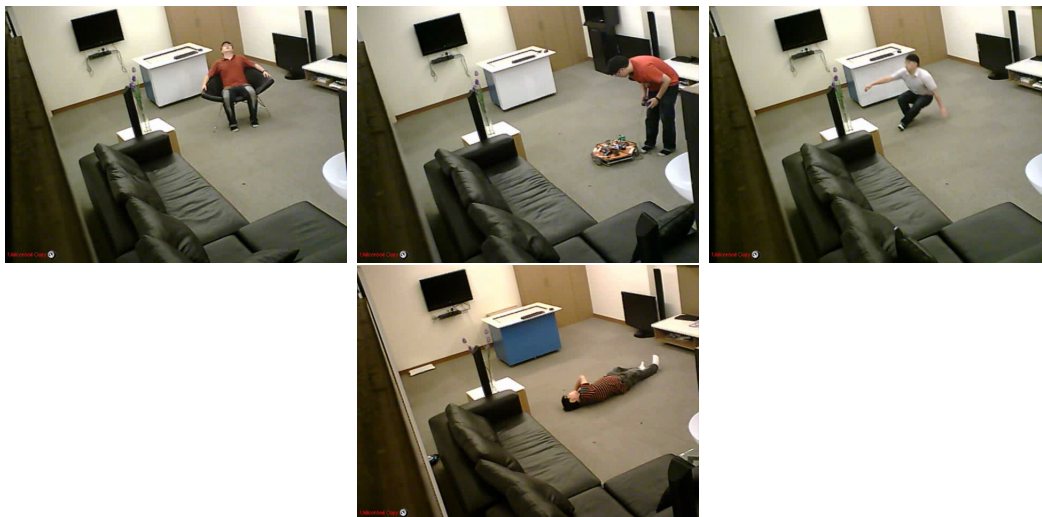


FIGURE 2.8 – Quelques images à partir de la base de données de [3]

2.3.2.3 Base de données de Charfi (2012)

Base de données⁶ [4] : Cette base de données a été enregistrée avec une seule caméra RGB positionnée 2 mètres en haut par rapport au sol. Elle se compose de 249 vidéos, dont elle est réalisée dans différents endroits ("Home", "Coffee room", "Office"

6. <http://le2i.cnrs.fr/Fall-detection-Dataset>

et "Lecture room"). Les volontaires portent des vêtements différents avec différentes couleurs et texture. Ils effectuent des diverses simulations d'activités quotidiennes normales (marchant dans des directions différentes, assis, debout, accroupi, le ménage, le déplacement d'une chaise) et les chutes (tomber en avant, tomber en voulant s'asseoir, la perte de l'équilibre). Toutes les activités sont prises dans des directions différentes, sans prendre en compte le point de vue de la caméra. Cependant, une installation dans laquelle un adulte âgé n'a qu'une seule pièce pour effectuer toutes ces activités Activités de vie quotidienne (Activity Daily Life (ADL) comme est souvent le cas dans une maison de soins infirmiers qu'il n'a pas été inclus. En outre, les enregistrements commencent toujours quand une personne est déjà dans la salle. Cela combiné avec le fait que les segments sont très courts, il est difficile d'utiliser des méthodes de soustraction de fond pour isoler la personne dans les cadres vidéo. La figure 2.9 montre quelques images de la base de données.

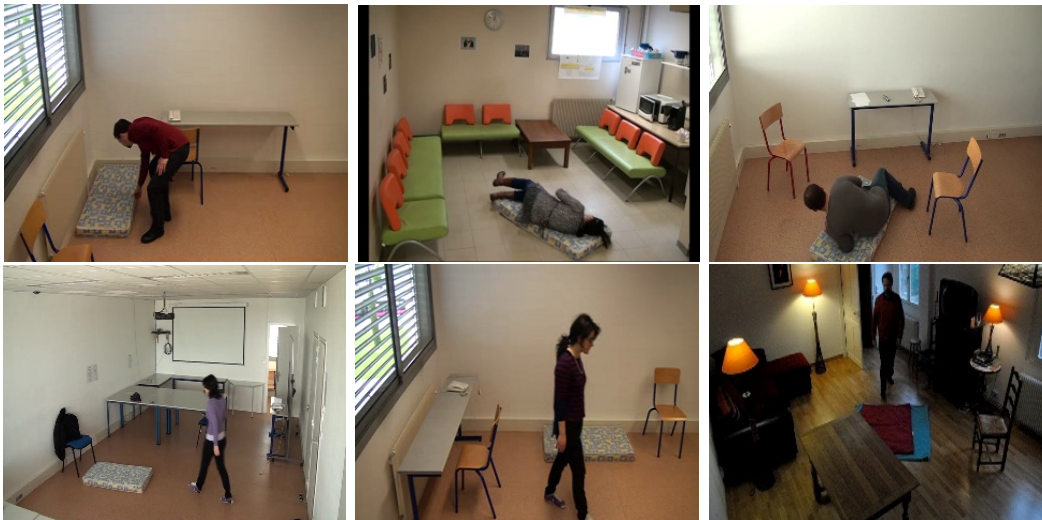


FIGURE 2.9 – Quelques images à partir de la base de données de [4]

2.3.2.4 Base de données de Baldewijns (2016)

Base de données⁷ [128] ont créé une base de données dans un environnement réel avec plusieurs scénarios. Les auteurs ont travaillé sur plusieurs critères pour créer la base de données. D'abord, la chambre était meublée pour ressembler à une chambre des soins infirmiers. La caméra qui joue un rôle très important est configurée avec une résolution de $640 * 480$ et une fréquence de 12fps. La chambre ne contient aucune fenêtre et pour cela ils ont utilisé une source de la lumière afin de changer l'éclairage de la chambre. Puis l'enregistrement des scénarios était présenté par 10 volontaires qui utilisent différents outils d'aide à la marche. Chaque type de chute a été répéter plusieurs fois et par différents volontaires. Des scénarios qui ne sont pas observés dans la vie réelle ont été simulés et ajoutés à la base de données. De plus, les auteurs ont enregistré plusieurs scénarios des activités quotidiennes où chaque scénario contient plusieurs activités normales. Ces activités normales sont enregistrées en utilisant

7. <http://iiw.kuleuven.be/onderzoek/advise/datasets>

différents outils d'aide à la marche, en changeant la place de la personne qui fait les actions et d'autres critères. L'occlusion est aussi un challenge qui est prise en considération dans l'enregistrement des vidéos. Pendant des scénarios de chute, les personnes sont partiellement ou complètement hors de la vision de caméra. Mais dans un seul scénario, plusieurs personnes sont bien dans le champs de vue de la caméra.

2.3.3 Bases de données réalisées à base de la caméra de profondeur RGBD

Nous présentons dans cette section les différentes base de données enregistrées par des caméras RGB à base de la profondeur. Dans la littérature, nous avons trouvé 4 bases de données qui sont les plus connues et utilisées par les chercheurs.

2.3.3.1 Base de données SDUFall de Haibo Wang (2017)

Base de données SDUFall⁸ : Cette base de données a été réalisée par 20 volontaires (femme et Homme) où chacune reprend 10 fois la simulation de 6 activités (chute, baisser, accroupi, s'asseoir, allongé, marchant). les activités sont enregistrées en variant plusieurs critères comme : la taille de la personne, le changement de la lumière, changement de directions et de position par rapport au caméra.

2.3.3.2 Base de données URFDS de Kwolek (2014)

Base de données URFDS⁹ : Cette base de données contient 70 vidéos incluent 30 vidéos des chutes et 40 vidéos des activités quotidiennes communes. Les chutes sont enregistrées avec deux caméras Kinect de Microsoft et les activités quotidiennes sont enregistrées par une seule caméra.

2.3.3.3 Base de données EDF de Zhong Zhang et Vassilis Athitsos (2012)

Base de données EDF¹⁰ : l'enregistrement a été réalisé par deux caméras Kinect à partir de deux points de vue. Elle se compose de différentes activités comme la chute, assis, marchant, allongé sur le sol, prendre quelques choses sous le lit et de s'habiller les chaussures.

8. url : "http ://pan.baidu.com/s/li5P83Gh" password :7ta9

9. [http ://fenix.univ.rzeszow.pl/mkepski/ds/uf.html](http://fenix.univ.rzeszow.pl/mkepski/ds/uf.html)

10. [https ://sites.google.com/site/kinectfalldetection/](https://sites.google.com/site/kinectfalldetection/)

2.3.3.4 Base de données de Planinc (2012)

Base de données [123]¹¹ : Cette base de données est constituée des chutes et des activités normales simulées par deux personnes dont chacune simule toutes les activités deux fois. La totalité des vidéos enregistrées est de 72 vidéos composées de 40 vidéos de chute et 32 vidéos des activités normales.

2.3.4 Synthèse

Dans le tableau 2.2, nous présentons les bases de données et les références des approches qui ont adopté ces bases de données pour évaluer leurs systèmes proposés. En résultat, nous pourrions observer que les bases de données les plus utilisées sont [2, 4, 43]. La raison derrière est que se sont parmi les premières bases de données qui sont réalisées et qui se sont rendues publiques. Ainsi, ils contiennent des vidéos qui ressemble à la réalité en terme de type des chutes et aussi de complexité des simulations des activités des participants.

TABLEAU 2.2 – Les bases de données et les approches qui ont adopté ces base de données pour évaluer leurs systèmes de la détection de chute

Année	Base de données	Référence d'approche
2016	Baldewijns et al. (2016)[128]	[94, 128]
2015	Gasparrini et al. (2015) [129]	[88]
2014	URFD [43] UR Fall Detection (URFD)	[34, 38, 41, 43, 52, 55, 66, 89, 98, 116, 118]
2014	SDUFall (2014)[101] (SDUFall 2014)	[23, 57, 92]
2013	Chua et al. (2015) [3]	[3, 50, 76]
2012	Charfi et al. (2012) [4]	[4, 20, 49, 52, 55, 73, 94, 114, 115, 116]
2012	EDF (2012) [130]	[15, 130]
2012	Planinc et al. (2012) [123]	[123]
2010	Rougier et al. (2010) [2]	[30, 31, 41, 53, 55, 59, 65, 67, 94, 98, 99, 114, 121]

11. <http://www.caa.tuwien.ac.at/cvl/research/cvl-databases/fall-database/>

2.4 Synthèse et conclusion

2.4.1 Synthèse

Dans ce **chapitre**, nous avons présenté un aperçu des systèmes de détection de chute basés sur les technologies de vision par ordinateur. En général, nous décrivons les principes des approches proposées pour les systèmes de détection de chute qui sont d'abord basés sur le type de caméra utilisé pour capturer les données et l'algorithme d'extraction de la silhouette humaine à partir de la vidéo. Ensuite, les meilleures caractéristiques qui peuvent décrire correctement la variation des activités humaines sont extraites. En outre, ces caractéristiques sont analysées dans le composant de détection des chutes à base d'une méthode appropriée pour classer les activités de chute et les activités normales. L'approche idéale devrait présenter une précision de 100% en taux de reconnaissance de chute et sans fausse d'alarme. Certaines approches du système de détection des chutes ont atteint ce taux de précision dans leurs expériences [41, 52, 121]. Mais lorsqu'ils conçoivent une approche générale, en changeant la base de données et qui revient à la position de caméra, leurs systèmes subissent une perte de performance spectaculaire [4].

Durant notre étude, nous avons identifié plusieurs catégories des systèmes de détection de chute. La première est la détection à base de l'analyse de forme et de mouvement. Cette catégorie se base sur le choix des bonnes caractéristiques à utilisés pour décrire correctement la variation de la forme et aussi d'analyser le mouvement de la personne. L'utilisation de la méthode d'apprentissage est le meilleur choix à appliquer pour la classification vu que le changement de la position de la caméra et aussi la différence de taille entre les personnes. La deuxième catégorie se base sur l'identification de la posture anormale de la personne avant de commencer à analyser la chute. La difficulté dans cette catégorie est d'avoir une meilleur technique qui permet d'extraire la silhouette 2D de la personne correctement. La reconnaissance de la posture demande un descripteur robuste extrait de la forme de la personne et qui est invariant au dégradation de la forme et aussi de l'angle de vu de caméra. Ensuite, une méthode robuste pour la confirmation de la chute est essentiel afin d'éviter tout fausse détection. La troisième catégorie est la méthode qui se base sur l'analyse de mouvement et la position de la tête. La détection de la tête est l'étape le plus important afin de continuer le traitement. La complexité est l'utilisation de plusieurs caméra calibré ou des caméras de profondeurs qui nécessite un temps de plus dans le traitement. La quatrième catégorie repose la variation spatio-temporelle des caractéristiques durant la séquence vidéo. L'arrivée des méthodes d'apprentissage approfondi et le succès connu ont donné l'occasion d'ajouter une nouvelle catégorie des techniques de détection de chute. Ce type des méthodes, et selon la phase où ils sont utilisés, n'ont pas besoin de la phase d'extraction de la personne ou l'extraction des caractéristiques, mais ils traitent le flux des vidéos brutes ou des vidéos adaptées, puis ils donnent le résultat en sortie directement.

Nous identifions plusieurs défis des approches pour atteindre de bonnes performances. Parmi ces défis, les simulations des jeux de données disponibles décrivent

précédemment se sont réalisées que par des jeunes. Par conséquent, tous les systèmes proposés devraient être adaptés aux personnes âgées dans la vie réelle. Ainsi, le développement d'un système réel efficace serait testé directement sur des bases de données réelles simulées par des adultes âgés. Un autre problème qui reste un inconvénient pour les chercheurs est celui de la personne partiellement occultée ou hors de vue de la caméra, en particulier pour les méthodes basées sur une seule caméra. Afin d'éviter cela, l'utilisation d'un réseau de caméras peut couvrir tous les angles de vue, ce qui permet de capturer avec précision la position et de suivre la forme entière de la personne [67, 30, 26, 131]. Cependant, cette méthode présente un temps de traitement calibré supplémentaire que les chercheurs doivent prendre en compte.

Alors que les méthodes existantes tentent de détecter une chute uniquement après que la personne soit tombée sur le sol, une autre solution peut-être utile pour la détection des chutes est la prédiction des chutes. L'événement de chute est prédit avant que la personne se tombe. Cette technique est utile et présente l'avantage de gagner en temps de traitement et de déclencher rapidement l'alarme pour sauver le patient tombé. Par ailleurs, dans [132], les auteurs tentent de prédire le risque de chute en modélisant la marche anormale des personnes âgées à l'aide du filtre de Kalman et de l'estimation du mouvement. Un facteur majeur qui différencie le modèle de marche des personnes âgées de celui des jeunes adultes est la vitesse de marche, tandis que dans [86], les auteurs prédisent la chute de la personne en estimant l'état anormal de la personne en utilisant un cadre de probabilité.

Les méthodes d'apprentissage profond ont connu un succès dans de nombreuses tâches informatiques telles que la reconnaissance des actions humaines et la reconnaissance des objets [133]. L'utilisation de cette méthode pour la détection des chutes devient parmi les principales tâches des chercheurs. Jusqu'à présent, plusieurs travaux ont été proposés [107, 109, 134, 110] qui ont introduit un apprentissage profond pour détecter les chutes en l'appliquant directement sur le flux vidéo ou l'utilisation des images adaptés. Néanmoins, l'apprentissage profond nécessite plus de données pour l'entraînement afin de construire un modèle robuste qui n'est pas le cas actuellement. Dans l'étude [134], les auteurs ont créé un nouveau jeu de données sur les chutes, collecté sur Youtube. Le grand souci avec ce type des méthodes et qu'elles ne sont pas paramétrables pour les adapter à la réalité et aux personnes âgées. elles ont besoin des bases de données de grand taille des vrais personnes âgées pour faire l'apprentissage qui n'est pas le cas aujourd'hui où il y a que des simulations des personnes jeunes.

2.4.2 Conclusion

Après avoir vu les différentes catégories des systèmes de détection de chute et identifié les différentes limites des approches dans l'état de l'art. Nous avons adopté de proposer une méthode qui se base sur la forme et le mouvement de la personne et une autre méthode qui se base sur la posture de la personne. La raison qui nous a mené à proposer une méthode dans chacun de ces deux catégories revient aux limites des choix des bonnes caractéristiques de forme ou de mouvement qui permettent

d'améliorer le taux de reconnaissance de la chute. Ainsi, le manque d'un descripteur robuste pour la reconnaissance de la posture. Une autre raison est de résoudre les problèmes d'utilisation d'une seule caméra dans ces deux domaines. En plus, le nombre de bases de données publiques pour évaluer notre proposition sont tellement disponibles et sont importantes pour la comparaison avec l'état de l'art.

Dans le chapitre suivant, nous allons voir notre première contribution à la détection de chute à base de la variation de la forme et de mouvement.

PROPOSITION D'UN SYSTÈME DE DÉTECTION DE CHUTE À BASE DE LA FORME ET DE MOUVEMENT

Sommaire

3.1	Introduction	59
3.2	Problématique et la littérature	60
3.3	L'algorithme du système proposé	61
3.4	Structure du système proposé	63
3.4.1	Détection de la personne et la segmentation du sol . . .	63
3.4.2	Extraction des caractéristiques	66
3.4.3	La structure proposée pour la reconnaissance de chute .	70
3.4.4	Méthode de segmentation des activités dans la vidéo . .	71
3.4.5	Méthode proposée pour la reconnaissance de la chute . .	72
3.5	Expérimentation et analyse de performance	73
3.5.1	Analyse des caractéristiques proposées	74
3.5.2	Évaluation quantitative des caractéristiques proposées .	76
3.5.3	Évaluation par scénario de notre approche et la classification	77
3.5.4	Discussion	81
3.6	Conclusion	82

3.1 Introduction

Nous présentons un système de reconnaissance des chutes pour les personnes âgées. Une chute est caractérisée par une déformation de forme et un mouvement élevé. Nous avons proposé de représenter la variation de la forme par trois caractéristiques, le rapport de la Rectangle (Bounding Box) (BB), l'orientation de l'ellipse et le rapport hauteur/largeur de l'histogramme de projection. Nous proposons de diviser la silhouette de la personnes en trois blocs. Ces blocs correspondent à la région de la tête, la région du centre du corps et la région des jambes. Dans chaque bloc, nous extrayons des caractéristiques en utilisant le flux optique pour la variation du mouvement. Ces caractéristiques extraites sont la vitesse de chaque bloc, la vitesse de la personne et la direction de la personne. Chaque type d'activité, chute ou non-chute, est représenté par un vecteur de caractéristiques construit à partir de la variation des caractéristiques de la forme et du mouvement au cours d'une séquence d'images. Ensuite, nous utilisons SVM pour classifier les activités de chute et

non-chute. Les expériences ont été menées sur trois bases de données différentes. Les résultats montrent l'efficacité de la méthode proposée.

3.2 Problématique et la littérature

Étant donnée que la détection de chute à base de la vision par ordinateur est la clé pour améliorer les performances des systèmes pour aider les personnes âgées à vivre dans un environnement sécurisé, il existe de nombreuses approches axées sur la construction d'un système puissant qui sert à réduire les fausses alarmes de chute et obtenir un taux de détection élevé.

Certaines des techniques classiques de détection de chute sont basées sur les règles de détection d'une chute. Dans [51], une chute est détectée si le rapport du cadre englobant est inférieur à un seuil, tandis que dans [49], les auteurs ont détecté un événement anormal lorsque le mouvement est supérieur à un seuil défini, alors ils ont confirmé la détection de chute si aucun autre mouvement ne s'est produit pendant une séquence d'images. Cependant, ces méthodes doivent être adaptées à tout changement de position de la caméra ou de l'environnement. Ainsi, les chercheurs ont proposé des méthodes basées sur l'apprentissage automatique pour réaliser un système général. Les classifieurs SVM, CNN, ELM, GMM, NB et KNN sont les méthodes d'apprentissage automatique les plus courantes. Ils ont été utilisés dans [40, 57, 92, 117, 31, 119, 108, 116].

Les auteurs dans [40] exploitent les informations RGB et de disparité fournies par une caméra de profondeur pour extraire plusieurs caractéristiques, 2D et 3D, de la silhouette humaine, y compris les caractéristiques de forme et de mouvement. Ils ont concaténé ces caractéristiques en un seul vecteur, qui est donné comme entrée au classificateur SVM pour prédire la classe de l'activité. Dans le travail [57], le codage Fisher Vector est utilisé pour décrire les actions basées sur la forme de l'échelle de courbure. Un classificateur SVM pré-entraîné est utilisé pour effectuer la classification finale. La forme humaine est également utilisée dans [116] pour la détection des chutes. Les auteurs ont défini 5 régions d'occupation. Ces régions ont été obtenues par simple partition centré sur la masse centrale du corps. Les rapports de surface de chaque trame sont calculés et utilisés comme données d'entrée pour la détection et la classification des chutes. Pour améliorer le résultat, les auteurs ont combiné le rapport de vraisemblance généralisé (GLR) et SVM. Dans [92], l'événement de chute est détecté en fonction de l'analyse de forme en utilisant les caractéristiques de volume d'orientation de la silhouette (SOV). Le SOV est construit à partir de l'image spatio-temporelle de l'orientation de la silhouette (SOI) [135]. Tout d'abord, chaque action humaine est représentée par le Sac de Mots (BoW) de SOV. Ensuite, le classificateur NB est utilisé pour classer les actions de chute des activités normales. La faiblesse de ces méthodes [57, 116, 135] est que les résultats pourraient être diminués en cas d'occlusion ou de mauvaise segmentation de la forme humaine. De plus, aucune information de mouvement n'est utilisée, ce qui pourrait aider à améliorer les résultats. Bien qu'Ismail et al. [31] ont proposé une approche qui vise à détec-

ter une chute en utilisant des descripteurs d'histogramme basés sur l'appartenance (MHD) comme une généralisation de BoW, le descripteur est obtenu en mappant les caractéristiques visuelles de bas niveau originales à un descripteur plus discriminant en utilisant des appartenances possibilistes. La caractéristique de gradient orienté histogramme (HOG) est extraite en tant que caractéristique de bas niveau. Le KNN a été utilisé pour attribuer chaque descripteur à la classe de chute ou de non-chute.

Une autre méthode a été présentée dans [31] où les auteurs ont utilisé le vote multi-vues des résultats issus d'un classificateur GMM pour détecter une chute basée uniquement sur la déformation de forme entre deux images consécutives d'une vidéo. L'utilisation uniquement des informations de forme est la principale limitation de cette méthode. Une combinaison des caractéristiques d'apparence, de forme et de mouvement est utilisée dans [55] pour détecter la chute. Chaque caractéristique est représentée comme un point mobile sur la variété riemannienne. Les statistiques de vitesse de ce point sur le collecteur sont utilisées avec un classificateur SVM pour faire la distinction entre l'activité de chute et l'activité normale. Les résultats de leurs expériences montrent une grande précision. Cependant, le temps de traitement est trop élevé. Récemment, Fan et al. [119] a extrait plusieurs caractéristiques d'une ellipse calculée à partir des silhouettes pour décrire la posture humaine. Ensuite, ils ont développé un modèle SVM pour classer la posture humaine liée à une chute dans chaque image. Par contre, dans [117], les auteurs ont considéré les incidents de chute comme une séquence de caractéristiques de la forme. Ces caractéristiques sont analysées à l'aide de l'Analyse des caractéristiques lentes (Slow Features Analysis) (SFA) [136]. Six caractéristiques de forme ont été extraites de la silhouette humaine, et ces caractéristiques ont été transformées en SFA qui, à la chute, peuvent être décrites par l'accumulation de dérivés temporels de premier ordre au carré de ces caractéristiques lentes. Le Graphe acyclique dirigé de SVM (DAG-SVM) est utilisé pour détecter les chutes.

Les méthodes de CNN ont un grand succès dans le domaine de la reconnaissance de forme. Une brève introduction peut être trouvée dans [105]. Les caractéristiques efficaces sont extraites à l'aide de CNN pour effectuer la détection et la classification des images. Les auteurs de [108] ont proposé une solution basée sur la vision par ordinateur utilisant CNN pour détecter si une séquence vidéo contient des incidents de chute. La solution CNN proposée est basée sur une image de flux optique comme entrée en essayant d'incorporer des informations de mouvement.

3.3 L'algorithme du système proposé

L'analyse de la forme humaine et la variation de mouvement sont les solutions les plus courantes pour la reconnaissance du comportement humain, comme la détection des chutes. Une chute est caractérisée par une déformation de forme et un mouvement élevé. Cependant, certaines activités normales sont similaires à une chute. Pour surmonter ce problème, nous proposons une nouvelle approche qui comprend trois phases, l'extraction de la forme humaine, l'extraction des caractéristiques avec des

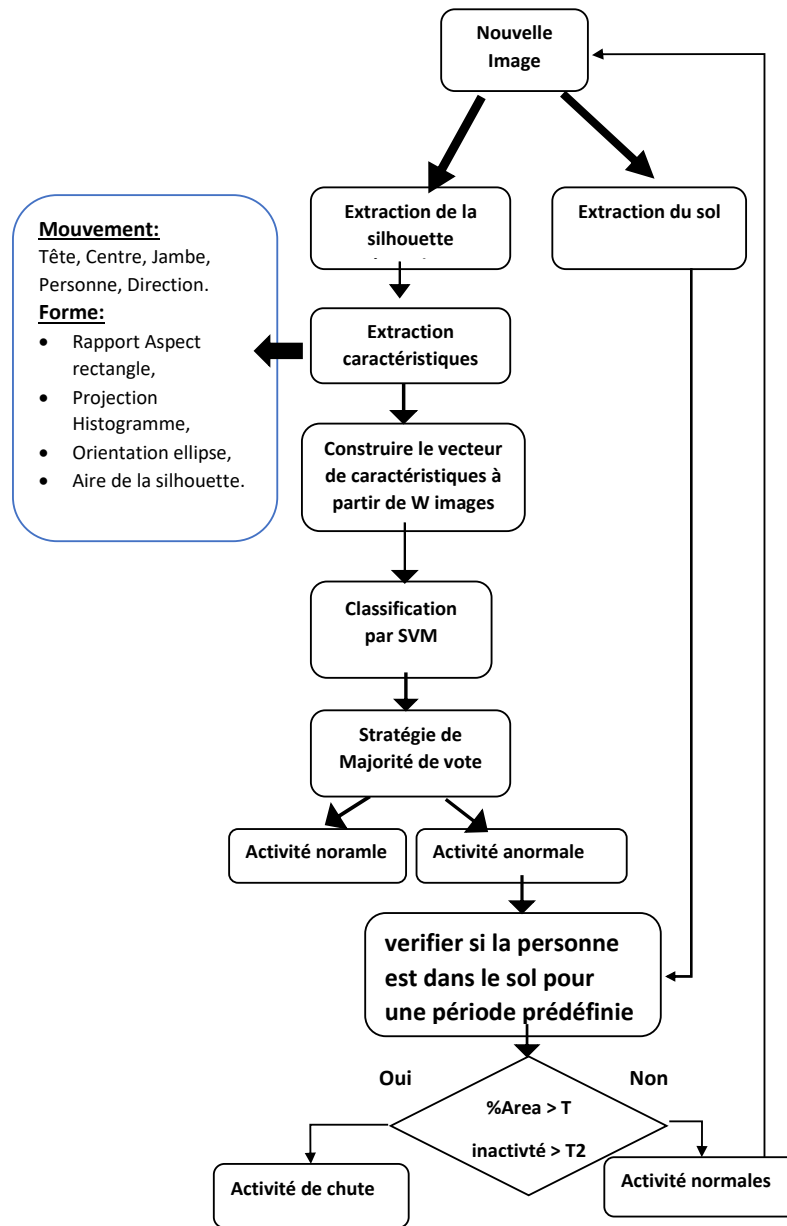


FIGURE 3.1 – Les principaux composants de notre approche

activités de segmentation et la détection des chutes par classification. Nous avons commencé par l'extraction de forme de la personne à partir de l'image vidéo en utilisant la méthode de soustraction d'arrière-plan, suivie d'un post-traitement pour extraire correctement la silhouette humaine, puis nous mettons à jour le modèle d'arrière-plan. Ensuite, la silhouette humaine est entourée par BB pour extraire plusieurs caractéristiques décrivant avec précision la variation de la posture humaine. Initialement, nous avons représenté la forme de la personne avec trois blocs basés sur BB ajusté sur la silhouette humaine comme montre la figure 3.2. Ensuite, nous avons calculé la vitesse à l'intérieur de chaque bloc et la vitesse de la personne afin d'analyser la variation de mouvement. De plus, nous extrayons les caractéristiques de

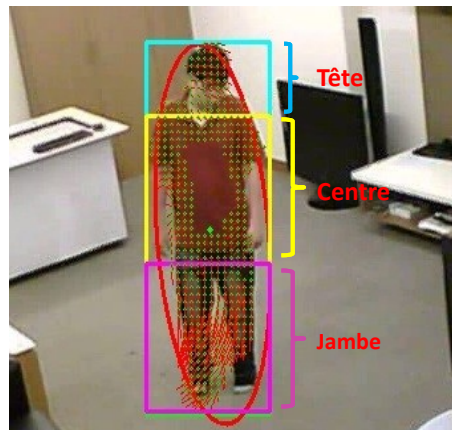


FIGURE 3.2 – Illustration de la représentation des trois blocs de la silhouette humaine

BB, d'ellipse et de projection de l'histogramme pour analyser la variation de forme. Aussi, nous avons extrait les caractéristiques de mouvement et de forme à travers une suite d'images. Ces caractéristiques ont été concaténées en un seul vecteur de caractéristiques pour représenter chaque activité. Enfin, ce vecteur a été utilisé comme entrée pour le classificateur SVM pour identifier la classe de l'activité. La chute est confirmée en utilisant la région du sol et la stratégie de vote majoritaire.

3.4 Structure du système proposé

3.4.1 Détection de la personne et la segmentation du sol

La première étape de notre méthode proposée consiste à extraire la silhouette humaine de l'arrière-plan. Selon la littérature, il existe de nombreux algorithmes proposés pour la détection d'objets en mouvement tels que l'algorithme basé sur GMM [36], le modèle CB [35] et le AMF [37]. Pour une comparaison et plus de détails, consultez la référence [33]. Une autre approche utilisant la décomposition du signal a été proposée dans [137]. Cette approche ne convient pas pour une utilisation où le traitement de l'image prend beaucoup de temps. En outre, l'apprentissage approfondi a été appliqué à la soustraction d'arrière-plan comme dans [138]. Malgré ses avantages pour l'extraction de la silhouette humaine, le temps de traitement et l'utilisation d'un nombre important de bases de données pour l'apprentissage constituent toujours des contraintes pour les systèmes d'exploitation avec l'unité CPU.

Pour notre système, nous avons utilisé l'algorithme CB en raison de ses avantages et de sa robustesse pour détecter les objets en mouvement et avec un temps d'exécution faible. Le résultat de la BS n'est généralement pas toujours satisfaisant à cause de l'ombre et du déplacement des meubles dans le modèle d'arrière-plan. Pour cette raison, nous avons d'abord voulu détecter et supprimer l'ombre du premier plan en s'inspirant de la méthode de [139]. Nous avons estimé l'ombre en utilisant la couleur

HSV et les informations de gradient, puis nous avons classé les pixels de l'ombre en fonction de seuils prédéfinis. Le résultat obtenu à l'aide de notre méthode d'algorithme CB contient de nombreux objets ou de blobs. Afin de déterminer la silhouette humaine, nous avons utilisé deux règles : (1) suppression de tous les blobs qui ont une petite zone (c'est-à-dire zone < 50 pixels) et (2) fusion de blob en classant les autres blobs dans de nombreuses classes en utilisant la distance du rectangle. La distance est définie comme la distance minimale de 4 [140] entre deux rectangles basée sur la formule 4.5.

$$Distance(B1, B2) = \min_{P1 \in R1, P2 \in R2} d_4(P1, P2) \quad (3.1)$$

Où B1, B2 correspond au blob 1 et au blob 2 ; P1 et P2 sont les points les plus proches des rectangles R1 et R2 respectivement. Si la distance calculée entre deux blobs est inférieure à 50 pixels, alors, ils sont dans la même classe, sinon, chacun est dans une classe différente. La figure 3.3 montre les différentes distances entre deux blobs en fonction de leurs positions.

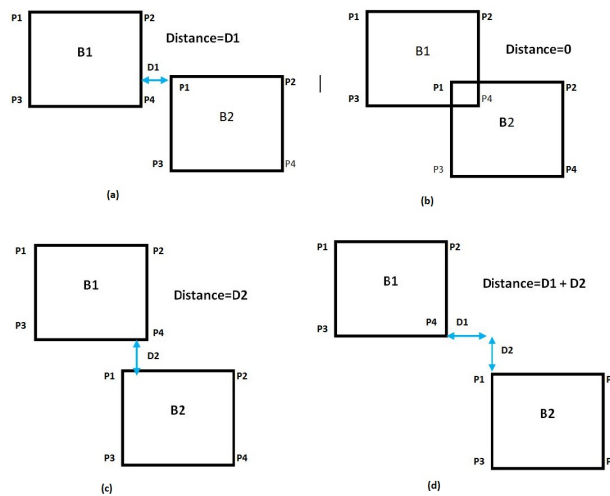


FIGURE 3.3 – Les quatre positions et distances entre deux blobs

Ensuite, nous avons déterminé la silhouette humaine en utilisant le mouvement des pixels du blob basé sur le résultat du flux optique et la distance entre la position actuelle de chaque classe et la position précédente. Ainsi, les blobs qui ont une petite distance et le mouvement élevé sont considérés comme le blob requis.

Dans la Figure 3.4, nous montrons le résultat de notre méthode d'extraction de la silhouette humaine. Tout d'abord, nous avons appliqué une méthode CB pour détecter les objets en mouvement en arrière-plan (c), puis nous avons détecté l'ombre (d). Ensuite, en utilisant une simple soustraction (différence pixel par pixel entre deux images), nous avons enlevé l'ombre de (e). Enfin, nous avons supprimé les petites zones (blobs) en utilisant la fusion de blob comme indiqué dans la dernière image (f).

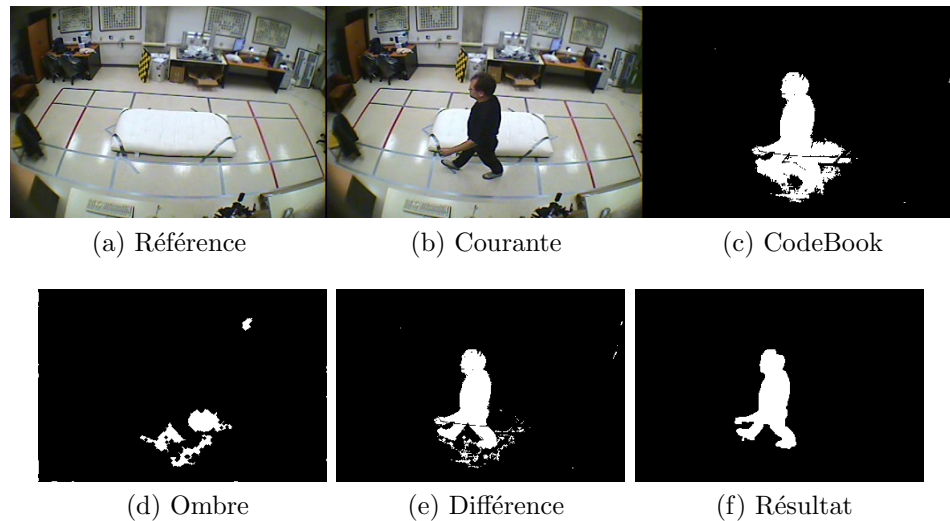


FIGURE 3.4 – Illustration de l'extraction de la personne. (a) Image de référence (b) Image courante, (c) Résultat de CB, (d) Détection de l'ombre, (e) Résultat de la soustraction par pixel de l'image (c) et l'image (d). (f) La segmentation de la silhouette

3.4.1.1 Extraction du sol

Comme l'événement de chute se termine toujours au sol, la détection du sol doit être déterminée en premier. Avant la phase de détection de chute, nous devons détecter la région du sol pour confirmer si l'activité actuelle peut être classée comme une chute ou une activité normale. A la place de la méthode manuelle, de nombreuses méthodes non supervisées [87, 18] ont été proposées pour estimer le plan du sol à partir de l'image. Dans [18], les auteurs ont marqué le sol comme la région proche du point extrême inférieur de l'ellipse lorsque la posture de la personne est classée comme debout ou assise. Néanmoins, les auteurs de [87] ont estimé le plan du sol en utilisant la carte de disparité obtenue par la caméra Kinect. Les méthodes supervisées ont été proposées comme discuté dans l'enquête [141]. Dans ce travail, nous avons utilisé la méthode dans [142] où les auteurs ont proposé un modèle SegNet, qui est une architecture codeur-décodeur profond pour la segmentation de pixels multi-classes. Il est efficace pour la compréhension de la scène en intérieur et également pour la segmentation de la scène de la route. Il peut segmenter 37 classes de scènes d'intérieur, y compris le mur, le sol, le plafond, la table, la chaise, le canapé et autres. Cependant, les personnes ne sont pas prises en compte. Pour cette raison, la seule façon d'utiliser cette méthode dans la détection de chute est dans la première image d'une vidéo, qui ne contient pas une personne. Dans la Figure 3.5, nous montrons le résultat obtenu en utilisant la méthode SegNet [142] testée sur notre jeu de données. Cela donne de bons résultats lorsque la personne n'est pas à l'intérieur de l'image. Ainsi, la région de sol correspond aux pixels verts.

Dans notre cas, nous nous sommes basés uniquement sur le processus manuel d'extraction de sol pour plusieurs raisons, la méthode SegNet ne prend pas en charge

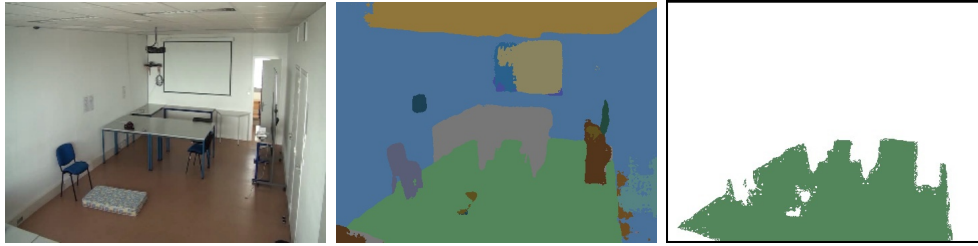


FIGURE 3.5 – Extraction du sol. Après l’utilisation de la segmentation par SegNet, nous segmentons le sol à travers cet résultat, qui correspond aux pixels avec la couleur verts dont ils appartient au même class(sol).

TABLEAU 3.1 – La description des caractéristiques

Caractéristiques	Description
F1	La vitesse de la personne à base de flux optique
F2	La vitesse de la région de la tête à base de flux optique
F3	La vitesse du région du centre à base de flux optique
F4	La vitesse de la région des pieds à base de flux optique
F5	Rapport de BB
F6	Rapport de la hauteur et la longueur de l’histogramme
F7	L’orientation de l’ Ellipse
F8	La direction du mouvement

la segmentation humaine, et il n’y a aucune information de profondeur dans notre base de données. La plupart des vidéos de la base de données enregistrées sont courtes, ce qui conduit à une incapacité à construire un segment de sol à partir de postures humaines comme dans [87, 18]. Compte tenu de ces limites dans nos bases de données, nous avons décidé d’extraire manuellement les informations de sol pour prouver et montrer son efficacité en tant que caractéristique de notre système de détection de chute

3.4.2 Extraction des caractéristiques

Après BS, nous proposons d’utiliser la silhouette humaine pour extraire plusieurs caractéristiques basées sur BB et l’ellipse. La variation de mouvement temporel et les informations de forme jouent un rôle important dans la détection et la reconnaissance des activités humaines. La chute est caractérisée par une déformation de forme et un mouvement élevé. En conséquence, nous avons proposé plusieurs caractéristiques, comme indiqué dans le tableau 3.1, qui décrivent plus précisément la forme et la variation de mouvement.

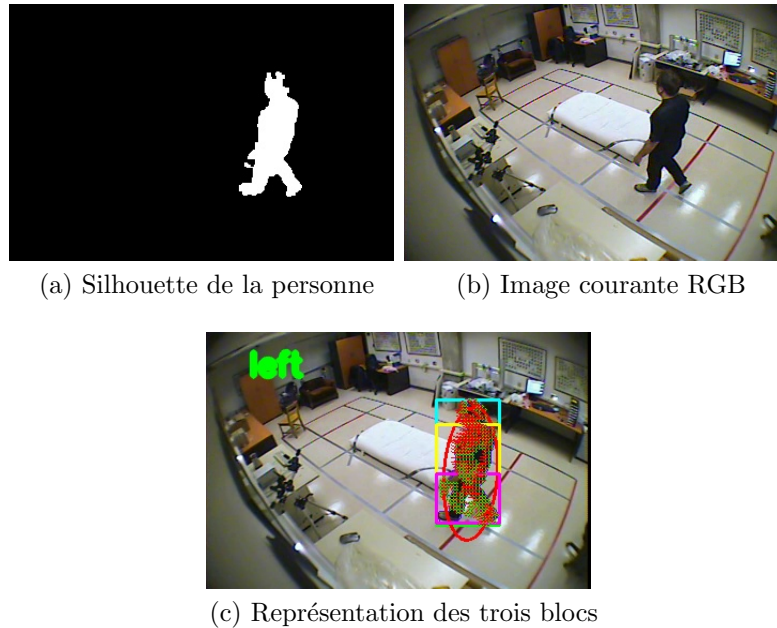


FIGURE 3.6 – L’extraction de la silhouette de la personne et la représentation des trois blocs. (a)Image original (b) Résultat de la BS, (c) La représentation des trois blocs et le flux optique, Le mot "left" dans l’image indique la direction du mouvement de la personne

3.4.2.1 Les caractéristiques de la forme

Pour décrire la déformation de la forme, nous avons extrait trois caractéristiques qui sont F5, F6 et F7. L’entité F5 peut être extraite de BB dessiné autour de la personne comme montre la figure 3.6. Le RA (R_B) est le rapport entre la hauteur et la largeur du BB. Pendant les activités d’une personne à la maison, la hauteur et la largeur changent lorsque la personne change de posture ; le R_B changera également.

Nous avons extrait la caractéristique F6 de l’histogramme de projection en se basant sur la silhouette 2D de la personne. Comme le montre la figure 3.7, nous avons projeté la silhouette sur l’axe y et l’axe x , ce qui donne des projections horizontales et verticales. La valeur de chaque bin dans l’histogramme correspond au nombre de pixels blancs dans une ligne ou une colonne de l’image. La valeur la plus élevée du bin horizontal ou vertical est considérée comme H_p et V_p . Le **rapport d’aspect** (R_H) entre H_p et V_p est considéré comme une caractéristique.

La caractéristique F7 peut être extraite de l’ellipse ajustée au corps humain comme le montre la figure 3.6. Tout d’abord, la méthode basée sur le moment [33] est appliquée pour ajuster l’ellipse. Ensuite, nous avons extrait l’**orientation** de la personne (θ). Dans la figure 3.2, nous observons que l’ajustement de l’ellipse peut décrire la posture du corps humain en utilisant l’orientation de l’ellipse.

En considérant que chaque activité prend plusieurs images, nous proposons d’analyser la déformation de la forme via la fenêtre de taille fixe W . W est le nombre

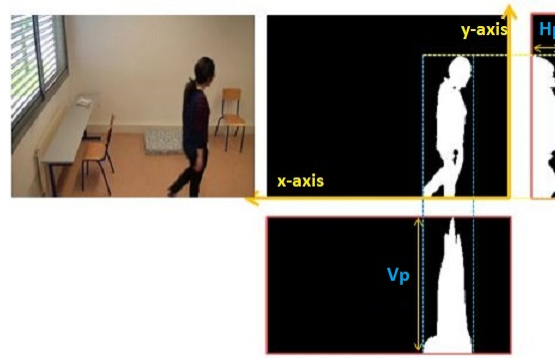


FIGURE 3.7 – Le résultat de la projection d’histogramme de la silhouette de la personne

d’images dans la fenêtre. Pour cela, la variation de la déformation de la forme au sein d’une fenêtre peut être représentée par le vecteur de caractéristiques défini dans l’équation 3.2.

$$F_{Forme} = [V_{R_B}, V_{R_H}, V_{\theta}] \quad (3.2)$$

où, V_{R_B} , V_{R_H} et V_{θ} représentent respectivement une entité vectorielle d’entités F5, F6 et F7 dans W images.

$$V_{R_B} = [R_b^1, \dots, R_b^W], V_{R_H} = [R_h^1, \dots, R_h^W], V_{\theta} = [\theta^1, \dots, \theta^W] \quad (3.3)$$

3.4.2.2 Les caractéristiques de mouvement

Le flux optique est un champ de déplacement visuel qui permet d’expliquer les variations d’une image en mouvement en termes de déplacement de points d’image. Dans la littérature, il existe plusieurs approches de détection de mouvement qui utilisent le flux optique telles que [143] et [144], où les auteurs calculent le flux pour chaque pixel dans la première image, et ils utilisent l’échelle multiple pour suivre les caractéristiques éparses.

Dans notre système, nous avons utilisé l’algorithme présenté dans [143]. Le suivi des pyramides d’images permet de capturer de grands mouvements par les fenêtres locales [143]. Le flux optique peut donner deux informations importantes comme caractéristiques pour analyser le comportement humain, en particulier la **vitesse** et la **direction** du mouvement de la personne pour discriminer entre deux activités : activités normales et activités anormales.

Certaines activités normales sont similaires à un événement de chute. Utiliser uniquement la vitesse de la personne ne suffit pas afin de faire la distinction entre ces activités. Lors d’une chute, par rapport à une activité couchée, il y a un mouvement élevé dans le changement de tête et de la région centrale du corps tandis que le mouvement dans la région des pieds est presque stable. Par conséquent, nous représentons la forme de la personne en trois blocs. Nous avons divisé le BB en trois blocs (tête, centre et jambes) comme le montre la figure 3.2. Les trois blocs ont un

ratio de 20% : 40% : 40 % respectivement. On estime que le rapport de 20% : 40% : 40% fait la distinction entre les parties supérieure, moyenne et inférieure du corps sur la base des expériences.

À partir de la silhouette de la personne et les trois blocs, comme le montre la figure 3.2, nous avons calculé la vitesse de la personne (F1) et la vitesse de chaque bloc (F2, F3 et F4) en fonction du résultat de l'algorithme de flux optique en utilisant ces formules :

$$Vitesse(Block_i) = \frac{\sum_1^{N_i} Vitesse(Pixel_{blob \cap Block_i})}{N_i} \quad (3.4)$$

$$Vitesse(personne) = \frac{1}{3} \sum_1^3 Vitesse(Block_i) \quad (3.5)$$

Où $Pixel_{blob \cap Block_i}$ signifie tous les pixels qui appartiennent à l'intersection du blob d'une personne et $Block_i$, $Vitesse(Block_i)$ signifie la vitesse de $Block_i$. $Vitesse(personne)$ désigne la vitesse de la personne. N_i est le nombre total de pixels dans chaque $Block_i$.

La direction du mouvement (F8) est la dernière caractéristique extraite du résultat du flux optique. L'orientation du déplacement du pixel est utilisée pour calculer la direction du mouvement. Quatre directions sont définies, à savoir **Haut**, **Gauche**, **Droit** et **Vers le bas** comme le montre la figure 3.8. Dans notre système, nous avons défini la direction bas par la valeur 1 qui correspond à la chute et 0 pour les autres directions. Puisqu'une chute, comme d'autres activités, est caractérisée par une va-

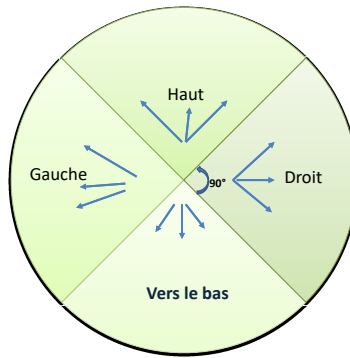


FIGURE 3.8 – Les quatre directions du mouvement possibles de la personne

riation de mouvement. Il est nécessaire de capturer plus d'informations sur cette variation à partir d'un groupe d'images. En conséquence, pour chaque activité, nous avons extrait cinq vecteurs caractéristiques du mouvement à partir d'images W qui sont définis comme suit :

$$V_P = [v_p^1, \dots, v_p^W], V_D = [d^1, \dots, d^W] \quad (3.6)$$

$$V_H = [v_h^1, \dots, v_h^W], V_C = [v_c^1, \dots, v_c^W], V_F = [v_f^1, \dots, v_f^W] \quad (3.7)$$

Où V_P est le vecteur de la vitesse d'une personne, V_D est le vecteur de la direction du mouvement, V_H est le vecteur de la vitesse de la tête, V_C est le vecteur de la vitesse du centre et V_F est un vecteur de la vitesse des pieds.

Enfin, nous avons concaténé tous ces vecteurs pour construire un vecteur de caractéristiques de mouvement défini comme indiqué dans cette formule :

$$F_{Mouvement} = [V_P, V_H, V_C, V_F, V_D] \quad (3.8)$$

3.4.2.3 Notre descripteur proposé

Chaque segment vidéo est caractérisé par une fonction vectorielle où nous avons concaténé des caractéristiques de mouvement (eq 3.8) et des caractéristiques de forme (eq 3.2). Le vecteur de caractéristiques est défini comme suit :

$$V = [F_{Mouvement}, F_{Forme}] \quad (3.9)$$

Ce vecteur est alimenté par un classificateur. Nous avons testé notre approche sur différents classificateurs tels que SVM, NN et ELM.

3.4.3 La structure proposée pour la reconnaissance de chute

La détection de chute est formulée comme un problème de classification binaire qui distingue entre la chute et des ADL ce qui signifie le nombre total de classes $K = 2$. Les activités normales sont traitées comme une classe négative, qui consiste à marcher, s'asseoir, se coucher, s'accroupir et se pencher. Les activités de marche sont envisagées car ce sont les activités les plus courantes de la vie quotidienne. Les autres sont considérés car ils sont proches de la chute des activités, ce qui confond le classifieur. En conséquence, ces activités peuvent montrer la robustesse de nos caractéristiques proposées.

Étant donné un ensemble d'entraînement $X = (x_i, y_i)_{i=1}^N$, où x_i est le vecteur caractéristique de l'activité, y_i correspond au libellé de la classe ($y_i \in \{-1, 1\}$) et N est le nombre total de données d'entraînement. Un classificateur SVM binaire [145] est entraîné avec X . En donnant n'importe quel vecteur caractéristique x représentant un segment d'activité de test, le classificateur SVM renvoie la marge de sortie a en utilisant une fonction signe, $y = \text{signe}(a)$ pour indiquer l'étiquette de classe de l'entrée x . La chute est indiquée par $y = +1$.

Dans la figure 3.9, nous montrons le schéma général de la classification des activités de chute et non chute. Pour chaque activité, nous avons extrait toutes les caractéristiques vectorielles et les avons concaténées pour construire un descripteur vectorielle. À partir de la fonction de vecteur N des activités, y compris la chute et l'ADL, nous avons construit des données d'entraînement pour entraîner le classificateur SVM qui va ensuite être utilisé pour classer les activités comme chute ou ADL dans la phase de test.

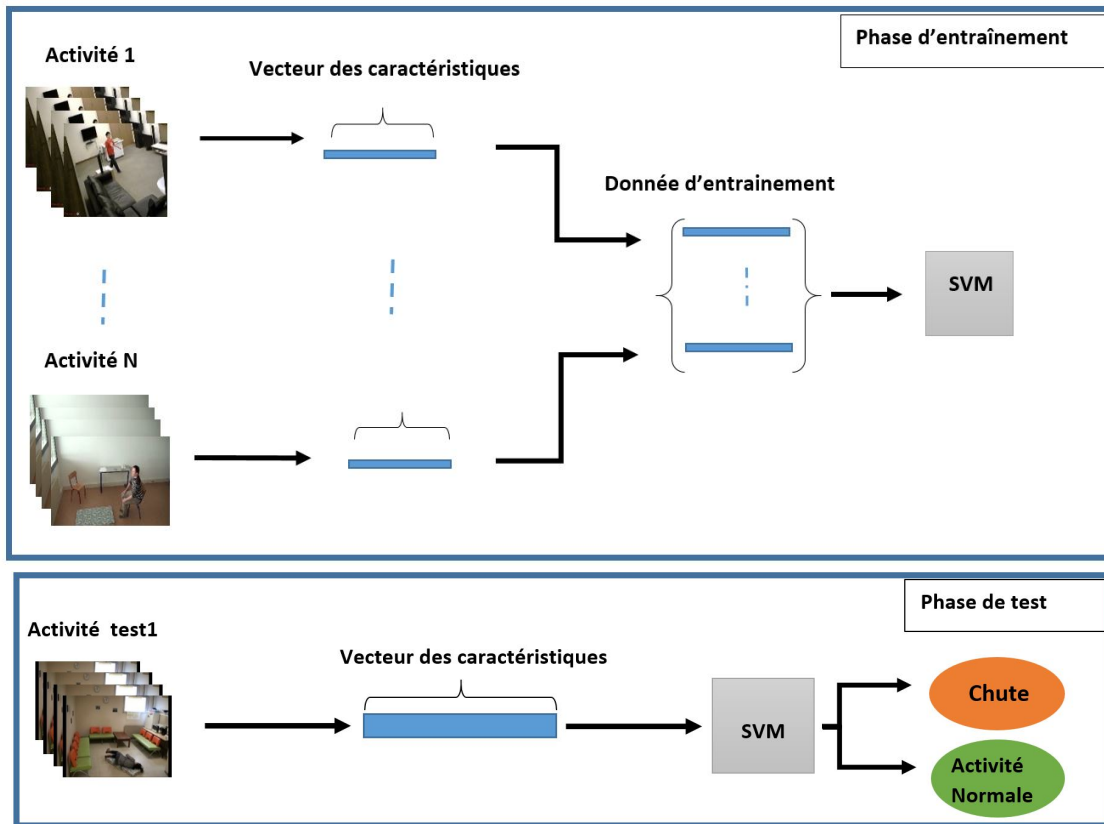


FIGURE 3.9 – Le schéma proposé pour la reconnaissance des activités

3.4.4 Méthode de segmentation des activités dans la vidéo

Cette partie décrit la méthode manuelle pour obtenir un segment vidéo qui contient uniquement un événement (activité) intéressant à partir d'une vidéo. Pour segmenter chaque activité de la vidéo, nous nous sommes basés sur la durée de la séquence de chute. Des bénévoles d'âges et de hauteurs différents ont simulé les activités. En outre, les vidéos ont été capturées à différentes fréquences d'images (par exemple 25 ips, 30 ips, 120 ips). Par conséquent, nous avons constaté que la durée des activités est différente entre chaque base de données. En considérant toutes les vidéos, nous avons observé que la durée d'une séquence d'activités de chute est en moyenne de 20 ou 40 images, qui commence en position debout ou assise et se termine par allongée sur le sol. Pour conserver tous les vecteurs de caractéristiques de sortie de segment vidéo à la même longueur, nous avons décidé de diviser la séquence d'activités qui prend 40 images en deux segments de 20 images. Le premier et le deuxième segment sont composés des images avec des indexes impairs et pairs respectivement. Ainsi, nous avons considéré 20 images comme la durée de segment dans toutes les expériences. Un autre facteur que nous avons pris en compte était la taille de l'image. Bien que les tailles d'image dans les vidéos soient différentes car les bases de données contiennent des vidéos réalisés avec différents types de caméras, nous avons normalisé toutes les tailles d'image en $320 * 240$.

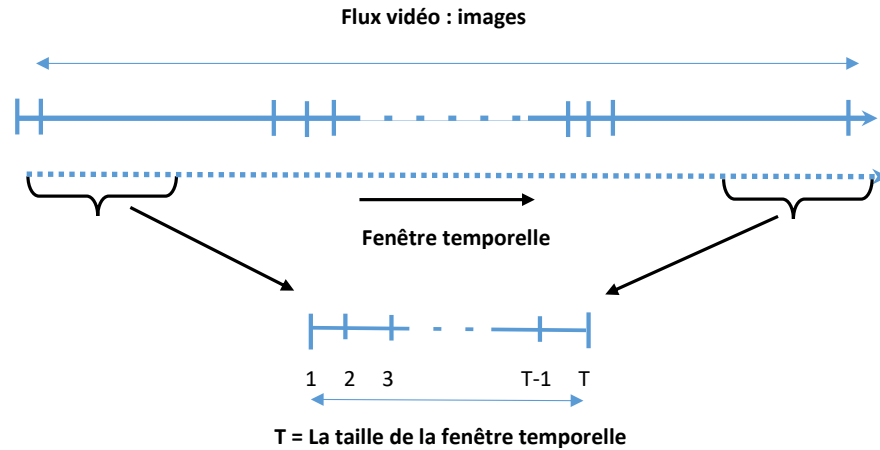


FIGURE 3.10 – Illustration de la proposition de la fenêtre temporelle utilisée pour la détection de chute dans une séquence de vidéo

3.4.5 Méthode proposée pour la reconnaissance de la chute

Avec la détection des chutes capturée dans la vidéo, nous devons déterminer le début et la fin des activités pour la classification. Comme indiqué dans la section 3.4.4, les activités humaines sont segmentées manuellement pour la classification. Il existe de nombreuses techniques pour effectuer automatiquement la segmentation vidéo des activités humaines telles que le HMM [146] et la Chaîne de Markov Monte Carlo (MCMC) [147]. Dans [55], ils ont observé que le rapport de BB est relativement faible lorsque la personne marche ou se tient debout. Mais, le ratio augmente lorsque la personne tombe. En conséquence, la segmentation est effectuée en trouvant la partie de transition de la vidéo brute qui contient une augmentation significative (bords montants) du rapport hauteur/largeur de la cible. Ils ont utilisé le MCMC pour trouver cette transition.

Dans ce travail, nous avons formulé notre détection des chutes en utilisant la stratégie du vote à la majorité. Bien que nous n'ayons aucune information sur le début des activités, et en considérant qu'entre l'activité de début et de fin, il existe de nombreuses activités similaires, nous avons décidé d'utiliser la fenêtre temporelle comme le montre la figure 3.10. En considérant T comme la durée de la fenêtre temporelle, on a $A = \{a_1, \dots, a_T\}$ comme l'ensemble des activités classées dans la fenêtre temporelle, $a_j \in Y = \{-1, 1\}$, où $j = 1, \dots, T$ et Y est l'ensemble des étiquettes (non-chute, chute). S'il y a plus de dix fois d'activité classée comme chute dans une fenêtre temporelle, alors nous avons considéré cet événement dans son ensemble comme un incident de chute. La condition peut être formulée comme dans l'équation 3.10.

$$I_T = 1_{\{\#\{a_i=-1\} \geq 10\}} \quad (3.10)$$

Nous avons utilisé les informations du sol et le temps d'inactivité pour la confirmation de chute. Nous avons vérifié si le corps entier ou une grande partie du corps

était à l'intérieur de la région du sol ou non, comme le montre la figure 3.11. La chute est confirmée lorsque plus de 75 % de la partie du corps qui se trouve à l'intérieur du sol et que le temps d'inactivité est supérieur à dix images. Alg. 4.1 montre la stratégie de détection des chutes dans la vidéo.

Algorithme 3.1 La stratégie pour détecter la chute

Input : Temporal window A, Body Area (BA), Inactivity time threshold (t).

Output : Fall

REPEAT

1. StrategyOfVote(A)
2. CheckInsideFloor(BA)
3. InactivityTime (t)

UNTIL the conditions tested in steps 1, 2 and 3 are **True**

RETURN Fall

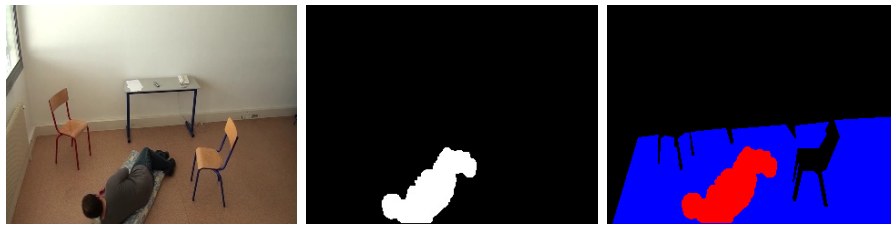


FIGURE 3.11 – Illustration de la chute au sol. La région en bleu représente le sol. La région en rouge représente l'intersection entre la région du sol et la silhouette de la personne

3.5 Expérimentation et analyse de performance

Pour évaluer notre méthode proposée, nous avons utilisé trois bases de données [3] [4] [2]. Les tableaux 3.3 et 3.2 montrent les informations quantitatives et certains défis de différents bases de données. Toutes les vidéos de chaque base de données ne contiennent qu'une seule personne car il n'est pas nécessaire que notre système fonctionne, s'il y en a plus d'une à la maison, la personne tombée pourrait être aidée par l'autre personne. Plus de détails sur chaque base de données sont fournis ci-dessous.

Base de données 1 (D1)[3] : A partir de cette base de données, nous avons segmenté les activités à partir de chaque vidéo en utilisant la méthode décrite dans la section 3.4.4.

Base de données 2 (D2) [4] : Nous avons créé deux bases de données appelés D21 & D22 à partir de cette base en utilisant la méthode décrite dans la section 3.4.4. A partir des expérimentations, nous avons observé que la durée de tous les activités de chute dans cette base prennent en moyenne 40 images (presque 2 secondes). Chaque scénario, D21 ou D22, contient 117 segments de chutes et 178 d'activités normales. En

TABLEAU 3.2 – Les difficultés et les challenges de chaque base de données

Base de données	Activités	Scène	Texture	Ombre	Meubles	Occlusion
D1	Tous les types, avec tous les angles de vue	Mobile, Claire	Complexe	Oui	Oui	Non
D2	Tous les types, avec tous les angles de vue	Immobile, claire	Complexe	Oui	Non	Non
D3	Tous les types, avec tous les angles de vue	Mobile, Bruité	Trop Complexe	Oui	Oui	Oui

TABLEAU 3.3 – Les informations quantitative de la base de données de l'expérimentation

Classe	Activités	D1	D2	D3
+1	chutes	# 29	# 234	# 200
-1	ADL	# 45	# 356	# 288

total, D21 + D22, nous avons 234 segments de chutes et 356 segments des activités normaux.

Base de données 3 (D3)[2] : Nous avons segmenté des activités normaux et des chutes à travers chaque vidéo en utilisant la méthode décrite dans la section 3.4.4. En total, nous avons 200 segments des chutes et 288 de segments des activités normales.

3.5.1 Analyse des caractéristiques proposées

Durant notre expérimentation, l'extraction des fonctionnalités des vidéos a été implémentée en C / C++ en utilisant la bibliothèque OpenCV avec Visual Studio 2013. La procédure de classification a été implémentée dans MATLAB. Toutes les expériences ont été menées sur un PC équipé d'un processeur Intel (R) Core i7 et de 12 Go de RAM. Nous avons effectué notre détection des chutes en utilisant la classification binaire (chute vs non-chute). Basé sur libSVM [145], nous avons utilisé le classificateur C-SVM avec la RBF. Le paramètre par défaut a été utilisé sauf pour gamma (g) de RBF et le coût (c) de C-SVM qui ont été modifiés avec des valeurs de 0,01 et 100 respectivement. La validation croisée par dix a été utilisée ici. Avant de présenter les performances de notre système, nous illustrons les résultats de la méthode BS et les caractéristiques extraites des activités de la personne.

Dans les figures 3.12, 3.13 et 3.14, la différence entre chaque activité (c'est-à-dire marcher, tomber et se coucher) est indiquée dans le graphique, en particulier durant la variations des caractéristiques de mouvement et la variation de l'orientation. Les trajectoires de l'activité de marcher sont presque stables, tandis que dans les autres

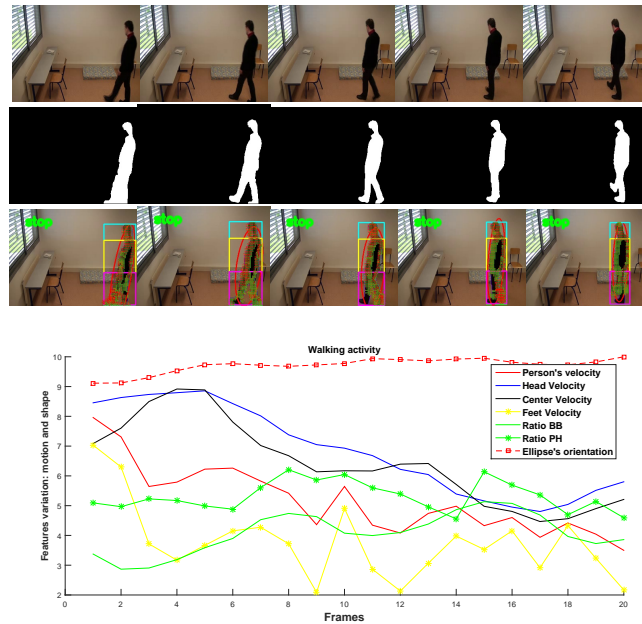


FIGURE 3.12 – Les images des rangées 1, 2 et 3 correspondent respectivement au cadre de référence, à l'extraction de silhouette humaine et à trois blocs de représentation d'une activité de la marche. Le graphique illustre la variation des caractéristiques d'une activité de marche pendant 20 images.

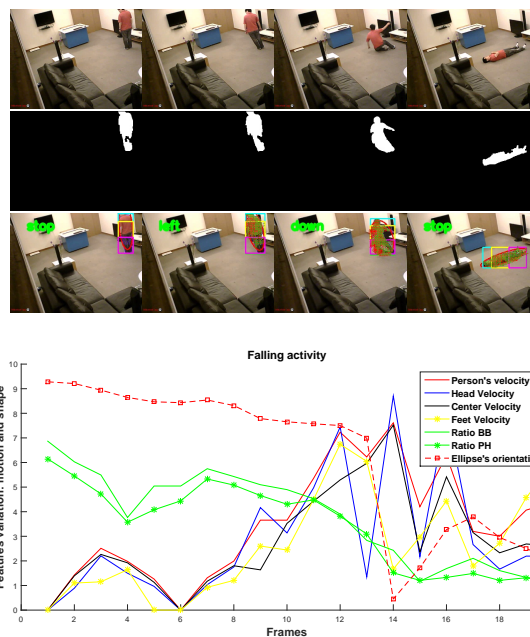


FIGURE 3.13 – Les images des rangées 1, 2 et 3 correspondent respectivement à l'image de référence, à l'extraction de silhouette humaine et à trois blocs de représentation d'une activité de la chute. Le graphique illustre la variation des caractéristiques d'une activité de la chute pendant 20 images.

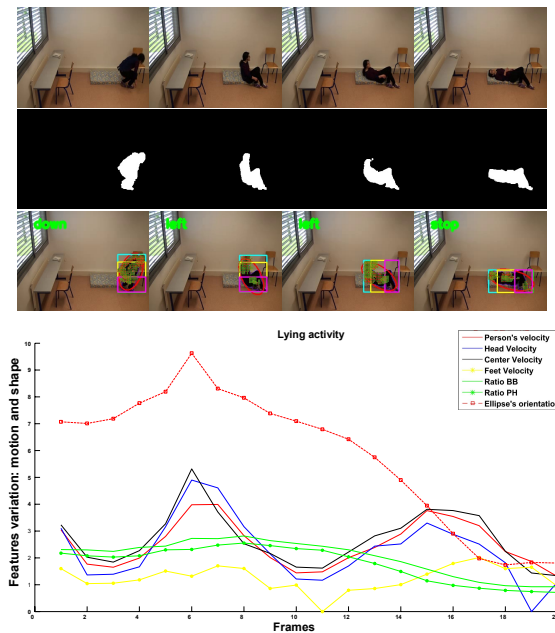


FIGURE 3.14 – Les images des rangées 1, 2 et 3 correspondent respectivement au cadre de référence, à l'extraction de silhouette humaine et à trois blocs de représentation d'une activité de se coucher. Le graphique illustre la variation des caractéristiques de l'activité couchée pendant 20 images.

activités, il y a des discontinuités brutales. La différence entre l'activité de chute et de se coucher est que la variation des caractéristiques prend plus de temps pour se changer lors de l'activité de couchage, alors qu'elles changent rapidement lors d'une chute.

3.5.2 Évaluation quantitative des caractéristiques proposées

Dans cette section, nous évaluons la classification entre les caractéristiques de forme, les caractéristiques de mouvement et la combinaison entre eux. Le classifieur SVM avec la validation croisée 10-fois a été utilisé .

Dans le tableau 4.3, nous montrons le résultat de la classification de chaque caractéristique. Il est clair que l'utilisation de la combinaison offre un taux de précision élevé par rapport à l'utilisation de l'une ou l'autre des caractéristiques seules.

TABLEAU 3.4 – Résultat de l'évaluation des caractéristiques

Base de donnée	Caractéristique de forme	Caractéristique de mouvement	Combinés
D1	93.24%	89.18%	94.60%
D21	89.15%	94.91%	95.93%
D22	90.81%	94.89%	95.91%
D3	90.36%	91.59%	96.31%

3.5.3 Évaluation par scénario de notre approche et la classification

Dans cette partie, nous évaluons notre approche avec les bases de données en utilisant deux scénarios, et aussi, nous comparons nos résultats de système avec les méthodes de l'état de l'art.

3.5.3.1 Évaluation par Scénario

Nous évaluons la performance de notre approche proposée à l'aide de deux scénarios. Le premier scénario concerne le test de notre approche sur chaque base de données individuellement, tandis que le second scénario est un test concernant un mélange de toutes les bases de données.

TABLEAU 3.5 – La performance de la méthode proposé en utilisant le premier scénario

Base de données	Exactitude (Accu)
D1	94.60%
D21	95.91%
D22	95.23%
D3	96.31%

Dans le tableau 3.5, nous montrons les performances obtenues par la méthode proposée en fonction du taux de précision en utilisant le premier scénario. Le schéma proposé montre une grande précision sur la base de données D2 et D3, alors qu'elle l'est moins pour la base de données D1. La cause du taux de précision moindre pour D1 pourrait être justifiée par le fait que la phase d'apprentissage ne dispose pas de suffisamment de données pour faire la distinction entre des activités similaires. Une autre cause possible est que l'ensemble de données contient des activités qui sont extrêmement similaires à une chute, par exemple, les activités accroupies et couchées ont un mouvement et une déformation de forme élevée.

TABLEAU 3.6 – La performance de la méthode proposé en utilisant le 2ème scénario

Base de données	Exactitude(Acc)
D1 + D21	95.65%
D1 + D22	94.83%
D1 + D21+ D22	97.43%
D1 + D3	94.66%
D21 + D22	98.12%
D21 + D22 + D3	96.75%
D1 + D21 + D22 + D3	96%

Dans le second scénario, comme montre le Tableau 3.6, nous avons observé que les résultats obtenus à partir de la méthode proposée sont améliorés dans certains cas et diminués dans d'autres. De plus, lorsque nous avons fusionné les bases de données

D21 et D22, la précision était de 98.12%, ce qui signifie que le système a bien appris à partir d'une base de données assez grande et devient efficace pour faire la distinction entre des activités similaires. De plus, lorsque nous avons fusionné D1 avec D21 et D22, la précision est passée à 97.43%. Cependant, en fusionnant la base de données D3 avec D1 ou D3 et les bases de données (D22 et D21), le résultat n'est pas aussi bon. Ainsi, on peut expliquer que les caractéristiques de la base de données D3 sont plus complexes que les autres base de données. Une autre cause possible pourrait être que prendre seulement 20 images comme durée de chaque activité, comme décrit dans la section 3.4.4, n'est pas suffisant pour discriminer entre les activités similaires. Dans la dernière expérience, nous avons fusionné toutes les bases de données (D1 + D21+D22 + D3), et même le système est entraîné avec plus de données, le résultat n'est pas bien amélioré où il atteint seulement 96%, ce qui est 0.75% de moins que le résultat obtenu en utilisant la base de données composée de D1 et (D21 et D22).

Dans le Tableau 3.7, nous montrons la comparaison entre trois classifieurs dont SVM, NN et ELM. Après avoir le tableau, le classifieur SVM offre une grande précision par rapport aux autres classifieurs NN et ELM.

TABLEAU 3.7 – Comparaison des différents classifieurs pour deux classes, chute et non-chute

Base de données	SVM	NN	ELM
D1	94.60 %	85%	85%
D21	95.91%	96%	94.60%
D22	95.23%	85.34%	85.33%
D3	96.31%	90.98%	93.44%
D21 + D22	98.12%	91.21%	93.24%
D1+ D21	95.65%	90.32%	94.62%
D1 + D22	94.83%	88.17%	88.17%
D1 + D3	94.12%	87.32%	88.73%
D3 + D21	95.01%	88.32%	87.81%
D3 + D22	94.89%	86.80%	88.32%
D1 + D21 + D22	97.43%	89.82%	94.61%
D3 + D21 + D22	96.75%	88.15%	91.11%
D3 + D1 + D21 + D22	96%	91.35%	93.08%

3.5.3.2 Comparaison avec les approches dans la littérature

Dans cette partie, nous comparons notre schéma proposé avec les méthodes discutées dans la section 3.2. La comparaison est basée sur les performances obtenues par chaque approche en utilisant les mêmes jeux de données.

Depuis le Tableau 3.8, nous observons que le résultat du schéma proposé est meilleur que les méthodes présentées dans [31], [116], [119], [108], [31] et [3]. Par rapport aux méthodes [31] [3] [116] qui se basent uniquement sur la variation de forme, la variation de mouvement est nécessaire pour discriminer entre une vraie chute et les

activités normal qui sont similaires aux chutes. La variation de forme présentée dans les méthodes de l'état de l'art donne une précision supérieure à notre méthode basé sur la variation de forme. Cependant, lorsque nous ajoutons la variation de mouvement, le résultat obtenu est amélioré et notre méthode atteint une grande précision. En considérant la comparaison avec les méthodes proposées dans [55] et [117], le schéma proposé atteint des résultats comparables même s'il surpasse constamment.

Nous comparons également notre méthode avec le travail de [49]. Comme on peut le voir dans le tableau 3.8, nous avons considéré l'indicateur de précision pour la comparaison car c'est le seul indicateur utilisé dans ce travail. Le schéma proposé montre des performances élevées dans la détection des chutes avec une précision de 100%, tandis que l'autre méthode n'atteint que 93,25%.

De plus, nous avons comparé notre méthode avec la méthode dans [40] qui n'est pas incluse dans le tableau 3.8. Leur système est basé sur une caméra RGB-D. En considérant cette méthode où leur système a atteint 97,5%, cette haute performance pourrait s'expliquer par l'utilisation de la classification uniquement entre les activités de chute et les activités de se coucher. S'ils ajoutaient d'autres activités normales similaires (par exemple, s'accroupir, s'asseoir rapidement) à la classification, le système pourrait être confus, alors la précision serait moindre.

À partir des résultats de la comparaison, bien que les méthodes existantes sont basées sur plusieurs caméras RGB ou qu'elles ont fusionné les informations RGB et de profondeur, la comparaison donne quelques indications sur les performances et la robustesse de la méthode proposée.

TABLEAU 3.8 – Comparaison avec les méthodes de l'état de l'art

Base de données	Méthode	Précision	Exactitude(%)
D1	Chua et al. [3]	-	91.9%
	Notre méthode	-	94.60%
D2	Nguyen et al. [49]	93.25%	-
	Ismail et al. [73]	-	94.73%
	Harrou et el. [116]	-	96.84%
	Adrian et al. [108]	-	97.00%
	Notre méthode	100%	98.12%
D3	Rougier et al. [31]	-	95.6%
	Fan et al. [119]	-	95.2%
	Kaibo Fan et al. [117]	-	97.10%
	Yun et al. [55]	-	97.19%
	Notre méthode	-	96.31%

3.5.3.3 Résultat de notre système proposé

La reconnaissance du type d'activité et l'extraction du sol sont utilisées pour détecter l'événement de chute selon les trois conditions mentionnées dans la section 3.4.5. Le résultat de notre système est présenté dans la figure 4.15. La première

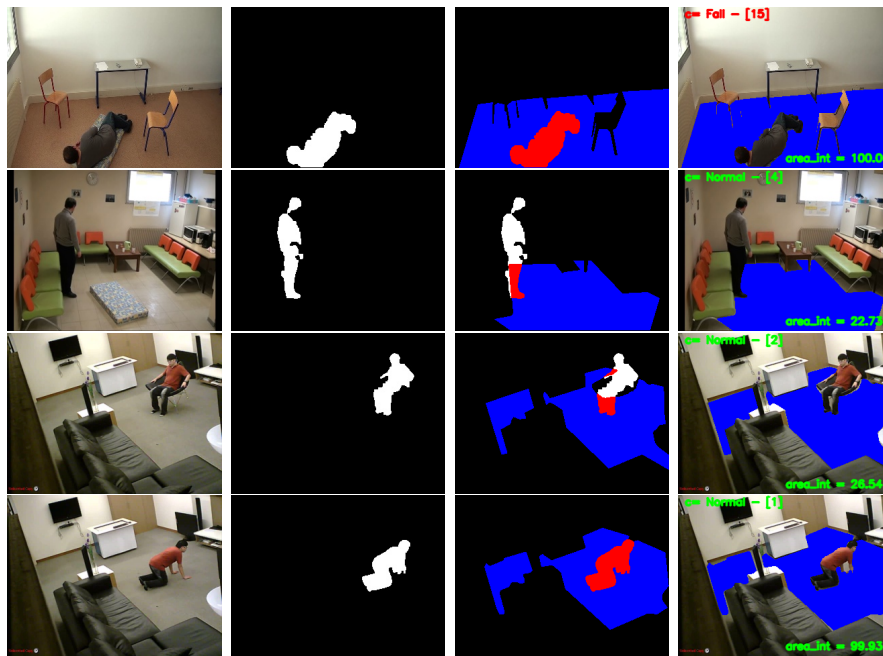


FIGURE 3.15 – Illustration de la reconnaissance des activités y compris la chute, debout, s’asseoir et accroupir.

ligne représente la chute au sol. La deuxième ligne représente l’action de debout sur le sol. La troisième ligne représente l’activité s’asseoir sur la chaise. Les images de références sont représentées par la première image de chaque ligne, la seconde image est la détection de silhouette humaine. La région bleue représente la détection du sol, la région rouge représente l’intersection du corps humain de premier plan avec la région de sol détectée. La région blanche représente la partie du corps humain de premier plan qui ne croise pas la région de sol. La dernière image représente le résultat de notre système.

Pour illustrer, nous présentons les quatre cas de la figure 4.15. Le premier cas illustre la personne qui tombe sur le sol, et l’activité est classée comme une chute. Comme la zone du corps est à l’intérieur du sol, en considérant un temps d’inactivité comme dix images, nous avons confirmé l’activité de chute. Dans le second cas, nous présentons que la personne est détectée comme debout dont l’activité est classée comme une activité normale. Comme la plus grande surface corporelle n’est pas à l’intérieur du sol, le système assure une activité normale. Dans le troisième cas, l’activité assise est détectée comme une activité normale parce que le classifieur l’assigne comme une activité normale. Dans le dernier cas, la posture de la personne est accroupie. Notre système l’a détectée comme activité normale. Malgré que la plus grande surface corporelle se trouve à l’intérieur du sol, notre système détecte correctement que cette activité est normal vu que la personne est toujours en mouvement au niveau de la tête.

3.5.3.4 Temps d'exécution de notre système

Pour tout système de surveillance de détection de chute, le temps de traitement joue un rôle important pour déclencher une alerte rapidement après la chute de la personne. Dans notre système, nous avons utilisé le matériel spécifique mentionné dans la section 3.5.1 pour tout le processus y compris le prétraitement, l'extraction de caractéristiques et la reconnaissance de la chute. Sans utiliser une optimisation de code ou de calcul parallèle, le temps moyen de prétraitement des données vidéo est de 0,24 seconde/image (0.24 s/f). La vitesse moyenne d'extraction des caractéristiques est d'environ 0.024 s/f. Pour la vérification des chutes, la vitesse moyenne est de 0.03 s/f. En considérant toutes les étapes, nous montrons que l'étape de pré-traitement a le temps de calcul le plus élevé, ce qui peut être justifié par l'utilisation de la méthode de flux optique. Pour accélérer le traitement, nous nous pencherons sur l'optimisation du code et nous nous concentrerons sur le calcul parallèle. Une autre clé est d'exploiter un FPGA (Field-programmable gate array) pour produire un système en temps réel.

Dans le Tableau 3.9, nous comparons notre système avec le travail associé dans le temps de traitement. Comme on peut le voir, les méthodes de [3] et [31] fournissent un temps d'exécution inférieur que notre méthode car elles n'utilisent que les informations de forme. Tant dis que l'extraction des informations de mouvement prend plus de temps. De plus, notre méthode est meilleurs que la méthode dans s[55].

TABLEAU 3.9 – Comparaison du système proposé et des méthodes de l'état de l'art en termes de temps d'exécution

Approche	Temps d'exécution	
Chua et al. [3]	0.19 s/f	5.26 FPS
Rougier et al. [31]	0.2 s/f	5 FPS
Yun et al. [55]	13.2 s/f	0.07 FPS
Notre méthode	0.33 s/f	3.03 FPS

3.5.4 Discussion

L'approche proposée est un système de détection de chute en utilisant une seule caméra. La vision par ordinateur fournit des informations plus complètes et détaillées sur la personne supervisée (par ex. Activité, posture, emplacement), ainsi que son environnement. Ce type de système est convenable dont aucune intervention humaine n'est nécessaire et aucun capteur à porter n'est nécessaire. L'évaluation a été menée sur trois bases de données connues. Malgré les challenges rencontrés dû à la différences entre ces bases de données, la fusion entre elles pour construire une seule base données d'apprentissage, et les difficultés de segmentation (ombre, objets en mouvement, vêtements différents, taille des personnes est différente, emplacements différents, etc.), les résultats de notre méthode de la reconnaissance de la chute sont très bonnes.

La combinaison des caractéristiques de forme et de mouvement est un moyen utile

de faire la distinction entre une activité normale et une activité anormale. Nous avons démontré que nos caractéristiques de forme extraites de la silhouette humaine et des caractéristiques de mouvement sont vraiment discriminantes pour la classification. L'étape de la procédure est totalement automatique et il n'est pas nécessaire de définir des seuils pour distinguer chaque activité.

Des caractéristiques utiles sont utilisées pour faire la distinction entre les activités normales et les activités anormales. La combinaison entre les informations de forme et la informations de mouvement améliore la classification des taux au lieu d'utiliser l'une ou l'autre de ces fonctions seule comme dans [3], [116] [31]. Ensuite, la confirmation de chute est appliquée lorsqu'une activité anormale est détectée par notre classifieur. Afin de faire la distinction entre une activité de chute réelle et une chute similaire, telle que l'activité couchée et s'accroupir, nous avons utilisé les informations du sol et le temps d'inactivité. La plupart des détections d'erreurs dans les systèmes proposés dans les approches précédentes se réfèrent à ce problème. Ainsi, en tant que solution, la confirmation à l'aide des informations de sol réduit la détection des erreurs.

Cependant, certains cas ne sont toujours pas pris en compte par notre système comme dans [3] [116]. La présence de plusieurs personnes à la maison n'est pas adéquate. S'il y a plus d'une personne à la maison, il n'est pas nécessaire de faire fonctionner notre système, l'autre personne peut demander de l'aide si la personne âgée tombe. En tant que solution, le système sera automatiquement désactivé. Il sera redémarré manuellement par la personne âgée ou automatiquement en utilisant certaines techniques de comptage de personnes telles que [148]. L'autre cas est le problème d'occlusion. L'environnement domestique contient souvent de nombreux objets comme un canapé, des chaises, des tables et autres. lorsque la personne est derrière l'un d'entre eux, cela provoque une occlusion et détériore les performances du système de détection de chute et surtout l'extraction de la silhouette de la personne. Pour éviter ce problème, l'ajout de plus d'une caméra pour surveiller la personne peut être utilisé pour garantir que tout le corps ou la majeure partie de son corps est au moins devant une caméra. Ensuite, chaque caméra effectue la détection de chute et les résultats sont combinés à l'aide de certaines techniques telles que le vote à la majorité pour prendre une décision. Une telle stratégie a été utilisée dans [31].

3.6 Conclusion

Dans ce chapitre, nous avons proposé un schéma de détection de chute pour les personnes âgées. La chute est caractérisée par une déformation de forme et un mouvement élevé. Nous avons extrait plusieurs caractéristiques pour construire deux vecteurs de caractéristiques qui représentent la forme et la variation de mouvement au cours d'un groupe d'images. Les caractéristiques de forme incluent le rapport hauteur/largeur de BB, l'orientation de l'ellipse et le rapport hauteur/largeur de l'histogramme de projection. Pour les caractéristiques de mouvement, nous avons divisé la silhouette de la personne en trois blocs y compris tête, centre et les pieds.

D'où, nous proposons d'extraire la vitesse de la tête, du centre et des pieds de trois blocs. De plus, nous avons extrait la vitesse et la direction de la personne en fonction du BB. En combinant ces deux vecteurs de caractéristiques, nous construisons notre descripteur d'activité sous forme d'un vecteur. Nous avons utilisé un classifieur SVM pour faire la distinction entre les activités de chute et non chute. La validation de la méthode proposée était basée sur trois bases de données accessibles au public. Les résultats ont montré que notre système de détection de chute est efficace pour obtenir un taux de précision élevé. Par rapport au travaux de l'état de l'art, notre méthode montre de bonnes performances même lorsque nous avons fusionné différents bases de données.

Pour une étude plus approfondie, des tests sur plus de bases de données sont prévus ainsi que la création de plus de scénarios pour tester la robustesse de la méthode proposée. En outre, nous nous concentrerons sur l'extension de la classification d'autres activités au lieu de les traiter comme une seule classe et développerons une méthode de segmentation automatique des segments. L'utilisation de plusieurs caméras peut être considérée comme une solution pour résoudre le problème d'occlusion.

PROPOSITION D'UN SYSTÈME DE DÉTECTION DE CHUTE À BASE DE LA POSTURE

Sommaire

4.1	Introduction	84
4.2	L'algorithme du système proposé	87
4.3	Structure du système proposé	88
4.3.1	Extraction de la silhouette de la personne	88
4.3.2	Extraction des caractéristiques de forme proposées	93
4.3.3	Méthode de reconnaissance de la posture de la personne	97
4.3.4	L'analyse et la détection de chute	97
4.4	Expérimentation et discussion	99
4.4.1	l'extraction de la silhouette de la personne	100
4.4.2	Classification des postures	100
4.4.3	Résultats et la performance de notre système	107
4.4.4	Discussion	109
4.5	Conclusion	111

4.1 Introduction

La détection des chutes peut être basée sur l'analyse de la posture de la personne. Le concept principal est de reconnaître certaines postures anormales de la personne, telles que se pencher, s'asseoir et se coucher, puis utiliser certaines caractéristiques pour vérifier l'occurrence de la chute. Dans [86], les auteurs ont proposé un algorithme de détection de chute dans lequel trois états différents sont identifiés : prédiction de chute, détection de chute et la vérification de chute. Dans le premier état, la posture de la personne suivie est identifiée. Ensuite, une prédiction de la posture future est opérée et ceci est comparé à des événements anormaux similaires à des chutes. L'étape de vérification vient après la confirmation de la chute. La posture de la personne est représentée par quatre caractéristiques : la zone occupée, la taille de la personne, la densité de la personne et le rapport hauteur/largeur de la BB. L'événement de chute est déclenché si la posture est classée comme couché ou assis sur le sol pendant une longue période. Cette détection doit être confirmée sur la base d'un ensemble de règles, la plus importante d'entre elles étant la capacité de la personne à récupérer de manière autonome en 1 minute.

Hung et al. [59, 99] ont proposé de combiner deux caméras orthogonales pour détecter les incidents de chute. Deux caractéristiques, la taille de la personne et la zone occupée, sont estimées et utilisées pour distinguer trois postures humaines (debout, assise et couchée). Sur la base de certains seuils bien définis, les incidents de chute peuvent être déduits de l'analyse chronologique de la transition de la pose humaine. La base de données présentés par [2] a été utilisé pour évaluer l'algorithme en se basant sur des vidéos enregistrées par les caméras 2 et 5.

Matilainen et al. [100] ont proposé une segmentation des parties du corps (BPS) pour la reconnaissance d'activités inhabituelles dans des environnements bruyants. Le BPS est utilisé pour estimer la similitude de la pose actuelle avec les poses acquises lors de la phase d'entraînement. Pour tester leur algorithme, ils ne considèrent que deux activités normales telles que marcher et s'asseoir tandis que toutes autres activités, y compris la chute, sont classées comme activité inhabituelle.

L'algorithme BPS utilise les classificateurs HMM et GMM qui sont entraînés pour chaque corps en fonction des caractéristiques de contexte de forme. Dans la solution proposée, trois séquences d'entraînement ont été utilisées pour représenter des exemples de la marche et de la chute. Par l'analyse statistique des séquences d'apprentissage, des valeurs des seuils sont conçues de manière à permettre la détection des poses inhabituelles. Ils ont également proposé d'utiliser un vote à la majorité sur un grand nombre de décisions consécutives pour deux raisons. Premièrement, les actions se produisent sur une période de temps. Deuxièmement, pour corriger l'effet d'avoir plusieurs décisions à partir de chaque image. Cependant, cette méthode n'est pas utile car elle est basée sur une base de données insuffisante des activités pour l'entraînement et la difficulté de choisir des valeurs optimales des seuils.

Une autre méthode de détection de chute basée sur des règles présentée dans [63] repose sur l'utilisation d'une fenêtre coulissante pour détecter une chute (SW-FDM) à base sur des postures humaines. Le travail proposé fait la distinction entre les postures des chutes réelles y compris la posture coucher et allonger en utilisant un seuil prédéfini. Les expériences ont montré que le SW-FDM proposé est excellent dans la détection de chute basée sur des règles, il est également plus rapide en temps de traitement et avec une faible consommation de la mémoire.

Les méthodes précédentes ne sont pas considérées comme des systèmes de détection de chute universels car les seuils doivent être adaptés à chaque changement comme la position de la caméra ou l'environnement. Afin d'éviter ces limitations, les chercheurs ont porté leur attention sur les techniques d'apprentissage automatique qui peuvent fournir un système général de détection des chutes. Dans [33, 18], les auteurs ont proposé un système de détection de chute qui utilise la méthode semi-supervisée pour détecter un événement de posture anormale et qui se base sur les règles de seuillage pour une décision finale basée sur l'analyse des mouvements. Le classificateur SVM utilise plusieurs fonctionnalités pour décrire la posture de la personne y compris l'angle et le rapport de l'axe a et de l'axe b extraits de l'ellipse, l'histogramme forme-contexte (shape-context) et la position de la personne. Le MEI est utilisée pour analyser le mouvement de la personne. La chute est détectée lorsque

la posture de la personne est classée comme couchée ou s'accroupit et que la vitesse de la personne dépasse un seuil prédéfini. Les expériences ont montré de bons résultats où le taux d'erreur ne dépasse pas 3% qui est inférieur aux résultats présentés dans [31].

Le travail de [72] présente un nouveau système de détection de chute basé sur la variation de la posture. L'idée principale est d'utiliser l'histogramme directionnel normalisé (NDH) extrait de l'ellipse de la région du corps. 12 caractéristiques locales et 8 caractéristiques globales sont extraites du NDH. Ensuite, un classifieur SVM est utilisé pour prédire dans chaque image la posture de la personne. L'accident en forme de chute est détecté en comptant les occurrences de postures couchées dans une fenêtre temporelle. L'événement de chute est déterminé sur la base d'une vérification d'inactivité de la personne après avoir procédé au vote à la majorité. Le système est évalué sur la base de données présentée dans [2] et il atteint une précision qui arrive à 97,1%. Cependant, dans les deux solutions présentées dans [72, 92], les informations de mouvement ne sont pas prises en compte, bien que cela puisse réduire le taux des fausses alarmes. Les informations de forme sont également utilisées dans [149] où les auteurs ont calculé les zones d'occupation autour du centre de gravité du corps, extrait leurs angles. Ces informations extraites sont utilisés dans divers classificateurs, parmi lesquels SVM qui donne les meilleurs résultats. Le même travail a été amélioré dans [116], où ils ont combiné le rapport de vraisemblance généralisé (GLR) et SVM. Se fonder uniquement sur des informations spatiales ne suffit pas, c'est pourquoi la combinaison des caractéristiques spatiales et des caractéristiques temporelles a été utilisée dans [89]. Deux étapes du classificateur SVM sont utilisées pour détecter une chute par rapport à une activité normale. L'avantage de cette méthode est qu'elle ne nécessite pas une technique de premier plan pour obtenir des résultats précis.

Utiliser uniquement des informations 2D peut présenter des inconvénients pour la détection des chutes. Pour cela, les auteurs de [88] exploitent la caméra Kinect pour extraire des informations 3D. La caméra Kinect fournit 25 articulations squelettiques utilisées pour représenter une activité. Le système décrit une activité en utilisant un ensemble de quelques postures de base extraites à l'aide de l'algorithme de clustering X-means. Un classificateur SVM multi-classes est utilisé pour discriminer les différentes postures. Une autre approche intéressante a été proposée dans [92] basée sur l'orientation du volume de la silhouette (SOV). SOV est un descripteur de forme qui a été défini pour représenter les actions humaines et aussi pour classer les chutes. Il offre une précision de classification importante et peut conduire à de meilleurs résultats lorsqu'il est combiné avec des outils de modélisation tels que Bag-of-Words et le classificateur BN. Le système est évalué sur la base de données présenté dans [101]. Le système proposé obtient une précision de 91,89 %, ce qui est inférieur à certains systèmes discutés précédemment, tout en étant supérieur à la précision de la méthode proposée dans [57]. D'autres travaux [115] se concentrent sur l'obtention des points clés du corps humain à l'aide des algorithmes Yolo et OpenPose. Ensuite, ils ont extrait deux types de caractéristiques appelées état descendant et état déchu. L'état de chute est principalement basée sur le changement de vitesse de la partie du corps humain. Les caractéristiques de l'état déchu sont les points clés du corps humain et l'ellipse externe. Ils ont combiné Multi-Layer Perception (MLP) et Random-Forest

(RF) pour la classification afin d'obtenir les résultats de détection de chute.

L'autre type de système de détection des chutes est basé sur les méthodes de DL. Les limites de l'apprentissage automatique traditionnel sont surmontées par l'apprentissage en profondeur qui a été appliqué dans plusieurs domaines d'application (par exemple, le traitement du langage naturel, la détection/classification d'objets, les actions humaines). Par conséquent, des travaux récents sont menés en utilisant l'apprentissage profond pour la détection des chutes. Feng P. et al. [150] utilise un DBN pour la classification de la posture humaine. Tout d'abord, ils ont extrait des silhouettes humaines d'images vidéo pour former DBN à la classification de la posture, puis ils ont adopté une méthode basée sur des règles pour la confirmation de la chute. La chute est confirmée lorsque les événements de posture couchée continuent à se produire pendant plus de 30 s. Dans [95], les auteurs ont utilisé le CNN pour la classification des postures. Les résultats obtenus par ces deux méthodes ont été conduits sur le même base de données [150], et CNN a obtenu une précision plus élevée que DBN. Le succès de l'application de l'apprentissage en profondeur dans de nombreux autres domaines pourrait être justifié par le fait que les systèmes d'apprentissage ont été menés sur des données vidéo réelles. Cependant, ce n'est pas pour le champ de détection des chutes car il existe des bases de données des chutes pour les personnes âgées qui ne sont pas accessibles au public. Les bases de données existantes ne sont que des simulations d'un jeune qui est totalement différent pour une personne âgée et malade et, par conséquent, les classifications basées sur ces bases de données sont capables de détecter incorrectement la chute d'une personne âgée dans la vie réelle. Ainsi, l'utilisation de l'apprentissage en profondeur pour la détection des chutes basée sur la vision par ordinateur est encore limitée.

Inspirés par les résultats prometteurs des méthodes discutées précédemment, nous proposons deux contributions majeures. Tout d'abord, nous proposons un nouveau descripteur de forme pour la reconnaissance de la posture humaine. La robustesse de notre descripteur de forme est évaluée sur différents ensembles de données en utilisant différents scénarios et comparaisons avec l'approche citée ci-dessus dans notre état de l'art. En exploitant le résultat de la reconnaissance de la posture, nous proposons un algorithme efficace pour la détection des chutes afin de garantir que notre reconnaissance de la posture conduit à améliorer la précision de la détection des chutes et à réduire les fausses alarmes.

4.2 L'algorithme du système proposé

Cette section présente l'approche proposée. La figure 4.1 présente l'ensemble du composant du système de détection de chute. Notre approche se compose de trois phases. La première phase consiste à extraire la silhouette humaine à partir de flux des images de la vidéo d'entrée. La deuxième phase consiste à extraire des caractéristiques locales et globales à travers la silhouette humaine. Ces caractéristiques sont combinées et utilisées pour la classification de la posture. En utilisant notre classificateur, nous faisons la distinction entre une posture normale et une posture anormale. En ce qui

concerne le post-traitement, nous fixons quatre règles pour valider si l'activité en cours peut être classée comme activité de chute ou activité normale.

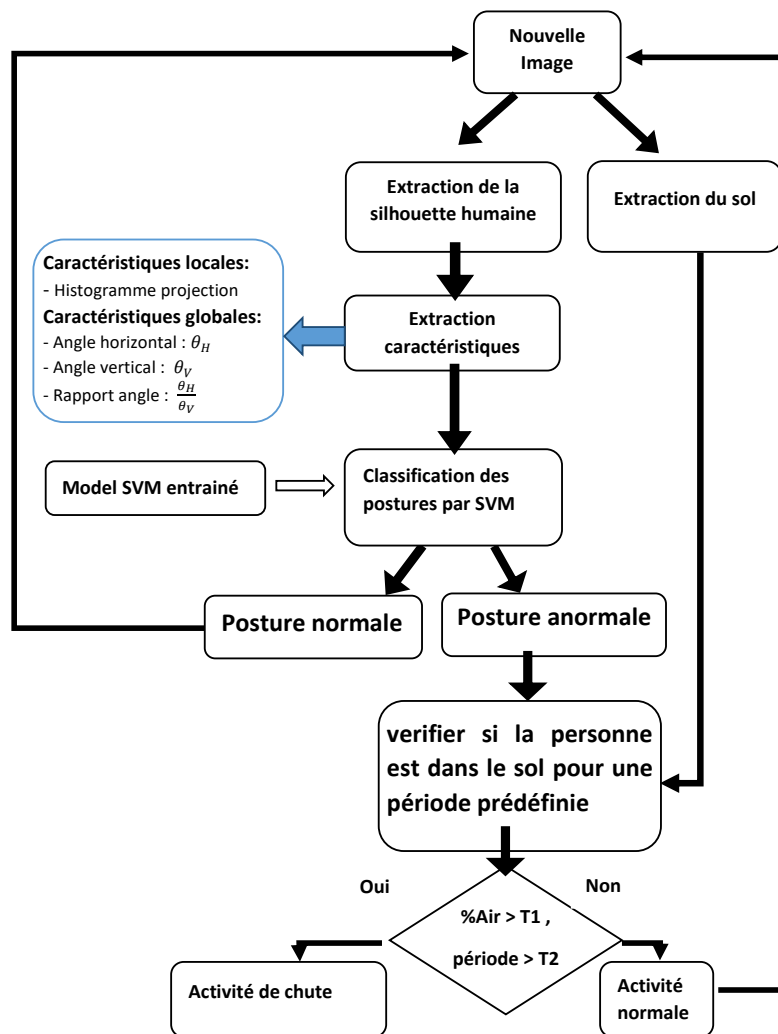


FIGURE 4.1 – Le diagramme de l'approche proposée.

4.3 Structure du système proposé

4.3.1 Extraction de la silhouette de la personne

Les systèmes de surveillance basés sur la vision par ordinateur consistent à une détection soit mobile ou statique, un suivi de la trace de l'objet dans la vidéo pour comprendre les événements qui se produisent dans la scène. Dans notre cas, nous nous intéressons à la détection et à l'extraction d'une personne en mouvement de l'arrière-plan, ce qui est la tâche la plus difficile dans la vision par ordinateur basée sur les systèmes de détection de chutes. Selon la littérature, la manière courante de distinguer les objets en mouvement de l'arrière-plan consiste à utiliser la BS.

Actuellement, de nombreux algorithmes basés sur BS sont proposés, parmi lesquels le GMM [36], AMF [37] et le modèle CB [35].

Pour détecter une personne en mouvement dans une séquence vidéo, les algorithmes doivent prendre en compte certaines difficultés. La taille de la forme du corps humain change lorsque la caméra est éloignée ou proche de l'humain. La couleur et la texture peuvent être affectées par les ombres ou lorsque le salon est à la lumière ambiante. Pour cette raison, nous considérerons la méthode CB [35] pour son avantage et sa robustesse (généralement) pour détecter et supprimer les ombres. Les résultats de la comparaison de ces trois algorithmes peuvent être trouvés dans [33].

4.3.1.1 Détection des objets mobiles

Dans un premier temps, notre tâche consiste à détecter un objet en mouvement et spécialement la personne âgée dans la séquence vidéo. La caméra utilisée est RGB et elle est installée à deux mètres de hauteur dans le salon. Le fond est généralement statique mais s'il y a des meubles en mouvement, il faut en tenir compte.

L'algorithme CB est une approche basée sur les pixels et il est composé de deux phases : phase d'apprentissage (modèle d'arrière-plan) où l'algorithme construit le CodeBook pour chaque pixel des premières images N dans une vidéo ; ensuite, ce CodeBook est utilisé à des fins de soustraction d'arrière-plan.

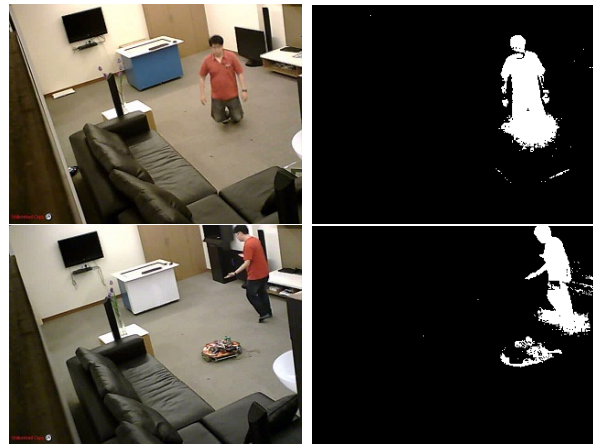


FIGURE 4.2 – La soustraction de l'arrière plan par la méthode de Codebook.

4.3.1.2 Détection de l'ombre

Généralement, le résultat obtenu par l'algorithme CB n'est pas parfait et il doit être amélioré afin d'obtenir la silhouette humaine avec plus de précision. Au niveau de la figure 4.2, nous montrons un exemple de deux problèmes ; tout d'abord, de nombreuses régions de pixels bruyants qui peuvent être produites par des ombres. Deuxièmement, le mouvement des meubles peut être détecté comme un objet en mouvement, puis l'autre région de pixels est prise comme un objet de premier plan. Ces

deux problèmes vont considérablement détériorer le résultat de l'extraction du corps. Pour résoudre ce problème, nous proposons d'ajouter une étape de post-traitement.

Au début de l'étape de post-traitement, nous nous intéressons à la détection et à la suppression des ombres du premier plan selon la méthode proposée dans [139]. Nous estimons les ombres à l'aide des informations de dégradé et de la couleur HSV, puis nous classons les pixels d'ombre en fonction de seuils prédéfinis. Pour chaque nouveau cadre, nous appliquons cette étape pour supprimer les ombres sans mettre à jour le modèle d'arrière-plan car l'ombre est un objet actif.

Pour détecter les pixels d'ombre, nous comparons d'abord les couleurs HSV de l'image courante F avec leur couleur HSV dans l'image de référence R . L'espace colorimétrique HSV a été utilisé en raison de ses avantages pour séparer la chromatique et l'intensité. Ensuite, les informations de dégradé sont utilisées pour affiner ces pixels d'ombre.

L'étape d'opération de prétraitement a été utilisée où nous avons normalisé le composant V dans l'espace colorimétrique HSV pour améliorer le contraste entre les zones d'ombre et de non-ombre. Soit $I = (H, S, V)$ l'image originale où H, S et V sont les composants de canal. Le canal V est normalisé comme suit :

$$V_n = \frac{V - \min(V)}{\max(V) - \min(V)}, \quad (4.1)$$

où V_n est le composant normalisé et V est le composant original de l'image I . En conséquence, l'image originale normalisée I obtenue est la suivante $I_n = (H, S, V_n)$.

Pour classer les pixels en pixels non-ombre / ombre, nous avons basé sur ces règles comme suit :

$$\begin{cases} \alpha \leq \left(\frac{F_p^V}{R_p^V} \right) \leq \beta, \text{ and} \\ \left(F_p^S - R_p^S \right) \leq \tau_S, \text{ and} \\ \left| F_p^H - R_p^H \right| \leq \tau_H \end{cases} \quad (4.2)$$

où (F_p^H, F_p^S, F_p^V) et (R_p^H, R_p^S, R_p^V) correspondent respectivement aux valeurs du pixel $p(i, j)$ dans l'image courante F et l'image de référence R ; β, τ_S et τ_H font référence aux seuils utilisés pour détecter les pixels d'ombre. La plage et les valeurs optimisées de ces seuils sont présentées dans le tableau 4.1.

Après avoir détecté tous les pixels d'ombre à l'aide de la couleur HSV, nous extrayons tous les Blobs de composants connectés (B) constitués par ces pixels d'ombre. La magnitude du gradient $|\nabla_p|$ et la direction du gradient θ_p sont calculées à chaque pixel d'ombre $p(i, j)$ à partir de (B) blob. Ensuite, nous extrayons uniquement les pixels de gradient significatifs supérieurs au seuil de magnitude du gradient (τ_m) ($|\nabla_p| \geq \tau_m$). Pour la direction du gradient des pixels d'ombre, nous calculons la différence entre eux dans l'image courante F et l'image de référence R comme présenté

dans l'équation 4.3.

$$\Delta\theta_p = \arccos \left[\frac{\nabla_x^F \nabla_x^R + \nabla_y^F \nabla_y^R}{\left\{ (\nabla_x^{F^2} + \nabla_y^{F^2}) (\nabla_x^{R^2} + \nabla_y^{R^2}) \right\}^{1/2}} \right] \quad (4.3)$$

Ensuite, nous estimons la corrélation de la direction du gradient entre l'image courante F et la référence R au pixel $p(i, j)$ dans le blob B par l'équation 4.4.

$$C^B = \frac{1}{N} \sum_{p=1}^N H(\tau_a - \Delta\theta_p), \quad (4.4)$$

où N est le nombre de pixels dans le blob B , $H(.)$ est une fonction qui renvoie 1 si $\tau_a - \Delta\theta_p$ est positif ou 0 sinon ; τ_a est le seuil de direction du gradient ; C^B est la direction moyenne du gradient des pixels qui sont similaires dans l'image F et référencent R . Tous les pixels dans le blob B sont détectés comme ombre si la condition $C^B \geq \tau_c$ est satisfaite, avec τ_c est le seuil de corrélation. Les valeurs des seuils τ_m, τ_a, τ_c sont présentées dans le tableau 4.1. Le résultat de la détection d'ombre de la méthode proposée est présenté dans la figure 4.3.

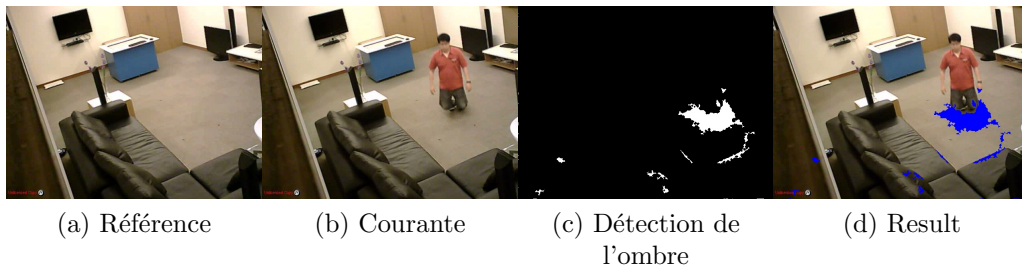


FIGURE 4.3 – La détection de l'ombre. (a) Image de référence. (b) Image Courante. (c) Les pixels de l'ombre détectés. (d) Les pixels de l'ombre en couleur blue sur l'image courante.

4.3.1.3 Extraction de la silhouette de la personne

Le premier plan extrait par la méthode CB et la détection des ombres contient de nombreux blobs qui forment des objets petits et grands. Afin d'identifier la silhouette humaine à partir de ces blobs, nous utilisons deux règles : (i) En utilisant la bibliothèque d'objets blob OpenCV [151], nous supprimons tous les blobs qui ont une petite zone (c'est-à-dire < 50 pixels), (ii) pour les gros blobs, nous les classons en plusieurs classes en utilisant la distance rectangle, chaque classe contient les blobs les plus proches en utilisant l'équation 4.5 (la distance minimale est définie empiriquement comme 50 pixels).

$$Distance(B1, B2) = \min_{P1 \in R1, P2 \in R2} distance(P1, P2), \quad (4.5)$$

TABLEAU 4.1 – La description et les valeurs des paramètres des seuils

Seuils	Description	Intervalle	Valeurs
α	seuil inférieur pour le canal V	$[0, 1)$	0,21
β	seuil supérieur pour le canal V	$[0, 1)$	0,99
τ_H	H seuil du canal H	$[0, 255]$	93
τ_S	S Seuil de canal V	$[0, 255]$	76
τ_m	Seuil de magnitude du gradient utilisé pour considérer chaque pixel $p(i, j)$ comme pixel d'ombre ou pixel de premier plan	$[0, 10]$	6
τ_a	seuil de cohérence de l'amplitude du gradient utilisé pour chaque pixel attribué au blob comme ombre ou comme premier plan	$[0, \pi]$	$\frac{\pi}{10}$
τ_c	Seuil de corrélation utilisé pour affecter chaque direction de gradient du blob comme zone d'ombre ou zone de premier plan	$[0, 1]$	0, 2

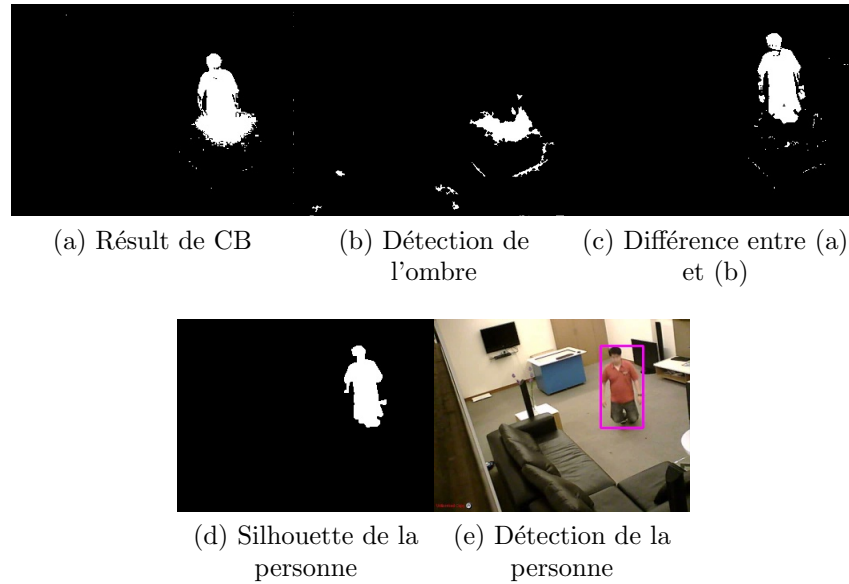


FIGURE 4.4 – Résultat d'extraction la silhouette de la personne. La dernière image illustre la personne suivi qui correspond à la silhouette de la personne encadré par un rectangle.

où $B1$, $B2$ correspondent au blob 1 et au blob 2; $P1$ et $P2$ sont respectivement les points les plus proches des rectangles $R1$ et $R2$. Si la distance calculée entre deux blobs est inférieure à 50 pixels, alors, ils sont dans la même classe, sinon, chacun est dans une classe différente.

Ensuite, nous déterminons la tache qui correspond à une silhouette humaine en utilisant le mouvement des pixels de la tâche en fonction du résultat du flux optique et de la distance entre les positions actuelle et précédente de chaque classe. Ainsi, la classe qui a une faible distance et un mouvement élevé est considérée comme la classe requise correspondant à la silhouette humaine souhaitée. Dans la figure 4.4, nous illustrons l'extraction de la silhouette humaine de l'arrière-plan en utilisant notre méthode où la dernière image montre le résultat recherché.

4.3.2 Extraction des caractéristiques de forme proposées

D'après la littérature, de nombreuses approches pour la reconnaissance de la posture humaine ont été proposées [18, 33, 150, 152, 153, 154]. Généralement, ces méthodes peuvent être divisées en méthodes basées sur des capteurs portables et des méthodes basées sur la vision par ordinateur. Pour la première catégorie, la personne doit porter sur son corps des capteurs ou une sorte de tissu qui offre plusieurs caractéristiques permettant d'identifier la posture de la personne. Ce type de capteurs consiste à porter un vêtement muni de capteurs de déformation pour reconnaître la posture du haut du corps [155], En utilisant un accéléromètre triaxial monté sur la taille du corps d'une personne afin de distinguer l'état des mouvements humains. [156].

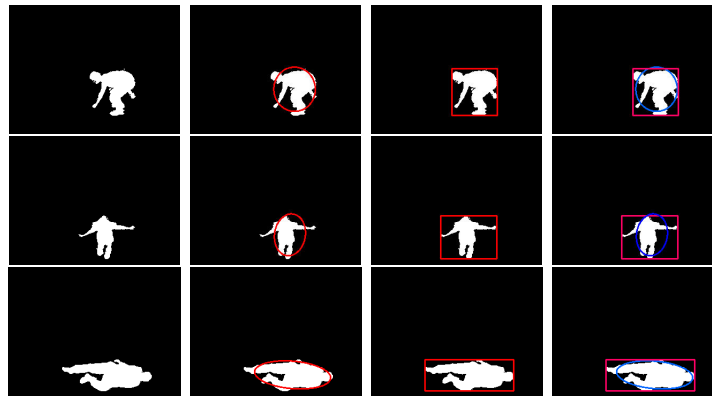


FIGURE 4.5 – La comparaison entre l'ellipse et la rectangle

Néanmoins, même s'ils présentent plusieurs avantages, ces capteurs ont besoin d'être rechargés ou de changer leur source d'alimentation périodiquement et ils doivent être portés par la personne âgée au cours de son ADL, ce qui en conclut que ces problèmes peuvent être un inconvénient pour eux.

La deuxième catégorie d'approches consiste à capturer des images du corps humain. Sur la base des techniques de traitement d'image, des caractéristiques variantes sont extraites de la forme humaine et utilisées pour faire la classification de la posture. Dans la littérature, l'enjeu central de l'analyse de forme est de décrire efficacement la forme où ses caractéristiques sont un problème fondamental. En général, nous pouvons diviser les techniques de description de forme en deux catégories. La première est des méthodes basées sur les contours [31, 157, 158, 159] qui analysent uniquement les informations sur les limites du corps humain et utilisent les techniques de mise en correspondance pour distinguer les différentes formes. Cependant, l'inconvénient de cette méthode est que les informations internes de la forme sont ignorées, ce qui peut être résolu par les méthodes basées sur la région telles que [33, 160, 161]. Ils prennent en compte toutes les informations de la forme et analysent le contenu intérieur. Ces techniques sont basées sur un histogramme de projection de la forme. Dans [33, 160], les auteurs extraient l'histogramme de la forme humaine en utilisant le contexte de forme centroïde basé sur la transformée log-polaire. Une autre technique utilise l'histogramme de projection Ellipse comme caractéristiques locales pour décrire la forme humaine [18].

Inspiré des techniques précédentes, nous proposons une nouvelle méthode de projection d'histogramme pour décrire plus correctement la forme humaine afin d'identifier la posture. L'histogramme de projection proposé est basé sur la BB, où nous divisons notre forme humaine en différentes partitions horizontalement et verticalement en utilisant plusieurs angles. L'intersection de ces partitions fournit notre histogramme de projection et il est considéré comme un descripteur de forme qui possède une forte capacité de discrimination.

Après avoir extrait la silhouette humaine de l'arrière-plan, nous nous sommes basés sur une BB adaptée à la forme humaine pour extraire notre histogramme de projection en tant que descripteur de forme humaine. Ici, la raison d'utiliser une BB

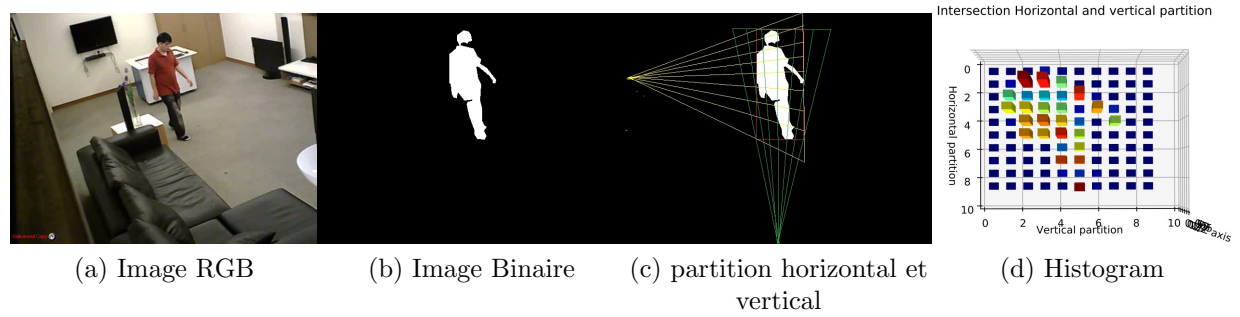


FIGURE 4.6 – Illustration de la représentation de l'historgramme proposé de la silhouette.

au lieu d'une ellipse est que l'ellipse ne prend pas en compte tous les pixels de la forme. La Figure 4.5 montre la comparaison entre ellipse et la rectangule. En fonction du centre du rectangle, de sa hauteur et de sa largeur, nous divisons la forme humaine horizontalement et verticalement comme le montre la figure 4.6. Nous présentons l'algorithme global 4.1 qui montre les différentes étapes de calcul de la partition horizontale et verticale pour extraire l'historgramme de projection en tant que descripteur de la posture humaine.

Le résultat de l'algorithme 4.1 est l'historgramme sous forme de matrice $\psi(i, l)_{N*N}$. L'historgramme est suivi d'une normalisation à l'aide de l'équation 4.6. Le but de la normalisation est de s'assurer que l'historgramme extrait est invariant en fonction de la taille humaine et de la distance de la caméra.

$$\hat{\psi}(i, l)_{N*N} = \frac{1}{M} \sum_i^N \sum_j^N \psi(i, l). \quad (4.6)$$

Algorithme 4.1 Algorithme proposé pour la projection de l'histogramme

```

1: Input : Number of partition  $N$  ; Binary shape  $S$  ; Height, Width of  $BB$  ; Reference
   Horizontal Point ( $P_h$ ) ; Reference Vertical Point ( $P_v$ )
2: Output :  $\psi(i, l)_{N*N}$ 
3: 1 : Initialization :
4:     Number of partition  $N$ 
5:      $\psi(i, l) \leftarrow 0; i, l = 1, \dots, N;$ 
6:      $P_v \leftarrow (x_v, y_v);$ 
7:      $P_h \leftarrow (x_h, y_h);$ 
8: 2 : Compute  $\theta_H, \theta_V$  :
9:      $D_H = distance(P_h, BB);$ 
10:     $D_V = distance(P_v, BB);$ 
11:     $\theta_H = 2 * \arctan(\frac{Height * D_H}{2});$ 
12:     $\theta_V = 2 * \arctan(\frac{Width * D_V}{2});$ 
13:     $\Delta\theta_H = \frac{\theta_H}{N};$ 
14:     $\Delta\theta_V = \frac{\theta_V}{N};$ 
15: 3 : Let  $A \leftarrow \{(x_t, y_t), t = 1, \dots, M\}$  be a Set of pixels in shape  $S$  and  $M$  is the
   total points.
16: pour  $t = 1, \dots, M$  faire
17:     if  $x_h - x_t \geq 0$  then
18:          $(\theta_H)_t = \arctan(\frac{x_h - x_t}{y_t - y_h});$ 
19:          $i = Round\left(\frac{(\theta_H)_t - \frac{\theta_H}{2}}{\Delta\theta_H}\right);$ 
20:     else
21:          $(\theta_H)_t = \arctan(\frac{x_t - x_h}{y_t - y_h});$ 
22:          $i = Round\left(\frac{(\theta_H)_t + \frac{\theta_H}{2}}{\Delta\theta_H}\right);$ 
23:     endif
24:     if  $y_v - y_t \geq 0$  then
25:          $(\theta_V)_t = \arctan(\frac{y_v - y_t}{x_t - x_v});$ 
26:          $l = Round\left(\frac{(\theta_V)_t - \frac{\theta_V}{2}}{\Delta\theta_V}\right);$ 
27:     else
28:          $(\theta_V)_t = \arctan(\frac{y_v - y_t}{x_t - x_v});$ 
29:          $l = Round\left(\frac{(\theta_V)_t + \frac{\theta_V}{2}}{\Delta\theta_V}\right);$ 
30:     endif
31:      $\psi(i, l) \leftarrow \psi(i, l) + 1;$ 
32: fin pour
33: Return  $\psi(i, l);$ 

```

Pour l'étude expérimentale, le nombre de partitions sera fixé à $N = 10$. En tenant compte de $C_{BB}(x_c, y_c)$ comme centre de la BB et H comme hauteur de l'image, le point vertical de référence (P_h) et le point horizontal de référence (P_v) est fixé à $(x_h = x_c, y_h = 0)$ et $(x_v = H, y_v = y_c)$.

Il ne suffit pas de ne considérer que les caractéristiques locales pour décrire toutes les postures humaines, où certaines postures similaires sont très difficiles à différencier entre elles, ce qui conduit le classifieur à être confus. Pour cela, nous ajoutons des caractéristiques globales telles que les angles horizontaux et verticaux (θ_H, θ_V) et le rapport entre eux ($\frac{\theta_H}{\theta_V}$). Ensuite, nous combinons les caractéristiques locales et les caractéristiques globales en tant qu'une caractéristique vectorielle entière pour la classification. Les résultats de classification de ces deux types de fonctionnalités seront discutés expérimentalement dans la section 4.4.2.2.

La dimension finale du vecteur de caractéristiques est $N * N + 3$, où $N * N$ représente la taille du vecteur des caractéristiques locales et 3 représente les trois caractéristiques globales. Le descripteur global proposé de la forme (F_S) est défini par l'équation 4.7 .

$$F_S = [\hat{\psi}(i, l)_{N*N}, \theta_V, \theta_H, \frac{\theta_H}{\theta_V}]. \quad (4.7)$$

4.3.3 Méthode de reconnaissance de la posture de la personne

Pour l'identification de la posture d'une personne, nous nous basons sur une méthode d'apprentissage via un classifieur. Le classificateur va attribuer notre caractéristique vectorielle F_S (voir l'équation 4.7) à l'une des quatre catégories de posture y compris position allongée, position debout, position s'asseoir et position s'accroupir. Ces quatre positions sont considérées parce qu'ils ont la relation avec la chute. Le classificateur est utilisé pour trouver une fonction qui associe chaque vecteur de caractéristiques de la posture et l'étiquette correspondant $Y_k = \{1, 2, 3, 4\}$, où 1, 2, 3 et 4 représentent respectivement la position s'accroupir, allongée, assise et debout. Comme nous avons plusieurs étiquettes, nous adoptons un classificateur multi-classe pour la classification des postures au lieu d'un classificateur binaire. L'ensemble de l'opération est présenté dans la figure 4.7.

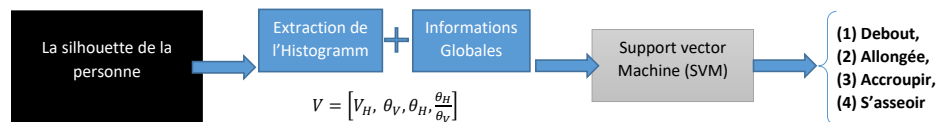


FIGURE 4.7 – La reconnaissance de la classe de la posture avec SVM.

4.3.4 L'analyse et la détection de chute

Après l'étape de classification de la posture, nous vérifions l'existence de la chute lorsque la sortie du classificateur est une posture anormale (couchée ou s'accroupir).

Pour cela, nous utilisons l'algorithme 4.2 qui est composé de quatre règles. Dans la première règle, nous vérifions si la posture est classée comme "coucher" ou "accroupir". Ensuite, nous vérifions si la posture est à l'intérieur du sol, ce qui signifie si le pourcentage de surface de la silhouette humaine est supérieur à un seuil défini. La valeur du seuil est fixée à 85 % et elle est fondée expérimentalement. Pour la transition de posture, comme la plupart des activités de chute commencent par la position debout et se terminent par la posture allongée ou commencent par la posture assise et se terminent par la posture allongée. Le temps qui passe pour une chute est en moyenne de 20 images trouvé à base de nos expérimentations sur la base de données adoptée. Enfin, si ces conditions ci-dessus sont maintenues à un certain moment sans mouvement, qui dépasse le seuil défini (nous utilisons 25 images), nous confirmons la détection de chute et un signal d'alarme d'aide est déclenché.

L'information du sol est crucial pour la confirmation de la chute, car l'activité de chute se termine toujours par une posture allongée sur le sol. De nombreux travaux antérieurs incorporaient les informations du plan au sol et donnaient de bons résultats [18, 45, 162].

En raison d'utilisation uniquement d'une seule caméra RGB pour la détection des chutes au lieu d'utiliser des caméras de profondeur, ainsi les bases de données ont été réalisées avec différentes textures de sol. Nous avons proposé d'utiliser uniquement la segmentation manuelle pour extraire les régions du sol afin d'évaluer notre algorithme de détection de chute au lieu d'utiliser des méthodes automatiques. Parmi les méthodes automatiques existantes dans la littérature, nous citons les méthodes supervisées telles que présentées dans [163]. Nous avons appliqué cette méthode uniquement pour la première image de toute séquence vidéo pour extraire les pixels du sol manuellement.

Pour faire la distinction entre des activités similaires comme chuter ou se coucher, nous avons basé sur la transition temporelle de la position debout et la posture allongée comme le montre la figure 4.8. À partir de cette figure, nous pouvons voir que l'activité de se coucher prend plus de 80 images (par exemple, plus de 3s). Cependant, pour l'activité de chute, le nombre d'image pour la transition de posture est inférieur à 25 images ce qui signifie presque 1 seconde, et ceci est normal car la chute a un mouvement non contrôlé et se coucher a un mouvement contrôlé. Pour effectuer cette transition temporelle, si la posture de la personne est classée comme accroupir ou coucher et que la majeure partie de sa région corporelle est à l'intérieur du sol, nous comptons le nombre de d'image entre le l'image actuel et l'image précédent où la personne est classée comme debout ou s'asseoir. Si le nombre des images est inférieur à 25 images, alors, nous retournons vrai car cette activité peut être considérée comme une chute si la personne reste inactive pendant un certain temps.

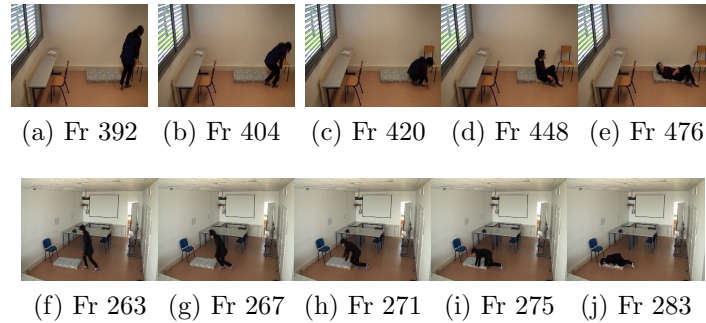


FIGURE 4.8 – Illustration des images de transition de posture de l'activité de chute et de l'activité couchée normale. Les images (a), (b), (c), (d) et (e) présentent une séquence de l'activité couchée. Les images (f), (g), (h), (i) et (j) présentent une séquence de l'activité de chute.

Algorithme 4.2 La stratégie de la détection de chute

- 1: **Input** : Human Posture, Body area, inactivity time threshold T
 - 2: **Output** : Fall or no-Fall
 - 3: **Repeat** :
 - 4: CheckAbnormalPosture()
 - 5: CheckInsideFloor(Area)
 - 6: TransitionPosture()
 - 7: CheckInactivityTime (T)
 - 8: **if** the conditions 4, 5, 6 and 7 are True **then** return Fall
 - 9: **Else** go to step 3
-

4.4 Expérimentation et discussion

Cette section montre les performances du système de détection des chutes proposé. L'architecture a été implémentée à l'aide du langage C++ avec l'IDE Microsoft Visual Studio Express 2012 et de la bibliothèque OpenCV 2.4.13 pour la soustraction d'arrière-plan, l'extraction de fonctionnalités et l'étape de classification. Les expériences (entraînement, tests) ont été réalisées sur un ordinateur portable avec processeur Intel (R) Core (TM) i7-6700HQ et 2,60 Hz et 12,00 Go de RAM. Avec intensivement, nous menons sur une base de données différent où nous les divisons en deux catégories; le jeu de données pour la reconnaissance de la posture et le jeu de données des vidéos de chute. Plus de détails sont présentés dans les sections suivantes.

4.4.1 l'extraction de la silhouette de la personne

Dans cette partie, nous montrons les performances de l'algorithme de soustraction d'arrière-plan utilisé. Quelques résultats significatifs sont illustrés dans la figure 4.9. Les inconvénients de toute méthode basée sur la soustraction de l'arrière-plan sont le changement d'éclairage et l'ombre. Pour cela, nous avons traité séparément la détection des ombres et la soustraction d'arrière-plan. En utilisant le résultat de la détection d'ombre, nous le soustrayons du résultat de l'algorithme CB, suivi d'une opération morphologique pour supprimer les petites régions inutiles. La dernière colonne (e) de la figure 4.9 représente le résultat de notre extraction de la silhouette humaine.

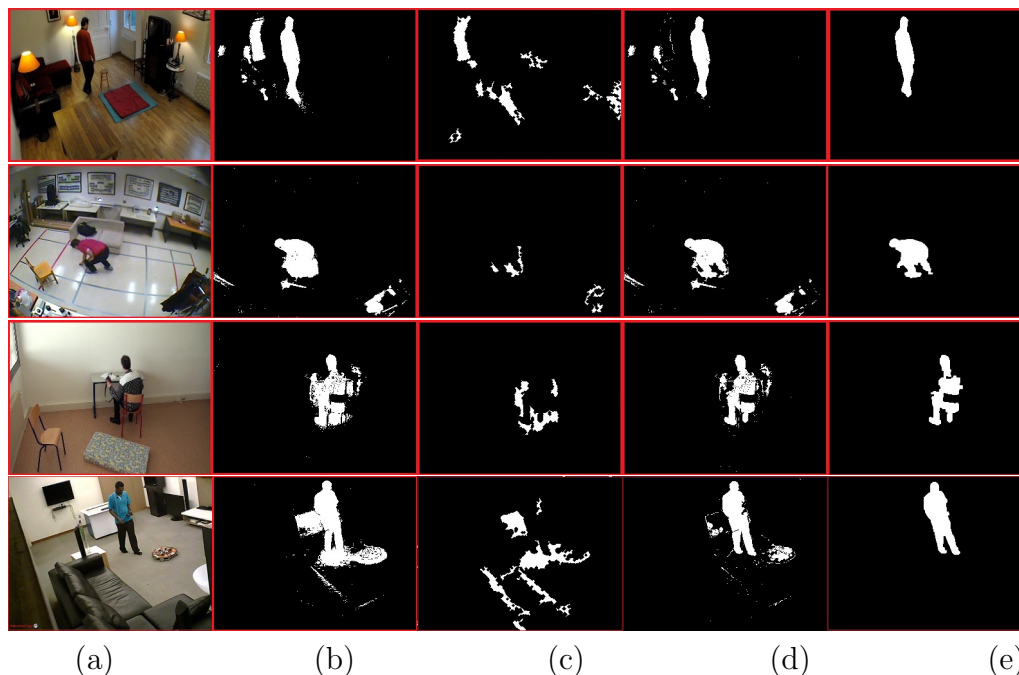


FIGURE 4.9 – Résultats de la soustraction de l'arrière-plan. (a) Image d'origine, (b) Résultat de l'algorithme de CB, (c) Détection d'ombre, (d) La soustraction pixel par pixel entre l'image (b) et l'image (c). (e) Résultat final.

Comme nous pouvons le voir sur la figure 4.9, la condition de la lumière est différente dans toutes les images originales (a). La personne peut marcher près d'un objet comme indiqué dans la dernière ligne et la personne continue de faire ses activités. Malgré ces conditions, la silhouette de la personne est correctement extraite de l'arrière-plan.

4.4.2 Classification des postures

Nous effectuons notre classification de posture basée sur la bibliothèque LIBSVM en utilisant le classifieur SVM [145]. Nous avons choisi d'utiliser le classifieur multi-classe C-SVM avec un noyau de fonction de base radiale (RBF). Tous Les paramètres

sont gardés par défaut sauf le paramètre gamma (g) de RBF et le coût (c) de C-SVM qui ont été modifiés respectivement à 0.01 et 100. La performance de notre méthode est menée sur les bases de données décrites dans la section 4.4.2.1. Nous avons réalisé quatre expérimentations principales notamment pour l'évaluation des caractéristiques, l'évaluation du nombre de partitions, la comparaison des méthodes d'extraction des caractéristiques de l'état de l'art et la comparaison entre les classificateurs.

4.4.2.1 La base de données des postures

Pour évaluer notre méthode de reconnaissance de la posture et la comparer avec d'autres méthodes, nous utilisons deux bases de données de posture qui sont composés des silhouettes des personnes 2D, y compris la posture coucher, debout, assise et accroupir.

Base de données (D1)[150] : Cette base de données a été réalisée en utilisant une seule caméra RGB. 10 personnes ont été invitées à participer en tant que volontaires aux expériences de simulation. On a demandé à chaque personne de simuler des postures dans différentes directions afin que le classificateur construit soit robuste aux différents angles de vue. Les postures de chaque personne i sont stockées dans un dossier nommé P_i avec $i = 1, \dots, 10$. La base de données contient 3216 images dont 810 postures de coucher, 804 postures de debout, 833 postures de se baisser et 769 postures de s'asseoir. Certaines images clés sont illustrées dans la figure 4.10.



FIGURE 4.10 – Exemples des postures de la base de données D1. Les colonnes 1, 2, 3 et 4 correspondent aux différents postures de debout, s'asseoir, allongée, et accroupir respectivement.

Base de données (D2) : En utilisant la base de données [4, 3], nous générons notre jeu de données de posture en utilisant l'algorithme de soustraction d'arrière-plan présenté auparavant dans la section 4.4.1 pour extraire la silhouette de la personne à partir des vidéos. Cette base de données construite est composée de 2865 images dont 444 postures de s'asseoir, 1394 postures de coucher, 453 postures de s'accroupir et 576 postures de debout. La figure 4.11 montre quelques images clés de cette base de données.

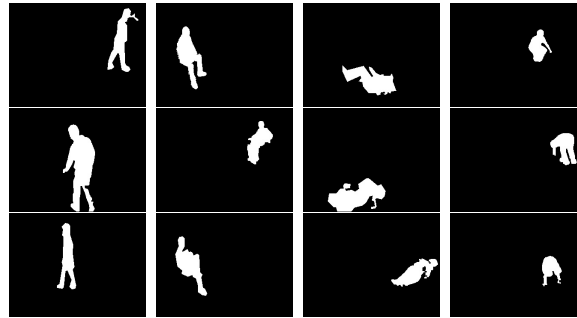


FIGURE 4.11 – Échantillons de posture de la base de données D2. Les colonnes 1 à 4 correspondent respectivement aux postures debout, assise, coucher et accroupir.

Dans le tableau 4.2, nous montrons la description de certaines caractéristiques et difficultés des deux bases de données.

TABLEAU 4.2 – Caractéristiques de chaque base de données

Base de données	Genre de la personne	Endroit	Dimension de forme de la personne	Angle de vue	Ambiguïté des postures
D1	Homme, Femme	Une seule chambre	Varié	Plusieurs	Oui
D2	Homme, Femme	5 salles différents	Varié	Plusieurs	Oui

4.4.2.2 Evaluation des caractéristiques Locales et Globales

Dans cette section, nous comparons les caractéristiques globales et locales pour la classification. La validation croisée 10 fois commune [164] a été utilisée pour évaluer les fonctionnalités.

Comme indiqué dans le Tableau 4.3, nous comparons le résultat de classification obtenu lorsque nous utilisons les caractéristiques locale ou globale seule et lorsque nous utilisons la combinaison entre elles. Le SVM est utilisé pour la classification et à partir de ce tableau, nous pouvons voir que la combinaison des caractéristiques offre une précision élevée par rapport à l'utilisation de l'une ou l'autre caractéristique seule.

Les Tableaux 4.4 et 4.5 illustrent les mesures de performance de la méthode proposée appliquée aux bases de données D1 et D2 à l'aide des caractéristiques combinées. La précision, le rappel et le F-Score sont les métriques d'évaluation sélectionnés pour la classification de la posture qui sont calculés à base de la matrice de confusion. A partir de ces tableaux, toutes les postures sont bien classées avec des valeurs élevées pour toutes les métriques.

TABLEAU 4.3 – Résultat de la comparaison des caractéristiques locales et globales pour les deux bases de données D1 et D2

Base de données		Globales	Locales	Combinées
D1	Exactitude	90.06%	97.00%	99.90%
	False Dtection	6.5433%	0.0625%	0.0401 %
D2	Exactitude	89.63%	98%	99.84%
	False Dtection	0.6303%	0.10%	0.0746%

TABLEAU 4.4 – Classification des postures pour la base de données D1

Postures	Précision	Sensibilité	F1-Score	#train	#test
Accroupir	99%	100%	99.49%	566	267
Allongée	100%	99%	99.49%	591	219
S’asseoir	100%	100%	100%	539	230
Debout	100%	100%	100%	555	249

TABLEAU 4.5 – Classification des postures pour la base de données D2

Postures	Précision	Sensibilité	F1-Score	#train	#test
Accroupir	100%	99%	99.49%	309	137
Allongée	100%	100%	100%	978	406
S’asseoir	99%	100%	99.49%	320	118
Debout	100%	99%	99.49%	374	188

4.4.2.3 Évaluation du descripteur de la posture proposée

Le nombre de partitions qui peuvent être efficaces pour la classification des postures est difficile à définir comme le montre la figure 4.12.

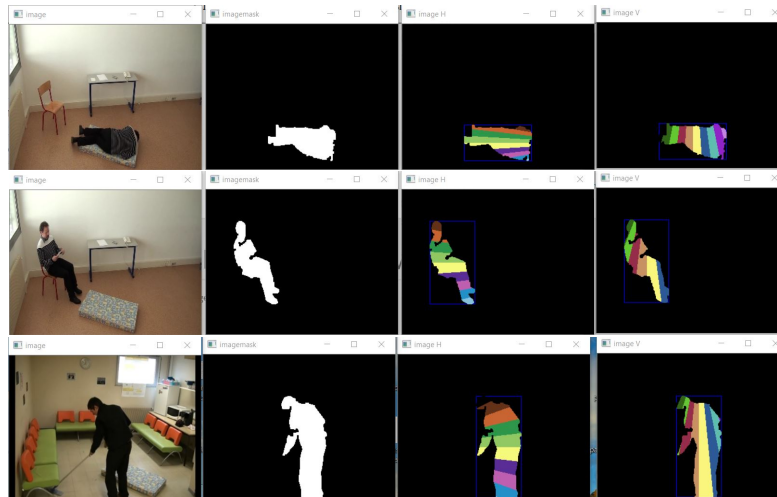


FIGURE 4.12 – Illustration le nombre de partition horizontales et verticales de la silhouette de la personne.

Par conséquent, nous avons défini plusieurs nombres de partitions et nous avons testé chacune d’elles en utilisant la base de données D1 comme indiqué dans le Tableau 4.6. Nous avons utilisé 5 partitions (bins), 10, 15 et 20 partitions. Comme nous pouvons le voir sur ce Tableau 4.6, les 10 partitions donnent une précision élevée par rapport aux autres dans la plupart des cas. Ces résultats peuvent se justifier que si le nombre de partitions est petit, on perd beaucoup d’informations où on a un grand nombre d’instances (pixels) dans chaque bin qui pourraient être similaires entre les postures. Pour un grand nombre de partitions, nous avons plusieurs partitions (bins) avec des instances vides ou avec des instances égales entre différentes postures, ce qui conduit à des difficultés pour les distinguer. Comme résultats, nous avons considéré 10 partitions comme valeur par défaut de notre système dans toutes les expériences. La figure 4.13 montre un exemple de quatre histogrammes de projection avec 10 partitions pour les postures debout, s’asseoir, allongée et accroupir.

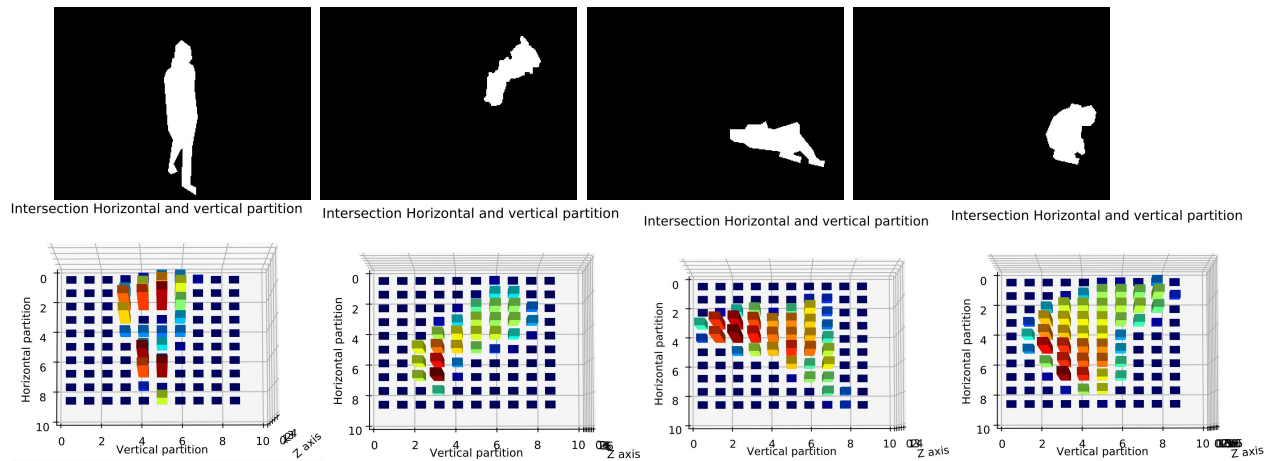


FIGURE 4.13 – Illustration de la posture du corps humain et histogramme de projection correspond. La première ligne présente les postures de la silhouette de la personne qui sont des postures debout, assise, couchée et accroupir. La deuxième ligne indique les histogrammes de projections correspondants.

TABLEAU 4.6 – Évaluation de nombre de partition pour l'extraction des caractéristiques

Le nombre de partition	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10
5	94.68	90.71	97.81	91.93	89.37	90.85	89.11	97.53	91.86	90.12
10	95.62	96.90	99.06	96.77	89.37	94.39	89.70	96.92	98.30	92.59
15	95.62	96.90	99.06	95.16	91.56	92.62	92.35	96.61	97.96	92.59
20	95.62	94.11	99.06	95.16	90.62	89.97	92.64	96.61	95.25	93.20

4.4.2.4 Comparaison des méthodes d'extraction des caractéristiques

Pour montrer les performances de notre méthode d'extraction de caractéristiques pour la reconnaissance de la posture. Nous comparons notre méthode avec les méthodes CNN [95], le descripteur d'ellipse [18] et le contexte de la forme [165] en utilisant la Base de données D1 et D2. Tableau 4.7 fait référence aux résultats obtenus en utilisant la base de données D1 et Tableau 4.8 fait référence aux résultats obtenus en utilisant la base de données D2.

TABLEAU 4.7 – Comparaison de différentes méthodes d'extraction des caractéristiques pour la base de données D1

Méthode	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10
CNN [95]	96.88%	97.83%	96.56%	92.58%	95.31%	92.04%	93.24%	96.62%	93.56%	94.75%
Shape Context [165]	92.18%	81.11%	88.12%	89.35%	85.93%	78.17%	79.41%	88.00%	83.72%	88.27%
Ellipse descriptor [18]	74.03%	67.93%	66.34%	71.85%	66.66%	66.46%	73.49%	72.38%	63.76%	71.65%
Notre méthode	95.62%	96.90%	99.06%	96.77%	89.37%	94.40%	89.70%	96.92%	98.30%	92.59%

Dans le Tableau 4.7, nous avons utilisé les postures de chaque individu particulier (P_i) pour les tests, les autres postures sont utilisées pour l'entraînement. Notre

TABLEAU 4.8 – Comparaison de différentes méthodes d’extraction de caractéristiques pour la base de données D2

Méthode	Exactitude	Taux d’erreur de détection
Shape Context [165]	98.37%	2.0%
Ellipse descriptor [18]	97.05%	3.5%
Notre Méthode	99.78%	0.5%

TABLEAU 4.9 – Comparaison des méthodes d’extraction des caractéristiques

Méthode	Exactitude(%)
Ellipse descriptor[18]	96.1
Silhouette area [166]	95.2
Bounding Box ratio [167]	82.2
Normalized directional histogram [72]	97.1
Notre Méthode	99.90

méthode présente des avantages significatifs par rapport aux autres méthodes où elle surpasse les méthodes de descripteur d’ellipse et de contexte de forme pour la majorité des cas individuels avec une plus grande précision. Par rapport à la méthode CNN, même si le deep learning a des performances élevées par rapport aux méthodes standard, notre méthode donne des bons résultats où elle le dépasse dans plusieurs cas.

Dans le Tableau 4.8, nous illustrons le résultat de la comparaison obtenu à l’aide de la base de données D2 où nous choisissons 70% de données pour l’entraînement et 30s% pour les tests. La base de données D2 contient 1865 postures. Les résultats montrent que notre méthode atteint une précision élevée par rapport aux autres méthodes et avec un faible taux de fausse détection.

Une autre comparaison est effectuée comme indiqué dans le Tableau 4.9. Nous comparons les performances obtenues par notre approche et certaines approches existantes d’extraction de caractéristiques discutées dans l’état de l’art en utilisant le même jeu de données. A partir de ce Tableau, l’approche proposée présente des résultats favorables par rapport à ceux obtenus par le descripteur d’ellipse. L’histogramme de projection [18] est basé uniquement sur une ellipse qui ne prend pas en compte tous les pixels de la silhouette de la personne. Pour la méthode de zone de silhouette [166], les variations de la zone de la silhouette de la personne sont invariantes à la vue, mais elles dépendent largement de la stratégie de mise à jour en arrière-plan. La méthode du ratio de BB [167] est basée uniquement sur des fonctionnalités globales et elle est très facile à mettre en œuvre. L’histogramme directionnel normalisé [72] est utilisé pour dériver des caractéristiques statiques et dynamiques basées sur l’ellipse.

4.4.2.5 Evaluation des classifieurs pour la reconnaissance des postures

Dans cette section, nous comparons les performances de la méthode proposée avec les résultats de certains algorithmes d'apprentissage automatique fréquemment utilisés, notamment Linear SVM (L-SVM), RF, Arbre de décision (Decision Tree) (DT), KNN, NN et SVM-RBF. La figure 4.14 présente les résultats de la matrice de confusion en utilisant le jeu de données D1. À partir de cette figure, nous pouvons voir que le classificateur SVM-RBF donne la plus grande précision par rapport aux autres classifieurs pour tous les types de postures.

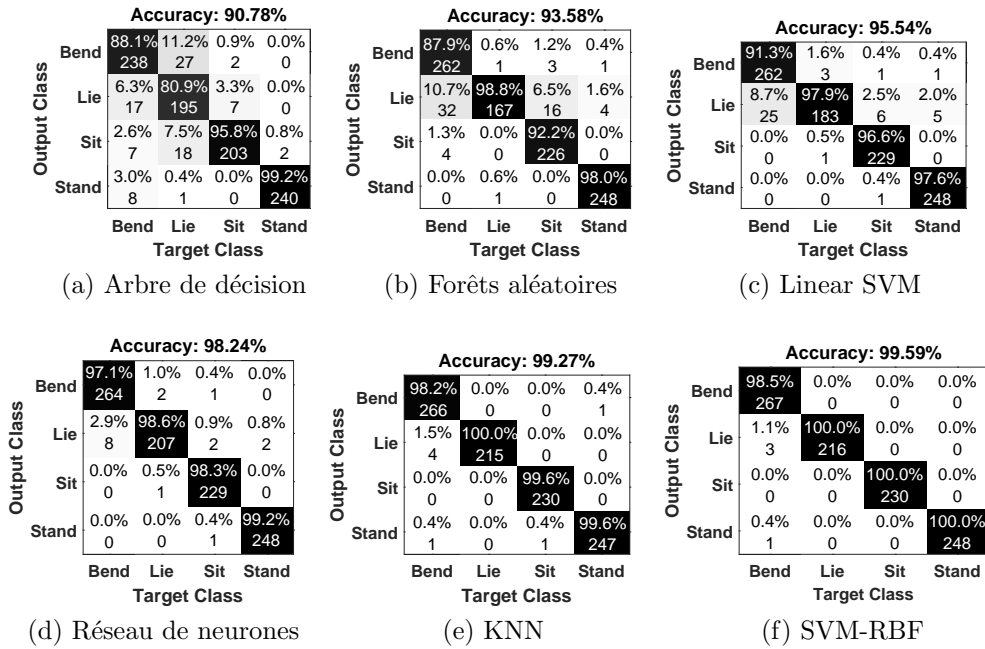


FIGURE 4.14 – La matrice de confusion de la classification de posture. Les images (a), (b), (c), (d), (e) et (f) présentent les résultats obtenus avec les classifieurs DT, RF, SVM, L-SVM, NN, KNN et SVM-RBF respectivement.

4.4.3 Résultats et la performance de notre système

Pour évaluer la performance de notre approche pour la reconnaissance de la chute dans la vidéo, nous avons utilisé trois bases de données différentes D3 (chua), D4(charfi), D5(rougier).

4.4.3.1 Détection de chute de notre système

Pour la détection des chutes, selon l'algorithme 4.2 de la section 4.3.4, la classification de la posture ainsi que les informations du sol de détection sont utilisées pour détecter les chutes. Dans la figure 4.15, nous montrons quelques résultats de classification de la posture humaine avec des informations sur le sol. L'image (a)

montre une personne qui est tombée sur le sol, la posture "couchée" est détectée et la plus grande partie du corps est à l'intérieur de la région du sol. Si cette posture est le résultat d'une activité qui est commencée en position debout ou assise et le nombre d'image de cette transition est inférieur à 25 images (durée de chute = 11 images). Alors la chute est détectée si cette posture est maintenue en inactivité pendant un certain temps dépassant un seuil défini de 1 seconde. La posture de s'accroupir est illustrée à la figure (b), où la région du corps n'est pas complètement à l'intérieur de la région du sol, ce qui confirme qu'il s'agit d'une activité normale. Un autre cas est illustré dans le l'image (c), la posture est classée comme accroupir et la transition temporelle de posture est inférieure au seuil, tandis que cette posture maintenue immobile pendant plus que le seuil, le système classe cette posture comme activité de chute. La posture dans l'image (d) est classée comme une posture assise, donc bien que cette posture soit considérée comme une activité normale, notre système continue à fonctionner normalement jusqu'à ce qu'il détecte certaines postures anormales. L'image (e) montre une personne qui se tient debout/marche sur le sol et le système la détecte comme une activité normale.

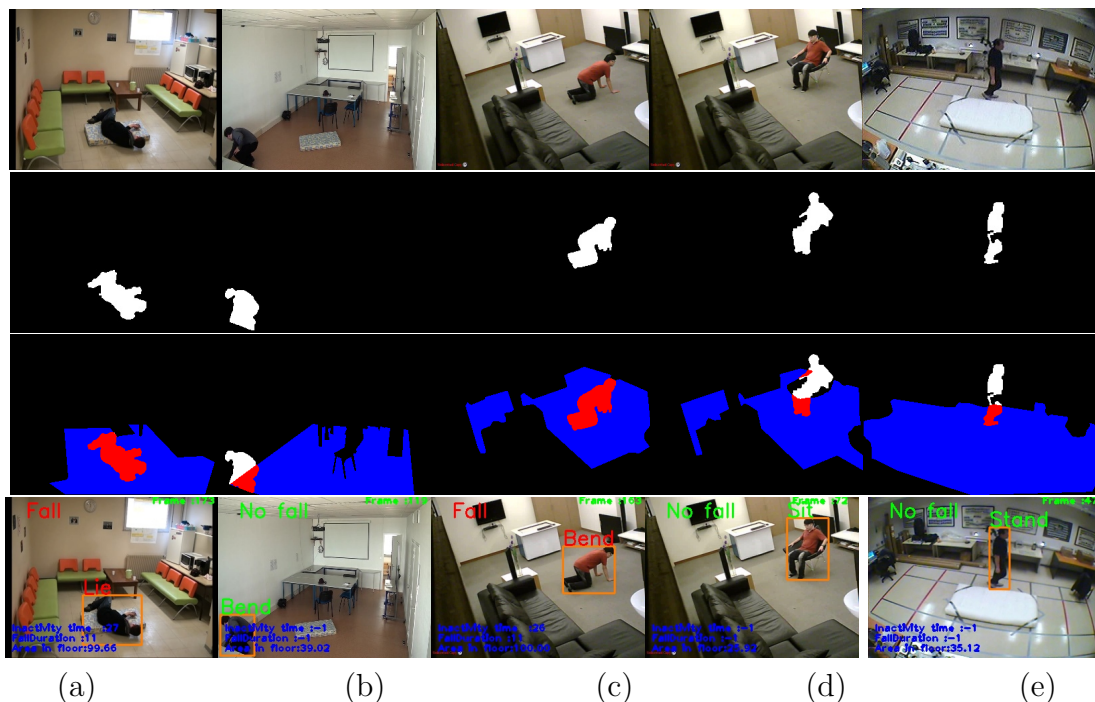


FIGURE 4.15 – Illustration des activités de chute, accroupir, s'asseoir et debout dans les colonnes 1,2,3 et 4. La dernière ligne présente le résultat de notre système pour la détection de chute.

4.4.3.2 Comparaison avec l'état de l'art

Pour évaluer notre système de détection des chutes, nous collectons plusieurs vidéos de chute à partir des bases de données D3 - D5. Certaines vidéos de ces bases de données ne sont pas prises en compte pour trois raisons pour lesquelles notre méthode de BS ne peut pas être appliquée :

- Certaines vidéos sont très courtes et il n’y a pas de temps supplémentaire pour la confirmation de chute.
- La première image dans les vidéos contient la personne ce qui est confondu avec notre algorithme de BS basé sur les 50 premières images pour construire le modèle d’arrière-plan, et aussi avec l’extraction du sol.
- La personne se trouve derrière un objet lorsqu’elle tombe où sa forme n’est pas visible par la caméra.

Comme montre le tableau 4.10, le nombre total d’activités de chute est de 159 et de 136 pour les activités non chute. Afin d’utiliser la classification de la posture de la personne pour la détection des chutes, les postures de la base de données D1 et de la base de données D2 sont combinées et utilisées pour entraîner le classifieur SVM. À partir de ce tableau, nous pouvons voir que notre système peut détecter 155 chutes sur 159 (97,48 % précision) activités de chute, tandis que pour les activités non chutes, seulement quatre activités sont détectées comme chutes sur 136 (2,94 % taux d’erreur); ces erreurs peuvent être justifiées par la raison pour laquelle la personne accroupi au sol pendant un certain temps sans mouvement qui dépasse les seuils, le système le détecte comme une chute.

TABLEAU 4.10 – Résultats de notre système de détection de chute proposé

	#Activités	Détecté comme chute	Détecté comme non-chute
Chuter	159	155	4
Marcher/Debout	40	0	40
S’asseoir	51	0	51
Accroupir	20	2	18
Coucher	25	2	23

Dans le Tableau 4.11, nous montrons le résultat des expériences de la méthode proposée et d’autres méthodes de l’état de l’art. Comme nous pouvons le voir, notre méthode atteint une précision élevée par rapport aux autres méthodes. La méthode proposée obtient moins de valeur de rappel par rapport aux méthodes [89, 116], alors qu’elle est plus précise en terme de détection de chute. Cela signifie que notre méthode est excellente pour la détection des chutes, alors qu’elle doit encore être améliorée pour réduire les erreurs de classification des activités normales. Par rapport à la méthode [115], notre méthode est meilleure pour réduire les fausses alarmes.

De ce fait, les résultats de l’expérience obtenus sont acceptables et prouvent l’efficacité de notre méthode de reconnaissance des chutes.

4.4.4 Discussion

L’approche proposée utilise la reconnaissance de la posture pour un système de détection de chute. Le système proposé est convenable où les personnes âgées n’ont pas besoin de porter de capteurs sur leur corps qui sont généralement affectés par le bruit de fond dans l’environnement. Le descripteur proposé pour la classification de la posture fournit un taux de classification élevé par rapport aux autres méthodes de

TABLEAU 4.11 – Comparaison avec l'état de l'arts

Méthodes	Exactitude(%)	Précision	Sensibilité
Chua et al. [3]	93.30	-	-
Rhuma et al. [18]	96.09	-	-
Harrou et al. [116]	96.66	94.00	100
Wang et al. [115]	96.91	97.64	96.51
Jeffin et al. [89]	97.14	93.75	100
Notre Méthode	97.29	97.48	97.48

reconnaissance de la posture. Avec un vaste base de données, comprenant différentes postures humaines capturées avec différentes caméras, le classificateur peut distinguer efficacement différents types de postures. La procédure est totalement automatique et il n'est pas nécessaire de définir des seuils pour les distinguer. C'est rapide et efficace lorsque nous nous basons uniquement sur une seule image pour extraire et effectuer la classification de posture au lieu d'utiliser un ensemble d'images comme méthodes basées sur les clips vidéo [112, 55]. De plus, nous n'utilisons que des classificateurs traditionnels au lieu d'utiliser des méthodes d'apprentissage en profondeur telles que [150, 95] qui sont riches en temps de calcul. En comparaison avec les méthodes de l'art, pour la classification des postures, notre approche proposée présente les caractéristiques suivantes, qui sont les nouveautés de notre travail.

Le principal composant de la détection des chutes étant basé sur la classification de la posture humaine, l'extraction humaine de l'arrière-plan doit être robuste et efficace. Pour cela, nous utilisons une soustraction d'arrière-plan efficace combinée à une détection d'ombre afin de faire face aux changements d'arrière-plan tels que les problèmes de lumière et d'ombre. Dans chaque image, les opérations de blob sont utilisées pour sélectionner les objets non requis qui sont classés comme meubles ou fantômes et les placer dans le modèle d'arrière-plan. Par conséquent, dans l'image suivante, ils disparaîtront du premier plan.

Un nouveau descripteur de silhouette de la personne est utilisé pour la classification de la posture qui est plus robuste que l'histogramme de projection basé sur Ellipse ou la méthode à base deCNN. La confirmation de chute nécessitait des informations sur le sol afin de faire la distinction entre un humain allongé sur le canapé et une chute humaine. La plupart des détections d'erreurs dans les systèmes précédents font référence à ce problème. Ainsi, comme solution, la confirmation à l'aide des informations du sol réduit la détection d'erreur. Cependant, certains problèmes se produisent encore dans notre système comme dans [3, 33, 72]. Notre système est conçu pour surveiller une personne seule vivant seule à la maison, ce qui n'est pas adéquat dans certains cas particuliers. De tels cas incluent la présence de plusieurs personnes dans la maison et lorsque les personnes âgées ont un animal domestique de grande taille (près de caméra) ou assez petit (loin de caméra). S'il y a plus d'une personne à la maison, il n'est pas nécessaire que notre système fonctionne, car les autres personnes peuvent demander de l'aide si la personne âgée tombe. Le système s'éteindra automatiquement et dormira jusqu'à ce que les personnes âgées se mettent

en marche manuellement ou en se basant sur certaines techniques de comptage du nombre de personnes, telles que [168, 148]. Pour le cas où la silhouette de la personne âgée est assez grande ou petite, la silhouette de la personne est la seule extraite du premier plan. Pour déterminer s'il s'agit d'un animal ou d'une silhouette de la personne, il existe des techniques de classification d'objets telles que [169, 170, 171], ou en utilisant des méthodes d'apprentissage en profondeur comme dans [172].

Un autre problème est l'occlusion. L'environnement domestique contient souvent de nombreux objets tels que des tables, des canapés, des chaises et d'autres objets. Ces objets provoquent parfois une occlusion qui se produit lorsque la personne âgée est derrière l'un d'entre eux. En conséquence, cela détériore les performances du système de détection de chute. A cet effet, l'ajout de plus d'une caméra pour surveiller les personnes âgées peut être utilisé pour s'assurer que tout le corps de la personne âgée ou la majeure partie de son corps se trouve devant au moins d'une caméra. Pour la détection des chutes, chaque caméra effectue la détection des chutes et les résultats sont combinés à l'aide de certaines techniques telles que le vote à la majorité pour prendre une décision. De telles stratégies de vote majoritaire ont été utilisées dans [31].

4.5 Conclusion

Dans cet contribution, nous avons proposé un système de détection des chutes pour les personnes âgées basé sur la reconnaissance de la posture à l'aide d'une seule caméra. Le processus de notre nouvelle approche est simple. En effet, dans notre système, nous soustrayons d'abord l'arrière-plan pour extraire la silhouette de la personne de l'image vidéo à l'aide de l'algorithme CB. Une détection d'ombre a été ajoutée pour améliorer la détection de premier plan. Ensuite, nous utilisons la méthode que nous proposons pour extraire les caractéristiques locales et globales de cette silhouette de la personne. Ces caractéristiques sont combinées et données au classifieur pour prédire son type de posture, comme la posture allongée, assise, debout et s'accroupir. Les caractéristiques locales sont l'histogramme de projection extrait à base de BB ; la silhouette de la personne est divisée horizontalement et verticalement en partitions égales en utilisant un pas d'angle. L'intersection des partitions horizontales et verticales fournit notre descripteur locale. Les caractéristiques globales sont l'angle horizontal et l'angle vertical combinés avec le rapport entre eux. L'évaluation des caractéristiques proposées extraites pour la reconnaissance de la posture montre un bon résultat par rapport aux méthodes de l'état de l'art. De plus, nous évaluons notre méthode en utilisant les classificateurs courants tels que SVM, NN, KNN, RF et DT. Les résultats expérimentaux ont montré que le classifieur SVM donne les meilleurs résultats.

Après l'étape de la reconnaissance de la posture, quatre règles pour la vérification de détection des chutes sont utilisées. La chute est détectée si ces conditions sont satisfaites. Les résultats obtenus montrent que notre méthode obtient des meilleures performances avec une grande précision pour la reconnaissance des chutes et une

faible détection de fausses détections.

Bien que ces résultats obtenus, nous avons encore quelques points que nous prévoyons de les ajouter et l'améliorer dans le future pour rendre notre système plus général. Le multi-personne et l'occlusion presque total sont deux point qui manque dans notre système. la solution prévu est d'utiliser au moins deux caméras en ajoutant des techniques de comptage de personnes et de reconnaissance d'objets.

CONCLUSION ET LES PERSPECTIVES

5.1 Conclusion

Notre premier objectif dans ce travail est de faire une étude globale sur les systèmes de détection de chute existant dans la littérature notamment les systèmes qui se basent sur des caméras. Ensuite, faire une comparaison de ces systèmes en termes de performance et de complexité. A partir de cette étude, nous arrivons à proposer deux contributions pour détecter la chute à travers une seule caméra.

La première contribution consiste à proposer un système qui permet de modéliser les activités humaines par des informations de forme et de mouvement. Les informations de forme permettent d'identifier le changement de la forme de la personne durant ses activités quotidiennes. En ce qui concerne les informations de mouvement, ils nous permettent de distinguer entre les mouvements contrôlés et les mouvements non contrôlés comme la chute. Au premier temps, nous avons proposé une méthode à base de seuillage pour détecter la chute. Vu le passage d'une base de données à une autre base de données différente nécessite l'adaptation des seuils de toutes les informations et ajuste toutes les règles pour identifier la chute. En deuxième temps, nous avons proposé d'utiliser une méthode supervisée pour la détection de chute automatiquement. La combinaison de ces informations construit notre descripteur qui identifie la classe ou le type de chaque activité de la personne y compris la chute. Pour savoir la performance de notre système, nous avons utilisé plusieurs bases de données en comparant avec les approches de l'état de l'art [31, 3, 73, 108, 116]. Le résultat obtenu nous montre des bons résultats par rapport aux approches précédentes. Le travail le plus proche est de [3] qui sont arrivés à 93.6% en précision. En comparant à cette approche, notre système arrive à 94.60%. Notre descripteur arrive à identifier la chute correctement par rapport à leur descripteur proposé.

La deuxième contribution consiste à développer la reconnaissance de la posture de la personne pour identifier les activités de chute à travers une seule caméra RGB. Nous avons proposé un descripteur de forme qui nous permet de modéliser et identifier la posture de la personne à n'importe quel moment dans la vidéo. Le descripteur se construit à partir des informations globales et des informations locales extraites uniquement de la silhouette 2D de la personne. Les classes des postures que nous arrivons à reconnaître sont : debout, allongée, s'asseoir et accroupir. Après la reconnaissance de posture, nous ajoutons une autre partie pour identifier et confirmer la chute. Si la posture est toujours classifiée comme allongée ou accroupie pendant 1 seconde, notre système commence l'analyse de la chute par l'ajout des caractéris-

tiques du sol et de mouvement afin de vérifier et confirmer la chute. Le sol nous permet de savoir si le corps de la personne est totalement ou partiellement sur le sol. Les informations de mouvement nous permettent de savoir si la personne ne bouge plus après un instant. Nous avons utilisé quatre bases de données pour évaluer notre système. 1 base de données des silhouettes pour entraîner notre modèle pour la reconnaissance de la posture. Les autres bases de données sont utilisées pour évaluer la performance de la reconnaissance de chute. Nous avons comparé notre système de reconnaissance de posture avec des méthodes supervisées dont la plus proche est [18], et aussi les méthodes de deeplearning [150, 95]. Notre système arrive à battre les travaux de [18, 150] en termes de performance en précision et en exactitude. En comparant notre résultat avec le travail [95], notre système atteint des bons résultats en exactitude pour la même base de données. En exploitant le résultat de la reconnaissance de posture dans la détection de chute, notre système ajoute la stratégie de vote et l'information de sol et de mouvement pour confirmer la chute. Cette technique nous permet d'aboutir des bons résultats en termes de taux de performance par rapport à l'état de l'art.

Malgré les résultats obtenus par notre système, il reste toujours des améliorations à faire pour construire un système complet et idéal. Nous discutons les limites et axes qu'on veut traiter dans le futur pour produire un système plus robuste.

5.2 Limites

Notre système proposé durant notre recherche n'est pas encore robuste dans tous les cas. Nous citons des limites de notre travail de recherche ci-dessous :

- Utilisation seulement la caméra RGB pose des difficultés au niveau de l'extraction de la silhouette de la personne dû au changement de lumière et aussi le déplacement des meubles.
- Reconnaissance de la posture échoue quand la personne n'est pas visible totalement par la caméra.
- Manque de traitement de plusieurs personnes dans la vidéo.
- Temps d'exécution du système reste toujours loin de temps réel dû à l'utilisation de l'algorithme de détection de mouvement avec flot optique qui prend plus de temps. Amélioration du temps d'exécution est primordiale dans un système de vidéo surveillance.
- La détection de la silhouette par l'utilisation de l'extraction de l'arrière plan rencontre des difficultés dans un environnement où le changement de luminosité existe souvent.
- Reconnaître seulement quatre types de posture (debout, allongée, s'asseoir, accroupir).
- Détection du sol est manuelle. L'ajout d'un traitement automatique est forcément préférable malgré l'augmentation du temps d'exécution du système.

5.3 Perspectives

Pour la suite de ce travail de recherche, nous allons poursuivre l'amélioration de plusieurs points ci-dessous :

- Combinaison de l'information RGB et l'information de profondeur pour améliorer l'analyse de la forme.
- Utilisation de plusieurs caméras pour éviter le problème d'occlusion.
- Amélioration du temps d'exécution de notre système par une optimisation de code avec l'utilisation de parallélisme.
- Ajout de traitement de la reconnaissance de plusieurs personnes.
- Ajout de l'apprentissage approfondi pour la reconnaissance de posture et la détection de chute.



BIBLIOGRAPHIE

- [1] N Noury, P Rumeau, AK Bourke, G ÓLaighin, and JE Lundy. A proposal for the classification and evaluation of fall detectors. *Irbm*, 29(6) :340–349, 2008.
- [2] Edouard Auvinet, Caroline Rougier, Jean Meunier, Alain St-Arnaud, and Jacqueline Rousseau. Multiple cameras fall dataset. *DIRO-Universite de Montreal, Tech. Rep*, 1350, 2010.
- [3] Jia-Luen Chua, Yoong Choon Chang, and Wee Keong Lim. A simple vision-based fall detection technique for indoor video surveillance. *Signal, Image and Video Processing*, 9(3) :623–633, 2015.
- [4] Imen Charfi, Johel Miteran, Julien Dubois, Mohamed Atri, and Rached Tourki. Definition and performance evaluation of a robust svm based fall detection solution. In *Signal Image Technology and Internet Based Systems (SITIS), 2012 Eighth International Conference on*, pages 218–224. IEEE, 2012.
- [5] Elizabeth R. Burns Gwen Bergen, Mark R. Stevens. *Falls and Fall Injuries Among Adults Aged more than 65 Years : Morbidity and Mortality Weekly Report (MMWR)*. 2014.
- [6] Organisation Mondiale De La Santé. Rapport mondial sur le vieillissement et la santé. *Récupéré de https://apps.who.int/iris/bitstream/handle/10665/206556/9789240694842_fre.pdf*, 2016.
- [7] Roxane Villeneuve, Céline Meillon, Jean-François, and Helene Amieva. Trajectory of quality of life before and after entering a nursing home : A longitudinal study. *Journal of Geriatric Psychiatry and Neurology*, page 0891988720964259, 2020.
- [8] Haider Mshali, Tayeb Lemlouma, Maria Moloney, and Damien Magoni. A survey on health monitoring systems for health smart homes. *International Journal of Industrial Ergonomics*, 66 :26–56, 2018.
- [9] Wael Maktouf, Sylvain Durand, Sébastien Boyas, Camille Pouliquen, and Bruno Beaune. Combined effects of aging and obesity on postural control, muscle activity and maximal voluntary force of muscles mobilizing ankle joint. *Journal of Biomechanics*, 79 :198–206, 2018.

-
- [10] Olivier Beauchet, V Dubost, Christine Revel-Delhom, Gilles Berrut, and J Belmin. How to manage recurrent falls in clinical practice : guidelines of the french society of geriatrics and gerontology. *The journal of nutrition, health & aging*, 15(1) :79–84, 2011.
- [11] Ellen Vlaeyen, Joke Coussement, Greet Leysens, Elisa Van der Elst, Kim Delbaere, Dirk Cambier, Kris Denhaerynck, Stefan Goemaere, Arlette Wertelaers, Fabienne Dobbels, et al. Characteristics and effectiveness of fall prevention programs in nursing homes : A systematic review and meta-analysis of randomized controlled trials. *Journal of the American Geriatrics Society*, 63(2) :211–221, 2015.
- [12] Rein Tideiksaar. *Falls in older people : Prevention and management*. Health Professions Press, 2002.
- [13] Hiram Ponce, Lourdes Martínez-Villaseñor, José Núñez-Martínez, Ernesto Moya-Albor, and Jorge Brieva. Open source implementation for fall classification and fall detection systems. In *Challenges and Trends in Multimodal Fall Detection for Healthcare*, pages 3–29. Springer, 2020.
- [14] B. Ma Y. Tang and H. Yan. Intelligent video surveillance system for elderly people living alone based on odvs. *Advances in Internet of Things*, 2013.
- [15] Zhong Zhang, Weihua Liu, Vangelis Metsis, and Vassilis Athitsos. A viewpoint-independent statistical method for fall detection. In *Pattern Recognition (ICPR), 2012 21st International Conference on*, pages 3626–3630. IEEE, 2012.
- [16] Glen Debar, Jonas Van den Bergh, Bert Bonroy, Mieke Deschodt, Eddy Dejaeger, Koen Milisen, Toon Goedemé, and Bart Vanrumste. Fallcam : Practical considerations in implementing a camera-based fall detection system. In *PoCA 09*, pages 17–23, 2009.
- [17] Ziyun Cai, Jungong Han, Li Liu, and Ling Shao. Rgb-d datasets using microsoft kinect or similar sensors : a survey. *Multimedia Tools and Applications*, pages 1–43, 2016.
- [18] Miao Yu, Adel Rhuma, Syed Mohsen Naqvi, Liang Wang, Jonathon Chambers, et al. A posture recognition-based fall detection system for monitoring an elderly person in a smart home environment. *IEEE transactions on information technology in biomedicine*, 16(6) :1274–1286, 2012.
- [19] Glen Debar, Greet Baldewijns, Toon Goedemé, Tinne Tuytelaars, and Bart Vanrumste. Camera-based fall detection using a particle filter. In *2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 6947–6950. IEEE, 2015.
- [20] Benaoumeur Senouci, Imen Charfi, Barthelemy Heyrman, Julien Dubois, and Johel Miteran. Fast prototyping of a soc-based smart-camera : a real-time fall detection case study. *Journal of Real-Time Image Processing*, pages 1–14, 2014.

-
- [21] Alban MEFFRE, Christophe COLLET, Nicolas LACHICHE, Pierre GANC, and ARSKI. Real-time fall detection method based on hidden markov modeling. 2012.
- [22] Kai Wang, Youjin Zhao, Qingyu Xiong, Xiling Shen, Min Fan, and Min Gao. Video recognition of human fall based on spatiotemporal features. *Intelligent Automation & Soft Computing*, 22(2) :303–309, 2016.
- [23] Xin Ma, Haibo Wang, Bingxia Xue, Mingang Zhou, Bing Ji, and Yibin Li. Depth-based human fall detection via shape features and improved extreme learning machine. *IEEE journal of biomedical and health informatics*, 18(6) :1915–1922, 2014.
- [24] Young-Sook Lee and Wan-Young Chung. Visual sensor based abnormal event detection with moving shadow removal in home healthcare applications. *Sensors*, 12(1) :573–584, 2012.
- [25] Yixiao Yun, C. Innocenti, G. Nero, H. Lind én, and Irene Yu-Hua Gu. Fall detection in rgb-d videos for elderly care. pages 422–427, Oct 2015.
- [26] Rita Cucchiara, Andrea Prati, and Roberto Vezzani. A multi-camera vision system for fall detection and alarm generation. *Expert Systems*, 24(5) :334–345, 2007.
- [27] Sebastian ZAMBANINI Jana MACHAJDIK and Martin KAMPEL. Fusion of data from multiple cameras for fall detection. 2010.
- [28] Alex Daniel Edgcomb. Automated video-based fall detection. 2014.
- [29] Robert J. Gripentog. Fall detection by using video. 2015.
- [30] Edouard Auvinet, Franck Multon, Alain Saint-Arnaud, Jacqueline Rousseau, and Jean Meunier. Fall detection with multiple cameras : An occlusion-resistant method based on 3-d silhouette vertical distribution. *IEEE Transactions on Information Technology in Biomedicine*, 15(2) :290–300, 2011.
- [31] Caroline Rougier, Jean Meunier, Alain St-Arnaud, and Jacqueline Rousseau. Robust video surveillance for fall detection based on human shape deformation. *IEEE Transactions on circuits and systems for video Technology*, 21(5) :611–622, 2011.
- [32] Glen Debar, Marc Mertens, Mieke Deschodt, Ellen Vlaeyen, Els Devriendt, Eddy Dejaeger, Koen Milisen, Jos Tournoy, Tom Croonenborghs, Toon Goedemé, et al. Camera-based fall detection using real-world versus simulated data : How far are we from the solution ? *Journal of Ambient Intelligence and Smart Environments*, 8(2) :149–168, 2016.
- [33] Miao Yu, Yuanzhang Yu, Adel Rhuma, Syed Mohsen Naqvi, Liang Wang, Jonathon A Chambers, et al. An online one class support vector machine-based person-specific fall detection system for monitoring an elderly individual in a

- room environment. *IEEE J. Biomedical and Health Informatics*, 17(6) :1002–1014, 2013.
- [34] Michal Kepski and Bogdan Kwolek. Person detection and head tracking to detect falls in depth maps. pages 324–331, 2014.
- [35] Kyungnam Kim, Thanarat H Chalidabhongse, David Harwood, and Larry Davis. Background modeling and subtraction by codebook construction. In *Image Processing, 2004. ICIP'04. 2004 International Conference on*, volume 5, pages 3061–3064. IEEE, 2004.
- [36] Chris Stauffer and W Eric L Grimson. Adaptive background mixture models for real-time tracking. In *Computer Vision and Pattern Recognition, 1999. IEEE Computer Society Conference on.*, volume 2. IEEE, 1999.
- [37] Nigel JB McFarlane and C Paddy Schofield. Segmentation and tracking of piglets in images. *Machine vision and applications*, 8(3) :187–193, 1995.
- [38] Carolina Maldonado, R Homero, Antonio Marin, et al. Feature selection to detect fallen pose using depth images. In *2016 International Conference on Electronics, Communications and Computers (CONIELECOMP)*, pages 94–100. IEEE, 2016.
- [39] Jin-Xiang Wang and Yan-Rong Hou. Pedestrian fall action detection and alarm in video surveillance. In *Information Science and Control Engineering (ICISCE), 2016 3rd International Conference on*, pages 502–505. IEEE, 2016.
- [40] D. P. Kumar, Y. Yun, and I. Y. H. Gu. Fall detection in rgb-d videos by combining shape and motion features. pages 1337–1341, March 2016.
- [41] Michal Kepski and Bogdan Kwolek. Fall detection using ceiling-mounted 3d depth camera. In *Computer Vision Theory and Applications (VISAPP), 2014 International Conference on*, volume 2, pages 640–647. IEEE, 2014.
- [42] Samuele Gasparrini, Enea Cippitelli, Susanna Spinsante, and Ennio Gambi. A depth-based fall detection system using a kinect® sensor. *Sensors*, 14(2) :2756–2775, 2014.
- [43] Bogdan Kwolek and Michal Kepski. Human fall detection on embedded platform using depth maps and wireless accelerometer. *Computer methods and programs in biomedicine*, 117(3) :489–501, 2014.
- [44] Michal Kepski and Bogdan Kwolek. Human fall detection by mean shift combined with depth connected components. In *International Conference on Computer Vision and Graphics*, pages 457–464. Springer, 2012.
- [45] Markus D Solbach and John K Tsotsos. Vision-based fallen person detection for the elderly. *CoRR*, abs/1707.07608, 2017.

-
- [46] Chao Tong, Yu Lian, Yang Zhang, Zhongyu Xie, Xiang Long, and Jianwei Niu. A novel real-time fall detection system based on real-time video and mobile phones. *Journal of Circuits, Systems and Computers*, 26(04) :1750056, 2017.
- [47] Thanh-Hai Tran, Thi-Lan Le, Van-Nam Hoang, and Hai Vu. Continuous detection of human fall using multimodal features from kinect sensors in scalable environment. *Computer Methods and Programs in Biomedicine*, 2017.
- [48] Chih-Yang Lin, Shang-Ming Wang, Jia-Wei Hong, Li-Wei Kang, and Chung-Lin Huang. Vision-based fall detection through shape features. In *Multimedia Big Data (BigMM), 2016 IEEE Second International Conference on*, pages 237–240. IEEE, 2016.
- [49] Viet Anh Nguyen, Thanh Ha Le, and Thuy Thi Nguyen. Single camera based fall detection using motion and human shape features. In *Proceedings of the Seventh Symposium on Information and Communication Technology*, pages 339–344. ACM, 2016.
- [50] Rui-dong Wang, Yong-liang Zhang, Ling-ping Dong, Jia-wei Lu, Zhi-qin Zhang, and Xia He. Fall detection algorithm for the elderly based on human characteristic matrix and svm. In *Control, Automation and Systems (ICCAS), 2015 15th International Conference on*, pages 1190–1195. IEEE, 2015.
- [51] Vinay Vishwakarma, Chittaranjan Mandal, and Shamik Sural. Automatic detection of human fall in video. *Pattern Recognition and Machine Intelligence*, pages 616–623, 2007.
- [52] Georgios Goudelis, Georgios Tsatiris, Kostas Karpouzis, and Stefanos Kollias. Fall detection using history triple features. In *Proceedings of the 8th ACM International Conference on Pervasive Technologies Related to Assistive Environments*, page 81. ACM, 2015.
- [53] Slim Abdelhedi, Ali Wali, and Adel M Alimi. Fuzzy logic based human activity recognition in video surveillance applications. In *Proceedings of the Second International Afro-European Conference for Industrial Advancement AECIA 2015*, pages 227–235. Springer, 2016.
- [54] Khosro Rezaee, Javad Haddadnia, and Ahmad Delbari. Intelligent detection of the falls in the elderly using fuzzy inference system and video-based motion estimation method. In *Machine Vision and Image Processing (MVIP), 2013 8th Iranian Conference on*, pages 284–288. IEEE, 2013.
- [55] Yixiao Yun and Irene Yu-Hua Gu. Human fall detection in videos via boosting and fusing statistical features of appearance, shape and motion dynamics on riemannian manifolds with applications to assisted living. *Computer Vision and Image Understanding*, 148 :111–122, 2016.
- [56] Nicolas Thome, Serge Miguet, and Sébastien Ambellouis. A real-time, multi-view fall detection system : A lhmm-based approach. *IEEE transactions on circuits and systems for video technology*, 18(11) :1522–1532, 2008.

-
- [57] Muzaffer Aslan, Abdulkadir Sengur, Yang Xiao, Haibo Wang, M Cevdet Ince, and Xin Ma. Shape feature encoding via fisher vector for efficient fall detection in depth-videos. *Applied Soft Computing*, 37 :1023–1028, 2015.
- [58] Lei Yang, Yanyun Ren, Huosheng Hu, and Bo Tian. New fast fall detection method based on spatio-temporal context tracking of head by using depth images. *Sensors*, 15(9) :23004–23019, 2015.
- [59] Dao Huu Hung and Hideo Saito. Fall detection with two cameras based on occupied area. In *Proc. of 18th Japan-Korea Joint Workshop on Frontier in Computer Vision*, pages 33–39, 2012.
- [60] Giovanni Diraco, Alessandro Leone, and Pietro Siciliano. An active vision system for fall detection and posture recognition in elderly healthcare. In *Proceedings of the conference on design, automation and test in Europe*, pages 1536–1541. European Design and Automation Association, 2010.
- [61] Mikaël A Mousse, Cina Motamed, and Eugène C Ezin. Fast moving object detection from overlapping cameras. In *Informatics in Control, Automation and Robotics (ICINCO), 2015 12th International Conference on*, volume 2, pages 296–303. IEEE, 2015.
- [62] Kewei Zhao, Kebin Jia, and Pengyu Liu. Fall detection algorithm based on human posture recognition. In *Advances in Intelligent Information Hiding and Multimedia Signal Processing : Proceeding of the Twelfth International Conference on Intelligent Information Hiding and Multimedia Signal Processing, Nov., 21-23, 2016, Kaohsiung, Taiwan, Volume 2*, pages 119–126. Springer, 2017.
- [63] Hyun-Gook Kang, Minseo Kang, and Jae-Gil Lee. Efficient fall detection based on event pattern matching in image streams. In *Big Data and Smart Computing (BigComp), 2017 IEEE International Conference on*, pages 51–58. IEEE, 2017.
- [64] Adel Rhuma, Miao Yu, and Jonathon Chambers. Posture recognition based fall detection system. *Lecture Notes on Software Engineering*, 1(4) :350, 2013.
- [65] Mikaël A Mousse, Cina Motamed, and Eugene c Ezin. Percentage of human-occupied areas for fall detection from two views. *The Visual Computer*, pages 1–12, 2016.
- [66] Nabil Zerrouki, Fouzi Harrou, Ying Sun, and Amrane Houacine. Accelerometer and camera-based strategy for improved human fall detection. *Journal of medical systems*, 40(12) :284, 2016.
- [67] Caroline Rougier, Jean Meunier, Alain St-Arnaud, and Jacqueline Rousseau. 3d head tracking for fall detection using a single calibrated camera. *Image and Vision Computing*, 31(3) :246–254, 2013.
- [68] Yi Ting Liao, Chung-Lin Huang, and Shih-Chung Hsu. Slip and fall event detection using bayesian belief network. *Pattern recognition*, 45(1) :24–32, 2012.

-
- [69] Homa Foroughi, Alireza Rezvanian, and Amirhossien Pazirae. Robust fall detection using human shape and multi-class support vector machine. In *Computer Vision, Graphics & Image Processing, 2008. ICVGIP'08. Sixth Indian Conference on*, pages 413–420. IEEE, 2008.
- [70] Caroline Rougier, Jean Meunier, Alain St-Arnaud, and Jacqueline Rousseau. Fall detection from human shape and motion history using video surveillance. In *Advanced Information Networking and Applications Workshops, 2007, AI-NAW'07. 21st International Conference on*, volume 2, pages 875–880. IEEE, 2007.
- [71] Liyuan Li, Weimin Huang, Irene YH Gu, and Qi Tian. Foreground object detection from videos containing complex background. In *Proceedings of the eleventh ACM international conference on Multimedia*, pages 2–10. ACM, 2003.
- [72] Kaibo Fan, Ping Wang, Yan Hu, and Bingjie Dou. Fall detection via human posture representation and support vector machine. *International Journal of Distributed Sensor Networks*, 13(5) :1550147717707418, 2017.
- [73] Mohamed Maher Ben Ismail and Ouiem Bchir. Automatic fall detection using membership based histogram descriptors. *Journal of Computer Science and Technology*, 32(2) :356–367, 2017.
- [74] Caroline Rougier, Edouard Auvinet, Jacqueline Rousseau, Max Mignotte, and Jean Meunier. Fall detection from depth map video sequences. In *International Conference on Smart Homes and Health Telematics*, pages 121–128. Springer, 2011.
- [75] Hao Liu and Yongcai Guo. A vision-based fall detection algorithm of human in indoor environment. In *Second International Conference on Photonics and Optical Engineering*, pages 1025644–1025644. International Society for Optics and Photonics, 2017.
- [76] Weiguo Feng, Rui Liu, and Ming Zhu. Fall detection for elderly person care in a vision-based home surveillance environment using a monocular camera. *Signal, Image and Video Processing*, 8(6) :1129–1138, 2014.
- [77] Glen Debar, Peter Karsmakers, Mieke Deschodt, Ellen Vlaeyen, Eddy Dejaeger, Koen Milisen, Toon Goedemé, Bart Vanrumste, and Tinne Tuytelaars. Camera-based fall detection on real world data. In *Outdoor and large-scale real-world scene analysis*, pages 356–375. Springer, 2012.
- [78] Jared Willems, Glen Debar, Bart Vanrumste, and Toon Goedemé. A video-based algorithm for elderly fall detection. In *World Congress on Medical Physics and Biomedical Engineering, September 7-12, 2009, Munich, Germany*, pages 312–315. Springer, 2009.
- [79] Chieh-Ling Huang, E-Liang Chen, and Pau-Choo Chung. Fall detection using modular neural networks with back-projected optical flow. *Biomedical Engineering : Applications, Basis and Communications*, 19(06) :415–424, 2007.

-
- [80] Leo Breiman. Random forests. *Machine learning*, 45(1) :5–32, 2001.
- [81] Mirto Musci, Daniele De Martini, Nicola Blago, Tullio Facchinetti, and Marco Piastra. Online fall detection using recurrent neural networks on smart wearable devices. *IEEE Transactions on Emerging Topics in Computing*, pages 1–1, 2020.
- [82] Daniele Berardini, Sara Moccia, Lucia Migliorelli, Iacopo Pacifici, Paolo di Massimo, Marina Paolanti, and Emanuele Frontoni. Fall detection for elderly-people monitoring using learned features and recurrent neural networks. *Experimental Results*, 1 :e7, 2020.
- [83] Seppo Enarvi, Peter Smit, Sami Virpioja, and Mikko Kurimo. Automatic speech recognition with very large conversational finnish and estonian vocabularies. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 25(11) :2085–2097, 2017.
- [84] Mikel Canizo, Isaac Triguero, Angel Conde, and Enrique Onieva. Multi-head cnn-rnn for multi-time series anomaly detection : An industrial case study. *Neurocomputing*, 363 :246–260, 2019. Article not yet published in issue. On temp embargo sheet. Will need to add volume, issue, pagination. KJB 31.07.2019.
- [85] Yuming Hua, Junhai Guo, and Hua Zhao. Deep belief networks and deep learning. In *Proceedings of 2015 International Conference on Intelligent Computing and Internet of Things*, pages 1–4, 2015.
- [86] Christopher Pramerdorfer, Rainer Planinc, Mark Van Loock, David Fankhauser, Martin Kampel, and Michael Brandstötter. Fall detection based on depth-data in practice. In *European Conference on Computer Vision*, pages 195–208. Springer, 2016.
- [87] Lei Yang, Yanyun Ren, and Wenqiang Zhang. 3d depth image analysis for indoor fall detection of elderly people. *Digital Communications and Networks*, 2(1) :24–34, 2016.
- [88] Alessandro Manzi, Filippo Cavallo, and Paolo Dario. A 3d human posture approach for activity recognition based on depth camera. In *European Conference on Computer Vision*, pages 432–447. Springer, 2016.
- [89] J Jeffin Gracewell and S Pavalarajan. Fall detection based on posture classification for smart home environment. *Journal of Ambient Intelligence and Humanized Computing*, pages 1–8, 2019.
- [90] Ying-Nong Chen, Chi-Hung Chuang, Hsin-Min Lee, Chih-Chang Yu, and Kuo-Chin Fan. Fall detection in dusky environment. *EURASIP Journal on Image and Video Processing*, 2016(1) :1, 2016.
- [91] Chenyang Zhang, Yingli Tian, and Elizabeth Capezuti. Privacy preserving automatic fall detection for elderly using rgbd cameras. In *International Conference on Computers for Handicapped Persons*, pages 625–633. Springer, 2012.

-
- [92] Erdem Akagündüz, Muzaffer Aslan, Abdulkadir Şengür, Haibo Wang, and Melih Cevdet Ince. Silhouette orientation volumes for efficient fall detection in depth videos. *IEEE journal of biomedical and health informatics*, 21(3) :756–763, 2017.
- [93] Dhiraj, Raunak Manekar, Sumeet Saurav, Somsukla Maiti, Sanjay Singh, Santanu Chaudhury, Neeraj, Ravi Kumar, and Kamal Chaudhary. Activity recognition for indoor fall detection in 360-degree videos using deep learning techniques. In Bidyut B. Chaudhuri, Masaki Nakagawa, Pritee Khanna, and Sanjeev Kumar, editors, *Proceedings of 3rd International Conference on Computer Vision and Image Processing*, pages 417–429, Singapore, 2020. Springer Singapore.
- [94] Yaxiang Fan, Martin D. Levine, Gongjian Wen, and Shaohua Qiu. A deep neural network for real-time detection of falling humans in naturally occurring scenes. *Neurocomputing*, 260 :43 – 58, 2017.
- [95] Miao Yu, Liyun Gong, and Stefanos Kollias. Computer vision based fall detection by a convolutional neural network. In *Proceedings of the 19th ACM International Conference on Multimodal Interaction*, pages 416–420. ACM, 2017.
- [96] Martha Magali Flores-Barranco, Mario-Alberto Ibarra-Mazano, and Irene Cheng. *Accidental Fall Detection Based on Skeleton Joint Correlation and Activity Boundary*, pages 489–498. Springer International Publishing, Cham, 2015.
- [97] Rami Alazrai, Yaser Mowafi, and Eyad Hamad. A fall prediction methodology for elderly based on a depth camera. In *2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 4990–4993. IEEE, 2015.
- [98] Yixiao Yun and Irene Yu-Hua Gu. Human fall detection via shape analysis on riemannian manifolds with applications to elderly care. In *Image Processing (ICIP), 2015 IEEE International Conference on*, pages 3280–3284. IEEE, 2015.
- [99] D. H. Hung and H. Saito. The estimation of heights and occupied areas of humans from two orthogonal views for fall detection. *IEEJ Transactions on Electronics, Information and Systems*, 133(1) :117–127, 2013.
- [100] Matti Matilainen, Mark Barnard, and Olli Silvén. Unusual activity recognition in noisy environments. In *International Conference on Advanced Concepts for Intelligent Vision Systems*, pages 389–399. Springer, 2009.
- [101] Xin Ma and al. Sdufall : A depth video database for fall detection. 2014.
- [102] Jaeyong Sung, Colin Ponce, Bart Selman, and Ashutosh Saxena. Unstructured human activity detection from rgbd images. In *2012 IEEE international conference on robotics and automation*, pages 842–849. IEEE, 2012.

-
- [103] Samuele Gasparrini, Enea Cippitelli, Ennio Gambi, Susanna Spinsante, Jonas Wåhslén, Ibrahim Orhan, and Thomas Lindh. Proposal and experimental evaluation of fall detection solution based on wearable and depth data fusion. In *International conference on ICT innovations*, pages 99–108. Springer, 2015.
- [104] Zhen-Peng Bian, Junhui Hou, Lap-Pui Chau, and Nadia Magnenat-Thalmann. Fall detection based on body part tracking using a depth camera. *IEEE journal of biomedical and health informatics*, 19(2) :430–439, 2015.
- [105] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in Neural information processing systems*, 2012.
- [106] John J Hopfield. Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the national academy of sciences*, 79(8) :2554–2558, 1982.
- [107] Kripesh Adhikari, Hamid Bouchachia, and Hammadi Nait-Charif. Long short-term memory networks based fall detection using unified pose estimation. In *Twelfth International Conference on Machine Vision (ICMV 2019)*, volume 11433, page 114330H. International Society for Optics and Photonics, 2020.
- [108] Nanez-Marcos Adrian, Azkune Gorka, and Ignacio Arganda-Carreras. Vision-based fall detection with convolutional neural networks. *Wireless Communications and Mobile Computing*, 2017(9474806) :16, 2017.
- [109] Na Lu, Yidan Wu, Li Feng, and Jinbo Song. Deep learning for fall detection : Three-dimensional cnn combined with lstm on video kinematic data. *IEEE journal of biomedical and health informatics*, 23(1) :314–323, 2018.
- [110] Kripesh Adhikari, Hamid Bouchachia, and Hammadi Nait-Charif. Activity recognition for indoor fall detection using convolutional neural network. In *2017 Fifteenth IAPR International Conference on Machine Vision Applications (MVA)*, pages 81–84. IEEE, 2017.
- [111] Weidong Min, Hao Cui, Hong Rao, Zhixun Li, and Leiyue Yao. Detection of human falls on furniture using scene analysis based on deep learning and activity characteristics. *IEEE Access*, 6 :9324–9335, 2018.
- [112] Yaxiang Fan, Gongjian Wen, Deren Li, Shaohua Qiu, and Martin D Levine. Early event detection based on dynamic images of surveillance videos. *Journal of Visual Communication and Image Representation*, 51 :70–75, 2018.
- [113] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv :1409.1556*, 2014.
- [114] Chadia Khraief, Faouzi Benzarti, and Hamid Amiri. Elderly fall detection based on multi-stream deep convolutional networks. *Multimedia Tools and Applications*, pages 1–24, 2020.

-
- [115] Bo-Hua Wang, Jie Yu, Kuo Wang, Xuan-Yu Bao, and Ke-Ming Mao. Fall detection based on dual-channel feature integration. *IEEE Access*, 8 :103443–103453, 2020.
- [116] Fouzi Harrou, Nabil Zerrouki, Ying Sun, and Amrane Houacine. An integrated vision-based approach for efficient human fall detection in a home environment. *IEEE Access*, 7 :114966–114974, 2019.
- [117] Kaibo Fan, Ping Wang, and Shuo Zhuang. Human fall detection using slow feature analysis. *Multimedia Tools and Applications*, Jan 2018.
- [118] Ahmad Lotfi, Suad Albawendi, Heather Powell, Kofi Appiah, and Caroline Langensiepen. Supporting independent living for older adults; employing a visual based fall detection through analysing the motion and shape of the human body. *IEEE Access*, 6 :70272–70282, 2018.
- [119] Kaibo Fan, Ping Wang, Yan Hu, and Bingjie Dou. Fall detection via human posture representation and support vector machine. *International Journal of Distributed Sensor Networks*, 13(5) :1550147717707418, 2017.
- [120] Tanvi Banerjee, James M Keller, Marjorie Skubic, and Erik Stone. Day or night activity recognition from video using fuzzy clustering techniques. *IEEE Transactions on Fuzzy Systems*, 22(3) :483–493, 2014.
- [121] Mei Jiang, Yuyang Chen, Yanyun Zhao, and Anni Cai. A real-time fall detection system based on hmm and rvm. In *Visual Communications and Image Processing (VCIP), 2013*, pages 1–6. IEEE, 2013.
- [122] Sen Wang, Zhongwen Xu, Yi Yang, Xue Li, Chaoyi Pang, and Alexander G Hauptmann. Fall detection in multi-camera surveillance videos : experimentations and observations. In *Proceedings of the 1st ACM international workshop on Multimedia indexing and information retrieval for healthcare*, pages 33–38. ACM, 2013.
- [123] Rainer Planinc and Martin Kampel. Robust fall detection by combining 3d data and fuzzy logic. In *Asian Conference on Computer Vision*, pages 121–132. Springer, 2012.
- [124] Yie-Tarng Chen, Yu-Ching Lin, and Wen-Hsien Fang. A hybrid human fall detection scheme. In *Image Processing (ICIP), 2010 17th IEEE International Conference on*, pages 3485–3488. IEEE, 2010.
- [125] Derek Anderson, Robert H Luke, James M Keller, Marjorie Skubic, Marilyn Rantz, and Myra Aud. Linguistic summarization of video for fall detection using voxel person and fuzzy logic. *Computer vision and image understanding*, 113(1) :80–89, 2009.
- [126] Homa Foroughi, Aabed Naseri, Alireza Saberi, and Hadi Sadoghi Yazdi. An eigenspace-based approach for human fall detection using integrated time motion image and neural network. In *Signal Processing, 2008. ICSP 2008. 9th International Conference on*, pages 1499–1503. IEEE, 2008.

-
- [127] Jia-Luen chua and al. Fall video dataset. 2013.
- [128] Greet Baldewijns, Glen Debar, Gert Mertes, Bart Vanrumste, and Tom Croonenborghs. Bridging the gap between real-life data and simulated data by providing a highly realistic fall dataset for evaluating camera-based fall detection algorithms. *Healthcare technology letters*, 3(1) :6–11, 2016.
- [129] S. Gasparrini, E. Cippitelli, E. Gambi, S. Spinsante, and F. Florez-Revuelta. Performance analysis of self-organising neural networks tracking algorithms for intake monitoring using kinect. In *Technologies for Active and Assisted Living (TechAAL), IET International Conference on*, pages 1–6, Nov 2015.
- [130] Zhong Zhang and Vassilis Athitsos. Fall detection using kinect sensor and viewpoint-independent training. 2012.
- [131] Shadi Khawandi, Bassam Daya, and Pierre Chauvet. Automated monitoring system for fall detection in the elderly. *International Journal of Image Processing (IJIP)*, 4(5) :476, 2010.
- [132] Khosro Rezaee, Javad Haddadnia, and Ahmad Delbari. Modeling abnormal walking of the elderly to predict risk of the falls using kalman filter and motion estimation approach. *Computers & Electrical Engineering*, 46 :471–486, 2015.
- [133] Matthias Schröder and Helge Ritter. Deep learning for action recognition in augmented reality assistance systems. 2017.
- [134] Yaxiang Fan, Martin D Levine, Gongjian Wen, and Shaohua Qiu. A deep neural network for real time detection of falling humans in naturally occurring scenes. *Neurocomputing*, 2017.
- [135] Erdem Akagündüz. Shape recognition using orientational and morphological scale-spaces of curvatures. *IET Computer Vision*, 9(5) :750–757, 2015.
- [136] Laurenz Wiskott and Terrence J. Sejnowski. Slow feature analysis : Unsupervised learning of invariances. *Neural Comput.*, 14(4) :715–770, April 2002.
- [137] Shervin Minaee and Yao Wang. An admn approach to masked signal decomposition using subspace representation. *IEEE Transactions on Image Processing*, 28(7) :3192–3204, 2019.
- [138] M. C. Bakkay, H. A. Rashwan, H. Salmane, L. Khoudour, D. Puig, and Y. =Ruichek. Bscgan : Deep background subtraction with conditional generative adversarial networks. In *2018 25th IEEE International Conference on Image Processing (ICIP)*, pages 4018–4022, 2018.
- [139] Vitor Gomes, Pablo Barcellos, and Jacob Scharcanski. Stochastic shadow detection using a hypergraph partitioning approach. *Pattern Recognition*, 63 :30–44, 2017.
- [140] Rafael C Gonzalez, Richard E Woods, and Barry R Masters. Digital image processing third edition. *Pearson International Edition*, 2008.

-
- [141] Shervin Minaee, Yuri Boykov, Fatih Porikli, Antonio Plaza, Nasser Kehtarnavaz, and Demetri Terzopoulos. Image segmentation using deep learning : A survey. *arXiv preprint arXiv :2001.05566*, 2020.
- [142] Vijay Badrinarayanan, Alex Kendall, and Robert Cipolla. Segnet :a deep convolutional encoder-decoder architecture for image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017.
- [143] Gunnar Farneback. Two-frame motion estimation based on polynomial expansion. In *Image Analysis*, pages 363–370. Springer), 2003.
- [144] Jean-Yves Bouguet. Pyramidal implementation of the lucas kanade feature tracker : Description of the algorithm. Technical report, Intel Microprocessor Research Labs, 2000.
- [145] Chih-Chung Chang and Chih-Jen Lin. Libsvm : a library for support vector machines. *ACM transactions on intelligent systems and technology (TIST)*, 2(3) :27, 2011.
- [146] Tae Meon Bae, Sung Ho Jin, and Yong Man Ro. Video segmentation using hidden markov model with multimodal features. In *International Conference on Image and Video Retrieval*, pages 401–409. Springer), 2004.
- [147] Yun Zhai and Mubarak Shah. Video scene segmentation using markov chain monte carlo. *IEEE transactions on Multimedia*, 8(4) :686–697, 2006.
- [148] S Khan, Giuseppe Vizzari, Stefania Bandini, and Saleh Basalamah. Detecting dominant motion flows and people counting in high density crowds. *Journal of WSCG*, 22(1) :21–30, 2014.
- [149] Nabil Zerrouki, Fouzi Harrou, Amrane Houacine, and Ying Sun. Fall detection using supervised machine learning algorithms. a comparative study. In *8th International Conference on Modelling, Identification and Control (ICMIC), 2016*, pages 665–670. IEEE, 2016.
- [150] Pengming Feng, Miao Yu, Syed Mohsen Naqvi, and Jonathon A Chambers. Deep learning for posture analysis in fall detection. In *Digital Signal Processing (DSP), 2014 19th International Conference on*, pages 12–17. IEEE, 2014.
- [151] G. Bradski. The OpenCV Library. *Dr. Dobb's Journal of Software Tools*, 2000.
- [152] J. Wang, Z. Huang, W. Zhang, A. Patil, K. Patil, T. Zhu, E. J. Shiroma, M. A. Schepps, and T. B. Harris. Wearable sensor based human posture recognition. In *2016 IEEE International Conference on Big Data (Big Data)*, pages 3432–3438, Dec 2016.
- [153] Jian Huang, Xiaoqiang Yu, Yuan Wang, and Xiling Xiao. An integrated wireless wearable sensor system for posture recognition and indoor localization. *Sensors*, 16(11) :1825, 2016.

-
- [154] Manoranjan Paul, Shah ME Haque, and Subrata Chakraborty. Human detection in surveillance videos and its applications-a review. *EURASIP Journal on Advances in Signal Processing*, 2013(1) :176, 2013.
- [155] Corinne Mattmann, Oliver Amft, Holger Harms, Gerhard Troster, and Frank Clemens. Recognizing upper body postures using textile strain sensors. In *Wearable Computers, 2007 11th IEEE International Symposium on*, pages 29–36. IEEE, 2007.
- [156] Lu Xia, Chia-Chih Chen, and Jake K Aggarwal. View invariant human action recognition using histograms of 3d joints. In *Computer vision and pattern recognition workshops (CVPRW), 2012 IEEE computer society conference on*, pages 20–27. IEEE, 2012.
- [157] Haibin Ling and David W Jacobs. Shape classification using the inner-distance. *IEEE transactions on pattern analysis and machine intelligence*, 29(2) :286–299, 2007.
- [158] Adnan Abou Nabout. Object shape recognition using wavelet descriptors. *Journal of Engineering*, 2013, 2013.
- [159] Quang Minh Tieng and WW Boles. Recognition of 2d object contours using the wavelet transform zero-crossing representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(8) :910–916, 1997.
- [160] Jun-Wei Hsieh, Yung-Tai Hsu, Hong-Yuan Mark Liao, and Chih-Chiang Chen. Video-based human movement analysis and its application to surveillance systems. *IEEE Transactions on Multimedia*, 10(3) :372–384, 2008.
- [161] Bin Wang and Yongsheng Gao. Structure integral transform versus radon transform : A 2d mathematical tool for invariant shape recognition. *IEEE Transactions on Image Processing*, 25(12) :5635–5648, 2016.
- [162] Simin Wang, Salim Zabir, and Bastian Leibe. Lying pose recognition for elderly fall detection. *Robotics : Science and Systems VII*, 345, 2012.
- [163] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. Segnet : A deep convolutional encoder-decoder architecture for image segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 39(12) :2481–2495, 2017.
- [164] RPW Duin and EM Pekalska. Pattern recognition : Introduction and terminology. 2016.
- [165] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(4) :509–522, 2002.
- [166] Behzad Mirmahboub, Shadrokh Samavi, Nader Karimi, and Shahram Shirani. Automatic monocular system for human fall detection based on variations in silhouette area. *IEEE Transactions on Biomedical Engineering*, 60(2) :427–436, 2013.

-
- [167] Chien-Liang Liu, Chia-Hoang Lee, and Ping-Min Lin. A fall detection system using k-nearest neighbor classifier. *Expert systems with applications*, 37(10) :7174–7181, 2010.
- [168] Satoshi Yoshinaga, Atsushi Shimada, and Rin-ichiro Taniguchi. Real-time people counting using blob descriptor. *Procedia-Social and Behavioral Sciences*, 2(1) :143–152, 2010.
- [169] Peng Tang, Xinggang Wang, Zilong Huang, Xiang Bai, and Wenyu Liu. Deep patch learning for weakly supervised object classification and discovery. *Pattern Recognition*, 71 :446–459, 2017.
- [170] Yi Wu, Jongwoo Lim, and Ming-Hsuan Yang. Object tracking benchmark. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(9) :1834–1848, 2015.
- [171] Federico Lecumberry, Álvaro Pardo, and Guillermo Sapiro. Simultaneous object classification and segmentation with high-order multiple shape models. *IEEE Transactions on Image Processing*, 19(3) :625–635, 2010.
- [172] Zhong-Qiu Zhao, Peng Zheng, Shou-tao Xu, and Xindong Wu. Object detection with deep learning : A review. *arXiv preprint arXiv :1807.05511*, 2018.