Reconnaissance d'actions humaines par apprentissage profond et génération de données étiquetées basées sur le jumeau numérique de poste cobotique industriel.

Mejdi Dallel

Normandie Université

THÈSE

Pour obtenir le diplôme de doctorat

Spécialité INFORMATIQUE

Préparée au sein de l'Université de Rouen Normandie

Reconnaissance d'actions humaines par apprentissage profond et génération de données étiquetées basées sur le jumeau numérique de poste cobotique industriel.

Présentée et soutenue par
MEJDI DALLEL

Thèse soutenue le 14/12/2022 devant le jury composé de		
M. ALI DOUIK	PROFESSEUR DES UNIVERSITES, Université de Sousse	Rapporteur du jury
M. BERNARD KAMSU FOGUEM	PROFESSEUR DES UNIVERSITES, ENIT (Tarbes)	Rapporteur du jury
M. VINCENT HAVARD	MAITRE DE CONFERENCES, CESI	Membre du jury
MME DANIELLE NUZILLARD	PROFESSEUR DES UNIVERSITES, UNIVERSITE REIMS CHAMPAGNE ARDENNE	Membre du jury
M. RÉMI BOUTTEAU	PROFESSEUR DES UNIVERSITES, Université de Rouen Normandie	Président du jury
M. DAVID BAUDRY	PROFESSEUR DES UNIVERSITES, CESI	Directeur de thèse

Thèse dirigée par DAVID BAUDRY (LABO. INNOVATION NUMERIQUE POUR ENTREPRISES ET APPRENTISSAGES SERV. COMPET. TERRITOIRES)

UNIVERSITÉ DE ROUEN NORMANDIE

CESI ÉCOLE D'INGÉNIEURS

MIIS

CESI LINEACT

Remerciements

Je tiens à exprimer ma profonde gratitude et mes sincères remerciements à David BAUDRY et Vincent HAVARD d'avoir accepté de diriger cette thèse. Merci de m'avoir accompagné et guidé dans de bonnes directions tout en m'accordant assez de confiance pour me laisser aborder la thèse avec une grande liberté. Merci pour vos directives précieuses et pour la qualité de votre suivi durant toute la période de la thèse. Merci pour votre esprit critique, j'ai beaucoup appris de nos discussions et du chemin que nous avons parcouru ensemble lors de cette période.

Mes remerciements s'adressent également à l'ensemble des membres du jury qui m'ont fait l'honneur de bien vouloir étudier avec attention mon travail : Ali DOUIK et Bernard KAMSU FOGUEM qui ont accepté le travail de rapporteurs et qui m'ont fourni des remarques pertinentes m'ayant permis d'améliorer mon manuscrit de thèse ; Danielle Nuzillard et Rémi BOUTTEAU pour avoir accepté d'examiner cette thèse ; et enfin Xavier SAVATIER et Yohan DUPUIS pour leur accompagnement et pour avoir accepté mon invitation à participer à ce jury.

Merci à mon comité de suivi de thèse (CSI), Danielle NUZILLARD et Francois GUERIN pour les conseils qu'ils m'ont adressés tout au long de ce travail qui m'a permis d'aller jusqu'au bout.

Je tiens à remercier également tous mes collègues au sein du laboratoire LINEACT - CESI, qui m'ont permis de m'épanouir et de partager ma passion pour l'informatique (Merouane MAZAR, Amine BENATIA, Mhamed SAHNOUN, Mourad MESSAADIA, Nicolas BRIANT, Elodie PILLION, Fabrice DUVAL, Belgacem BETTAYEB, Yiyi XU, Anne LOUIS, Marie NICOLINI, Thibaud BOUNHAR, Zaher YAMAK, Anne-Laure DUBOT, Amal AYADI, Noureddine BOUAZIZ, Jordan HIS). Je remercie aussi tous ceux que j'ai pu oublier. Merci au CESI en particulier le laboratoire LINEACT pour avoir créé un environnement aussi agréable et idéal pour accomplir ma thèse. Ce fut un honneur pour moi de faire partie de votre équipe. Merci également à mes collègues du département Informatique de l'ESIGELEC.

Merci à mes parents, Habib DALLEL et Sabeh LETAIEF, pour leurs amours inestimables, leurs soutiens et toutes les valeurs qu'ils ont su m'inculquer. Merci à mon frère Mhamed DALLEL et ma sœur Mouna DALLEL pour leurs tendresses, leurs complicités et leurs présences. Merci à mes amis de m'avoir accompagné moralement dans cette thèse et à nos soirées jeux vidéo pour décompresser en fin de thèse (Walid BEN OUNIS, Mejdi AMARA, Abdallah MAZAR, Maher BEN AHMED).

Mes remerciements vont enfin à toute personne qui a contribué de près ou de loin à l'élaboration de ce travail.

Résumé

La reconnaissance d'actions humaines (HAR) permet de faciliter les interactions et la collaboration humain-robot (HRC) au sein de l'industrie 4.0. En effet, les robots collaboratifs sont de plus en plus présents dans notre quotidien et induisent une interaction de plus en plus étroite entre l'humain et le robot, concept rassemblé dans le terme « cobotique ». Afin de garantir une collaboration efficace, les robots doivent être capables de comprendre leur environnement et doivent pouvoir communiquer sur leurs tâches en cours et leurs intentions. Cette communication et ces interactions représentent un enjeu majeur de performance et de sécurité. Dans ce contexte, cette thèse aborde le problème de reconnaissance d'actions humaines dans un milieu industriel pour répondre aux exigences de ses applications et aborder les problématiques de traitement en temps réel en impliquant un cas d'utilisation industriel lié à l'assemblage sur un poste cobotique d'un produit manufacturé.

Dans un premier temps, nous avons réalisé un état de l'art sur la collaboration humain-robot, les jeux de données de HAR et les méthodes associées. Cette étude a mis en évidence le manque des jeux de données de HAR dans un contexte industriel et nous a amené à proposer le jeu de données d'actions humaines industrielles nommé InHARD portant sur l'assemblage sur poste cobotique. L'introduction de ce jeu de données a révélé que l'entrainement des algorithmes de HAR pouvait bénéficier de l'apport des outils de Réalité Virtuelle (RV) permettant de simuler les interactions humains robots afin de surmonter les problèmes liés à la labélisation et au manque de données. Ainsi, nous avons proposé une méthodologie couplant jumeau numérique (DT) et réalité virtuelle pour extraire un modèle numérique des humains et permettre la génération automatique de données labélisées. Cette méthodologie a été appliquée pour créer le jeu de données InHARD-DT et nous avons évalué la robustesse et la généralisation de notre méthode en entrainant l'algorithme de HAR avec les données du jumeau numérique et en validant sur des données du jumeau physique. Les résultats montrent une généralisation atteignant les 89% de précision et de F1-score, prouvant la pertinence de l'approche proposée.

Nos études sur les algorithmes d'apprentissage profond basés sur des données squelettes ont été approfondies et ont permis de proposer une nouvelle méthode utilisant les réseaux de neurones convolutifs à graphes spatio-temporel avec une fenêtre glissante et un vote majoritaire nommé STGCN-SWMV. Cette approche permet une détection en temps réel sur des données en flux continu. Nous avons montré l'efficacité de la méthode présentée qui, en comparaison avec les méthodes de HAR de l'état de l'art, a obtenu de meilleures performances de classification sur les jeux de données OAD et UOW.

Les travaux de cette thèse ouvrent différentes possibilités et applications pour améliorer la collaboration humain-robot, qui est en adéquation avec la transition de l'industrie 4.0 vers l'industrie 5.0 plaçant l'humain au cœur de l'industrie.

Mots clés: Reconnaissance d'actions humaines industrielles (HAR), Industrie 4.0, Collaboration humain-robot (HRC), Jumeaux numériques (DT), Réseaux convolutifs à graphes spatio-temporels (ST-GCN).

Abstract

Human Action Recognition (HAR) facilitates human-robot (HRC) interactions and collaboration within Industry 4.0. Indeed, collaborative robots are increasingly present in our daily lives and induce an increasingly close interaction between humans and robots, a concept brought together in the term “cobotics”. In order to ensure effective collaboration, robots must be able to understand their environment and must be able to communicate about their ongoing tasks and intentions. This communication and these interactions represent a major performance and security challenge. In this context, this thesis addresses the problem of recognizing human actions in an industrial environment to meet the requirements of its applications and to address the problems of real-time processing by involving an industrial use case related to assembly on a cobotic station of a manufactured product.

First, we carried out a state of the art on human-robot collaboration, HAR datasets and associated methods. This study highlighted the lack of HAR datasets in an industrial context and led us to propose the Industrial Human Action Recognition dataset named InHARD relating to assembly on a cobotic station. The introduction of this dataset revealed that the training of HAR algorithms could benefit from the contribution of Virtual Reality (VR) tools allowing to simulate human-robot interactions in order to overcome the problems related to labeling and lack of data. Thus, we have proposed a methodology coupling Digital Twins (DT) and Virtual Reality to extract a digital model of humans and allow automatic labeled data generation. This methodology was applied to create the InHARD-DT dataset and we evaluated the robustness and generalizability of our method by training the HAR algorithm with data from the digital twin and validating on data from the physical twin. The results show a generalization reaching 89% of Accuracy and F1-score, proving the effectiveness of the proposed approach.

Our studies on Deep Learning (DL) algorithms based on skeleton data have been deepened and allowed to propose a new method using Spatial-Temporal Graph Convolutional Neural Networks with a Sliding Window and a Majority Voting named STGCN-SWMV. This approach allows real-time detection on continuous data streams. We have shown the efficiency of the presented method which, in comparison with state-of-the-art HAR methods, obtained better classification performance on the OAD and UOW datasets.

The work of this thesis opens up different possibilities and applications to improve human-robot collaboration, which is in line with the transition from industry 4.0 to industry 5.0, placing humans at the heart of industry.

Keywords: Industrial Human Action Recognition (HAR), Industry 4.0, Human-Robot Collaboration (HRC), Digital Twins (DT), Spatial-Temporal Graph Convolutional Networks (ST-GCN).

Table des matières

Liste des figures

Liste des tableaux

Liste des abréviations

ANN - Artificial Neural Network

BVH - Biovision Hierarchical Data

cAP - Calibrated Average Precision

CNN - Convolutional Neural Network

cP - Calibrated Precision

CP - Cross Setup

CS - Cross Subject

CV - Cross View

DAP - Deep Action Proposals

DBM - Deep Boltzmann Machine

DBN - Deep Belief Network

DCNN - Deep Convolutional Neural Network

DT - Digital Twins

EL - End Localization

EV - Environnements Virtuels

EVA - Environnements Virtuels pour l'Apprentissage

EVAH - Environnements Virtuels pour l'Apprentissage Humain

FPS - Frames per Second

GCA-LSTM - Global Context-Aware Attention LSTM

GCN - Graph Convolutional Network

GNN - Graph Neural Network

HAR - Human Action Recognition

HBM - Hierarchical Bayesian Model

HCF - Hierarchical Compound Features

HDP - Hierarchical Dirichlet Process

HMM - Hidden Markovian Models

HRC - Human Robot Collaboration

HTM - Hierarchical Temporal Memory

IA - Intelligence Artificielle

IHM - Interfaces Homme-Machine

IMU - Inertial Measurement Units

InHARD - Industrial Human Action Recognition Dataset

InHARD-DT - Industrial Human Action Recognition Dataset with Digital Twins

IoU - Intersection-over-Union

ISA - Independent Subspace Analysis

JCR-RNN - Joint Classification Regression - Recurrent Neural Network

JTM - Joint Trajectory Maps

LRCN - Long-term Recurrent Convolutional Networks

LSTM - Long Short-Term Memory

MAN - Memory Attention Network

mAP - Mean Average Precision

MoC - Mean-over-Classes

MoF - Mean-over-Frames

MV - Majority Voting

NN - Neural Network

PT - Physical Twin

RA - Réalité Augmentée

RBM - Restricted Boltzmann Machine

R-CNN - Region-based Convolutional Neural Network

RF - Random Forests

RI - Region of Interest

RGB-D - Red Green Blue Depth

RNN - Recurrent Neural Network

RV - Réalité Virtuelle

SDA - Stacked Denoising Autoencoder

ST-GCN - Spatial-Temporal Graph Convolutional Neural Network

STGCN-SWMV - Spatial-Temporal Graph Convolutional Neural Network with Sliding Window and Majority Voting

SL - Start Localization

SVM - Support Vector Machine

SW - Sliding Window

T-C3D - Temporal Convolutional 3D Network

TOF - Time of Flight

TRN - Temporal Recurrent Network

Introduction générale

Dans le contexte de l'industrie du futur, les évolutions technologiques telles que la robotique collaborative, les objets connectés, les algorithmes de traitement et de décisions et les nouvelles interfaces humains-machines ouvrent de nombreuses possibilités pour rendre les systèmes de productions plus flexibles. L'introduction de bras robotisés collaboratifs ou encore de robots mobiles autonomes au sein de l'atelier de production nécessitent des modes de communication et d'interaction appropriés entre humains et robots, ces derniers évoluant dans des espaces partagés. La communication et l'interaction entre les opérateurs du système de production et leur environnement représentent donc un enjeu majeur de performance et de sécurité et imposent une formation en amont des utilisateurs. Afin de garantir une collaboration efficace, l'humain doit pouvoir interrompre le robot dans son action pour le reconfigurer et le robot doit pouvoir communiquer sur sa tâche en cours et son intention et comprendre quelles actions l'humain effectue à côté de lui.

Il existe de nombreuses tâches de haut niveau que les humains exécutent automatiquement et inconsciemment. Mais en réalité, ces tâches impliquent un traitement complexe, qui a lieu dans notre cerveau. L'affirmation ci-dessus est particulièrement vraie pour toute tâche liée à la vision par ordinateur. Des tâches telles que la reconnaissance d'actions, la détection d'objets, ou l'identification de personnes sont des tâches extrêmement difficiles pour les ordinateurs. Les tâches de vision par ordinateur sont difficiles en raison des fortes variations d'apparence, de points de vue, de conditions d'éclairage etc. Toutefois, elles sont aisément résolues par le cerveau humain. Les ordinateurs ont été principalement conçus pour effectuer des tâches de calcul rapides et bien définies, mais pas des raisonnements complexes. Ainsi, le principal objectif de la vision ou perception par ordinateur est d'améliorer les capacités de l'ordinateur à interpréter les informations provenant de diverses sources : images, vidéos, capteurs IMU, etc. Ces capacités jouent un rôle clé dans les machines intelligentes telles que les robots ou les véhicules autonomes.

Les interactions entre l'humain et le robot, en environnement réel ou virtuel, et les prises de décisions associées entre les différents agents du système nécessitent donc une reconnaissance des actions réalisées par les opérateurs. La reconnaissance d'actions est l'un des composants nécessaires des systèmes intelligents. La capacité d'interpréter les informations disponibles rapprochera finalement les ordinateurs des compétences humaines. La reconnaissance d'actions humaines (HAR) peut être définie comme la capacité de déterminer si une action à identifier se produit dans le flux de données. La HAR grâce aux méthodes d'apprentissages et notamment les travaux récents basés sur des approches d'apprentissage profond ont montré leur efficacité.

Cependant, ces techniques nécessitent d'avoir de très grands jeux de données labélisées qui requièrent une intervention humaine pour les produire.

Dans ce contexte, cette thèse aborde le problème de reconnaissance d'actions humaines dans un milieu industriel pour répondre aux exigences de ses applications et aborder les problématiques de traitement en ligne en impliquant un cas d'utilisation industriel d'assemblage sur un poste cobotique d'un produit manufacturé. L'enjeu est tout d'abord d'explorer, d'expérimenter et d'évaluer des approches innovantes de reconnaissance d'actions humaines en ligne pour répondre aux besoins de sécurité et d'interactivité en industrie. Les approches étudiées utilisent des méthodes basées vision et apprentissage profond (CNN, GNN, RNN). Par ailleurs, afin de répondre à la problématique des grands jeux de données nécessaires, nos travaux s'orientent sur la génération automatique de jeux de données étiquetées, par simulation en réalité virtuelle basée sur le jumeau numérique, pour les algorithmes d'apprentissages profonds pour la reconnaissance d'actions industrielles. Cette étape d'apprentissage automatique pour de la reconnaissance gestuelle d'opérations réalisées dans des systèmes industriels est nécessaire pour permettre la collaboration humain-robot en environnement réel ou virtuel.

Plan de thèse

Le reste de ce document est organisé en quatre chapitres, qui sont décrits brièvement ici :

Chapitre 1 : Dans ce chapitre nous mettons d'abord en contexte le sujet de thèse dans son cadre scientifique et théorique lié à l'Industrie 4.0 et la Collaboration Humain-Robot (HRC) tout en exposant la reconnaissance d'actions humaines (HAR). Nous avons présenté par la suite les différents jeux de données de HAR les plus utilisés dans la littérature et nous avons passé en revue les méthodes de HAR les plus récentes et comparé leurs performances.

Chapitre 2 : Dans ce chapitre, nous proposons un jeu de données d'actions humaines industrielles nommé « Industrial Human Action Recognition Dataset (InHARD) ». Dans le contexte de la HRC, ce dernier présente des actions réalisées sur un poste d'assemblage manuel, assistées par un bras cobotique. Ce jeu de données peut aider la communauté scientifique à progresser dans la HAR dans les environnements industriels. En effet, les jeux de données existants comprennent principalement des actions du quotidien ou bien des actions liées à la santé, ce qui limite l'usage de la HAR dans l'industrie.

Chapitre 3 : Ce chapitre sera consacré à présenter une méthodologie qui utilise les jumeaux numériques (DT) pour générer des données automatiquement labélisées. Ces dernières sont utilisées pour le processus de HAR basé sur un algorithme d'apprentissage profond. Un jeu de données de reconnaissance d'actions humaines nommé InHARD-DT a été créé pour valider un cas d'utilisation réel dans lequel nous utilisons les données DT pour entraîner notre algorithme de HAR et les données du Jumeau Physique (PT) du jeu de données InHARD pour tester la

robustesse de l'apprentissage.

Chapitre 4 : Dans ce chapitre, nous proposons d'adapter un réseau de neurones convolutifs à graphe spatio-temporel (ST-GCN) avec une approche basée sur la technique de fenêtre glissante avec vote majoritaire pour attaquer la problématique de reconnaissance d'actions humaines en ligne. Nous avons évalué notre méthode sur deux jeux de données en ligne basés sur des données squelettes ; InHARD & InHARD-3-DT que nous avons proposés ainsi que deux jeux de données en ligne de la communauté nommés OAD et UOW.

Finalement, ce manuscrit se conclut par une synthèse du travail de cette thèse et nous proposons des perspectives à celui-ci.

Chapitre 1

État de l'art sur la collaboration humain-robot et la reconnaissance d'actions humaines

Sommaire

I.1 Introduction

Ce chapitre aura pour objectif de situer notre travail dans son cadre scientifique et théorique lié à l'Industrie 4.0 et la Collaboration Homme-Robot (HRC) (Section 1.2). Nous nous intéresserons après à présenter la reconnaissance et la détection d'actions humaines (HAR) (Section 1.3). Nous exposons ensuite les différents jeux de données de HAR les plus utilisés dans la littérature (Section 1.4). Par la suite nous passons en revue les méthodes de HAR les plus récentes et nous comparons leurs performances (Section 1.5). Nous examinons dans ce qui suit la reconnaissance d'actions humaines dans l'industrie (Section I.6). Enfin, nous concluons ce chapitre (Section 1.6).

I.2 Contexte Industrie 4.0 et Collaboration Humain-Robot (HRC)

Le facteur capital humain est un aspect d'une importance cruciale pour la mise en œuvre correcte de l'industrie 4.0. Même si l'on s'attend à ce que les usines intelligentes soient équipées d'un niveau d'automatisation plus élevé, « cette innovation importante n'élimine pas entièrement

le besoin d'opérateurs humains - au contraire - elle exige qu'ils collaborent avec les robots et exécutent des tâches hybrides » (Askarpour, et al. 2019). Les principaux avantages de la collaboration humain-robot dans une configuration partagée sont les avantages axés sur la fiabilité des robots et la flexibilité des humains. Les questions de sécurité et de durabilité, ainsi que les attitudes à l'égard de la collaboration humain-robot, la protection de la santé, la volonté de coopérer au sein d'équipes de travail humain-robot, doivent être prises en compte de manière exhaustive lors de l'analyse de la création d'une base d'informations partagées dans le cadre du concept d'industrie 4.0. Le bien-être des travailleurs est la condition préalable à un processus réussi et durable d'adoption de l'industrie 4.0.

Grâce à la numérisation de la fabrication, un changement majeur dans la façon dont les produits sont fabriqués est constaté (Brandao et Wynn 2008). Ce changement est si convaincant qu'il a été appelé Industrie 4.0, représentant la quatrième révolution industrielle. De la première révolution industrielle (mécanisation grâce à l'énergie hydraulique et à la vapeur) à la deuxième utilisant l'électricité pour la production de masse et les chaînes de montage, la quatrième révolution industrielle tirera parti de l'adoption des ordinateurs et de l'automatisation introduite au début de la troisième révolution industrielle et l'améliorera avec des systèmes intelligents et autonomes alimentés par les données et l'apprentissage automatique (Maslarić, Nikolicic et Mirčetić 2016).

L'industrie 4.0 est basée sur l'interconnexion des composants technologiques individuels et des travailleurs humains communiquant entre eux à un rythme sans précédent. Le concept d'industrie 4.0 est l'introduction de systèmes intelligents reliés en réseau, qui assurent une production autorégulée : les personnes, les machines, les équipements et les produits communiqueront entre eux (Li, et al. 2016). L'industrie 4.0 nécessite des réponses et des solutions pour différents sujets : les hommes et la main-d'œuvre, les exemples d'entreprises et de stratégies, la gestion de l'afflux de données, la cyber sécurité, les normes et l'interopérabilité, etc. (Maslarić, Nikolicic et Mirčetić 2016).

L'industrie 4.0 se caractérise par une utilisation importante des données et par une connectivité accrue. L'activité industrielle ainsi que les conditions de travail des travailleurs ont été impactées par l'utilisation fréquente des nouvelles technologies. Les entreprises sont confrontées à une mutation complète et profonde. La robotisation, l'impression 3D, les interfaces intelligentes représentent quelques-uns des sujets qui obligent à repenser le modèle de fonctionnement des industries.

Comme l'exprime la Figure 1.1, neuf piliers technologiques constituant l'industrie 4.0 sont identifiés (Montaigne 2018):

- ✓ Robots collaboratifs et smart machines : par exemple des bras collaboratifs qui se chargent de fournir et de tenir les pièces lourdes pour les opérateurs humains afin de

faciliter l'assemblage et de leur permettre de les réaliser en toute sécurité et de manière ergonomique.

- ✓ Internet des objets: permettant aux opérateurs, machines et robots d'être connectés et communicants. Ainsi le système de production peut remonter l'état d'avancement des différents agents impliqués dans la production.
- ✓ Big Data, analyse de données et intelligence artificielle : l'analyse de données permet de nettoyer, de transformer et de modéliser les données afin d'extraire des informations utiles et de prendre des décisions.
- ✓ Intégration verticale et horizontale : permettant de faciliter le partage de données entre plusieurs sous-systèmes impliqués dans l'industrie.
- ✓ Outils de simulation avancée : simuler virtuellement des tâches ou des produits avant de les tester réellement via le jumeau numérique limitant ainsi les coûts et les prises de risque.
- ✓ Fabrication additive, matériaux et processus innovants : créer des prototypes grâce à l'impression 3D.
- ✓ Réalité augmentée/virtuelle : pour créer des environnements immersifs pour la formation ou des environnements augmentés pour guider les opérateurs (Havard, et al. 2019).
- ✓ Blockchain : qui est une technologie qui permet de résoudre le problème des doubles enregistrements en gardant la trace d'un ensemble de transactions de manière décentralisée et sécurisée.
- ✓ Cloud et cybersécurité : stocker les données d'une manière sécurisée et dématérialisée.

FIGURE 1.1 - Briques technologiques de L'industrie 4.0 (4.0) (MONTAIGNE 2018)

Notre projet de thèse inclut 4 parmi les 9 piliers de l'industrie 4.0 qui sont : les robots collaboratifs, la simulation avancée à travers le jumeau numérique, la réalité virtuelle et l'Intelligence Artificielle.

Nous allons présenter synthétiquement dans les sections suivantes les 3 premiers piliers, le 4ème pilier portant sur l'IA pour la reconnaissance d'actions humaines étant traité dans une partie dédiée.

I.2.1 Robotique industrielle et collaborative

Cela fait des décennies que les experts ont commencé à examiner les types d'interactions nécessaires pour des échanges souhaitables entre les robots et les humains (J. Scholtz 2003). Ainsi, la création d'équipes humain-robot efficaces et efficientes, tirant parti des compétences de chaque membre de l'équipe, est devenue l'un des objectifs des théories de gestion des systèmes informatiques. D'un point de vue managérial, l'interaction humain-robot diffère des interactions homme-ordinateur et homme-machine déjà connues car « elle concerne des systèmes de contrôle complexes et dynamiques, qui font preuve d'autonomie et de cognition et qui opèrent dans un environnement réel changeant » (J. Scholtz 2003).

La conception centrée sur l'homme dans les interactions humain-robot doit aller au-delà des questions technologiques et prendre en compte des questions telles que la répartition des tâches entre les personnes et les robots, la sécurité et la structure du groupe etc. Ces questions doivent être prises en compte dès les premières étapes de la conception des technologies. Si elles ne sont prises en compte qu'au stade final, ces questions deviennent secondaires et ont peu d'impact sur les considérations de conception (J. C. Scholtz 2002). Les professionnels de l'informatique et les praticiens des facteurs humains, dans le cadre d'une coopération multidisciplinaire, peuvent se concentrer sur l'optimisation du processus d'adaptation à la technologie (dans les intentions d'adaptation à la technologie et l'analyse du comportement d'adaptation à la technologie) ainsi que sur le développement des compétences qui sont importantes et bénéfiques pour travailler dans des équipes hybrides humain-robot.

Les humains et les robots travaillent plus étroitement ensemble à mesure que la technologie progresse. Cela augmente la productivité des entreprises et la qualité des produits, menant à l'efficacité et à la croissance. Dans de nombreux cas, les robots augmentent tellement la production que des emplois complémentaires sont créés. Les chercheurs et les entreprises améliorent la sécurité des systèmes robotiques afin que les humains puissent travailler à proximité de robots devenus collègues, plus que de simples outils. De cette façon, les travailleurs humains peuvent entreprendre des tâches qui nécessitent de la flexibilité, tandis que les robots gèrent des tâches qui tirent le meilleur parti de leur force et de leur vitesse (Maurtua, et al. 2017).

La prochaine étape pour les fabricants de robots, les éditeurs de logiciels et les ingénieurs consiste à affiner encore l'augmentation du nombre d'emplois humain-robot afin d'accroître les gains de productivité et de libérer les personnels des tâches pénibles ou à faible valeur ajoutée.

Cependant, de nombreuses applications industrielles exigent une solution robotique pour

augmenter la productivité lorsque le travailleur humain exécute des tâches pouvant être assistées par un robot, par exemple :

- La tâche est simple et répétée de manière identique à plusieurs reprises, mais doit être précise.
- Nécessite plus d'un opérateur mais pas autant que deux.
- La tâche pourrait causer du stress, voire des blessures à l'opérateur.

Depuis des décennies, les robots remplissent l'une des deux grandes applications : (Hadall 2017)

1. Les grands robots industriels et les véhicules à guidage automatique (AGVs), programmés hors ligne, et qui fonctionnent selon des trajectoires définies pour déplacer, assembler ou souder un composant fabriqué sur une chaîne de production.
2. Des robots domestiques ou de service : Applications plus légères et plus douces, elles aident dans des contextes plus nuancés, allant de l'inspection d'environnements hostiles à l'aide apportée aux humains au quotidien.

Jusqu'à récemment, l'utilité des robots était limitée. Ils effectuent un nombre limité de tâches de manière linéaire, sont programmés hors ligne et ne peuvent pas répondre aux nouveaux stimuli en cours de tâche. Les robots industriels sont lourds, rapides et potentiellement dangereux pour les humains travaillant à proximité. Le défi consiste à modifier les paramètres d'un robot pour le rendre sensible à la présence humaine, à ralentir sa vitesse et sa puissance à proximité de l'homme et à développer une technologie permettant au « robot collaboratif » ou cobot, d'assister l'humain comme un collègue intuitif, et non pas comme une menace lourde et dangereuse (Maurtua, et al. 2017).

Le développement des Cobots ou les « Robots collaboratifs » est né d'une nouvelle application de robots qui prend rapidement racine à l'échelle mondiale : la « Collaboration Humain-robot (HRC) ». Les Cobots sont des robots industriels conçus pour réaliser des applications en interaction étroite avec l'équipe de production. Ils sont plus légers, plus facile à programmer et moins onéreux que les robots industriels classiques. Même si le principe de la robotique collaborative est de fonctionner en binôme « humain/robot », cela ne signifie pas forcément que le cobot travaille en interaction permanente avec l'opérateur. L'opérateur peut ainsi programmer son cobot puis le laisser travailler à côté de lui. Il constitue ainsi un troisième bras. En mode collaboratif, le cobot interagit plus étroitement avec l'humain. Il va par exemple prendre une pièce dans un bac et la donner à l'opérateur dans le bon sens, pour faciliter l'assemblage. Dans tous les cas, le principe du cobot est d'offrir un haut niveau de sécurité pour pouvoir opérer dans la même zone de travail que l'humain. Pour cela, il est équipé de capteurs qui lui permettent de détecter la présence humaine à quelques centimètres afin de s'arrêter automatiquement s'il y a un risque de contact. Le cobot est principalement utilisé pour des tâches répétitives, comme l'assemblage de pièces, le polissage du verre, etc. (El Zaatari, et al. 2019).

Cette notion de collaboration humain-robot peut se décliner en 5 niveaux selon le degré

d'interaction souhaité comme l'explique la Figure 1.2 (Malik et Bilberg 2019) :

- **Isolé :** ce type de collaboration est le plus basique. Dans ce type de collaboration le robot est complètement isolé de l'opérateur et chacun travaille à part dans son environnement.
- **Coexistence :** ici le robot et l'opérateur travaillent dans le même environnement mais sur des pièces différentes.
- **Synchronisation :** le robot et l'opérateur se partagent un même espace et travaillent sur la même pièce l'un après l'autre.
- **Coopération :** le robot et l'opérateur se partagent un même espace et travaillent sur des pièces différentes.
- **Collaboration :** le robot et l'opérateur se partagent un même espace et travaillent simultanément sur la même pièce.

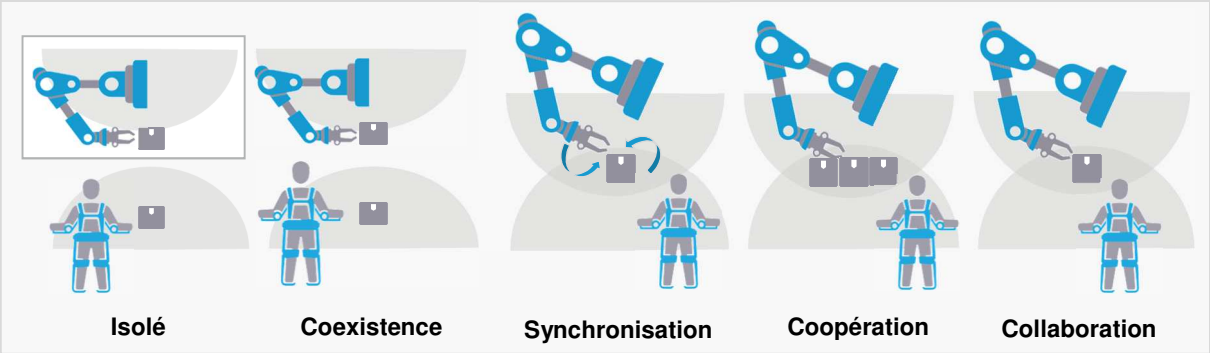


FIGURE 1.2 - Interactions humain-robot (MALIK ET BILBERG 2019)

Les Cobots peuvent être dotés de capteurs de couple/force très sophistiqués afin de détecter si des personnes sont proches d'eux : une personne peut désormais interagir dans le même espace en sachant que le robot s'arrêtera s'il s'immisce dans les paramètres définis et poursuit ensuite ses activités.

Les tâches simples pour un humain peuvent être complexes pour un robot. Par exemple, déterminer à quoi ressemble un objet parmi de nombreux objets identiques ou similaires et comment saisir cet objet est très compliqué pour un robot. Le défi pour un robot devient de plus en plus complexe lorsque plusieurs objets se trouvent dans le même espace, que d'autres objets obscurcissent leur vision et qu'il doit l'aborder sous un nouvel angle. Avec la puissance de calcul actuelle, il est devenu possible d'effectuer de nombreux calculs sur l'apparence de cet objet dans différents scénarios et le robot peut ainsi « apprendre » et « déterminer » les objets qui l'entourent. Aussi, les actions humaines sont souvent fortement corrélées à l'acteur, qui exécute une action spécifique. Comprendre à la fois l'objet, l'acteur ou encore ses actions peut être vital pour les applications de la vie réelle, telles que la navigation des robots et la surveillance des patients (Vrigkas, Nikou et Kakadiaris 2015).

Les progrès de la collaboration entre robots et humains ont conduit à des progrès dans tous les domaines. La HRC implique une productivité accrue, donc une croissance, une répétabilité, un redéploiement de la main-d'œuvre dans de meilleurs emplois, des profits plus importants et une exposition réduite au travail répétitif ou physiquement difficile avec moins de risques liés à la sécurité et l'ergonomie.

Les robots doivent atteindre des niveaux plus élevés d'interaction et de communication avec les humains et comprendre en conséquence leurs comportements ou leurs actions afin de libérer les travailleurs des tâches difficiles, de rendre les systèmes de production plus flexibles et d'améliorer la qualité des produits. Par exemple, les tâches d'assemblage dans les environnements industriels pourraient être beaucoup plus faciles si les robots étaient capables de savoir exactement ce que l'opérateur fait à chaque instant, afin de fonctionner plus efficacement et de transmettre les équipements appropriés à chaque opération de la tâche d'assemblage (Dallel, Havard et Baudry, et al. 2020). Quelques scénarios de la Collaboration Humain-robot dans la littérature extraits de (El Zaatari, et al. 2019) sont donnés par le Tableau 1.1.

TABLEAU 1.1 - Exemples de scénarios de la HRC dans la littérature(EL ZAATARI, ET AL. 2019)

Scénario	**Exemple**	**Tâche de l'humain**	**Tâche du cobot**
Co-manipulation	(Lichiardopol, van de Wouw et Nijmeijer 2009)	L'homme et le cobot tiennent et déplacent tous deux un objet	
		L'homme guide la trajectoire de l'objet	Le cobot gère le poids de l'objet
Assemblage	(Johannsmeier et Haddadin 2017)	Les actions d'assemblage sont réparties entre l'homme et le cobot en fonction de la charge de travail prévue et de l'énergie consommée	
Pick-and-Place	(Gabler, et al. 2017)	L'homme choisit des objets au hasard pour les déplacer	Le cobot choisit les objets à déplacer en tenant compte de la distance, de l'accessibilité et des déplacements prévus de l'opérateur
Contrôle	(El Makrini, et al. 2017)	L'humain visse des boulons dans des trous	Le cobot vérifie si tous les trous sont vissés et émet un avertissement en cas de boulon manquant
Vissage	(Cherubini, et al. 2015)	L'homme insère des boulons dans les trous d'un côté d'une plaque	Le cobot serre les boulons de l'autre côté de la plaque

I.2.2 Jumeaux numériques (DT) et simulations RV dans l'Industrie 4.0

Dans cette section, nous définissons d'abord les jumeaux numériques et ses différentes typologies. Par la suite, nous présentons l'usage du jumeau numérique (DT) couplé à la réalité virtuelle (RV) dans le contexte de la HRC.

I.2.2.1 Définition du Jumeau numérique et typologie

Un DT est décrit comme la numérisation d'un système physique capable de fonctionner sur différentes disciplines de simulation qui sont caractérisées par la synchronisation entre les systèmes virtuels et physiques grâce aux données détectées et aux dispositifs intelligents connectés, aux modèles mathématiques et à l'élaboration de données en temps réel.

Dans (Glaessgen et Stargel 2012), les auteurs donnent une définition plus détaillée du DT en tant que simulation probabiliste multi-physiques et multi-échelles d'un produit complexe et qui utilise les meilleurs modèles physiques disponibles, les mises à jour des capteurs, etc. pour refléter la vie de son jumeau correspondant.

En pratique, il n'existe pas de consensus parfait sur ce qui constitue un jumeau numérique. Ces derniers se présentent sous diverses formes avec de nombreux attributs différents. Toutefois, le DT présente des caractéristiques clés sur lesquelles un consensus apparait. Les attributs qui peuvent définir les jumeaux numériques et aider à les comprendre et à les différencier d'autres types de simulation ou de modèles informatiques sont les suivants :

- Un jumeau numérique est un modèle virtuel d'un objet/système réel/physique.
- Un jumeau numérique simule à la fois l'état physique et le comportement de l'objet/système.
- Un jumeau numérique est associé à une instance spécifique de l'objet/système.
- Un jumeau numérique est connecté à l'objet/système, et est mis à jour en fonction des changements connus de l'état, de la condition ou du contexte de l'objet/système.
- Un jumeau numérique fournit une valeur ajoutée par la visualisation, l'analyse, la prédiction ou l'optimisation.

En ce qui concerne l'industrie, le DT est un sujet qui intéresse de plus en plus (Fuller, et al. 2020) ; celui-ci désigne la réplique virtuelle d'un produit ou d'un processus à l'aide de différents capteurs et repose sur de grandes quantités de données (Liu, et al. 2020). En utilisant ces données en temps réel qui permettent l'apprentissage, le raisonnement et le recalibrage, cette représentation virtuelle du système physique ou d'un processus opérationnel est utilisée pour comprendre ou prédire l'homologue physique en tirant parti à la fois des données du système et de son expérience du monde physique pour améliorer la prise de décision (Negri, Fumagalli et Macchi 2017) (Lee, Bagheri et Kao 2015).

La représentation numérique doit inclure toutes les informations concernant le système physique. Le DT inclut également les modèles qui décrivent leurs homologues réels. Sur la base de ces données traitées, il peut ensuite décider des actions à entreprendre (Havard, et al. 2019).

Ce concept apporte de nouveaux niveaux de performance pour l'analyse des jeux de données qui s'accroissent en permettant de construire des modèles optimaux. Ils permettent également de tester virtuellement des produits ou des processus à un coût bien plus faible que lorsque cela est

fait physiquement, c'est-à-dire sur le produit réel. De plus, le DT collecte des données en temps réel et le système physique peut donc être représenté à tout moment. Des algorithmes d'apprentissage profond sont alors déployés, permettant par exemple de détecter les éventuels besoins de maintenance, et donc de fournir des alertes précoces pour prévenir les défaillances ou les pannes, comme dans (Grieves et Vickers 2017) (Franciosa, et al. 2020). Le DT peut largement bénéficier aux humains puisqu'il peut être utilisé dans de nombreux domaines tels que la HAR qui elle-même peut être utilisée dans plusieurs domaines tels que les industries de production ou les établissements de santé (Dallel, Havard et Baudry, et al. 2020).

Aujourd'hui, les DTs sont de plus en plus utilisées dans différents domaines tels que le dépannage et les réparations à distance, la formation des nouveaux employés, les inspections sur site, etc. car elles aident les entreprises et les industries à mieux comprendre les solutions et les dispositifs utilisés et donc à les améliorer, tout cela d'une manière sûre, efficace et rentable (Aivaliotis, et al. 2019).

Le rôle actuel dans les systèmes de fabrication de l'industrie 4.0 est d'exploiter ces caractéristiques pour prévoir et optimiser, en temps réel, le comportement du système de production à chaque phase du cycle de vie. Il existe de multiples solutions existantes du DT, c'est pourquoi une compréhension différente existe. Pour mieux comprendre le concept du DT, ses niveaux d'intégration (Tao, et al. 2018) sont examinés dans l'Annexe - Typologie de jumeau numérique.

I.2.2.2 Jumeau numérique et réalité virtuelle dans un contexte de HRC

Dans l'article (Burghardt, et al. 2020), les auteurs présentent une méthode dans laquelle ils programment des robots en utilisant la réalité virtuelle et les jumeaux numériques. Ils ont construit un jumeau numérique d'une station robotique en utilisant des modèles CAO d'éléments existants de station industrielle. L'objectif de ce système de réalité virtuelle était d'enregistrer les mouvements humains dans un environnement virtuel qui sont reproduits ultérieurement par un robot réel. Cette méthode a été développée pour faciliter les situations dans lesquelles le robot doit reproduire les mouvements de l'homme tout en réalisant un processus complexe.

Dans (Havard, et al. 2019), les auteurs ont conçu un jumeau numérique d'un système de production cobotique et ont proposé une architecture de co-simulation et de communication entre le jumeau numérique et une application de réalité virtuelle. La proposition permet de concevoir un poste cobotique et de l'étudier en termes de sécurité et d'ergonomie tout en conservant un comportement réaliste du bras cobotique grâce au jumeau numérique de ce dernier. Le système de fabrication flexible et son jumeau numérique sont donnés par la Figure 1.3.

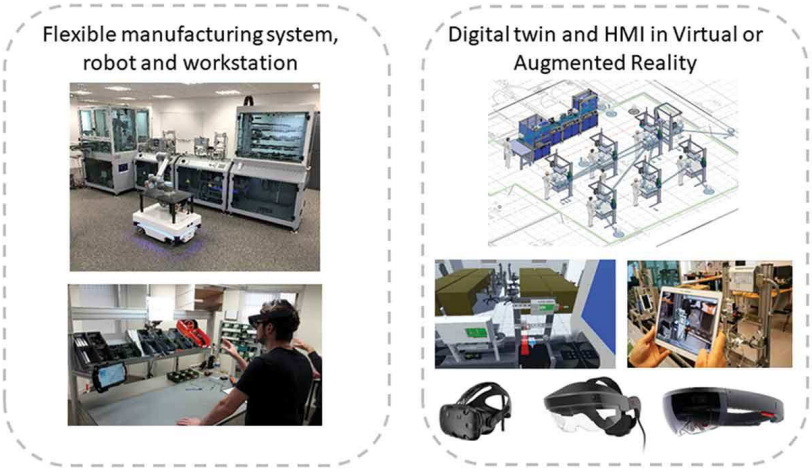


FIGURE 1.3 - Système de fabrication flexible et son jumeau numérique (HAVARD, ET AL. 2019)

Dans (Malik et Bilberg 2018), les auteurs ont présenté un framework de jumeau numérique pour faciliter la conception, la construction et le contrôle de la collaboration humain-robot. Ils ont utilisé des simulations informatiques afin de développer un jumeau numérique d'un environnement de travail collaboratif humain-robot pour une tâche d'assemblage. Le système numérique développé est mis à jour pendant tout le cycle de vie du système de production en représentant régulièrement le système physique intégré pour des améliorations continues. Pour valider et développer leur framework, ils ont présenté le cas d'une entreprise de fabrication avec des équipes de travail humain-robot.

Dans (Matsas et Vosniakos 2017), les auteurs ont présenté un système de formation interactif et immersif en réalité virtuelle sous forme d'un jeu sérieux dans lequel une coopération en temps réel entre des manipulateurs robotiques industriels et des humains est simulée dans le but d'exécuter des tâches de fabrication. Le scénario implique la manipulation collaborative pour la pose de rubans adhésifs utilisés pour construire des pièces composites aérospatiales. L'objectif de cet article était d'étudier l'expérience et le comportement des utilisateurs dans l'environnement virtuel tout en coopérant avec le robot. Le système peut ensuite être utilisé pour valider l'acceptabilité de la collaboration humain-robot dans un environnement de travail.

Dans (Pérez, et al. 2020), les auteurs ont présenté une nouvelle méthode pour concevoir l'automatisation des processus, mettre en œuvre et surveiller en temps réel un ensemble d'opérations en créant un jumeau numérique d'un processus de fabrication avec une interface de réalité virtuelle immersive utilisée comme banc d'essai virtuel avant la mise en œuvre physique. La méthodologie peut être utilisée pour former les opérateurs, surveiller en temps réel et étudier la faisabilité de futures optimisations. Afin de valider leur méthode, les auteurs ont utilisé leur méthode sur un cas d'utilisation réel fournissant une solution pour un processus de fabrication d'assemblage.

Notre travail de recherche s'inscrit dans le contexte de l'industrie 4.0 qui vise à développer de nouvelles approches pour la détection et la reconnaissance d'actions humaines dans les milieux industriels pour faciliter la collaboration humain-robot en s'appuyant sur le jumeau numérique. Nous nous intéressons plus spécifiquement dans la suite de cet état de l'art au pilier lié à l'IA pour la reconnaissance d'actions humaines.

I.3 Reconnaissance et détection d'actions humaines

De nos jours, les industries manufacturières font appel à la fois à des humains et à des robots d'une grande complexité pour améliorer le rendement et la qualité de leurs produits (El Zaatari, et al. 2019). Les capacités de perception et de décision des robots évoluant progressivement dans des environnements plus peuplés d'humains et notamment vers des situations nécessitent une collaboration humain-robot dans les environnements de travail industriels. Par conséquent, outre le fait que les robots doivent fonctionner de manière sûre et compétente, ils doivent également être capables d'interagir et de communiquer avec les humains.

Afin d'aider l'humain à atteindre un ensemble d'objectifs, il appartient au robot d'estimer son intention et d'agir en conséquence. Une personne peut communiquer son intention soit délibérément par une communication explicite, soit implicitement par des actions. Les manières de communiquer l'intention telles qu'elles sont élaborées dans (Bauer, Wollherr et Buss 2008) sont présentées dans le Tableau 1.2.

TABLEAU 1.2 - Principaux moyens de communiquer les intentions (BAUER, WOLLHERR ET BUSS 2008)

Communication d'intention	Discours	Informations explicites
		Émotions
	Geste	Tête/yeux
		Gestes communicatif
	Action	Gestes de manipulation
		Exécution proactive des tâches
	Signal haptique	Force/Couple
		Angles/Orientation
	Signal physiologique	Approbation

La reconnaissance d'actions a attiré l'attention des chercheurs ces dernières années (Li, Liu, et al. 2020) (Kamel, et al. 2019) (Song, Lan, et al. 2018). Les méthodes actuelles sont principalement conçues pour la reconnaissance d'actions segmentées, c'est-à-dire que le type d'action est reconnu une fois que la séquence d'action entière est observée. Cette méthode est également appelée reconnaissance d'action hors ligne. Néanmoins, il serait souhaitable que la reconnaissance d'action se fasse pendant l'exécution de l'action ce qui est nécessaire pour des applications en temps réel. Les travaux récents s'intéressent plutôt à ce type de reconnaissance d'actions appelé aussi reconnaissance en ligne vu leurs larges champs d'applications dans plusieurs domaines comme la surveillance vidéo, les interactions homme-machine, etc. (C. Liu, Y. Li, et

al. 2017) (Tang, et al. 2018).

Les mouvements humains sont classés en trois types différents : les gestes, les actions et les activités (Aggarwal et Ryoo 2011) (Devanne, et al. 2014) (Turaga, et al. 2008).

- **Gestes :** Les gestes sont les éléments élémentaires décrivant le mouvement significatif d'une personne, et sont perceptibles visuellement par les humains et qui sont facilement annotés. Ce type de mouvement n'implique qu'une partie du corps humain, comme le bras, la tête ou la jambe, etc. En général, les gestes sont très brefs et ne durent que quelques secondes. De plus, ils sont réalisés sans l'utilisation d'un objet quelconque. Exemples de gestes : agiter un bras, lever un pied, lever un bras, etc.
- **Actions :** Les actions sont définies comme une séquence de gestes organisée temporellement. Ainsi, une telle variété de mouvements peut inclure le mouvement de plusieurs parties du corps, contrairement aux gestes. La durée des actions est évidemment plus importante que la durée du geste et peut durer jusqu'à une minute. Les actions peuvent impliquer un objet, mais celui-ci doit être présent du début à la fin de l'action. Exemples d'actions : sauter, marcher, courir, nager, balancer une balle de golf, lancer un objet, etc.
- **Activités :** Les activités représentent le plus haut niveau de mouvement et sont composées d'une séquence d'actions. Ils impliquent des interactions avec des objets, ce qui les rend plus complexes à comprendre et à reconnaître. De plus, la connaissance du contexte est parfois nécessaire pour évaluer correctement l'activité. Leur durée est relativement importante par rapport aux actions et aux gestes (de l'ordre de quelques minutes).
  Exemples d'activités : parler au téléphone, manger une pomme, jouer à des jeux vidéo, etc.

I.3.1 Reconnaissance d'actions segmentée/en ligne

Le but du problème de la reconnaissance d'action est de prédire l'étiquette de l'action sur la base d'une série temporelle de données. La tâche de classification des actions consiste à prédire pour chaque vidéo la présence ou l'absence de chacune des classes d'actions prédéfinies de l'ensemble de données. Il s'agit d'une tâche de classification binaire par action, car les actions ne sont pas mutuellement exclusives - une action donnée peut se produire une fois, plusieurs fois ou jamais dans une vidéo de test. Cela contraste avec la tâche multi-classes typique à choix forcé dont le but est d'attribuer une étiquette de classe à une vidéo donnée à partir d'un ensemble de classes prédéfinies. Pour la tâche de classification, des valeurs réelles de confiance pour chaque vidéo de test pour les différentes actions sont fournies. Une confiance faible pour une action particulière signifie que la vidéo contient une autre action ou aucune des classes des actions.

Il y a deux types de reconnaissance d'actions : segmentée et en ligne (Li, et al. 2016) (Dallel, Havard et Dupuis, et al. 2022):

- La reconnaissance segmentée : vise à, compte tenu d'une vidéo segmentée $V_{seg} = \{f_1, \dots, f_N\}$ avec N frames, déterminer si une image f_t à l'instant t correspond à une action parmi les K classes prédéfinies. Par conséquent, l'étape de reconnaissance d'action est effectuée après l'observation de la séquence d'action complète. Ainsi, déterminer l'étiquette de l'action à chaque intervalle de temps revient à maximiser sa probabilité postérieure qui peut être formulée comme suit :

$$y^* = \underset{y}{argmax}\, P\left(y \middle| V_{seg}\right) \tag{1.1}$$

- La reconnaissance en ligne : contrairement à la reconnaissance d'actions segmentée, qui utilise la séquence vidéo complète pour déterminer l'étiquette de l'action, la reconnaissance en ligne vise à détecter une action au sein d'une longue séquence d'actions à la volée, le plus tôt possible, c'est-à-dire sans utiliser de futures informations. La reconnaissance en ligne peut être formulée comme suit :

$$y_t^* = \underset{y_t}{argmax}\, P(y_t | f_1, \dots, f_t) \tag{1.2}$$

Les reconnaissances d'actions segmentée et en ligne sont illustrées dans la Figure 1.4.

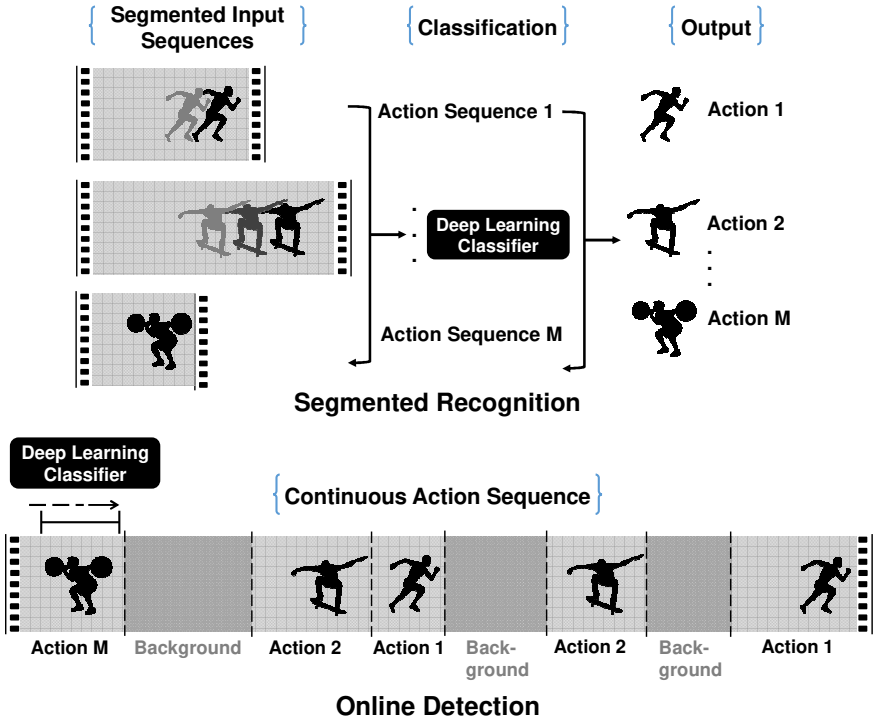


FIGURE 1.4 - Reconnaissance d'actions segmentée/en ligne (DALLEL, HAVARD ET DUPUIS, ET AL. 2022)

Dans sa forme générale, la HAR est un problème multi-classes avec c classes « intéressantes » plus une classe « NULL / No action ». Cette dernière comprend toutes les parties du signal où aucune activité pertinente n'a eu lieu.

Le problème de reconnaissance d'action peut être étendu au problème de détection d'action

qui présente deux défis principaux. Outre le fait d'avoir un coût de calcul plus élevé, elle doit également reconnaître l'action en cours, tout en étant capable de la segmenter, ce qui signifie détecter le début et la fin de chaque action. Dans de nombreux scénarios et cas d'applications pratiques, l'aspect temporel est plus important que l'aspect spatial qui permet seulement de situer un objet/action dans une image/vidéo, généralement par des boites englobantes (Bounding Box) encadrant l'objet/action en question. Il est plus bénéfique de connaitre le label de l'action ainsi que son temps de début et de fin (détection temporelle) que de la localiser au sein d'un image/vidéo (détection spatiale). Dans ce problème, le but est de prévoir tous les labels pour une vidéo donnée, ainsi que le moment où l'action commence et se termine. Pour la tâche de détection temporelle, des intervalles temporels et des valeurs de confiance correspondantes pour toutes les instances détectées de classes d'actions présélectionnées sont fournis. La tâche de classification est incluse dans la détection temporelle, ce qui la rend comparativement plus complexe. Par exemple, une instance d'une action qui est correctement localisée dans le temps mais à laquelle est attribuée un label de classe incorrect sera traitée comme une détection incorrecte.

Nous nous intéressons dans la section suivante aux capteurs utilisés pour la reconnaissance d'actions humaines.

I.3.2 Capteurs et modalités d'acquisition des données

Les capteurs utilisés pour la reconnaissance d'actions se divisent en deux branches : Capteurs basés sur la vision et les capteurs portables. Quelques exemples de ces capteurs sont donnés par la Figure 1.5 (Liu et Wang 2018).

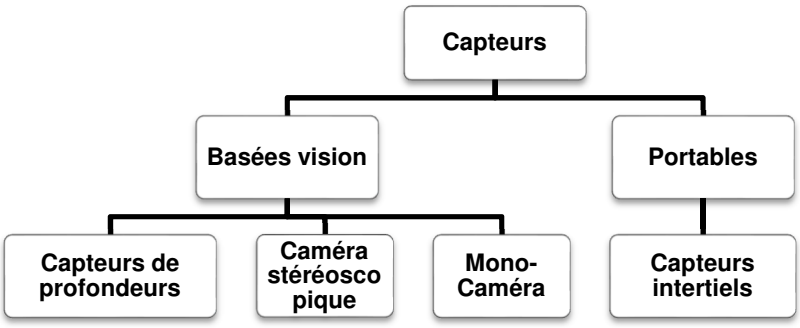


FIGURE 1.5 - Exemples de capteurs utilisés pour la reconnaissance d'actions (LIU ET WANG 2018)

Les capteurs intelligents sont les moteurs de l'Industrie 4.0 et de l'Internet des objets industriels (IIoT) dans les usines. Une fois mise en œuvre, la combinaison de capteurs sophistiqués et d'une puissance de calcul accrue permettra de trouver de nouvelles façons d'analyser les données et d'obtenir des informations exploitables pour améliorer de nombreux domaines d'activité. Il en résultera des processus de production réactifs et agiles qui garantissent et améliorent les performances dans toute une série de secteurs industriels.

Bien que, l'impact des capteurs sur les performances des entreprises reste un potentiel important, principalement par le biais de la réduction des coûts, leur installation au sein des milieux industriels est soumise à des contraintes pouvant prévenir les industries de les déployer largement. Plus spécifiquement si l'on s'intéresse aux ilots de production et postes de travail, l'installation de capteurs est contrainte par des problématiques spatiales de couvertures, de masquages, de performance, de modification de l'infrastructure, de caractère intrusif ou encore d'éthique (P. Wang, W. Li et P. Ogunbona, et al. 2018). Pour prendre en compte ces contraintes, nous devons donc nous orienter vers un système robuste, basé sur des capteurs qui peuvent être utilisés dans un milieu industriel sans contraintes d'éthique.

Les techniques classiques de HAR basées sur l'apprentissage automatique utilisent des caractéristiques extraites à la main (hand-crafted) et peuvent être classées selon la modalité de données utilisée (RGB, Profondeur, Squelettes et Multimodales) en utilisant un ou plusieurs capteurs présentés précédemment par la Figure 1.5.

I.3.2.1 La modalité RGB

La modalité RGB fait généralement référence à des images ou des vidéos (séquences d'images) capturées par des caméras RGB qui visent à recréer ce que l'œil humain voit. Les données RGB sont généralement faciles à collecter et elles contiennent des informations d'apparence riches du contexte de la scène capturée. La HAR basée sur des données RGB a un large éventail d'applications, telles que la surveillance visuelle, la navigation autonome, etc. (Liu et Wang 2018) (Caetano, et al. 2019). Cependant, la reconnaissance d'actions à partir des données RGB est souvent difficile, en raison de la grande variabilité des formes ou des postures humaines, l'encombrement ou les variations des arrière-plans posant parfois des difficultés de distinction entre les premiers et les arrières plans ou encore en raison des changements des points de vue dans ces données. De plus, les caméras peuvent être sensibles aux changements d'éclairage, aux occlusions et aux changements d'orientation. Par conséquent, pour être efficaces et performantes, les méthodes basées sur les données RGB nécessitent d'énormes quantités de données et de traitements. En outre, les vidéos RGB ont généralement de grandes tailles de données, ce qui entraîne des coûts de calcul élevés lors de la modélisation du contexte spatio-temporel pour la HAR ce qui rend les méthodes basées sur les données RGB une tâche fastidieuse.

I.3.2.2 La modalité squelette

Les données squelettiques codent les trajectoires des articulations du corps humain, qui caractérisent les mouvements humains informatifs. Par conséquent, les données squelettiques sont également une modalité appropriée pour la HAR. Ces données peuvent être acquises en utilisant des capteurs inertiels. Les données du squelette peuvent également être obtenues en appliquant des algorithmes d'estimation de pose sur des vidéos RGB ou des cartes de profondeur. Des outils comme OpenPose (Cao, Hidalgo, et al. 2018) ou AlphaPose (Xiu, et al. 2018) peuvent être utilisés

pour extraire des données squelettes à partir des flux RGB en temps réel, ce qui permettra des estimations de pose rapides et précises (Yan, Xiong et Lin 2018). En général, l'estimation de la pose humaine est sensible aux variations du point de vue. Les données de squelettes peuvent également être collectés avec des systèmes de capture de mouvement qui sont insensibles à la vue et à l'éclairage et peuvent fournir ainsi des données squelettes fiables.

Les approches basées sur les données squelettes 2D/3D sont devenues plus populaires ces dernières années en raison des avantages qu'elles présentent par rapport aux données RGB conventionnelles (Cheng, et al. 2020) (Dallel, Havard et Dupuis, et al. 2022). Ces données sont de plus en plus utilisées en raison de la structure corporelle du squelette et de ses informations de pose fournies, de sa représentation essentiellement simple et informative, de son invariance d'échelle et de sa robustesse contre les variations de textures et d'arrière-plans encombrés contrairement aux données RGB. Les méthodes basées sur les squelettes sont fiables pour estimer la silhouette du corps humain plus facilement et n'ont pas non plus besoin d'une quantité énorme de données pour effectuer la tâche de reconnaissance (Yan, Xiong et Lin 2018). En raison de ces avantages et également de la disponibilité de capteurs de profondeur précis et peu coûteux, la HAR basée sur les données squelettes a récemment attiré beaucoup d'attention dans la communauté des chercheurs.

I.3.2.3 La modalité profondeur

Les méthodes basées sur la profondeur reposent sur l'extraction de descripteurs de caractéristiques extraits à la main (hand-crafted) et représentatifs à partir des ensembles de points dans l'image/vidéo de profondeur (voir Tableau 1.3). Les cartes de profondeur font référence à des images où les valeurs de pixels représentant les informations de distance d'un point de vue donné aux points de la scène. La modalité de profondeur, qui est souvent robuste aux variations de couleur et de texture, fournit des informations 3D/2D fiables sur la forme structurelle et géométrique des sujets humains, et peut donc être utilisée pour la HAR. Différents types de capteurs ont été développés pour obtenir des images de profondeur, qui incluent des capteurs actifs (par exemple, des caméras à temps de vol et à lumière structurée) et des capteurs passifs (par exemple, des caméras stéréoscopiques). Des capteurs actifs émettent un rayonnement dans la direction des objets/sujets de la scène, puis mesurent l'énergie thermique réfléchie par les objets pour acquérir les informations de profondeur. En revanche, les capteurs passifs mesurent l'énergie thermique émise ou réfléchie par les objets/sujets de la scène (P. Wang, W. Li et C. Li, et al. 2016).

Les caractéristiques extraites à la main (hand-crafted) des cartes de profondeur peuvent être superficielles et également dépendantes du jeu de données. Ainsi, modéliser la dynamique d'une action est un enjeu important qui peut être résolu différemment selon les approches. Les caractéristiques du squelette peuvent être extraites des cartes de profondeur et conçues pour

capturer à la fois des configurations spatiales et temporelles, mais ces caractéristiques sont extraites à la main (hand-crafted), ce qui les rendent moins expressives ce qui entraîne des difficultés de généralisation.

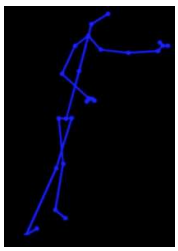
I.3.2.4 Multimodales

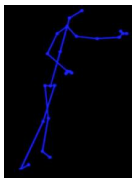
Les solutions hybrides combinent les informations d'au moins deux modalités de données (Squelette, RGB ou Cartes de profondeur) pour modéliser les actions. Ils visent à traiter et à relier les informations sensorielles de plusieurs modalités. En agrégeant les avantages et les capacités de diverses modalités de données, l'apprentissage multimodal peut souvent fournir une HAR plus robuste et plus précis et donc améliorer sa performance (C. Liu, Y. Li, et al. 2017).

I.3.2.5 Comparaison des modalités de données pour la HAR

Une comparaison entre les différentes modalités de données utilisées pour la HAR est donnée par le Tableau 1.3.

TABLEAU 1.3 - Avantages et inconvénients des modalités de données utilisées pour la HAR

Modalité	Exemples	Avantages	Inconvénients
RGB	 Ex : Boire extrait de (LI, ET AL. 2016)	+ Fournit des informations d'apparence riches + Facile à acquérir et à utiliser + Large éventails d'applications	- Sensible aux points de vue - Sensible aux arrière-plans - Sensible à l'éclairage - Problèmes de confidentialité
Squelette	 Ex : Ecrire extrait de (Li, et al. 2016)	+ Fournit des informations structurelles 3D de la pose du sujet humain + Simple mais informatif + Insensible aux points de vue + Insensible aux arrière-plans	- Manque d'informations sur l'apparence - Manque d'informations détaillées sur la forme - Manque de précision lié aux signaux bruités
Profondeur	 Ex : Saluer extrait de (Xia, Chen et Aggarwal 2012)	+ Fournit des informations structurelles 3D + Fournit des informations sur la forme géométrique + Peu sensible aux arrières plans	- Manque d'informations sur la couleur et la texture - Distance de travail limitée

Modalité	Exemples	Avantages	Inconvénients
Multimodales	 +  Ex : Boire extrait de (J. Liu, et al. 2019)	+ Agrège des caractéristiques qui sont souvent très discriminantes et puissantes pour la HAR	- Problèmes de sur-apprentissage

I.3.2.5.1 Choix du type de données d'entrée

Ce défi est l'un des plus fondamentaux, car le type de données d'entrée détermine de manière décisive les capacités de reconnaissance. L'entrée RGB, par exemple, fournit beaucoup d'informations sur la texture et elle semble être proche des informations d'entrée traitées par les humains. D'autre part, le RGB encode également beaucoup d'informations qui ne sont pas importantes pour la reconnaissance d'actions comme les vêtements, les arrière-plans etc. Par exemple, l'apparence RGB dépend fortement des conditions d'éclairage. Pendant le processus de reconnaissance, la représentation préférable devrait être invariante à l'illumination. Dans ce cas, les méthodes qui travaillent avec des données RGB doivent trouver des régularités ou des informations pertinentes et les distinguer d'autres non pertinentes, telles que les changements d'éclairage. Par ailleurs, les données de profondeur sont exemptes de certains problèmes que nous rencontrons avec les données RGB. Par exemple, la modalité profondeur est invariante aux changements d'éclairage, et il est plus facile de distinguer la scène en avant de celle en arrière-plan en plus elle fournit des informations 3D sur la scène. Néanmoins, la modalité de profondeur a aussi certaines limites : elle ne fournit pas d'informations sur la texture et introduit généralement beaucoup de bruit de mesure (ceci concerne particulièrement les capteurs à faible coût, comme Microsoft Kinect, Asus Xtion). D'autre part, le squelette humain est le dernier type de données d'entrée. Ce type de données est généralement obtenu en appliquant différentes méthodes (Wei, et al. 2016) (Shotton, et al. 2013) (Pishchulin, et al. 2016) (Cao, Simon, et al. 2017) soit sur des données RGB, soit sur des données de profondeur. Ainsi, le squelette peut renseigner des informations et des caractéristiques de haut-niveau. D'après (Johansson 1973), les humains sont capables de reconnaître de nombreuses actions en se basant uniquement sur les informations de squelette.

Vu les avantages que renferment les données squelettiques, dans les travaux menés dans cette thèse nous allons nous concentrer principalement que sur ces données.

I.3.2.5.2 Modélisation des caractéristiques

Le problème ici est le niveau de granularité des caractéristiques (niveau de détails). Certains auteurs (Bobick et Davis 2001) (Gorelick, et al. 2007) (Simonyan et Zisserman 2014) proposent de modéliser l'entrée dans son ensemble (méthodes holistiques). D'autres auteurs (Yang et Tian

2012) (Ohn-Bar et Trivedi 2013) (Halim, et al. 2016) (Ellis, et al. 2013) proposent d'utiliser une segmentation détaillée (méthodes du squelette). Enfin, certains auteurs (Laptev 2005) proposent de rechercher les points d'intérêt saillants (méthodes des caractéristiques locales).

I.3.2.5.3 Modélisation du mouvement de l'action

Le mouvement de l'action semble être l'information la plus importante dans la reconnaissance d'action. De nombreux auteurs ont proposé différentes manières de modéliser le mouvement dans la reconnaissance d'action. Certains auteurs ont proposé de concevoir des caractéristiques de bas niveau (par exemple, le flux optique (Simonyan et Zisserman 2014)) ou des caractéristiques de niveau supérieur (par exemple, des trajectoires (P. Wang, W. Li et C. Li, et al. 2016)) et de les utiliser ensuite avec des classifieurs qui ignorent l'aspect temporel des données. D'autres auteurs (Caetano, et al. 2019) ont proposé d'utiliser des classifieurs pouvant modéliser des données séquentielles (Hidden Markov Model - HMM, Conditional Random Field - CRF, Recurrent Neural Network - RNN) et modéliser ainsi le mouvement dans des vidéos.

I.3.2.5.4 Conception des caractéristiques

Il y a deux manières courantes d'aborder ce défi. Nous pouvons soit utiliser des connaissances expertes qui nous amèneraient à concevoir des caractéristiques extraites à la main (hand-crafted). D'autre part, nous pouvons appliquer une méthode qui trouverait automatiquement des caractéristiques basées sur des données d'entrée. Les progrès récents menés en apprentissage profond montrent que la deuxième approche conduit à des performances supérieures à celles des caractéristiques extraites à la main (Qiao, et al. 2017).

I.3.3 Métriques d'évaluation

Afin de comparer les différentes approches de HAR, il est nécessaire d'utiliser des métriques communes. Cette section présente les métriques utilisées par la communauté scientifique. Une méthodologie valable pour l'évaluation des performances doit répondre à deux critères de base (Huang, Fei-Fei et Niebles 2016) :

1. Elle doit être objective et sans ambiguïté. Le résultat d'une évaluation ne doit pas dépendre d'hypothèses ou de paramètres arbitraires.

2. Elle doit non seulement noter, mais aussi caractériser les performances. Lors de la comparaison de systèmes, la méthode doit donner plus qu'une décision binaire, telle que « A est meilleur que B ». Elle doit plutôt quantifier les forces et les faiblesses de chacun d'eux et donner au concepteur du système des indications sur la manière d'apporter des améliorations.

Les métriques d'évaluation utilisées dans la HAR peuvent ne pas refléter correctement les

performances du système de HAR utilisé. Par exemple, les mesures existantes ne révèlent pas si une activité a été fragmentée en plusieurs activités plus petites, si plusieurs activités ont été fusionnées en une seule grande activité, ou s'il existe des décalages temporels dans la reconnaissance d'une activité. Cela peut conduire à une présentation des résultats qui peut être confuse, voire trompeuse (Kukleva, et al. 2019) (Li, Lei et Todorovic 2019) (VidalMata, Scheirer et Kuehne 2020).

Les mesures de performance sont généralement calculées en trois étapes. Tout d'abord, une comparaison est faite entre la sortie du système renvoyée et la vérité terrain. À partir de cette comparaison, un score est attribué aux correspondances et aux erreurs. Enfin, ces scores sont résumés par un ou plusieurs indicateurs, généralement exprimés sous forme de taux ou de pourcentage normalisé. Deux unités de comparaison de base sont généralement utilisées ; les images ou les événements (Li, Lei et Todorovic 2019) :

- Images : une image est une unité de temps de longueur fixe et de débit fixe. Il s'agit souvent de la plus petite unité de mesure définie par le système (la fréquence d'échantillonnage) et, dans ce cas, il se rapproche du temps continu. En raison de la one-to-one mapping entre la vérité terrain et la sortie, la notation des frames est triviale, les frames étant affectées à l'une des catégories suivantes : vrai positif (TP), vrai négatif (TN), faux positif (FP) ou faux négatif (FN).

- Evénements : un événement est défini comme une séquence de durée variable de frames positives dans une série temporelle continue. Il a un temps de début et un temps de fin. Étant donné une séquence de test de g événements connus, $E = \{e_1, e_2, \dots e_g\}$, une reconnaissance produit h événements de retour, $R = \{r_1, r_2, \dots r_h\}$. Il n'y a pas nécessairement de relation one-to-one entre E et R. Une comparaison peut être faite en utilisant des moyens alternatifs : par exemple « Dynamic Time Warping - DTW » qui mesure la similarité entre deux suites qui peuvent varier au cours du temps, la mesure de la plus longue sous-séquence commune, ou une combinaison de différentes transformations. Un événement peut ensuite être noté comme étant soit correctement détecté, soit faussement inséré, lorsqu'il n'y a pas d'événement correspondant dans la vérité terrain, soit supprimé, lorsqu'il n'y a pas eu de détection d'un événement.

Les métriques d'évaluation communément recommandées comprennent :

- Accuracy : L'accuracy est une métrique qui décrit généralement les performances du modèle pour toutes les classes. Elle est utile lorsque toutes les classes sont bien distribuées. L'accuracy est calculée comme le rapport entre le nombre de prédictions correctes et le nombre total de prédictions (voir Figure 1.6). Un vrai positif (TP) est le résultat où le modèle prédit correctement la classe positive. Un vrai négatif (TN) est le résultat où le modèle prédit correctement la classe négative. Un faux positif (FP) est le résultat où le

modèle prédit incorrectement la classe positive. Et un faux négatif (FN) est le résultat où le modèle prédit incorrectement la classe négative. L'Accuracy est donnée par :

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{1.3}$$

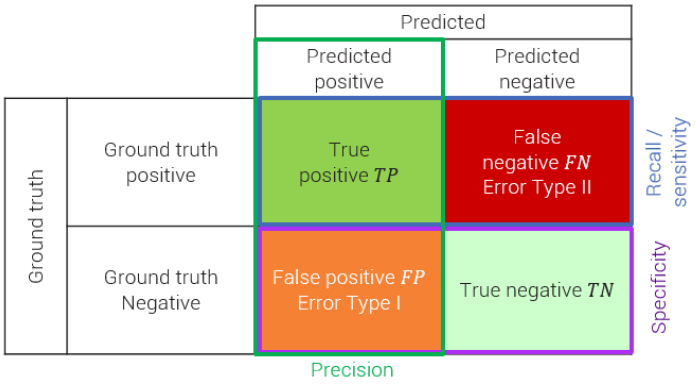


FIGURE 1.6 - Illustration d'une matrice de confusion

- Précision : La précision est le rapport entre les vrais positifs et tous les positifs prédits. Elle mesure l'exactitude et la capacité du modèle à ne pas faire des erreurs de prédiction. La précision peut être écrite comme suit :

$$Précision = \frac{TP}{TP + FP} \tag{1.4}$$

- Rappel ou sensibilité : Le rappel ou la sensibilité est le rapport entre le nombre de prédictions positives et tous les exemples positifs du jeu de données. Il mesure la capacité du modèle à détecter les exemples positifs. Un rappel élevé signifie qu'il y a un grand nombre d'échantillons détectés comme positifs. Le rappel est défini comme suit :

$$Rappel = \frac{TP}{TP + FN} \tag{1.5}$$

- Spécificité : La spécificité est définie comme la proportion de négatifs réels, qui ont été prédits comme négatifs (ou vrais négatifs). Cela implique qu'il y aura une autre proportion de négatifs réels, qui ont été prédits comme positifs (ou faux positifs). Cette métrique peut également être appelée taux de faux positifs (FPR False Positive Rate). La spécificité est donnée par :

$$Spécificité = FPR = \frac{TN}{TN + FP} \tag{1.6}$$

- F1-Score : Le F1-score mesure la précision d'un modèle sur un ensemble de données. Il

est utilisé pour interpréter la moyenne pondérée de la précision et du rappel. L'avantage d'utiliser le F1-score est qu'il incorpore à la fois la précision et le rappel en une seule métrique. Un F1-score élevé signifie que le modèle est performant, même dans des situations où les classes d'action peuvent être déséquilibrées. Le F1-Score est défini comme suit :

$$F1\text{-}Score = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall} \tag{1.7}$$

- Intersection-over-Union (IoU) : est utilisé pour déterminer si un intervalle prédit pour une action est correct ou non (voir Figure 1.7). L'intervalle prédit est considéré comme correct lorsqu'un rapport de chevauchement, IoU, entre la prédiction I et l'étiquette de vérité terrain I^* dépasse un seuil défini (Li, et al. 2016). L'illustration de la Figure 1.7 montre un IoU pour des images 2D, cependant l'IoU est aussi utilisé dans les séries temporelles. L'IoU est donné par :

$$IoU = \frac{|I \cap I^*|}{|I \cup I^*|} \tag{1.8}$$

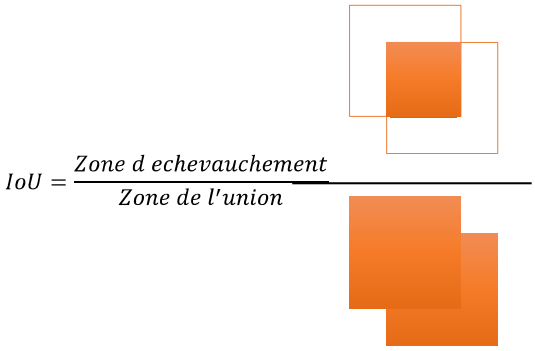


FIGURE 1.7 - Illustration de l'Intersection-over-Union (IoU)

- Mean-over-Frames (MoF) : la MoF est le pourcentage moyen de frames correctement prédites (Li, Lei et Todorovic 2019). La MoF ne convient pas aux jeux de données qui sont déséquilibrés.

- Mean-over-Classes : La MoC est calculée en évaluant la précision de chaque classe d'action, image par image, puis en faisant la moyenne sur le nombre total de classes d'action de base (Kuehne, Richard et Gall 2016).

- Mean Average Precision (mAP) : La mAP c'est la moyenne des précisions moyennes pour chaque action (Xu, et al. 2018). Pour $AP(k)$ étant la précision moyenne pour l'action k et N le nombre d'actions, mAP est définie comme suit :

$$mAP = \frac{\sum k\, AP(k)}{N} \tag{1.9}$$

- Latency rate : La Latency correspond au délai entre le début de l'action et le moment où le système reconnaît l'action en cours. La Latency Rate correspond au pourcentage de l'action où celle-ci n'est pas encore reconnue (Tang, et al. 2018) comme expliqué par la Figure 1.8. Si l'intervalle temporel entre l'image à partir de laquelle l'action commence et l'image où les classifieurs prennent finalement leur décision est h, et la durée totale d'une action est H, alors la Latency rate est définie comme suit :

$$Latency\ rate = \frac{h}{H} \tag{1.10}$$

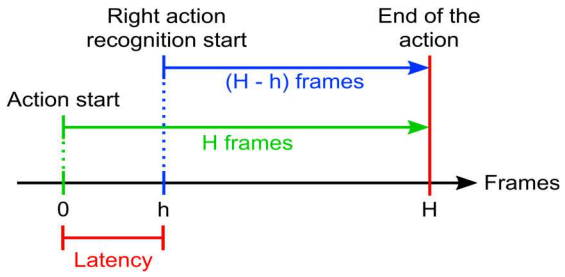


FIGURE 1.8 - Illustration du calcul de la métrique Latency Rate (TANG, ET AL. 2018)

- Error rate : L'Error Rate correspond au pourcentage de frames de l'action où celle-ci n'est pas reconnue, ou que l'action reconnue n'est pas la bonne (Tang, et al. 2018) comme le montre la Figure 1.9. Si une action peut être détectée et que la durée de cette action est W, mais que pendant ces W frames, il y a w frames détectées comme autres actions, alors l'Error rate est défini comme suit :

$$Error\ rate = \frac{w}{W} \tag{1.11}$$

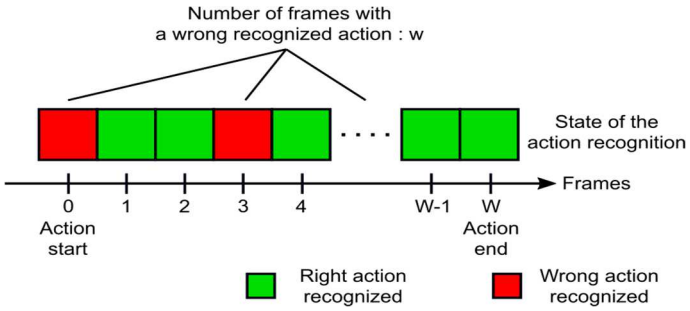


FIGURE 1.9 - Illustration du calcul de la métrique Error Rate (TANG, ET AL. 2018)

- Miss rate : Le Miss Rate correspond au pourcentage d'occurrences d'une action qui ne sont pas correctement reconnu au cours des tests de reconnaissance (Tang, et al. 2018).

Si une action apparaît réellement n fois dans une séquence vidéo, et que la méthode la détecte m fois, alors le Miss rate est défini comme suit :

$$Miss\ rate = \frac{n - m}{n} \tag{1.12}$$

- Calibrated Precision (cP) : la cP est introduit pour résoudre les problèmes de la mAP qui est sensible aux variations du rapport entre les images positives et les images d'arrière-plan négatives (si les classifieurs ne sont pas parfaits). S'il y a (relativement parlant) plus de données d'arrière-plan, la probabilité que certaines images d'arrière-plan soient faussement détectées augmente avec une plus grande confiance que certaines images positives. L'AP diminuera donc. Il est donc difficile de comparer l'AP de deux classes différentes lorsqu'elles n'ont pas le même rapport positif/négatif (Xu, et al. 2018).

  La cP est donnée par :

$$cP = \frac{TP}{TP + \frac{FP}{\omega}} = \frac{\omega \cdot TP}{\omega \cdot TP + FP} \tag{1.13}$$

  où ω est le rapport entre les images négatives et positives.

- Calibrated Average Precision (cAP) : la précision moyenne calibrée cAP est calculée sur la base de la précision calibrée (cP) (Xu, et al. 2018). Similaire à la AP, la cAP est donnée par :

$$cAP = \frac{\sum k\ cP \cdot I(k)}{P} \tag{1.14}$$

  avec $I(k)$ une fonction indicatrice égale à 1 si l'image k est un vrai positif, et égale à 0 sinon. P désigne le nombre total de vrais positifs, et ω est le rapport entre les images négatives et positives. L'avantage de la cAP est qu'elle corrige le déséquilibre de classe entre les échantillons positifs et négatifs.

- Start Localization (SL) / End Localization (EL) Scores : Le SL score correspond à la distance entre le temps de départ prédit d'une action donnée et la vérité de terrain. Le SL score est donné par :

$$SL = e^{-\frac{|t - t_{start}|}{t_{end} - t_{start}}} \tag{1.15}$$

  Le EL score correspond à la distance entre le temps de fin prédit d'une action donnée et la vérité de terrain (voir Figure 1.10) (Li, et al. 2016). Le EL score est donné par :

$$EL = e^{-\frac{|t-t_{end}|}{t_{end}-t_{start}}} \tag{1.16}$$

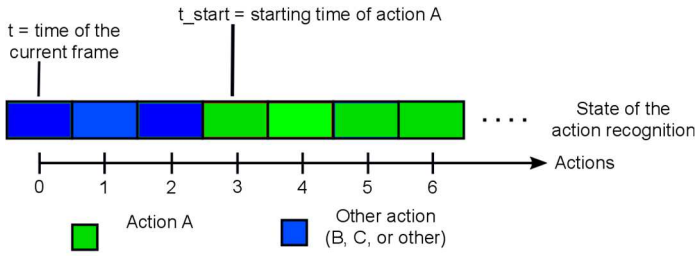


FIGURE 1.10 - Illustration du calcul de la métrique Start Localization (SL) (LI, ET AL. 2016)

Dans le cas où la classification de l'image t est fausse, le SL/EL score prend la valeur 0.

- RAccuracy : dans cette métrique nous considérons qu'une classification d'action est correcte si et seulement si l'étiquette de l'action prédite est la même que l'étiquette de vérité terrain dans un intervalle de +/- k images relatives à l'action actuelle dans la vidéo (voir Figure 1.11) (You et Jiang 2018).

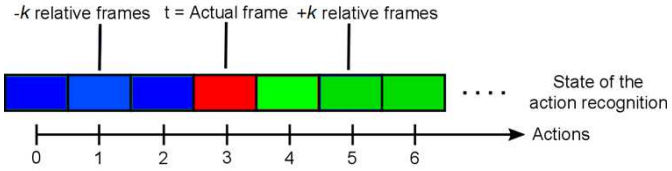


FIGURE 1.11 - Illustration du calcul de la métrique RAccuracy (YOU ET JIANG 2018)

Dans cette section, nous avons défini les métriques d'évaluation des méthodes de reconnaissance d'actions humaines les plus utilisées dans la littérature. Ces métriques permettent d'évaluer les performances des modèles de HAR sur des jeux de données. Les métriques d'évaluation sont propres au type de tâche d'apprentissage automatique effectuée par un modèle. Pour la HAR, les principales métriques sont : Accuracy, F1-score, MoF, IoU, Latency, Error Rate et SL/EL score car elles évaluent au mieux le modèle de HAR en mesurant le degré de correspondance des actions prédites avec les actions réelles. Dans la section suivante, nous allons présenter les jeux de données de HAR les plus récents.

I.4 Jeux de données de HAR

Au cours de la dernière décennie, un certain nombre de jeux de données RGB-D de référence ont été collectés et mis à la disposition de la communauté des chercheurs. Les sources de ces jeux de données sont principalement de trois catégories : les systèmes de capture de mouvement (Mocap), les caméras à lumière structurée (par exemple, Kinect v1) et les caméras à temps de vol (ToF) (par exemple, Kinect v2). Les modalités de ces jeux de données couvrent donc le RGB,

la profondeur, le squelette et leurs combinaisons (multimodalités). Des études récentes et complètes de ces jeux de données ont été publiées dans la littérature (P. Wang, W. Li et P. Ogunbona, et al. 2018). Dans ce qui suit, nous présentons les jeux de données qui sont couramment adoptés pour évaluer les méthodes basées sur l'apprentissage profond. Les jeux de données ont été divisés en deux groupes : les jeux de données segmentés et les jeux de données continus/en ligne (voir Figure 1.5). Les jeux de données avec « * » seront utilisés dans ces travaux de thèse. Une synthèse des différents jeux de données sera présentée à la fin de cette section.

I.4.1 Jeux de données segmentés

Par jeux de données segmentés, nous entendons les jeux de données dans lesquels les échantillons correspondent à des actions/gestes complets de début et de fin, avec un segment/séquence par action. Ils sont principalement utilisés à des fins de classification. Nous présentons dans ce qui suit les jeux de données segmentés qui sont couramment les plus utilisés pour l'évaluation des méthodes basées sur l'apprentissage profond.

I.4.1.1 CMU Mocap

Le jeu de données CMU Mocap (CMU Mocap Dataset s.d.) est l'un des premières ressources qui comprend une grande variété d'actions humaines, notamment l'interaction entre deux sujets, la locomotion humaine, les sports et d'autres actions humaines. Il est enregistré avec une fréquence de 120 Hz avec des images d'une résolution de 4 mégapixels. Ce jeu de données fournit des données RGB et squelettes. Un extrait de ce jeu de données est donné par la Figure 1.12.

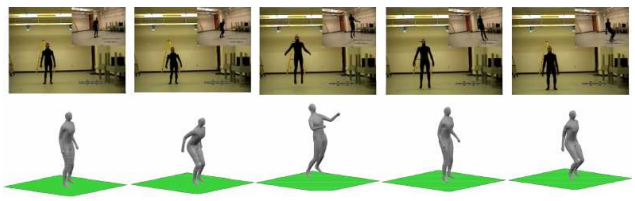


FIGURE 1.12 - Extrait du jeu de données CMU Mocap

I.4.1.2 HDM05

Le jeu de données de capture de mouvement HDM05 (Müller, et al. 2007) a été capturé par une technologie basée sur des marqueurs optiques avec une fréquence de 120 Hz. Il contient 2337 séquences avec 130 classes d'actions sportives réalisées par 5 acteurs non professionnels avec 31 articulations dans chaque image. Outre les données du squelette, ce jeu de données fournit également des données RGB. Un extrait de ce jeu de données est donné par la Figure 1.13.

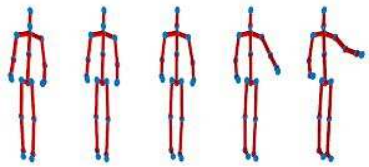


FIGURE 1.13 - Extrait du jeu de données HDM05

I.4.1.3 MSR-Action3D

MSR-Action3D (Li, Zhang et Liu 2010) est le premier jeu de données d'action RGB-D public de référence collectées à l'aide du capteur Kinect par « Microsoft Research », « Redmond » et « l'Université de Wollongong » en 2010. Dix sujets ont effectué 20 actions sportives trois fois. Toutes les vidéos ont été enregistrées à partir d'un point de vue fixe et les sujets faisaient face à la caméra pendant l'exécution des actions. L'arrière-plan de l'ensemble de données a été supprimé par un post-traitement. Plus précisément, si une action doit être effectuée avec un bras ou une jambe, les acteurs devaient l'effectuer avec le bras ou la jambe droite. Un extrait de ce jeu de données est donné par la Figure 1.14.

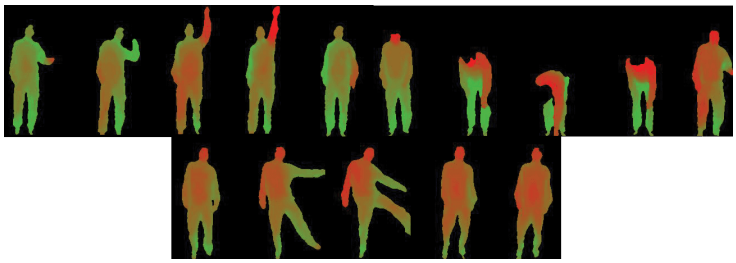


FIGURE 1.14 - Extrait du jeu de données MSR-Action3D

I.4.1.4 MSRC-12

Le jeu de données MSRC-12 (Fothergill, et al. 2012) a été collecté par « Microsoft Research Cambridge » et « l'Université de Cambridge » en 2012. Les auteurs ont fourni aux participants trois modalités d'instruction familières et faciles à préparer, ainsi que leurs combinaisons. Ces modalités sont (1) un texte descriptif décomposant la cinématique de l'exécution, (2) une série ordonnée d'images statiques d'une personne exécutant le geste avec des flèches annotant, et (3) une vidéo (images dynamiques) d'une personne exécutant le geste. Il y a 30 participants au total réalisant 12 gestes divers de la vie quotidienne. Ce jeu de données a été capturé à l'aide d'un capteur Kinect et seules les données du squelette sont disponibles. Les fichiers de mouvement contiennent les traces de 20 articulations estimées à l'aide du pipeline « Kinect Pose Estimation ». La pose du corps est capturée à un taux d'échantillonnage de $30Hz$ avec une précision de $\sim 2cm$ dans les positions des articulations. Un extrait de ce jeu de données est donné par la Figure 1.15.

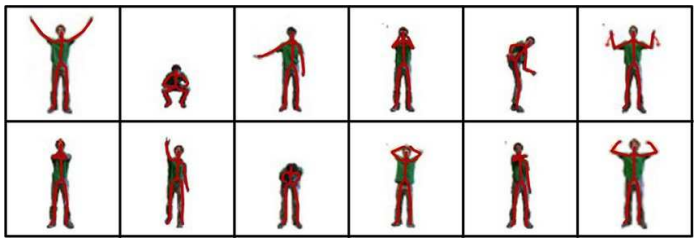


FIGURE 1.15 - Extrait du jeu de données MSRC-12

I.4.1.5 MSRDailyActivity3D

Le jeu de données MSRDailyActivity3D (Wang, et al. 2012) a été collecté par « Microsoft » et « l'Université Northwestern » en 2012 et se concentre sur les activités quotidiennes. La motivation était de couvrir les activités quotidiennes humaines dans le salon. Il comprend 16 actions (ex : boire, manger, marcher etc.) qui ont été réalisées par 10 acteurs alors qu'ils étaient assis sur le canapé ou debout près du canapé. La caméra était fixée en face du canapé. En plus des données de profondeur, les données du squelette sont également enregistrées, mais les positions des articulations extraites par le capteur sont très bruitées car les acteurs étaient soit assis sur le canapé, soit debout près du canapé. Un extrait de ce jeu de données est donné par la Figure 1.16.

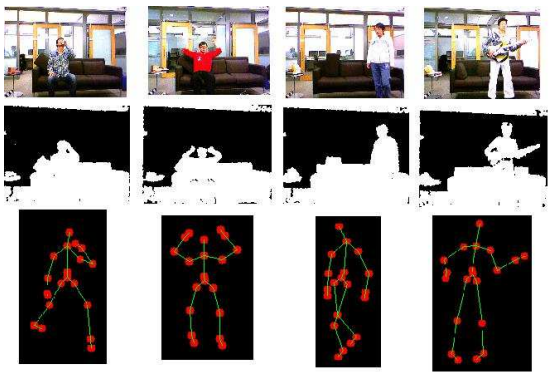


FIGURE 1.16 - Extrait du jeu de données MSRDailyActivity3D

I.4.1.6 UTKinect

Le jeu de données UTKinect (Xia, Chen et Aggarwal 2012) a été collecté par l'Université du Texas à Austin en 2012. Dix types d'actions humaines de la vie quotidienne (ex : marcher, s'asseoir, se lever etc.) ont été réalisées deux fois par 10 sujets. Les sujets ont effectué les actions à partir d'une variété de vues. L'un des défis de ce jeu de données est dû au fait que les actions sont exécutées avec une grande variabilité en fonction de l'acteur. En outre, les occlusions entre l'homme et l'objet et les parties du corps hors du champ de vision ont encore accru la difficulté de l'ensemble de données. La vérité terrain en termes d'étiquettes d'action et de segmentation des séquences est fournie. Un extrait de ce jeu de données est donné par la Figure 1.17.

FIGURE 1.17 - Extrait du jeu de données UTKinect

I.4.1.7 SBU Kinect Interaction Dataset

Le jeu de données SBU Kinect Interaction (Yun, et al. 2012) a été collecté par l'université de Stony Brook en 2012. Il contient huit types d'interactions sociales entre 2 personnes (ex : s'approcher, s'éloigner, serrer la main etc.). Toutes les vidéos ont été enregistrées avec le même fond intérieur. Sept participants ont été impliqués dans l'exécution des activités. Le jeu de données est segmenté en 21 ensembles et chaque ensemble contient une ou deux séquences de chaque catégorie d'action. Deux types d'informations de base sont fournies : les étiquettes d'action de chaque vidéo segmentée et l'identification de l'acteur « actif » et de l'acteur « passif ». Un extrait de ce jeu de données est donné par la Figure 1.18.

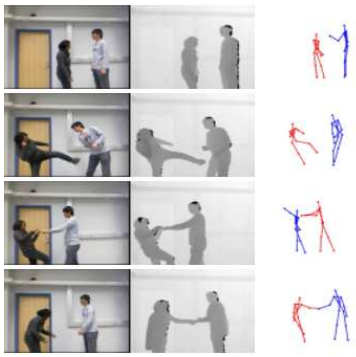


FIGURE 1.18 - Extrait du jeu de données SBU Kinect Interaction

I.4.1.8 Berkeley MHAD

Le jeu de données multimodale d'actions humaines de Berkeley (Berkeley MHAD) (Ofli, et al. 2013), collectée par l'Université de Californie à Berkeley et l'Université Johns Hopkins en 2013, a été capturée avec cinq modalités différentes afin d'élargir les champs d'application. Les modalités sont dérivées de : un système Mocap optique, quatre caméras de vision stéréo multi-vues, deux caméras Microsoft Kinect v1, six accéléromètres sans fil et quatre microphones. Douze sujets ont effectué 11 actions, cinq fois chacune. Trois catégories d'actions sont incluses : (1) actions avec mouvement de toutes les parties du corps, par exemple, sauter sur place, etc., (2) actions avec une dynamique élevée dans les extrémités supérieures, par exemple, agiter les mains, taper des mains, etc. et (3) actions avec une dynamique élevée dans les extrémités inférieures,

par exemple, s'asseoir, se lever. Les actions ont été exécutées avec des variations de style et de vitesse. Un extrait de ce jeu de données est donné par la Figure 1.19.

FIGURE 1.19 - Extrait du jeu de données Berkeley MHAD

I.4.1.9 Northwestern-UCLA Multiview Action 3D

Northwestern-UCLA Multiview Action 3D (wang, et al. 2014) a été collecté par l'Université Northwestern et l'Université de Californie à Los Angles en 2014. Ce jeu de données contient des données prises à partir de différents points de vue. Il comprend 10 actions de la vie quotidienne (ex : s'asseoir, saluer, jeter etc.). Les actions ont été réalisées par 10 acteurs et capturées par trois caméras Kinect v1 simultanées. Un extrait de ce jeu de données est donné par la Figure 1.20.

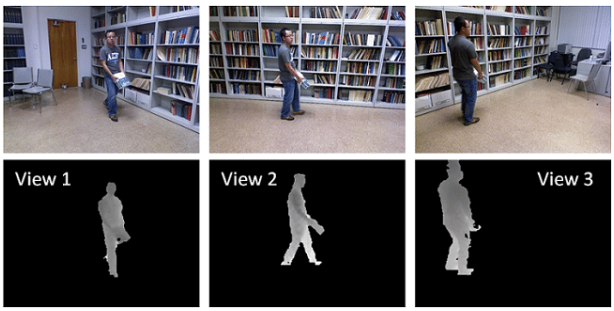


FIGURE 1.20 - Extrait du jeu de données Northwestern-UCLA Multiview Action 3D

I.4.1.10 ChaLearn LAP IsoGD

Le jeu de données ChaLearn LAP IsoGD (Wan, et al. 2016) est un grand jeu de données RGB-D pour la reconnaissance de gestes segmentés, et il a été collecté par la caméra Kinect v1. Il comprend 47933 séquences de profondeur RGB-D, chaque vidéo RGB-D représentant une instance de geste. Il inclut 249 gestes effectués par 21 individus différents. L'ensemble de données est divisé en ensembles d'apprentissage, de validation et de test. Les trois ensembles sont composés d'échantillons de sujets différents afin de garantir que les gestes d'un sujet dans les ensembles de validation et de test n'apparaîtront pas dans l'ensemble d'entraînement. Un extrait de ce jeu de données est donné par la Figure 1.21.

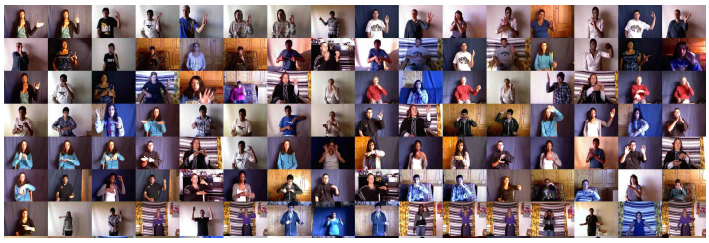


FIGURE 1.21 - Extrait du jeu de données ChaLearn LAP IsoGD

I.4.1.11 NTU RGB+D

Le jeu de données NTU RGB+D (Shahroudy, Liu, et al. 2016) est actuellement le plus grand jeu de données de reconnaissance d'action en termes de nombre d'échantillons par action. Les données RGB-D sont capturées par des caméras Kinect v2. Le jeu de données comprend plus de 56 000 séquences et 4 millions d'images, contenant 60 actions de la vie quotidienne (ex : boire, manger, lire etc.) réalisées par 40 sujets âgés de 10 à 35 ans. Il se compose d'une vue de face, de deux vues latérales et de vues à 45 degrés à gauche et à droite. Un extrait de ce jeu de données est donné par la Figure 1.22.

FIGURE 1.22 - Extrait du jeu de données NTU RGB+D

I.4.1.12 NTU RGB+D 120

Le jeu de données NTU RGB+D 120 (Liu, Shahroudy et Perez, et al. 2020) est une extension du jeu de données NTU RGB+D (Shahroudy, Liu, et al. 2016). Ce jeu de données à grande échelle est proposé pour la reconnaissance d'action humaine RGB + D, et est collecté auprès de 106 sujets distincts et contient plus de 114 000 échantillons vidéo et 8 millions d'images. Ce jeu de données inclut 120 classes d'action différentes, comprenant des activités quotidiennes (ex : boire, manger, lire etc.), mutuelles (ex : coups de poing, coups de pied etc.) et liées à la santé (ex : éternuer, chanceler etc.). Il inclut des vidéos RGB, cartes de profondeur, données squelettiques 3D et vidéos infrarouges (IR). Les séquences vidéo ont été capturés simultanément par trois caméras Microsoft Kinect V2. Les résolutions des vidéos RGB sont de 1920 × 1080, les cartes de profondeur et les vidéos IR sont toutes en 512 × 424, et les données squelettiques 3D contiennent les emplacements 3D de 25 articulations principales du corps à chaque image. Un extrait de ce jeu de données est donné par la Figure 1.23.

FIGURE 1.23 - Extrait du jeu de données NTU RGB+D 120

I.4.1.13 UCF101 - Action Recognition Dataset

UCF101 (Soomro, Zamir et Shah 2012) est un jeu de données de reconnaissance d'actions composé de vidéos d'actions réalistes, collectées à partir de YouTube, comportant 101 catégories d'actions. Ce jeu de données est une extension du jeu de données UCF50 (Reddy et Shah 2013) qui comporte 50 catégories d'actions.

Il inclut 13320 vidéos de 101 catégories d'actions, UCF101 offre de grandes variations dans le mouvement de la caméra, l'apparence et la pose de l'objet, le point de vue, l'arrière-plan encombré, les conditions d'éclairage, etc. C'est l'un des jeux de données les plus difficile vu les variations citées précédemment. Comme la plupart des jeux de données de reconnaissance d'action disponibles ne sont pas réalistes et sont mis en scène par des acteurs, l'UCF101 vise à encourager la poursuite des recherches sur la reconnaissance d'actions en apprenant et en explorant de nouvelles catégories d'action réalistes. Les catégories d'actions peuvent être divisées en cinq types : 1) Interaction homme-objet 2) Mouvement corporel uniquement 3) Interaction homme-homme 4) Jouer des instruments de musique 5) Sports. Un extrait de ce jeu de données est donné par la Figure 1.24.

FIGURE 1.24 - Extrait du jeu de données UCF101

I.4.1.14 HMDB51

Le jeu de données HMDB51 (Kuehne, et al. 2011) est une vaste collection de vidéos d'activités réelles provenant de diverses sources, y compris des films avec une petite proportion obtenue à

partir de jeux de données open source telles que les archives Prelinger et YouTube. Ce jeu de données est composé de 6 849 clips vidéo de 51 catégories d'actions de la vie quotidienne (ex : sauter, rire etc.) où chaque catégorie contient au moins 101 clips. Un extrait de ce jeu de données est donné par la Figure 1.25.

FIGURE 1.25 - Extrait du jeu de données HMDB51

I.4.1.15 Kinetics-400

Kinetics-400 (Kay, et al. 2017) fait partie de la collection Kinetics qui inclut des jeux de données à grande échelle et de liens URL de haute qualité contenant jusqu'à 650 000 clips vidéo qui couvrent 400/600/700 classes d'action humaine de la vie quotidienne, selon la version du jeu de données. Les vidéos incluent des interactions homme-objet telles que jouer des instruments, ainsi que des interactions homme-homme telles que serrer la main. Chaque classe d'action a au moins 400/600/700 clips vidéo. La version Kinetics-400 contient 400 clips vidéo. Chaque clip dure environ 10 secondes et est tiré d'une vidéo YouTube différente. Un extrait de ce jeu de données est donné par la Figure 1.26.

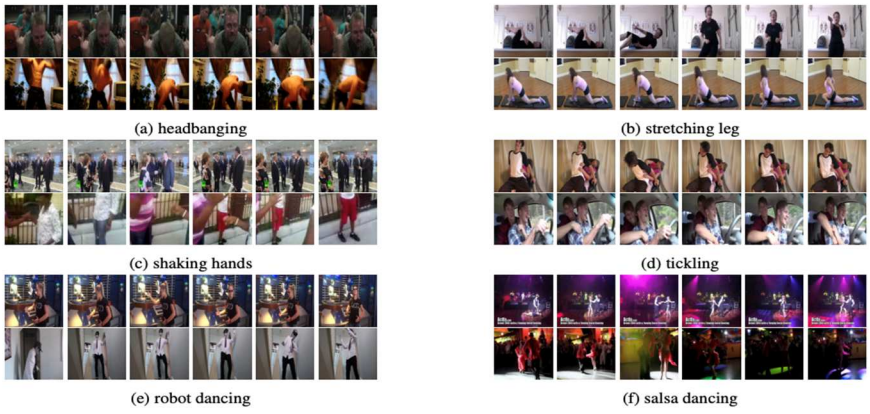


FIGURE 1.26 - Extrait du jeu de données Kinetics-400

I.4.2 Jeux de données en ligne

Les jeux de données continus/en ligne font référence aux jeux de données où chaque séquence vidéo peut contenir une ou plusieurs actions/gestes. Ces jeux de données sont principalement utilisés pour la détection, la localisation temporelle et spatiale et la prédiction d'actions en ligne. Il existe peu de jeux de données de ce type, c'est pour cette raison que certains travaux (Tang,

et al. 2018) utilisent des jeux de données segmentés concaténés pour simuler des jeux de données en ligne.

I.4.2.1 G3D

Le jeu de données Gaming 3D (G3D) (Bloom, Makris et Argyriou 2012) capturé par l'Université de Kingston en 2012 se concentre sur la reconnaissance d'actions en temps réel dans un scénario de jeu. Il contient 10 sujets effectuant 20 actions de jeu (ex : coup de poing à droite, défendre, service de tennis etc.). Chaque sujet a effectué ces actions trois fois. Il contient des données synchronisées de vidéo, de profondeur et de squelette. Un extrait de ce jeu de données est donné par la Figure 1.27.

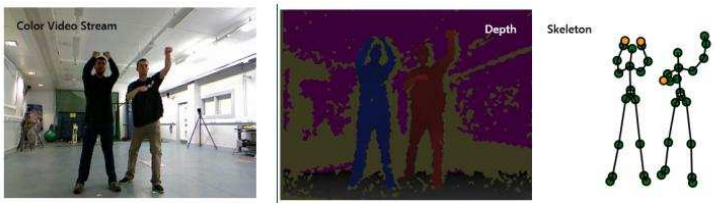


FIGURE 1.27 - Extrait du jeu de données G3D

I.4.2.2 ChaLearn2014 Multimodal Gesture Recognition

ChaLearn2014 Multimodal Gesture Recognition (Escalera, et al. 2015) est un jeu de données multimodal collecté par le capteur Kinect v1, incluant les modalités RGB, profondeur, squelette et audio. Dans toutes les séquences, un seul utilisateur est enregistré devant la caméra, effectuant des gestes de communication naturels. Les images de début et de fin de chaque geste sont annotées avec l'étiquette de classe du geste. Il contient près de 14 000 gestes étiquetés manuellement (images de début et de fin) dans des séquences vidéo continues, avec un vocabulaire de 20 catégories de gestes de signes italiens (ex : marcher, se battre etc.). Il y a 1 720 800 images étiquetées dans 13 858 fragments vidéo d'environ 1 à 2 minutes échantillonnés à 20 Hz. Les gestes sont exécutés par 27 personnes différentes dans des conditions diverses, notamment des vêtements, des positions, des arrière-plans et des éclairages variés. Un extrait de ce jeu de données est donné par la Figure 1.28.

FIGURE 1.28 - Extrait du jeu de données ChaLearn2014 Multimodal Gesture Recognition

I.4.2.3 ChaLearn LAP ConGD

Le jeu de données ChaLearn LAP ConGD (Wan, et al. 2016) est un grand jeu de données RGB-D pour la reconnaissance de gestes en continu. Il a été collecté par le capteur Kinect v1 et

comprend 47933 instances de gestes RGB-D dans 22535 vidéos. Chaque vidéo peut contenir un ou plusieurs gestes. Il inclut 249 gestes effectués par 21 individus différents. Un extrait de ce jeu de données est donné par la Figure 1.29.

FIGURE 1.29 - Extrait du jeu de données ChaLearn LAP ConGD

I.4.2.4 PKU-MMD

PKU-MMD (Liu, Hu, et al. 2017) est un jeu de données à grande échelle crée pour la compréhension d'actions humaines 3D multimodales continues et couvre un large éventail d'activités humaines complexes. Il a été capturé par le capteur Kinect v2. Il contient 1076 longues séquences vidéo dans 51 catégories d'actions de la vie quotidienne (ex : porter veste, peigner les cheveux etc.), réalisées par 66 sujets dans trois vues de caméra. Il contient près de 20 000 instances d'action et 5,4 millions d'images au total. Il fournit des sources de données multimodales, notamment RGB, profondeur, rayonnement infrarouge et squelette. Un extrait de ce jeu de données est donné par la Figure 1.30.

FIGURE 1.30 - Extrait du jeu de données PKU-MMD

I.4.2.5 ActivityNet

ActivityNet (Heilbron, et al. 2015) est un jeu de données de référence pour la compréhension des activités humaines. Il vise à couvrir un large éventail d'activités humaines complexes qui présentent un intérêt pour les gens dans leur vie quotidienne. Dans sa version actuelle, ActivityNet fournit des échantillons de 203 classes d'activités avec une moyenne de 137 vidéos non rognées par classe, pour un total de 849 heures de vidéo. Chaque vidéo comporte en moyenne 1,41 activité annotée. Un extrait de ce jeu de données est donné par la Figure 1.31.

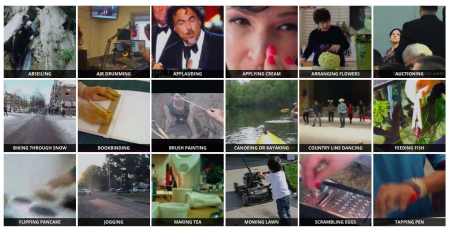


FIGURE 1.31 - Extrait du jeu de données ActivityNet

I.4.2.6 Thumos

Le jeu de données THUMOS (Idrees, et al. 2016) vise à explorer de nouveaux défis et de nouvelles approches pour la reconnaissance d'actions à grande échelle avec un grand nombre de classes à partir de vidéos libres en ligne dans un cadre réaliste. Les participants pourront entraîner leurs méthodes en utilisant des clips segmentés, mais devront tester leurs systèmes sur des données en ligne. Toutes les vidéos sont collectées sur YouTube, et leurs caractéristiques de bas niveau pré extraites sont mises à disposition. Ce jeu de données THUMOS peut donc être utilisé pour deux tâches : la reconnaissance d'actions et la détection temporelle d'actions.

La plupart des jeux de données de reconnaissance d'action existants sont composés de vidéos qui ont été coupées manuellement pour délimiter l'action d'intérêt. Il s'agit d'une limitation considérable car elle ne correspond pas à la manière dont la reconnaissance d'actions est appliquée dans la pratique. C'est la raison pour laquelle ce jeu de données été introduit. Un extrait de ce jeu de données est donné par la Figure 1.32.

FIGURE 1.32 - Extrait du jeu de données THUMOS

I.4.2.7 OAD *

Le jeu de données Online Action Detection Dataset (OAD) (Li, et al. 2016) a été capturé à l'aide du capteur Kinect v2, qui collecte des images couleur, des images de profondeur et des articulations du squelette humain de manière synchrone. Il a été capturé dans un environnement intérieur de la vie quotidienne. Différents acteurs ont effectué librement 10 actions quotidiennes, parmi lesquelles boire, manger, écrire, etc. Il est composé de 59 longues séquences vidéo à 8 images par seconde (au total 103 347 images sur 216 minutes). Un extrait de ce jeu de données est donné par la Figure 1.33.

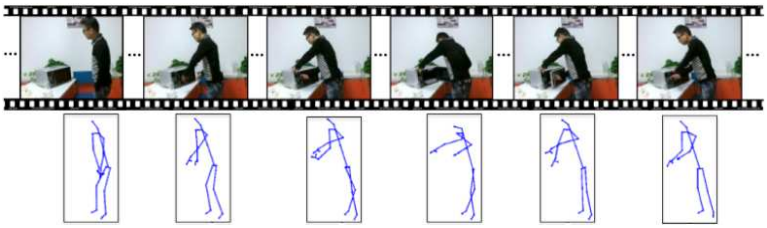


FIGURE 1.33 - Extrait du jeu de données OAD

I.4.2.8 UOW *

Le jeu de données UOW Online Action3D Dataset (Tang, et al. 2018) se compose de 20 séquences vidéos squelettes d'action de la vie quotidienne (ex : applaudir, saluer etc.) enregistrées avec un Microsoft Kinect V2. 20 participants ont été invités à effectuer ces actions en fonction de leurs habitudes personnelles. Chaque action est répétée 3 à 5 fois puis, en continu, chaque opérateur réalise les 20 actions dans un ordre aléatoire. Un extrait de ce jeu de données est donné par la Figure 1.34.

FIGURE 1.34 - Extrait du jeu de données UOW

I.4.3 Synthèse des jeux de données pour la HAR

Nous présentons dans le Tableau 1.4 les jeux de données de reconnaissance d'actions segmentées et en ligne les plus utilisés dans la littérature. Des informations tels que le nombre d'échantillons, le nombre de classes d'actions, le nombre de personnes impliquées, les capteurs utilisés, les modalités de données utilisées, etc. sont spécifiées (Asadi-Aghbolaghi, et al. 2017).

TABLEAU 1.4 - Les jeux de données de HAR les plus utilisés dans la littérature complété à partir de (ASADI-AGHBOLAGHI, ET AL. 2017). Les jeux de données notés avec ** sont ceux qui ont été ajoutés. Les jeux de données avec * sont ceux qui ont été ajoutés et utilisés ultérieurement dans nos expérimentations de HAR.

Jeu de données	Segmenté En ligne	Nb. Échantillon	Classes	Sujets	Vues	Capteurs	Type des actions	Modalités
CMU Mocap** (2001)	☑	2235	45	144	1	Mocap	Activités quotidiennes (Ex. Marcher)	RGB+S[1]
HDM05 (2007)	☑	2337	130	5	1	Mocap	Activités quotidiennes (Ex. Tourner à gauche)	RGB+S
MSRAction3D (2010)	☑	567	20	10	1	Kinect v1	Activités quotidiennes (Ex. Saluer)	D[2]+S
MSRC-12** (2012)	☑	6244	12	30	1	Kinect v1	Activités quotidiennes (Ex. Saluer)	RGB+D+S
CAD-60** (2011)	☑	60	12	4	-	Kinect v1	Activités quotidiennes (Ex. Brosser les dents)	RGB+D+S
RGBD-HuDaAct** (2011)	☑	1189	13	30	1	Kinect v1	Activités quotidiennes (Ex. Manger un repas)	RGB+D
MSRDailyActivity3D (2012)	☑	320	16	10	1	Kinect v1	Activités quotidiennes (Ex. Lire un livre)	RGB+D+S
Act42** (2012)	☑	6844	14	24	4	Kinect v1	Activités quotidiennes (Ex. Boire)	RGB+D
CAD-120** (2013)	☑	120	20	4	-	Kinect v1	Activités quotidiennes (Ex. Brosser les dents)	RGB+D+S
3D Action Pairs** (2013)	☑	360	12	10	1	Kinect v1	Paires d'actions quotidiennes (Ex. Porter/enlever)	RGB+D+S

[1] Squelette

[2] Profondeur (Depth)

Jeu de données	Segmenté	En ligne	Nb. Échantillon	Classes	Sujets	Vues	Capteurs	Type des actions	Modalités
Multiview 3D Event** (2013)	☑		3815	8	8	3	Kinect v1	Événements quotidiens (ex. Lire un livre)	RGB+D+S
Northwestern-UCLA (2014)	☑		1475	10	10	3	Kinect v1	Activités quotidiennes (Ex. Se lever)	RGB+D+S
UWA3D Multiview** (2014)	☑		900	30	10	1	Kinect v1	Activités quotidiennes (Ex. Sauter)	RGB+D+S
Office Activity** (2014)	☑		1180	20	10	3	Kinect v1	Activités de bureau (Ex. Répondre au téléphone)	RGB+D
UTD-MHAD** (2015)	☑		861	27	8	1	Kinect v1+WIS[3]	Actions quotidiennes (Ex. Balayez vers la gauche)	RGB+D+S+ID
UWA3D Multiview II** (2016)	☑		1075	30	10	5	Kinect v1	Activités quotidiennes (Ex. Sauter)	RGB+D+S
NTU-RGB+D (2016)	☑		56880	60	40	80	Kinect v2	Actions quotidiennes, mutuelles et liées à la santé (Ex. Boire, éternuer, donner des coups de poing)	RGB+D+IR[4]+S
G3D** (2012)		☑	234	20	10	1	Kinect v1	Actions de jeu (Ex. Service de tennis)	RGB+D+S
Online RGB+D Action** (2014)		☑	336	7	24	1	Kinect v1	Activités quotidiennes (Ex. Boire)	RGB+D+S
ChaLearn2014** (2014)		☑	13858	20	27	1	Kinect v1	Activités quotidiennes (Ex. Marcher)	RGB+D+S
ChaLearn LAP ConGD** (2016)		☑	22535	249	21	1	Kinect v1	Activités quotidiennes (Ex. Sauter)	RGB+D
PKU-MMD** (2017)		☑	1076	51	66	3	Kinect v2	Actions quotidiennes	RGB+D+S

[3] Wireless Inductive System

[4] Infrarouge

Jeu de données	Segmenté	En ligne	Nb. Échantillon	Classes	Sujets	Vues	Capteurs	Type des actions	Modalités
								(Ex. Serrer la main)	
OAD* (2016)		☑	59	10	-	1	Kinect v2	Activités quotidiennes (Ex. Boire)	RGB+D+S
UOW* (2018)		☑	567	20	20	1	Kinect v2	Activités quotidiennes (Ex. Boire)	RGB+D+S

Dans la suite de cette thèse, nous allons choisir les deux jeux de données OAD et UOW qui sont les plus utilisés dans la littérature pour évaluer notre méthode. D'autres jeux de données de HAR n'ont pas été pris en compte, car les classes d'action dans ces jeux de données sont soit des activités longues, comprenant des humains en vue rapprochée, soit n'incluent pas d'humains du tout. Par conséquent, l'extraction de squelettes humains à partir de ces données n'était pas possible pour évaluer notre approche.

Comme le montre le Tableau 1.4, tous les jeux de données présentés comprennent pour certains des actions de la vie quotidienne, des interactions entre individus ou des actions liées à la santé. Nous remarquons l'absence total de jeux de données comprenant des actions humaines dans un contexte industriel qui, s'ils existent, devraient faciliter et permettre l'étude et le développement de diverses techniques d'apprentissage pour la tâche d'analyse des actions humaines dans des environnements industriels impliquant des collaborations humains-robots, ce qui limite l'utilisation de la HAR dans l'industrie. Par conséquent, nous proposons par la suite de combler cette lacune en proposant un jeu de données d'actions humaines industrielles nommé « Industrial Human Action Recognition Dataset (InHARD) » qui sera présenté dans le prochain chapitre.

I.5 Vue d'ensemble des approches de HAR

La reconnaissance d'actions humaines (HAR) est un sujet qui a attiré de plus en plus les chercheurs ces dernières années vu son large champ d'applications. La HAR est constituée d'une étape d'extraction de caractéristiques, d'une étape de classification et d'une étape de détection temporelle. D'une part, l'extraction des caractéristiques consiste à identifier les informations distinctives à partir d'une séquence d'action. D'autre part, l'étape de classification permet d'identifier et de classer les actions à l'aide des méthodes d'apprentissage automatique en l'une des classes d'actions prédéfinies en prenant en compte la variabilité inter et intra-classes, en particulier si l'action est effectuée par des opérateurs de genre ou de taille différents, ou encore lorsque la manière ou bien la vitesse d'exécution des actions varient. Enfin, dans le cas de reconnaissances d'actions en ligne, l'étape de détection consiste à identifier à quel moment l'action

se déroule dans le flux de données en plus de classifier l'action. Dans cette section, nous présentons d'abord une vue d'ensemble sur les approches de reconnaissance d'actions humaines. Ensuite, nous allons exposer les principales approches d'extraction de caractéristiques et de classification utilisées dans l'état de l'art. Nous allons d'abord subdiviser les approches selon la modalité de données utilisée ; RGB, Squelettes, Profondeur ou multimodales. Ensuite, ces approches peuvent être encore subdivisées en deux sous-groupes : des approches segmentées et des approches en ligne. Puis nous discuterons des approches segmentées/en ligne basées sur des algorithmes d'apprentissage profond et des méthodes non basées sur des algorithmes d'apprentissage profond. La Figure 1.35 catégorise les méthodes de reconnaissance d'actions humaines selon les critères et les modalités[5] cités précédemment (P. Wang, W. Li et P. Ogunbona, et al. 2018).

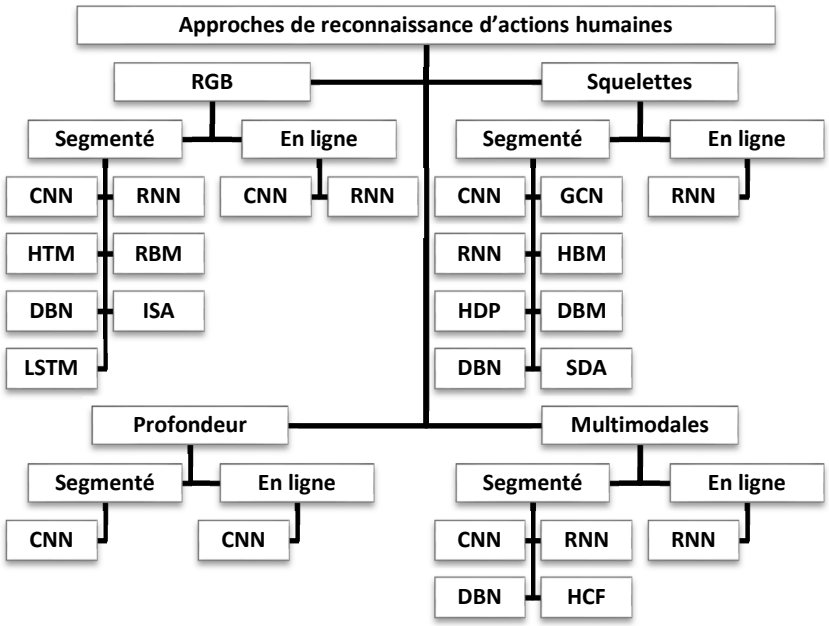


FIGURE 1.35 - Méthodes utilisées pour la HAR selon la modalité de donnée (P. WANG, W. LI ET P. OGUNBONA, ET AL. 2018)

[5] CNN: Convolutional Neural Networks
RNN: Recurrent Neural Networks
HTM: Hierarchical Temporal Memory
RBM: Restricted Boltzmann Machine
DBN: Deep Belief Networks
ISA: Independent Subspace Analysis
LSTM: Long Short-Term Memory
HBM: Hierarchical Bayesian Model
HDP: Hierarchical Dirichlet Process
DBM: Deep Boltzmann Machine
SDA: Stacked Denoising Autoencoder
HCF: Hierarchical Compound Features

I.5.1 Approches de HAR basées sur des données RGB

Dans cette partie, nous passons en revue quelques méthodes qui utilisent la modalité RGB pour la HAR. Plus précisément, étant donné que les vidéos comprennent la dynamique temporelle des mouvements humains qui est souvent cruciale pour la HAR, la plupart des travaux existants se sont concentrés sur l'utilisation de vidéos RGB pour la HAR.

Avant l'apparition des méthodes d'apprentissage profond, de nombreuses approches basées sur les caractéristiques extraites à la main (hand-crafted) et conçues pour des données RGB ont été proposées. Cette modalité, faisant référence à des images ou des vidéos, est la plus utilisée dans littérature puisqu'elle est facile à capturer via des caméras et contiennent des informations contextuelles utiles pour la HAR. Récemment, avec les grands progrès des techniques d'apprentissage profond, diverses architectures ont également été proposées. Ces dernières se caractérisent par une forte capacité de généralisation et dépassent les performances de la plupart des précédentes méthodes. Ces dernières années, la recherche dominante dans ce domaine se concentre sur la conception de différents types de frameworks d'apprentissage profond ayant pour but d'améliorer les performances de HAR. Par conséquent, nous présentons tout d'abord les travaux basés sur l'ingénierie des caractéristiques ainsi que les travaux avancés d'apprentissage profond pour la HAR basé sur des données RGB. Nous catégorisons ensuite les approches selon le type de reconnaissance d'actions utilisée ; segmenté ou en ligne. Chaque catégorie peut être principalement divisée en quatre sous-catégories selon l'architecture utilisée, à savoir, les réseaux de neurones convolutifs 2D à deux flux (Two-Stream 2D CNN), les réseaux de neurones récurrents (RNN) et les réseaux CNN 3D.

I.5.1.1 Approches de HAR segmentée

I.5.1.1.1 Approches basées sur des CNN 2D à deux flux

Le framework CNN 2D à deux flux comporte généralement deux branches CNN 2D qui prennent en entrée différentes caractéristiques extraites à partir des images ou des séquences vidéos RGB des actions pour la HAR. Une illustration de ce framework est donnée par la Figure 1.36-a. Ensuite, le résultat final est obtenu via des stratégies de fusion. Dans (Karpathy, et al. 2014), les auteurs ont étudié quatre méthodes de fusion temporelle illustrées par la Figure 1.37. La première méthode de fusion, « Single Frame », consiste à classifier des séquences vidéos d'actions en agrégeant les prédictions sur les images. Le modèle « Late Fusion » d'autre côté combine les sorties après classification. Autrement dit, il prédit la sortie finale en considérant les labels ou les scores des classifieurs impliqués. La fusion de données dans le modèle « Early Fusion » est effectuée au niveau des caractéristiques. Les vecteurs de caractéristiques sont concaténés en un seul grand vecteur de caractéristiques qui est utilisé par la suite pour la classification. Enfin, le modèle « Slow Fusion » est un mélange entre deux modèles qui fusionne lentement les informations temporelles, d'une manière que les couches supérieures ont

progressivement accès à des informations plus globales dans les dimensions spatiales et temporelles.

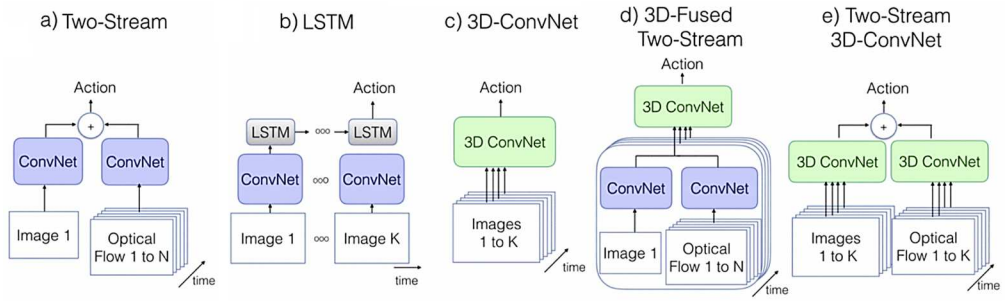


FIGURE 1.36 - Illustration des différentes méthodes d'apprentissage profond basées sur les données RGB pour la HAR(CARREIRA ET ZISSERMAN 2017). K représente le nombre total d'images dans une vidéo, tandis que N représente un sous-ensemble d'images voisines de la vidéo.

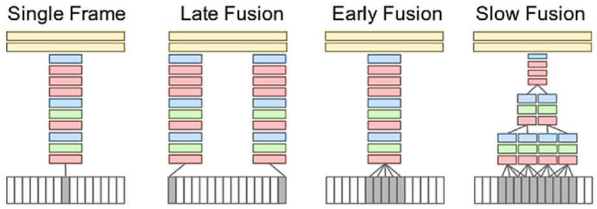


FIGURE 1.37 - Approches pour fusionner les informations temporelles (KARPATHY, ET AL. 2014). Les cases rouges, vertes et bleues indiquent respectivement les couches de convolution, de normalisation et de pooling.

Dans (Simonyan et Zisserman 2014), les auteurs ont proposé un modèle CNN à deux flux (voir Figure 1.36-a) constitué d'un réseau spatial et d'un autre temporel. La partie spatiale, caractérisée par les frames d'apparence, contient les informations sur les scènes et les objets représentées dans la vidéo. La partie temporelle, caractérisée par les mouvements à travers les images, traduit le mouvement de l'observateur (la caméra) et les objets. Par conséquent, les caractéristiques d'apparence et les caractéristiques de mouvement ont été utilisées pour l'apprentissage par le réseau CNN 2D à deux flux pour la HAR. Enfin, les scores de classification de ces deux flux ont été fusionnés pour générer le résultat final de classification. Dans d'autres travaux (Karpathy, et al. 2014), les mêmes auteurs ont utilisé des images RGB de basses résolutions et un flux fovéal haute résolution pour alimenter un réseau CNN 2D à deux flux. Ils ont étudié plusieurs approches pour étendre le réseau CNN au domaine temporel afin de tirer parti des informations spatio-temporelles locales. Les stratégies de fusion Early Fusion, Late Fusion et Slow Fusion ont été étudiées pour modéliser la dynamique temporelle dans les vidéos (voir Figure 1.37). Plusieurs études se sont efforcées d'étendre et d'améliorer les réseaux CNN classiques à deux flux, nous en examinons quelques-unes dans ce qui suit.

Dans (Chéron, Laptev et Schmid 2015), les auteurs ont utilisé les positions des articulations du corps humain pour recadrer plusieurs parties du corps à partir des images RGB et du flux

optique, qui ont été transmises à un réseau CNN 2D à deux flux pour extraire les caractéristiques, suivi d'un classifieur SVM pour la HAR. Dans (Wang, Qiao et Tang 2015), les auteurs ont alimenté un réseau CNN 2D à deux flux avec des images multi-échelles extraites des vidéos ainsi que des flux optiques pour extraire des cartes de caractéristiques de convolution. Ces dernières ont ensuite été échantillonnées sur les tubes spatio-temporels, représentant les séquences d'action spatio-temporelle détectées, centrés sur les trajectoires extraites. Les caractéristiques résultantes ont ensuite été agrégées à l'aide d'une représentation vectorielle de Fisher suivie d'une Machine à Vecteurs de Support (SVM) pour la HAR.

D'autres travaux comme dans (Wang, Xiong, et al. 2016), (Diba, Sharma et Gool 2016), (Girdhar, et al. 2017), ont étendu le framework CNN 2D à deux flux pour agréger les caractéristiques utilisées pour la HAR en utilisant des stratégies d'échantillonnage. Dans les travaux de (Wang, Xiong, et al. 2016), les auteurs ont divisé chaque vidéo en trois segments et traité chaque segment avec un réseau CNN 2D à deux flux ; un flux avec une seule image RGB et un flux avec un lot de champs de flux optiques consécutifs. Ensuite, les scores de classification des trois segments ont été fusionnés pour produire les prédictions. Dans (Girdhar, et al. 2017), les auteurs ont introduit une nouvelle représentation vidéo pour la classification d'actions permettant d'agréger les caractéristiques locales produites par les convolutions du réseau CNN sur toute l'étendue spatio-temporelle de la vidéo en intégrant un réseau à deux flux avec caractéristiques spatio-temporelles agrégées.

Les méthodes de HAR utilisant des données RGB sont confrontées à plusieurs problèmes surtout liés aux coûts de calcul élevés des flux optiques précis, ce qui les empêchent d'être en temps réel (Simonyan et Zisserman 2014). Pour faire face à ce type de problème, les auteurs dans (Zhang, et al. 2016) ont proposé une méthode pour accélérer une architecture à deux flux en remplaçant les flux optiques par des vecteurs de mouvement qui peuvent être obtenus directement à partir des vidéos compressées sans calcul supplémentaire et en temps réel. Ils ont utilisé des vecteurs de mouvement comme entrée d'un réseau CNN. Ainsi, des approches pour le transfert des connaissances du réseau CNN avec flux optique au CNN avec vecteurs de mouvement ont été proposées. Le transfert des connaissances depuis le réseau CNN avec flux optique vers le CNN avec vecteurs de mouvement a permis d'améliorer les performances de HAR. Cependant, les vecteurs de mouvement manquent de structures fines et contiennent des modèles de mouvement bruyants et imprécis, ce qui peut entraîner une dégradation évidente des performances de HAR.

I.5.1.1.2 Approches basées sur des CNN 3D

De nombreuses recherches comme dans (Tran, Bourdev, et al. 2015), (S. Ji, et al. 2013) ont étendu les CNN 2D (Figure 1.38-a) aux structures 3D (voir Figure 1.38-b), pour modéliser simultanément les informations spatiales et temporelles dans des vidéos RGB pour la HAR. Dans

(S. Ji, et al. 2013), l'un des premiers travaux de HAR basé sur des CNN 3D, les auteurs ont segmenté des vidéos des sujets humains en utilisant une méthode de détection, de localisation et de suivi des têtes dans les images. Ensuite, ils transmettent les séquences vidéos segmentées à un réseau CNN 3D pour extraire les caractéristiques spatio-temporelles. Contrairement à (S. Ji, et al. 2013), les auteurs dans (Tran, Bourdev, et al. 2015) ont introduit un réseau CNN 3D, nommé C3D, pour produire un apprentissage de bout en bout (end-to-end) des caractéristiques spatio-temporelles à partir de vidéos brutes. Cependant, les réseaux CNN 3D étaient limités dans l'apprentissage aux segments de séquence d'action seulement limitant ainsi l'extraction des dépendances spatio-temporelles à longue portée dans les vidéos.

C'est pourquoi, plusieurs approches se sont concentrées sur la modélisation des dépendances spatio-temporelles à longue portées dans les vidéos. Dans les travaux de (Diba, Fayyaz, et al. 2017), les auteurs ont étendu les réseaux DenseNet, introduits dans (Huang, Liu et Weinberger 2016), au domaine temporelle. Avec des filtres 3D et des noyaux de pooling intégrés au réseau DenseNet, ils ont conçu un réseau CNN 3D temporel appelé T3D, dans lequel la couche de transition temporelle peut modéliser des profondeurs variables de noyau de convolution temporelle. Ce réseau a permis de capturer l'apparence et les informations temporelles à court, moyen et long termes.

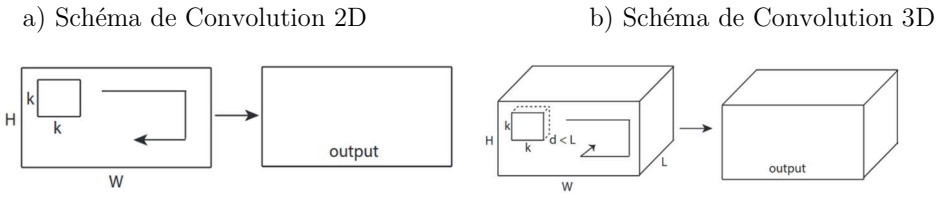


FIGURE 1.38 - Schémas des convolutions 2D/3D

Pour améliorer les performances de HAR, plusieurs autres travaux comme dans (Wang, Gao et Wang, et al. 2018), (Li, Liu, et al. 2020) ont étudié les modèles CNN 3D (Figure 1.36-c, Figure 1.36-d, Figure 1.36-e) à deux ou plusieurs flux. Par exemple, dans les travaux de (Wang, Gao et Wang, et al. 2018), les auteurs ont intégré un CNN 3D à deux flux avec un modèle LSTM pour capturer les dépendances temporelles à long terme. Dans (Li, Liu, et al. 2020), les auteurs ont introduit un CNN 3D spatio-temporel à deux flux avec des mécanismes d'attention pour capturer les dépendances spatiales et temporelles à longue distance.

Les architectures CNN 2D à deux flux ou CNN 3D ont la capacité d'apprendre différents types d'informations spatiales et temporelles à partir des séquences vidéos RGB, puis effectuent des techniques de fusion pour obtenir le résultat final. Cela leur permet de gérer efficacement les données vidéo RGB et d'obtenir des performances de HAR élevées. Cependant, ces types d'architectures ne sont toujours pas assez puissants pour modéliser les dépendances temporelles à long terme. Les réseaux de neurones récurrents à mémoire court et long terme (Long Short-Term Memory - LSTM), représentent une solution pour ce genre de problèmes liés à la

modélisation des informations temporelles.

I.5.1.1.3 Approches basées sur des RNN

Les réseaux de neurones récurrents (RNN) sont utilisés pour capturer les informations temporelles. Cela est assuré en raison des connexions récurrentes dans leurs couches cachées. Néanmoins, les réseaux RNNs traditionnels souffrent du problème de la disparition du gradient, où le réseau serait incapable de propager les informations de gradient utiles de l'extrémité de sortie du modèle vers les couches proches de l'extrémité d'entrée du modèle, ce qui rend ce type de réseau incapable de modéliser efficacement la dépendance temporelle à long terme. Ainsi, la plupart des méthodes existantes ont adopté des architectures RNN à portes (Gated-RNN), telles que les réseaux Long-Short Term Memory (LSTM), pour modéliser la dynamique temporelle à long terme dans les séquences vidéo.

Les méthodes basées sur des RNNs utilisent généralement des CNN 2D pour extraire les caractéristiques, suivis d'un réseau LSTM pour extraire les dépendances temporelles. Ces informations spatio-temporelles modélisant les actions sont combinées et utilisées pour la HAR. Dans les travaux de (Donahue, et al. 2017), les auteurs ont introduit un réseau de convolution récurrent à long terme (LRCN), qui consiste en un réseau CNN 2D permettant d'extraire les caractéristiques spatiales RGB au niveau de l'image, suivis des réseaux LSTMs pour prendre en compte les informations temporelles afin de classifier l'action.

L'introduction de mécanismes d'attention spatiale, temporelle ou bien d'attention spatio-temporelle, ont permis d'améliorer les performances de HAR des méthodes basées sur des réseaux LSTMs. Dans les travaux de (Sudhakaran, Escalera et Lanz 2018), les auteurs ont proposé un réseau récurrent, nommé LSTA, avec un module d'attention spatiale intégrée pour localiser les informations spatiales discriminante dans une séquence vidéo.

I.5.1.2 Approches de HAR en ligne

La plupart des méthodes de reconnaissance d'actions examinées précédemment reposent sur des vidéos coupées ou segmentées pour l'apprentissage de leurs modèles pour la HAR. Acquérir un jeu de données segmenté à grande échelle est souvent très coûteux et chronophage. Ces dernières années, l'apparition des jeux de données de vidéo RGB non segmentées (ou en ligne) a permis de faire évoluer la recherche et les défis de la HAR en ligne ouvrant les portes à de nouveaux champs d'applications intéressants.

I.5.1.2.1 Approches basées sur des CNN 2D

Plusieurs méthodes de reconnaissance d'actions à partir de vidéos non segmentées sont proposées. Ces méthodes utilisent souvent la technique de fenêtre glissante pour générer d'abord un nombre de fenêtres temporelles, puis un classifieur d'actions pour classifier chaque fenêtre

indépendamment dans l'une des classes d'actions prédéfinies. Sur la base du concept à deux flux, les auteurs dans (Gkioxari et Malik 2014) ont classé les régions d'intérêt représentant les régions possibles qui contiennent l'action basées sur des frames à l'aide d'indices statiques et dynamiques. Les régions ont été ensuite liées entre les frames sur la base des prédictions et de leur chevauchement spatial produisant ainsi des boites englobantes d'action respectivement pour chaque action et vidéo. Dans les travaux de (Weinzaepfel, Harchaoui et Schmid 2015), les auteurs ont également utilisé des régions d'intérêts extraites au niveau des images, pour ensuite sélectionner celles qui sont les plus marquantes. Ces images sont ensuite suivies tout au long de la vidéo. Par la suite, les auteurs ont utilisé une approche de fenêtre glissante pour détecter le contenu temporel d'une action. Dans (Liu, Yang et Ginhac 2021), les auteurs ont proposé un réseau compact de détection d'action ciblant le edge computing en temps réel, nommé ACDnet, qui aborde à la fois l'efficacité et la précision. Ce réseau exploite intelligemment la cohérence temporelle entre les images successives de la vidéo pour approcher les caractéristiques extraites par des réseaux CNN 2D plutôt que de les extraire. Ce réseau intègre également l'agrégation de caractéristiques de mémoire à partir d'images vidéo passées pour améliorer la stabilité de la détection en cours en modélisant implicitement de longs indices temporels au fil du temps. Dans les travaux de (Wang, Xiong et Lin, et al. 2017), les auteurs ont proposé un réseau, appelé UntrimmedNet, pour générer des propositions des morceaux de séquence pouvant contenir des instances d'action pour la HAR en ligne. Ce réseau se compose de 3 modules ; un module d'extraction des caractéristiques qui permet d'extraire les informations discriminantes à partir des morceaux de séquences d'actions après la génération de propositions, un module de classification permettant de classifier chaque proposition en l'une des classes prédéfinies et un module de sélection permettant de sélectionner les propositions de morceaux de séquences contenant le plus probablement des instances d'action. En complément des méthodes précédentes, les auteurs dans (Zhu, Vial et Lu 2017) ont proposé un framework, appelé TORNADO, pour la détection de propositions d'actions humaines dans des séquences vidéo non segmentées. Ils ont proposé un réseau convolutif spatio-temporel qui combine les avantages des réseaux basés sur la régression et des réseaux de mémoire CNN à long terme (Long-term Recurrent Convolutional Networks - LRCN) en dotant le réseau LSTM convolutif d'une capacité de régression. L'approche proposée consiste en un réseau convolutif de régression temporel (T-CRN) et un réseau de régression spatial (S-CRN) qui sont entraînés de bout en bout sur des données RGB et des flux optiques. Ils fusionnent les informations d'apparence, de mouvement et des contextes temporels pour régresser les boites englobantes (Bounding Box) des actions humaines simultanément.

I.5.1.2.2 Approches basées sur des CNN 3D

Dans les travaux de (Hou, Chen et Shah 2017), les auteurs ont proposé une approche CNN 3D de bout en bout pour la détection d'actions dans les vidéos. Un réseau de propositions de boites englobantes a été introduit pour tirer parti de la skip-pooling dans le domaine temporel en extrayant les informations à plusieurs échelles et niveaux d'abstraction, ce qui permet de

préserver les informations temporelles pour la localisation des actions dans les volumes 3D. Des propositions de boites englobantes (Bounding Box) sont générées et utilisées par la suite pour la détection d'actions par une régression de boites englobantes. Une couche de pooling des tubes d'intérêt a été proposée pour atténuer efficacement le problème des tailles spatiales et temporelles variables des propositions de boites englobantes. Dans (K. Liu, et al. 2018), les auteurs ont proposé une nouvelle architecture de reconnaissance d'action en temps réel, appelée « Temporal Convolutional 3D Network (T-C3D) », qui apprend les représentations d'action vidéo de manière hiérarchique. Les caractéristiques apprises sont capables de modéliser non seulement l'évolution temporelle de l'apparence entre de courts séquences d'actions, mais également la dynamique temporelle globale de l'ensemble de la vidéo. Ils combinent un réseau CNN 3D résiduel qui capture des informations de mouvement entre des images consécutives avec une nouvelle méthode de codage temporel pour extraire la dynamique temporelle de l'ensemble de la vidéo.

I.5.1.2.3 Approches basées sur des RNN

Plusieurs méthodes basées sur des réseaux RNN pour la modélisation temporelle ont été introduites. Dans les travaux de (Xu, et al. 2018), les auteurs ont proposé un framework appelé « Temporal Recurrent Network (TRN) », dans lequel les informations futures sont prédites en tant que tâche d'anticipation et utilisées par la suite pour reconnaître l'action de l'image actuelle (voir Figure 1.39). Ce réseau est basé sur une unité récurrente, la cellule TRN. A chaque instant donné, cette cellule prend en entrée un vecteur caractéristique représentant l'observation à un instant donné et sort une probabilité sur les classes d'action. L'avantage des cellules TRN est qu'en plus de modéliser les informations temporelles antérieures, elles tirent également parti des corrélations temporelles entre les actions actuelles et futures et sont ainsi capables d'utiliser ces informations pour reconnaître l'action en cours et anticiper les actions futures.

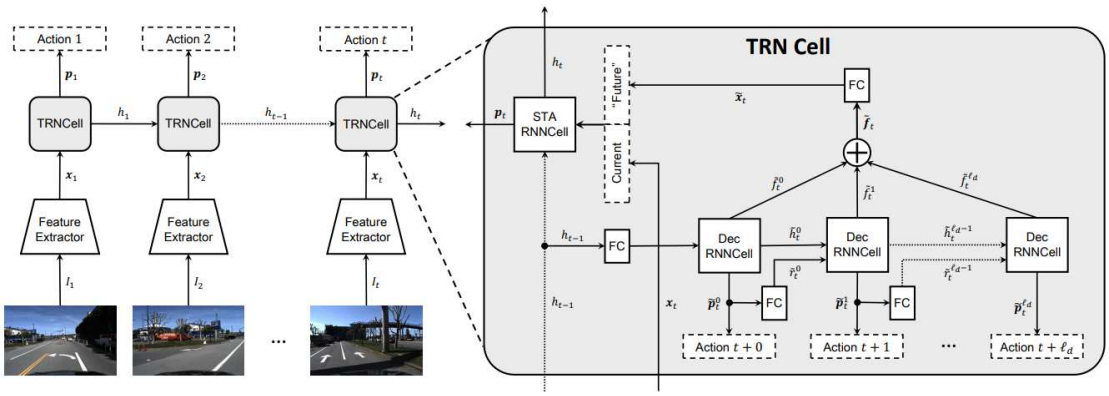


FIGURE 1.39 - Architecture du framework TRN pour la détection d'actions en ligne (XU, ET AL. 2018)

Dans (Escorcia, et al. 2016), les auteurs ont introduit les « Deep Action Proposals (DAP) » qui permettent de générer des propositions d'actions temporelles à partir de longues vidéos non segmentées pour la détection et la classification d'actions. Ils ont adopté le réseau C3D introduit dans (Tran, Bourdev, et al. 2015) comme encodeur visuel et le réseau LSTM comme encodeur de

séquence. Cependant, toutes ces méthodes ont généré des propositions par une approche de fenêtre glissante, qui découpe la vidéo en des courtes fenêtres temporelles se chevauchant, ce qui peut être coûteux en termes de temps de calcul.

Les méthodes basées sur des CNN 3D sont très puissantes pour modéliser les caractéristiques discriminantes à partir des dimensions spatiales et temporelles pour la HAR. Cependant, de nombreux frameworks basés sur des CNN 3D contiennent un grand nombre de paramètres à gérer et nécessitent donc une immense quantité de données d'apprentissage (Sun, et al. 2020).

Les données RGB représentent la modalité la plus utilisée pour la HAR dans la littérature, vu qu'elles sont faciles à collecter via des caméras RGB et incluent de riches informations. Cependant, l'extraction de caractéristiques à partir de vidéos RGB nécessite très souvent des calculs complexes vu la quantité d'informations qu'elles comprennent et qui ne sont pas toutes utiles pour la HAR. De plus, les méthodes de HAR basées sur la modalité RGB sont très sensibles aux variations de point de vue et aux encombrements d'arrière-plan, etc. (Sun, et al. 2020). Par conséquent, des méthodes de HAR avec d'autres modalités, telles que les cartes de profondeur, qui fournissent non seulement des informations structurelles géométriques 3D, mais conservent également les informations de forme, ont également fait l'objet d'une grande attention, et sont donc abordées dans la section suivante.

I.5.2 Approches de HAR basées sur des données de Profondeur

Comme présenté dans la partie I.3.2.3, la modalité de profondeur est insensible aux variations d'illumination, de couleur et de texture, fiable pour estimer la silhouette et le squelette du corps humain, et fournit aussi de riches informations structurelles 3D de la scène. Cependant, il n'y a que peu de travaux publiés sur la HAR basée sur la profondeur. Cela peut être expliqué par l'absence de couleur et de texture dans les cartes de profondeur qui affaiblit le pouvoir de représentation discriminatoire des modèles d'apprentissage tel que les CNNs, qui sont des extracteurs de caractéristiques et des classifieurs basés principalement sur la texture. Actuellement, il n'existe que des méthodes basées sur les réseaux CNN pour la HAR basée sur la profondeur.

I.5.2.1 Approches de HAR segmentée

Dans les travaux de (Rahmani et Mian 2016), les auteurs ont proposé un modèle de représentation de poses humaines invariant aux points de vue. Des images de profondeur synthétiques générées à partir d'un faible nombre de poses humaines ont été utilisées pour entrainer un réseau CNN profond. En apprenant à transférer les poses humaines de n'importe quelle vue vers un espace de haut niveau, le modèle proposé est robuste au changement de points de vue (Cross-view).

La méthode proposée comprend deux étapes ; (i) Entrainer un modèle de pose humaine général invariant aux points de vue sur des images synthétiques de profondeur via un réseau CNN (voir Figure 1.40). (ii) Modéliser par une pyramide temporelle l'action en utilisant un algorithme de transformation de Fourier permettant de trouver les caractéristiques les plus discriminantes pour la reconnaitre. Pour augmenter les données d'apprentissage pour les réseaux CNNs, ils ont généré les données d'entraînement de manière synthétique en ajustant des modèles humains 3D obtenus par motion capture. Ces données synthétiques sont générées depuis un grand nombre de points de vue.

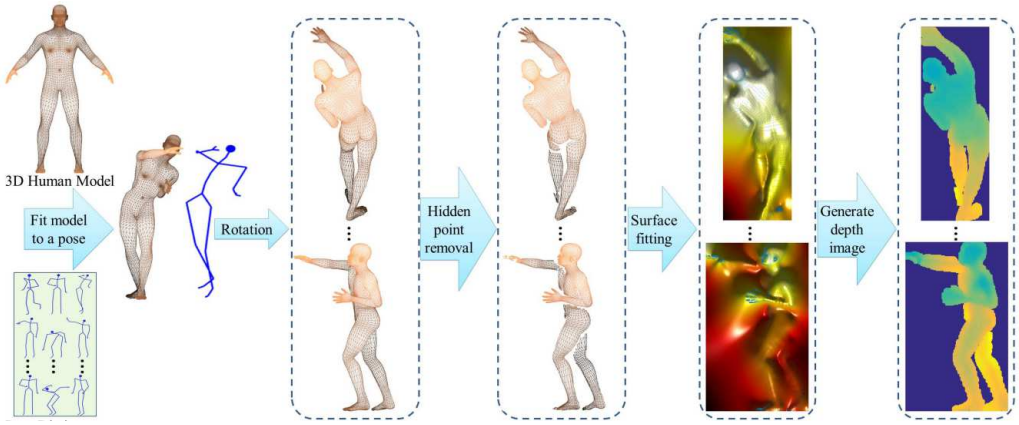


FIGURE 1.40 - Pipeline pour générer des images de profondeur synthétiques (RAHMANI ET MIAN 2016)

Dans (Xiao, et al. 2019), les auteurs ont utilisé des images multi-vues et dynamiques, extraites par des projections à partir de vidéos de profondeur. L'approche consiste à faire pivoter la caméra virtuelle dans l'espace 3D puis à projeter la vidéo de profondeur depuis différents points de vue. Les auteurs ont ensuite utilisé un réseau CNN pour l'apprentissage de caractéristiques à partir des images dynamiques générées.

Afin de capturer efficacement les informations spatio-temporelles à partir des séquences d'images de profondeur fournies par une caméra RGB-D, (Sanchez-Caballero, de López Diz, et al. 2020) ont proposé une architecture CNN 3D pour la HAR. Leur proposition est basée sur un réseau CNN 3D qui code automatiquement les motifs spatio-temporels à partir de séquences de profondeur sans prétraitements.

Dans les travaux de (P. Wang, W. Li et S. Liu, et al. 2016), les auteurs ont encodé les séquences de cartes de profondeur en trois types d'images dynamiques : (i) images de profondeur dynamiques (DDI), (ii) images normales de profondeur dynamiques (DDNI) et (iii) images normales de mouvement de profondeur dynamique (DDMNI). Ces images sont construites à partir d'une séquence de cartes de profondeur basées sur la mise en commun (pooling) bidirectionnelle pour encoder les informations spatiales concernant la posture et les informations temporelles concernant le mouvement. Ces trois représentations sont complémentaires les unes aux autres et améliorent les performances de HAR (voir Tableau 1.5 présenté dans la section

I.5.5).

I.5.2.2 Approches de HAR en ligne

Dans les travaux de (P. Wang, W. Li et S. Liu, et al. 2016), les auteurs ont adopté une approche de segmentation et de classification d'actions en utilisant un réseau CNN. La première étape consiste à segmenter les actions en fonction de la quantité de mouvement (QOM) à partir d'un flux de cartes de profondeur. Ensuite, pour chaque séquence d'action segmentée, une carte de profondeur de mouvement (IDMM) est construite à partir des cartes de profondeur. L'IDMM est calculée à partir de la quantité de mouvement estimée par rapport à la première image du segment d'action. Enfin, les réseaux CNNs sont utilisés pour apprendre les caractéristiques dynamiques de l'IDMM pour la HAR en ligne.

Une autre approche a été proposé dans (You et Jiang 2018) avec le système Action4DNet. Ce système est capable de reconnaître en temps réel les actions d'un grand ensemble d'utilisateurs. Pour ce faire, les données doivent être récupérées à l'aide de plusieurs caméras RGB disposées tout autour du lieu dans lequel on veut détecter et reconnaître les actions. En effet ce système a été pensé pour fonctionner sur des caméras de surveillance. Les données de silhouettes des différentes personnes présentes sont tout d'abord extraites et combinées entre toutes les caméras. Ainsi, une silhouette 3D est reconstruite pour chaque personne. Le système Action4DNet utilise ces silhouettes 3D pour la HAR en ligne comme l'illustre la Figure 1.41.

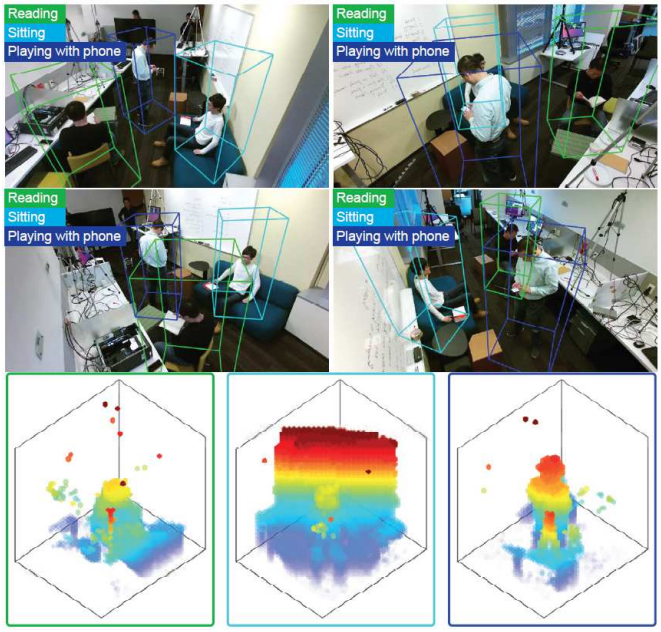


FIGURE 1.41 - Reconnaissance d'actions dans une scène avec Action4DNet (YOU ET JIANG 2018)

La structure du système Action4DNet est visible sur la Figure 1.42. Une fois les données 3D obtenues, le système les regroupe sur le temps selon une fenêtre glissante afin d'obtenir une représentation 4D des personnes présentes, et donc pouvoir analyser les mouvements effectués.

Pour ce faire, le système travaille avec un ensemble de CNNs 3D qui vont traiter chaque silhouette 3D pour en extraire les caractéristiques visuelles 3D. Le résultat est alors traité pour extraire d'une part les caractéristiques visuelles récurrentes sur la silhouette 4D (la représentation 3D du mouvement), et d'autre part les caractéristiques visuelles spécifiques à l'objet avec lequel interagit la personne (s'il y en a un). En effet, l'objet avec lequel la personne interagit peut-être utile pour définir l'action (exemples : lire un livre, utiliser un téléphone, etc.). Enfin, le résultat des deux traitements en parallèles sont combinés dans un réseau LSTM qui va se charger de reconnaître les actions après la phase d'apprentissage.

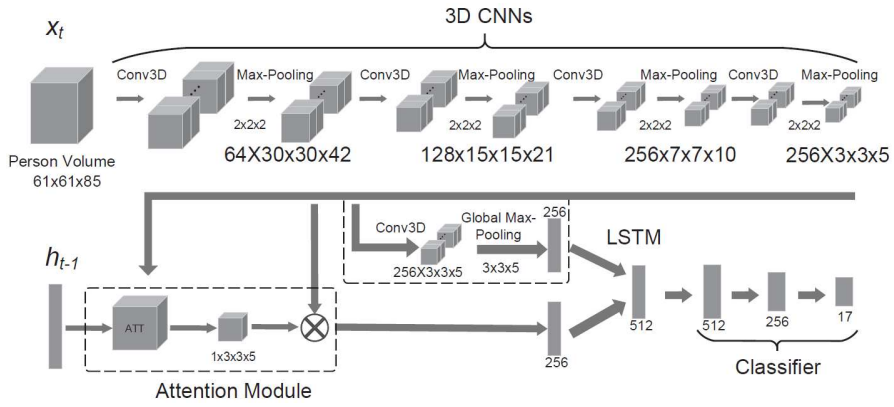


FIGURE 1.42 - Framework Action4DNet (YOU ET JIANG 2018)

En général, la modalité de profondeur fournit des informations sur les formes géométriques utiles pour la HAR. L'utilisation de données de profondeur garantit aussi que la HAR est effectuée en protégeant mieux la vie privée des personnes, car leurs identités sont plus difficilement reconnues à partir de ces données qu'à partir de données RGB. Cependant, les données de profondeur ne sont pas souvent utilisées seules, en raison du manque d'informations sur l'apparence, parfois utiles pour la HAR, les occlusions et le bruit sur les bords des cartes de profondeurs. De plus, l'absence de couleur et de texture dans les cartes de profondeur affaiblit le pouvoir de représentation discriminant des modèles CNN utilisés pour extraire les caractéristiques. La HAR avec d'autres modalités, telles que les données squelettes 2D/3D, a également reçu une grande attention ces dernières années, et est donc discuté dans la section suivante.

I.5.3 Approches de HAR basées sur des données Squelettes

Les données squelettiques codent les trajectoires des articulations du corps humain, qui caractérisent les mouvements humains. Par conséquent, les données squelettes sont également une modalité appropriée pour la HAR. Les données de squelette 2D peuvent être acquises en appliquant des algorithmes ou des outils d'estimation de pose sur des vidéos RGB ou des cartes de profondeur comme OpenPose (Cao, Hidalgo, et al. 2018) (Figure 1.43-a) ou AlphaPose (Xiu, et al. 2018) (Figure 1.43-b) ou encore MediaPipe (Lugaresi, et al. 2019). Cette modalité peut

également être collectée avec des systèmes de capture de mouvement. Les systèmes de capture de mouvement peuvent fournir des données squelettiques 3D fiables. Ainsi de nombreux travaux récents sur la HAR à base de squelettes sont proposés.

a) OpenPose b) AlphaPose

FIGURE 1.43 - Extraction des données squelettiques à partir des données RGB

Contrairement aux données RGB et de profondeur, les données de squelette contiennent les positions des articulations humaines, qui peuvent être considérées comme des caractéristiques de haut niveau pour la HAR. Les données de squelette sont résistantes aux changements d'échelle et d'éclairage, et peuvent être invariantes par rapport aux points de vue de la caméra, à la rotation du corps humain et à la vitesse du mouvement. En raison de ces avantages et de la disponibilité de capteurs précis et peu coûteux, la HAR à base de squelettes a récemment attiré beaucoup d'attention dans la communauté des chercheurs.

Nous catégorisons les approches à base de squelettes selon le type de HAR utilisé ; segmenté ou en ligne. Chaque catégorie peut être principalement divisé en trois sous-catégories selon l'architecture utilisée, à savoir, les réseaux CNN, les réseaux de neurones récurrents (RNN) et les réseaux de neurones à graphes convolutifs (GCN).

I.5.3.1 Approches de HAR segmentée

I.5.3.1.1 Approches basées sur des CNN

Dans (P. Wang, W. Li et C. Li, et al. 2016), les auteurs ont présenté une nouvelle méthode pour la reconnaissance d'actions humaines permettant de coder l'information spatio-temporelle des séquences de squelettes 3D en plusieurs images 2D appelés Cartes de Trajectoires Articulaires (Joint Trajectory Maps - JTM). Ils ont opté pour les réseaux CNNs pour exploiter les caractéristiques discriminantes pour la reconnaissance d'actions humaines en temps réel. La méthode proposée comprend quatre composantes ; (i) la rotation des données du squelette, (ii) la construction des JTMs, (iii) l'apprentissage des CNNs et (iv) la multiplication de la fusion des scores des JTMs comme illustré par la Figure 1.44. Dans la première composante, les auteurs ont essayé d'imiter plusieurs vues des squelettes pour une reconnaissance d'action croisée en pivotant les données des squelettes ce qui permet d'augmenter ainsi le nombre de données dans la phase

d'apprentissage des CNNs. La deuxième composante consiste à construire les JTMs. Comme l'action humaine se caractérise naturellement par l'évolution du corps humain au cours du temps. De ce fait, les données de squelettes contenant la position 3D des différentes parties du corps fournissent une représentation bien précise de la posture du corps humain. Ces caractéristiques peuvent être extraites et suivies à partir des JTMs. La troisième composante consiste dans l'apprentissage des CNNs. Dans cette partie, les auteurs ont opté pour les modèles AlexNet afin d'affiner les trois réseaux CNNs. La quatrième et la dernière composante représente la multiplication de la fusion des scores des JTMs. Cette procédure s'interprète comme étant une opération élémentaire afin de booster les résultats de classification. Elle combine les sorties des CNNs en multipliant les vecteurs de scores générés par un élément élémentaire sans affecter les propriétés d'indépendance linéaire. Le score maximum dans le vecteur résultant sera donc désigné comme étant la probabilité de la séquence de test et son index représente ainsi la classe ou l'étiquette finale.

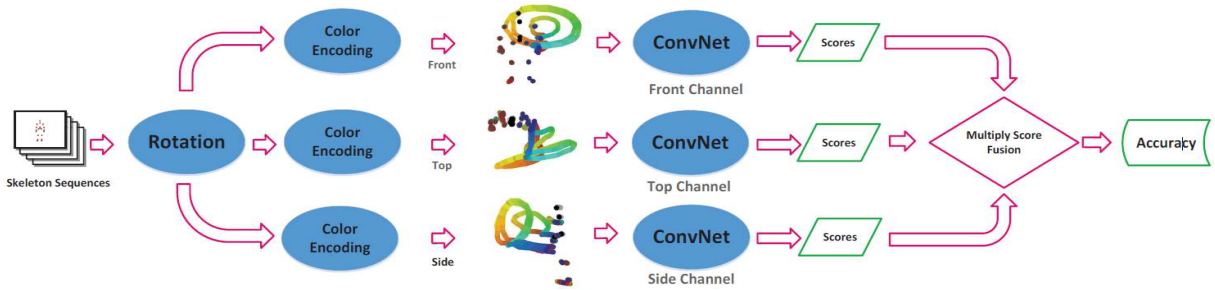


FIGURE 1.44 - Framework JTM pour la HAR basée sur les données squelettes (P. WANG, W. LI ET C. LI, ET AL. 2016)

Dans les travaux de (Caetano, et al. 2019), les auteurs ont proposé une nouvelle représentation des données squelette, appelée SkeleMotion. L'approche proposée encode la dynamique temporelle en calculant explicitement les valeurs de magnitude et d'orientation des articulations du squelette. Pour calculer les valeurs de mouvement, différentes échelles temporelles ont été utilisées permettant d'agréger plus de dynamique temporelle à la représentation, ce qui la rend capable de capturer les interactions articulaires à longue portée et de filtrer les données de mouvement bruitées. Ces représentations sont ensuite transmises pour alimenter un réseau CNN pour la HAR. Dans (Kim et Reiter 2017), les auteurs ont utilisé le réseau CNN temporel (TCN) introduit dans (Lea, et al. 2017) fournissant explicitement un type de représentation spatio-temporelle pour la reconnaissance 3D d'actions humaines. Le TCN utilise une hiérarchie de convolutions temporelles pour effectuer une segmentation des actions. Le TCN encodeur-décodeur utilise la mise en commun (pooling) et le sur-échantillonnage pour capturer efficacement les informations temporelles à longue portée.

Les réseaux CNN ont suscité beaucoup plus d'intérêt de la part des chercheurs dans l'analyse d'images 2D en raison de leurs capacités à apprendre les caractéristiques du domaine spatial. Cependant, la modélisation des informations spatio-temporelles devient un défi pour la HAR

basée sur des données squelette. Par conséquent, de nombreuses approches basées sur des RNNs ont été proposées.

I.5.3.1.2 Approches basées sur des RNN

Diverses méthodes ont appliqué et adapté les réseaux RNNs et les LSTMs pour modéliser efficacement les informations temporelles au sein des séquences squelettes pour la HAR.

Dans les travaux de (Du, Wang et Wang 2015), les auteurs ont proposé un réseau RNN hiérarchique de bout en bout qui divise le squelette humain en cinq parties du corps (deux bras, deux jambes et tronc + tête) au lieu d'utiliser la totalité du squelette dans chaque image. Ces cinq parties ont ensuite été transmises séparément à plusieurs réseaux RNN bidirectionnels, dont les représentations de sortie ont été fusionnées de manière hiérarchique pour générer des représentations de haut niveau de l'action en question. Dans (J. Liu, G. Wang, et al. 2017), les auteurs ont développé un réseau LSTM basé sur l'attention nommé « Global Context-Aware Attention LSTM (GCA-LSTM) » permettant de se concentrer sélectivement sur les articulations informatives qui sont les plus représentatives. Le réseau GCA-LSTM comprend deux couches LSTM ; la première couche code la séquence de squelette et génère une mémoire de contexte global, tandis que la seconde couche génère des représentations d'attention pour affiner le contexte global. Enfin, un classifieur « Softmax » est utilisé pour prédire la classe de l'action.

Dans les travaux de (S. Li, W. Li, et al. 2018), les auteurs ont proposé un nouveau type de réseau RNN, appelé réseau de neurones récurrents indépendants (IndRNN), là où les neurones d'une même couche sont indépendants les uns des autres et sont connectés entre les couches. Ce réseau peut être facilement régulé pour empêcher les problèmes d'explosion et de disparition du gradient, tout en permettant au réseau d'apprendre les dépendances à long terme. Plusieurs couches du réseau IndRNN peuvent être empilées efficacement, en particulier avec des connexions résiduelles sur des couches, pour augmenter la profondeur du réseau. Cela permet de mieux interpréter le comportement des neurones IndRNN dans chaque couche en raison de l'indépendance des neurones dans chaque couche.

Dans les travaux de (Song, Lan, et al. 2017), les auteurs ont proposé un framework de bout en bout avec deux types de modules d'attention basés sur des réseaux LSTM en utilisant des données squelettes en entrée. Un module d'attention spatiale, doté de grilles de sélection d'articulation, est conçu pour attribuer de manière adaptative différentes attentions à différentes articulations du squelette d'entrée dans chaque image. Le réseau conçu permet de sélectionner les articulations dominantes au sein de chaque image via le module d'attention spatiale et attribuer différents degrés d'importance à différentes images via le module d'attention temporelle. Cela permet au modèle de se concentrer davantage sur les articulations discriminantes de manière adaptative.

Dans les travaux de (Xie, et al. 2018), les auteurs ont proposé un schéma de réétalonnage temporel puis spatial pour atténuer les variations spatio-temporelles complexes des articulations du squelette. Ils ont introduit un réseau d'attention nommé « Memory Attention Network (MAN) » pour apprendre les pondérations d'attention en accordant plus d'attention aux images squelettiques représentatives. De plus le réseau utilise l'attention temporelle apprise pour recalibrer la séquence du squelette d'origine. Le MAN est constitué d'un module de réétalonnage de l'attention temporelle (TARM) et d'un module de convolution spatio-temporel (STCM). L'apprentissage de TARM vise à extraire des attentions particulières dans une cellule de mémoire pour capturer les informations de la mémoire temporelle tout au long de la séquence d'action. Le STCM traite les séquences articulaires du squelette calibrées comme des images et exploite les réseaux de neurones convolutifs (CNN) pour modéliser davantage les informations spatiales et temporelles des données du squelette. En exploitant la robustesse à la déformation des CNN, STCM extrait les représentations de caractéristiques de haut niveau afin de mieux faire face aux variations spatio-temporelles des articulations du squelette. Néanmoins, le modèle d'attention spatiale a tendance parfois à ignorer quelques articulations, bien qu'elles soient importantes pour déterminer le type d'action.

Représenter les données de squelette sous la forme d'une séquence vectorielle traitée par des réseaux RNNs, ou de cartes 2D/3D traitées par des CNNs, ne peut pas modéliser entièrement les configurations spatio-temporelles complexes et les corrélations des articulations du corps humain. Les représentations de graphes peuvent être plus appropriées pour représenter les données du squelette. En conséquence, de nombreuses méthodes de HAR basées sur des réseaux de neurones à graphes (GNN) et des réseaux convolutifs à graphes (GCN) ont été proposées.

I.5.3.1.3 Approches basées sur des GNN/GCN

Dans les travaux de (Yan, Xiong et Lin 2018), les auteurs ont proposé une nouvelle représentation des séquences squelettiques pour la HAR en étendant les réseaux convolutifs à graphes (GCN) au domaine spatio-temporel (ST-GCN). Les GCNs généralisent le fonctionnement de l'opération de convolution à partir de données traditionnelles (images) en données de graphes comme illustré par la Figure 1.45. Le fonctionnement consiste à apprendre à une fonction f de générer la représentation d'un nœud v_i en agrégeant ses propres caractéristiques X_i et les caractéristiques des voisins X_j, où $j \in N(v_i)$.

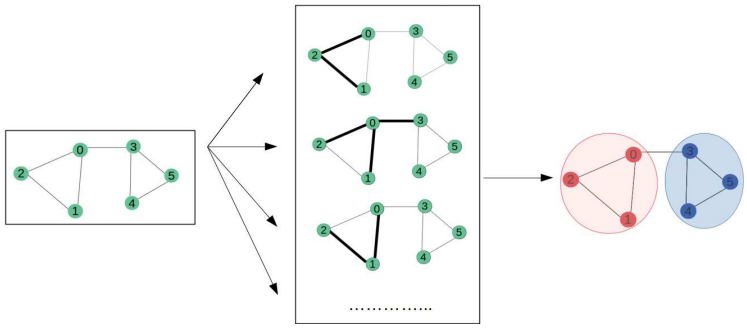


FIGURE 1.45 - Illustration des réseaux de neurones convolutifs à graphes (GCNs) (DEFFERRARD, BRESSON ET VANDERGHEYNST 2016)

Lors d'une action donnée, les articulations se déplacent en groupes locaux, que les auteurs dans (Yan, Xiong et Lin 2018), ont appelé parties du corps (Body parts) ce qui facilite leurs modélisations. Cela est dû au fait que certaines parties limitent la modélisation des trajectoires des articulations au sein de régions locales par rapport au squelette complet, formant ainsi une représentation hiérarchique des séquences du squelette comme exprimé par la Figure 1.46. Étant donné les séquences d'articulations du corps sous forme de coordonnées 2D ou 3D, les auteurs ont ensuite construit un graphe spatio-temporel là où les articulations sont représentées par les nœuds du graphe, et les connectivités naturelles dans les structures du corps humain sont représentées par les arrêtes du graphe. Cette construction se déroule en 2 étapes ; (i) d'abord les articulations d'une image sont reliées par des bords en fonction de la connectivité de la structure squelettique du corps humain. (ii) ensuite, chaque articulation sera connectée à la même articulation dans l'image suivante.

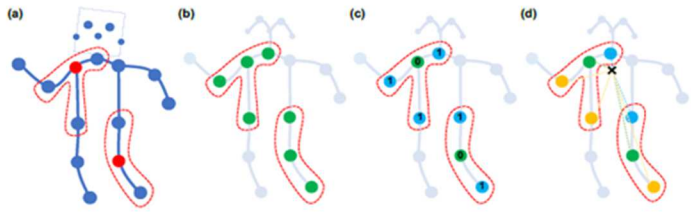


FIGURE 1.46 - Illustration de la représentation des articulations du squelette en groupe locaux (YAN, XIONG ET LIN 2018)

L'ensemble des arrêtes E du graphe G est composé de deux sous-ensembles, le premier sous-ensemble décrit la connexion intra-squelette pour chaque image, noté $ES = \{v_{ti}v_{tj} \mid (i,j) \in H\}$, où H est l'ensemble des articulations du corps humain naturellement connectées. Le deuxième sous-ensemble contient les arrêtes inter-images, qui relient les mêmes liaisons dans des images consécutives $EF = \{v_{ti}v_{(t+1)i}\}$. Toutes les arrêtes dans EF pour une articulation particulière i représenteront ainsi sa trajectoire au cours du temps. L'architecture du modèle ST-GCN est donnée par la Figure 1.47.

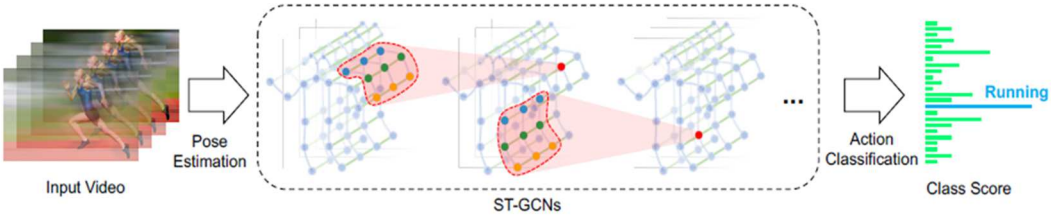


FIGURE 1.47 - Architecture du modèle ST-GCN (YAN, XIONG ET LIN 2018)

La HAR segmentée basée sur les données squelettes a été intensivement étudiée et développée ces dernières années. La HAR en ligne reste en revanche une tâche difficile et moins développée. Même si des positions 3D précises de différentes articulations sont disponibles, la tâche de reconnaissance d'action en ligne est une tâche beaucoup plus complexe que la HAR segmentée en raison des fortes variations temporelles et spatiales dans la manière dont une action est effectuée puisqu'elle vise à détecter l'action après avoir observé partiellement sa séquence et sans utiliser d'autres informations.

I.5.3.2 Approches de HAR en ligne basées sur des RNN

Dans (Li, et al. 2016), les auteurs ont proposé un framework de classification et de régression de bout en bout basé sur des réseaux RNNs pour mieux explorer le type des actions et les informations de localisation temporelle. Ils ont étudié le problème de détection et de reconnaissance en ligne d'actions humaines à partir des données squelettiques. Ce framework est capable de déterminer automatiquement le début et la fin de chaque action sans la nécessité d'utiliser de fenêtre glissante ou d'avoir une perspective explicite en avant ou en arrière de l'action. Le framework proposé comprend 3 sous-réseaux ; (i) un sous-réseau LSTM pour l'extraction de caractéristiques et la modélisation dynamique temporelle, (ii) un sous-réseau de classification pour prédire la classe discrète d'un ensemble d'entrée et finalement (iii) un sous-réseau de régression qui permet de prévoir l'action avant qu'elle ne se produise. L'architecture du framework JCR-RNN est donnée par la Figure 1.48.

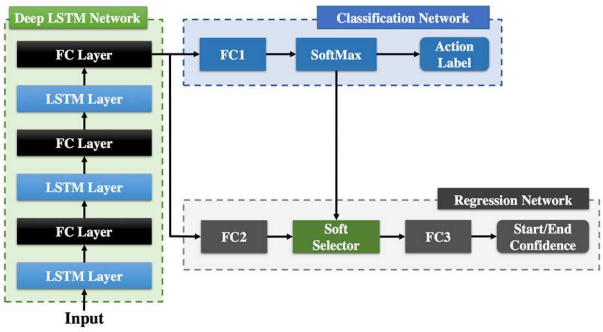


FIGURE 1.48 - Architecture du framework JCR-RNN(Li, et al. 2016)

Nous remarquons que les méthodes de HAR basées sur les réseaux GNNs en général et sur les GCNs en particulier sont les plus efficaces en termes de performances de HAR par rapport aux

méthodes basées sur les CNNs et RNNs et montrent leurs robustesses au changement de points de vue, de sujets et de setups avec les différentes configurations CS, CV et CP.

En résumé, la modalité squelette fournit des informations sur la structure corporelle. Celles-ci sont robustes aux changements d'échelle et d'illumination, et peuvent être invariantes aux points de vue de la caméra ; elles sont donc pertinentes pour représenter les comportements humains. Néanmoins, la HAR utilisant des données squelettiques fait toujours face à des défis, en raison du bruit lié à l'acquisition des données. Par conséquent, certains des travaux existants sur la HAR se sont concentrés sur la fusion des modalités de données présentées précédemment permettant de bénéficier de leurs avantages et leurs capacités ; ces approches sont donc discutées dans la section suivante.

I.5.4 Approches de HAR basées sur des données Multimodales

L'apprentissage automatique multimodal est une approche de modélisation visant à traiter et à relier les informations sensorielles de plusieurs modalités (RGB, Squelettes ou Profondeur). En agrégeant les avantages et les capacités de diverses modalités de données, l'apprentissage automatique multimodal peut souvent fournir une HAR plus robuste et plus précise. Par conséquent, plusieurs méthodes multimodales de HAR ont été proposées.

I.5.4.1 Approches de HAR segmentée

Dans les travaux de (Wang, et al. 2020), les auteurs ont proposé un réseau hybride pour la HAR à partir de modalités multiples (RGB et Profondeur). Le réseau est construit sur des images dynamiques (Dynamic Images - DI) qui transforme une séquence vidéo en une ou plusieurs images dynamiques encodant à la fois les informations spatiales et temporelles, puis applique un réseau CNN pour classer les images dynamiques. Le réseau hybride proposé exploite efficacement les forces des approches émergentes basées sur les réseaux CNNs et les réseaux RNNs pour relever spécifiquement les défis qui surviennent dans la HAR. Les caractéristiques extraites par les deux sous-réseaux sont fusionnées via une analyse canonique des corrélations, puis transmises à une machine à vecteurs de support (SVM) linéaire pour la classification.

Dans (Song, Lan, et al. 2018), les auteurs ont proposé un framework pour la HAR avec des données multimodales. Une procédure d'apprentissage des caractéristiques indexées sur les données squelettes est développée pour exploiter davantage les caractéristiques locales des vidéos RGB et des flux optiques. En particulier, le framework proposé est construit sur la base d'un réseau CNN et d'un réseau RNN à mémoire court et long terme (LSTM). Une couche de transformation indexée sur le squelette est conçue pour extraire automatiquement les caractéristiques visuelles autour des articulations clés. Plusieurs schémas de fusion sont explorés pour exploiter les données multimodales. L'architecture proposée est entraînée de bout en bout et intègre différentes modalités pour apprendre des représentations de caractéristiques d'une

manière efficace.

Dans les travaux de (Cai, et al. 2020), les auteurs ont introduit un réseau GCN à deux flux, nommé JOLO-GCN (voir Figure 1.49), prenant respectivement des données squelettiques et des patchs de flux (Joint-aligned optical Flow Patches - JFP) alignés sur les articulations obtenues à partir de vidéos RGB comme entrées pour la HAR. Plus précisément, ils ont utilisé des patchs de flux optiques (JFP) centrés sur les articulations afin de capturer les mouvements locaux autour de chacune d'elle. Ensuite afin d'apprendre l'évolution cinétique encodée par les JFP, les auteurs utilisent un réseau GCN (voir Figure 1.49), comme architecture d'apprentissage. Comparé à la HAR basée sur des données squelettes pures, ce réseau hybride a amélioré efficacement les performances de HAR, tout en maintenant un faible coût de calcul et de mémoire.

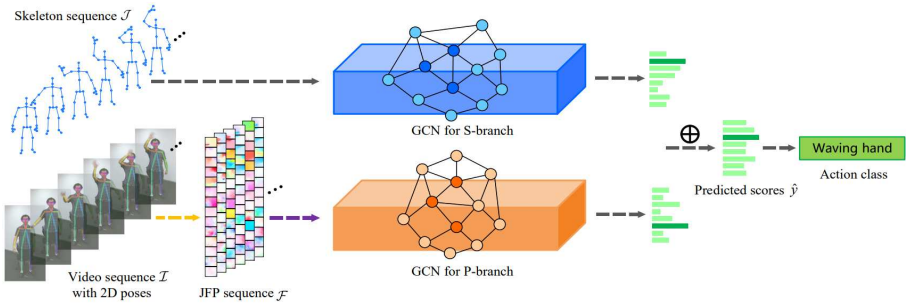


FIGURE 1.49 - Framework JOLO-GCN (CAI, ET AL. 2020)

I.5.4.2 Approches de HAR en ligne

Dans les travaux de (J. Liu, et al. 2019), les auteurs ont proposé un nouveau réseau basé sur des données multimodales et appelé « Multi-Modality Multi-Task RNN » pour la HAR en ligne en se basant sur leurs anciens travaux portant sur l'architecture JCR-RNN (Li, et al. 2016) présentée dans la partie I.5.3.2. Une version multitâche du réseau a été proposée (C. Liu, Y. Li, et al. 2017) afin de pouvoir détecter plusieurs actions en même temps. Enfin, dans (J. Liu, et al. 2019), l'idée de traiter à la fois des données squelette et des données RGB a été proposée. En effet, les données squelettes sont précises tout en étant peu nombreuses à traiter à chaque image ce qui permet d'avoir des systèmes de HAR en temps réel rapides et efficaces. Cependant, les données RGB offrent également des informations importantes pour la HAR car elles permettent de prendre en compte ce avec quoi est en train d'interagir la personne (un outil, un objet transporté, etc.). Le fait de combiner les deux types de données permet ainsi d'augmenter le panel d'actions reconnaissables ainsi que la précision du système. Le nouveau système proposé combine donc deux JCR-RNN fonctionnant en parallèle comme illustré dans la Figure 1.50. Ainsi, le JCR-RNN des données RGB est simplement précédé d'un CNN chargé d'extraire les caractéristiques visuelles, et le JCR-RNN de données squelettes est quant à lui précédé d'un réseau chargé d'extraire les caractéristiques de mouvement depuis les articulations significatives des différentes actions.

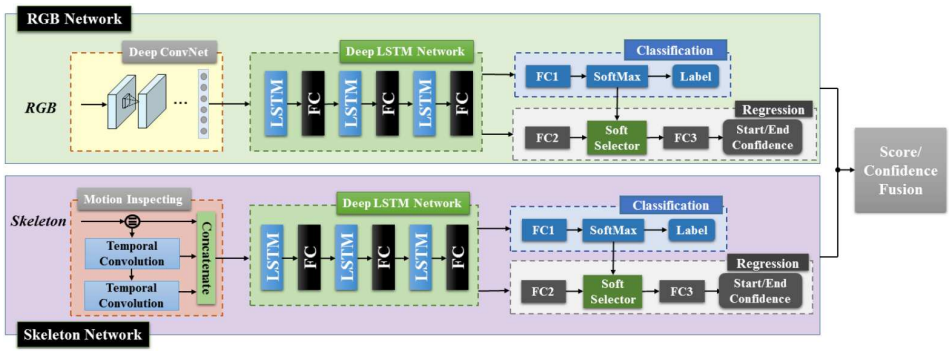


FIGURE 1.50 - Framework Multi-Modality Multi-Task RNN (J. Liu, et al. 2019)

En regroupant les avantages et les capacités de diverses modalités de données, les méthodes de HAR multimodales peuvent souvent fournir des performances plus robuste et plus précise vu qu'elles se servent des informations d'apparence riches du contexte de la scène capturée et agrègent des caractéristiques différentes de chaque modalité qui sont très discriminantes et puissantes pour la HAR. Cependant, l'intégration d'informations provenant de deux modalités ou plus peut mener à des problèmes de sur-apprentissage (Overfitting) se traduisant par le fait que le modèle de HAR soit trop adapté aux données d'apprentissage et ne se généralise pas à de nouvelles données qui lui sont inconnues.

I.5.5 Synthèse des approches de HAR

Le Tableau 1.5 montre une synthèse des approches de HAR étudiées et fournit leurs performances ainsi que leurs points forts et leurs limites.

TABLEAU 1.5 - Synthèse des approches de HAR étudiées.

Méthode	Modalité	En ligne/ Segmenté	Jeux de données	Résultats[6]	+Avantages et - limites
MAN (Xie, et al. 2018)	Squelettes	Segmenté	NTU RGB+D	82.67% (CS), **93.22% (CV)**	+ Les modules d'attention TARM et STCM permettent au modèle de se concentrer davantage sur les articulations discriminantes de manière adaptative. - Le modèle d'attention spatiale est adapté pour des scénarios segmentés (reconnaissance segmentée) et non pas en ligne. En
			HDM05	98.85%	
			SYSU-3D	87.63%	
			UT-Kinect	**100.0%**	

[6] La méthode la plus performante par jeu de donnée est indiquée en gras.

Méthode	Modalité	En ligne/ Segmenté	Jeux de données	Résultats[6]	+Avantages et - limites
					plus, le modèle est complexe dû à l'utilisation des réseaux de neurones profonds ResNet.
(C. Li, P. Wang, et al. 2017)	Squelettes	Segmenté	NTU RGB+D	**82.89% (CS)**, 90.10% (CV)	+ Le geste est synthétisé dans 3 images projetés dans les 3 plans 3D et permet de caractériser le geste de manière spatiotemporelle. - Les caractéristiques du domaine temporel (TPF) sont extraites à partir des Cartes de Trajectoires Articulaires (JTM) qui nécessitent un calcul intensif pour les générer.
ST-GCN (Yan, Xiong et Lin 2018)	Squelettes	Segmenté	NTU RGB+D	81.5% (CS), 88.3% (CV)	+ Le ST-GCN est capable d'exploiter le schéma local et la corrélation des squelettes. - Le modèle ST-GCN marche très bien dans des scénarios segmentés (reconnaissance segmentée) et n'est pas adapté pour la reconnaissance en ligne.
			Kinetics	30.7% (Top-1), 52.8% (Top-5)	
STA-LSTM (Song, Lan, et al. 2017)	Squelettes	Segmenté	SBU	91.51%	+ Les modules d'attention permettent au modèle de se concentrer davantage sur les articulations discriminantes de manière adaptative. - Le modèle d'attention spatiale a tendance à ignorer quelques articulations importantes.
			NTU RGB+D	73.4 % (CS), 81.2% (CV)	
JTM (P. Wang, W. Li et C. Li, et al. 2016)	Squelettes / Images 2D	Segmenté	MSRC-12	93.12%	+ Les JTM contiennent des caractéristiques discriminantes pour la reconnaissance d'actions humaines en temps réel (début et fin de l'action, vitesse, etc.) - Les Cartes de Trajectoires Articulaires (JTM) nécessitent un calcul intensif pour les générer.
			G3D	94.24%	
			UTD-MHAD	85.81%	
(Hou, et al. 2018)	Spectres optiques du	Segmenté	G3D	**95.45%**	+ La représentation spectrale permet d'utiliser des architectures

Méthode	Modalité	En ligne/ Segmenté	Jeux de données	Résultats[6]	+Avantages et - limites
	squelette		MSRC-12	**94.27%**	CNN standard pour apprendre les caractéristiques dynamiques appropriées à partir de séquences de squelettes sans avoir à réentraîner des millions de paramètres ce qui est utile lorsque les données vidéo d'apprentissages labélisées sont insuffisantes. - Les données squelettes ne sont pas encodées de bout en bout avec les CNN.
SFAM (P. Wang, W. Li, et al. 2017)	RGB-D	Segmenté	ChaLearn LAP	36.27%	+ Auto-étalonnage permettant d'estimer les flots de scène à partir de données RGB-D non enregistrées. - Quelques représentations des SFAM n'ont pas pu converger de façon constante.
			M2I	89.4% (SV), 91.2%(FV) et 87.6% (SV->FV), 76.5% (FV->SV)	
(K. Wang, et al. 2014)	RGB-D	Segmenté	CAD120	81.2%	+ Propose un modèle profond structuré qui peut gérer une grande variance intra-classe avec les structures latentes. - Le modèle n'intègre pas les informations sémantiques de haut niveau pour traiter les actions ou les évènements complexes avec des intentions sous-jacentes.
			OA1	60.1%	
			OA2	45.0%	
(Luvizon, Tabia et Picard 2017)	Squelettes	Segmenté	MSR-Action3D	97.1%	+ Propose des caractéristiques locales spatiales et temporelles permettant d'extraire les informations les plus pertinentes pour la classification. - La partie de combinaison des fonctionnalités dans l'apprentissage nécessite un calcul complexe ce qui rend le système moins adaptable aux applications en temps réel.
			UTKinect-Action3D	98.00%	
			Florence 3D Actions	94.39%	

Méthode	Modalité	En ligne/ Segmenté	Jeux de données	Résultats[6]	+Avantages et - limites
JCR-RNN (Li, et al. 2016)	Squelettes	En ligne	G3D	73.50% (Combattre) 96.70% (Golf)	+ Le framework JCR-RNN est capable de déterminer automatiquement le début et la fin de chaque action sans la nécessité d'utiliser de fenêtre glissante ou d'avoir une perspective explicite en avant ou en arrière de l'action. - Le framework JCR-RNN semble reconnaitre les actions qui sont spatialement séparées par l'emplacement des squelettes plutôt que par les mouvements des actions elles-mêmes (pour le jeu de données OAD). - Résultats de toutes les actions non révélés pour le jeu de données G3D.
			OAD	65.30%	
MM-MT RNN (J. Liu, et al. 2019)	RGB + Squelettes + Flux optiques	En ligne	G3D	86.00% (Combattre) 100% (Golf) 82.90% (Tennis) 100% (Bowling) 62.70% (FPS) 100% (Conduite) 98.60% (Divers)	+ La combinaison des deux types de données RGB et squelettes permet d'augmenter le panel d'actions reconnaissables ainsi que la précision du système. - Le framework MM-MT RNN ne prête pas attention à la description explicite des interactions homme-objet, ce qui conduit à manquer un grand potentiel de modélisation et de différenciation de certaines actions similaires.
			OAD	**79.50%**	
TRN (Xu, et al. 2018)	RGB	En ligne	HDD	40.80%	+ Possibilité d'estimer les caractéristiques visuelles à venir en fonction de celles déjà observées. - D'autres modèles temporels pouvant améliorer les performances de HAR comme les GRUs et les TCNs n'étaient pas explorés.
			TVSeries	83.70%	
			THUMOS'14	47.20%	
Action4DNet (You et Jiang 2018)	Profondeur	En ligne	Action4D	84.10%	+ Reconnaissance d'actions de plusieurs personnes, dans une représentation 4D encombrée. + Méthode invariante aux points

Méthode	Modalité	En ligne/ Segmenté	Jeux de données	Résultats[6]	+Avantages et - limites
					de vue de la caméra. - La génération des cubes 4D nécessite un calcul complexe.

Nous avons présenté dans les sections précédentes la reconnaissance d'actions humaines dans son cadre global en exposant quelques approches de HAR de la littérature. Dans ce qui suit nous allons nous concentrer sur l'application de la HAR dans l'industrie.

I.6 Reconnaissance des actions humaines dans l'industrie

Comme présenté dans la section contexte, les innovations en matière de systèmes de production se produisent plus rapidement que jamais. Les travailleurs humains doivent fréquemment apprendre de nouvelles méthodes et acquérir de nouvelles compétences. Dans des systèmes de production évoluant rapidement, les industries possédant des systèmes de production capables de comprendre le comportement des travailleurs humains et d'évaluer leurs performances opérationnelles en temps quasi réel obtiendront de meilleurs résultats que leurs homologues. C'est pourquoi la reconnaissance d'actions humaines dans les milieux industriels peut servir amplement cet objectif (Al-Amin, et al. 2019).

Dans (Roitberg, et al. 2014), les auteurs ont présenté une approche de reconnaissance d'activité pour des applications dans un contexte d'interaction humain-robot dans un milieu industriel. L'approche proposée est basée sur des caractéristiques spatiales et temporelles extraites de données squelettiques de travailleurs humains effectuant des tâches d'assemblage. Ces caractéristiques ont été utilisées pour entraîner un framework d'apprentissage automatique, qui classifie les images temporelles avec des Forêts Aléatoires (RF) et modélise ensuite les dépendances temporelles avec un Modèle de Markov Caché (HMM). Un jeu de données a été aussi créé où plusieurs travailleurs humains ont été invités à effectuer des tâches d'assemblage dans un environnement nécessitant des interactions humain-robot. Bien que leur méthode ait obtenu de bonnes performances de HAR (précision moyenne de 73%) sur la majorité des activités, elle ne semble pas très bien gérer les mouvements qui se chevauchent, ce qui entraîne quelques échecs de classification, surtout si les actions se sont produites au début ou à la fin de l'activité correspondante, ce qui peut être crucial pour les applications dans le contexte de l'interaction humain-robot en milieu industriel.

Dans (Patalas-Maliszewska, Halikowski et Damaševičius 2021), un framework pour générer des instructions sur le lieu de travail et pour la reconnaissance d'activités des travailleurs a été

proposé. La méthode proposée utilise les réseaux CNN, SVM et CNN basés sur des régions pour reconnaître et vérifier les tâches effectuées par les opérateurs. Les auteurs ont analysé les enregistrements vidéo du processus de travail, puis les images correspondant aux étapes de l'activité de travail ont été déterminées. Ensuite, les caractéristiques et les objets liés au processus de travail ont été extraites à partir de ces images à l'aide d'un réseau CNN avec un classifieur SVM. La génération automatique d'instructions et la reconnaissance en temps réel des activités des travailleurs présentées dans cet article forment un sujet intéressant qui peut soulager les travailleurs de tâches pénibles et diminuer les risques liés à la sécurité, mais aussi augmenter la production en permettant des profits plus élevés. Bien que ce travail soit principalement axé sur la reconnaissance d'activités des opérateurs qui implique une interaction homme-objet permanente, de telles informations contextuelles n'ont pas été incorporées dans leur framework lors de l'apprentissage, ce qui peut créer plus de caractéristiques à apprendre et donc améliorer de manière significative les performances de HAR.

Dans (Tao, et al. 2018), les auteurs ont proposé une méthode de reconnaissance d'activités des travailleurs humains dans la fabrication intelligente dans laquelle ils ont utilisé des capteurs inertiels (IMU) et des signaux d'électromyographie de surface (sEMG). Ils ont concaténé les signaux IMU pour créer une image de signal qui est, en appliquant une transformation de Fourier discrète, transformée en une image d'activité. Les images générées sont ensuite transmises à un réseau CNN pour l'extraction de caractéristiques. Ils ont également publié un jeu de données de reconnaissance d'activités des travailleurs qui a été utilisé pour tester le modèle CNN développé. Ils ont obtenu des précisions élevées atteignant 98 % et 87 % sur deux expériences. Dans la première, ils ont mélangé les données de manière aléatoire, puis une moitié du jeu de données a été utilisée pour l'apprentissage et l'autre moitié pour le test. Dans la deuxième expérience, ils ont utilisé 7 des 8 échantillons des participants pour l'apprentissage et l'échantillon restant pour le test. Bien que la précision moyenne rapportée dans cet article soit élevée, ce qui peut s'expliquer par le fait qu'il n'y a que 6 activités simples réalisées dans l'ensemble du jeu de données, le pourcentage de classification erronée pour certains opérateurs était assez élevé. En outre, le même protocole d'acquisition de données a été suivi par tous les participants, ce qui rend les scénarios de validation inter-sujets (cross-subject) difficiles à employer.

Ces dernières années, comme nous l'avons présenté dans la section I.4, de nombreux jeux de données de référence ont été créés pour faciliter le développement et l'évaluation de nouveaux algorithmes de HAR. Néanmoins, ces jeux de données ne représentent pas des activités dans un contexte industriel (assemblage, maintenance, …) et ils sont pour la plupart capturés avec un petit nombre d'échantillons d'actions, ce qui entrave le développement d'algorithmes de plus haut niveau pour les applications du monde réel. En outre, les variations dans la façon dont les actions sont exécutées entre les différents opérateurs sont subtiles, ce qui rend la labélisation de ces données complexe. Pourtant, les applications du monde réel nécessitent des algorithmes qui généralisent bien avec différentes personnes, contextes, vues ou autres facteurs environnementaux

(Y. Ji, et al. 2019). Par conséquent, le fait de disposer d'une grande quantité de données labélisées rend les modèles d'apprentissage profond plus robustes face à ces variabilités, améliorant ainsi les performances des modèles et la précision de la HAR. C'est pourquoi, la nécessité de synthétiser et de générer des données pour entraîner les modèles d'apprentissage profond a conduit les chercheurs à utiliser différentes techniques telles que des outils de traitement d'image ou des représentations virtuelles de systèmes physiques utilisant les jumeaux numériques pour attaquer les problèmes de vision par ordinateur, en particulier le problème de la HAR (Dandekar, Zen et Bressan 2017).

I.7 Conclusion et problématiques

Durant ce chapitre, après un positionnement de l'industrie 4.0, nous avons recensé un état de l'art sur la collaboration humain-robot et la reconnaissance d'actions humaines. Nous avons étudié les différentes modalités des données (RGB, Profondeur, Squelettes et Multimodales) utilisées pour la HAR. Cette étude a renforcé notre choix pour les données squelettes en tant que modalité principale car elles sont efficaces et robustes. De plus, vu notre contexte de création d'un jeu de données dans un environnement industriel contraint par plusieurs facteurs dont des contraintes éthiques, le choix de la modalité squelette est le plus pertinent. Par la suite, nous avons exposé les jeux de données de HAR les plus utilisés dans la littérature. Par ailleurs, l'étude a révélé que la majorité des jeux de données présentent des actions de la vie quotidienne, effectuées seules ou en interaction avec une autre personne, ou liées à la santé. Nous constatons l'absence de jeux de données de reconnaissance d'actions humaines effectuées dans un contexte industriel, limitant ainsi l'usage de la reconnaissance d'actions humaines et de la collaboration humain-robot dans l'industrie.

Par conséquent, dans le deuxième chapitre, nous nous intéressons à cette problématique en proposant un jeu de données d'actions humaines industrielles qui peut aider la communauté de la recherche à progresser dans la HAR dans les environnements industriels, facilitant ainsi la collaboration humain-robot. Lors de la création de ce jeu de données, l'un des principaux problèmes rencontrés était l'étape de la labélisation, au cours de laquelle nous passons en revue l'ensemble des données enregistrées et assignons manuellement chaque action effectuée par un opérateur à l'une des classes d'action prédéfinies. Cette tâche est très délicate en raison de la précision et des erreurs possibles de labélisation des données, car elle est effectuée manuellement. Comme nous sommes dans un contexte de l'industrie 4.0 et de la HRC dans lesquels l'utilisation de la simulation et du jumeau numérique couplé à la réalité virtuelle se développe de plus en plus, des avancés remarquables peuvent être exploitées dans plusieurs scénarios et peuvent résoudre plusieurs problèmes reconnus. Cela nous amène donc à la problématique du Chapitre 3 dans lequel nous avons étudié la possibilité de créer un jeu de données basé sur le jumeau numérique. Nous présentons une méthodologie qui utilise les jumeaux numériques pour générer des données étiquetées d'une manière automatique utilisées pour le processus de HAR basé sur

un algorithme d'apprentissage profond. Le jumeau numérique développé simule un flux de travail d'assemblage dans un poste de travail industriel.

L'avancement des méthodes récentes de HAR dans la littérature par les différentes modalités de données a été aussi souligné dans ce chapitre pour mieux comprendre la progression de la recherche dans ce domaine. Nous avons étudié les deux types de reconnaissance d'actions ; segmentée et en ligne. Nous avons exposé les avantages de la reconnaissance d'actions en ligne en exprimant les différents champs d'application possibles et les défis par rapport à la reconnaissance d'actions segmentée. L'état de l'art sur les algorithmes d'apprentissage profond mené nous montre que le choix le plus pertinent pour la reconnaissance d'actions humaines en ligne est d'utiliser des données squelettes et de surcroît utiliser des réseaux convolutifs à graphes comme les articulations du squelette humain peuvent être facilement représentés par des nœuds du graphe et leurs connexions spatiales par des arrêtes. Dans le quatrième chapitre, nous proposons donc une approche à fenêtre glissante et à vote majoritaire utilisant les réseaux de neurones convolutifs à graphe spatio-temporel permettant d'acquérir une meilleure précision et robustesse avec des données en flux continu pour la reconnaissance d'actions en temps réel.

Chapitre 2

Création d'un jeu de données de HAR dans un contexte industriel pour la HRC

Sommaire

II.1 Introduction

Dans le premier chapitre, nous avons exposé les jeux de données de HAR les plus utilisés dans la littérature. Cependant, ces jeux de données existants ne proposaient pas de

panel d'actions effectuées dans un contexte industriel et comprennent simplement des actions quotidiennes ou bien liées à la santé ce qui limite l'usage de la HAR dans l'industrie. Par conséquent, dans ce chapitre nous proposons un jeu de données d'actions humaines industrielles nommé « Industrial Human Action Recognition Dataset (InHARD) » (Dallel, Havard et Baudry, et al. 2020). Ce dernier présente des actions réalisées dans un contexte industriel, plus précisément sur un poste d'assemblage manuel assisté par un bras cobotique. Ce jeu de données peut aider la communauté de la recherche à progresser dans la HAR dans les environnements industriels, facilitant ainsi la collaboration humain-robot. La suite de ce chapitre est organisée comme-suit : tout d'abord le protocole expérimental d'acquisition est présenté, puis une étude statistique des données acquises est détaillée, ensuite l'architecture de HAR ST-GCN (Yan, Xiong et Lin 2018) est utilisée pour obtenir les premiers résultats sur ce jeu de données InHARD.

II.2 Protocole expérimental

II.2.1 Objectifs de l'activité

Afin de faciliter la Collaboration Humain-robot dans les milieux industriels, le jeu de données InHARD est créé sur la base d'un cas d'utilisation réel dans un environnement industriel. Ce cas d'utilisation implique un assemblage de diverses pièces et composants (voir Tableau 2.1), réalisé sur différentes étapes à l'aide du bras robotique Universal Robot UR10 pour assembler des vélos d'enfants. Le cobot UR10 est un bras robotique industriel collaboratif, conçu pour des tâches à grande échelle, permettant d'automatiser des processus et des tâches tels que l'emballage, la palettisation, l'assemblage et le pick and place. Ce poste collaboratif fait partie d'un atelier flexible de production (voir Figure 2.1) et correspond à une partie des opérations d'assemblage réalisées.

FIGURE 2.1 - Plateforme industrie et atelier flexible de production

Le processus de création du jeu de données InHARD est donné par la Figure 2.2. La première étape consiste à équiper les opérateurs par une combinaison permettant de récupérer leurs données squelettes durant toute l'acquisition des données. Les opérateurs ont été aussi filmés par des caméras placées dans 3 emplacements différents pour capturer les données RGB avec trois vues différentes (gauche, droite et dessus). La Figure 2.3 présente un participant durant l'acquisition des données ainsi que les flux vidéos obtenus. La deuxième étape permet de

synchroniser les flux vidéos enregistrés en même temps que les données squelettes puisque ces données sont capturées par deux capteurs différents et avec une fréquence d'acquisition différente (voir Figure 2.2). Ensuite, les données acquises sont prétraitées. La dernière étape consiste à parcourir toutes les données acquises et attribuer des labels pour chaque segment d'action réalisée.

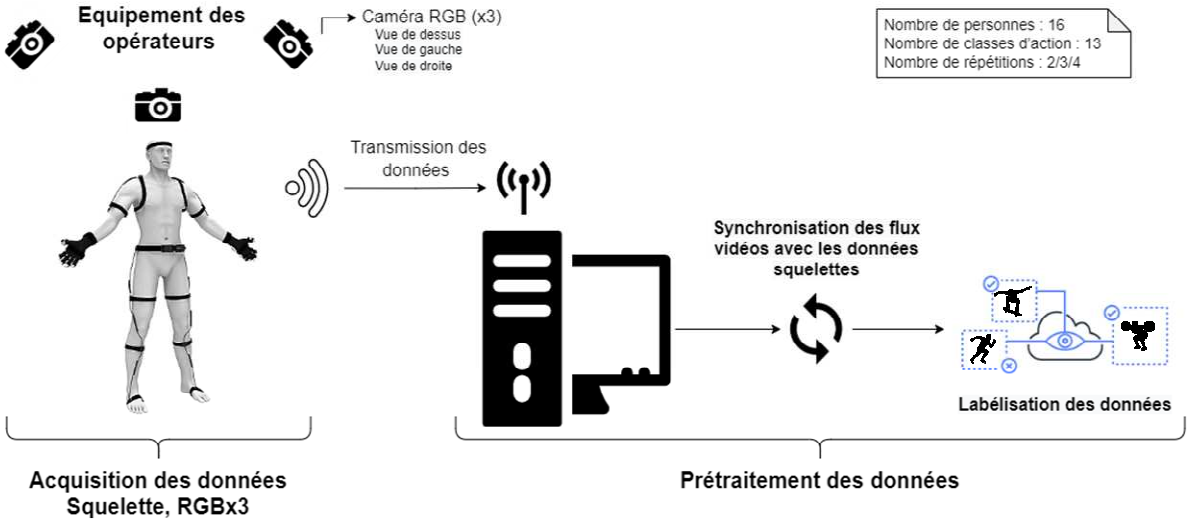


FIGURE 2.2 - Processus de création du jeu de données InHARD

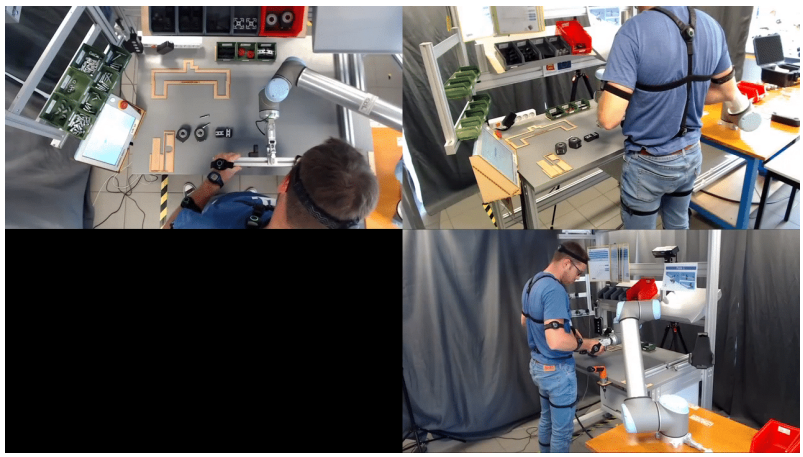


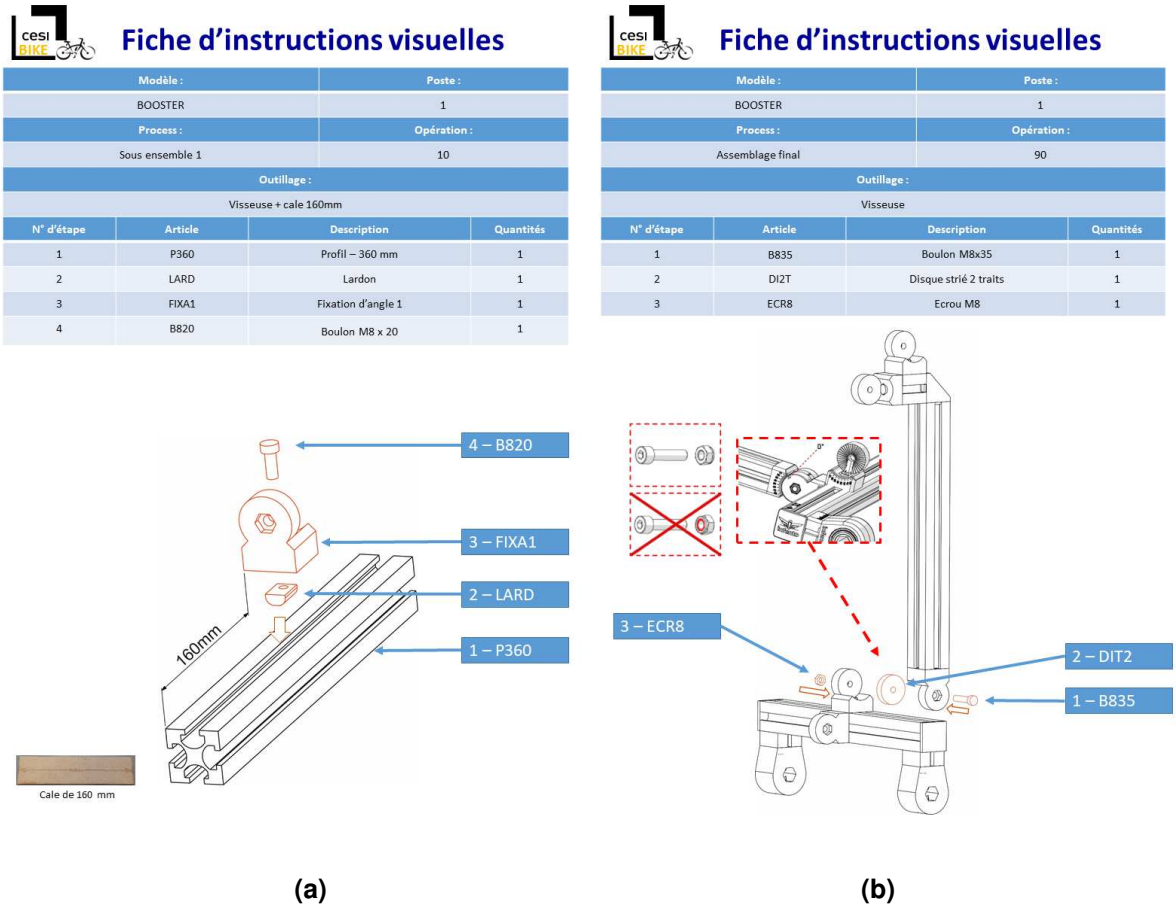
FIGURE 2.3 - Illustration de l'acquisition du jeu de données InHARD

Afin d'aider la communauté de recherche à aller plus loin dans la collaboration humain-robot dans les milieux industriels, nous mettons ce jeu de données à la disposition du public. Ce dernier comprend toutes les modalités de données acquises. Afin de faciliter la manipulation et l'utilisation du jeu de données, nous fournissons également un dataframe avec toutes les informations nécessaires, y compris les noms des fichiers, les participants, les opérations, les étiquettes d'actions, le début/la fin des actions, leurs durées, etc.

Dans cette partie, nous présentons les détails du jeu de données InHARD proposé, y compris

les modalités de données et les différents capteurs utilisés tout au long de la collecte de données. Nous détaillons ensuite les classes d'action identifiées, les participants et les différentes vues capturées.

Pour la collecte de ce jeu de données, les participants sont invités à suivre et consulter des fiches d'instructions afin de choisir les objets convenants et d'assembler les bons composants pour obtenir un sous-système final comme le montre la Figure 2.4. Le reste des fiches d'instructions visuelles sont présentées dans l'Annexe - Fiches d'instructions visuelles dans InHARD.



Fiche d'instructions visuelles

Modèle :	Poste :
BOOSTER	1
Process :	**Opération :**
Sous ensemble 1	10
Outillage :	
Visseuse + cale 160mm	

N° d'étape	Article	Description	Quantités
1	P360	Profil – 360 mm	1
2	LARD	Lardon	1
3	FIXA1	Fixation d'angle 1	1
4	B820	Boulon M8 x 20	1

Fiche d'instructions visuelles

Modèle :	Poste :
BOOSTER	1
Process :	**Opération :**
Assemblage final	90
Outillage :	
Visseuse	

N° d'étape	Article	Description	Quantités
1	B835	Boulon M8x35	1
2	DI2T	Disque strié 2 traits	1
3	ECR8	Ecrou M8	1

(a) **(b)**

FIGURE 2.4 - Première et dernière opération de la manipulation d'assemblage dans le jeu de données InHARD. (L'ensemble des fiches est disponible en Annexe - Fiches d'instructions visuelles dans InHARD)

La configuration initiale de l'installation est exposée dans la Figure 2.5.

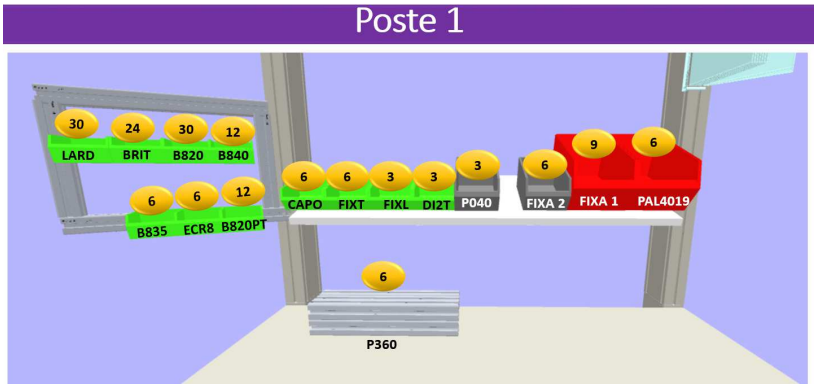


FIGURE 2.5 - Configuration initiale de l'installation

Les composants intervenants dans l'activité sont données par le Tableau 2.1.

TABLEAU 2.1 - Description des articles utilisées dans la manipulation

Article	Description
P360	Profil – 360 mm
LARD	Lardon
FIXA1	Fixation d'angle 1
B820	Boulon M8 x 20
CAPO	Capot
FIXT	Fixation en T
PAL4019	Bloc de palier 40x40x19
B840	Boulon M8 x 40
BRIT	Bride en T
FIXL	Fixation en L
B820PT	Boulon M8 x 20 (petite tête)
FIXA2	Fixation d'angle 2
P040	Profil – 40 mm
B820PT	Boulon M8 x 20 (petite tête)
DI2T	Disque strié 2 traits
ECR8	Écrou M8
B835	Boulon M8 x 35

II.2.2 Participants

Pour la collecte des données, nous avons invité 16 participants distincts (4 femmes et 12 hommes) pour effectuer une tâche d'assemblage d'un système industriel à l'aide du bras cobotique UR10. La tranche d'âge des participants dans notre jeu de données InHARD va de 23 à 43 ans, leurs tailles varient de 1.60m jusqu'à 1.95m. Les compétences en assemblage des participants ont été évaluées. Par conséquent, le temps d'assemblage pour une personne expérimentée varie de 4 minutes et 7 secondes pour le plus court à 14 minutes et 45 secondes pour le plus long. Ces particularités des participants apportent une variation plus réaliste à chacune des actions ce qui permettra d'évaluer la robustesse des architectures de HAR à la variabilité d'exécution des actions. Afin d'anonymiser les données, un numéro d'identification unique est attribué à chaque

participant. Chacun effectue la tâche deux à quatre fois selon son niveau de compétence et en fonction de sa performance, l'opérateur met plus ou moins de temps à exécuter les tâches d'assemblage. Un temps pour chaque participant a donc été défini. Ainsi, nous nous assurons que toutes les actions soient bien représentées mais aussi pour avoir un nombre suffisant d'actions dans chaque classe car lors de la procédure complète, certaines actions n'étaient effectuées que une à deux fois.

II.2.3 Modalités des données

Pour notre jeu de données, nous avons recueilli des données en utilisant deux modalités de données par le biais de différents capteurs.

II.2.3.1 Modalité du squelette

Nous avons utilisé un système de capture de mouvements « Combinaison Perception Neuron 32 Edition v2 » pour acquérir les données du squelette avec taux d'échantillonnage égal à 120 Hz. Ce système de capture de mouvements utilise des centrales inertielles (IMU). Cela consiste en effet à équiper une personne de capteurs de mouvements aux différents points clefs du corps afin de suivre les coordonnées de ces différents points au fil du temps. Chaque capteur correspond alors à une « jointure » des données squelette, c'est-à-dire un des points de contrôles du squelette. Les données du squelette comprennent les positions 3D (Tx, Ty et Tz) des 17 principales articulations du corps détectées et suivies durant toute la scène, ainsi que les 3 rotations autour de chaque axe (Rx, Ry et Rz), comme le montre les Figure 2.6 et Figure 2.7.

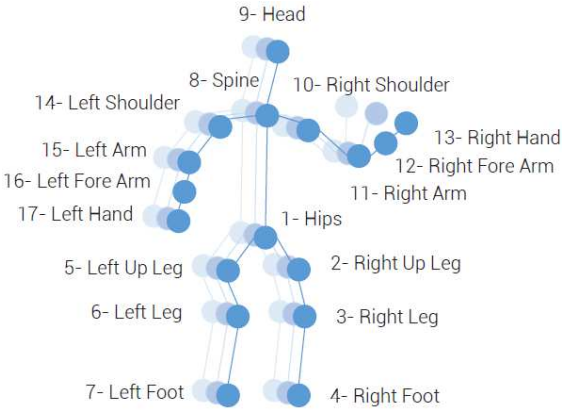


FIGURE 2.6 - Configuration des 17 articulations du corps dans le jeu de données InHARD

Les données squelettes sont enregistrées dans des fichiers au format BVH. Le nom BVH signifie « Biovision Hierarchical Data » (Meredith et Maddock 2001). Il fournit des informations sur la hiérarchie du squelette en plus des données de mouvement. Le format BVH est un excellent format global qui inclut les décalages en translation des segments enfants par rapport à leur parent. La structure hiérarchique du fichier BVH est donnée par la Figure 2.7.

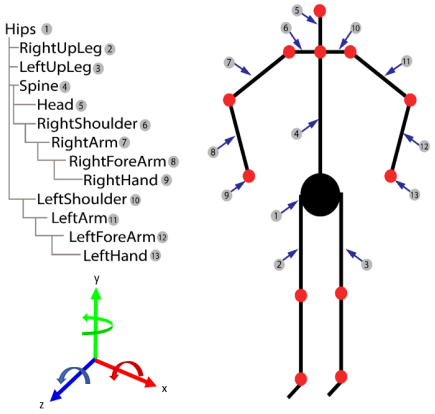


FIGURE 2.7 - Structure hiérarchique du fichier BVH

II.2.3.2 Modalité vidéo

Nous avons également utilisé trois caméras « Logitech C920 » couvrant trois points de vues différents (vue de haut, gauche et droite) pour capturer des données RGB. Les flux vidéo des trois caméras sont assemblés dans un fichier vidéo RGB capturé avec une résolution de 1280x720 et un taux de rafraîchissement égal à 30 fps. Pour chaque configuration, deux caméras ont été placées à la même hauteur mais à deux angles horizontaux différents : -45° et +45° pour capturer les côtés gauche et droite. La troisième caméra est placée au-dessus des participants pour capturer la vue de dessus.

La caméra 1 observe toujours la vue de dessus et est affichée dans le quart supérieur gauche de la vidéo RGB. La caméra 2 observe la vue du côté gauche et s'affiche sur le quart supérieur droit de la vidéo RGB. Enfin, la caméra 3 observe la vue du côté droit et est affichée dans le quart inférieur droit de la vidéo RGB comme le montre la Figure 2.8.

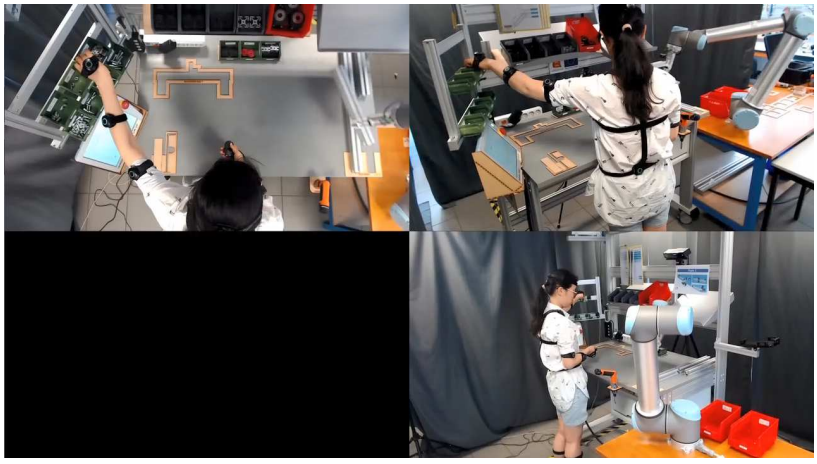


FIGURE 2.8 - Les 3 vues des caméras capturant la scène (vue de haut, gauche et droite)

II.2.4 Classes d'actions

Nous avons identifié 14 classes d'actions haut niveau et 72 classes d'actions bas niveau là où

les étiquettes sont beaucoup plus précises. Par exemple pour l'action haut niveau « Picking en face », ses équivalents bas niveau sont « Attraper le Profil P040 », « Attraper la Fixation FIXA1 », « Attraper la Fixation FIXA2 » ou bien « Attraper le Bloc de palier PAL4019 ». Pour la suite de notre travail, nous utiliserons les classes d'actions de haut niveau décrites dans le Tableau 2.2.

TABLEAU 2.2 - Classes d'actions identifiées dans le jeu de données InHARD

ID Action	**Label d'action (Haut niveau)**	**Label d'action (Bas niveau)**
0	Aucune action	Aucune action
1	Consulter fiches	Consulter fiches
2	Tourner fiches	Tourner fiches
3	Attraper visseuse	Attraper visseuse
4	Poser visseuse	Poser visseuse
5	Picking en face	Attraper le Profil P040 Attraper la Fixation FIXA1 Attraper la Fixation FIXA2 Attraper le Bloc de palier PAL4019
6	Picking gauche	Attraper la Fixation FIXT Attraper la Fixation FIXL Attraper le Capot CAPO Attraper l'écrou ECR8 Attraper le Boulon B835 Attraper le Boulon B840 Attraper le Boulon B820 Attraper le Lardon LARD Attraper le Boulon B820PT Attraper la Bride BRIT
7	Attraper Toise	Attraper Toise
8	Poser Toise	Poser Toise
9	Prendre BTR	Prendre BTR
10	Poser BTR	Poser BTR
11	Prendre composant	Attraper la Fixation FIXT Attraper la Fixation FIXL Attraper le Capot CAPO Attraper l'écrou ECR8 Attraper le Boulon B835 Attraper le Boulon B840 Attraper le Boulon B820 Attraper le Lardon LARD Attraper le Boulon B820PT Attraper la Bride BRIT
12	Poser composant	Poser la Fixation FIXT Poser la Fixation FIXL Poser le Capot CAPO Poser l'écrou ECR8 Poser le Boulon B835

ID Action	Label d'action (Haut niveau)	Label d'action (Bas niveau)
		Poser le Boulon B840
		Poser le Boulon B820
		Poser le Lardon LARD
		Poser le Boulon B820PT
		Poser la Bride BRIT
		Poser la Fixation FIXT
		Poser la Fixation FIXL
		Poser le Capot CAPO
		Poser l'écrou ECR8
		Poser le Boulon B835
		Poser le Boulon B840
		Poser le Boulon B820
		Poser le Lardon LARD
		Poser le Boulon B820PT
		Poser la Bride BRIT
13	Assembler système	Placer LARD sur le Profil P360-1
		Placer FIXA1 sur LARD à 160mm
		Visser FIXA1 avec B820
		Placer LARD sur P360-1 (sur l'extrémité)
		Placer CAPO sur les 2 extrémités du P360-1
		Placer BRIT1 et BRIT2 sur les deux côtés du P360-2
		Placer FIXL sur l'extrémité de P360-2
		Placer FIXA2 sur l'autre extrémité de P360-2
		Visser FIXL avec B820PT
		Visser FIXA2 avec B820
		Placer BRIT1 et BRIT2 sur P040
		Placer P040 sur P360-2
		Visser P040 avec B820PT
		Placer LARD dans P040 (P360-2)
		Placer FIXA1 sur P360-2
		Placer FIXA2 sur P360-2
		Placer DI2T sur P360-2
		Placer ECR8 dans DI2T
		Visser P360-2 avec B835

L'action 0 (Aucune action) représente toute action réalisée par l'opérateur qui n'est pas liée à la tâche d'assemblage confiée. La tâche d'assemblage sur ce poste cobotique comporte 7 opérations. Chaque opération implique entre 6 et 18 actions. Pendant toute la manipulation, un opérateur effectue au minimum 100 si son niveau de compétence en assemblage est élevé et donc le montage est réalisé sans erreur ni d'actions inutiles et 180 au maximum pour une personne novice. La Figure 2.9 montre la distribution du nombre d'actions effectuées par chaque participant ainsi que la durée totale des actions.

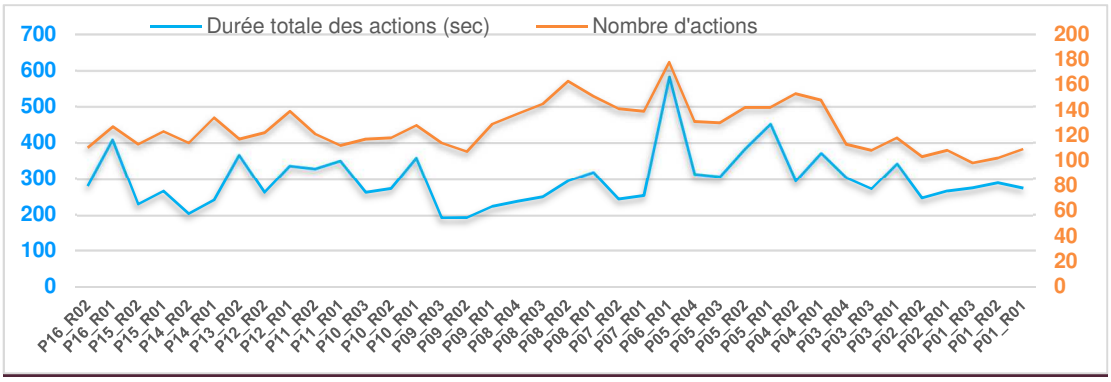


FIGURE 2.9 - Graphique croisé dynamique du nombre d'actions et leurs durées dans le jeu de données InHARD

Dans la Figure 2.10, nous représentons la distribution de la durée totale des actions pour tous les participants. Les actions durent entre 0.5 et 26.9 secondes. Comme le montre la Figure 2.10, l'action « Assembler le système » a une variance élevée car elle se produit lorsque l'opérateur assemble plusieurs pièces. Par conséquent, elle peut prendre quelques secondes sur certaines étapes de l'opération et plusieurs dizaines de secondes sur d'autres en fonction du nombre d'actions à effectuer et notamment celles comportant des actions de vissage.

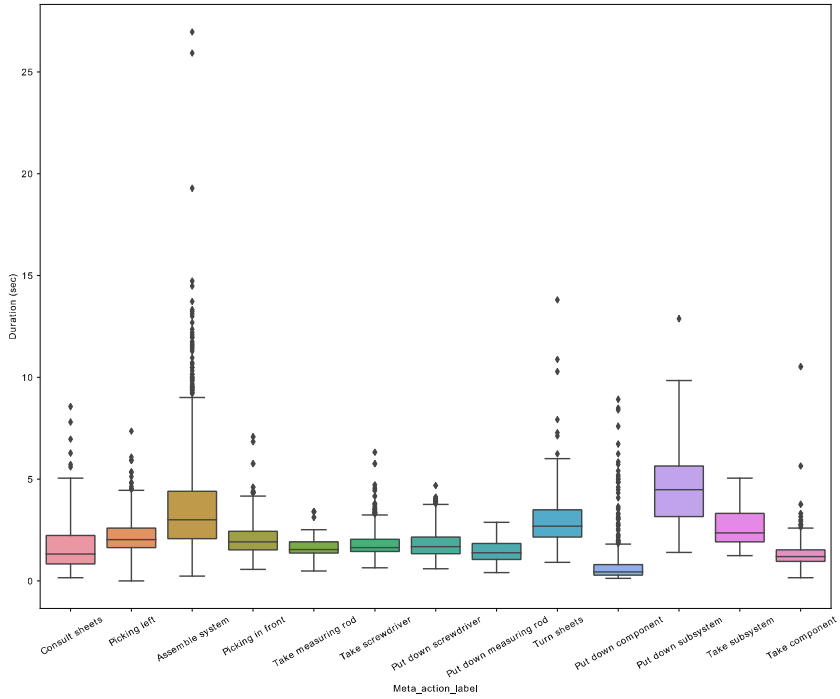


FIGURE 2.10 - Distribution de la durée des actions dans le jeu de données InHARD

II.2.5 Enchainement des actions

Pendant la manipulation, les opérateurs sont amenés à attraper des composants et certaines actions doivent être faites les unes à la suite des autres pour avoir le bon système final et d'autres

actions peuvent être réalisées sans ordre préférentiel. Notons par « && » l'ordre d'attrapage des objets entre deux actions d'une opération donnée qui doit être respectée (avec contrainte d'ordre) et par « || » l'ordre d'attrapage des objets qui peut être interchangeable entre les actions d'une telle opération (sans contrainte d'ordre). Par exemple, pour une opération « O1 » donnée incluant 3 actions différentes A, B et C qui sont toutes avec contrainte d'ordre, le déroulement de O1 doit se passer uniquement comme suit :

```
O1 : A && B && C ➔ 1) Action A puis 2) Action B puis 3) Action C
```

Pour une opération « O2 » donnée incluant 3 actions différentes A, B et C qui sont toutes sans contrainte d'ordre, le déroulement de O2 peut se dérouler selon plusieurs scénarios différents mais conduisant au même produit final comme suit :

```
S1-O2 : A || B || C ➔ 1) Action A puis 2) Action B puis 3) Action C
                                        ou
S2-O2 : A || B || C ➔ 1) Action B puis 2) Action A puis 3) Action C
                                        ou
S3-O2 : A || B || C ➔ 1) Action C puis 2) Action B puis 3) Action A
                                        …
```

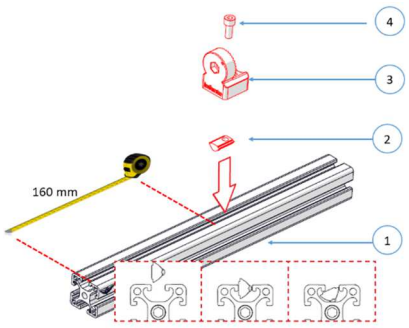
NB : Nous pouvons trouver des actions avec et sans contrainte d'ordre dans la même opération. Par exemple, pour une opération « O3 » donnée incluant 3 actions différentes A, B et C dont A et B sont sans contrainte d'ordre et C avec contrainte d'ordre, le déroulement de O3 peut se passer seulement en deux scénarios conduisant au même produit final comme suit :

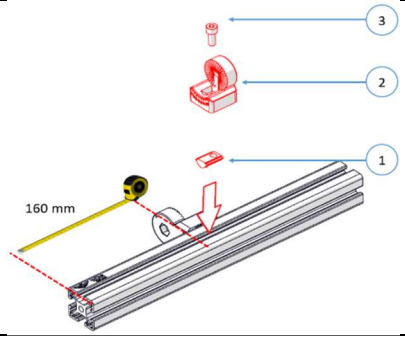
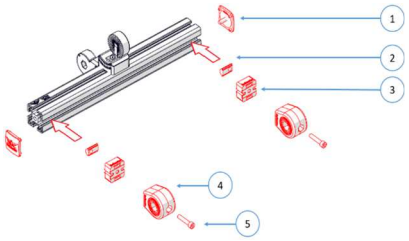
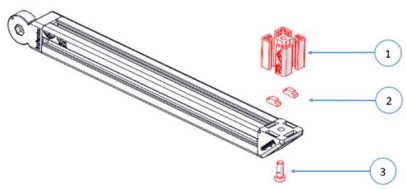
```
S1-O3 : A || B && C ➔ 1) Action A puis 2) Action B puis 3) Action C
                                        ou
S2-O3 : A || B && C ➔ 1) Action B puis 2) Action A puis 3) Action C
```

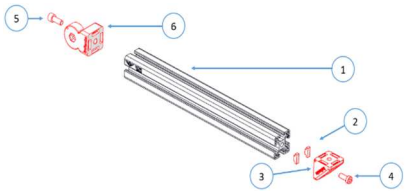
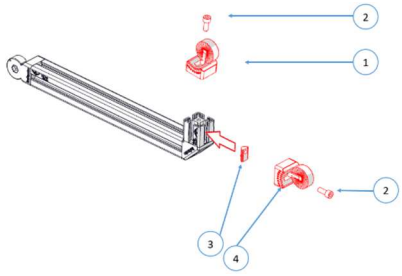
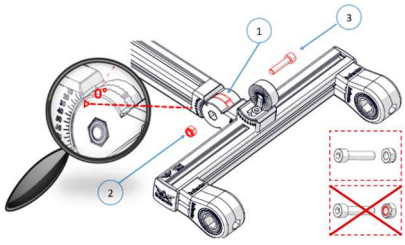
II.2.6 Labélisation des actions

Le Tableau 2.3 exprime une description détaillée des différentes opérations de la manipulation dans le jeu de données InHARD y inclut les acteurs impliqués (opérateur ou cobot), l'ordre de déroulement des actions et les fiches d'instructions visuelles.

TABLEAU 2.3 - Description des différentes opérations de la manipulation dans InHARD

Opération	Actions	Acteur	Ordre	Description
OP01 : Assembler FIXA1 avec P360-1 (Face 1)	- Récupérer le Profil P360-1 - Récupérer le Lardon LARD - Récupérer la Fixation FIXA1 - Récupérer le Boulon B820 - Placer LARD dans P360-1 - Placer FIXA1 sur LARD - Visser FIXA1 avec B820	Cobot Opérateur Opérateur Opérateur Opérateur Opérateur Opérateur	(||) (||) (||) (||) (&&) (&&) (&&)	

Opération	Actions	Acteur	Ordre	Description
Sortie OP01 : P360-1				
OP02 : Assembler FIXA1 avec P360-1 (Face 2)	- Récupérer le Lardon LARD - Récupérer la Fixation FIXA1 - Récupérer le Boulon B820 - Placer LARD dans P360-1 - Placer FIXA1 sur LARD - Visser FIXA1 avec B820	Opérateur Opérateur Opérateur Opérateur Opérateur Opérateur	(||) (||) (||) (&&) (&&) (&&)	
Sortie OP02 : P360-1				
OP03 : Assembler FIXT, CAPO & PAL4019 avec P360-1	- Récupérer le Capot CAPO (x2) - Récupérer le Lardon LARD (x2) - Récupérer la Fixation FIXT (x2) - Récupérer le Bloc de palier PAL4019 (x2) - Récupérer le Boulon B840 (x2) - Placer LARD dans P360-1 (x2) - Placer FIXA1 sur LARD (x2) - Visser FIXA1 avec B820 (x2) - Placer CAPO sur P360-1(x2)	Opérateur Opérateur Opérateur Opérateur Opérateur Opérateur Opérateur Opérateur Opérateur Opérateur	(||) (||) (||) (||) (||) (&&) (&&) (&&) (||)	
Sortie OP03 : P360-1				
OP04 : Assembler FIXT, CAPO & PAL4019 avec P360-2	- Récupérer le Profil P360-2 - Récupérer la Fixation FIXL - Récupérer le Boulon B820PT - Récupérer le Boulon B820 - Récupérer la Fixation FIXA2 - Placer FIXL sur P360-2 - Visser FIXL avec B820PT - Placer FIXA2 sur P360-2 - Visser FIXA2 avec B820	Cobot Opérateur Opérateur Opérateur Opérateur Opérateur Opérateur Opérateur Opérateur	(||) (||) (||) (||) (&&) (&&) (&&) (&&)	
Sortie OP04 : P360-2				

Opération	Actions	Acteur	Ordre	Description
OP05 : Assembler P040 avec P360-2	- Récupérer le Profil P360-2 - Récupérer le Profil P040 - Récupérer la Bride BRIT (x2) - Récupérer le Boulon B820PT - Placer les BRITS sur P040 - Placer P040 sur P360-2 - Visser P040 avec B820PT	Cobot Opérateur Opérateur Opérateur Opérateur Opérateur Opérateur	(|||) (|||) (|||) (|||) (&&) (&&) (&&)	
Sortie OP05 : P360-2				
OP06 : Assembler FIXA1 avec P040	- Récupérer la Fixation FIXA1 - Récupérer le Boulon B820 (x2) - Récupérer le Lardon LARD - Récupérer la Fixation FIXA2 - Placer LARD dans P040 - Placer FIXA1 sur P360-2 - Visser FIXA1 avec B820 - Placer FIXA2 sur P360-2 - Visser FIXA2 avec B820	Opérateur Opérateur Opérateur Opérateur Opérateur Opérateur Opérateur Opérateur Opérateur	(|||) (|||) (|||) (|||) (&&) (&&) (&&) (|||) (&&)	
Sortie OP06 : P360-2				
OP07 : Assembler P360-2 avec P360-1	- Récupérer P360-1 - Récupérer P360-2 - Récupérer le Disque DI2T - Récupérer l'Ecrou ECR8 - Récupérer le Boulon B835 - Placer DI2T sur P360-2 - Placer ECR8 dans DI2T - Visser P360-2 avec B835	Cobot Opérateur Opérateur Opérateur Opérateur Opérateur Opérateur Opérateur	(|||) (|||) (|||) (|||) (|||) (&&) (&&) (&&)	
Sortie OP07 : P360-2 assemblé avec P360-1 (Cadre inférieur)				

Afin d'étiqueter les actions, nous avons utilisé ANVIL (Kipp, et al. 2014), un outil de recherche libre permettant de synchroniser et d'étiqueter des données vidéo, audio et squelettique (au format BVH). Cet outil permet de spécifier un schéma de codage formel qui sert de plan pour le codage spécifique au projet. Le processus de codage est mené sur des pistes parallèles alignées dans le temps.

Afin d'utiliser le même formalisme pour l'ensemble des données, un fichier d'annotations contenant toutes les actions de toutes les opérations a été produit. Un extrait du fichier de formalisation d'annotations créé pour le jeu de données InHARD est illustré dans la Figure 2.11.

```
<?xml version="1.0" encoding="ISO-8859-1" standalone="no"?>
<annotation-spec>
  <head>
    <valuetype-def>
      <valueset name="InHARD_Annotations">
        <value-el color="green">[OP010] Turn sheets</value-el>
        <value-el color="green">[OP010] Consult sheets</value-el>
        <value-el color="green">[OP010] Catch Fixture key LARD</value-el>
        <value-el color="green">[OP010] Catch Fixation FIXA1</value-el>
        <value-el color="green">[OP010] Catch Bolt B820</value-el>
        <value-el color="green">[OP010] Put down component</value-el>
        <value-el color="green">[OP010] Catch component</value-el>
        <value-el color="green">[OP010] Place LARD on Profile P360-1</value-el>
        <value-el color="green">[OP010] Take measuring rod</value-el>
        <value-el color="green">[OP010] Place FIXA1 on LARD at 160mm</value-el>
        <value-el color="green">[OP010] Put down measuring rod</value-el>
        <value-el color="green">[OP010] Take screwdriver</value-el>
        <value-el color="green">[OP010] Screw FIXA1 with B820</value-el>
        <value-el color="green">[OP010] Put down screwdriver</value-el>
        <value-el color="magenta">[OP020] Turn sheets</value-el>
        <value-el color="magenta">[OP020] Consult sheets</value-el>
        <value-el color="magenta">[OP020] Catch Fixture key LARD</value-el>
        <value-el color="magenta">[OP020] Catch Fixation FIXA1</value-el>
        <value-el color="magenta">[OP020] Catch Bolt B820</value-el>
        <value-el color="magenta">[OP020] Put down component</value-el>
        <value-el color="magenta">[OP020] Catch component</value-el>
```

FIGURE 2.11 - Extrait du fichier d'annotations contenant les différents labels des actions dans InHARD

Une fois le fichier de formalisation d'annotations créé, les fichiers vidéo et de données squelettes BVH sont importés dans l'outil ANVIL. Il est alors possible d'annoter avec les labels prédéfinis dans le fichier de formalisation. La Figure 2.12 montre un exemple de labélisation d'une séquence du jeu de données InHARD.

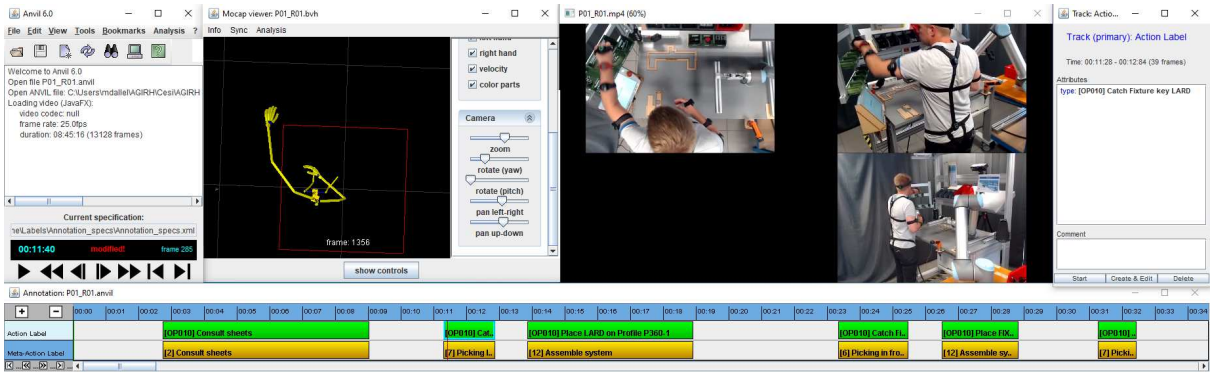


FIGURE 2.12 - Labélisation des actions dans le jeu de données InHARD avec l'outil ANVIL. En vert le label d'action (bas niveau), en jaune le label d'action (haut niveau).

Une fois la labélisation terminée, le projet est sauvegardé dans un fichier avec le format « .anvil ». Ce fichier contient toutes les informations de la séquence concernée incluant les fichiers vidéo/BVH correspondants, les labels des actions ainsi que le temps de début et de fin de chaque action.

Pour capturer tous les flux de scène ensemble, nous avons utilisé le logiciel OBS Studio qui est un logiciel libre et Open Source d'enregistrement vidéo et de streaming en direct. Puisque la fréquence d'acquisition entre le flux vidéo et les données squelettes diffère, un top de synchronisation au début et à la fin de l'acquisition a été mis en place pour recaler les deux flux.

Pour cela le participant effectue un clap de mains qui sera utilisé par la suite pour synchroniser les deux flux.

II.2.7 Synthèse du jeu de données InHARD

Une synthèse du jeu de données InHARD est donnée par le Tableau 2.4.

TABLEAU 2.4 - Synthèse du jeu de données InHARD

Nb. participants	Nb. occurrences	Nb. classes	Nb. vues	Modalités	Type d'actions	Capteurs
16	4808	14	3	Squelette + RGB	Industrielles	Perception Neuron V32 + 3 Caméras C920

Le jeu de données InHARD fournit pour chaque récurrence effectuée par chaque participant :

- Un fichier vidéo (ext : mp4)
- Un fichier de données squelettes au format BVH (ext : bvh)

La convention de nommage des données que nous avons utilisée est expliquée ci-dessous.

InHARD segmenté :

Ex: D01_P01_R01_A01_Tstart_Tfin.ext

InHARD en ligne :

Ex: P01_R01.ext

Où :

« D » désigne le jour de la capture de données

« P » désigne l'opérateur qui réalise l'action

« R » désigne le numéro de récurrence ou de répétition

« A » désigne l'identifiant de l'action effectuée

« Tstart » désigne le temps de début de l'action effectuée

« Tfin » désigne le temps de fin de l'action effectuée

Par ailleurs InHARD fournit un fichier au format .csv qui rassemble toutes les informations liées au jeu de données InHARD y compris : les noms des fichiers vidéos RGB et BVH de chaque opération, le numéro de l'opérateur, l'opération en cours, le label et le numéro de l'action réalisée, le temps en frames et en secondes de début et de fin de chaque action (RGB + Squelette) et la durée de chaque action. La Figure 2.13 montre un extrait du fichier .csv fourni avec le jeu de données InHARD.

	A	B	C	D	E	F	H	I	J
1	File	Subject	Operation	Recurrence	Action_label	Meta_action_label	Action_start_rgb_sec	Action_end_rgb_sec	Action_start_rgb_frame
2	P01_R01	P01	OP010	R01	[OP010] Consult sheets	Consult sheets	2.72	9.0	82
3	P01_R01	P01	OP010	R01	[OP010] Catch Fixture key LARD	Picking left	11.28	12.84	338
4	P01_R01	P01	OP010	R01	[OP010] Place LARD on Profile P360-1	Assemble system	13.84	18.88	415
5	P01_R01	P01	OP010	R01	[OP010] Catch Fixation FIXA1	Picking in front	23.32	25.44	699
6	P01_R01	P01	OP010	R01	[OP010] Place FIXA1 on LARD at 160mm	Assemble system	26.48	28.8	794
7	P01_R01	P01	OP010	R01	[OP010] Catch Bolt B820	Picking left	31.24	32.4	936
8	P01_R01	P01	OP010	R01	[OP010] Take measuring rod	Take measuring rod	36.56	37.92	1096
9	P01_R01	P01	OP010	R01	[OP010] Take screwdriver	Take screwdriver	39.36	40.4	1180
10	P01_R01	P01	OP010	R01	[OP010] Screw FIXA1 with B820	Assemble system	41.2	44.56	1235
11	P01_R01	P01	OP010	R01	[OP010] Put down screwdriver	Put down screwdriver	44.76	47.24	1341
12	P01_R01	P01	OP010	R01	[OP010] Put down measuring rod	Put down measuring rod	47.28	49.12	1417

FIGURE 2.13 - Informations sur les actions dans le jeu données InHARD

II.3 Prétraitement des données

Le prétraitement des données est une étape primordiale dans l'apprentissage automatique, car la qualité des données et les informations utiles qui peuvent en être tirées influent directement sur la capacité d'apprentissage de notre modèle. Les données du monde réel contiennent généralement des bruits, des valeurs manquantes, et peuvent-être dans un format inutilisable qui ne peut pas être exploité directement pour les modèles d'apprentissage automatique. Le prétraitement des données permet de nettoyer ces données et de les rendre appropriées pour un modèle d'apprentissage automatique, ce qui augmente également la précision et l'efficacité du modèle. Dans cette partie, nous parlons des différents prétraitements que nous avons proposés pour préparer nos données.

II.3.1 Nettoyage des données

Par nettoyage des données, nous entendons le nettoyage des données qui est particulièrement effectué dans le cadre du prétraitement des données par exemple en remplissant les valeurs manquantes, en lissant les données bruitées, en résolvant les incohérences et en supprimant les valeurs aberrantes. Pour notre cas, cette étape consiste à éliminer les fichiers BVH là où les squelettes des opérateurs sont déformés. Dans la pratique, les centrales inertielles sont typiquement affectées d'une dérive de leurs estimations des positions et des rotations. Le fonctionnement de ces capteurs s'écarte des équations idéales à cause des erreurs qui affectent les mesures des rotations et des accélérations causées par le bruit des appareils à proximité, des facteurs d'échelle, des non-linéarités etc. ce qui engendrent des dérives au cours du temps. Un exemple des données squelettes distordues et pertinentes est donné par la Figure 2.14.

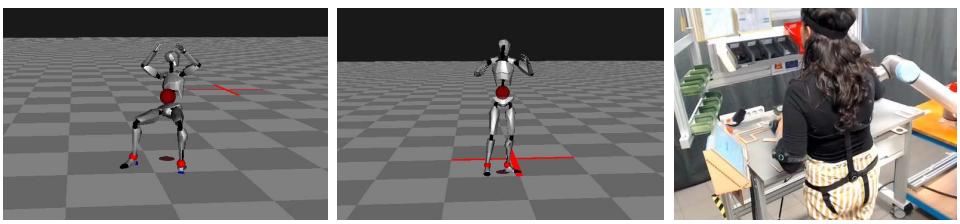


FIGURE 2.14 - Exemples des données squelettes distordues (à gauche) et pertinentes (squelette au milieu avec la personne sur le flux RGB à droite)

II.3.2 Mise à zéro des Hips

Le nœud racine dans un fichier BVH (Biovision Hierarchical Data), généré par le logiciel de capture de mouvement « Axis Neuron Pro » fourni avec le capteur « Combination Perception Neuron 32 Edition v2 » comprenant les données squelettiques, est l'articulation de la hanche. Comme la combinaison est basée sur des unités de mesure inertielle (IMU), elle a parfois tendance à dériver après un certain temps. Vu qu'il y a très peu de déplacements des opérateurs pendant la manipulation d'assemblage, pour contourner ce problème dans notre cas d'usage, une mise à zéro des positions et des rotations de l'articulation de la hanche est effectuée permettant au squelette d'être figé au point de départ initial (0,0) et reste orienté vers le poste de travail pendant toute l'opération sans affecter la dynamique des autres articulations et sans affecter la posture de la personne comme le montre la Figure 2.15..

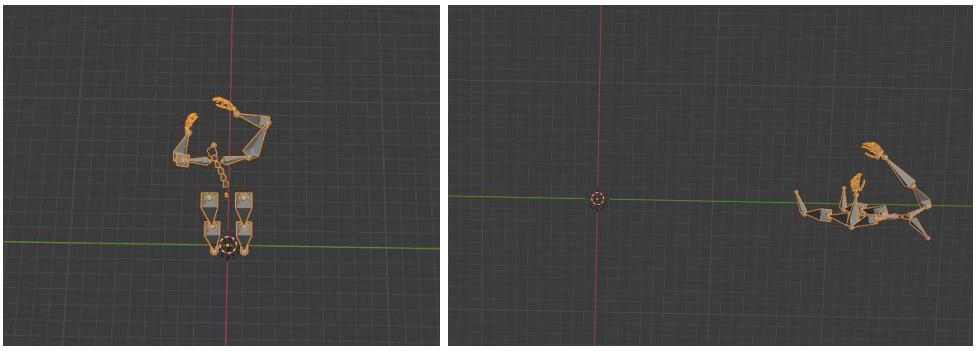


FIGURE 2.15 - Exemples des données squelettes corrigées avec le prétraitement de mise à zéro des Hips (gauche) et données squelettes sans correction (droite)

II.3.3 Ré-échantillonnage

Le prétraitement de ré-échantillonnage consiste à sur-échantillonner ou sous-échantillonner l'ensemble des actions pour les adapter à la taille de la longueur de fenêtre spécifiée. Etant donné la différence plus ou moins importante dans la durée des actions, le ré-échantillonnage des actions permettra de prendre l'action entière pendant l'apprentissage et pas seulement le début de l'action, si sa durée est plus grande que la taille de la longueur de fenêtre spécifiée et d'éviter la mise à zéro si sa durée est plus petite comme l'explique la Figure 2.16.

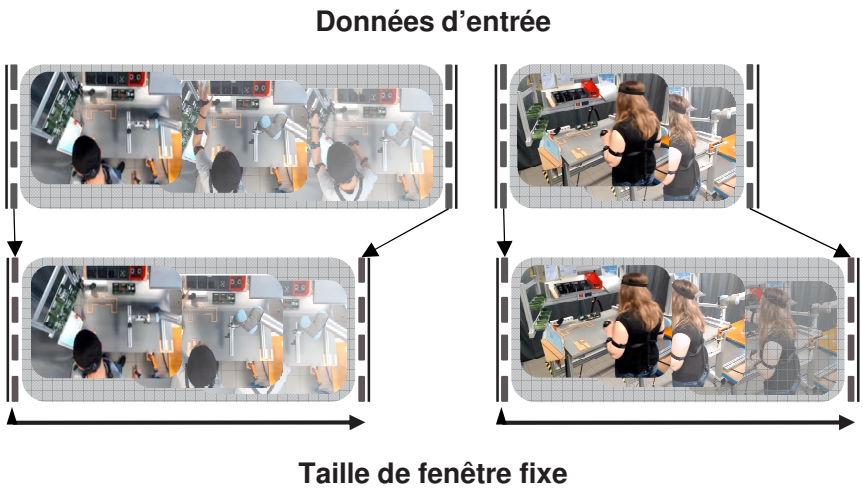


FIGURE 2.16 - Illustration du prétraitement de ré-échantillonnage appliqué sur les données d'entrée du jeu de données InHARD

Dans cette partie, nous avons introduit les différents prétraitements proposés pour le jeu de données InHARD. La prochaine étape consiste à les exploiter avec un algorithme sélectionné.

II.4 Algorithme ST-GCN pour la HAR

II.4.1 Rappel sur les modalités des données utilisées pour la HAR

Comme expliqué dans le Chapitre 1, les approches développées peuvent être regroupées en quatre catégories principales selon la modalité de données utilisée : les méthodes basées sur des données RGB, les méthodes basées sur les cartes de profondeur, les méthodes basées sur le squelette et les méthodes hybrides ou multimodales où au moins deux modalités sont combinées.

Les approches basées sur des données RGB sont les méthodes les plus basiques utilisées pour la reconnaissance des actions. Les caméras conventionnelles ont représenté les systèmes d'acquisition sur lesquels la plupart des méthodes ont été basées. En complément de la prédiction des actions à partir des données RGB, des informations telles que les postures ou les poses peuvent également être déduites des images ou des vidéos. Cependant, la grande variabilité des formes ou des postures humaines et l'encombrement des arrière-plans dans de telles données peuvent être un problème. Les caméras peuvent également être sensibles aux changements d'éclairage, aux occlusions et aux changements d'orientation. Par conséquent, pour être efficaces et performantes, les méthodes basées sur les données RGB nécessitent une grande quantité de données et de traitements, ce qui peut être une tâche fastidieuse (P. Wang, W. Li et P. Ogunbona, et al. 2018).

Les méthodes basées sur les cartes de profondeurs reposent sur l'extraction de descripteurs de caractéristiques représentatifs extraites à la main (hand-crafted) à partir des ensembles de points dans l'image/vidéo de profondeur. De telles caractéristiques peuvent être peu profondes et

également dépendantes du jeu de données. Ainsi, modéliser la dynamique d'une action est un enjeu important qui peut être résolu différemment selon les approches. Les caractéristiques du squelette peuvent être extraites des cartes de profondeur et conçues pour capturer à la fois des configurations spatiales et temporelles, mais ces caractéristiques sont extraites à la main (hand-crafted), ce qui entraîne un pouvoir expressif limité et des difficultés de généralisation.

Les approches basées sur le squelette 2D/3D sont devenues plus populaires ces dernières années en raison des avantages qu'elles présentent par rapport aux données RGB conventionnelles. Ces approches sont de plus en plus utilisées car elles sont efficaces, fiables puisqu'elles transmettent des informations plus représentatives compte tenu du fait qu'elles surmontent les arrière-plans encombrés et sont insensibles aux changements d'éclairage contrairement aux méthodes basées sur les données RGB. Les méthodes basées sur le squelette sont fiables pour estimer la silhouette du corps plus facilement et n'ont pas non plus besoin d'une énorme quantité de données pour effectuer la tâche de reconnaissance.

Les solutions hybrides ou multimodales combinent les informations d'au moins deux modalités de données (Squelette, RGB ou Cartes de profondeur) pour modéliser les actions ce qui peut améliorer les performances de la reconnaissance d'action.

Le Tableau 1.5 montre que les approches GCN (ou basées graphes) obtiennent de meilleures performances de reconnaissances d'actions que leurs équivalents basés RNN ou CNN. De plus, les données squelettiques sont plus adaptées à la reconnaissance d'actions en ligne car elles ont de faibles dimensions, ce qui réduit à la fois les temps d'apprentissage et de test, ce que nous verrons plus loin dans le Chapitre 4. C'est pourquoi, le choix le plus pertinent pour la reconnaissance d'actions serait d'utiliser des données squelettes et de surcroît utiliser des algorithmes de réseau à graphe convolutif.

II.4.2 Vue d'ensemble des GCN et des ST-GCN

Les réseaux de neurones (NN) ont connu un grand succès ces dernières années, au vu de leur large champ applicatif comme le traitement d'image, reconnaissance de caractères et de signatures etc. (Asadi-Aghbolaghi, et al. 2017). Les premières variantes de ces réseaux utilisaient des données régulières et euclidiennes (Dallel, Havard et Dupuis, et al. 2022). Cependant, la plupart des données du monde réel contiennent de manière sous-jacente des structures de graphe ; c'est pourquoi des réseaux de neurones à graphes (GNN) ont été mis en place (Bronstein, et al. 2017). Il existe différentes variantes de ces réseaux, les réseaux à graphes convolutifs (GCNs) étant l'une d'entre elles particulièrement utilisée dans la reconnaissance d'actions humaines (Yan, Xiong et Lin 2018) (Song, et al. 2021).

Les CNNs sont conçus pour fonctionner sur des données structurées euclidiennes régulières telles que des images ou des vidéos. Tandis que les GCNs, une version généralisée des CNNs,

sont utilisés avec des données irrégulières non euclidiennes où le nombre de connexions de nœuds varie et les nœuds ne sont pas ordonnés. Les GCNs effectuent des opérations de convolutions similaires à celles des CNNs, néanmoins, le modèle apprend les caractéristiques en inspectant les nœuds du graphe plutôt que les pixels de l'image/vidéo.

Plus formellement, un GCN est un réseau de neurones qui est basé sur des graphes. Un graphe G est représenté par un ensemble V de N nœuds $n_i \in V$ et un ensemble d'arêtes E où $e_{i,j} = (n_i, n_j)$. Étant donné un graphe $G = (V, E)$, un GCN prend en entrée une matrice de caractéristiques $N \times F_0$ (où F_0 est le nombre de caractéristiques d'entrée pour chaque nœud) et une représentation matricielle $N \times N$ de la structure du graphe généralement sous la forme d'une matrice d'adjacence A. Quant à la sortie, il génère une matrice de caractéristiques $N \times F$ (où F représente le nombre de caractéristiques de sortie par nœud).

Plus généralement, une couche cachée dans le GCN peut être écrite comme $h^i = f(h^{i-1}, A)$ où h^0 est la matrice de caractéristiques d'entrée et f représente la fonction de propagation. Chaque couche h^i est une matrice de caractéristiques $N \times F^i$ où chaque ligne est une représentation de caractéristiques d'un nœud. Les entités sont agrégées à chaque couche pour former les entités de la couche suivante à l'aide de la fonction de propagation f. A chaque couche consécutive, les caractéristiques deviennent de plus en plus abstraites.

Les informations de localité peuvent être obtenues en utilisant un graphe de calcul. En voyant comment chaque nœud est connecté à ses voisins et aux voisins de ses voisins, toutes les connexions possibles seront déterminées et un graphe de calcul sera ensuite formé. De cette façon, la structure ainsi que les informations sur les caractéristiques sont capturées. Les informations de localité seront par la suite agrégées. Cela se fait essentiellement à l'aide de réseaux de neurones comme le montre la Figure 2.17.

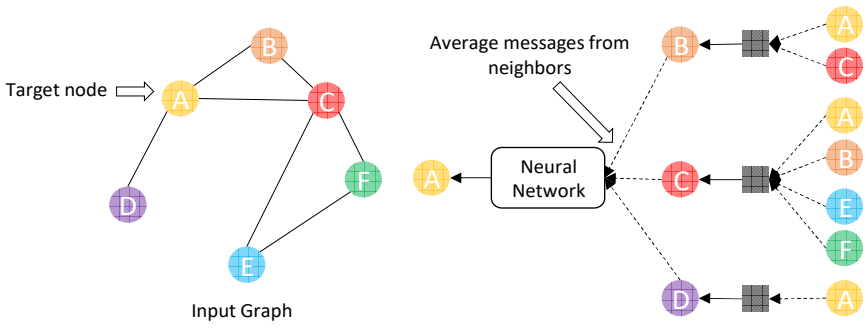


FIGURE 2.17 - Illustration d'un réseau de neurones à graphes (GNN)

Les GCNs eux-mêmes peuvent être classés en deux groupes principaux : les réseaux de neurones convolutifs à graphe spectral et les réseaux de neurones convolutifs à graphe spatio-temporel. Les premiers gèrent la localité de convolution du graphe sous la forme d'une analyse spectrale. Les réseaux de neurones convolutifs à graphe spatio-temporel utilisent l'opération de

convolution en l'appliquant sur les nœuds du graphe et leurs voisins.

En ce qui concerne la HAR, ce sont les réseaux de neurones convolutifs à graphe spatio-temporel qui sont principalement utilisés. Comme le décrit la Figure 2.18, l'opération consiste à apprendre une fonction de mappage f pour générer la représentation d'un nœud n_i en agrégeant ses propres caractéristiques x_i et les caractéristiques de ses voisins x_j dans l'espace et dans le temps.

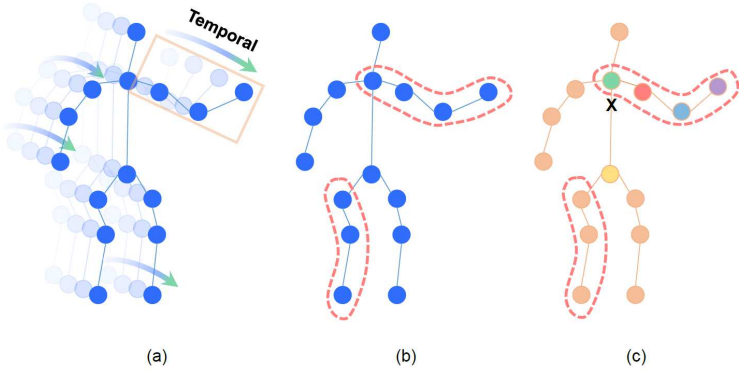


FIGURE 2.18 - a) Illustration du graphe spatio-temporel d'une séquence squelette indiquant le mouvement humain dans l'espace et dans le temps utilisé dans les ST-GCN. Les points bleus indiquent les articulations du squelette et les flèches représentent leur connexion dans des frames consécutifs. (b) Stratégie d'échantillonnage dans une couche de convolution pour une seule image. (c) Illustration de la stratégie de mappage.

Dans (Yan, Xiong et Lin 2018), une nouvelle approche a été proposée qui consiste à utiliser les Réseaux de Graphes Convolutionnels (GCN) à partir des données squelettes pour la reconnaissance d'actions segmentées. Ils ont proposé une nouvelle représentation des séquences squelettiques en étendant les réseaux convolutifs à graphes (GCN) au domaine spatio-temporel. En effet, les auteurs proposent de traiter différemment les données squelettiques. Ces dernières ne sont pas simplement une suite de frames où chacune d'elle présente un ensemble de coordonnées dans l'espace, mais les jointures sont reliées les unes aux autres selon un schéma spatio-temporel logique, celles des connexions naturelles du corps humain (hanche, buste, tête, épaules, coudes, poignet…). Chaque image de donnée squelette est donc un graphe 3D où les jointures sont les nœuds du graphe. De plus, chaque coordonnée est nommée, c'est une jointure que l'on va donc retrouver sur chacune des frames et que l'on peut suivre à travers le temps. Il est donc proposé de voir les données squelettes comme un graphe 4D, où chaque jointure est reliée aux jointures voisines, ex. le coude droit à l'épaule droite et au poignet droit, mais aussi reliée dans le temps à lui-même à l'image suivante et à l'image précédente, ex. le coude droit à l'instant t au coude droit à l'instant $t-1$ et à l'instant $t+1$.

Un nouveau modèle de squelettes dynamiques appelé réseaux à graphes convolutifs spatio-temporel (ST-GCN) est proposé dans (Yan, Xiong et Lin 2018), qui va au-delà des limites des méthodes précédentes en apprenant automatiquement les modèles spatiaux et temporels à partir

des données squelettes.

II.4.3 Explication de l'algorithme ST-GCN utilisé

Le framework utilisé pour traiter les données du graphe 4D est nommé GCN spatio-temporel (ST-GCN) (Yan, Xiong et Lin 2018). Celui-ci fonctionne de manière similaire à un CNN, c'est-à-dire qu'il traite les données avec des filtres de convolution qui vont traiter les nœuds du graphe par groupes locaux. Au cours d'une action donnée, les articulations se déplacent en groupes locaux, appelés « parties du corps », ce qui facilite leur modélisation. Cela est due au fait que les parties du corps restreignent la modélisation des trajectoires des articulations dans les « régions locales » par rapport à l'ensemble du squelette, formant ainsi une représentation hiérarchique des séquences du squelette. A partir des séquences telles que des coordonnées 2D ou 3D des jointures, un graphe spatio-temporel est construit où les articulations sont représentées par les nœuds du graphe, et la connectivité naturelle dans les structures du corps humain est représentée par les arrêtes du graphe (voir Figure 2.19). La construction se fait en deux étapes : (i) Premièrement, les articulations d'une image sont reliées par des arrêtes en fonction de la connectivité de la structure squelettique du corps humain. (ii) Ensuite, chaque articulation sera connectée à la même jointure dans l'image suivante. Ainsi, à la première couche du ST-GCN, chaque nœud est convolué avec ses nœuds limitrophes physiquement et temporellement dans le graphe. Puis les autres couches effectueront des convolutions en chaînes reprenant chaque fois les résultats de la couche précédente. En sortie, le système calculera un score de similitude pour chacune des actions connues. Ce score correspond au taux de reconnaissance de l'action dans les données du graphe en cours d'analyse. Le score le plus élevé correspond donc à l'action apprise en comparant le graphe le plus proche au graphe actuel. Le système va donc reconnaître l'action au score le plus élevé pour le graphe de l'instant t comme le décrit la Figure 2.19.

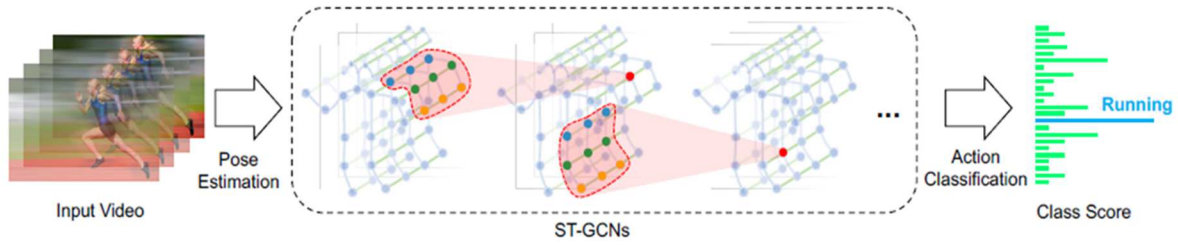


FIGURE 2.19 - Framework ST-GCN (Yan, Xiong et Lin 2018)

Sur la base des travaux de (Yan, Xiong et Lin 2018), nous avons utilisé le réseau de neurones convolutifs à graphe spatio-temporel pour la reconnaissance d'actions humaines segmentées sur le jeu de données d'actions humaines industrielles (InHARD) que nous avons créé.

Nous avons adapté le module ST-GCN pour qu'il puisse prendre en entrée des vecteurs de coordonnées des articulations du squelette qui peuvent être extraites de deux manières différentes :

1. La première consiste à utiliser les données squelettiques 3D du jeu de données InHARD brutes générées par les capteurs de mouvement utilisés comme le montre la Figure 2.20.

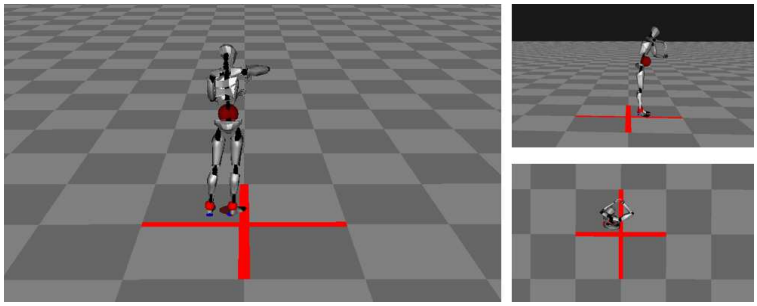


FIGURE 2.20 - Exemple des données squelettiques 3D dans le jeu de données InHARD

2. La deuxième consiste à extraire les données squelettiques 2D à partir des flux vidéos du jeu de données InHARD en utilisant le framework OpenPose (Cao, Hidalgo, et al. 2018) comme le montre la Figure 2.21. Ce dernier permet d'estimer l'emplacement de 18 articulations du squelette humain sur chaque image des clips vidéos. OpenPose fournit des coordonnées 2D (X, Y) dans le système de coordonnées des pixels et des scores de confiance C pour les 18 articulations. Chaque articulation est représentée par un 3-uplet (X, Y, C) et une image du squelette est enregistrée comme un tableau de 18 3-uplets. Les vecteurs de coordonnées des articulations du squelette ainsi que la fiabilité de l'estimation de la i-ème articulation d'une image t formeront les données d'entrée du module ST-GCN.

Compte tenu des séquences des articulations squelettiques, le module ST-GCN exploite la corrélation entre chaque articulation squelettique en construisant un graphe spatio-temporel non orienté $G = (V, E)$ sur une séquence squelettique avec N articulations et T images avec $V = \{v_{ti} | t = 1, .., T ; i = 1, .., N\}$ (V : nœud, E : arrête).

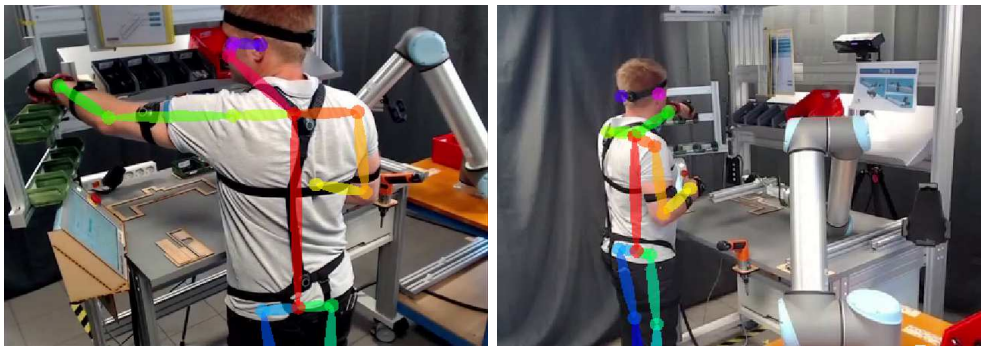


FIGURE 2.21 - Exemple d'extraction des données squelettiques 2D à partir des deux vues (Gauche et Droite) d'une vidéo extraite du jeu de données InHARD avec le framework OpenPose

II.5 Expérimentations

Dans cette section, nous évaluons les performances de HAR segmenté avec la méthode ST-

GCN sur notre jeu de données InHARD en utilisant les données squelettes. D'abord, nous comparons les résultats obtenus en utilisant les données squelettes 3D issues des capteurs de mouvement utilisés dans l'acquisition du jeu de données InHARD avec les données squelettes 2D extraites par le framework OpenPose à partir des clips vidéos. Par la suite, nous évaluons l'influence du prétraitement des données squelettes 3D sur les performances de l'algorithme ST-GCN. Enfin, nous présentons l'influence du type de référentiel utilisé (données de position ou données de rotation).

II.5.1 Environnement de travail

II.5.1.1 Environnement matériel

Cette section présente les configurations matérielles et logicielles de base utilisées. D'un point de vue matériel, toutes nos expérimentations sont effectuées en utilisant un serveur Ubuntu 18.04 avec une configuration matérielle ayant les caractéristiques suivantes :

- Processeur : Intel(R) Core (TM) i9-9980XEHQ @ 4.40 GH
- Cartes graphique : 4 x GeForce NVIDIA RTX 2080 (11 Go)
- Mémoire vive (RAM) : 128 Go

II.5.1.2 Environnement logiciel

Dans cette partie nous spécifions les bibliothèques que nous avons utilisé pour l'ensemble des expérimentations effectuées.

- pyTorch 1.2.0
- CUDA 10.2
- cuDNN 8.3.2

II.5.2 Discussions des résultats

II.5.2.1 Influence de la modalité squelette 2D et squelette 3D sur les performances du ST-GCN

Le Tableau 2.5 présente le comparatif des performances de HAR en utilisant les deux types des données squelettiques d'entrée (Données squelettes 3D / Données squelettes 2D extraites par OpenPose).

TABLEAU 2.5 - Accuracy et F1-Score moyennes sur le jeu de données InHARD en utilisant les données squelettes brutes et les données extraites par OpenPose

Type des données d'entrée	Accuracy	F1-Score
Données squelettes 3D	0.919	0.921
Données squelettes 2D (OpenPose)	**0.864**	**0.863**

Les Figure 2.22 et Figure 2.23 montrent les matrices de confusion du jeu de données InHARD

avec la méthode ST-GCN en utilisant les données squelettiques 2D extraites par OpenPose et les données squelettes 3D respectivement. Dans la Figure 2.22, la méthode ST-GCN a réussi de bonnes performances de HAR sur l'ensemble des actions, sauf pour l'action « Consulter fiches ». Cela peut être expliqué par le fait que cette action est très peu représentée dans le jeu de données InHARD et donc il y a peu d'instances pour l'apprentissage. En outre, l'action « consulter fiches » ne demande que très peu de mouvement du corps, seul la tête (ou le regard) s'oriente vers la fiche. Par conséquent, cette action est plus difficile à distinguer des autres. Dans la Figure 2.22, la méthode ST-GCN a réussi aussi de bonnes performances de HAR en moyenne sur l'ensemble des actions, sauf pour les deux actions « Prendre toise » et « Prendre sous-système » qui sont également très peu représentées dans le jeu de données InHARD car elles sont exécutées que 2 à 4 fois durant toute la manipulation.

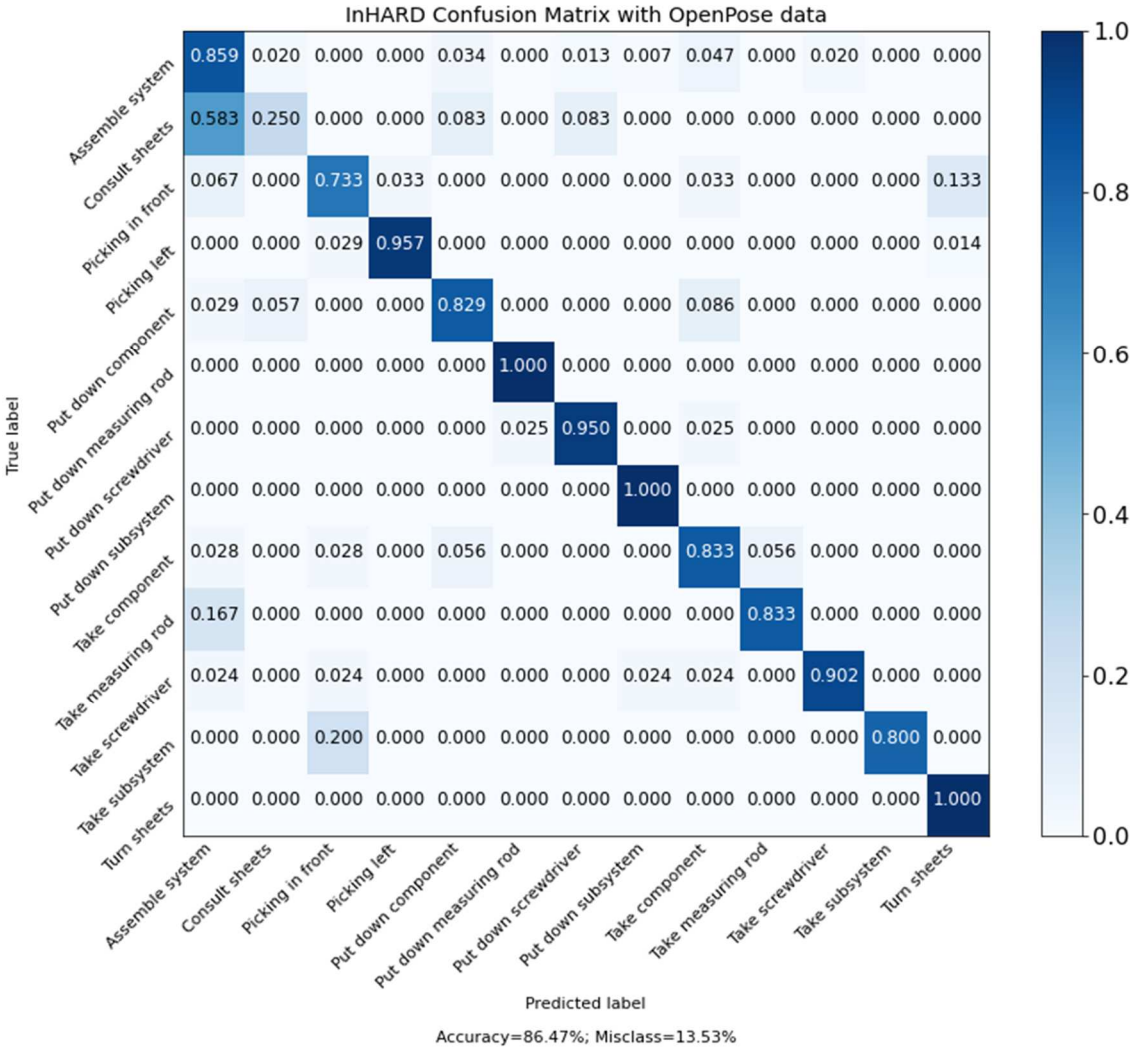


FIGURE 2.22 - Matrice de confusion du jeu de données InHARD avec la méthode ST-GCN en utilisant les données squelettiques 2D extraites par OpenPose

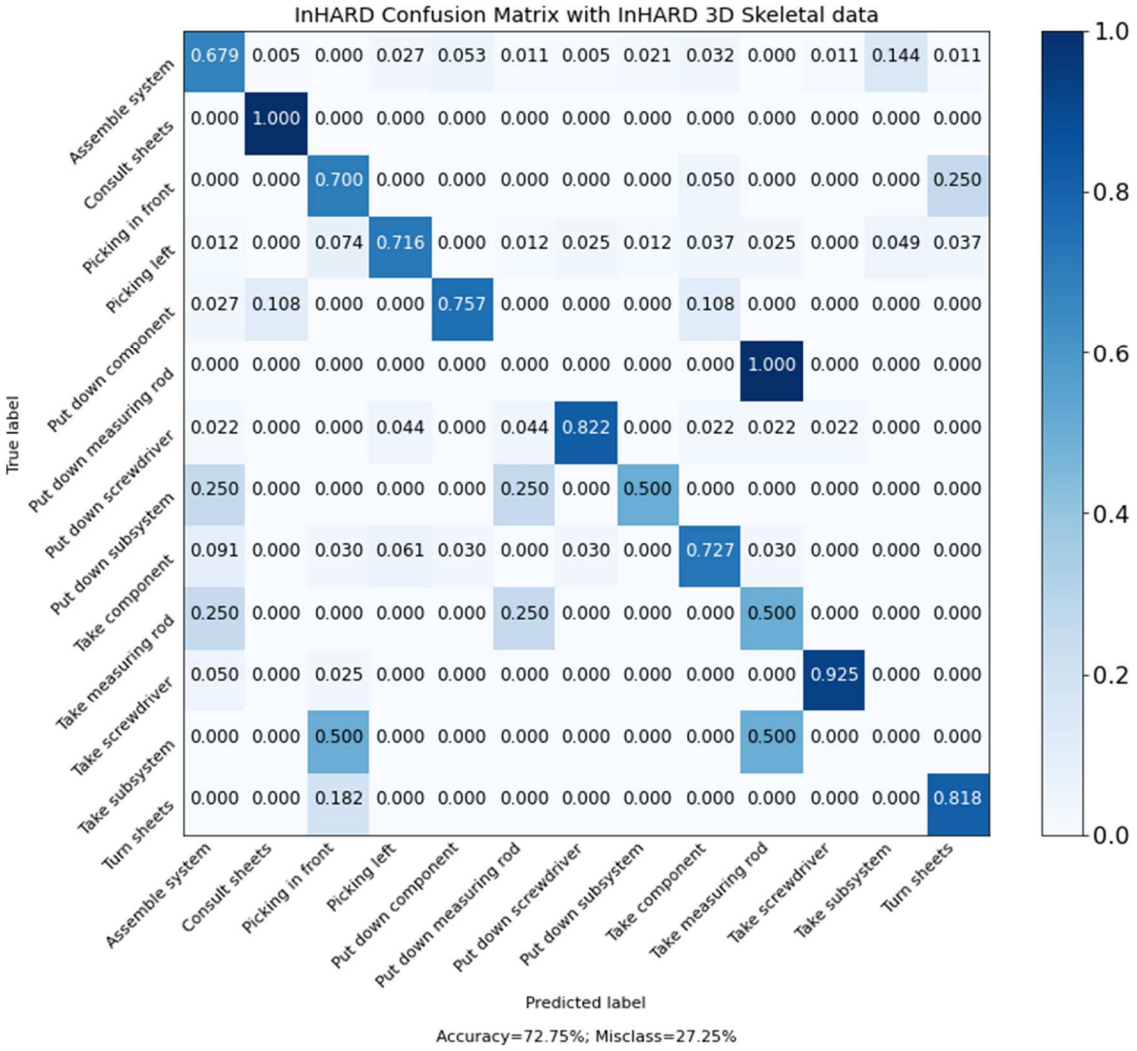


FIGURE 2.23 - Matrice de confusion du jeu de données InHARD avec la méthode ST-GCN en utilisant les données squelettiques 3D

La Figure 2.24 montre un exemple de reconnaissance d'actions en temps réel en utilisant une séquence vidéo extraite du jeu de données InHARD (en haut à gauche) comme entrée pour le framework OpenPose permettant d'extraire les données squelettiques 2D utilisées comme entrées pour le module ST-GCN (en haut à droite). L'algorithme ST-GCN propose un module d'attention (en bas à gauche et à droite) spatiale doté de grilles de sélection d'articulation qui est conçu pour attribuer de manière adaptative différentes attentions à différentes articulations du squelette d'entrée dans chaque image permettant au modèle de se concentrer davantage sur les articulations discriminantes de manière adaptative.

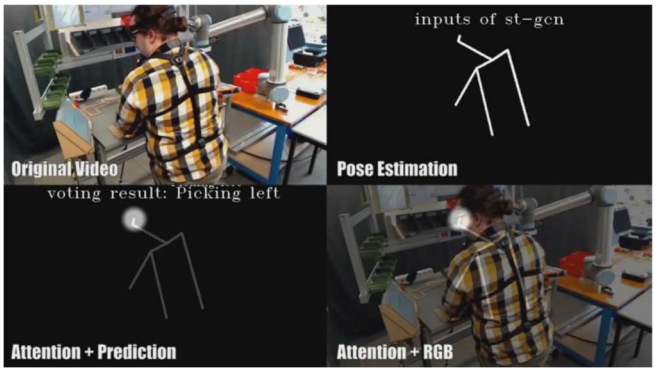


FIGURE 2.24 - Reconnaissance d'actions en temps réel du jeu de données InHARD avec la méthode ST-GCN en utilisant l'outil OpenPose

Dans l'espace 3D, il est possible d'unifier les orientations des squelettes en faisant pivoter leurs corps le long de l'axe vertical Y pour obtenir des directions d'orientation identiques (Poppe, van der Zee, et al. 2014). Dans les squelettes 2D extraits de vidéos, le point de vue de la caméra permettant d'acquérir les données d'apprentissage contraint l'usage ; cela empêche de modifier la position de la caméra en cas de nécessité (modification du poste, environnement évoluant). En outre, les orientations des corps peuvent difficilement être unifiées sans approximation, ou sans connaître la correspondance de l'environnement capturé avec le monde réel, ou les paramètres focaux de la caméra. De plus, dans les squelettes 2D, les proportions du corps changent de manière significative en fonction de l'orientation du corps et de l'angle de vue (Elias, Sedmidubský et Zezula 2021) comme décrit par la Figure 2.21.

Une diminution rapide de la précision en 2D peut être observée dans les scénarios où des caméras différentes sont utilisées pour l'apprentissage et le test, en cross-view (Elias, Sedmidubský et Zezula 2021) . Dans les squelettes 3D, cette erreur peut être atténuée par une normalisation appropriée qui unifie les directions d'orientation des échantillons d'apprentissage et de test. Cependant, cette normalisation ne peut pas être appliquée dans les squelettes 2D, où l'erreur de reconnaissance est en général plus élevée que dans les squelettes 3D. Contrairement aux changements de taille des squelettes, les changements de proportions du corps dus aux différentes orientations du corps ont un impact négatif sur la reconnaissance des squelettes 2D (Elias, Sedmidubský et Zezula 2021). Nous avons analysé la sensibilité des squelettes 2D aux variations de taille et d'orientation des squelettes. C'est pour cela, par la suite, nous utiliserons les données squelettes 3D dans le reste de nos expérimentations.

II.5.2.2 Influence du prétraitement des données squelettes 3D sur les performances du ST-GCN

Dans cette section, nous étudions l'influence des prétraitements sur le jeu de données InHARD. Les paramètres suivants sont étudiés dans cette section : le référentiel utilisé (c.à.d. données relatives ou absolue), données de rotation ou données de position des jointures et mise à zéro de

la jointure des Hips.

II.5.2.2.1 Influence du type de référentiel utilisé

Comme expliqué précédemment dans la Section II.2.3.1, le format BVH comprenant les données squelettes 3D, inclut les emplacements 3D (Tx, Ty et Tz) des 17 articulations du corps humain (voir Figure 2.25) durant toute l'acquisition, ainsi que les 3 rotations autour de chaque axe (Rx, Ry et Rz). Les données de positions et de rotations sont relatives à la jointure parent. Seule la jointure Hips utilise des données absolues par rapport au repère monde. Par conséquent, cette dernière peut poser des problèmes à cause des dérives pouvant survenir sur les centrales inertielles des capteurs utilisés vu que le squelette pourrait s'éloigner petit à petit du point de départ bien que l'opérateur ne se soit pas physiquement déplacé. Par ailleurs, les données de rotations sont fournies relativement à la jointure parente, ce qui assure le même point de repère pour toutes les séquences d'actions. Vu cette dépendance du point de départ, les positions et les rotations d'une même action pourraient ainsi être différentes d'une séquence à une autre.

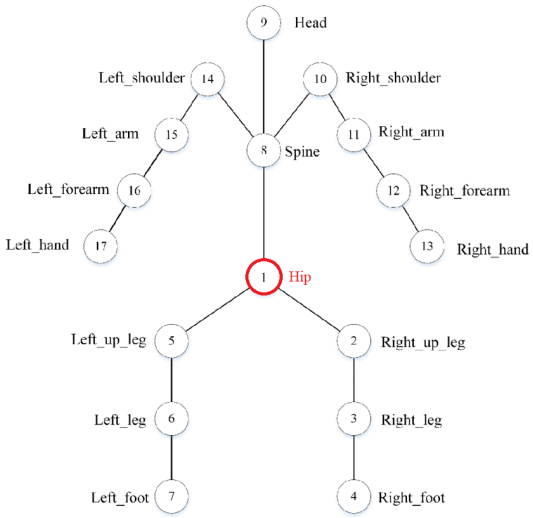


FIGURE 2.25 - Représentation des 17 articulations du squelettes dans un fichier BVH

II.5.2.2.2 Influence de la taille de la fenêtre de données et de l'échantillonnage

La taille de la fenêtre de données d'entrée ainsi que le FPS utilisé peuvent avoir un impact important sur la précision de la reconnaissance. Le choix de la taille de la fenêtre ainsi que le FPS sont effectués afin de garantir une caractérisation satisfaisante des activités transitoires mais aussi de s'assurer que toutes les actions, y compris les actions éphémères, sont bien représentées.

Pour évaluer l'impact des deux types de données (positions et rotations) sur les performances de HAR, et pour valider le choix du FPS, nous avons d'abord entrainé le modèle ST-GCN avec les données de positions en changeant à chaque fois la taille de la fenêtre (allant de 0.5 à 4.5 secondes) et le FPS (30, 60 et 120).

La Figure 2.26 montre une comparaison de l'impact de la taille de la fenêtre et du FPS sur les performances de HAR en utilisant les données de positions.

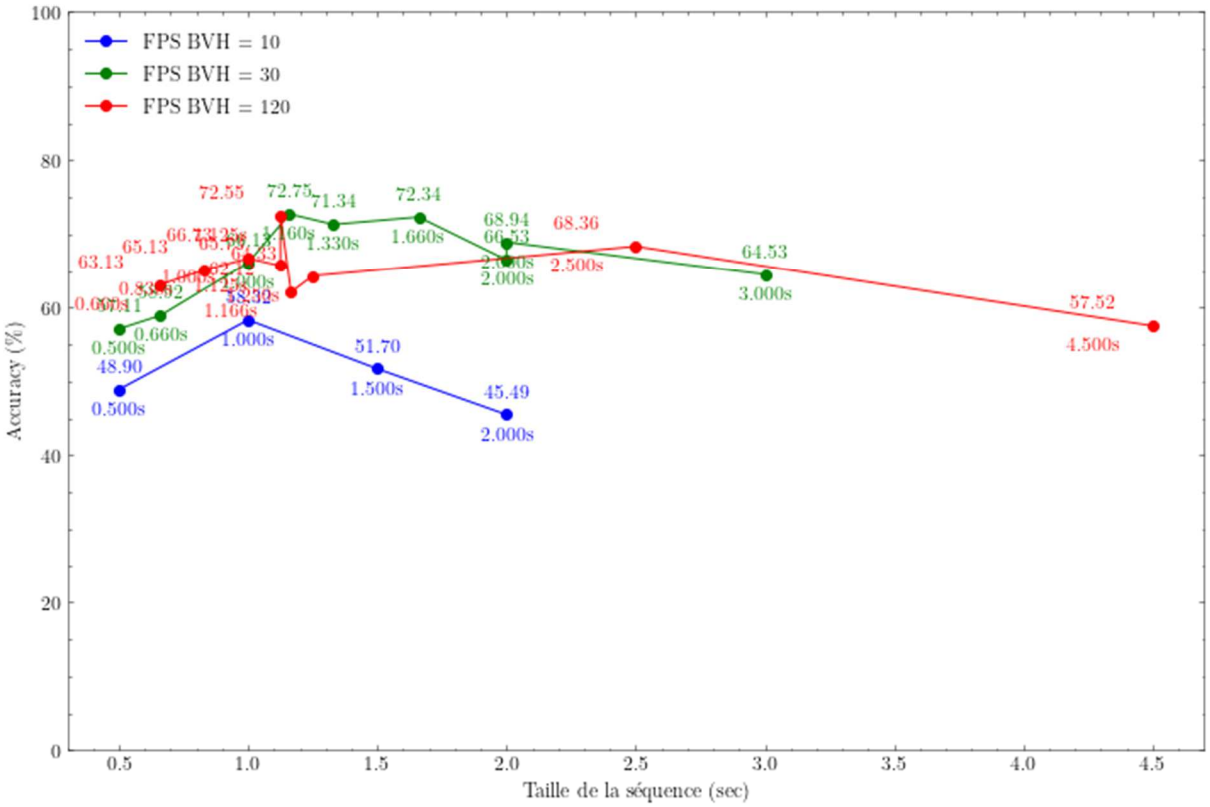


FIGURE 2.26 - Comparaison de l'impact de la taille de la fenêtre et du FPS sur les performances de HAR en utilisant les données de positions

Nous remarquons, qu'en choisissant un FPS égale à 30, et avec une taille de fenêtre égale à 1.33 secondes, nous avons obtenu les meilleures performances de HAR avec une précision moyenne de **72.75%.** Cela peut être expliqué par le fait qu'un FPS trop élevé (120 par exemple) comporte trop d'informations non nécessaires et redondantes dans le processus de reconnaissance d'actions. De plus, le temps d'apprentissage avec un FPS égale à 30 est moins important qu'avec un FPS égale à 120 comme le présente le Tableau 2.6. D'autre part, avec un FPS très faible (10 par exemple), nous perdons les informations caractérisant d'une manière significative les actions réalisées, plus particulièrement dans le cas où ces dernières sont de très courtes durées, comme dans le jeu de données InHARD, où la durée moyenne d'une action est d'environ 2 secondes.

TABLEAU 2.6 - Comparaison du temps d'apprentissage en utilisant des FPS différents

FPS	Temps d'apprentissage	Nombre d'époques
30	3h et 34min	1000
120	6h et 10min	1000

II.5.2.3 Choix du type de données

Une fois que nous avons validé le choix du FPS, dans ce qui suit nous évaluons le choix entre

données de positions absolues de chaque jointure et données de rotations relatives de chaque jointure par rapport à sa jointure parente. Pour se faire, nous avons lancé des tests avec un FPS fixé à 30 en variant la taille de la fenêtre et en apprenant cette fois-ci avec les données de rotations. Les résultats de ces tests sont présentés par la Figure 2.27.

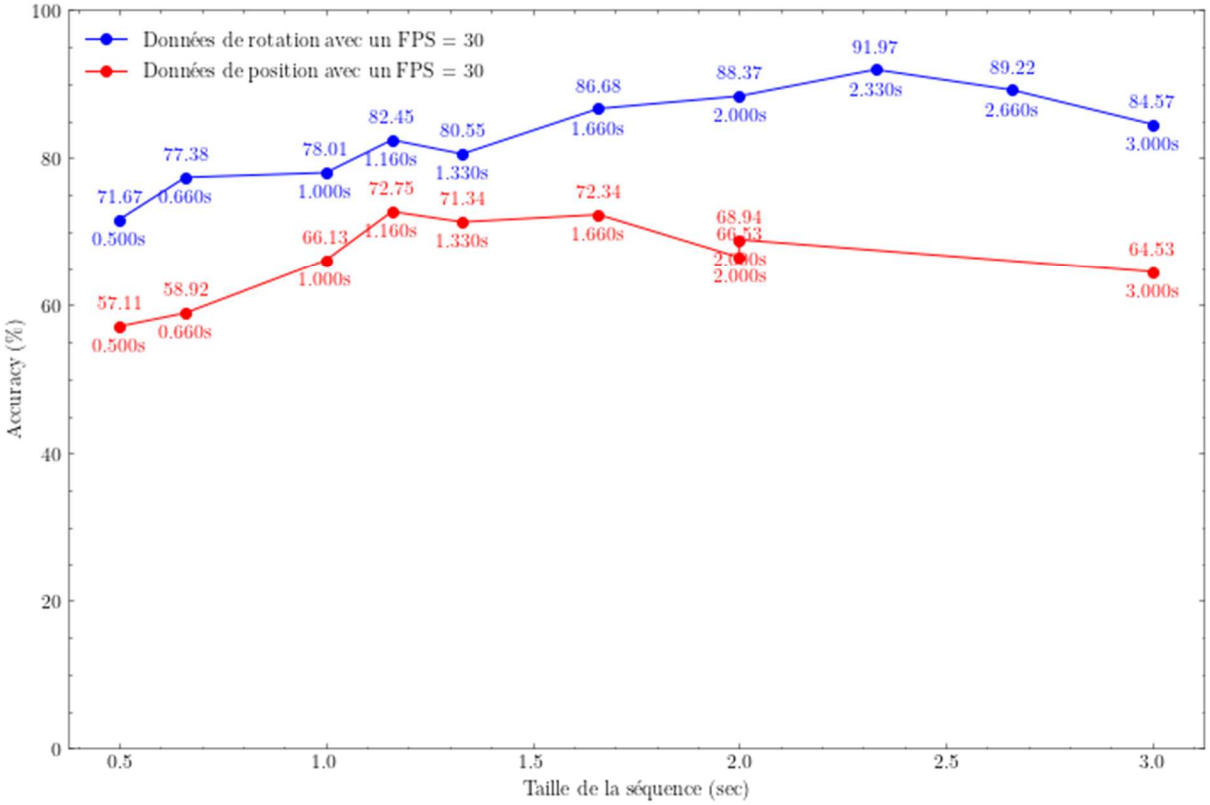


FIGURE 2.27 - Comparaison de l'impact de la taille de la fenêtre avec un FPS fixé à 30 sur les performances de HAR en utilisant les données de rotations (en bleu) et de position (en rouge)

Nous observons que les données de rotation relative obtiennent de meilleures performances que les données de position absolue. Ce résultat est expliqué par le fait que ces données sont fournies relativement à la jointure parente, contrairement à un référentiel absolu pour les données de positions des jointures. Une échelle absolue pose des problèmes à cause des dérives qui surviennent sur les centrales inertielles et le squelette pourrait ainsi s'éloigner du point de départ même si l'opérateur reste dans la même position. Par ailleurs, l'orientation initiale de l'opérateur peut créer de la variabilité dans les données de position absolue. Avec les données de rotations relatives, nous nous assurons que nous aurons le même point de repère pour toutes les séquences d'actions et pour tous les opérateurs. Ainsi, les mêmes actions seront approximativement représentées par les mêmes valeurs de rotations des jointures par rapport à la jointure parente.

Nous remarquons bien qu'avec les données de rotations, et en choisissant une taille de séquence fixée à **2.33** secondes, qui est inférieure ou égale à plus de 50% de toutes les actions effectuées (voir Figure 2.28), et avec un FPS égale à **30**, nous avons obtenu les meilleures performances de HAR en atteignant une Accuracy égale à **91.97%** et un F1-score égale à **92.11%**. La Figure 2.28

montre la distribution du nombre d'actions dans le jeu de données InHARD en fonction de leur durée. Nous remarquons aussi que la majorité des actions sont aux alentours de 2.33 secondes ce qui explique les bonnes performances obtenues avec ces paramètres.

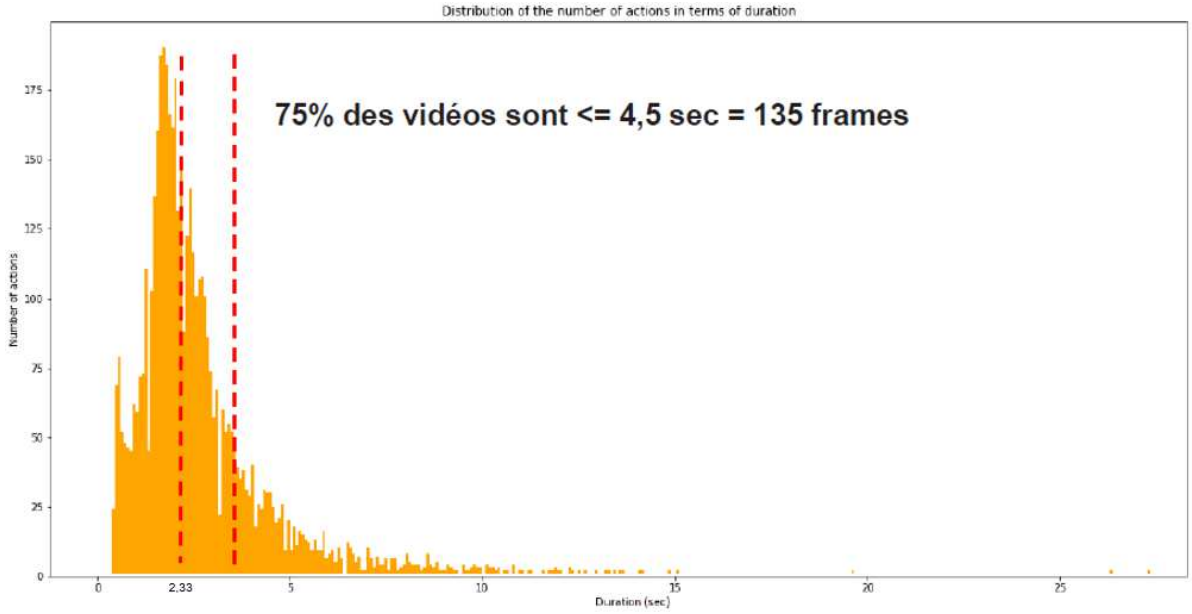


FIGURE 2.28 - Distribution du nombre d'actions dans le jeu de données InHARD en fonction de leur durée

Dans le cas où une séquence possède une taille inférieure à la taille de la fenêtre fixée à 70 frames (arrondi de $2.33\,secondes \times 30$), la technique du zéro-padding est appliquée. Cette dernière permet de mettre à zéro les bords de la séquence de données d'entrée pour avoir la même taille spatiale pour toutes les séquences. Dans le cas où la taille d'une séquence est plus grande que la taille de la fenêtre fixée à 70 frames, elle sera coupée si non le zéro-padding se fait à la fin de la séquence en question

La Figure 2.29 montre la matrice de confusion du jeu de données InHARD avec la méthode ST-GCN en utilisant un FPS égale à 30 et une taille de séquence égale à 70 frames (2.33 secondes).

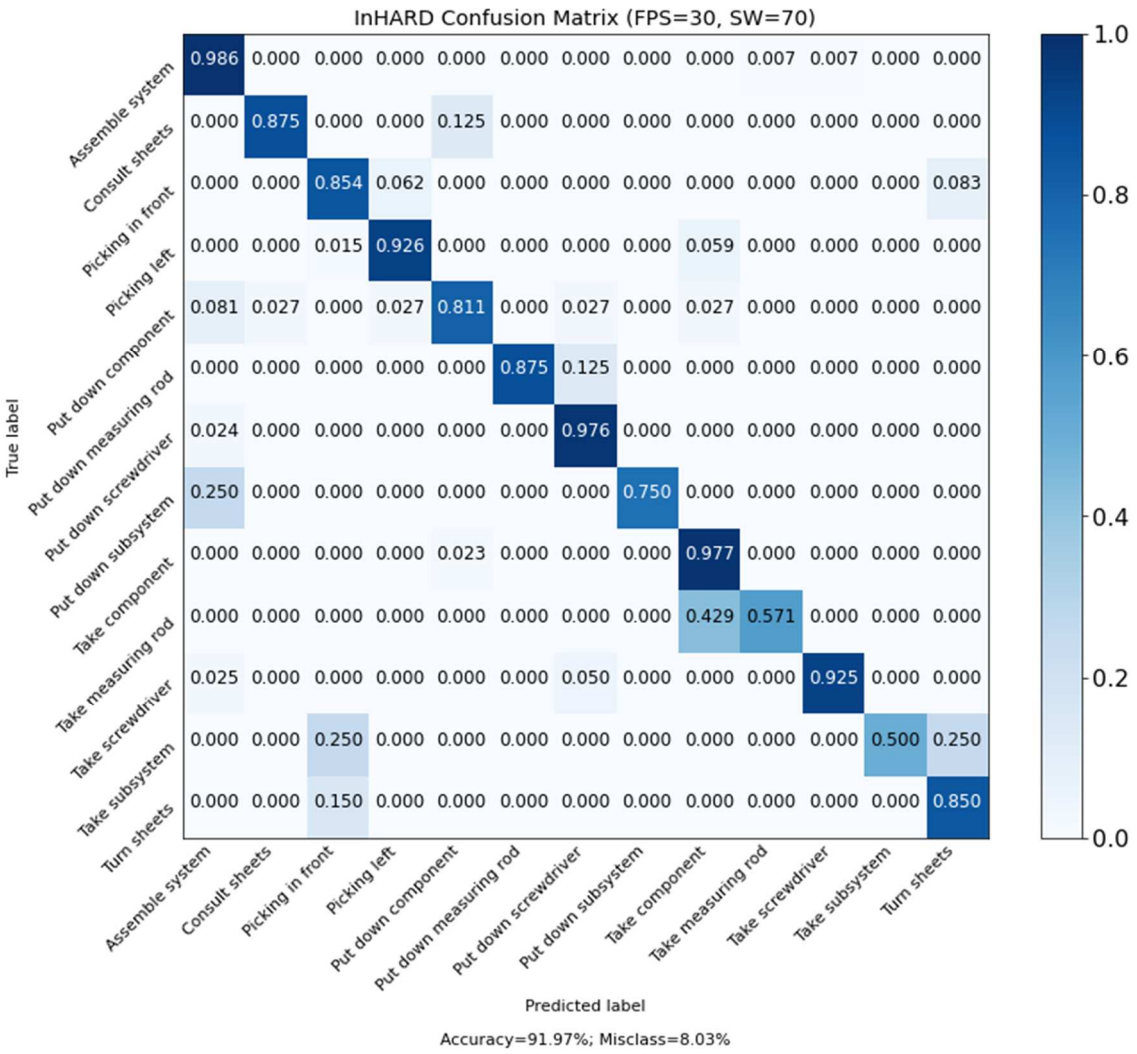


FIGURE 2.29 - Matrice de confusion du jeu de données InHARD avec la méthode ST-GCN en utilisant un FPS égale à 30 et une taille de séquence égale à 70 frames (2.33 secondes)

Nous remarquons que l'algorithme ST-GCN atteint une accuracy moyenne de **91.91%** sur la plupart des actions excepté les actions « Prendre toise » et « Prendre sous-système ». Cette baisse de performance peut s'expliquer par le fait que ces deux actions sont très peu représentées dans le jeu de données InHARD vu que la tâche d'assemblage complète les implique qu'une-à-deux fois par séquence mais aussi car ces actions sont proches l'une de l'autre comme il s'agit dans les 2 cas de prises d'un élément sur le plan de travail même si l'un est normalement plus au centre que l'autre.

Le Tableau 2.7 montre les différents paramètres d'apprentissage utilisés avec le jeu de données InHARD.

TABLEAU 2.7 - Paramètres d'apprentissage utilisés avec le jeu de données InHARD

Nb. époques	Learning Rate	Taille du batch	Fonction de loss
5000	0,1 --> 0,00001[7]	64	Weighted Cross Entropy / Focal Loss[8]

Une fois que nous avons validé le choix de type de données à utiliser (postions ou rotations), la taille de la fenêtre ainsi que le FPS, dans ce qui suit, nous comparons les résultats avec les différents prétraitements proposés pour évaluer leur impact sur les performances de HAR.

II.5.2.4 Influence des prétraitements sur les performances du ST-GCN

Le Tableau 2.8 montre l'Accuracy et les F1-scores moyens en utilisant les différents prétraitements proposés précédemment sur le jeu de données InHARD en utilisant les données de rotations, un FPS égale à 30 et une taille de séquence fixée à 70 frames (2.33 secondes).

TABLEAU 2.8 - Accuracy et F1-Score moyennes sur le jeu de données InHARD avec les différents prétraitements

Prétraitement	Accuracy	F1-Score
Pas de prétraitements	0.919	0.921
Mise à zéro des Hips	0.890	0.893
Ré-échantillonnage	**0.949**	**0.949**
Mise à zéro des Hips + Ré-échantillonnage	0.926	0.927

Nous remarquons qu'en utilisant les données de rotations relatives qui représentent mieux l'action que les données de positions absolues ainsi qu'avec les prétraitements de mise à zéro des Hips et de ré-échantillonnage permettant de prendre la totalité de l'action, nous avons obtenu une Accuracy de 92.60% sur le jeu de données InHARD et un F1-Score égale 92.76%. Comme nous l'avons expliqué précédemment, le prétraitement de mise à zéro des Hips permet de corriger la posture des opérateurs afin qu'ils aient tous la même position durant toute la manipulation d'assemblage. De plus, le prétraitement de ré-échantillonnage permet d'avoir la totalité de la séquence pour représenter une action donnée et non pas que le début de l'action si la taille de la fenêtre correspondante est plus petite, et éviter le découpage de la séquence si elle est plus grande que la taille de la fenêtre fixée.

Les Figure 2.30 et Figure 2.31 expriment les matrices de confusion du jeu de données InHARD

[7] Le Learning Rate (LR) est ajusté en utilisant la fonction Learning Rate Scheduling permettant de multiplier le LR par un coefficient s'il n'y a pas eu d'amélioration de la Loss ou de l'Accuracy pendant k époques.

[8] Des fonctions de pondération de classe permettant au modèle d'accorder plus d'attention aux exemples de la classe minoritaire qu'à la classe majoritaire dans les ensembles de données avec une distribution de classe fortement asymétrique.

avec la méthode ST-GCN en utilisant respectivement les prétraitements « Mise à zéro des Hips + Ré-échantillonnage » et « Ré-échantillonnage ».

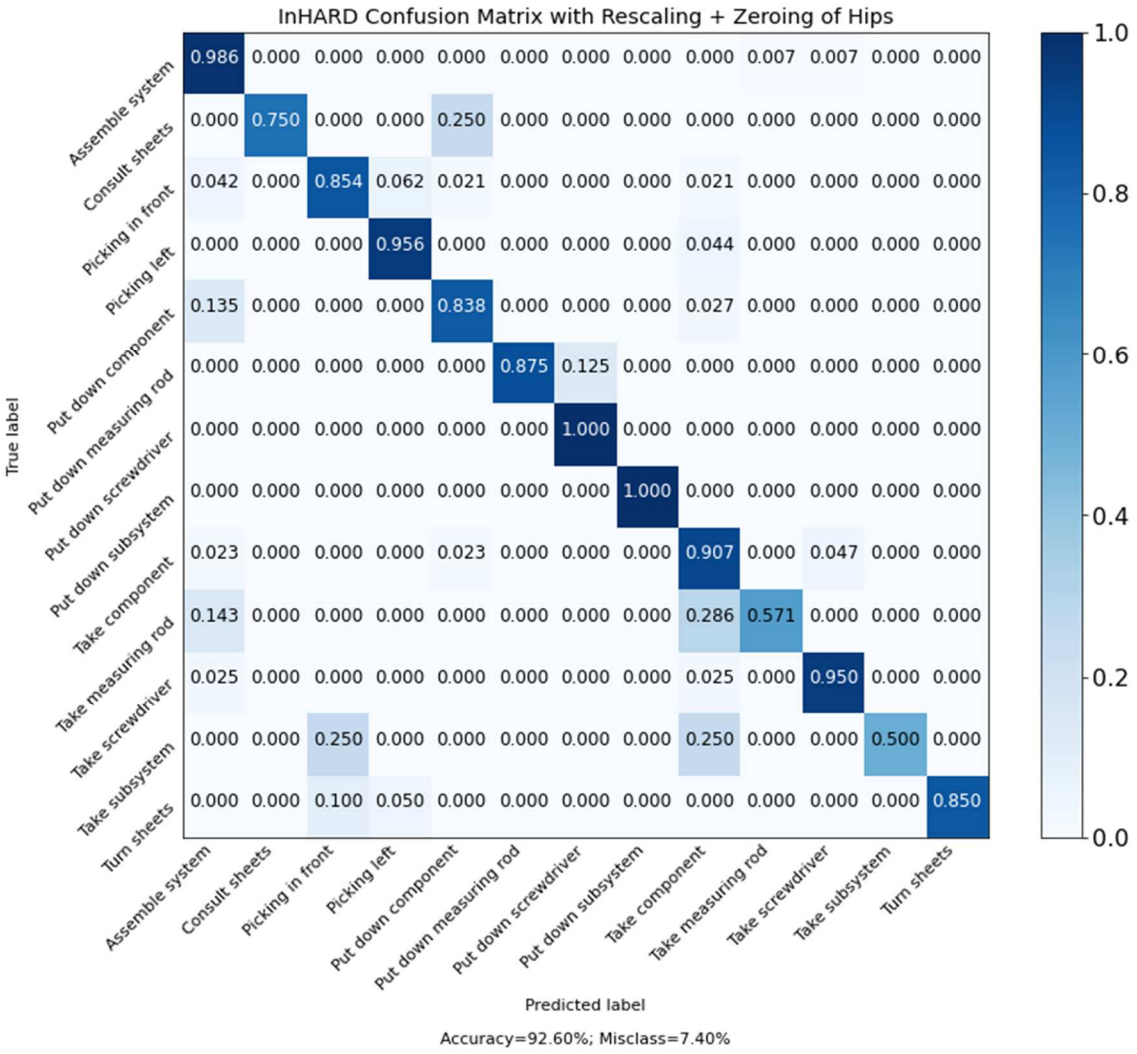


FIGURE 2.30 - Matrice de confusion du jeu de données InHARD avec la méthode ST-GCN en utilisant les prétraitements de Mise à zéro des Hips avec ré-échantillonnage

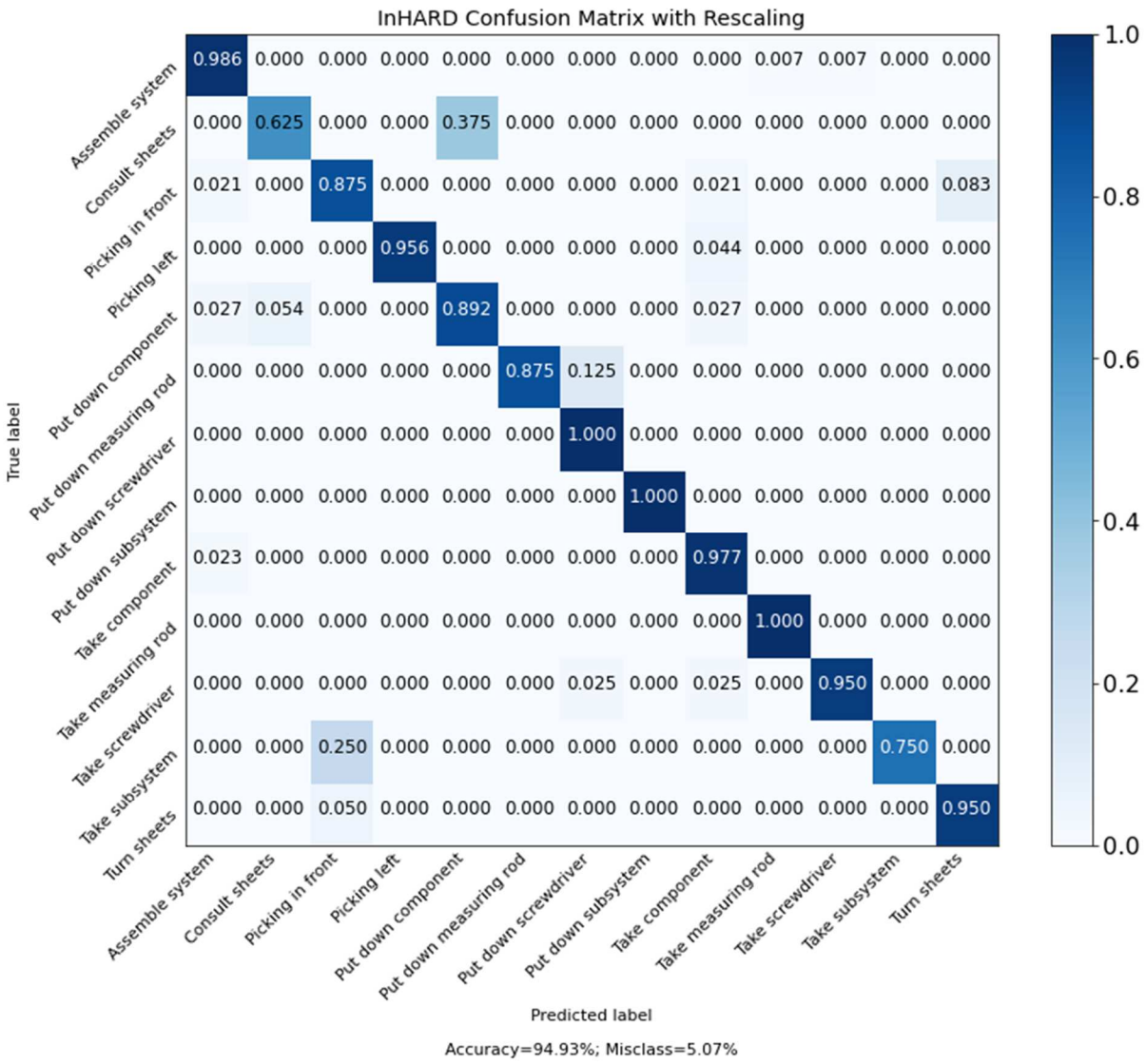


FIGURE 2.31 - Matrice de confusion du jeu de données InHARD avec la méthode ST-GCN en utilisant le prétraitement de ré-échantillonnage

Nous remarquons qu'avec le prétraitement de ré-échantillonnage seulement, l'algorithme ST-GCN confond moins les actions « Prendre sous-système » et « Prendre toise » avec une accuracy moyenne de 94.93% par rapport aux prétraitements « Ré-échantillonnage + Mise à zéro des Hips » avec une accuracy moyenne de 92.60%. Cette baisse de performance est dû au fait que la mise à zéro des Hips (les hanches) fige les hanches au point (0,0) durant toute la manipulation : or, ces actions pour la plupart du temps impliquent des mouvements des hanches pour les effectuer. C'est pourquoi, figer ces hanches au point (0,0) a entrainé une légère dégradation par la perte d'information dû à ce prétraitement.

Vu que nous avons essayé d'acquérir un jeu de données le plus réaliste possible, les opérateurs ont été invités à réaliser la manipulation de l'assemblage complet avec le robot UR10 de manière naturelle et non contrôlé. Dans cet assemblage, il y a des actions qui sont beaucoup plus réalisées que d'autres. Des actions comme « Prendre toise » et « Prendre sous-système » sont moins

représentées que d'autres ce qui explique parfois la baisse de performances pour ces actions puisqu'il en existe très peu d'instances. Une mauvaise prédiction pour une ou deux instances seulement de ces actions entraîne, dans ce cas, une baisse de performance Par ailleurs, des actions comme « Prendre toise » ou « Prendre composant » se basent sur des mouvements de jointures similaires et peuvent être facilement confondues lors de la tâche de reconnaissance. Toutefois, la méthode ST-GCN a montré sa robustesse et arrive à atteindre une performance de HAR avec une Accuracy de 92.60% et un F1-score de 92.76%.

II.6 Conclusion

Dans ce chapitre, nous avons introduit un jeu de données de reconnaissance d'actions humaines industrielles nommé « Industrial Human Action Recognition Dataset (InHARD) » qui a été publié dans la conférence IEEE ICHMS en 2020 (Dallel, Havard et Baudry, et al. 2020). Notre jeu de données inclut deux modalités de données (RGB + Squelette) et comprend 4804 échantillons d'actions différents, parmi 14 classes d'actions et répartis sur 38 vidéos. Cette première contribution permet de combler un manque dans les jeux de données de reconnaissance d'action existants, qui comprennent principalement des actions de la vie quotidienne ou liées à la santé. En effet, InHARD propose des actions d'assemblage sur poste manuel assisté par un bras cobotique dans un contexte industriel avec pour objectif d'aider la communauté scientifique travaillant sur des problématiques de la collaboration humain-robot dans les milieux industriels. À notre connaissance, il s'agit du premier jeu de données de reconnaissance d'action industrielle à être proposé. Nous avons également proposé un ensemble de conditions d'utilisation de notre jeu de données pour une utilisation future afin de permettre une évaluation équitable entre les différentes approches.

En nous basant sur l'état de l'art, des expérimentations avec l'algorithme ST-GCN ont été menées sur InHARD et ont permis d'obtenir des résultats avec une Accuracy de 92.60% et un F1-score de 92.76% sur les données segmentées en utilisant les informations de rotations de chaque jointure relatives à la jointure parente. En outre, une étude sur les prétraitements des données utilisées a été présentée permettant d'améliorer les performances de la HAR. Nous avons montré qu'avec le prétraitement de ré-échantillonnage, nous avons obtenu les meilleures performances avec une Accuracy de 94.9% et un F1-Score égale à 94.9%.

L'un des défis que nous avons rencontrés lors de la création de notre jeu de données InHARD était l'étape de la labélisation. Cette étape consistait à assigner manuellement un label ou une étiquette prédéfinie pour chaque action effectuée par l'opérateur. Cette tâche est très délicate en raison de la précision et des erreurs possibles de labélisation puisqu'elle est effectuée manuellement, et est également fastidieuse et longue, surtout lorsqu'il s'agit d'un jeu de données à grande échelle comme le nôtre. Disposer d'une importante quantité de données étiquetées rend les modèles d'apprentissage profond plus robustes face à la grande variabilité des formes, des

postures et des vêtements humains ou à la complexité de leurs mouvements, ce qui améliore la performance des modèles et la précision de la reconnaissance. Cette variabilité est intéressante principalement dans le cas d'architecture basée sur des caméras RGB. Néanmoins, la taille de la personne permet d'être robuste à la variabilité des humains réalisant les actions. Par conséquent, la nécessité de synthétiser et de générer des données pour entrainer des modèles a conduit les chercheurs à utiliser différentes techniques telles que des outils de traitement d'images ou des représentations virtuelles de systèmes physiques via des jumeaux numériques pour résoudre les problèmes de vision par ordinateur, en particulier le problème de HAR. Par conséquent, le chapitre suivant étudie la possibilité d'utiliser les jumeaux numériques de système industriel pour générer des données labélisées d'une manière automatique, données utilisées pour la HAR.

Chapitre 3

Proposition d'une méthodologie de génération de données auto-étiquetées basée sur le jumeau numérique et la RV pour la HAR dans le contexte HRC

Sommaire

III.1 Introduction

Comme présenté dans les chapitres précédents, la reconnaissance d'actions humaines basée sur des méthodes d'intelligence artificielle pour permettre la collaboration humain-robot (HRC) dans des environnements de travail reste un défi, notamment en raison des énormes jeux de données d'apprentissage nécessaires. Dans ce contexte, l'apprentissage de nouvelles activités nécessite de pouvoir simuler ces interactions et les EVAH (Environnement Virtuel pour l'Apprentissage Humain) représentent une solution prometteuse.

Les Environnements Virtuels (EV) pour l'Apprentissage Humain (EVAH) sont devenus des objets d'étude incontournables pour les recherches menées sur l'apprentissage humain, les Interfaces Homme-Machine (IHM) et les interactions multimodales (Pellas, et al. 2017) (Natsis, et al. 2012). La plupart des travaux actuels menés sur les EVAHs, ne permettent pas à l'utilisateur de naviguer et d'interagir librement dans l'espace en trois dimensions en temps réel

limitant ainsi son expérience.

Au sein d'un EV immersif et à l'aide d'un scénario d'apprentissage prédéfini, un apprenant doit généralement effectuer une succession d'actions, consistant en la modification des propriétés d'un ou plusieurs artefacts virtuels (e.g. assembler une pièce, attraper un objet etc.). Du point de vue de l'évaluation de l'activité humaine, de tels EVAH ne sont pas si différents de n'importe quel système informatique dédié à l'apprentissage. En effet, ces systèmes informatiques analysent les activités en collectant des traces et en générant des indicateurs à partir des états discrets du système et de ses artefacts d'une part, et d'une succession d'actions effectuées avec la souris et le clavier d'autre part (Markowska-Kaczmar, Kwasnicka et Paradowski 2010). Cependant, les EVs offrent des possibilités d'interaction bien plus avancées telles que celles basées sur le mouvement et le geste humain pour la sélection et la manipulation d'objet 3D, la navigation et le contrôle de l'application (Penichet, Peñalver et Gallud 2013) (Emma-Ogbangwo, et al. 2014).

Grâce à l'arrivée des jumeaux numériques (Digital Twin - DT), les mondes physique et numérique peuvent désormais être gérés comme un seul univers et il est désormais possible d'interagir avec l'homologue numérique des objets physiques. Les DT aident déjà les entreprises à concevoir, visualiser, surveiller, gérer et entretenir leurs savoirs plus efficacement. Les DT des systèmes de production centrés sur l'homme sont de plus en plus développés et utilisés dans les phases de conception et d'exploitation.

Le DT intègre l'Intelligence Artificielle (IA), l'Apprentissage Automatique, la Réalité Virtuelle/Augmentée et le Big Data afin de créer des représentations ou des simulations virtuelles de biens physiques. Le DT peut être utilisé dans diverses industries : Fabrication, automobile, construction, services publics et santé, etc. Les technologies de l'information et de la communication sont la prochaine étape importante de l'industrie 5.0, qui conduira au développement de nouveaux produits et processus (Chryssolouris, et al. 2009) (Kritzinger, et al. 2018). Le DT peut être utilisé comme un moyen d'interagir avec le monde physique et de permettre la collecte de données en temps réel par le biais de capteurs qui peuvent être utilisés pour des tâches de maintenance prédictive ou d'optimisation. En effet, le modèle virtuel étant mis à jour de manière continue et transparente tout au long du cycle de vie du produit, cela permet de savoir comment un produit se comporterait avant même d'être produit (Negri, Fumagalli et Macchi 2017). Ainsi, le jumeau numérique d'un produit ou d'un système reproduit le comportement de sa version physique de manière réaliste. C'est pourquoi, il devient utile pour produire de la donnée afin d'alimenter des algorithmes d'apprentissage profond.

Cependant, peu de travaux traitent de l'utilisation du DT comme outil de génération de jeux de données. Par conséquent, dans ce chapitre nous explorons l'utilisation du DT d'un poste de travail industriel, présenté dans le chapitre précédent, impliquant des tâches d'assemblage avec un bras robotique, interfacé avec la réalité virtuelle (VR) pour extraire un modèle humain

numérique. Le DT simule des opérations d'assemblage effectuées par des humains dans le but de générer des données auto-étiquetées et d'alimenter un algorithme d'apprentissage profond.

Dans ce chapitre nous présentons donc une méthodologie qui utilise le DT pour générer automatiquement des données étiquetées utilisées pour le processus de HAR basé sur un algorithme d'apprentissage profond. Le DT développé simule un flux de travail d'assemblage dans un poste de travail industriel. Ainsi, un jeu de données de reconnaissance d'actions humaines appelé InHARD-DT a été créé et sera validé sur un cas d'utilisation réel. Tout d'abord, nous avons utilisé les données DT auto-étiquetées, acquises grâce à la représentation virtuelle du poste de travail utilisé pour l'acquisition du jeu de données InHARD, présenté dans le chapitre précédent. Ces données ont permis d'entraîner le réseau de neurones convolutifs à graphe spatio-temporel (ST-GCN) avec les données squelettes. Afin de valider l'approche, les données du Jumeau Physique (PT) du jeu de données InHARD ont été utilisées pour tester le transfert entre l'apprentissage sur données DT et le test sur les données PT.

L'approche proposée est basée sur trois parties de modèle. Le premier modèle concerne le comportement cinématique et dynamique d'un atelier industriel avec un bras robotique (UR10). Le deuxième modèle concerne l'environnement RV et la programmation des événements déclencheurs des actions effectuées au cours de la tâche d'assemblage, utilisés par la suite pour labéliser les données. Le dernier modèle concerne un modèle humain numérique qui consiste en des données squelettiques utilisées pour entraîner un réseau de neurones convolutif à graphe spatio-temporel (ST-GCN) pour la HAR.

La suite de ce chapitre est organisée comme-suit : tout d'abord nous passons en revue la génération des données synthétique et la labélisation des données en présentant quelques méthodes de la littérature. Par la suite, nous présentons notre méthodologie basée sur le DT pour générer un jeu de données d'actions industrielles nommé InHARD-DT labélisé automatiquement utilisé par la suite pour la HAR. Ensuite, nous présentons les résultats obtenus et nous évaluons la robustesse de la généralisation par une méthode de validation croisée basée sur les deux jeux de données créés : InHARD-DT et InHARD.

Les principales contributions de ce chapitre résident dans trois aspects :

- Une méthodologie basée sur un DT d'un atelier industriel simulant des tâches d'assemblage qui permet de produire des données labélisées automatiquement utilisées pour la HAR dans le contexte de la HRC.

- Créer et publier un jeu de données de HAR industriel nommé InHARD-DT désignant la représentation virtuelle du jeu de données InHARD à l'aide du DT.

- Valider notre approche en apprenant un réseau de neurones convolutif à graphe spatio-

temporel (ST-GCN) avec les données DT acquises du jeu de données InHARD-DT et en testant avec les données PT du jeu de données InHARD (Dallel, Havard et Baudry, et al. 2020).

III.2 Génération de données synthétiques et labélisation des données

La génération de données synthétiques est cruciale car elle peut être utilisée pour préserver la confidentialité, tester des systèmes opérationnels ou créer des données d'apprentissage pour des algorithmes d'apprentissage profond tout en garantissant une qualité de données équivalente, à celle des données réelles. Ainsi, les données synthétiques peuvent contribuer à améliorer les modèles d'apprentissage profond. Par ailleurs, elles peuvent également être modifiées afin de minimiser les contraintes liées à l'utilisation de données réglementées (i.e. par anonymisation) pour correspondre à des conditions que les données réelles ne permettent pas. Enfin, elles ont surtout l'avantage de permettre de générer de grands jeux de données d'apprentissage sans nécessiter une labélisation manuelle de celles-ci. Les jeux de données générés synthétiquement offrent en outre, un moyen de conserver l'utilité des données originales tout en préservant la vie privée de leurs propriétaires (Kamthe, Assefa et Deisenroth 2021).

Dans (Ekbatani., Pujol. et Segui. 2017), les auteurs ont proposé un framework pour la génération de données synthétiques permettant le comptage automatique du nombre de piétons situés sur une chaussée. Ils ont utilisé un algorithme pour créer des images synthétiques qui sont ensuite transmises à un réseau de neurones convolutifs profond (DCNN) pour l'apprentissage. Les auteurs montrent que le modèle proposé est capable de compter avec précision le nombre d'individus dans une scène réelle en apprenant sur des données synthétiques. Ils ont utilisé l'erreur quadratique moyenne (MSE) pour évaluer les performances de leur approche. Ils ont obtenu une MSE égale à 0.942 en utilisant les données synthétiques et une MSE égale à 3.61 en utilisant les données réelles.

Dans (Tripathi, et al. 2019), les auteurs ont proposé une approche axée sur les tâches pour générer des données synthétiques. Afin de générer les échantillons d'apprentissage, un réseau synthétiseur a été utilisé. Ce dernier ainsi que les réseaux cibles ont été formés de manière contradictoire, c'est-à-dire que chaque réseau est mis à jour et a l'intention de surpasser l'autre. En associant le générateur à un discriminateur entraîné sur des images du monde réel, des données réalistes sont donc générées.

Dans (Varol, Laptev, et al. 2021), les auteurs ont présenté une méthodologie pour l'augmentation des jeux de données de reconnaissance d'action avec des vidéos synthétiques. Les variations des données synthétiques, telles que les points de vue et les mouvements, ont été explorées. Ils ont utilisé une méthode d'estimation du mouvement humain pour capturer en 3D la dynamique humaine à partir d'une vidéo RGB d'une seule vue. Ils ont ensuite combiné la séquence obtenue avec des éléments générés aléatoirement (par exemple, point de vue, vêtements)

pour diversifier les données d'apprentissage. Ils ont utilisé un modèle CNN spatio-temporel pour la HAR et ont obtenu de bonnes performances à partir des points de vue invisibles. Cependant, dans les scènes encombrées, leur approche n'a pas donné de bons résultats car sa performance dépend de la performance de l'estimation de la pose 3D qui, dans de tels scénarios, ne permet pas d'obtenir des estimations bien précises.

Dans (de Melo, et al. 2020), les auteurs ont présenté une approche d'apprentissage profond basée sur des données RGB pour contrôler des robots par le biais de gestes. Ils ont utilisé des données synthétiques générées à partir d'un simulateur qu'ils ont développé en créant un jeu de données de vidéos de gestes de contrôle synthétiques et réelles pour entraîner un modèle I3D pour la reconnaissance des gestes dans un contexte HRC. Le jeu de données créé comprend des vidéos réelles avec des participants humains ainsi que des vidéos synthétiques provenant du simulateur qu'ils ont développé. Plusieurs expériences ont été décrites pour évaluer l'utilité des données synthétiques en étudiant des propriétés telles que les variations de gestes, la variété des caractères et la généralisation à de nouveaux gestes. La méthode montre que l'utilisation de données RGB synthétiques avec le réseau CNN améliore la précision de la reconnaissance des gestes. Toutefois, les performances sont bien meilleures lorsqu'elles sont mélangées avec des données réelles. Par conséquent, les données réelles sont toujours nécessaires. De plus, cette étude est basée sur des gestes plus simples en comparaison de ceux d'un contexte industriel.

Dans (Liu, et al. 2019), les auteurs ont proposé un système appelé « Human Pose Models » dont l'objectif est de reconnaitre des poses humaines à partir d'une caméra RGB-D en étant robustes aux variations des textures des vêtements, des arrière-plans, des conditions d'éclairage et des points de vue de la caméra. La contrainte pour ce type de système est d'obtenir un large dataset. Pour cela, les auteurs ont tout d'abord développé un framework pour synthétiser les données. Ils ont ainsi appris des poses humaines à partir des données du squelette humain et ont généré des représentations synthétiques en 3D avec différentes formes de corps tout en faisant varier aléatoirement les textures des vêtements, les arrière-plans et l'éclairage. Ensuite, ils ont utilisé des réseaux adverses génératifs (GAN) pour réduire l'écart entre les distributions d'images synthétiques et réelles. Ils ont ensuite formé des modèles CNN utilisés comme extracteurs de caractéristiques invariantes à partir de données RGB et de profondeur réelles de vidéos d'actions humaines ; les variations temporelles ont été modélisées à l'aide de la pyramide temporelle de Fourier. Certaines des variations des facteurs environnementaux appliqués aux données n'étaient pas très différenciables et évidentes, même à l'œil nu, ce qui pourrait limiter la capacité de généralisation du modèle de pose humaine appris à partir d'images synthétiques.

Tout au long de la création du jeu de données InHARD (Dallel, Havard et Baudry, et al. 2020), l'un des défis que nous avons rencontrés était l'étape de labélisation de données, au cours de laquelle nous passons en revue l'ensemble des données capturées et assignons manuellement chaque action effectuée par un opérateur à l'une des classes d'action prédéfinies. Cette tâche est

très délicate en raison de la précision et des erreurs de labélisation possibles puisqu'elle est effectuée manuellement. Elle est également fastidieuse, surtout lorsqu'il s'agit d'un jeu de données à grande échelle.

Par conséquent, pour aider à minimiser la charge cognitive et le changement de contexte et aussi pour aider à contrecarrer l'erreur des annotateurs individuels, dans ce chapitre nous proposons une méthode qui permet de générer une énorme quantité de données auto-labélisées en utilisant un DT couplé à un environnement virtuel permettant de simuler des opérations industrielles telles que des opérations d'assemblage ou de maintenance. En effet, comme présenté dans le chapitre d'état de l'art, de nombreuses approches ont souligné l'importance du DT dans les phases de conception et d'exploitation pour la HRC. La digitalisation du système physique permet différents modes de simulation grâce à la synchronisation entre les systèmes virtuels et physiques, aux données collectées en temps réel et aux modèles mathématiques. Par ailleurs, la RV couplée au DT facilite la conception et la formation des opérateurs et permet d'acquérir des données comportementales réalistes des opérateurs utilisant le système industriel. Les jumeaux numériques sont de plus en plus déployés et largement utilisés dans l'industrie car ils peuvent être exploités dans de nombreux scénarios, tels que la formation, la conception de postes de travail, les études d'ergonomie, la sécurité et l'évaluation du comportement des robots, etc. (voir la Figure 3.). Puisque le DT est déjà existant et utilisé dans le processus industriel, nous proposons donc d'améliorer le DT existant en l'utilisant en plus comme outil de génération de données auto-labélisées pour entraîner un modèle d'apprentissage profond (voir la Figure 3.1). Ainsi, lorsque les opérateurs sont invités à effectuer un processus d'assemblage dans la RV à des fins de formation ou d'études ergonomiques, les données DT sont en outre collectées et utilisées pour entraîner une architecture HAR. Cette proposition sera détaillée dans la section suivante.

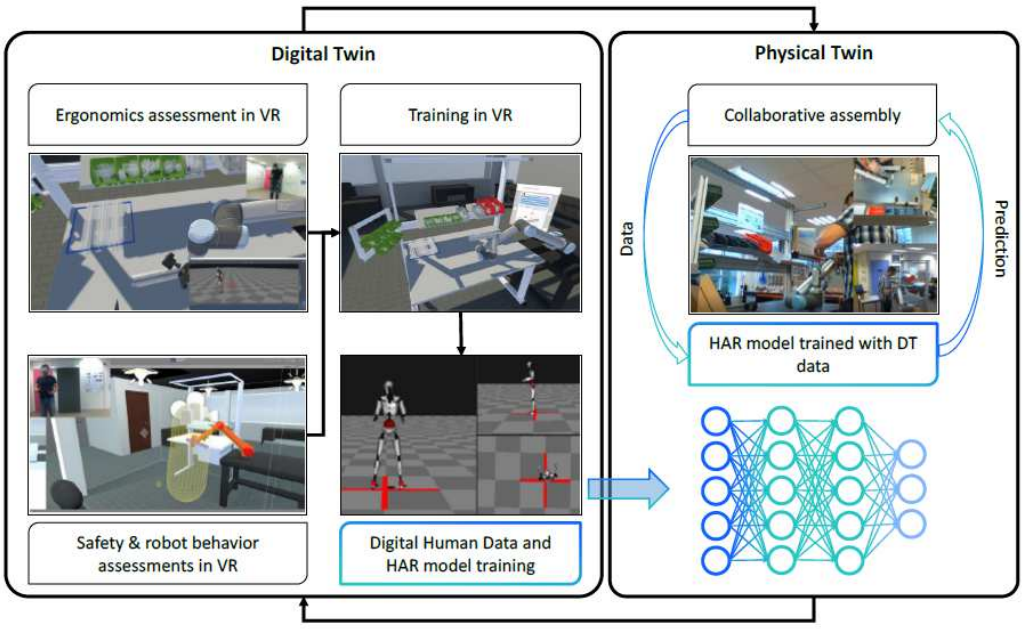


FIGURE 3.1 - Les différents usages du DT dans un contexte HRC.

Dans ce qui suit, nous exposons de manière détaillée notre méthode en présentant, tout d'abord, le jumeau numérique d'un poste de travail industriel utilisé dans le jeu de données InHARD (Dallel, Havard et Baudry, et al. 2020). Puis, nous proposons d'utiliser la réalité virtuelle afin de générer des données auto-labélisées qui seront ensuite utilisées pour entrainer l'architecture ST-GCN pour la HAR dans un contexte de HRC.

III.3 Matériels et méthodes

Cette section présente la méthodologie développée pour la génération de données auto-labélisées. Tout d'abord une description concise de l'application de réalité virtuelle utilisée, basée sur un DT d'un poste de travail d'assemblage d'un système industriel, est présentée. Ensuite, la solution proposée pour la génération de données auto-labélisées est présentée. Enfin, les résultats expérimentaux et les discussions sont détaillés.

III.3.1 Concept d'application : Le DT comme générateur de données

Comme nous l'avons présenté dans la section précédente, l'objectif de ce chapitre est d'adresser l'un des problèmes les plus rencontrés par la communauté de l'IA utilisant les méthodes d'apprentissage automatique, à savoir le manque de données labélisées. En effet, nous avons rencontré cette problématique lors de la création de notre jeu de données InHARD (Dallel, Havard et Baudry, et al. 2020), présenté dans le chapitre précédent ; les algorithmes d'apprentissage automatique ont besoin d'une grande quantité de données labélisées afin de générer un modèle robuste, capable de faire face à la grande variabilité des postures humaines ainsi qu'à la complexité de leurs mouvements et de prédire correctement les actions similaires non apprises lors de la phase d'entrainement (Martinez-Gonzalez, et al. 2019).

Pour surmonter ces problèmes, nous avons décidé de nous appuyer sur le jumeau numérique associé à une application RV comme générateur de données auto-labélisées. Le cas d'usage correspond à une station de travail impliquant l'assemblage de diverses pièces à l'aide du bras robotique UR10 (voir Figure 3.3).

III.3.2 Jumeau Physique et Jumeau Numérique (PT & DT)

Les travaux menés au sein du laboratoire par (Richard, et al. 2021) ont permis de produire un environnement de travail en réalité virtuelle associé au jumeau numérique permettant de simuler un poste physique d'assemblage avec un bras robotique (UR10) (voir Figure 3.3). Ce travail permet notamment de concevoir, former et simuler l'assemblage dans un jumeau numérique du poste industriel (voir Figure 3.3). Nous avons donc adapté ce travail en définissant le poste de travail ainsi que les différentes parties, objets, outils, sous-systèmes et systèmes utilisés pendant toute la manipulation pour recréer entièrement le scénario du poste de travail réel, en réalité virtuelle ; chaque partie du système d'assemblage a été soigneusement définie à l'aide du

framework INTERVALES (Richard, et al. 2021) pour s'assurer qu'elle se comporterait et serait associée à d'autres pièces et composants de la même manière que dans le scénario du monde réel. Par ailleurs, les instructions à suivre sur ce poste de travail sont exactement les mêmes que celles du jeu de données InHARD. L'objectif consiste en un assemblage de composants et systèmes, réalisé sur différentes étapes et assisté par un bras robotique (UR10). Ce dernier est utilisé pour assister les opérateurs dans les tâches d'assemblage en lui présentant et en maintenant des pièces spécifiques. Afin d'avoir un comportement du bras robot similaire au bras réel, le DT de l'UR10 a été co-simulé avec RoboDK et intégré dans la scène VR.

Le flux de travail du jeu de données InHARD-DT consiste en l'assemblage et au vissage de différentes pièces, réparties sur 9 étapes différentes au sein d'un poste de travail collaboratif (voir Figure 3.2-a & b). Ce poste fait partie d'un atelier de production flexible et correspond à une partie des opérations d'assemblage réalisées afin de produire des vélos pour enfants (voir la Figure 3.2-d). Le jeu de données InHARD-DT est mis à la disposition du public[9].

FIGURE 3.2 - Plate-forme industrielle et atelier de production flexible utilisés pour la construction des jeux de données. a) InHARD-DT et b) InHARD. c) Système final construit sur le premier poste de travail et utilisé dans le jeu de données InHARD-DT d) Vue d'ensemble du système final de l'ensemble du processus d'assemblage réalisé sur tous les postes de travail.

[9] https://github.com/mejdidallel/InHARD-DT

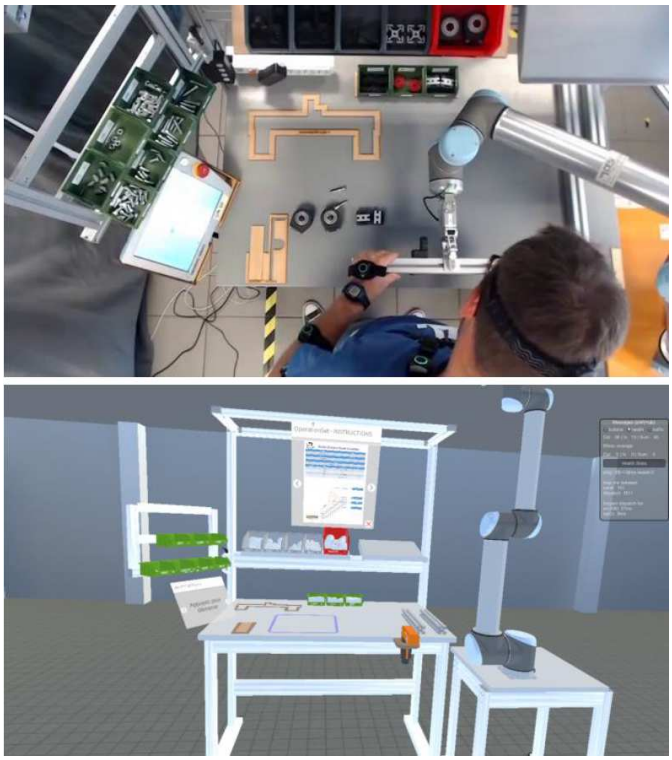
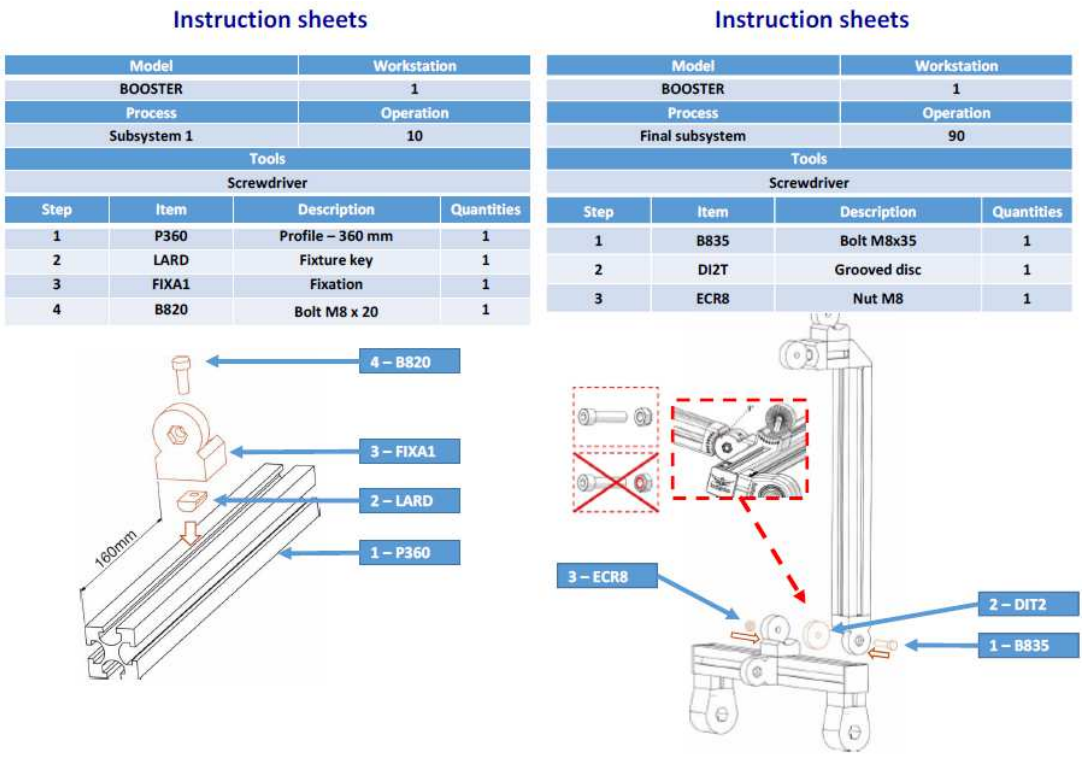


FIGURE 3.3 - Poste de travail réel (en haut) et son jumeau numérique rendu en réalité virtuelle (en bas).

À l'aide d'un casque et contrôleurs VR HTC-Vive Pro Eye présentés dans la Figure 3.4, les participants peuvent interagir avec la station de travail et les différentes pièces et systèmes ; ils leur permet aussi de se déplacer librement dans l'espace 3D de la station de travail comme s'ils étaient dans un espace réel et d'interagir avec la station de travail et tous les objets impliqués dans la tâche d'assemblage.

FIGURE 3.4 - Casque et contrôleurs HTC utilisés pour interagir avec l'application RV

Les participants étaient invités à suivre et à se référer aux mêmes fiches d'instructions que celles présentées lors du chapitre précédent (voir Figure 3.5) pour sélectionner et assembler les bons composants afin d'obtenir exactement le même système final que dans le jeu de données InHARD (Dallel, Havard et Baudry, et al. 2020) (voir Figure 3.2-c).



Instruction sheets

Model	Workstation
BOOSTER	1
Process	Operation
Subsystem 1	10
Tools	
Screwdriver	

Step	Item	Description	Quantities
1	P360	Profile – 360 mm	1
2	LARD	Fixture key	1
3	FIXA1	Fixation	1
4	B820	Bolt M8 x 20	1

Instruction sheets

Model	Workstation
BOOSTER	1
Process	Operation
Final subsystem	90
Tools	
Screwdriver	

Step	Item	Description	Quantities
1	B835	Bolt M8x35	1
2	DI2T	Grooved disc	1
3	ECR8	Nut M8	1

FIGURE 3.5 - Première et dernière fiche d'instructions du flux d'assemblage dans le jeu de données InHARD-DT

Dans la tâche d'assemblage, les opérateurs suivent les étapes suivantes:

1. Consulter la fiche pour comprendre ce qu'il faut assembler (voir Figure 3.5 & Figure 3.6-b).
2. Attraper le composant à placer (composant A) (voir Figure 3.6-a)
3. L'approcher du composant sur lequel le placer (composant B) (voir Figure 3.6-d)
4. Lorsque le participant approche le composant A du composant B, un fantôme du composant à placer s'affiche pour indiquer la position finale du composant A dans le composant B (voir Figure 3.6-d).
5. L'opérateur relâche le bouton et le composant A se place sur le composant B. C'est cela qui est défini comme une association dans la suite du document.

La Figure 3.6 représente un extrait du flux de travail d'assemblage sur la station de travail depuis l'application RV.

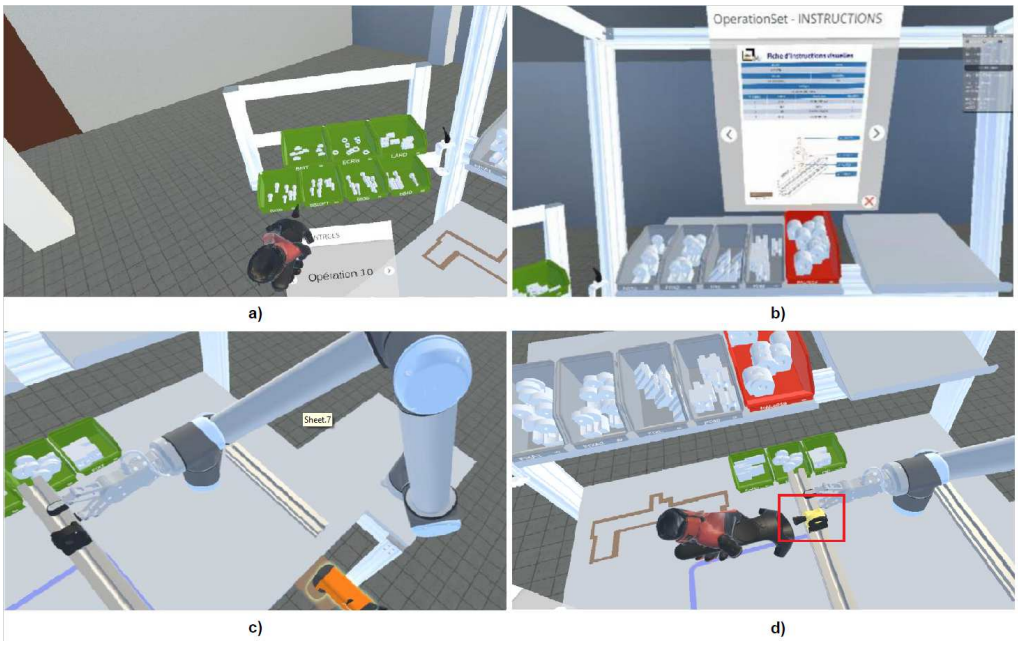


FIGURE 3.6 - Flux de travail d'assemblage depuis l'application RV. a) Picking à gauche. b) Consulter les fiches. c) Prendre visseuse. d) Assembler système

III.3.3 Protocole expérimental

Pour collecter le jeu de données InHARD-DT, nous avons recueilli des données en utilisant deux modalités différentes (RGB et squelette) via deux capteurs, les mêmes que ceux utilisés dans le jeu de données InHARD.

III.3.3.1 Modalités de données et capteurs

Nous avons capturé les données selon différentes modalités. Pour la modalité squelette, nous avons utilisé le capteur de capture de mouvement (MOCAP) « Perception Neuron 32 Edition v2 » (NOITOM-Ltd 2018) pour capturer les données du squelette (voir Figure 3.7). Ce capteur de mouvement utilise des unités de mesure inertielle (IMU) et capture les données avec un taux d'échantillonnage égale à 120 Hz. Il s'agit d'un système de capture de mouvement adaptable et abordable. Il a été initialement développé pour analyser les mouvements pour diverses industries, des cinéastes aux développeurs de jeux, en passant par les chercheurs et les personnels de santé. Les données peuvent être stockées localement sur une carte mémoire ou être transférées via WiFi. Des études récentes ont été menées sur les capteurs Perception Neuron, affirmant leur efficacité à mesurer des mouvements précis du corps entier (Choo, Chow et Komar 2022), (Sers, et al. 2020). Chaque unité IMU correspond à une articulation dans les données du squelette. Ces dernières comprennent les positions 3D (Tx, Ty et Tz) des 17 articulations principales du corps détectées et suivies pendant toute la scène, ainsi que les 3 rotations autour de chaque axe (Rx, Ry et Rz). Les données squelettiques sont ensuite exportées et stockées dans des fichiers au format BVH (Biovision Hierarchical Data) via le logiciel de capture de mouvement nommé « Axis Neuron Pro ».

Pour capturer les données RGB, nous avons utilisé deux caméras « Logitech C920 » situées à la même hauteur mais placées sur deux angles horizontaux différents ($-45°$ et $+45°$) couvrant deux points de vues différents de la même action (côté gauche et côté droit). Les deux flux vidéo ont été synchronisés grâce à la plateforme iMotions (iMotions Biometric Research Platform 7.1 2019) et sont capturés avec une résolution de 1280×720 et une fréquence d'images égal à 30.

Comme le montre la Figure 3.7, le processus d'acquisition du jeu de données InHARD-DT se déroule comme suit : les participants sont d'abord équipés, puis ils sont invités à effectuer les tâches d'assemblage dans la simulation DT pendant que la plateforme iMotions collecte les données et les événements permettant de labéliser automatiquement les données. Une fois l'acquisition terminée, le jeu de données est récupéré pour obtenir le jeu de données final.

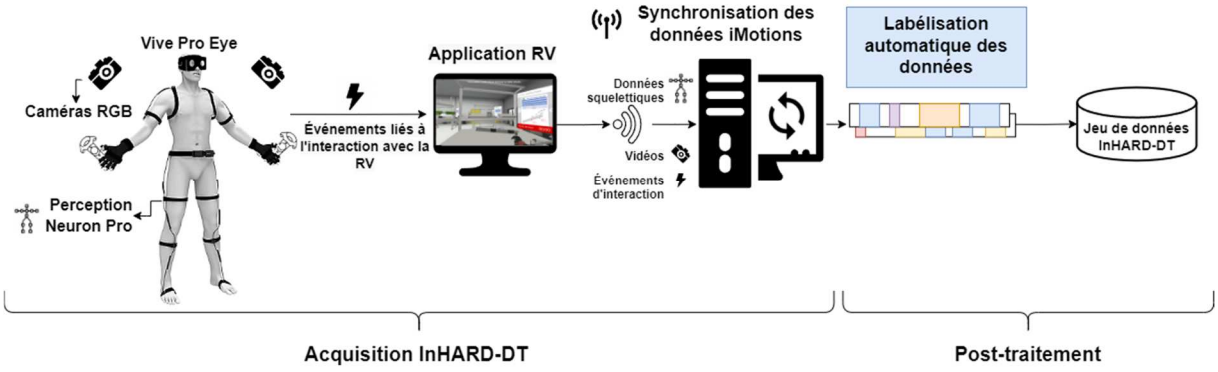


FIGURE 3.7 - Protocole d'acquisition du jeu de données InHARD-DT

Pour chaque étude enregistrée, nous exportons les vidéos et les données squelettiques collectées ainsi que les chronologies des flux de données avec les labels des actions correspondantes. Ces flux de données ont le format intermédiaire suivant qui est traité par la suite pour créer le jeu de données InHARD-DT comme décrit dans la Section II.2.7:

```
{Row, Timestamp, Duration, MarkerName, MarkerDescription}
```

Où :

- Row : représente le numéro de l'image en cours.
- Timestamp : représente le temps écoulé depuis le début de la tâche.
- Duration : représente la durée de chaque action.
- MarkerName : représente le nom de l'événement/action en cours.
- MarkerDescription : représente le nom du composant retenu en fonction de l'opération d'assemblage en cours.

Les sous-sections suivantes détaillent le processus de recrutement des participants et les caractéristiques du groupe expérimental, ainsi que l'installation et la configuration initiales de la RV.

III.3.3.2 Participants et activité demandée

Pour la collecte des données DT, nous avons invité 12 participants distincts de notre laboratoire de recherche (4 femmes et 8 hommes) pour effectuer des tâches d'assemblage dans l'application RV du poste de travail industriel. Les participants étaient un mélange d'experts et de débutants en fonction de leur niveau de manipulation et de leur expertise avec le flux de travail d'assemblage. L'âge des participants était compris entre 25 et 45 ans. Un numéro d'identification unique est attribué à chaque participant et ce dernier effectue la tâche d'assemblage complète une fois. Pour équilibrer les actions qui sont moins représentées que d'autres pendant l'assemblage, nous avons également demandé à chaque participant de répéter chaque action séparément 30 fois. Ainsi, nous nous assurons que les actions moins exécutées sont suffisamment représentées ce qui doit permettre de disposer de suffisamment de données pour entraîner l'architecture de réseau de neurones sélectionnée.

Pour les aspects éthiques, chaque participant a donné son consentement éclairé par écrit, et nous avons anonymisé les données collectées en attribuant un identifiant unique à chaque utilisateur. De plus, nous avons obtenu l'approbation éthique d'un comité d'éthique interne qui a validé le protocole avant de procéder à l'acquisition des données.

III.3.3.3 Labélisation automatique des données

Le processus de labélisation automatique des données pour la HAR a été entièrement intégré dans la « Plateforme de recherche biométrique iMotions » (iMotions Biometric Research Platform 7.1 2019). Cette plateforme intègre et synchronise de manière transparente plusieurs capteurs et données dans une chronologie commune.

Avant de commencer la collecte des données, chaque opérateur doit être physiquement aligné à la même position et orientation. Pour cela, les opérateurs doivent se positionner au centre de la zone d'acquisition afin de pouvoir interagir correctement et facilement avec les différents objets de la scène virtuelle. Une fois cela fait, la scène virtuelle est déplacée et orientée afin de placer le poste de travail virtuel en face de l'opérateur. Par conséquent, chaque participant commence à la même position et même orientation dans le monde physique et dans le monde virtuel.

Pour labéliser les actions des opérateurs au cours de la tâche d'assemblage, nous avons attribué à chaque composant, sous-système, système ou objet impliqué dans la tâche d'assemblage un événement déclencheur spécifique activé dès que l'opérateur interagit avec eux à l'aide des contrôleurs. Ainsi, chaque action effectuée possède son propre moyen de détection au travers d'une interaction en réalité virtuelle. Dans le Tableau 3.1, nous détaillons toutes les actions/évènements prédéfinies et leurs moyens de détection respectifs.

TABLEAU 3.1 - Les actions dans le jeu de données InHARD-DT et leurs moyens de détection

Action/Évènement	**ID**	**Moyen de détection/Description**
Assembler système (Début)	0	Quand le fantôme d'une association est affiché
Assembler système (Fin)	1	Quand une pièce est associée à un objet
Consulter fiches (Début)	2	Quand le regard se pose sur une fiche pendant plus de 300ms
Consulter fiches (Fin)	3	Quand un regard posé sur la fiche ne l'est plus pendant plus de 300ms
Picking en face (Haut)	4	Quand une pièce est prise d'une boite sur la partie haute du poste
Picking en face (Bas)	5	Quand une pièce est prise d'une boite sur la table
Picking à gauche	6	Quand une pièce est prise d'une boite sur la partie gauche du poste
Poser composant	7	Quand une pièce est placée dans la partie bleue du poste (voir Figure 3.3 en bas)
Poser toise	8	Quand la cale est remise à son endroit original
Poser visseuse	9	Quand la visseuse est remise à son endroit original
Poser sous-système	10	Quand la P360 de l'opération 1 est placée sur la table
Changement d'état du robot	11	Quand le robot passe à une autre étape en appuyant sur le bouton précédent ou suivant
Prendre composant	12	Quand une pièce est récupérée de la partie bleue du poste
Prendre toise	13	Quand la cale est prise en main
Prendre visseuse	14	Quand la visseuse est prise en main
Prendre sous-système	15	Quand la P360 de l'opération est récupérée de sa position sur la table
Tourner fiches	16	Quand la page est tournée en appuyant sur le bouton précédent ou suivant
Synchronisation	17	Quand la barre escape est appuyée (Utilisé pour synchroniser les données vidéos et BVH)

Avec l'environnement RV développé, nous sommes capables de générer une grande quantité de données automatiquement labélisées d'une manière précise et en un temps très court. En outre, l'ajout d'actions ou d'objets supplémentaires à la scène RV est une tâche rapide et facile qui ne nécessite pas d'acquérir et de labéliser toutes les données à partir de zéro, ce qui peut être le cas dans une configuration réelle.

Par conséquent, à chaque fois qu'un opérateur effectue une interaction, un label parmi les classes d'action prédéfinies est attribué en fonction des événements déclencheurs décrits précédemment. La Figure 3.8 représente la visualisation de la labélisation des actions en temps réel via la plateforme iMotions. Les marqueurs colorés sur cette figure représentent les actions/évènements réalisés par les opérateurs (voir Tableau 3.1).

FIGURE 3.8 - Visualisation de la labélisation automatique sur le logiciel iMotions. (En bas de la figure) ligne de temps de l'ensemble de l'acquisition avec les marqueurs pour chacun des événements effectués par l'utilisatrice lors de la manipulation dans le jumeau numérique. (À gauche) label de chacun des événements.

L'horodatage d'une action (Timestamp) commence au moment où l'opérateur interagit avec l'objet (ou le système) sur la base des événements déclencheurs définis dans le Tableau 3.1. Comme décrit, les actions « Assembler système » et « Consulter fiches » ont deux événements déclencheurs (début et fin). Les autres actions n'ont qu'un seul événement déclencheur qui caractérise le milieu de l'action. Par conséquent, un processus d'analyse pour transformer l'événement déclencheur en un intervalle d'action est utilisé sur la base de la durée moyenne de chaque action dans le jeu de données InHARD, comme le montre la Figure 3.9. La demi-durée moyenne de l'action de classe k d_k parmi K actions est définie ainsi :

$$d_k = \frac{1}{2 \cdot n_k} \sum_{i=1}^{n_k} l_i^{(k)}, \forall k \in [1, K] \tag{3.1}$$

avec, n_k le nombre d'échantillons d'intervalle de la classe k, K le nombre de classes et $l_i^{(k)}$ la durée en secondes du $i^{ème}$ échantillon de la classe k.

L'intervalle d'action $I_i^{(k)}$ d'un $i^{ème}$ événement déclencheur de classe k apparaissant à l'instant $t_i^{(k)}$ est défini comme suit :

$$I_i^{(k)} = \left[t_i^{(k)} - d_k, t_i^{(k)} + d_k\right], \forall i \in [1, n_k] \tag{3.2}$$

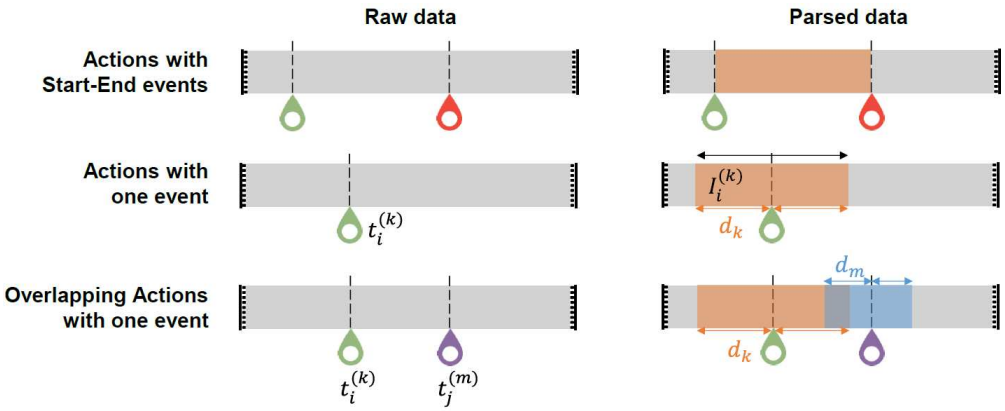


FIGURE 3.9 - Données brutes et analysées dans le jeu de données InHARD-DT

Dans le cas où les horodatages de deux actions consécutives sont très proches, un chevauchement entre ces actions est créé pour s'assurer que le début et la fin de chaque action soient bien labélisés, comme décrit dans la Figure 3.9. Il est à noter que la durée moyenne des actions liée aux gammes d'assemblage est une information importante et donc déterminée dans les activités industrielles et ne nécessiterait donc pas de travail supplémentaire de la part de l'entreprise lors d'une acquisition sur jumeau numérique.

Les statistiques complètes du jeu de données InHARD-DT sont données par le Tableau 3.2.

TABLEAU 3.2 - Statistiques complètes du jeu de données InHARD-DT

Nb. participants	**Nb. samples totales**	**Nb. samples/ action**[10]	**Durée totale/ action**
12	4799	0 : 544	0 : 16 minutes & 40 secondes
		1 : 222	1 : 06 minutes & 20 secondes
		2 : 310	2 : 16 minutes & 13 secondes
		3 : 491	3 : 25 minutes & 28 secondes
		4 : 387	4 : 04 minutes & 54 secondes
		5 : 207	5 : 08 minutes & 45 secondes
		6 : 266	6 : 12 minutes & 37 secondes
		7 : 251	7 : 13 minutes & 46 secondes
		8 : 420	8 : 05 minutes & 41 secondes

[10]
0 : Assembler système
1 : Consulter fiches
2 : Picking en face
3 : Picking à gauche
4 : Poser composant
5 : Poser toise
6 : Poser visseuse
7 : Poser sous système
8 : Prendre composant
9 : Prendre toise
10 : Prendre visseuse
11 : Prendre sous système
12 : Tourner fiches

Nb. participants	Nb. samples totales	Nb. samples/ action[10]	Durée totale/ action
		9 : 364	9 : 17 minutes & 23 secondes
		10 : 533	10 : 21 minutes & 24 secondes
		11 : 335	11 : 16 minutes & 42 secondes
		12 : 469	12 : 15 minutes & 10 secondes

La Figure 3.10 montre l'histogramme de distribution du nombre d'échantillons par action dans le jeu de données InHARD-DT.

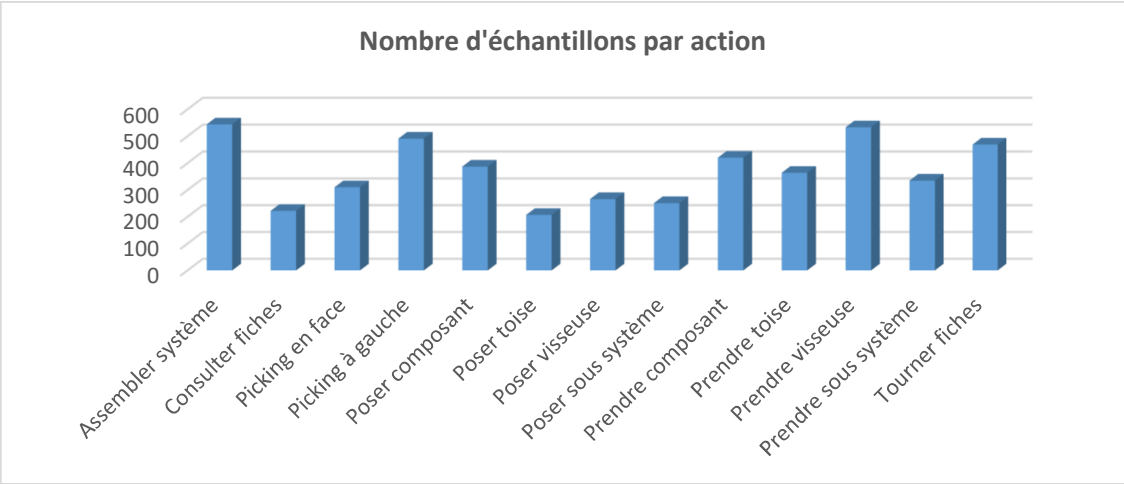


FIGURE 3.10 - Histogramme de distribution du nombre d'échantillons par action dans le jeu de données InHARD-DT

III.3.3.4 Réseaux à graphes ConvNets spatio-temporel pour la HAR

L'objectif de la proposition est d'entrainer une architecture d'apprentissage profond pour la reconnaissance d'actions humaines avec des données issues du jumeau numérique afin de généraliser sur les données issues du jumeau physique. Pour valider notre méthode, nous avons utilisé les données squelettiques recueillies pour entraîner un réseau de neurones convolutifs à graphe spatio-temporel (ST-GCN) développé dans (Yan, Xiong et Lin 2018). L'entrée du module ST-GCN est constituée des vecteurs de coordonnées des 17 articulations principales du corps humain sur les nœuds du graphe. Étant donné les séquences des articulations du squelette, le module ST-GCN capture automatiquement les modèles intégrés dans la configuration spatiale des articulations du squelette et exploite la corrélation entre chaque articulation en construisant un graphe spatio-temporel ainsi que leur dynamique temporelle comprenant les connexions intra-corps et inter-frames.

Comme décrit dans la Figure 3.11, l'architecture du réseau convolutif à graphes spatio-temporel (ST-GCN) se compose d'une couche d'entrée dont la taille des données d'entrée est de 3x17x17 où 3 représente la dimension 3D des 17 principales articulations du corps humain. La couche d'entrée est suivie d'une couche de normalisation qui normalise les données d'entrée dans

toutes les couches. Une succession de 9 couches d'opérateurs de convolution de graphes spatio-temporels est employée pour permettre l'intégration des informations dans les dimensions spatiales et temporelles. Les coordonnées des articulations d'entrée sont concaténées pour former les caractéristiques d'entrée à chaque image. La convolution temporelle est ensuite appliquée sur cette entrée. Les vecteurs de caractéristiques résultants sont ensuite transmis à un classifieur SoftMax. Les modèles ST-GCN sont appris en utilisant la descente de gradient stochastique avec un taux d'apprentissage de 0,01 qui est ajusté pendant l'apprentissage. Le changement de dimensions de toutes les couches peut être observé sur la Figure 3.11.

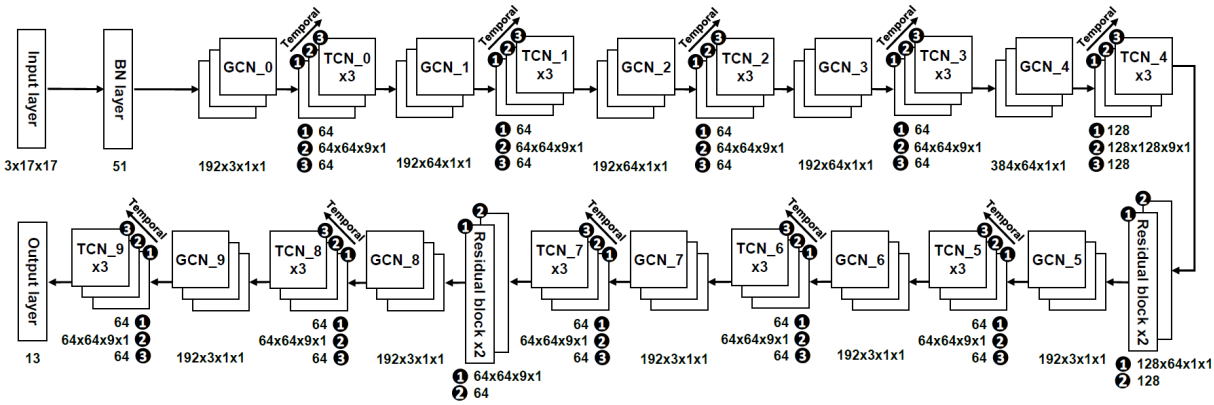


FIGURE 3.11 - Architecture de réseau convolutif à graphes spatio-temporels (ST-GCN)

La Figure 3.12 décrit la structure du réseau de neurones convolutif à graphe spatio-temporel (ST-GCN) utilisé pour la HAR.

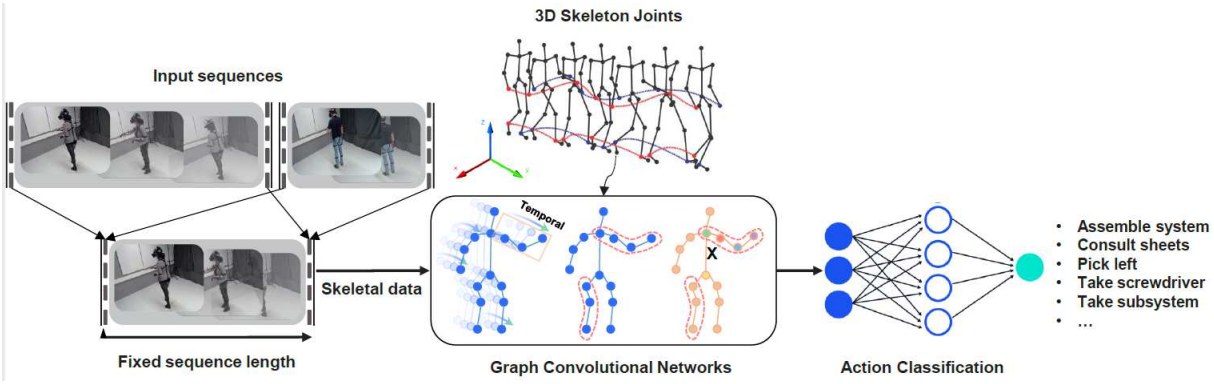


FIGURE 3.12 - La structure du réseau de neurones convolutif à graphe spatio-temporel (ST-GCN) utilisé pour la HAR

III.4 Traitement des données et protocole d'évaluation

Dans la section précédente, nous avons présenté le protocole d'acquisition du jeu de données InHARD-DT, le processus d'auto-labélisation des données ainsi que l'algorithme utilisé pour la HAR. Cette section se concentre sur l'évaluation de la robustesse et de la généralisation de notre méthode, qui consiste à entrainer l'algorithme de HAR avec des données DT et à valider sur des

données PT. Cette évaluation est basée sur la validation croisée entre le jeu de données présenté InHARD-DT dans ce chapitre et le jeu de données existant InHARD.

Tout d'abord, nous détaillerons comment les données sont traitées avant l'apprentissage, puis nous expliquerons les différentes distributions de données DT et PT utilisées lors de l'entrainement pour évaluer notre méthode.

III.4.1 Prétraitement de données

Pour préparer les données d'apprentissage, nous avons utilisé les mêmes prétraitements «Nettoyage de données» , «Mise à zéro de l'articulation de la hanche» et «Ré-échantillonnage» présentés dans la Section II.3 permettant respectivement d'éliminer les fichiers de données où les squelettes des opérateurs sont déformés, les figer au point de départ initial (0,0) et sous-échantillonner l'ensemble des actions pour les adapter à la taille de la longueur de fenêtre spécifiée.

III.4.2 Paramètres des jeux de données

Afin d'évaluer les performances de la HAR de notre méthode en utilisant les données labélisées générées par le DT, deux configurations différentes pour les ensembles d'apprentissage et de test ont été étudiées.

- Configuration de base (PT-PT) : La première expérience a été menée en utilisant uniquement les données acquises sur le PT, c'est-à-dire en utilisant uniquement le jeu de données InHARD. Par conséquent, les phases d'apprentissage, de validation et de test ont été réalisées en utilisant uniquement ces données. L'objectif principal de cette expérience est d'obtenir une performance de référence. Celle-ci permettra de comparer, par la suite, les résultats obtenus par apprentissage avec les données générées par le DT provenant du jeu de données InHARD-DT.

- Configuration jumeau numérique/physique (DT-PT) : Dans cette expérience, qui constitue l'objectif principale de ce chapitre, nous utiliserons les données DT du jeu de données InHARD-DT pour former notre réseau de neurones, puis nous le testerons à l'aide des données PT, démontrant ainsi l'applicabilité de notre méthode.

En résumé, lors de l'apprentissage de notre modèle avec les données PT, nous injecterons progressivement des données DT provenant des données InHARD-DT dans les données PT du jeu de données InHARD dans les phases d'apprentissage et de validation, afin de voir si le modèle peut correctement généraliser sur les données PT en apprenant sur des données DT. La phase de test, quant à elle, est toujours menée en utilisant uniquement les données PT. L'objectif principal de nos expériences est d'éliminer complètement les données PT et de se baser uniquement les données DT dans les phases d'apprentissage et de validation. Par conséquent, nous avons défini

différents paramètres pour la phase d'apprentissage comme décrit dans le Tableau 3.3. Les Figure 3.13 et Figure 3.14 montrent les histogrammes de distribution des actions dans les jeux de données InHARD et InHARD-DT respectivement. Nous remarquons que l'acquisition de données DT avec la labélisation automatique a permis de compléter le jeu de données avec des échantillons d'actions qui sont moins représentés.

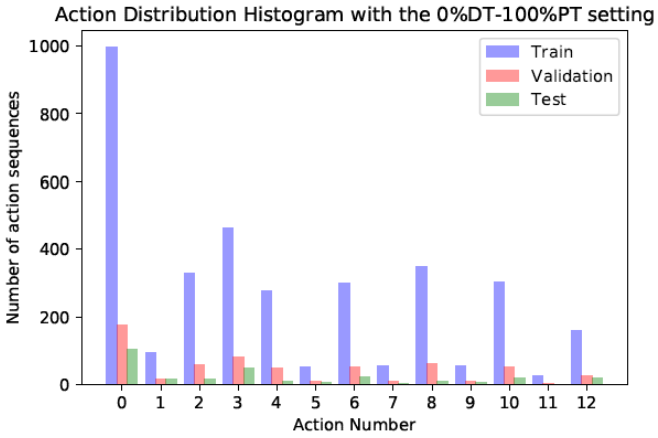


FIGURE 3.13 - Histogramme de distribution des actions dans le jeu de données InHARD

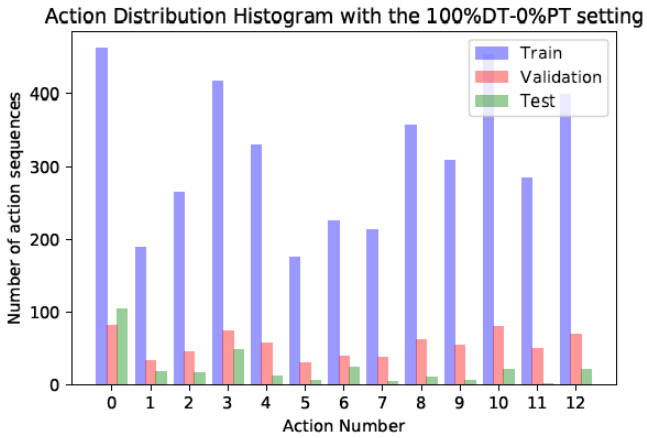


FIGURE 3.14 - Histogramme de distribution des actions dans le jeu de données InHARD-DT

TABLEAU 3.3 - Configuration de la répartition des données InHARD et InHARD-DT lors de la phase d'apprentissage.

Configuration	**Données d'apprentissage et de validation**	**Données de test**
0%DT / 100%PT	Seules les données PT sont utilisées	Données PT
25%DT / 75%PT	25% des données DT injectées à 75% des données PT	Données PT
50%DT / 50%PT	50% des données DT injectées à 50% des données PT	Données PT
75%DT / 25%PT	75% des données DT injectées à 25% des données PT	Données PT
100%DT / 0%PT	Seules les données DT sont utilisées	Données PT

Afin de valider les résultats de notre méthode, nous avons veillé à utiliser, dans chaque configuration, le même pourcentage de données injectées que pour la configuration précédente. Par exemple, dans le cas de la configuration $DT\ 50\%\ /\ PT\ 50\%$, nous avons utilisé les mêmes

25% des données DT de la configuration *DT* 25% / *PT* 75% précédente et nous avons ajouté 25% de données DT et retiré 25% de données PT comme expliqué dans la Figure 3.15.

En outre, afin d'avoir une comparaison équitable entre les différentes configurations et afin de démontrer la généralisation des données DT avec les données PT, il convient de noter que dans chaque configuration, l'apprentissage de l'architecture ST-GCN a toujours été effectué à partir de zéro, c'est-à-dire qu'aucun réglage fin ou apprentissage par transfert n'a été appliqué au modèle lors de la fusion des données DT et PT.

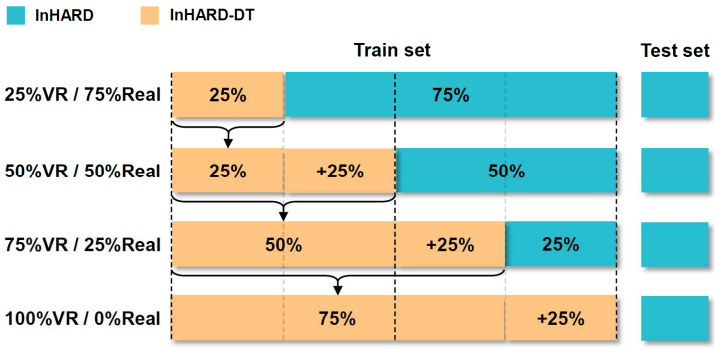


FIGURE 3.15 - Préparation des ensembles d'apprentissage et de test InHARD-DT/InHARD

III.4.3 Métrique d'évaluation

Pour calculer la précision entre une séquence d'actions prédites et les labels de classe de la vérité terrain, nous avons utilisé les F1-scores moyens et la précision moyenne de toutes les actions. Le F1-score est utilisé pour interpréter la moyenne harmonique de la précision et du rappel. Il atteint sa meilleure valeur à 1 et son pire score à 0. La formule du F1-score est exprimée comme suit :

$$F1\text{-}Score = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall} \tag{3.3}$$

III.5 Résultats et discussion

Dans la section précédente, nous avons présenté les différents traitements proposés pour préparer nos données ainsi que le protocole d'évaluation à suivre.

Dans cette section, nous évaluons la performance de HAR de notre méthode sur les jeux de données InHARD et InHARD-DT en utilisant les configurations et métrique mentionnées précédemment. Pour rappel, il y a 13 classes d'actions définies dans le jeu de données InHARD qui sont : « Assembler système, Consulter fiches, Picking en face, Picking à gauche, Poser composant, Poser toise, Poser visseuse, Poser sous-système, Prendre composant, Prendre toise, Prendre visseuse, Prendre sous-système et Tourner les fiches ».

Nous avons utilisé le réseau de neurones convolutif à graphe spatio-temporel développé par

(Yan, Xiong et Lin 2018) pour tester notre méthode. Les résultats obtenus avec les différentes configurations sont décrits dans le Tableau 3.4. Ce dernier montre l'Accuracy et les F1-scores moyens de toutes les actions sur les jeux de données InHARD et InHARD-DT. Nous pouvons voir que la méthode proposée obtient de bons résultats dans les configurations DT et PT. Malgré la différence entre l'environnement et la façon dont une action est exécutée dans les configurations PT et DT, notre méthode a montré sa robustesse.

TABLEAU 3.4 - Résultats de la méthode proposée sur les jeux de données InHARD et InHARD-DT avec les différentes configurations

Configuration	**Accuracy**	**F1-score**
0%DT / 100%PT	0.906	0.906
25%DT / 75%PT	**0.956**	**0.955**
50%DT / 50%PT	0.902	0.901
75%DT / 25%PT	0.875	0.880
100%DT / 0%PT	0.889	0.888

Pour mieux analyser les résultats, les Figures Figure 3.16, Figure 3.17, Figure 3.18, Figure 3.19 et Figure 3.20. montrent les matrices de confusion des jeux de données InHARD/InHARD-DT en utilisant respectivement les configurations $0\%DT\ /\ 100\%PT$, $25\%DT\ /\ 75\%PT$, $50\%DT\ /\ 50\%PT$, $75\%DT\ /\ 25\%PT$ et $100\%DT\ /\ 0\%PT$.

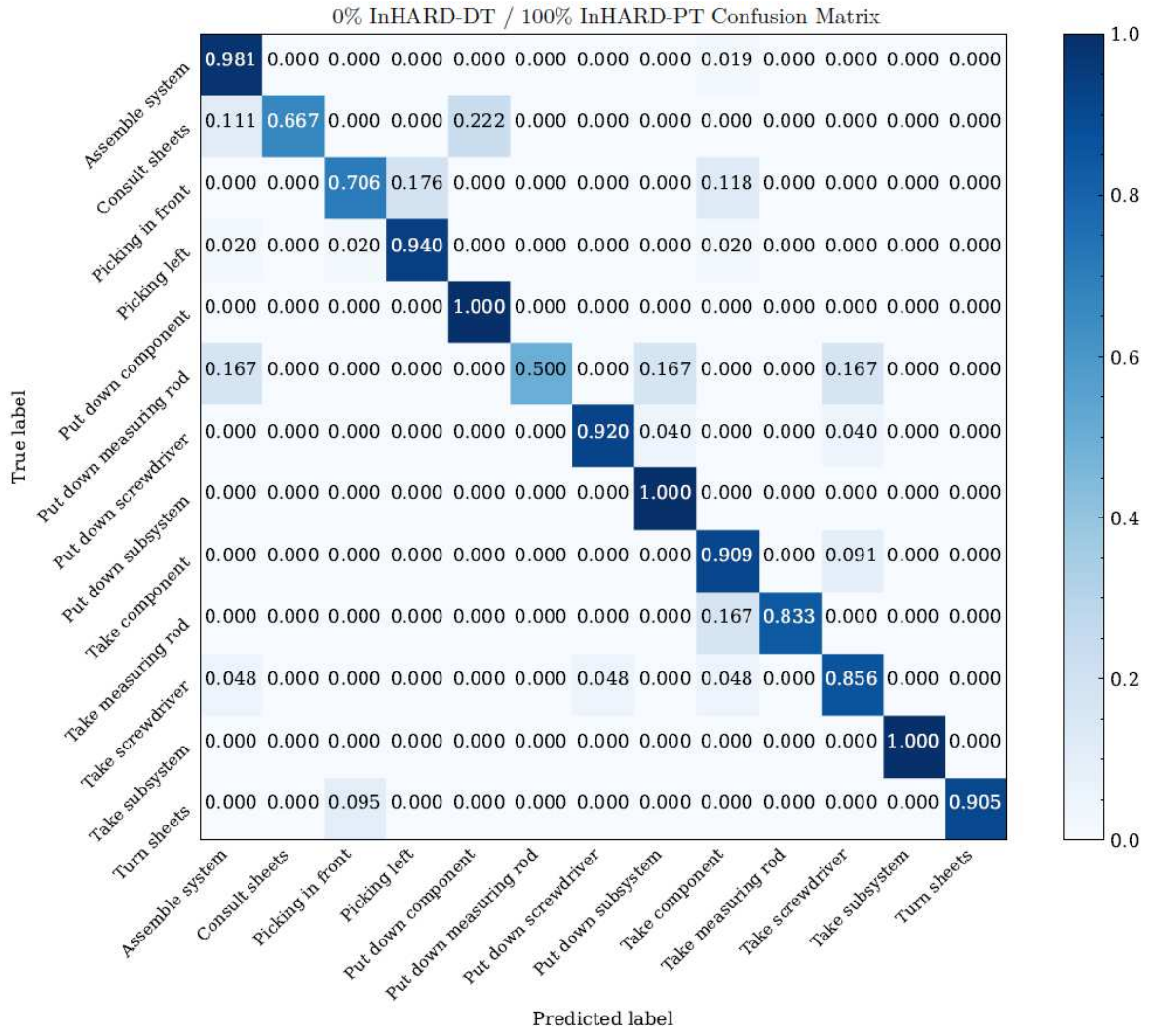


FIGURE 3.16 - Matrice de confusion InHARD avec la configuration 0%DT/100%PT

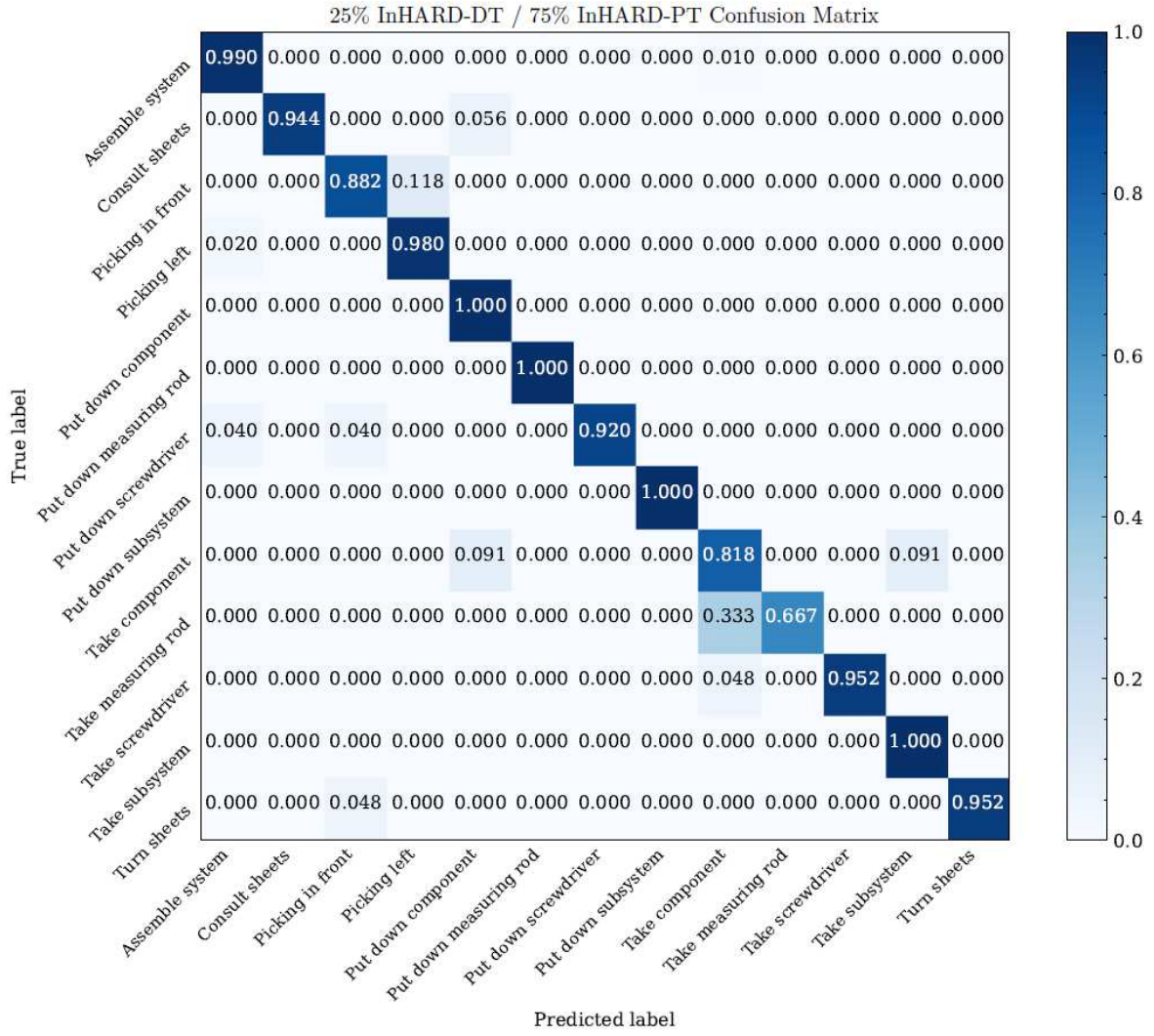


FIGURE 3.17 - Matrice de confusion InHARD/InHARD-DT avec la configuration 25%DT/75%PT

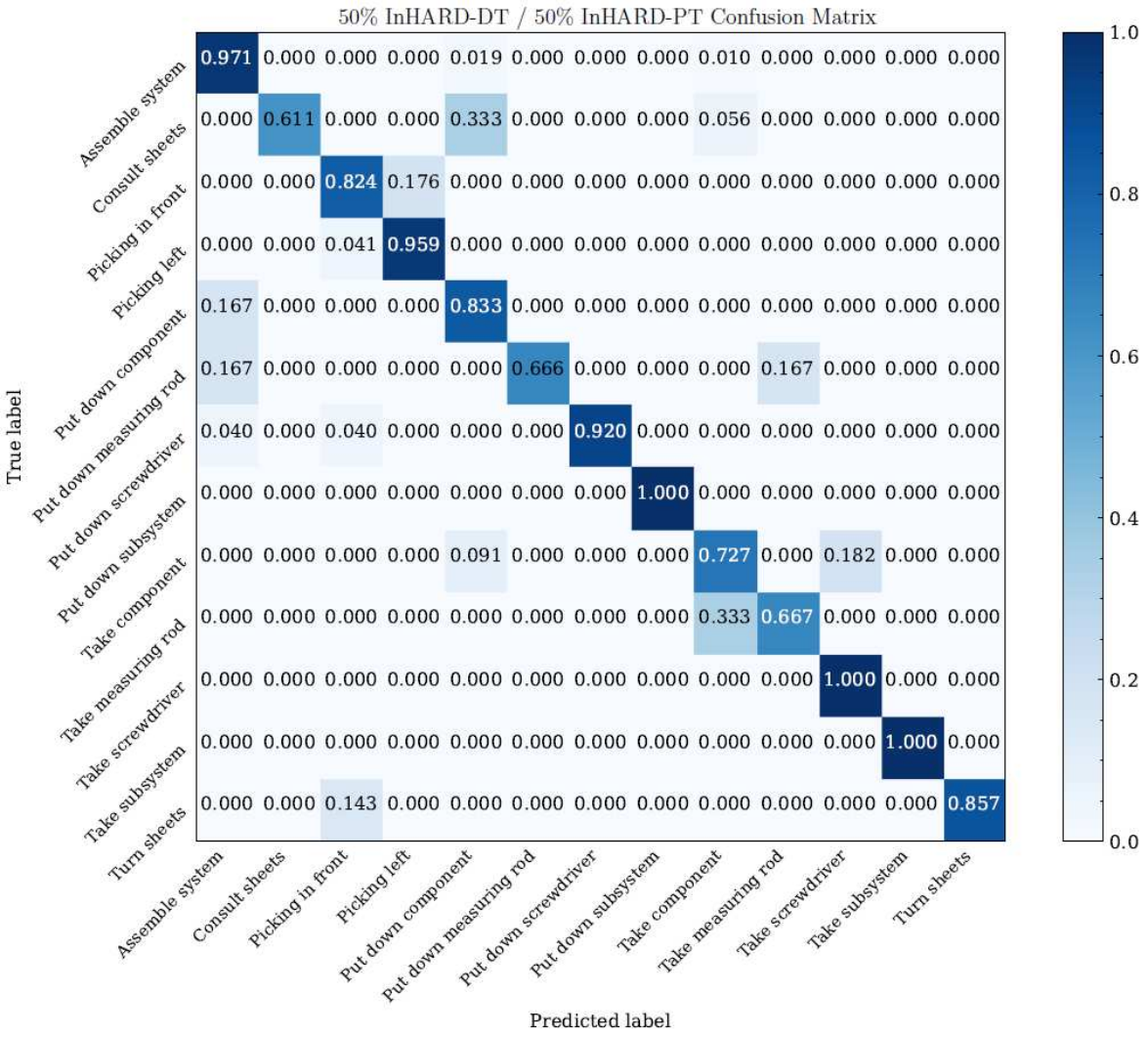


FIGURE 3.18 - Matrice de confusion InHARD/InHARD-DT avec la configuration 50%DT/50%PT

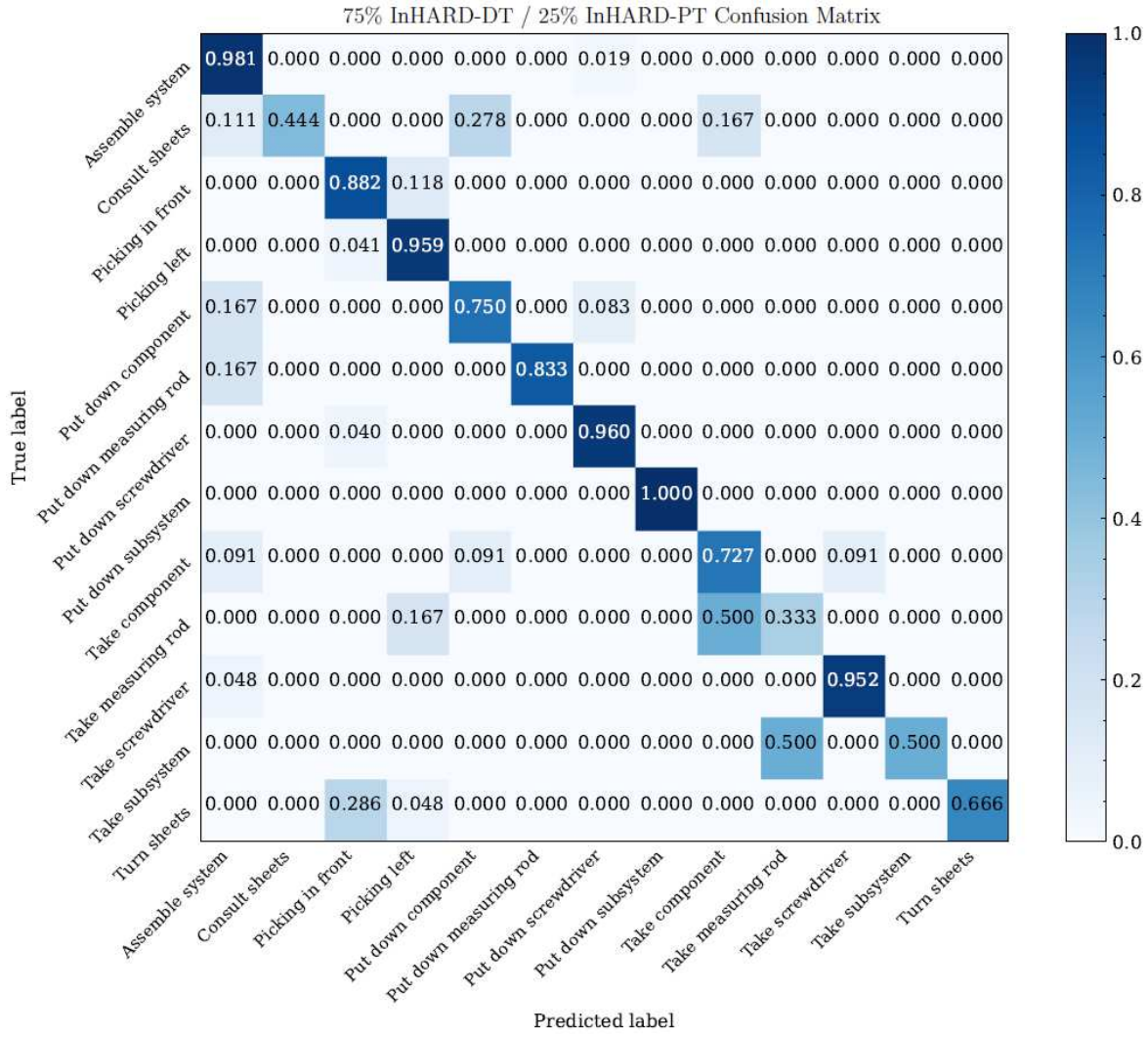


FIGURE 3.19 - Matrice de confusion InHARD/InHARD-DT avec la configuration 75%DT/25%PT

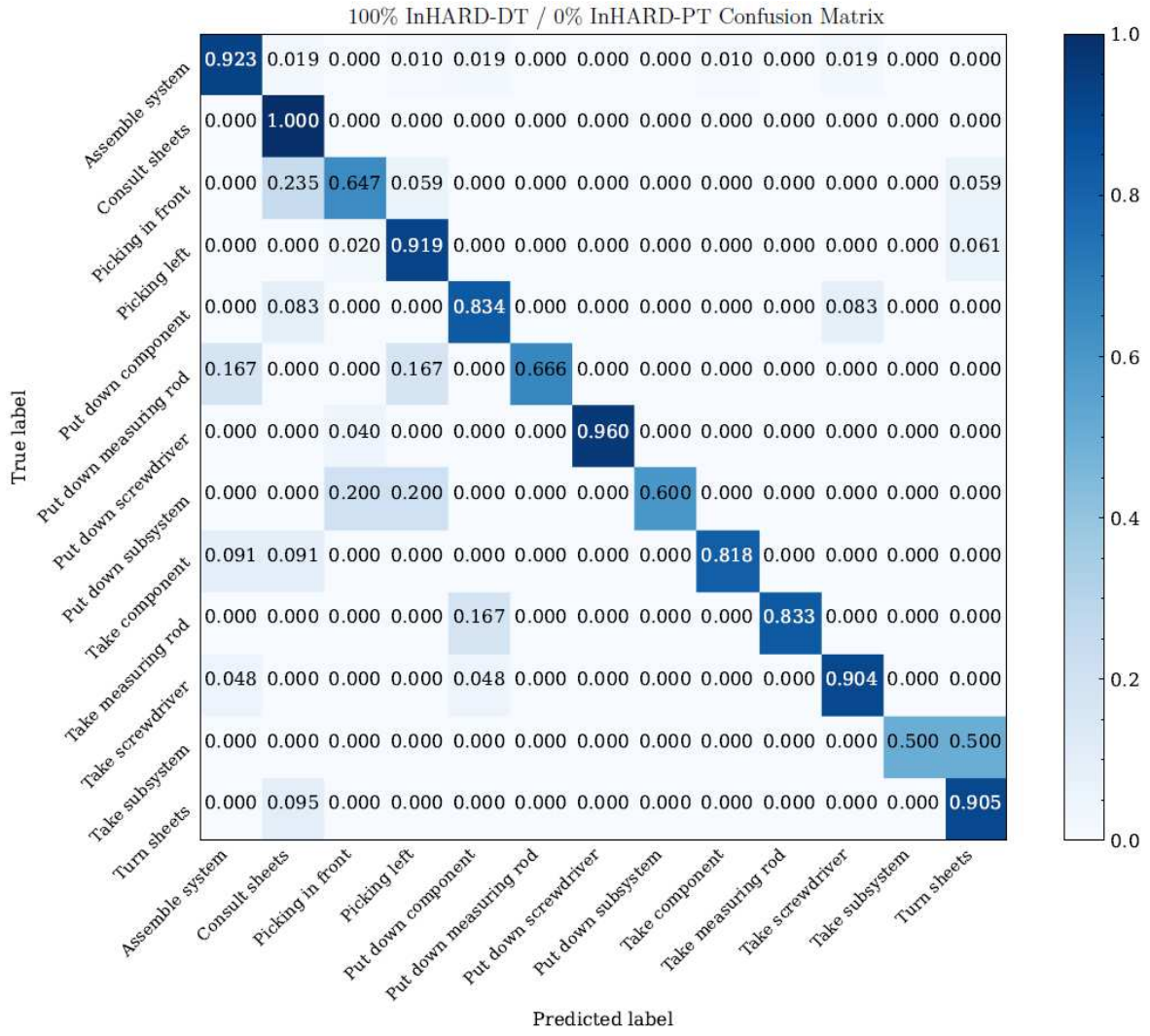


FIGURE 3.20 - Matrice de confusion InHARD/InHARD-DT avec la configuration 100%DT/0%PT

Avant de discuter les résultats obtenus, quelques observations préliminaires peuvent être faites sur les jeux de données InHARD et InHARD-DT. Tout d'abord, nous pouvons constater que la manière dont certaines actions sont exécutées dans le monde réel dans le jeu de données InHARD diffèrent de celle exécutées dans la RV dans le jeu de données InHARD-DT. Par exemple, le début de l'action « Consulter fiches » dans le monde réel du jeu de données InHARD est identifié par le fait de pointer les yeux de l'opérateur vers les fiches et de se pencher un peu vers elles, alors que dans InHARD-DT, cette action est déclenchée lorsque les lentilles du casque VR reposent sur une fiche pendant plus de $300ms$, ce qui peut se produire pendant l'exécution d'autres actions sans l'intention de regarder les fiches. De même, l'action « Tourner fiches », dans le jeu de données InHARD, est identifiée par le mouvement des mains qui glissent et tournent, alors que cette action dans InHARD-DT est effectuée en déplaçant les contrôleurs et en cliquant sur un bouton placé sur les fiches d'instructions. En outre, les opérateurs semblent manipuler les objets plus facilement dans le monde réel que dans la RV, en particulier lorsqu'il s'agit de très petits objets. Par exemple, les participants dans la RV s'y reprennent parfois à

plusieurs fois pour attraper les petits objets en cliquant sur les contrôleurs jusqu'à ce qu'ils les attrapent réellement. Néanmoins, comme nous avons utilisé les mêmes capteurs lors de l'acquisition des jeux de données PT et DT, les actions réalisées dans les deux jeux de données impliquaient le corps entier et pas seulement une partie de celui-ci, comme les doigts par exemple, ce qui donne des actions de haut niveau. Par conséquent, le système apprend des mouvements impliquant l'ensemble du corps à partir des données physiques et numériques et obtient de bons résultats malgré quelques différences dans la façon dont certaines actions sont réalisées dans le DT et le PT.

En outre, si nous devions apprendre des mouvements fins, par exemple dans lesquels les actions impliqueraient les doigts, nous aurions besoin de dispositifs plus spécifiques, comme des gants de capture de mouvement. Notamment, les auteurs (Noblecourt, et al. 2021) ont étudié l'impact des contrôleurs utilisés dans la formation RV (Knuckles ou Vive Controllers) sur le temps passé à attraper les objets virtuels et ont prouvé que le type de contrôleurs avaient un impact non négligeables sur la manipulations de petits objets, comparé à des plus gros, en réalité virtuelle. Nous pouvons donc supposer que ce phénomène aurait un impact sur la labélisation des données.

L'acquisition de données DT avec la labélisation automatique permet de compléter un jeu de données avec des échantillons d'actions qui sont moins représentés. Par exemple, avec le jeu de données InHARD-DT, en plus du processus complet de la tâche d'assemblage, nous avons également demandé aux opérateurs de répéter chaque action 30 fois pour équilibrer les actions qui sont moins exécutées que d'autres au cours du processus d'assemblage complet. Cela permet de compléter ce jeu de données, même si certaines actions peuvent varier dans l'exécution et perdre en spontanéité en raison du processus de répétition. Par exemple, des actions telles que « Poser sous-système » et « Prendre sous-système » sont beaucoup plus représentées dans le jeu de données InHARD-DT que dans le jeu de données InHARD, car elles ne sont effectuées qu'une à deux fois au cours de la manipulation complète d'assemblage.

Dans la Figure 3.20, avec la configuration $100\%DT/0\%PT$, nous pouvons clairement observer que notre méthode a toujours une bonne performance sur presque toutes les actions. Toutefois, quelques baisses de performance sont notables sur certaines actions comme « Consulter fiches » et « Prendre sous-système ». Au final, la précision moyenne de la configuration $100\%DT/0\%PT$, atteint 88.93%, en comparaison avec configuration de base $0\%DT/100\%PT$ qui atteint 90.60% (voir Figure 3.16). De plus, comme nous pouvons le voir dans le Tableau 3.4, avec la configuration $25\%DT/75\%PT$, nous avons obtenu une précision de 95,6% avec une augmentation de 5% par rapport à la configuration de base $0\%DT/100\%PT$. Nous pensons que les données DT ont équilibré les actions qui sont très peu représentées dans le jeu de données InHARD, comme « Consulter fiches », « Poser toise », « Prendre sous-système », etc. ce qui a amélioré les performances de HAR.

III.6 Conclusion

Dans ce chapitre, nous avons proposé une nouvelle méthode de DT couplée à la RV pour obtenir un modèle humain numérique utilisé pour générer des données auto-labélisées pendant une procédure d'assemblage industriel. Tout d'abord, nous avons introduit un jeu de données de reconnaissance d'actions humaines industrielles, appelé InHARD-DT, dans lequel des opérateurs humains ont effectué des tâches d'assemblage basées sur un scénario réel d'utilisation, dérivé de notre jeu de données InHARD. Sur la base du DT d'un poste de travail industriel utilisant la RV, nous avons pu collecter une grande quantité de données labélisées automatiquement que nous avons utilisées par la suite pour entrainer le réseau de neurones convolutif à graphe spatio-temporel (ST-GCN) pour la HAR.

Nous avons entraîné ce modèle en utilisant différentes configurations dans lesquelles nous avons progressivement injecté des données DT du jeu de données InHARD-DT dans les données PT dans les phases d'apprentissage et de validation pour but d'éliminer complètement les données PT et de ne compter que sur les données DT. La phase de test d'autre côté a été menée en utilisant uniquement les données PT.

En utilisant différentes configurations d'apprentissage et de validation, les expériences sur les deux jeux de données à grande échelle InHARD et InHARD-DT démontrent l'efficacité de notre méthode. Nous avons montré que notre approche permettait de surmonter les problèmes de labélisation en s'appuyant sur un générateur de données auto-labélisées en utilisant le DT. Notre méthode a permis d'obtenir une précision moyenne de 90.6% et un F1-score de 90.6% dans la configuration de base en utilisant que les données PT dans les phases d'apprentissage, validation et de test. En entrainant l'algorithme avec les données DT seulement et en testant avec les données PT, nous avons obtenu une précision moyenne de 88.9% et un F1-score de 88.8%, ce qui représente une baisse de moins de 2% par rapport à la configuration de référence basée sur des données réelles. Cela montre la pertinence de l'approche proposée.

Dans le cadre des perspectives à ces travaux, nous avons l'intention d'explorer des mélanges d'approches (modèles hybrides) et de passer à la labélisation floue (Rad et Balas 2020) des actions pour prendre en compte les approximations induites par la labélisation unique des actions en ayant 3 événements déclencheurs par action qui représentent respectivement l'état de début, en cours et de fin de chaque action, ce qui aboutira à un découpage plus précis des séquences d'actions. Nous aimerions également capturer des mouvements fins, pour lesquels nous aurions besoin de capteurs plus spécifiques pour classer les actions impliquant uniquement la main. De plus, nous prévoyons d'incorporer plus d'informations contextuelles telles que l'apparence des objets ou l'interaction humain-objet pour aider à reconnaître des actions plus complexes en réduisant l'ambiguïté et le mélange entre des actions qui peuvent avoir des poses similaires, en particulier lorsqu'il s'agit de très petits objets, ce qui peut donc améliorer les performances de la

HAR mais aussi permettre d'avoir un modèle DT plus précis et donc correspondant davantage au PT.

Le développement d'algorithmes de HAR précis et efficaces reste une tâche difficile en raison de la grande variabilité des formes et des postures humaines, ainsi que de la complexité de leurs mouvements, mais surtout lorsqu'il s'agit d'utiliser des flux de données continus ou non découpés. Alors que la HAR à partir de séquences segmentées a été intensivement étudiée et développée ces dernières années, la HAR en ligne reste une tâche complexe et est moins développée. Dans le chapitre suivant, nous proposons une approche basée sur des données squelette avec une technique de fenêtre glissante et de vote majoritaire pour la HAR en ligne en utilisant des réseaux de neurones convolutifs à graphe spatio-temporel nommé STGCN-SWMV.

Chapitre 4

Mise en place d'un algorithme de reconnaissance d'actions humaines en ligne

Sommaire

IV.1 Introduction

De nos jours, la reconnaissance des actions humaines (HAR) en ligne est devenue un problème important car elle est largement utilisée dans la vidéo-surveillance, la collaboration humain-robot dans l'industrie, etc. (Dallel, Havard et Baudry, et al. 2020). La HAR en ligne représente une thématique de grande envergure mais reste une tâche difficile et moins développée que la HAR segmentée.

La reconnaissance en temps réel des actions humaines à partir de flux de données squelettes est au cœur de plusieurs applications. Il permet une coordination homme-machine fluide et contribue à améliorer la sécurité au travail en vérifiant les situations dangereuses. Cependant, déterminer le début et la fin de chaque action en temps réel rend la tâche plus complexe pour l'algorithme.

La reconnaissance d'actions en ligne a connu une croissance rapide ces dernières années. Avec

des séquences d'actions partiellement observées, elle vise à localiser le segment d'action, qui peut également être appliqué en temps réel. Comme abordé dans le Chapitre I, il existe deux types de reconnaissance d'action dans les algorithmes de HAR : la reconnaissance d'action segmentée (hors ligne) et la reconnaissance d'action en ligne. La reconnaissance en ligne présente deux défis principaux par rapport à la reconnaissance segmentée. Outre le fait d'avoir un coût de calcul plus élevé, elle doit aussi reconnaître l'action en cours, tout en étant capable de la segmenter. Contrairement à la reconnaissance d'action hors ligne, qui détermine l'action après son observation complète, la reconnaissance d'action en ligne vise à détecter l'action à la volée, le plus tôt possible tout en localisant précisément et rapidement le type, le début et la fin de chaque action dans le temps (Dallel, Havard et Dupuis, et al. 2022).

Bien que la reconnaissance d'actions en ligne soit cruciale pour de nombreuses applications de nos jours notamment dans un contexte industriel par exemple pour faciliter la collaboration humain-robot qui nécessite une coordination fluide et efficace entre le robot et les opérateurs humains, peu de travaux ont été proposés pour aborder ce problème. Pour atteindre cet objectif, la reconnaissance d'actions humaines en temps réel est indispensable. Par conséquent, nous étudions le problème de la reconnaissance d'action en ligne en utilisant des données squelettiques continues et nous proposons une méthode nommée STGCN-SWMV (Spatial-Temporal Graph Convolutional Neural Network with a Sliding Window and Majority Voting) basée sur les réseaux de neurones convolutifs à graphe spatio-temporel (ST-GCN) et la technique de fenêtre glissante avec un vote majoritaire.

Les principales contributions de ce chapitre sont résumées comme suit :

- Nous étudions le problème de la HAR en ligne en utilisant les données squelettiques en continu qui est difficile et moins développé dans la littérature que la HAR segmentée en tirant parti d'une approche basée sur la fenêtre glissante et le vote majoritaire à l'aide du module ST-GCN.
- Les résultats expérimentaux démontrent l'efficacité de notre méthode évaluée sur quatre jeux de données à base de squelette en ligne nommés InHARD & InHARD-3-DT, OAD et UOW dépassant les performances sur la HAR en ligne.

IV.2 Méthode STGCN-SWMV pour HAR en ligne

Sur la base des travaux de (Yan, Xiong et Lin 2018), nous proposons un réseau de neurones convolutifs à graphe spatio-temporel utilisant une approche de fenêtre glissante et de vote majoritaire (STGCN-SWMV) pour résoudre le problème de reconnaissance d'actions humaines en ligne. La structure de cette approche est illustrée par la Figure 4.1.

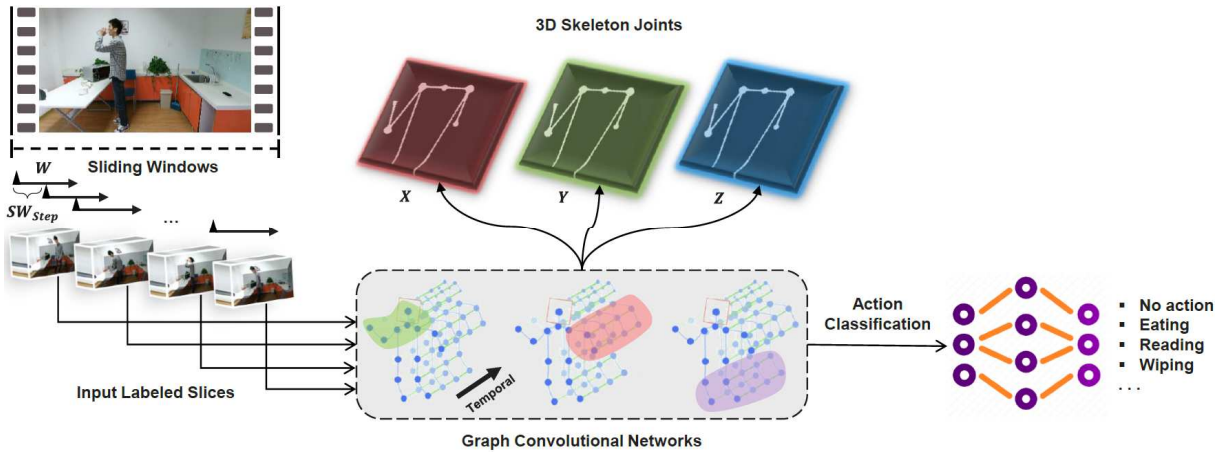


FIGURE 4.1 - Structure du réseau de neurones convolutif à graphe spatio-temporel proposé pour la reconnaissance d'actions en ligne

Les entrées du module STGCN-SWMV sont les vecteurs de coordonnées des articulations sur les nœuds du graphe $F(v_{ti})$ qui se compose de vecteurs de coordonnées des articulations de squelette au cours du temps. Compte tenu des séquences des articulations squelettiques, le module ST-GCN exploite la corrélation entre chaque articulation squelettique en construisant un graphe spatio-temporel non orienté $G = (V, E)$ sur une séquence squelettique avec N articulations et T images avec $V = \{v_{ti} | t = 1, .., T ; i = 1, .., N\}$ (V : nœud, E : arrête) comme expliqué dans la Section II.4.

La construction du graphe G se fait en deux étapes où les jointures d'une image particulière sont reliées par des arcs, puis chaque jointure est connecté à la même jointure sur des images consécutives. Ainsi, la trajectoire d'une articulation particulière au cours du temps est caractérisée par toutes les arêtes du graphe comme le montre la Figure 2.18 du Chapitre II.

La sortie d'une opération de convolution s'appliquant sur des images 2D ou des cartes de caractéristiques, pouvant toutes deux être traitées comme des grilles 2D, est à nouveau une grille 2D. Avec une foulée égale à 1 et un padding approprié, les cartes de caractéristiques en sortie peuvent avoir la même taille que les cartes de caractéristiques en entrée. Étant donné un opérateur de convolution avec une taille du noyau de $K \times K$, et une carte de caractéristiques en entrée f_{in} avec c canaux. La valeur de sortie pour un seul canal à l'emplacement spatial x peut-être écrite comme suit :

$$f_{out}(x) = \sum_{h=1}^{K} \sum_{w=1}^{K} f_{in}\big(p(x,h,w)\big) \cdot w(h,w) \tag{4.1}$$

où p est une fonction d'échantillonnage qui énumère les voisins de l'emplacement x et w une fonction de poids qui fournit un vecteur de poids dans l'espace réel de dimension c pour calculer le produit scalaire avec les vecteurs de caractéristiques d'entrée échantillonnés de dimension c.

L'équation de ce ST-GCN pour le $i$ème nœud à un instant t, n_{ti}, est formulée comme suit :

$$f_{out}(n_{ti}) = \sum_{n_{tj} \in B(n_{ti})} \frac{1}{Z_{ti}(n_{tj})} f_{in}(n_{tj}) \cdot w(l_{ti}(n_{tj})) \tag{4.2}$$

où $B(n_{ti})$ représente les voisins du nœud n_{ti}, Z représente la cardinalité du sous-ensemble correspondant, f_{in} est la carte des caractéristiques d'entrée et w est la fonction de poids similaire au noyau d'une convolution 2D. Pour la fonction de poids, en convolution 2D, une grille fixe existe naturellement autour de l'emplacement central. Les pixels voisins peuvent avoir un ordre spatial fixe. Dans le ST-GCN, il simplifie le processus en configurant le partitionnement spatial.

IV.2.1 Approche de fenêtre glissante

La fenêtre glissante est la technique la plus utilisée dans la littérature pour la reconnaissance d'actions en ligne car elle ne nécessite pas de prétraitement des données et elle est plus adaptée pour des applications temps réel (Dehghani, et al. 2019). Cette technique consiste à découper le signal à analyser en une série de fenêtres de taille fixe. Une superposition entre les séries générées peut être utilisée ; elle peut varier de 10 % à 90 %. Ce fenêtrage est notamment utilisé pour la reconnaissance d'activités périodiques (par exemple la marche ou la course) et pour la reconnaissance d'activités statiques (par exemple le repos). La taille de la fenêtre glissante a un impact majeur sur la précision de la reconnaissance (Dehghani, et al. 2019) (Banos, et al. 2014) (G. Wang, et al. 2018).

Dans nos travaux, le choix de la taille de la fenêtre glissante est basé sur ce qui est le plus utilisé dans la littérature, i.e. en utilisant la durée moyenne de toutes les actions (Banos, et al. 2014) (G. Wang, et al. 2018) (Mehrang, Pietilä et Korhonen 2018). La taille de la fenêtre glissante, W, suit l'équation suivante :

$$W = \frac{1}{N} \sum_{i=1}^{N} Seq_{duration}(i) \tag{4.3}$$

où N désigne le nombre de séquences d'action d'entrée et $Seq_{duration}$ désigne la durée de la $i$ème séquence d'action.

La taille du pas de la fenêtre glissante, SW_{step}, représentant le nombre d'images passées entre deux classifications de fenêtre glissante, est définie sur 1 image, ce qui entraîne un chevauchement entre les fenêtres glissantes. De cette façon, nous nous assurons que toutes les actions, y compris les actions courtes comme poser/prendre composant, sont bien évaluées. Par exemple, si une action donnée est courte (inférieur ou égale à 1 seconde), une taille de pas de la fenêtre glissante équivalente à 1 image permettra de générer plusieurs échantillons représentant cette action. Cela

permettra également à l'algorithme de mieux se généraliser en s'entrainant sur différents échantillons de la même action. De plus, un tel chevauchement permet, d'une part, d'avoir une quantité de données suffisante lors de l'apprentissage, d'autre part, lors de la phase de test, la fenêtre glissante évaluera plusieurs fois la même image, ce qui permettra de mettre en place le vote majoritaire que nous présentons dans la section suivante.

IV.2.2 Vote majoritaire

IV.2.2.1 Principe du vote majoritaire

Un vote majoritaire est un modèle d'apprentissage automatique qui combine les prédictions de plusieurs classifieurs. Il s'agit d'une technique qui peut être utilisée pour améliorer les performances d'un classifieur en combinant les prédictions de plusieurs modèles. Il peut être utilisé pour la classification ou la régression (Witten, Frank et Hall 2011). Dans le cas de la régression, les prédictions des modèles sont moyennées. Dans le cas de la classification, les prédictions pour chaque label sont additionnées et le label ayant la majorité des votes est prédit.

Le vote majoritaire pondéré se formalise suivant l'équation suivante :

$$\hat{y} = \underset{i \in K}{argmax} \sum_{j=1}^{N_v} \alpha_j \boldsymbol{C_j} \tag{4.4}$$

où $\hat{y}$ est le label de classe prédite à l'image t, K est le nombre de classes, N_v est le nombre de classifieurs (ou de votes), α_j est un coefficient de pondération appliqué au classifieur $\boldsymbol{C_j}$ et $\boldsymbol{C_j}$ représente la sortie vectorielle (soft ou hard) représentant la probabilité de chacune des K labels de classe.

Dans le cas d'une fenêtre glissante sur une ligne de temps, le nombre de votes dépend de la taille et du pas de la fenêtre glissante. Il est défini comme suit :

$$N_v = \frac{W + 1}{SW_{step}} \tag{4.5}$$

Lorsque les coefficients α_j sont égaux, l'équation précédente peut être simplifiée et écrite comme suit :

$$\hat{y} = \text{mode}\{C_1(x), C_2(x), \dots, C_n(x)\} \tag{4.6}$$

L'approche de la fenêtre glissante avec le principe du vote majoritaire est illustrée par la Figure 4.2.

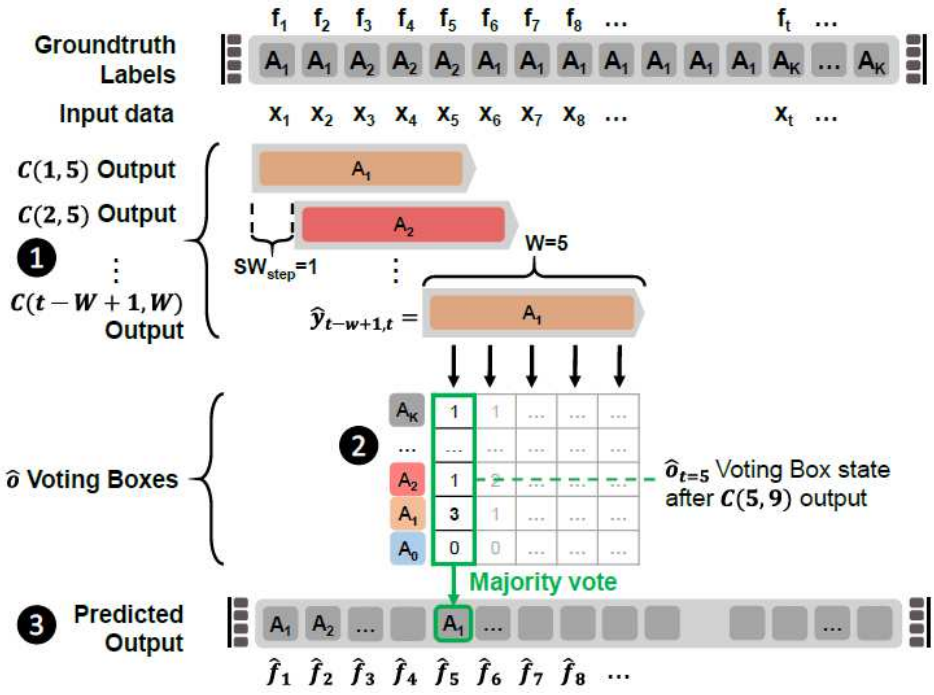


FIGURE 4.2 - Approche de la fenêtre glissante avec le principe du vote majoritaire

IV.2.2.2 STGCN-SWMV : Le vote majoritaire appliqué au ST-GCN avec fenêtre glissante

Dans la méthode proposée, nous utilisons cette approche de vote majoritaire avec fenêtre glissante (voir Figure 4.2). Tout d'abord, le même classifieur C de la fenêtre glissante (SW) donne une prédiction plusieurs fois pour la même image f_t. En effet, l'étiquette de sortie prédite $\hat{y}$ du classifieur de la SW dépend du placement des données d'entrée x_t à l'intérieur de la fenêtre glissante (voir Figure 4.2-1).

Par conséquent, nous pouvons considérer les fenêtres glissantes comme N_v différents classifieurs. Le classifieur de la SW peut être exprimé comme :

$$\begin{aligned} C(t,W) &= \hat{y}_{t,W} \\ &= [\hat{y}_t, \hat{y}_{t+1}, \dots, \hat{y}_{t+i}, \dots, \hat{y}_{t+w-1}] \\ &= C([x_t, x_{t+1}, \dots, x_{t+i}, \dots, x_{t+w-1}]) \end{aligned} \tag{4.7}$$

où $\hat{y}_{t,W}$ est un vecteur des classes prédites pour chaque donnée d'entrée de l'image t avec un SW de taille W. Lors de l'utilisation du module ST-GCN comme classifieur, les sorties sont les mêmes pour chaque image incluse dans la fenêtre glissante. Par exemple, dans la Figure 4.2-1, le label d'action de sortie de l'image $f_{t=5}$ n'est pas la même lors de l'utilisation des données de l'image f_{t-w+1} à f_t que lors de l'utilisation des données de l'image f_{t-w+2} à f_{t+1}. Par conséquent, $\hat{o}_t$ accumule la classe votée par chaque classifieur de $C(t - W + 1, W)$ à $C(t, W)$ (voir Figure 4.2-1 & 2)

Enfin, la classe d'action prédite réelle d'une image f_t est définie sur l'action qui a reçu le plus grand nombre de votes (voir Figure 4.2-3).

Les résultats sont extraits avec cette équation :

$$\hat{y}_t = mode\{\hat{o}_t\} \tag{4.8}$$

L'approche proposée de vote majoritaire permet de lisser les labels de sortie car les actions sont votées à partir de différents points de vue de la ligne de temps.

Le code et les modèles pré-entraînés de la méthode STGCN-SWMV sont rendus publics[11].

IV.3 Expérimentations

Dans cette section, nous évaluons les performances de reconnaissance d'actions en ligne de la méthode STGCN-SWMV proposée sur quatre jeux de données en ligne à base de squelettes : le jeu de données « Online Action Detection (OAD) » (Li, et al. 2016), le jeu de données « Online Action3D Dataset (UOW) » (Tang, et al. 2018) et les deux jeux de données que nous avons développés InHARD (Dallel, Havard et Baudry, et al. 2020) et InHARD-3-DT.

IV.3.1 Jeux de données d'évaluation

Pour évaluer notre approche, nous avons utilisé les jeux de données « Online Action Detection (OAD) » et « UOW Online Action3D » présentés dans les Sections I.4.2.7 et I.4.2.8 respectivement, le jeu de données InHARD présenté dans le Chapitre II ainsi qu'un nouveau jeu de données nommé InHARD-3DT que nous introduisons dans ce qui suit. D'autres jeux de données de HAR en ligne, tels que THUMOS (Idrees, et al. 2016) ou ActivityNet (Heilbron, et al. 2015), n'ont pas été pris en compte dans l'évaluation de notre méthode, car les classes d'action dans ces jeux de données sont soit des activités longues, comprenant des humains en vue rapprochée, soit n'incluent pas d'humains du tout. Par conséquent, l'extraction de squelettes humains à partir de ces données n'était pas possible pour évaluer l'approche proposée.

Pour rappel, le jeu de données OAD comprend 59 longues séquences où chaque sujet répète arbitrairement 10 actions du quotidien différentes ; boire, écrire, manger, essuyer, ouvrir le placard, se laver les mains, ouvrir le micro-ondes, balayez le sol, se gargariser, jeter les ordures et essuyer. Le jeu données UOW d'autre part se compose de 20 séquences d'actions réalisées par 20 participants distincts qui sont : Agiter le bras (haut), agiter le bras (horizontale), marteau, attraper les mains, coup de poing avant, jeter vers le haut, dessiner X, dessiner une coche, dessiner

[11] https://github.com/mejdidallel/STGCN-SWMV

un cercle, clap de main, agiter les deux mains, boxing, pencher, coup de pied avant, coup de pied latéral, jogging, se balancer une raquette de tennis, service de tennis, se balancer un club de golf, ramasser et lancer (voir Figure 1.33). Le jeu de données InHARD comprend 4808 séquences segmentés d'actions industrielles et 38 longues séquences et inclut 13 actions industrielles ; Assembler système, Consulter fiches, Picking en face, Picking à gauche, Poser composant, Poser toise, Poser visseuse, Poser sous-système, Prendre composant, Prendre toise, Prendre visseuse, Prendre sous-système et Tourner fiches.

IV.3.1.1 Matériels et méthodes

Nous avons été amené à développer un système permettant de simplifier grandement la récupération de données squelettes par rapport à celle présentée dans le chapitre précédent où les actions réalisées n'ont qu'un seul événement déclencheur qui caractérise le milieu de l'action (sauf pour les actions « Assembler système » et « Consulter les fiches » qui ont deux événements déclencheurs représentant le début et la fin de l'action). Le nouveau système, dont l'architecture est décrite dans la Figure 4.3, aboutira à un découpage plus précis des séquences d'actions en ayant 3 événements déclencheurs par action qui représentent respectivement l'état de début, en cours et de fin de chaque action ce qui permettra donc d'avoir des données d'apprentissage plus pertinentes et donc améliorer les performances de la HAR mais aussi permettra d'avoir un modèle DT plus précis et donc correspondant davantage au PT.

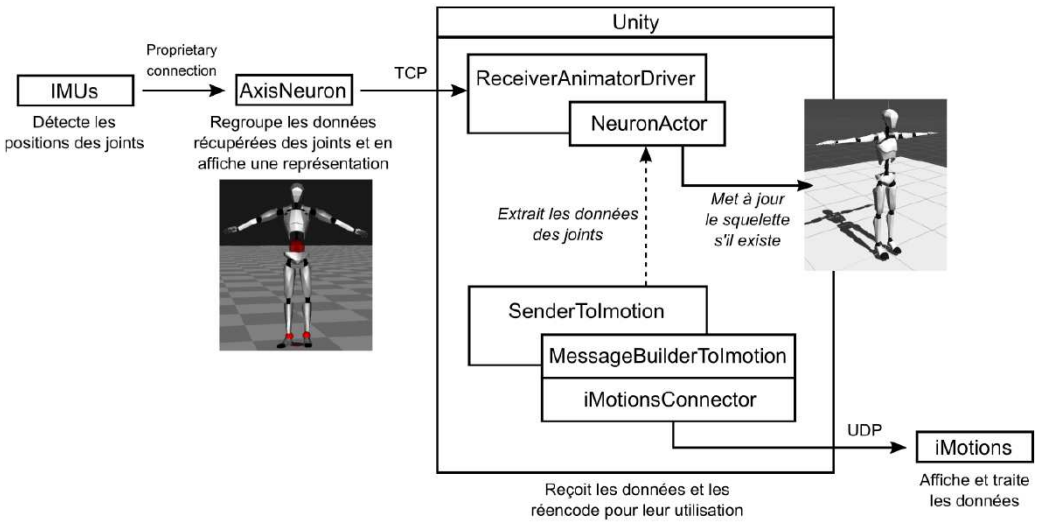


FIGURE 4.3 - Architecture du système de capture de données utilisé dans le jeu de données InHARD-3DT

Initialement, les données de la combinaison de capture de mouvement sont récupérées par un logiciel propriétaire se chargeant de les enregistrer sous forme de fichiers « .BVH ». En parallèle de cela, l'environnement de test que nous utilisons est un logiciel VR développé avec le moteur « Unity » et simulant un poste de travail avec lequel les personnes interagissent et réalisent donc les actions pour construire notre jeu de données. Ainsi, nous profitons de ce logiciel pour savoir quelles actions sont réalisées par l'utilisateur et à quels moments. En effet, notre simulation

VR se charge donc quant à elle de générer les labels d'actions.

Pour automatiser la labélisation des données, nous avons donc besoin de regrouper les données des capteurs depuis le logiciel « Axis Neuron » d'un côté, et les labels générés par la simulation VR avec « Unity » de l'autre. Pour ce faire, nous avons décidé de récupérer toutes ces données en temps réel au cours de l'enregistrement pour les envoyer vers le logiciel « iMotions », un logiciel permettant de regrouper et synchroniser des données depuis de nombreux capteurs. Celui-ci permet donc de synchroniser toutes ces données puis de générer des fichiers « .CSV » en résultant.

Afin de réaliser les connexions entre les différents logiciels en jeu, nous avons développé deux modules sous « Unity » qui peuvent ensuite être intégrés à n'importe quel projet. De cette manière, ces deux modules peuvent être utilisés pour créer différents jeux de données expérimentaux, simplement en les ajoutant au logiciel de simulation que nous voulons utiliser. Comme le présente la Figure 4.3, un premier module nommé « ReceiverAnimatorDriver » se charge de créer une connexion en TCP avec le logiciel « AxisNeuron » pour récupérer les données de la combinaison de capteurs. Si besoin, ce module peut également animer un squelette 3D sous « Unity » afin de visualiser les données reçues par « Unity » en temps réel. Le second module quant à lui, nommé « SenderToImotion », se charge de créer une connexion en UDP avec « iMotions » et d'envoyer les données, chaque joint étant envoyé comme un capteur à part entier pour « iMotions ». C'est également ce deuxième module qui est en charge d'envoyer les labels d'actions lorsqu'ils sont générés.

La Figure 4.4 présente une vue de la simulation RV utilisée pour la création du jeu de données InHARD-3-DT, ainsi que les zones d'interactions pour chacune des 3 classes. Notant que nous avons le même environnement que celui utilisé pour capturer le jeu de données InHARD-DT présenté dans le Chapitre III. Celle-ci reproduit l'établi utilisé pour la réalisation du jeu de données InHARD présenté dans le deuxième chapitre. Lorsque l'on se trouve dans la simulation, nous pouvons voir à notre gauche et face à nous les boîtes contenant les différentes pièces à utiliser pour les assemblages. En bas se trouve le plan de travail, en bas à droite l'outil perceuse et en bas à gauche une fenêtre permettant d'indiquer l'étape actuel en cas d'interaction avec le bras robot sur la droite. Enfin, le livret d'instruction se trouve face à nous en hauteur.

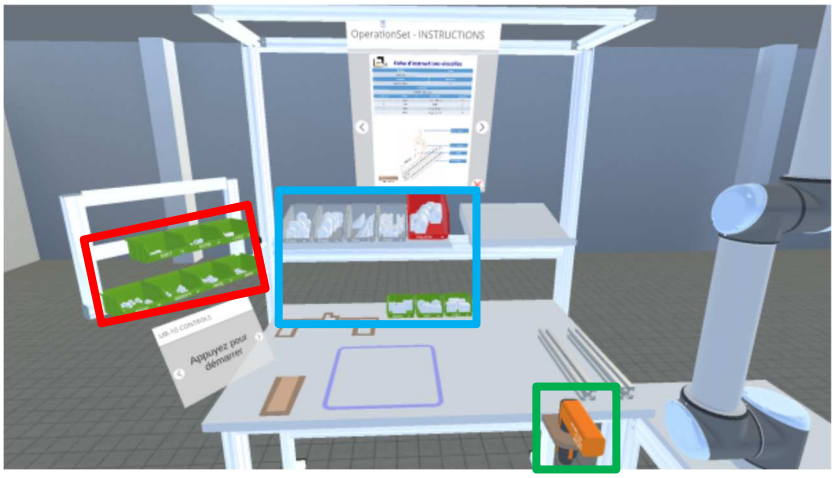


FIGURE 4.4 - Vue de la simulation RV pour la création des jeux de données InHARD-3-DT. Les zones d'interactions pour les 3 actions Picking à gauche, Picking en face et Prendre visseuse sont respectivement représentées par les 3 rectangles rouge, bleu et vert.

IV.3.1.2 InHARD-3-DT

InHARD et InHARD-DT sont deux jeux de données qui représentent des situations industrielles réelles et leurs acquisitions a été basé sur un cas d'utilisation réel. Les données de ces deux jeux de données ont été capturés dans des conditions non contrôlées où les opérateurs se comportent de manière naturelle pour effectuer les tâches d'assemblage sans aucune contrainte ce qui les rend des jeux de données complexes pour la HAR en ligne. Afin de surmonter ces complexités et pouvoir tester le bon fonctionnement des différents algorithmes de HAR en ligne avant d'observer leurs résultats sur des cas plus complexes, nous avons créé le jeu de données InHARD-3-DT. Ce dernier est basé sur une activité d'assemblage industriel. Nous avons tout d'abord utilisé ce jeu de données plus contrôlé, i.e. comportant moins de classes d'actions et avec des actions bien cadencées les unes après les autres, comme pour le jeu de données OAD et UOW.

Le jeu de données InHARD-3-DT présente un ensemble de 435 actions, réparties en 3 classes d'action différentes, réalisées sur une simulation en réalité virtuelle. Les 10 sujets ont réalisé chacun 10 à 15 fois les actions de chaque classe contrairement à InHARD et InHARD-DT où ils sont invités à réaliser l'assemblage complet à l'aide du cobot UR10. Dans InHARD-3DT, les actions sont présentées par séries de 3, à savoir une de chaque classe, tirées dans un ordre aléatoire. Les actions possibles sont : Picking à gauche, Picking en face et Prendre visseuse. Le choix de ces actions a été essentiellement motivé par la séparation spatiale des actions.

Le Tableau 4.1 montre un récapitulatif des caractéristiques des 3 jeux de données InHARD qui sont : InHARD, InHARD-DT et InHARD-3DT.

TABLEAU 4.1 - Récapitulatif des caractéristiques des jeux de données InHARD

Jeu de données	Année	Type(s) de données	Source	Classes d'Actions	Échantillons
InHARD	2019	Squelette 3D + RGB	Combinaison IMUs + Caméras RGB	13	4804
InHARD-DT	2021	Squelette 3D + RGB	Combinaison IMUs + Caméras RGB	13	4799
InHARD-3-DT	2021	Squelette 3D	Combinaison IMUs	3	435

Avant la réalisation de l'expérience, chaque participant est équipé des capteurs ainsi que du casque et des manettes VR. Ensuite, un temps de prise en main de l'outil VR lui est proposé, afin que le participant se familiarise avec la simulation et les contrôles des manettes. Ce temps de prise en main est fait par une série de 10 groupes de 3 actions. Les données récupérées sont uniquement les données squelettes issues des capteurs.

IV.3.2 Réglages des paramètres

Pour le jeu de données OAD, nous avons défini une taille de fenêtre glissante SW à 40 frames avec un FPS d'origine égal à 8. Pour le jeu de données UOW, nous avons défini la taille de la SW à 35 frames avec un FPS d'origine égal à 20. Le choix de ces paramètres a été basé sur la durée moyenne de toutes les actions donnant dans la majorité des cas les meilleurs résultats comme expliqué dans la Section IV.2.1. Pour ces deux jeux de données, et pour avoir une évaluation juste entre les différentes méthodes avec lesquelles nous nous comparons, nous avons utilisé les mêmes séquences d'apprentissage, de validation et de test définies dans les articles originaux (Li, et al. 2016) (Tang, et al. 2018) . Pour les jeux de données InHARD, nous avons utilisé une taille de fenêtre glissante SW à 70 frames avec un FPS égal à 30.

IV.3.3 Métriques d'évaluation

Pour évaluer les performances de notre méthode de reconnaissance des actions en ligne, nous avons utilisé les trois métriques d'évaluation F1-score, Intersection-over-Union (IoU) et Mean-over-Frames (MoF) que nous avons déjà défini dans le Chapitre I. De plus, nous avons utilisé la métrique Latence pour évaluer la sensibilité de notre méthode.

- Latence : compte tenu d'un intervalle temporel de durée H, la latence représente le délai normalisé pour prédire la classe de l'action en cours. En introduisant h comme étant la différence entre l'instant de prédiction et l'instant fourni par la vérité terrain, la latence de cette action est définie comme suit (Tang, et al. 2018) :

$$Latency = \frac{h}{H} \tag{4.9}$$

IV.3.4 Résultats & Performances de HAR

Dans cette section, nous évaluons la performance de la méthode de HAR en ligne proposée sur les jeux de données basés sur des squelettes présentés précédemment.

IV.3.4.1 Résultats sur le jeu de données OAD

Le Tableau 4.2 montre la précision et le F1-Score de toutes les actions sur le jeu de données OAD. Certains travaux ayant été évalués en utilisant uniquement la précision, d'autres en utilisant uniquement le F1-Score, nous avons mis non évalué (N/E) sur les métriques qui ne sont pas fournies.

TABLEAU 4.2 - Précision moyenne et F1-score obtenus sur le jeu de données OAD

Méthode	**Métriques d'évaluation**	
	F1-Score	**Accuracy**
JCR-RNN (Li, et al. 2016)	0.653	N/E
MD-RNN (C. Liu, Y. Li, et al. 2017)	N/E	0.628
RNN-SW (Zhu, et al. 2016)	0.600	N/E
MM-MT RNN (C. Liu, Y. Li, et al. 2017)	0.795	N/E
RF+ST (Baek, Kim et Kim 2017)	0.672	N/E
ST-LSTM (Liu, Shahroudy et Xu, et al. 2018)	N/E	0.770
FSNet (Liu, Shahroudy et Wang, et al. 2019)	N/E	0.800
SSNet (Liu, Shahroudy et Wang, et al. 2019)	N/E	0.820
MC-LSTM (Yin, et al. 2021)	0.848	N/E
SW-GCN (Delamare, et al. 2021)	0.900	0.900
STGCN-SWMV (Dallel, Havard et Dupuis, et al. 2022)	**0.953**	**0.954**

Le Tableau 4.3 détaille le F1-Score de chaque classe d'action et le F1-Score global de toutes les actions sur le jeu de données OAD. Comme nous pouvons le voir, notre méthode STGCN-SWMV surpasse les méthodes de l'état de l'art en utilisant les mêmes conditions de test. Cela montre l'efficacité du modèle que nous proposons. De plus, nous pouvons également voir que les scores de toutes les actions sont également distribués (de 94% à 100%) contrairement par exemple à (Li, et al. 2016), (Zhu, et al. 2016) et (Delamare, et al. 2021) là où la marge entre les scores des classes varie de 37% à 90%.

TABLEAU 4.3 - F1-Scores de toutes les actions obtenus sur le jeu de données OAD

Actions	JCR-RNN (Li, et al. 2016)	RNN-SW (Zhu, et al. 2016)	MM-MT RNN (C. Liu, Y. Li, et al. 2017)	RF+ST (Baek, Kim et Kim 2017)	SW-GCN (Delamare, et al. 2021)	**STGCN-SWMV** (Dallel, Havard et Dupuis, et al. 2022)
Boire	0.574	0.441	0.538	0.517	0.090	**0.979**
Manger	0.523	0.550	0.658	0.645	0.840	**1.000**

Actions	JCR-RNN (Li, et al. 2016)	RNN-SW (Zhu, et al. 2016)	MM-MT RNN (C. Liu, Y. Li, et al. 2017)	RF+ST (Baek, Kim et Kim 2017)	SW-GCN (Delamare, et al. 2021)	**STGCN-SWMV** (Dallel, Havard et Dupuis, et al. 2022)
Écrire	0.822	0.859	0.892	0.803	0.920	**0.993**
Ouvrir le placard	0.495	0.321	0.643	0.555	0.890	**1.000**
Se laver les mains	0.718	0.668	0.800	0.860	0.780	**1.000**
Ouvrir le micro-ondes	0.703	0.665	0.780	0.610	0.780	**0.963**
Balayez le sol	0.643	0.590	0.882	0.437	0.930	**0.966**
Se gargariser	0.623	0.550	0.658	0.722	0.950	**0.984**
Jeter les ordures	0.459	0.674	0.748	0.688	0.880	**0.949**
Essuyer	0.780	0.747	0.943	**0.977**	0.960	0.973
Total	0.653	0.600	0.795	0.672	0.900	**0.954**

Pour mieux appréhender les résultats, la Figure 4.5 montre la matrice de confusion des F1-Scores de toutes les actions sur le jeu de données OAD. Quelques observations intéressantes peuvent être faites. Tout d'abord, la classe « No action » couvre plus de 60% du jeu de données et est surreprésentée par rapport aux autres actions. Toutefois, notre méthode a obtenu de bonnes performances sur toutes les autres actions. Cela peut expliquer pourquoi toutes les autres actions sont confondues avec la classe « No Action ».

Par ailleurs, quelques actions peuvent également avoir des poses similaires et peuvent être facilement mélangées lors de l'exécution de la tâche de reconnaissance. Cela n'empêche que notre méthode a montré sa robustesse et surpasse les autres méthodes dans presque toutes les actions avec un score de reconnaissance minimum de 94.9 %.

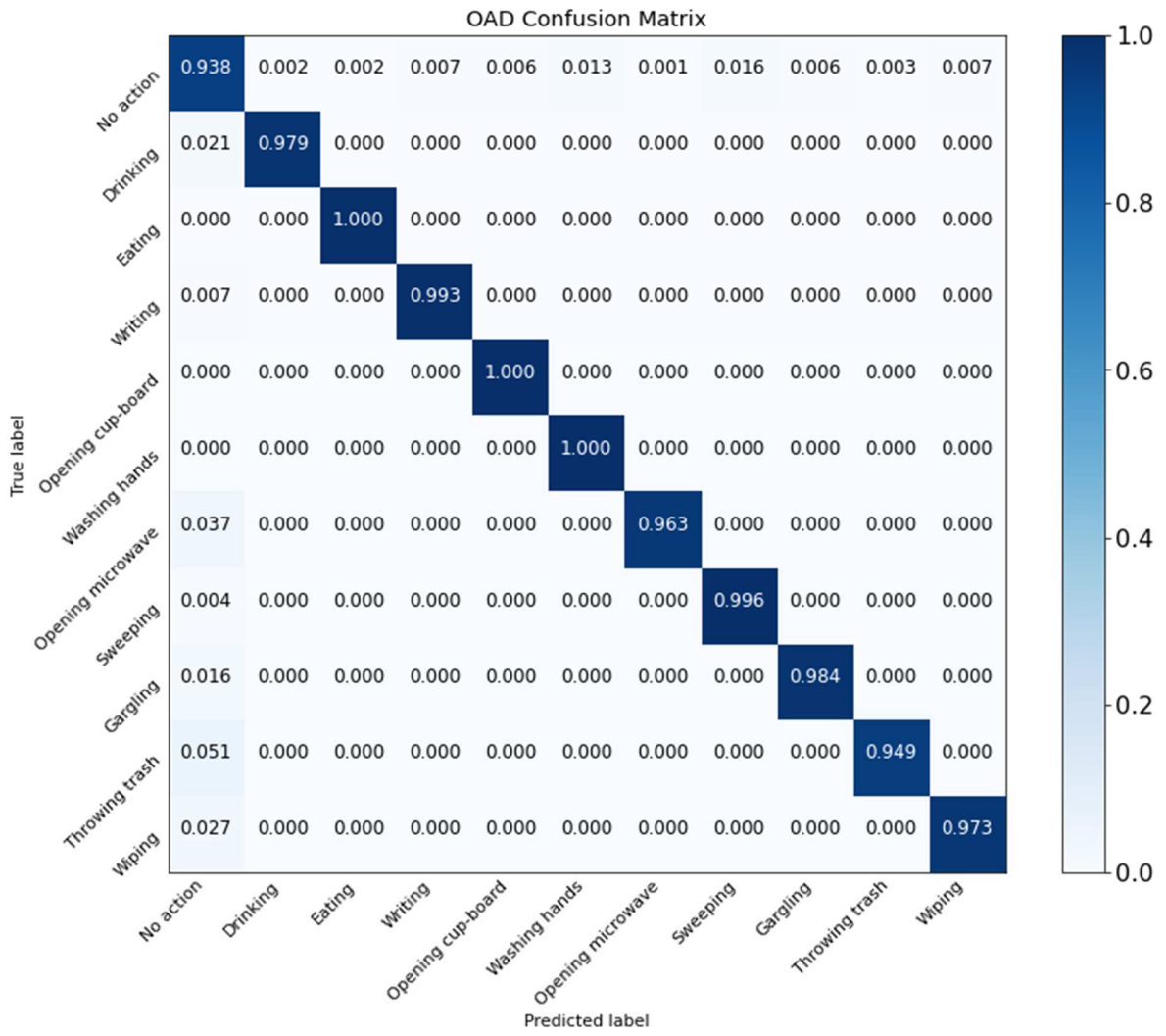


FIGURE 4.5 - Matrice de confusion du jeu de données OAD avec la méthode STGCN-SWMV

Cependant, l'efficacité de la méthode proposée implique une latence de W images, puisqu'une image doit être votée par chaque classifieur SW avant de prendre la décision.

Étant donné que la plupart des actions impliquent une interaction entre les humains et les objets (par exemple ouvrir le micro-ondes, ouvrir le placard etc.), l'incorporation de telles informations contextuelles via une reconnaissance d'objets, tels que les pièces ou les outils utilisés, dans la phase d'apprentissage devrait aider à réduire l'ambiguïté et le mélange entre les actions qui peuvent avoir des poses similaires. La Figure 4.6 montre la timeline comparant les labels de vérité terrain et les labels prédits en fonction du temps sur le jeu de données OAD. Comme nous pouvons le voir, les labels de prédiction correspondent à celles de la vérité terrain dans presque tous les frames, y compris la durée ainsi que le temps de début et de fin de chaque action, ce qui démontre l'efficacité de notre méthode et son adéquation à la reconnaissance d'actions en ligne.

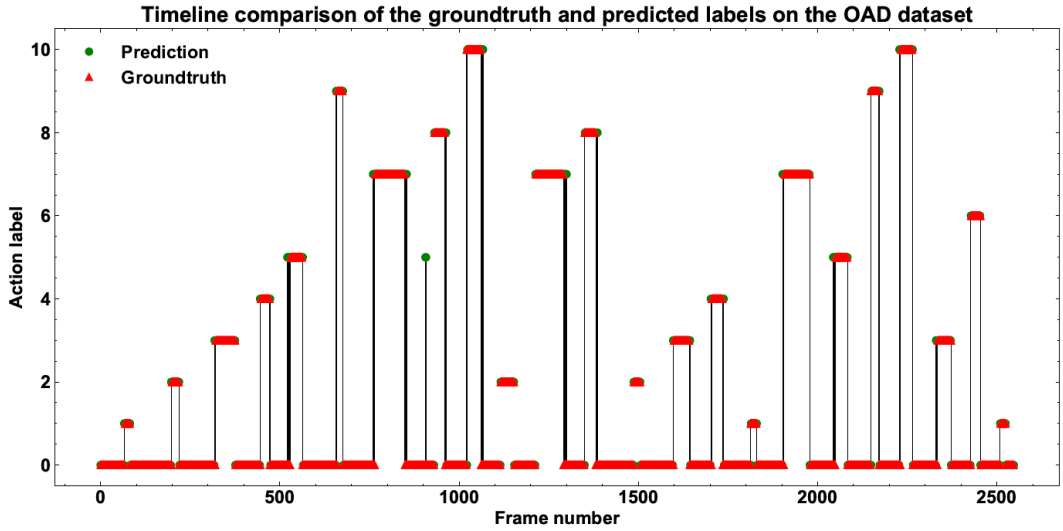


FIGURE 4.6 - Timeline de la vérité terrain et des labels prédits pour le jeu de données OAD avec la méthode STGCN-SWMV

IV.3.4.2 Résultats sur le jeu de données UOW

Le Tableau 4.4 montre la précision et le F1-Score de toutes les actions sur le jeu de données UOW. Nous avons mis non évalué (N/E) sur les métriques qui ne sont pas fournies dans les autres études.

TABLEAU 4.4 - Précision moyenne et F1-Score obtenus sur le jeu de données UOW

Méthode	**Métriques d'évaluation**		
	F1-Score	**Accuracy**	**Latence**
LE-KSVM (Tang, et al. 2018)	N/E	N/E	0.306
NN (Tang, et al. 2018)	N/E	N/E	0.316
CovaAct (Kviatkovsky, Rivlin et Shimshoni 2014)	N/E	N/E	0.382
Cov3DJ (Tang, et al. 2018)	N/E	N/E	0.340
SW-CCN (Delamare, et al. 2021)	0.680	0.680	N/E
SW-GCN (Delamare, et al. 2021)	0.755	0.750	N/E
STGCN-SWMV (Dallel, Havard et Dupuis, et al. 2022)	**0.934**	**0.936**	**0.047**

La Figure 4.7 montre la matrice de confusion des F1-Scores sur le jeu de données UOW. Notre méthode a réussi des performances élevées par rapport aux méthodes de l'état de l'art ce qui qui confirme sa robustesse et son adaptation pour la reconnaissance d'actions en ligne.

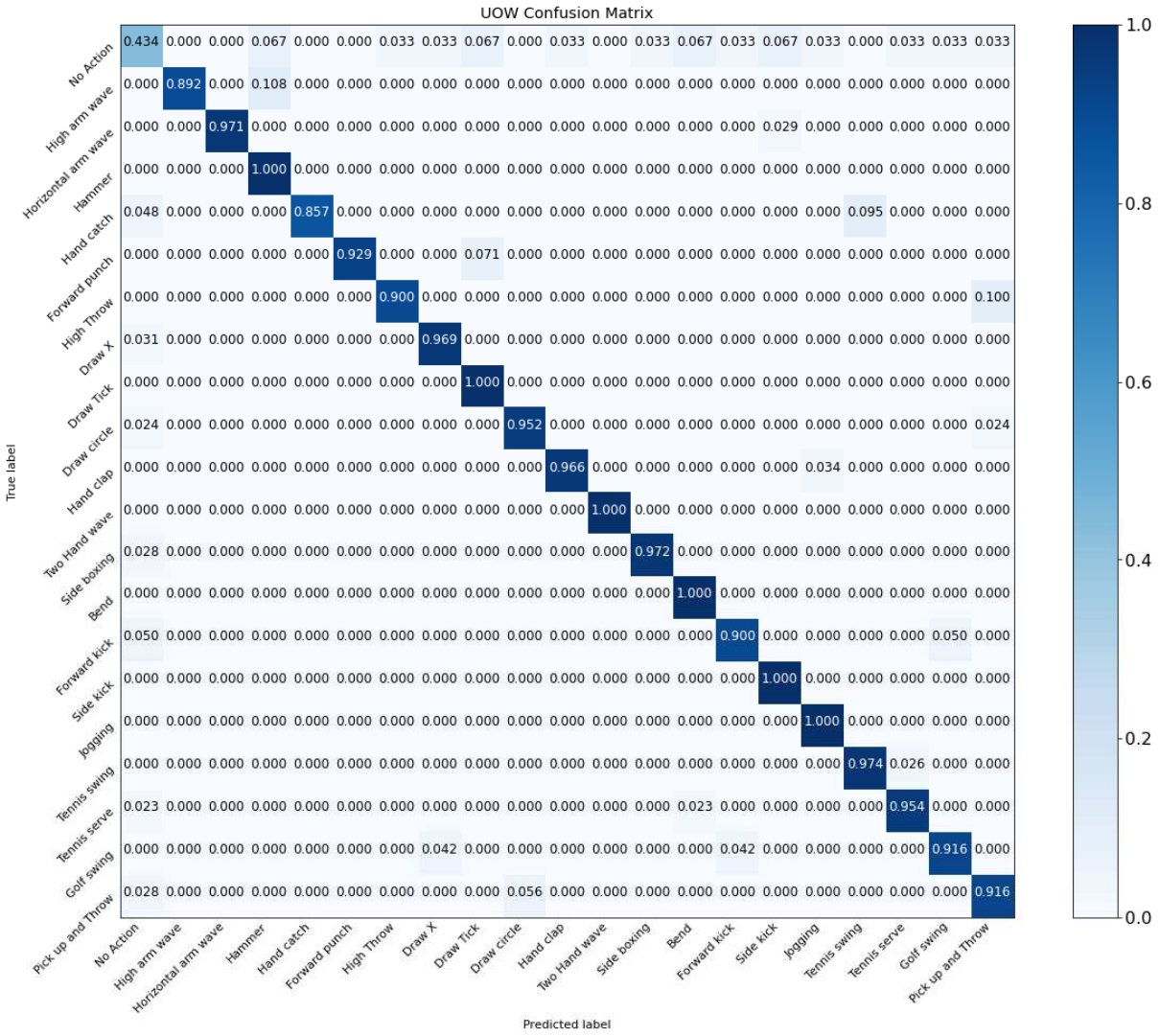


FIGURE 4.7 - Matrice de confusion du jeu de données UOW avec la méthode STGCN-SWMV

En ce qui concerne le jeu de données UOW, la classe « No action » n'est pas bien reconnue (voir la Figure 4.7), ce qui peut s'expliquer par le fait qu'il y a très peu d'échantillons de cette classe et qu'elle présente une grande variabilité, ce qui rend difficile pour notre algorithme de la différencier des autres actions. Par ailleurs, toutes les autres actions sont très bien reconnues et notre méthode a donné de très bons résultats pour chacune d'entre elles.

La Figure 4.8 montre la timeline comparant les labels de vérité terrain et les labels prédits en fonction du temps sur le jeu de données UOW. Comme nous pouvons le voir, les étiquettes de prédiction coïncident majoritairement avec celles de la vérité terrain pour presque toutes les frames. Les différentes actions prédites correspondent aussi à celles de la vérité terrain, en durée des actions ainsi qu'en temps de début et de fin. Cela montre encore l'efficacité de notre méthode et son adéquation à la reconnaissance d'actions en ligne.

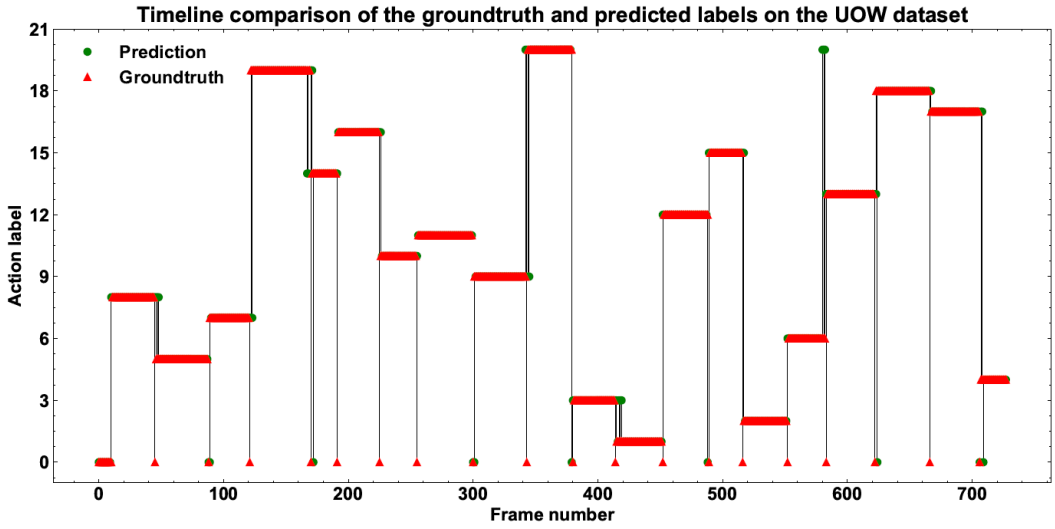


FIGURE 4.8 - Timeline de la vérité terrain et des labels prédits pour le jeu de données UOW avec la méthode STGCN-SWMV

Nous avons également évalué notre méthode à l'aide de la métrique Intersection-over-Union (IoU), mais comme tous les autres travaux avec lesquels nous comparons ne l'utilisaient pas, nous indiquons simplement les résultats. Sur le jeu de données OAD, nous avons atteint une IoU de 0,977 et de 0,958 sur le jeu de données UOW avec un seuil de rapport de chevauchement égal à 0,8. Nous pouvons voir que la méthode proposée obtient les meilleurs résultats sur tous les jeux de données avec toutes les métriques d'évaluation en comparaison avec les méthodes de l'état de l'art.

IV.3.4.3 Résultats sur le jeu de données InHARD

Le Tableau 4.5 montre la précision Top-k[12] moyenne et F1-Scores moyens de toutes les actions (13 actions d'InHARD + No action) obtenus sur le jeu de données InHARD en utilisant la méthode STGCN-SWMV.

TABLEAU 4.5 - Accuracy Top-k et F1-Score moyens obtenus sur le jeu de données InHARD

Méthode	**Métriques d'évaluation**		
	Accuracy		**F1-Score**
	Top 1	**Top 5**	
STGCN-SWMV (Dallel, Havard et Dupuis, et al. 2022)	**0.344**	**0.625**	**0.433**

Nous remarquons que l'algorithme STGCN-SWMV a montré des difficultés pour reconnaitre les différentes classes d'actions dans le jeu de données InHARD en ligne avec une précision

[12] Top-k : Cette métrique calcule le nombre de fois où le label correct figure parmi les k meilleures labels prédits (classées par les scores prédits) (Carreira et Zisserman 2017).

moyenne de 0.344 en Top-1, 0.625 en Top-5 et un F1-Score moyen de 0.433 et la prédiction de la classe No action seulement.

InHARD est un jeu de données qui a été capturé en se basant sur un cas d'utilisation réel, dans des conditions non contrôlées où les opérateurs se comportent de manière naturelle pour effectuer les actions d'assemblage ce qui n'est pas le cas pour la majorité des jeux de données existants où les opérateurs sont limités à réaliser les actions d'une manière bien spécifique voire contrainte, par exemple en utilisant la même main pour certaines actions ou bien en se positionnant exactement dans la même position pour effectuer telle action ou encore de faire des pauses entre chaque action réalisée etc.. De plus, comme le montre la Figure 4.9, l'action « No action » dans le jeu de données InHARD couvre plus de 60% du jeu de données et est très variable. Par exemple, le label « No action » peut représenter une personne se grattant les cheveux ou une personne claquant les doigts pendant la manipulation. C'est pour cela que nous avons décidé d'enlever cette action dans nos expérimentations suivantes. En outre, l'action « Assembler système » inclut plusieurs actions impliquant un assemblage et qui ne sont pas toutes réalisées de la même manière ce qui augmente aussi la variabilité intra-classe. Également, des actions comme « Poser sous-système », « Prendre sous-système », « Poser composant » et « Prendre composant » sont très peu représentées dans InHARD puisqu'elles sont réalisées trop peu de fois durant la situation d'assemblage proposée. Ce déséquilibre entre les classes d'actions rend le jeu de données InHARD compliqué dans des scénarios en ligne et donc la reconnaissance de ces actions s'avère une tâche complexe. Cela nous semble fortement contribuer à la baisse des performances de reconnaissance pour ce jeu de données.

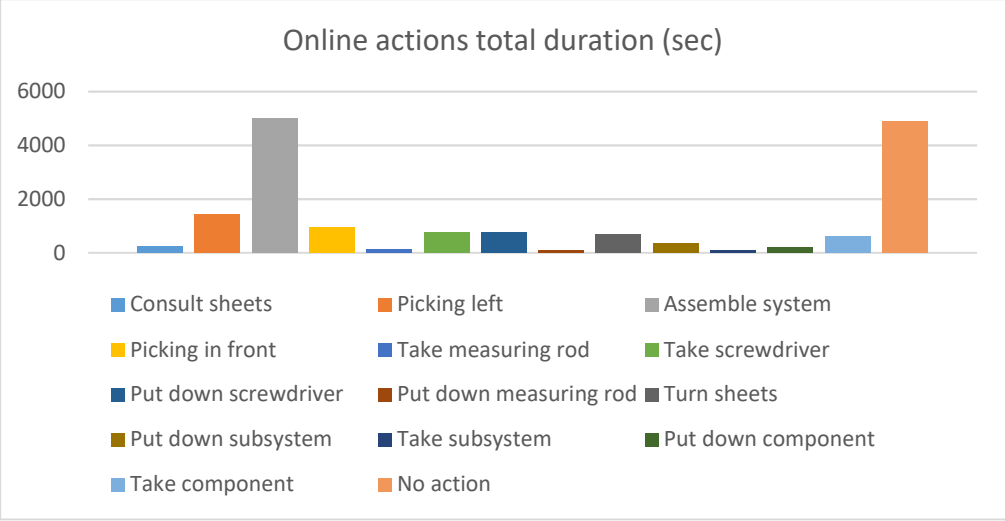


FIGURE 4.9 - Durée totale des actions en ligne dans le jeu de données InHARD

La Figure 4.10 montre la timeline de la vérité terrain et des labels prédits pour le jeu de données InHARD avec la méthode STGCN-SWMV en retirant la classe « No action ». Comme nous pouvons le voir, les étiquettes de prédiction coïncident dans plusieurs occasions avec celles

de la vérité terrain. Aussi, les différentes actions prédites correspondent plus ou moins à celles de la vérité terrain, en durée des actions ainsi qu'en temps de début et de fin. Cela montre encore que la classe « No action », couvrant plus de 60% du jeu de données et étant très variable, rend l'algorithme STGCN-SWMV incapable de se généraliser sur les autres actions. Nous remarquons qu'en retirant la classe « No action », notre algorithme arrive à bien distinguer le reste des différentes classes d'action du jeu de données InHARD dans la HAR en ligne.

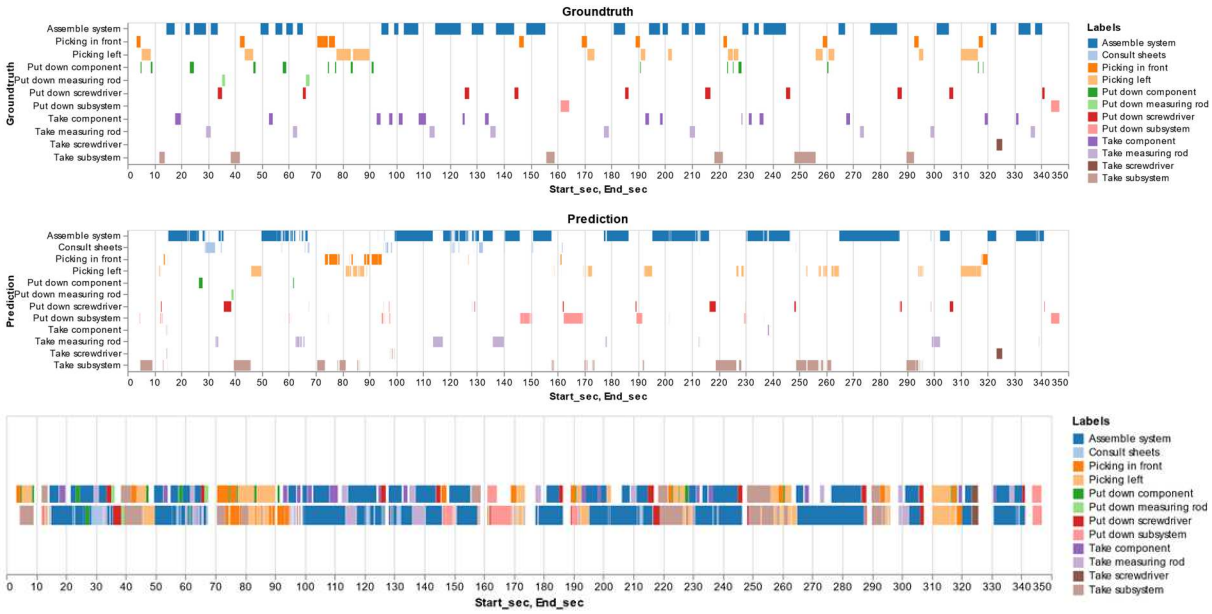


FIGURE 4.10 - Timeline de la vérité terrain et des labels prédits pour le jeu de données InHARD avec la méthode STGCN-SWMV

Pour affiner les résultats en ligne du jeu de données InHARD, nous avons développé un système de débruitage des prédictions. Ce système prend en entrée les prédictions générées par l'algorithme STGCN-SWMV ensuite prend la majorité des votes de labélisation sur une fenêtre glissante. Cela permet d'éliminer les prédictions parasites au sein des fenêtres glissantes en éliminant le bruit et affiner ainsi le résultat final. La Figure 4.11 explique l'algorithme de débruitage proposé.

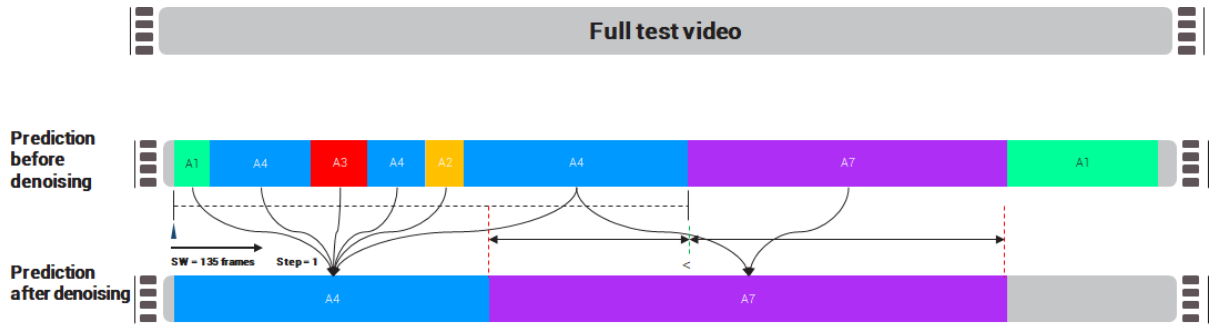


FIGURE 4.11 - Algorithme de dé bruitage des prédictions

La Figure 4.12 montre la timeline de la vérité terrain et des labels prédits pour le jeu de données InHARD avec la méthode STGCN-SWMV après débruitage.

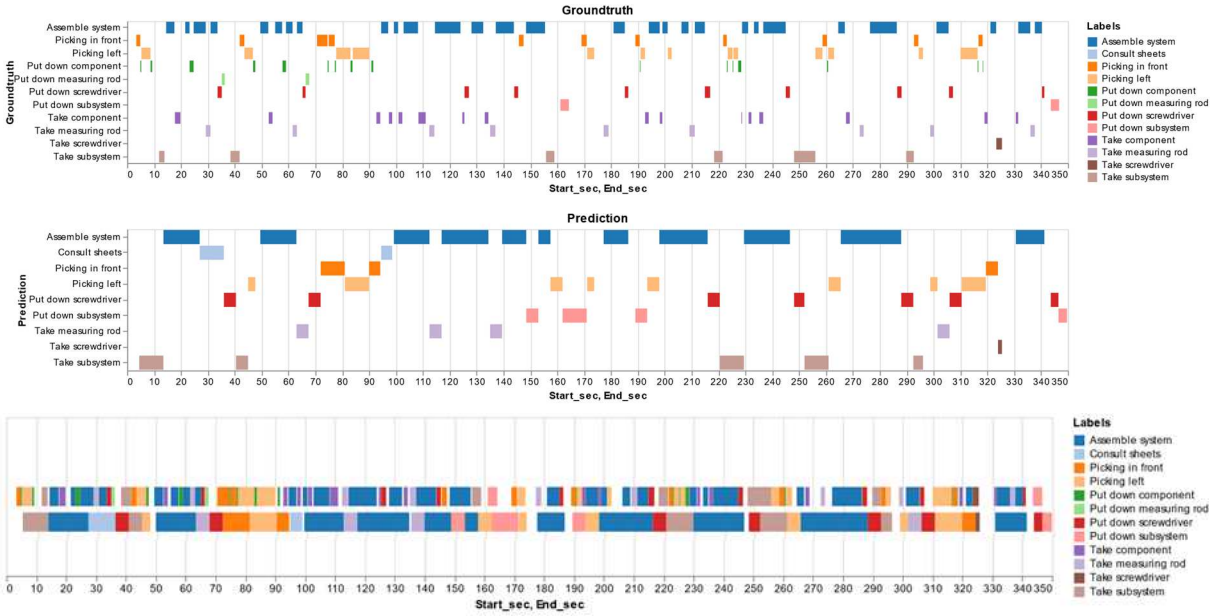


FIGURE 4.12 -Timeline de la vérité terrain et des labels prédits pour le jeu de données InHARD avec la méthode STGCN-SWMV après débruitage

Le Tableau 4.6 montre les résultats obtenus sur le jeu de données InHARD avant et après avoir appliqué l'algorithme de débruitage incluant la précision et la Mean-over-Frames (MoF) moyennes définies dans la Section I.3.3.

TABLEAU 4.6 - Accuracy moyenne et MoF obtenus sur le jeu de données InHARD avant débruitage

Méthode	**Avec/sans débruitage**	**Métriques d'évaluation**		
		Accuracy		**MoF**
		Top-1	**Top-5**	
STGCN-SWMV (Dallel, Havard et Dupuis, et al. 2022)	Sans	0.327	0.603	0.278
	Avec	0.366	0.631	0.359

Les faibles performances de HAR en ligne sur le jeu de données InHARD avec la méthode STGCN-SWMV confirme que la problématique est essentiellement dû à la complexité du jeu de données qui implique des situations industrielles réelles complexes comme nous l'avons évoqué précédemment. Pour mieux appréhender l'impact de contrôle de l'acquisition des données sur les performances de HAR en ligne, nous avons créé un nouveau jeu de données nommé InHARD-3DT, qui est plus contrôlé et qui comporte moins de classes d'actions qui sont bien cadencées les unes après les autres, comme pour le jeu de données OAD et UOW.

IV.3.4.4 Résultats sur le jeu de données InHARD-3-DT

Le Tableau 4.7 montre la précision et le F1-Score de toutes les actions sur le jeu de données InHARD-3-DT.

TABLEAU 4.7 - Accuracy moyenne et F1-Score obtenus sur le jeu de données InHARD-3-DT

Méthode	Jeux de données	Métriques d'évaluation	
		Accuracy	F1-Score
STGCN-SWMV (Dallel, Havard et Dupuis, et al. 2022)	InHARD-3-DT	**0.975**	**0.975**

La Figure 4.13 montre la matrice de confusion du jeu de données InHARD-3DT en ligne avec la méthode STGCN-SWMV.

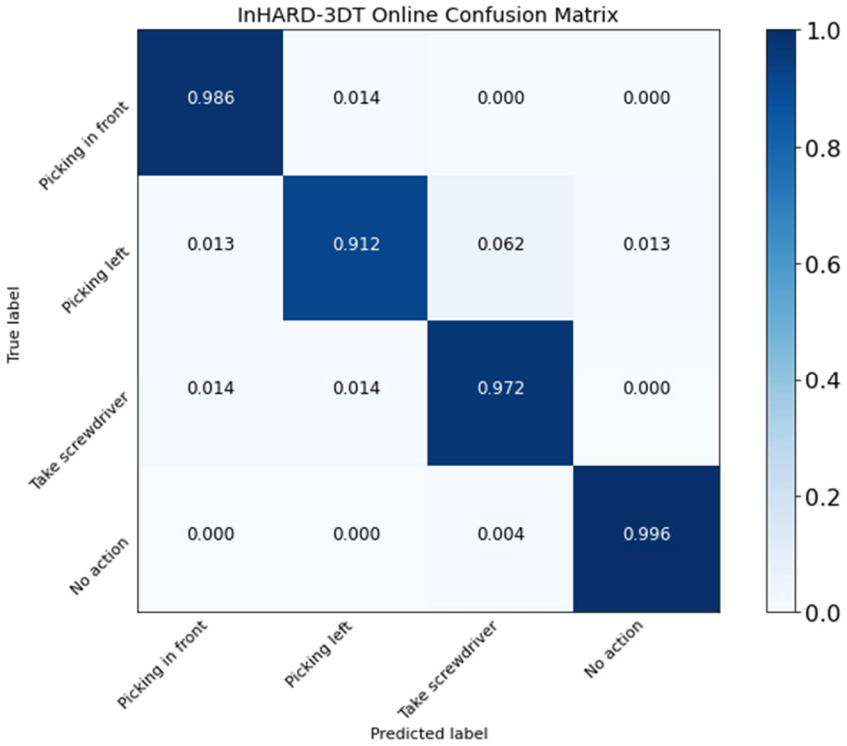


FIGURE 4.13 - Matrice de confusion du jeu de données InHARD-3-DT avec la méthode STGCN-SWMV

Nous remarquons que le fait d'avoir un jeu de données équilibré est très important (c'est-à-dire le nombre de samples par action est similaire) et qui est réalisé dans des conditions plus ou moins contrôlés permet de booster les performances de reconnaissance d'une manière significative.

La Figure 4.14 montre une illustration de la reconnaissance d'actions en ligne avec le jeu de données InHARD-3DT.

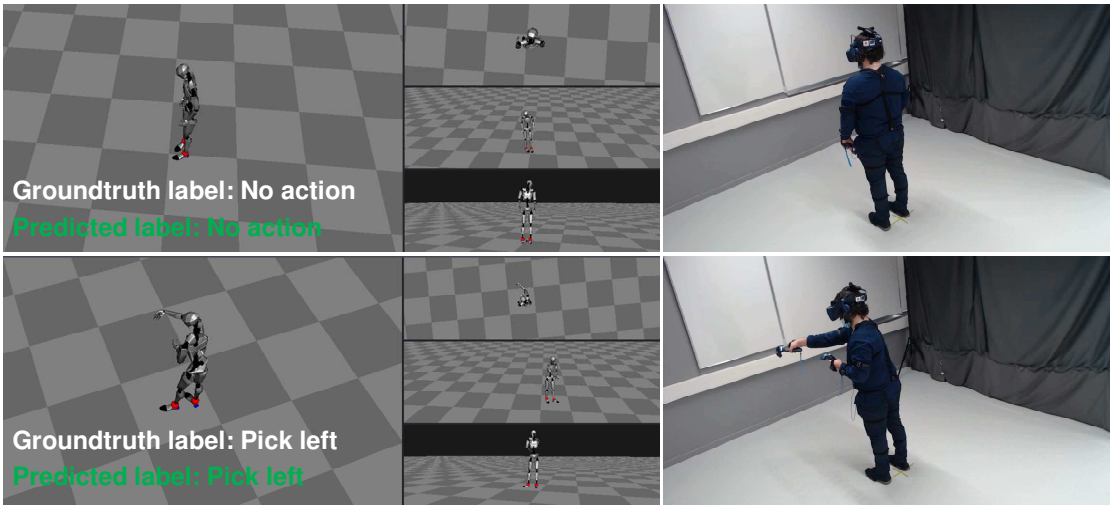


FIGURE 4.14 - Reconnaissance d'actions en ligne avec le jeu de données InHARD-3DT

Dans nos expérimentations avec le jeu de données InHARD-3DT, nous avons gardé la classe « No action », au cours de laquelle les opérateurs sont invités de bouger le moins possible entre deux actions consécutives et nous remarquons que l'algorithme STGCN-SWMV arrive à bien distinguer les différentes classes d'actions dans la tâche de reconnaissance avec une précision moyenne de 0.975 et un F1-Score moyen de 0.975.

IV.4 Conclusion

Dans ce chapitre, nous avons présenté un algorithme de reconnaissance d'actions en ligne basée sur des données squelettes en utilisant un réseau de neurones convolutif à graphe spatio-temporel avec une approche de fenêtre glissante et de vote majoritaire nommée STGCN-SWMV. L'utilisation des ST-GCNs nous a permis de tirer parti des informations de mouvement capturées dans les séquences squelettiques dynamiques, car elles lissent le bruit autour des articulations. En effet, ces dernières se déplacent en groupes locaux ce qui facilite leur modélisation en agrégeant leurs propres caractéristiques et les caractéristiques de leurs voisins dans le temps. Nous avons montré que l'utilisation d'une fenêtre glissante et d'une approche de vote majoritaire avec les ST-GCNs permettait non seulement un plus grand pouvoir prédictif, mais également une plus grande capacité de généralisation surmontant les limitations des méthodes précédentes en apprenant automatiquement à la fois des informations spatiales et temporelles à partir des données, ce qui rend cette méthode plus appropriée que d'autres pour la reconnaissance d'action en ligne avec des flux de données continus. Bien que les résultats soient bons et démontrent l'efficacité de notre méthode en surpassant les méthodes de l'état de l'art sur deux jeux de données en ligne à base de squelettes nommés OAD, et UOW où nous avons obtenu un F1-score de 0.953 et une précision moyenne de 0.954 sur le jeu de données OAD et un F1-score de 0.934, une précision moyenne de 0.936 et une Latence de 0.047 sur le jeu de donnée UOW, les expériences sur le jeu de données InHARD ont montré les limites de l'algorithme ST-GCN-SWMV dans des

scénarios en ligne avec un jeu de données complexe. En effet, nous avons obtenu une Accuracy moyenne de 0.344 en Top-1 et de 0.625 en Top-5 et un F1-Score moyen de 0.433 en ne prédisant que la classe « No action ». Ces expérimentations menées avec le jeu de données InHARD ont montré sa complexité pour des scénarios en ligne avec une grande variabilité inter et intra-classes en plus du déséquilibre entre les classes. Pour confirmer l'impact de la complexité du jeu de données liée à une activité industrielle de collaboration humain robot réelle nous avons proposé un nouveau jeu de données nommé InHARD-3DT plus contrôlé et comportant moins de classes d'actions et avec des actions bien cadencées les unes après les autres. Nous avons réussi une précision moyenne de 0.975 et un F1-Score moyen de 0.975 avec le jeu de données InHARD-3DT.

Les algorithmes de HAR actuels fonctionnent bien quand les jeux de données utilisées pour l'apprentissage sont contrôlés ce qui joue un rôle très important dans la phase de reconnaissance et permet de booster les performances de reconnaissance en ligne. Néanmoins, il reste encore des défis pour permettre une utilisation en industrie avec de la variabilité.

Il existe de nombreuses perspectives pour ce projet et la recherche peut prendre plusieurs directions, notamment la création d'un nouveau jeu de données contrôlé avec plus d'actions peut se faire ainsi que l'étude des méthodes d'augmentation des données squelettiques du jeu de données InHARD qui peut également améliorer davantage les performances de HAR. Une nouvelle collecte de données avec plus d'actions peut également se faire en environnement réel à condition que le protocole expérimental préserve la complexité et l'équilibre du jeu de données. Également l'étude des méthodes d'augmentation des données pour les données squelettiques et la tentative de les appliquer au jeu de données InHARD (Kwon et Lee 2020).

Conclusion générale et perspectives

A travers cette thèse, nous avons abordé la problématique de reconnaissance d'actions humaines dans un milieu industriel en temps réel sur un flux de données continu. Cette reconnaissance d'actions est assurée grâce aux méthodes d'apprentissage profond (Deep Learning). Ces techniques nécessitent d'avoir de très grands jeux de données labélisées qui requièrent une intervention humaine pour les produire. Nous avons donc abordé cette problématique en orientant nos travaux sur la génération automatique de jeux de données étiquetées pour les algorithmes d'apprentissages profonds pour la reconnaissance d'actions humaines dans un contexte industriel.

Au sein du premier chapitre, nous avons tout d'abord présenté un état de l'art sur l'industrie 4.0, la collaboration humain-robot (HRC) et le jumeau numérique. Puis, l'état de l'art s'est concentré sur la reconnaissance d'action humaines (HAR), en commençant par définir les deux types de reconnaissance d'actions humaines : segmentée et en ligne. Ensuite, les modalités de données associées et les capteurs utilisés pour la collecte de données ont été détaillés. Par la suite nous avons présenté les jeux de données de HAR les plus récents et les plus utilisés dans la littérature. L'état de l'art s'est terminé par la présentation des méthodes de reconnaissance d'actions humaines en fonction des modalités utilisées ainsi que leurs applications dans un contexte industriel.

Dans le deuxième chapitre, nous nous sommes intéressés à la création d'un jeu de données spécifique au contexte industriel. En effet, l'étude faite précédemment a démontré le manque de jeux de données de HAR dans un contexte industriel ; les jeux de données de HAR existants comprennent simplement des actions de la vie quotidienne ou bien des actions liées à la santé limitant ainsi l'usage de la HAR dans l'industrie. Nous avons donc proposé le jeu de données d'actions humaines dans un contexte industriel nommé « Industrial Human Action Recognition Dataset (InHARD) ». Ce jeu de données est basé sur l'assemblage d'un produit manufacturé sur des stations équipées de cobots UR10 travaillant en collaboration avec les opérateurs. Il comporte plus de 2 millions d'images et contient 13 classes d'actions industrielles différentes réalisées par 16 personnes distinctes et plus de 4800 échantillons d'actions. Ce jeu de données pourra contribuer à aider la communauté de la recherche à progresser dans la HAR dans les environnements industriels et à faciliter la collaboration humain-robot. L'état de l'art mené sur les algorithmes d'apprentissage profond nous a montré que le choix le plus pertinent pour la reconnaissance d'actions humaines serait d'utiliser des données squelettes et de surcroît utiliser des algorithmes de réseau à graphe convolutif. Des expérimentations avec l'algorithme ST-GCN ont étés menées sur InHARD et ont permis d'obtenir de bonnes performances en HAR avec

une Accuracy de 92.60% et un F1-score de 92.76% sur les données segmentées.

Dans le troisième chapitre, nous avons proposé une méthodologie couplant les jumeaux numériques (DT) et la Réalité Virtuelle (RV) pour produire un modèle numérique des humains et permettre ainsi la génération automatique de données labélisées. Nous avons ainsi construit un nouveau jeu de données nommé « Industrial Human Action Recognition Dataset - Digital Twins (InHARD-DT) », généré grâce à de la labélisation automatique. En effet, l'outil de labélisation automatique est intégré au jumeau numérique associé à un Environnement Virtuel permettant de former l'opérateur, équipé de capteurs, à travailler en collaboration avec le cobot UR10. Cette mise en situation permet, par conséquent, de générer de la donnée labélisée automatiquement pour l'algorithme de reconnaissance d'actions humaines. La robustesse et la généralisation de notre méthode ont été évaluées à travers l'algorithme STGCN en l'entrainant avec les données du jumeau numérique et en validant sur des données du jumeau physique. Les résultats obtenus ont montré une généralisation atteignant les 89% de précision et de F1-score, prouvant la pertinence de l'approche proposée. Nous avons montré que l'introduction du jeu de données InHARD-DT a révélé que l'entrainement des algorithmes de HAR pouvait bénéficier de l'apport des outils de réalité virtuelle permettant de simuler les interactions humains robots afin de surmonter les problèmes liés à la labélisation et au manque de données rencontrés lors de la création du jeu de données InHARD.

Dans un quatrième chapitre, nous avons approfondi nos études sur les algorithmes d'apprentissage profond basés sur des données squelettes ce qui nous a permis de proposer une nouvelle méthode utilisant les réseaux de neurones convolutifs à graphes spatio-temporel avec une fenêtre glissante et un vote majoritaire nommé « Spatial-Temporel Graph Convolutional Neural Network with a Sliding Window and Majority Voting - STGCN-SWMV ». La technique de fenêtre glissante est une bonne approche pour la reconnaissance d'actions en ligne avec des flux continus. Notre algorithme exploite la corrélation entre chaque articulation squelettique du corps humain en construisant un graphe spatio-temporel utilisé par la suite pour la HAR. Notre approche permet une détection en temps réel d'actions humaines sur des données en flux continu. Nous avons montré l'efficacité de la méthode présentée qui, en comparaison avec les méthodes de HAR de l'état de l'art, a obtenu de meilleures performances de classification sur deux jeux de données en ligne basées sur des données squelettes nommés OAD et UOW. Des expériences menées sur le jeu de données InHARD ont montré ces limites. Ce dernier étant complexe pour la HAR en ligne vu qu'il représente des situations industrielles réelles où l'acquisition des données a été réalisée dans des conditions non contrôlées. En effet, les opérateurs se comportent de manière naturelle pour effectuer les tâches d'assemblage, menant à des variabilités inter et intra-classes importantes qui influencent les performances de HAR. Fort de ce constat, un jeu de données InHARD-3DT, plus contrôlé, a été créé. Sur ce jeu de données, les résultats montrent l'impact de la complexité du jeu de données sur les performances de HAR où nous avons obtenu avec une précision moyenne de 0.975 et un F1-Score moyen de 0.975.

Les travaux de cette thèse ouvrent différentes possibilités et applications. Ainsi, les perspectives pour la recherche peuvent prendre plusieurs directions pour améliorer la collaboration humain-robot au sein des milieux industriels. Pour le jeu de données InHARD, une nouvelle collecte de données avec plus d'actions peut se faire en environnement réel en gardant à l'esprit d'avoir un protocole expérimental qui préserve la complexité et l'équilibre des données. Des techniques d'augmentation des données squelettiques pourraient également être déployées pour générer plus de données pour la tâche de HAR.

Pour le jeu de données InHARD-DT, passer à la labélisation floue des actions et explorer des mélanges d'approches (modèles hybrides) pourraient être bénéfique pour la HAR. Cela peut permettre de prendre en compte les approximations induites par la labélisation unique des actions en ayant trois événements déclencheurs par action représentant respectivement l'état de début, en cours et de fin de chaque action. Cela aboutirait à un découpage beaucoup plus précis des séquences d'actions et devrait donc permettre d'améliorer les performances de HAR. Nous aimerions également capturer des mouvements correspondant à des gestes fins, pour lesquels nous aurions besoin de capteurs plus spécifiques pour classer les actions impliquant uniquement la main. De plus, incorporer des informations contextuelles telles que l'apparence des objets via une reconnaissance d'objets ce qui permet de créer plus de caractéristiques à apprendre pour l'algorithme de HAR en ayant des connaissances à priori sur la séquence d'assemblage et permet ainsi d'améliorer les performances de HAR en réduisant l'ambiguïté et le mélange entre des actions complexes qui peuvent avoir des poses similaires, en particulier lorsque les actions impliquent de très petits objets. Cela devrait permettre d'améliorer les performances de la HAR mais aussi permettre d'avoir un modèle de jumeau numérique (DT) plus précis et correspondant davantage au jumeau physique (PT). D'autres perspectives portent sur l'utilisation de jeux de données complètement synthétique basés sur le jumeau numérique et intégrant des mannequins virtuels dans les simulations d'opérations d'assemblage.

L'une des limites de la technique de la fenêtre glissante déployée dans notre algorithme de HAR en ligne est le choix de sa taille. Celle-ci doit être choisie avec précaution afin de garantir une caractérisation satisfaisante des activités transitoires mais aussi de s'assurer que toutes les actions, y compris les actions courtes, soient bien représentées. A cet égard, l'utilisation de fenêtres glissantes dynamiques pourrait améliorer les performances de HAR puisqu'elles peuvent s'adapter à différentes durées d'actions. D'autres améliorations peuvent être apportées sur notre système de reconnaissance d'actions. Des perspectives portent également sur l'intégration de ces approches dans le contrôleur du cobot pour permettre une détection d'actions en temps réel. Enfin, il aurait aussi été intéressant de tester notre algorithme de HAR sur d'autres cas d'étude d'autres types d'industrie pour mieux évaluer sa généralisation.

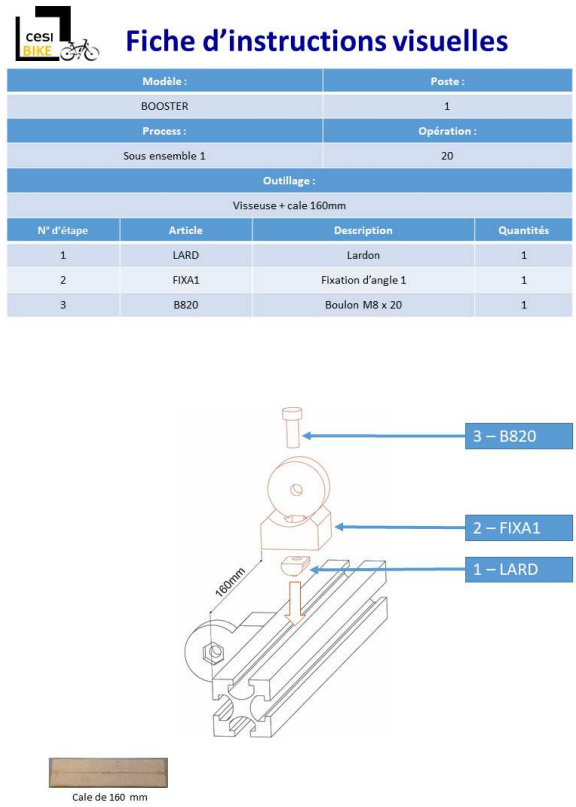
Liste des publications

- M. DALLEL, V. HAVARD, D. BAUDRY and X. SAVATIER, "InHARD - Industrial Human Action Recognition Dataset in the Context of Industrial Collaborative Robotics", 2020 IEEE International Conference on Human-Machine Systems (ICHMS), 2020, pp. 1-6, doi: 10.1109/ICHMS49158.2020.9209531.

- M. DALLEL, V. HAVARD, Y. DUPUIS, and D. BAUDRY. 2022. A Sliding Window Based Approach With Majority Voting for Online Human Action Recognition using Spatial Temporal Graph Convolutional Neural Networks. In 2022 7th International Conference on Machine Learning Technologies (ICMLT) (ICMLT 2022). Association for Computing Machinery, New York, NY, USA, 155–163. https://doi.org/10.1145/3529399.3529425.

- M. DALLEL, V. HAVARD, Y. DUPUIS, and D. BAUDRY, Digital twin of an industrial workstation: A novel method of an auto-labeled data generator using virtual reality for human action recognition in the context of human–robot collaboration. Engineering Applications of Artificial Intelligence (2022) 105655, https://doi.org/10.1016/j.engappai.2022.105655.

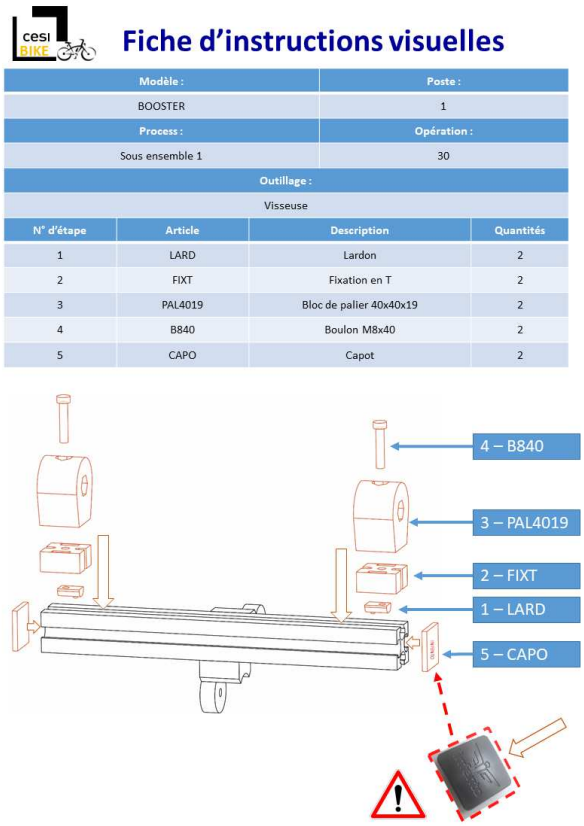
Annexe - Fiches d'instructions visuelles dans InHARD



Fiche d'instructions visuelles

Modèle :	Poste :
BOOSTER	1
Process :	**Opération :**
Sous ensemble 1	20
Outillage :	
Visseuse + cale 160mm	

N° d'étape	Article	Description	Quantités
1	LARD	Lardon	1
2	FIXA1	Fixation d'angle 1	1
3	B820	Boulon M8 x 20	1



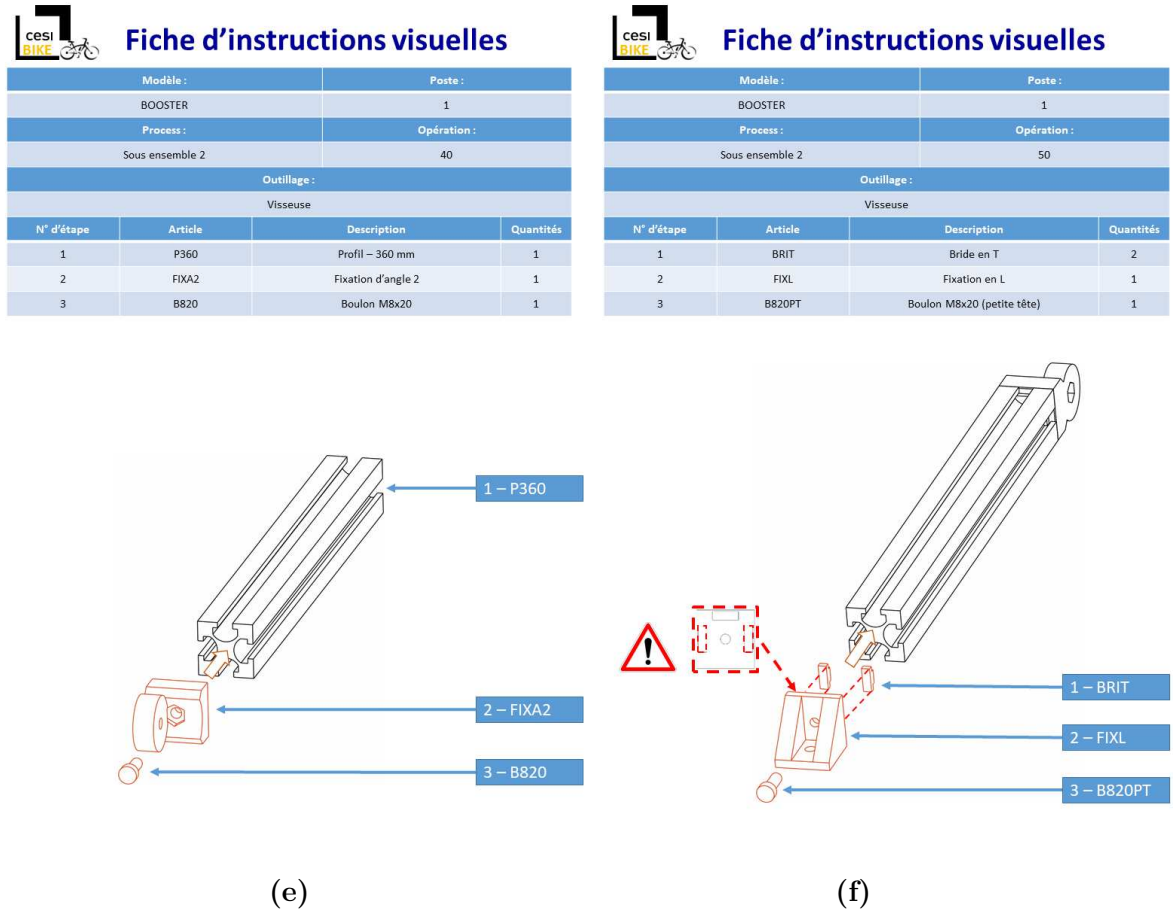
Fiche d'instructions visuelles

Modèle :	Poste :
BOOSTER	1
Process :	**Opération :**
Sous ensemble 1	30
Outillage :	
Visseuse	

N° d'étape	Article	Description	Quantités
1	LARD	Lardon	2
2	FIXT	Fixation en T	2
3	PAL4019	Bloc de palier 40x40x19	2
4	B840	Boulon M8x40	2
5	CAPO	Capot	2

(c) **(d)**

FIGURE 1- Deuxième (c) et troisième opération (d) de la manipulation d'assemblage dans le jeu de données InHARD



Fiche d'instructions visuelles

Modèle :	Poste :
BOOSTER	1
Process :	**Opération :**
Sous ensemble 2	40
Outillage :	
Visseuse	

N° d'étape	Article	Description	Quantités
1	P360	Profil – 360 mm	1
2	FIXA2	Fixation d'angle 2	1
3	B820	Boulon M8x20	1

(e)

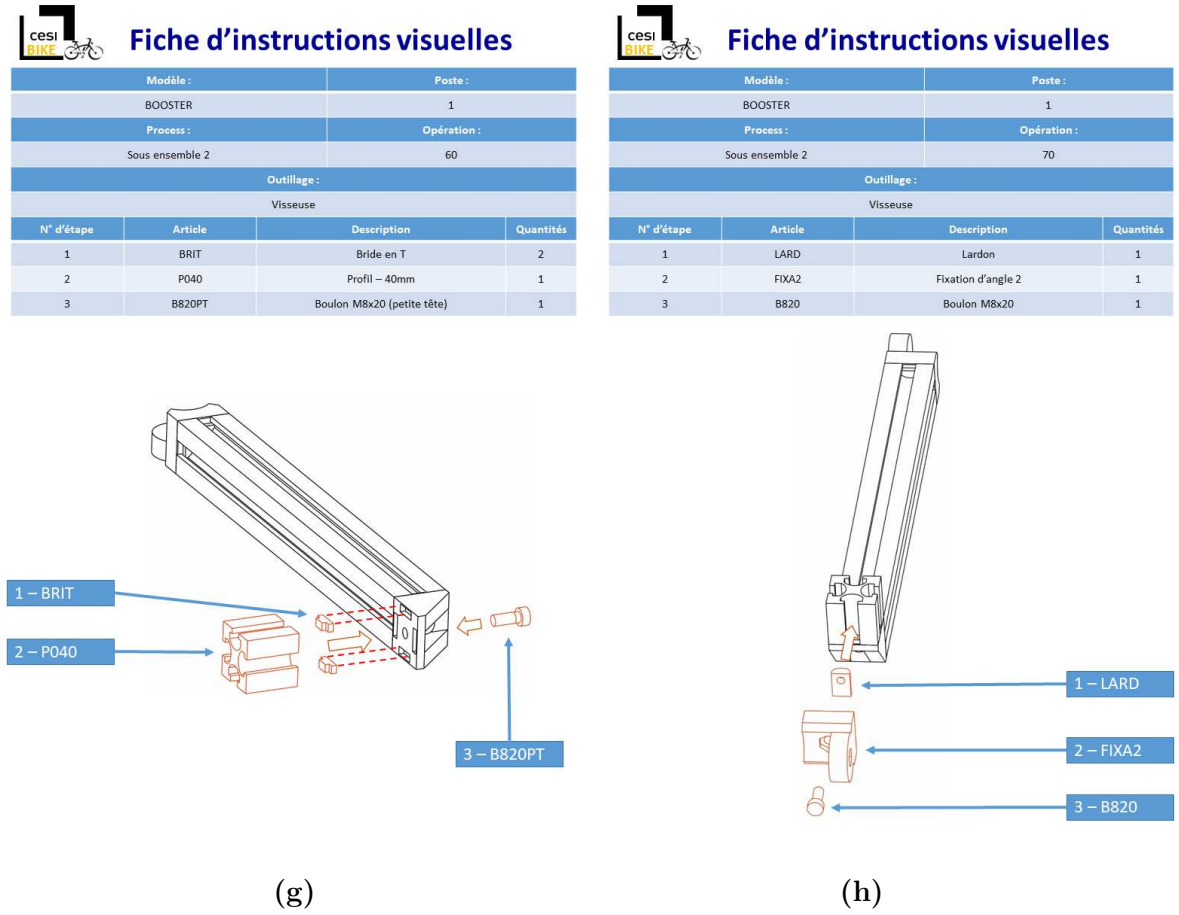
Fiche d'instructions visuelles

Modèle :	Poste :
BOOSTER	1
Process :	**Opération :**
Sous ensemble 2	50
Outillage :	
Visseuse	

N° d'étape	Article	Description	Quantités
1	BRIT	Bride en T	2
2	FIXL	Fixation en L	1
3	B820PT	Boulon M8x20 (petite tête)	1

(f)

FIGURE 2- Quatrième (e) et cinquième opération (f) de la manipulation d'assemblage dans le jeu de données InHARD



Fiche d'instructions visuelles

Modèle :	Poste :
BOOSTER	1
Process :	**Opération :**
Sous ensemble 2	60
Outillage :	
Visseuse	

N° d'étape	Article	Description	Quantités
1	BRIT	Bride en T	2
2	P040	Profil – 40mm	1
3	B820PT	Boulon M8x20 (petite tête)	1

(g)

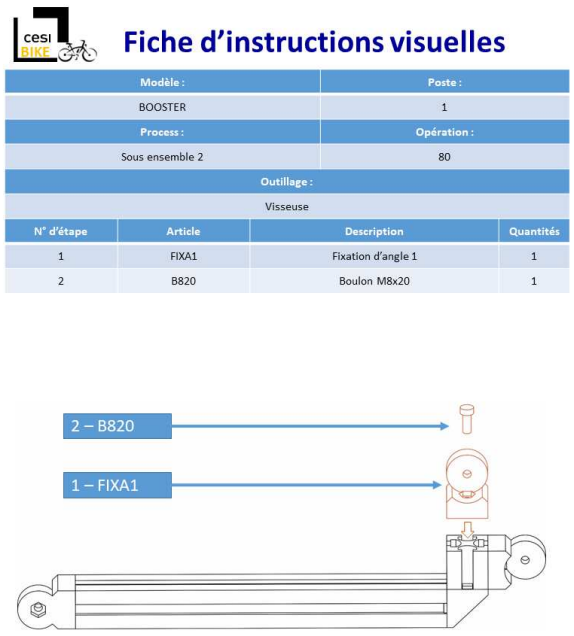
Fiche d'instructions visuelles

Modèle :	Poste :
BOOSTER	1
Process :	**Opération :**
Sous ensemble 2	70
Outillage :	
Visseuse	

N° d'étape	Article	Description	Quantités
1	LARD	Lardon	1
2	FIXA2	Fixation d'angle 2	1
3	B820	Boulon M8x20	1

(h)

FIGURE 3- Sixième (g) et septième opération (h) de la manipulation d'assemblage dans le jeu de données InHARD



Fiche d'instructions visuelles

Modèle :	Poste :
BOOSTER	1
Process :	**Opération :**
Sous ensemble 2	80
Outillage :	
Visseuse	

N° d'étape	Article	Description	Quantités
1	FIXA1	Fixation d'angle 1	1
2	B820	Boulon M8x20	1

(i)

FIGURE 4- Septième (i) opération de la manipulation d'assemblage dans le jeu de données InHARD

Annexe - Typologie de jumeau numérique

En fonction de du niveau d'intégration des données entre les systèmes physiques et numériques et de son automatisation, trois sous-catégories sont identifiées : modèle numérique, ombre numérique et jumeau numérique. Ce niveau d'intégration des données va des modèles de représentation numérique manuels aux modèles entièrement intégrés avec échange de données en temps réel (Kritzinger, et al. 2018).

1. Modèle numérique

Un modèle numérique est une représentation numérique d'un système physique réel, qui n'utilise aucune forme d'échange de données automatisé entre le système physique et son homologue numérique. Par conséquent, une description plus ou moins complète du système physique peut être déduite de la représentation numérique. Ces modèles peuvent inclure des algorithmes et des modèles mathématiques, qui n'utilisent aucune forme d'intégration automatique des données. Pour développer de tels modèles, les données numériques du système physique peuvent être utilisées. Néanmoins, l'échange de données se fait manuellement, c'est-à-dire que tout changement d'état du système physique n'a pas d'effet explicite sur le système numérique et vice-versa, comme le montre la Figure 1FIGURE .

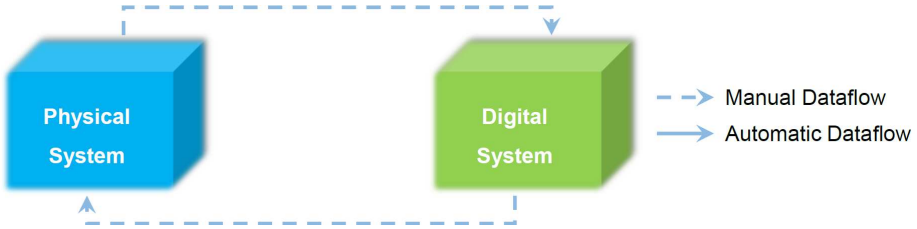


FIGURE 1 - Le flux de données dans un modèle numérique

2. Ombre numérique

Sur la base de la définition précédente, nous pouvons qualifier un modèle d'ombre numérique s'il existe un flux de données unidirectionnel automatisé entre le système physique et son homologue numérique. Tout changement d'état explicite du système physique entraîne un changement d'état du système numérique, mais pas l'inverse, comme le décrit la Figure 2.

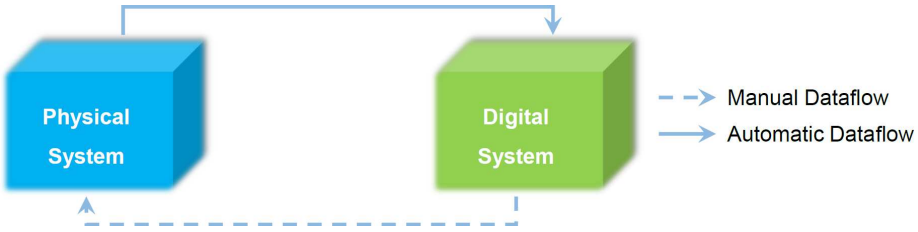


FIGURE 2 - Le flux de données dans une ombre numérique

3. Jumeau numérique

Nous parlons de jumeau numérique lorsque le flux de données entre le système physique et le système numérique est automatisé dans les deux directions. Par conséquent, le système numérique peut agir comme une instance de contrôle du système physique. Il est aussi possible que d'autres systèmes/sous-systèmes physiques ou numériques puissent modifier l'état du système physique. Par conséquent, tout changement d'état du système physique entraîne explicitement un changement d'état du système numérique et vice-versa, comme le montre la Figure 3.

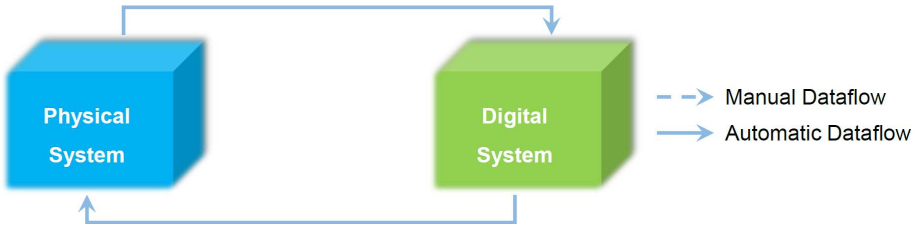


FIGURE 3 - Le flux de données dans un jumeau numérique

Bibliographie

Aggarwal, Jake K., et Michael S. Ryoo. «Human activity analysis: A review.» *ACM Computing Surveys (CSUR)* (ACM) 43 (2011): 16.

Aivaliotis, P., K. Georgoulias, Z. Arkouli, et S. Makris. «Methodology for enabling Digital Twin using advanced physics-based modelling in predictive maintenance.» *Procedia CIRP* 81 (2019): 417-422.

Al-Amin, Md., et al. «Action Recognition in Manufacturing Assembly using Multimodal Sensor Fusion.» *Procedia Manufacturing* 39 (2019): 158-167.

Asadi-Aghbolaghi, Maryam, et al. «A survey on deep learning based approaches for action and gesture recognition in image sequences.» *2017 12th IEEE international conference on automatic face & gesture recognition (FG 2017).* 2017. 476-483.

Askarpour, Mehrnoosh, Dino Mandrioli, Matteo Rossi, et Federico Vicentini. «Formal model of human erroneous behavior for safety analysis in collaborative robotics.» *Robotics and Computer-Integrated Manufacturing* 57 (2019): 465-476.

Baek, Seungryul, Kwang In Kim, et Tae-Kyun Kim. «Real-time Online Action Detection Forests using Spatio-temporal Contexts.» *Computing Research Repository (CoRR)* abs/1610.09334 (2017).

Banerjee, Avinandan, Pawan Kumar Singh, et Ram Sarkar. «Fuzzy Integral-Based CNN Classifier Fusion for 3D Skeleton Action Recognition.» *IEEE Transactions on Circuits and Systems for Video Technology* 31 (2021): 2206-2216.

Banos, Oresti, Juan-Manuel Galvez, Miguel Damas, Hector Pomares, et Ignacio Rojas. «Window Size Impact in Human Activity Recognition.» *Sensors* 14 (2014): 6474–6499.

Bauer, Andrea Maria, Dirk Wollherr, et Martin Buss. «Human-Robot Collaboration: a Survey.» *Int. J. Humanoid Robotics* 5 (2008): 47-66.

Bilen, Hakan, Basura Fernando, Efstratios Gavves, et Andrea Vedaldi. «Action Recognition with Dynamic Image Networks.» *IEEE Transactions on Pattern Analysis and Machine Intelligence* 40 (2018): 2799-2813.

Bloom, Victoria, Dimitrios Makris, et Vasileios Argyriou. «G3D: A gaming action dataset and real time action recognition evaluation framework.» *2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops.* 2012. 7-12.

Bobick, Aaron F., et James W. Davis. «The recognition of human movement using temporal templates.» *IEEE Transactions on Pattern Analysis & Machine Intelligence* (IEEE), 2001: 257-267.

Brandao, Raul, et Martin Wynn. «Product Lifecycle Management Systems and Business Process Improvement – A Report on Case Study Research.» *2008 The Third International Multi-Conference on Computing in the Global Information Technology (iccgi 2008).* 2008. 113-118.

Bronstein, Michael M., Joan Bruna, Yann LeCun, Arthur Szlam, et Pierre Vandergheynst. «Geometric Deep Learning: Going beyond Euclidean data.» *IEEE Signal Processing Magazine* 34 (2017): 18-42.

Bryndin, Evgeniy. «Formation and Management of Industry 5.0 by Systems with Artificial Intelligence and Technological Singularity.» *American Journal of Mechanical and Industrial Engineering* Vol. 5, No. 2 (2020).

Burghardt, Andrzej, Dariusz Szybicki, Piotr Gierlak, Krzysztof Kurc, Paulina Pietruś, et Rafał Cygan. «Programming of Industrial Robots Using Virtual Reality and Digital Twins.» *Applied Sciences* 10 (1 2020): 486.

Caetano, Carlos, Jessica Sena, Francois F. Bremond, Jefersson Alex Dos Santos, et William Robson Schwartz. «SkeleMotion: A New Representation of Skeleton Joint Sequences Based on Motion Information for 3D Action Recognition.» *AVSS 2019 - 16th IEEE International Conference on*

Advanced Video and Signal-based Surveillance. Taipei, Taiwan, 2019.

Cai, Jinmiao, Nianjuan Jiang, Xiaoguang Han, Kui Jia, et Jiangbo Lu. «JOLO-GCN: Mining Joint-Centered Light-Weight Information for Skeleton-Based Action Recognition.» *CoRR* abs/2011.07787 (2020).

Cao, Zhe, Gines Hidalgo, Tomas Simon, Shih-En Wei, et Yaser Sheikh. «OpenPose: Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields.» *CoRR* abs/1812.08008 (2018).

Cao, Zhe, Tomas Simon, Shih-En Wei, et Yaser Sheikh. «Realtime multi-person 2d pose estimation using part affinity fields.» *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.* 2017. 7291-7299.

Carreira, Joao, et Andrew Zisserman. «Quo Vadis, Action Recognition? A New Model and the Kinetics Dataset.» *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR).* 2017.

Carreira, João, et Andrew Zisserman. «Quo Vadis, Action Recognition? A New Model and the Kinetics Dataset.» *CoRR* abs/1705.07750 (2017).

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

Cheng, Ke, Yifan Zhang, Xiangyu He, Weihan Chen, Jian Cheng, et Hanqing Lu. «Skeleton-Based Action Recognition With Shift Graph Convolutional Network.» *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020: 180-189.

Chéron, Guilhem, Ivan Laptev, et Cordelia Schmid. «P-CNN: Pose-based CNN Features for Action Recognition.» *CoRR* abs/1506.03607 (2015).

Cherubini, Andrea, Robin Passama, Philippe Fraisse, et André Crosnier. «A unified multimodal control framework for human–robot interaction.» *Robotics and Autonomous Systems* 70 (2015): 106-115.

Choo, Corliss Zhi Yi, Jia Yi Chow, et John Komar. «Validation of the Perception Neuron system for full-body motion capture.» *PLOS ONE* (Public Library of Science) 17 (1 2022): 1-18.

Chryssolouris, G., D. Mavrikios, N. Papakostas, D. Mourtzis, G. Michalos, et K. Georgoulias. «Digital manufacturing: History, perspectives, and outlook.» *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture* 223 (2009): 451-462.

«CMU Mocap Dataset.» *CMU Mocap Dataset.* s.d.

Crasto, Nieves, Philippe Weinzaepfel, Karteek Alahari, et Cordelia Schmid. «MARS: Motion-Augmented RGB Stream for Action Recognition.» *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR).* 2019. 7874-7883.

Dallel, Mejdi, Vincent Havard, David Baudry, et Xavier Savatier. «InHARD - Industrial Human Action Recognition Dataset in the Context of Industrial Collaborative Robotics.» *2020 IEEE International Conference on Human-Machine Systems (ICHMS).* 2020. 1-6.

Dallel, Mejdi, Vincent Havard, Yohan Dupuis, et David Baudry. «A Sliding Window Based Approach With Majority Voting for Online Human Action Recognition using Spatial Temporal Graph Convolutional Neural Networks.» *2022 7th International Conference on Machine Learning Technologies (ICMLT).* 2022.

Dandekar, Ashish, Remmy A. M. Zen, et S. Bressan. «Comparative Evaluation of Synthetic Dataset Generation Methods.» 2017.

Das, Srijan, Saurav Sharma, Rui Dai, Francois F. Bremond, et Monique Thonnat. «VPN: Learning Video-Pose Embedding for Activities of Daily Living.» *ECCV 2020 - 16th European Conference on Computer Vision.* Glasgow (Virtual), United Kingdom, 2020.

de Melo, Celso M., Brandon Rothrock, Prudhvi Gurram, Oytun Ulutan, et B. S. Manjunath. «Vision-Based Gesture Recognition in Human-Robot Teams Using Synthetic Data.» *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS).* 2020. 10278-10284.

Defferrard, Michaël, Xavier Bresson, et Pierre Vandergheynst. «Convolutional Neural Networks on Graphs

with Fast Localized Spectral Filtering.» *Proceedings of the 30th International Conference on Neural Information Processing Systems.* Red Hook, NY, USA: Curran Associates Inc., 2016. 3844–3852.

Dehghani, Akbar, Omid Sarbishei, Tristan Glatard, et Emad Shihab. «A Quantitative Comparison of Overlapping and Non-Overlapping Sliding Windows for Human Activity Recognition Using Inertial Sensors.» *Sensors* 19 (2019).

Delamare, Mickael, Cyril Laville, Adnane Cabani, et Houcine Chafouk. «Graph Convolutional Networks Skeleton-based Action Recognition for Continuous Data Stream: A Sliding Window Approach.» *16th International Conference on Computer Vision Theory and Applications.* Online, Streaming: SCITEPRESS - Science and Technology Publications, 2021. 427-435.

Devanne, Maxime, Hazem Wannous, Stefano Berretti, Pietro Pala, Mohamed Daoudi, et Alberto Del Bimbo. «3-d human action recognition by shape analysis of motion trajectories on riemannian manifold.» *IEEE transactions on cybernetics* (IEEE) 45 (2014): 1340-1352.

Dhiman, Chhavi, et Dinesh Kumar Vishwakarma. «View-Invariant Deep Architecture for Human Action Recognition Using Two-Stream Motion and Shape Temporal Dynamics.» *IEEE Transactions on Image Processing* 29 (2020): 3835-3844.

Diba, Ali, et al. «Temporal 3D ConvNets: New Architecture and Transfer Learning for Video Classification.» *CoRR* abs/1711.08200 (2017).

Diba, Ali, Vivek Sharma, et Luc Van Gool. «Deep Temporal Linear Encoding Networks.» *CoRR* abs/1611.06678 (2016).

Donahue, Jeff, et al. «Long-Term Recurrent Convolutional Networks for Visual Recognition and Description.» *IEEE Transactions on Pattern Analysis and Machine Intelligence* 39 (2017): 677-691.

Du, Yong, Wei Wang, et Liang Wang. «Hierarchical recurrent neural network for skeleton based action recognition.» *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR).* 2015. 1110-1118.

Duan, Haodong, Yue Zhao, Kai Chen, Dian Shao, Dahua Lin, et Bo Dai. «Revisiting Skeleton-based Action Recognition.» *CoRR* abs/2104.13586 (2021).

Duan, Haodong, Yue Zhao, Yuanjun Xiong, Wentao Liu, et Dahua Lin. «Omni-Sourced Webly-Supervised Learning for Video Recognition.» 670-688. 2020.

Ekbatani., Hadi Keivan, Oriol Pujol., et Santi Segui. «Synthetic Data Generation for Deep Learning in Counting Pedestrians.» *Proceedings of the 6th International Conference on Pattern Recognition Applications and Methods - Volume 1: ICPRAM,.* SciTePress, 2017. 318-323.

El Makrini, Ilias, Kelly Merckaert, Dirk Lefeber, et Bram Vanderborght. «Design of a collaborative architecture for human-robot assembly tasks.» *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS).* 2017. 1624-1629.

El Zaatari, Shirine, Mohamed Marei, W. Li, et Zahid Usman. «Cobot programming for collaborative industrial tasks: An overview.» *Robotics and Autonomous Systems* 116 (6 2019): 162-180.

Elias, Petr, Jan Sedmidubský, et Pavel Zezula. «Understanding the limits of 2D skeletons for action recognition.» *Multimedia Systems* 27 (2021): 547-561.

Ellis, Chris, Syed Zain Masood, Marshall F. Tappen, Joseph J. LaViola, et Rahul Sukthankar. «Exploring the trade-off between accuracy and observational latency in action recognition.» *International Journal of Computer Vision* (Springer) 101 (2013): 420-436.

Emma-Ogbangwo, Chika, Nick Cope, Reinhold Behringer, et Marc Fabri. «Enhancing user immersion and virtual presence in interactive multiuser virtual environments through the development and integration of a gesture-centric natural user interface developed from existing virtual reality technologies.» *International Conference on Human-Computer Interaction.* 2014. 410-414.

Escalera, Sergio, et al. «ChaLearn Looking at People Challenge 2014: Dataset and Results.» Édité par Lourdes Agapito, Michael M. Bronstein et Carsten Rother. *Computer Vision - ECCV 2014*

Workshops. Cham: Springer International Publishing, 2015. 459–473.

Escobedo Cardenas, Edwin, et Guillermo Camara Chavez. «Multimodal Human Action Recognition Based on a Fusion of Dynamic Images Using CNN Descriptors.» *2018 31st SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI).* 2018. 95-102.

Escorcia, Victor, Fabian Caba Heilbron, Juan Carlos Niebles, et Bernard Ghanem. «DAPs: Deep Action Proposals for Action Understanding.» *European Conference on Computer Vision (ECCV 2016)*, 2016.

Fayyaz, Mohsen, et al. «3D CNNs with Adaptive Temporal Feature Resolutions.» *3D CNNs with Adaptive Temporal Feature Resolutions.* 11 2020.

Feichtenhofer, Christoph, Axel Pinz, et Andrew Zisserman. «Convolutional Two-Stream Network Fusion for Video Action Recognition.» *CoRR* abs/1604.06573 (2016).

Feichtenhofer, Christoph, Axel Pinz, et Richard P. Wildes. «Spatiotemporal Multiplier Networks for Video Action Recognition.» *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR).* 2017. 7445-7454.

Feichtenhofer, Christoph, Axel Pinz, et Richard Wildes. «Spatiotemporal Residual Networks for Video Action Recognition.» 11 2016.

Feichtenhofer, Christoph, Haoqi Fan, Jitendra Malik, et Kaiming He. «SlowFast Networks for Video Recognition.» *CoRR* abs/1812.03982 (2018).

Fothergill, Simon, Helena Mentis, Pushmeet Kohli, et Sebastian Nowozin. «Instructing People for Training Gestural Interactive Systems.» Dans *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 1737–1746. New York, NY, USA: Association for Computing Machinery, 2012.

Franciosa, Pasquale, Mikhail Sokolov, Sumit Sinha, Tianzhu Sun, et Dariusz Ceglarek. «Deep learning enhanced digital twin for Closed-Loop In-Process quality improvement.» *CIRP Annals* 69 (2020): 369-372.

Fuller, Aidan, Zhong Fan, Charles Day, et Chris Barlow. «Digital Twin: Enabling Technologies, Challenges and Open Research.» *IEEE Access* 8 (2020): 108952-108971.

Gabler, Volker, Tim Stahl, Gerold Huber, Ozgur Oguz, et Dirk Wollherr. «A game-theoretic approach for adaptive action selection in close proximity human-robot-collaboration.» *2017 IEEE International Conference on Robotics and Automation (ICRA).* 2017. 2897-2903.

Ge, Hongwei, Zehang Yan, Wenhao Yu, et Liang Sun. «An attention mechanism based convolutional LSTM network for video action recognition.» *Multim. Tools Appl.* 78 (2019): 20533–20556.

Ghadiyaram, Deepti, Du Tran, et Dhruv Mahajan. «Large-Scale Weakly-Supervised Pre-Training for Video Action Recognition.» 2019. 12038-12047.

Girdhar, Rohit, Deva Ramanan, Abhinav Gupta, Josef Sivic, et Bryan C. Russell. «ActionVLAD: Learning spatio-temporal aggregation for action classification.» *CoRR* abs/1704.02895 (2017).

Girshick, R., J. Donahue, T. Darrell, et J. Malik. «Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation.» *Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation.* 6 2014. 580-587.

Girshick, Ross B. «Fast R-CNN.» *CoRR* abs/1504.08083 (2015).

Gkioxari, Georgia, et Jitendra Malik. «Finding Action Tubes.» *CoRR* abs/1411.6031 (2014).

Glaessgen, Edward, et David Stargel. «The digital twin paradigm for future NASA and U.S. air force vehicles.» 2012.

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

Grieves, Michael, et John Vickers. «Digital Twin: Mitigating Unpredictable, Undesirable Emergent Behavior in Complex Systems.» 85-113. 2017.

Hadall, Jeremy. *Deciphering Industry 4.0 - Part 2: Human and Robot Collaboration.* 2017.

Halim, A. Aly, Christel Dartigues-Pallez, Frédéric Precioso, Michel Riveill, Abderrahim Benslimane, et S. Ghoneim. «Human action recognition based on 3D skeleton part-based pose estimation and temporal multi-resolution analysis.» *2016 IEEE International Conference on Image Processing (ICIP).* 2016. 3041-3045.

Havard, V., M. Sahnoun, B. Bettayeb, et D. Baudry. «An Architecture for Data Management, Visualisation and Supervision of Cyber-Physical Production Systems.» *Advances in Manufacturing Technology XXXIII: Proceedings of the 17th International Conference on Manufacturing Research, incorporating the 34th National Conference on Manufacturing Research, 10-12 September 2019, Queen's University, Belfast.* 2019. 81.

Havard, Vincent, Benoit Jeanne, Marc Lacomblez, et David Baudry. «Digital Twin and Virtual Reality: A Co-Simulation Environment for Design and Assessment of Industrial Workstations.» *Production & Manufacturing Research* (Informa UK Limited) 7 (1 2019): 472–489.

He, Jun-Yan, Xiao Wu, Zhi-Qi Cheng, Zhaoquan Yuan, et Yu-Gang Jiang. «DB-LSTM: Densely-connected Bi-directional LSTM for human action recognition.» *Neurocomputing* 444 (2021): 319-331.

Heilbron, Fabian Caba, Victor Escorcia, Bernard Ghanem, et Juan Carlos Niebles. «ActivityNet: A large-scale video benchmark for human activity understanding.» *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR).* 2015. 961-970.

Hou, Rui, Chen Chen, et Mubarak Shah. «Tube Convolutional Neural Network (T-CNN) for Action Detection in Videos.» *CoRR* abs/1703.10664 (2017).

Hou, Yonghong, Zhaoyang Li, Pichao Wang, et Wanqing Li. «Skeleton Optical Spectra-Based Action Recognition Using Convolutional Neural Networks.» *IEEE Transactions on Circuits and Systems for Video Technology* 28 (2018): 807-811.

Hu, Jian-Fang, Wei-Shi Zheng, Jiahui Pan, Jianhuang Lai, et Jianguo Zhang. «Deep Bilinear Learning for RGB-D Action Recognition: 15th European Conference, Munich, Germany, September 8–14, 2018, Proceedings, Part VII.» 346-362. 2018.

Huang, De-An, Li Fei-Fei, et Juan Carlos Niebles. «Connectionist Temporal Modeling for Weakly Supervised Action Labeling.» *CoRR* abs/1607.08584 (2016).

Huang, Gao, Zhuang Liu, et Kilian Q. Weinberger. «Densely Connected Convolutional Networks.» *CoRR* abs/1608.06993 (2016).

Idrees, Haroon, et al. «The THUMOS Challenge on Action Recognition for Videos "in the Wild".» *CoRR* abs/1604.06182 (2016).

iMotions Biometric Research Platform 7.1. «iMotions A/S Copehnhagen, Denmark.» *iMotions A/S Copehnhagen, Denmark.* 2019.

Ji, Shuiwang, Wei Xu, Ming Yang, et Kai Yu. «3D Convolutional Neural Networks for Human Action Recognition.» *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35 (2013): 221-231.

Ji, Yanli, Feixiang Xu, Yang Yang, Fumin Shen, Heng Tao Shen, et Wei-Shi Zheng. «A Large-scale Varying-view RGB-D Action Dataset for Arbitrary-view Human Action Recognition.» *CoRR* abs/1904.10681 (2019).

Jiang, Feng, Shengping Zhang, Shen Wu, Yang Gao, et Debin Zhao. «Multi-Layered Gesture Recognition with Kinect.» *J. Mach. Learn. Res.* (JMLR.org) 16 (1 2015): 227–254.

Johannsmeier, Lars, et Sami Haddadin. «A Hierarchical Human-Robot Interaction-Planning Framework for Task Allocation in Collaborative Industrial Assembly Processes.» *IEEE Robotics and Automation Letters* 2 (2017): 41-48.

Johansson, Gunnar. «Visual perception of biological motion and a model for its analysis.» *Perception & psychophysics* (Springer) 14 (1973): 201-211.

Kalfaoglu, M. E., Sinan Kalkan, et Aydin Alatan. «Late Temporal Modeling in 3D CNN Architectures with BERT for Action Recognition.» *ECCV Workshops.* 2020.

Kamel, Aouaidjia, Bin Sheng, Po Yang, Ping Li, Ruimin Shen, et David Dagan Feng. «Deep Convolutional

Neural Networks for Human Action Recognition Using Depth Maps and Postures.» *IEEE Transactions on Systems, Man, and Cybernetics: Systems* 49 (2019): 1806-1819.

Kamthe, Sanket, Samuel Assefa, et Marc Deisenroth. «Copula Flows for Synthetic Data Generation.» *Copula Flows for Synthetic Data Generation.* 2021.

Kar, Amlan, Nishant Rai, Karan Sikka, et Gaurav Sharma. «AdaScan: Adaptive Scan Pooling in Deep Convolutional Neural Networks for Human Action Recognition in Videos.» *CoRR* abs/1611.08240 (2016).

Karpathy, Andrej, George Toderici, Sanketh Shetty, Thomas Leung, Rahul Sukthankar, et Li Fei-Fei. «Large-Scale Video Classification with Convolutional Neural Networks.» *2014 IEEE Conference on Computer Vision and Pattern Recognition.* 2014. 1725-1732.

Kay, Will, et al. «The Kinetics Human Action Video Dataset.» *CoRR* abs/1705.06950 (2017).

Ke, Qiuhong, Mohammed Bennamoun, Senjian An, Ferdous Sohel, et Farid Boussaid. «Learning Clip Representations for Skeleton-Based 3D Action Recognition.» *IEEE Transactions on Image Processing* 27 (2018): 2842-2855.

Ke, Qiuhong, Senjian An, Mohammed Bennamoun, Ferdous Sohel, et Farid Boussaid. «SkeletonNet: Mining Deep Part Features for 3-D Action Recognition.» *IEEE Signal Processing Letters* 24 (2017): 731-735.

Khaire, Pushpajit, Javed Imran, et Praveen Kumar. «Human Activity Recognition by Fusion of RGB, Depth, and Skeletal Data.» 409-421. 2018.

Kim, Tae Soo, et Austin Reiter. «Interpretable 3D Human Action Analysis with Temporal Convolutional Networks.» *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops 2017, Honolulu, HI, USA, July 21-26, 2017.* IEEE Computer Society, 2017. 1623–1631.

Kipp, Michael, Levin Freiherr von Hollen, Michael Christopher Hrstka, et Franziska Zamponi. «Single-Person and Multi-Party 3D Visualizations for Nonverbal Communication Analysis.» *LREC.* 2014. 3393–3397.

Korban, Matthew, et X. Li. «DDGCN: A Dynamic Directed Graph Convolutional Network for Action Recognition.» *ECCV.* 2020.

Kovács, Gy, et Sebastian Kot. «New logistics and production trends as the effect of global economy changes.» *Polish Journal of Management Studies* 14 (6 2016): 115-126.

Kritzinger, Werner, Matthias Karner, Georg Traar, Jan Henjes, et Wilfried Sihn. «Digital Twin in manufacturing: A categorical literature review and classification.» *IFAC-PapersOnLine* 51 (2018): 1016-1022.

Kuehne, H., H. Jhuang, E. Garrote, T. Poggio, et T. Serre. «HMDB: A large video database for human motion recognition.» *2011 International Conference on Computer Vision.* 2011. 2556-2563.

Kuehne, Hilde, Alexander Richard, et Juergen Gall. «Weakly supervised learning of actions from transcripts.» *CoRR* abs/1610.02237 (2016).

Kukleva, Anna, Hilde Kuehne, Fadime Sener, et Juergen Gall. «Unsupervised learning of action classes with continuous temporal embedding.» *CoRR* abs/1904.04189 (2019).

Kviatkovsky, Igor, Ehud Rivlin, et Ilan Shimshoni. «Online action recognition using covariance of shape and motion.» *Comput. Vis. Image Underst.* 129 (2014): 15-26.

Kwon, Beom, et Sanghoon Lee. «Human Skeleton Data Augmentation for Person Identification over Deep Neural Network.» *Applied Sciences* 10 (2020).

Laptev, Ivan. «On space-time interest points.» *International journal of computer vision* (Springer) 64 (2005): 107-123.

Lea, Colin S., Michael D. Flynn, R. Vidal, A. Reiter, et Gregory Hager. «Temporal Convolutional Networks for Action Segmentation and Detection.» *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017: 1003-1012.

Lee, Hye-Woo, Noo-ri Kim, et Jee-Hyong Lee. «Deep Neural Network Self-training Based on Unsupervised

Learning and Dropout.» *The International Journal of Fuzzy Logic and Intelligent Systems* 17 (3 2017): 1-9.

Lee, Inwoong, Doyoung Kim, Seoungyoon Kang, et Sanghoon Lee. «Ensemble Deep Learning for Skeleton-Based Action Recognition Using Temporal Sliding LSTM Networks.» *2017 IEEE International Conference on Computer Vision (ICCV).* 2017. 1012-1020.

Lee, Jay, Behrad Bagheri, et Hung-An Kao. «A Cyber-Physical Systems architecture for Industry 4.0-based manufacturing systems.» *Manufacturing Letters* 3 (2015): 18-23.

Lev, Guy, Gil Sadeh, Benjamin Klein, et Lior Wolf. «RNN Fisher Vectors for Action Recognition and Image Annotation.» Édité par Bastian Leibe, Jiri Matas, Nicu Sebe et Max Welling. *Computer Vision – ECCV 2016.* Cham: Springer International Publishing, 2016. 833–850.

Li, Bo, Yuchao Dai, Xuelian Cheng, Huahui Chen, Yi Lin, et Mingyi He. «Skeleton based action recognition using translation-scale invariant image mapping and multi-scale deep CNN.» *2017 IEEE International Conference on Multimedia Expo Workshops (ICMEW).* 2017. 601-604.

Li, Chao, Qiaoyong Zhong, Di Xie, et Shiliang Pu. «Co-occurrence Feature Learning from Skeleton Data for Action Recognition and Detection with Hierarchical Aggregation.» 4 2018.

Li, Chuankun, Pichao Wang, Shuang Wang, Yonghong Hou, et Wanqing Li. «Skeleton-based action recognition using LSTM and CNN.» *2017 IEEE International Conference on Multimedia & Expo Workshops (ICMEW).* 2017. 585-590.

Li, Chuankun, Yonghong Hou, Pichao Wang, et Wanqing Li. «Joint Distance Maps Based Action Recognition With Convolutional Neural Networks.» *IEEE Signal Processing Letters* 24 (2017): 624-628.

Li, Jianan, Xuemei Xie, Qingzhe Pan, Yuhan Cao, Z. Zhao, et Guangming Shi. «SGM-Net: Skeleton-guided multimodal network for action recognition.» *Pattern Recognit.* 104 (2020): 107356.

Li, Jun, Peng Lei, et Sinisa Todorovic. «Weakly Supervised Energy-Based Learning for Action Segmentation.» *CoRR* abs/1909.13155 (2019).

Li, Jun, Xianglong Liu, Mingyuan Zhang, et Deqing Wang. «Spatio-temporal deformable 3D ConvNets with attention for action recognition.» *Pattern Recognition* 98 (2020): 107037.

Li, Kunchang, Xianhang Li, Yali Wang, Jun Wang, et Yu Qiao. «CT-Net: Channel Tensorization Network for Video Classification.» *CoRR* abs/2106.01603 (2021).

Li, Linguo, Minsi Wang, Bingbing Ni, Hang Wang, Jiancheng Yang, et Wenjun Zhang. «3D Human Action Representation Learning via Cross-View Consistency Pursuit.» *ArXiv* abs/2104.14466 (2021).

Li, Maosen, Siheng Chen, Xu Chen, Ya Zhang, Yanfeng Wang, et Qi Tian. «Symbiotic Graph Neural Networks for 3D Skeleton-based Human Action Recognition and Motion Prediction.» *IEEE transactions on pattern analysis and machine intelligence* PP (2021).

Li, Shuai, W. Li, Chris Cook, Ce Zhu, et Y. Gao. «Independently Recurrent Neural Network (IndRNN): Building A Longer and Deeper RNN.» *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018: 5457-5466.

Li, Shuai, Wanqing Li, Chris Cook, Yanbo Gao, et Ce Zhu. «Deep Independently Recurrent Neural Network (IndRNN).» *CoRR* abs/1910.06251 (2019).

Li, Wanqing, Zhengyou Zhang, et Zicheng Liu. «Action recognition based on a bag of 3D points.» *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Workshops.* 2010. 9-14.

Li, Yanghao, Cuiling Lan, Junliang Xing, Wenjun Zeng, Chunfeng Yuan, et Jiaying Liu. «Online Human Action Detection using Joint Classification-Regression Recurrent Neural Networks.» 4 2016.

Li, Yanshan, Rongjie Xia, Xing Liu, et Qinghua Huang. «Learning Shape-Motion Representations from Geometric Algebra Spatio-Temporal Model for Skeleton-Based Action Recognition.» *2019 IEEE International Conference on Multimedia and Expo (ICME).* 2019. 1066-1071.

Li, Zhenyang, Efstratios Gavves, Mihir Jain, et Cees Snoek. «VideoLSTM Convolves, Attends and Flows for Action Recognition.» *Computer Vision and Image Understanding*, 7 2016.

Lichiardopol, Stefan, Nathan van de Wouw, et Henk Nijmeijer. «Control scheme for human-robot co-manipulation of uncertain, time-varying loads.» *2009 American Control Conference.* 2009. 1485-1490.

Lim, Kendrik Yan Hong, Pai Zheng, et Chun-Hsien Chen. «A state-of-the-art survey of Digital Twin: techniques, engineering product lifecycle management and business innovation perspectives.» *Journal of Intelligent Manufacturing*, 8 2020.

Liu, Chunhui, Yanghao Li, Yueyu Hu, et Jiaying Liu. «Online action detection and forecast via Multitask deep Recurrent Neural Networks.» *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP).* 2017. 1702-1706.

Liu, Chunhui, Yueyu Hu, Yanghao Li, Sijie Song, et Jiaying Liu. «PKU-MMD: A Large Scale Benchmark for Continuous Multi-Modal Human Action Understanding.» *CoRR* abs/1703.07475 (2017).

Liu, Hongyi, et Lihui Wang. «Gesture recognition for human-robot collaboration: A review.» *International Journal of Industrial Ergonomics* (Elsevier) 68 (2018): 355-367.

Liu, Jian, Hossein Rahmani, Naveed Akhtar, et Ajmal Mian. «Learning Human Pose Models from Synthesized Data for Robust RGB-D Action Recognition.» *International Journal of Computer Vision* 127 (10 2019): 1545-1564.

Liu, Jiaying, Yanghao Li, Sijie Song, Junliang Xing, Cuiling Lan, et Wenjun Zeng. «Multi-Modality Multi-Task Recurrent Neural Network for Online Action Detection.» *IEEE Transactions on Circuits and Systems for Video Technology* 29 (2019): 2667-2682.

Liu, Jun, Amir Shahroudy, Dong Xu, A. Kot, et G. Wang. «Skeleton-Based Action Recognition Using Spatio-Temporal LSTM Network with Trust Gates.» *IEEE Transactions on Pattern Analysis and Machine Intelligence* 40 (2018): 3007-3021.

Liu, Jun, Amir Shahroudy, Gang Wang, Ling-Yu Duan, et Alex C. Kot. «Skeleton-based online action prediction using scale selection network.» *IEEE transactions on pattern analysis and machine intelligence* (IEEE) 42 (2019): 1453–1467.

Liu, Jun, Amir Shahroudy, Mauricio Perez, G. Wang, Ling-yu Duan, et A. Kot. «NTU RGB+D 120: A Large-Scale Benchmark for 3D Human Activity Understanding.» *IEEE Transactions on Pattern Analysis and Machine Intelligence* 42 (2020): 2684-2701.

Liu, Jun, Gang Wang, Ping Hu, Ling-Yu Duan, et Alex C. Kot. «Global Context-Aware Attention LSTM Networks for 3D Action Recognition.» *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR).* 2017. 3671-3680.

Liu, Kun, Wu Liu, Chuang Gan, Mingkui Tan, et Huadong Ma. «T-C3D: Temporal Convolutional 3D Network for Real-Time Action Recognition.» *AAAI.* 2018.

Liu, Mengnan, Fang Shuiliang, Huiyue Dong, et Cunzhi Xu. «Review of digital twin about concepts, technologies, and industrial applications.» *Journal of Manufacturing Systems* 58 (7 2020).

Liu, Mengyuan, Hong Liu, et Chen Chen. «Enhanced skeleton visualization for view invariant human action recognition.» *Pattern Recognition* 68 (2017): 346-362.

Liu, Yu, Fan Yang, et Dominique Ginhac. «ACDnet: An Action Detection network for real-time edge computing based on flow-guided feature approximation and memory aggregation.» *Pattern Recognition Letters* (Elsevier) 145 (5 2021): 118-126.

Liu, Zhenbing, Zeya Li, Ruili Wang, Ming Zong, et Wanting Ji. «Spatiotemporal saliency-based multi-stream networks with attention-aware LSTM for action recognition.» *Neural Computing and Applications* 32 (9 2020).

Liu, Ziyu, Hongwen Zhang, Zhenghao Chen, Zhiyong Wang, et Wanli Ouyang. «Disentangling and Unifying Graph Convolutions for Skeleton-Based Action Recognition.» *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020: 140-149.

Lugaresi, Camillo, et al. «MediaPipe: A Framework for Perceiving and Processing Reality.» *Third Workshop on Computer Vision for AR/VR at IEEE Computer Vision and Pattern Recognition (CVPR) 2019.* 2019.

Luvizon, Diogo Carbonera, Hedi Tabia, et David Picard. «Learning features combination for human action recognition from skeleton sequences.» *Pattern Recognition Letters* 99 (2017): 13-20.

Majd, Mahshid, et Reza Safabakhsh. «Correlational Convolutional LSTM for human action recognition.» *Neurocomputing* 396 (2020): 224-229.

Malik, Ali Ahmad, et Arne Bilberg. «Collaborative robots in assembly: A practical approach for tasks distribution.» *Procedia Cirp* (Elsevier) 81 (2019): 665-670.

Malik, Ali Ahmad, et Arne Bilberg. «Digital Twins of Human Robot Collaboration in a Production Setting.» *Procedia Manufacturing* 17 (2018): 278–285.

Markowska-Kaczmar, Urszula, Halina Kwasnicka, et Mariusz Paradowski. «Intelligent techniques in personalization of learning in e-learning systems.» Dans *Computational Intelligence for Technology Enhanced Learning*, 1-23. Springer, 2010.

Martinez-Gonzalez, Pablo, Sergiu Oprea, Alberto Garcia-Garcia, Alvaro Jover-Alvarez, Sergio Orts-Escolano, et Jose Garcia-Rodriguez. «UnrealROX: an extremely photorealistic virtual reality environment for robotics simulations and synthetic data generation.» *Virtual Reality* (Springer), 2019: 1–18.

Maslarić, Marinko, Svetlana Nikolicic, et Dejan Mirčetić. «Logistics Response to the Industry 4.0: The Physical Internet.» *Open Engineering* 6 (1 2016).

Matsas, Elias, et George-Christopher Vosniakos. «Design of a Virtual Reality Training System for Human–Robot Collaboration in Manufacturing Tasks.» *International Journal on Interactive Design and Manufacturing (IJIDeM)* 11 (5 2017): 139–153.

Maurtua, Iñaki, Aitor Ibarguren, Johan Kildal, Loreto Susperregi, et Basilio Sierra. «Human–robot collaboration in industrial applications: Safety, interaction and trust.» *International Journal of Advanced Robotic Systems* 14 (2017): 1729881417716010.

Mehrang, Saeed, Julia Pietilä, et Ilkka Korhonen. «An Activity Recognition Framework Deploying the Random Forest Classifier and A Single Optical Heart Rate Monitoring and Triaxial Accelerometer Wrist-Band.» *Sensors* 18 (2018).

Meng, Yue, et al. «AR-Net: Adaptive Frame Resolution for Efficient Action Recognition.» 7 2020.

Meredith, Michael, et Steve Maddock. «Motion Capture File Formats Explained.» *Production*, 1 2001.

Montaigne, Institut. *Industrie du futur, prêts, partez !* 2018.

Müller, Meinard, Tido Röder, Michael Clausen, Bernhard Eberhardt, Björn Krüger, et Andreas Weber. «Documentation Mocap database HDM05.» 2007.

Natsis, Antonis, Ioannis Vrellis, Nikiforos M. Papachristos, et Tassos A. Mikropoulos. «Technological factors, user characteristics and didactic strategies in educational virtual environments.» *2012 IEEE 12th International Conference on Advanced Learning Technologies.* 2012. 531-535.

Negri, Elisa, Luca Fumagalli, et Marco Macchi. «A Review of the Roles of Digital Twin in CPS-based Production Systems.» *Procedia Manufacturing* 11 (2017): 939-948.

Ng, Joe, Matthew Hausknecht, Sudheendra Vijayanarasimhan, Oriol Vinyals, Rajat Monga, et George Toderici. «Beyond short snippets: Deep networks for video classification.» 2015. 4694-4702.

Noblecourt, Sylvain, Geoffrey Bourgoin, Vincent Havard, et David Baudry. «Evaluating the Influence of Interaction Technology on Procedural Learning Using Virtual Reality.» *Proceedings of the 27th ACM Symposium on Virtual Reality Software and Technology.* New York, NY, USA: Association for Computing Machinery, 2021.

NOITOM-Ltd. *Perception Neuron 32 Edition v2.* 2018. https://www.noitom.com/perception-neuron-series.

Ofli, Ferda, Rizwan Chaudhry, Gregorij Kurillo, René Vidal, et Ruzena Bajcsy. «Berkeley MHAD: A comprehensive Multimodal Human Action Database.» *2013 IEEE Workshop on Applications of Computer Vision (WACV).* 2013. 53-60.

Ohn-Bar, Eshed, et Mohan Trivedi. «Joint angles similarities and HOG2 for action recognition.» *Proceedings of the IEEE conference on computer vision and pattern recognition workshops.* 2013.

465-470.

Pan, Y., et al. «Compressing Recurrent Neural Networks with Tensor Ring for Action Recognition.» *ArXiv* abs/1811.07503 (2019).

Paraschiakos, Stylianos, et al. «Activity recognition using wearable sensors for tracking the elderly.» *User Modeling and User-Adapted Interaction*, 7 2020.

Patalas-Maliszewska, Justyna, Daniel Halikowski, et Robertas Damaševičius. «An Automated Recognition of Work Activity in Industrial Manufacturing Using Convolutional Neural Networks.» *Electronics* 10 (2021).

Pellas, Nikolaos, Ioannis Kazanidis, Nikolaos Konstantinou, et Georgia Georgiou. «Exploring the educational potential of three-dimensional multi-user virtual worlds for STEM education: A mixed-method systematic literature review.» *Education and Information Technologies* (Springer) 22 (2017): 2235-2279.

Penichet, Victor M. R., Antonio Peñalver, et José A. Gallud. *New trends in interaction, virtual reality and modeling.* Springer, 2013.

Pérez, Luis, Silvia Rodríguez-Jiménez, Nuria Rodríguez, Rubén Usamentiaga, et Daniel F. García. «Digital Twin and Virtual Reality Based Methodology for Multi-Robot Manufacturing Cell Commissioning.» *Applied Sciences* 10 (5 2020): 3633.

Pishchulin, Leonid, et al. «Deepcut: Joint subset partition and labeling for multi person pose estimation.» *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.* 2016. 4929-4937.

Poppe, Ronald. «Vision-based human motion analysis: An overview.» *Computer vision and image understanding* (Elsevier) 108 (2007): 4-18.

Poppe, Ronald, Sophie van der Zee, Dirk K. J. Heylen, et Paul J. Taylor. «AMAB: Automated measurement and analysis of body motion.» *Behavior Research Methods* 46 (2014): 625-633.

Qiao, Ruizhi, Lingqiao Liu, Chunhua Shen, et Anton van den Hengel. «Learning discriminative trajectorylet detector sets for accurate skeleton-based action recognition.» *Pattern Recognition* 66 (2017): 202-212.

Rad, Dana, et Valentina Balas. «A Novel Fuzzy Scoring Approach of Behavioural Interviews in Personnel Selection.» BRAIN. Broad Research in Artificial Intelligence and Neuroscience (7 2020): 178-188.

Rahmani, Hossein, Arif Mahmood, Du Huynh, et Ajmal Mian. «Histogram of Oriented Principal Components for Cross-View Action Recognition.» *IEEE Transactions on Pattern Analysis and Machine Intelligence* 38 (2016): 2430-2443.

Rahmani, Hossein, et Ajmal Mian. «3D Action Recognition from Novel Viewpoints.» *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR).* 2016. 1506-1515.

Rahmani, Hossein, et Mohammed Bennamoun. «Learning Action Recognition Model from Depth and Skeleton Videos.» *2017 IEEE International Conference on Computer Vision (ICCV).* 2017. 5833-5842.

Rani, S. Sandhya, G. Apparao Naidu, et V. Usha Shree. «Kinematic joint descriptor and depth motion descriptor with convolutional neural networks for human action recognition.» *Materials Today: Proceedings* 37 (2021): 3164-3173.

Reddy, Kishore, et Mubarak Shah. «Recognizing 50 human action categories of web videos.» *Machine Vision and Applications* 24 (7 2013).

Ren, Ziliang, Qieshi Zhang, X. Gao, Pengyi Hao, et Jun Cheng. «Multi-modality learning for human action recognition.» *Multimedia Tools and Applications*, 2021: 1-19.

Richard, Killian, Vincent Havard, Jordan His, et David Baudry. «INTERVALES: INTERactive Virtual and Augmented framework for industriaL Environment and Scenarios.» *Advanced Engineering Informatics* 50 (10 2021): 101425.

Roitberg, Alina, Alexander Perzylo, Nikhil Somani, Manuel Giuliani, Markus Rickert, et Alois Knoll. «Human activity recognition in the context of industrial human-robot interaction.» *Signal and*

Information Processing Association Annual Summit and Conference (APSIPA), 2014 Asia-Pacific. 2014. 1-10.

Sanchez-Caballero, A., David Fuentes-Jiménez, et Cristina Losada-Gutiérrez. «Exploiting the ConvLSTM: Human Action Recognition using Raw Depth Video-Based Recurrent Neural Networks.» *ArXiv* abs/2006.07744 (2020).

Sanchez-Caballero, A., et al. «3DFCNN: Real-Time Action Recognition using 3D Deep Neural Networks with Raw Depth Information.» *ArXiv* abs/2006.07743 (2020).

Scholtz, J. «Theory and evaluation of human robot interactions.» *36th Annual Hawaii International Conference on System Sciences, 2003. Proceedings of the.* 2003. 10 pp.-.

Scholtz, Jean C. «Human-Robot Interactions: Creating Synergistic Cyber Forces.» Édité par Alan C. Schultz et Lynne E. Parker. *Multi-Robot Systems: From Swarms to Intelligent Automata.* Dordrecht: Springer Netherlands, 2002. 177–184.

Sers, Ryan, Steph Forrester, Esther Moss, Stephen Ward, Jianjia Ma, et Massimiliano Zecca. «Validity of the Perception Neuron inertial motion capture system for upper body motion analysis.» *Measurement* 149 (2020): 107024.

Shahroudy, Amir, Jun Liu, Tian-Tsong Ng, et Gang Wang. «NTU RGB+D: A Large Scale Dataset for 3D Human Activity Analysis.» 2016.

Shahroudy, Amir, Tian-Tsong Ng, Yihong Gong, et Gang Wang. «Deep Multimodal Feature Analysis for Action Recognition in {RGB}$\mathplus$D Videos.» *IEEE Transactions on Pattern Analysis and Machine Intelligence* (Institute of Electrical and Electronics Engineers (IEEE)) 40 (5 2018): 1045–1058.

Shi, Lei, Yifan Zhang, Jian Cheng, et Hanqing Lu. «Skeleton-Based Action Recognition With Directed Graph Neural Networks.» *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR).* 2019. 7904-7913.

Shi, Yemin, Yonghong Tian, Yaowei Wang, Wei Zeng, et Tiejun Huang. «Learning Long-Term Dependencies for Action Recognition with a Biologically-Inspired Deep Network.» 2017. 716-725.

Shotton, Jamie, et al. «Real-time human pose recognition in parts from single depth images.» *Communications of the ACM* (ACM) 56 (2013): 116-124.

Si, Chenyang, Wentao Chen, Wei Wang, L. Wang, et T. Tan. «An Attention Enhanced Graph Convolutional LSTM Network for Skeleton-Based Action Recognition.» *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019: 1227-1236.

Si, Chenyang, Ya Jing, Wei Wang, Liang Wang, et T. Tan. «Skeleton-Based Action Recognition with Spatial Reasoning and Temporal Stack Learning.» *ArXiv* abs/1805.02335 (2018).

Simonyan, Karen, et Andrew Zisserman. «Two-stream convolutional networks for action recognition in videos.» *Advances in neural information processing systems.* 2014. 568-576.

Song, Sijie, Cuiling Lan, Junliang Xing, Wenjun Zeng, et Jiaying Liu. «An End-to-End Spatio-Temporal Attention Model for Human Action Recognition from Skeleton Data.» *AAAI-17*, 2017.

—. «Skeleton-Indexed Deep Multi-Modal Feature Learning for High Performance Human Action Recognition.» *2018 IEEE International Conference on Multimedia and Expo (ICME).* 2018. 1-6.

Song, Yisheng, Zhang Zhang, Caifeng Shan, et Liang Wang. «Constructing Stronger and Faster Baselines for Skeleton-based Action Recognition.» *ArXiv* abs/2106.15125 (2021).

Soomro, Khurram, Amir Roshan Zamir, et Mubarak Shah. «UCF101: A Dataset of 101 Human Actions Classes From Videos in The Wild.» *CoRR* abs/1212.0402 (2012).

Stroud, Jonathan C., David A. Ross, Chen Sun, Jia Deng, et Rahul Sukthankar. «D3D: Distilled 3D Networks for Video Action Recognition.» *2020 IEEE Winter Conference on Applications of Computer Vision (WACV).* 2020. 614-623.

Su, Benyue, Qingfeng Tang, Jing Jiang, Min Sheng, Ali Abdullah Yahya, et Guangjun Wang. «A novel method for short-time human activity recognition based on improved template matching technique.» *Proceedings of the 15th ACM SIGGRAPH Conference on Virtual-Reality Continuum*

and Its Applications in Industry-Volume 1. 2016. 233-242.

Sudhakaran, Swathikiran, Sergio Escalera, et Oswald Lanz. «LSTA: Long Short-Term Attention for Egocentric Action Recognition.» *CoRR* abs/1811.10698 (2018).

Sun, Lin, Kui Jia, Dit-Yan Yeung, et Bertram E. Shi. «Human Action Recognition Using Factorized Spatio-Temporal Convolutional Networks.» *2015 IEEE International Conference on Computer Vision (ICCV).* 2015. 4597-4605.

Sun, Lin, Kui Jia, Kevin Chen, Dit Yeung, Bert Shi, et Silvio Savarese. «Lattice Long Short-Term Memory for Human Action Recognition.» 2017. 2166-2175.

Sun, Zehua, Jun Liu, Qiuhong Ke, Hossein Rahmani, Mohammed Bennamoun, et Gang Wang. «Human Action Recognition from Various Data Modalities: A Review.» 12 2020.

Tang, Chang, Wanqing Li, Pichao Wang, et Lizhe Wang. «Online human action recognition based on incremental learning of weighted covariance descriptors.» *Information Sciences* 467 (2018): 219-237.

Tang, Yansong, Yi Tian, Jiwen Lu, Peiyang Li, et Jie Zhou. «Deep Progressive Reinforcement Learning for Skeleton-Based Action Recognition.» *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition.* 2018. 5323-5332.

Tao, Fei, Jiangfeng Cheng, Qinglin Qi, Meng Zhang, He Zhang, et Fangyuan Sui. «Digital twin-driven product design, manufacturing and service with big data.» *The International Journal of Advanced Manufacturing Technology* 94 (2 2018).

Tao, Wenjin, Ze-Hao Lai, Ming C. Leu, et Zhaozheng Yin. «Worker Activity Recognition in Smart Manufacturing Using IMU and sEMG Signals with Convolutional Neural Networks.» *Procedia Manufacturing* 26 (2018): 1159-1166.

Tran, Du, Heng Wang, Lorenzo Torresani, Jamie Ray, Yann LeCun, et Manohar Paluri. «A Closer Look at Spatiotemporal Convolutions for Action Recognition.» *CoRR* abs/1711.11248 (2017).

Tran, Du, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, et Manohar Paluri. «Learning Spatiotemporal Features with 3D Convolutional Networks.» 2015. 4489-4497.

Tripathi, Shashank, Siddhartha Chandra, Amit Agrawal, Ambrish Tyagi, James M. Rehg, et Visesh Chari. «Learning to Generate Synthetic Data via Compositing.» *CoRR* abs/1904.05475 (2019).

Turaga, P., R. Chellappa, V. Subrahmanian, et O. Udrea. «A survey of vision-based methods for action representation, segmentation and recognition.» *Comput Vision Image Understanding* 18 (2008): 1473-88.

Uhlemann, Thomas H.-J., Christoph Schock, Christian Lehmann, Stefan Freiberger, et Rolf Steinhilper. «The Digital Twin: Demonstrating the Potential of Real Time Data Acquisition in Production Systems.» *Procedia Manufacturing* 9 (2017): 113-120.

Varol, Gül, Ivan Laptev, Cordelia Schmid, et Andrew Zisserman. «Synthetic Humans for Action Recognition from Unseen Viewpoints.» *International Journal of Computer Vision* (Springer Science and Business Media LLC) 129 (5 2021): 2264–2287.

Varol, Gül, Ivan Laptev, et Cordelia Schmid. «Long-Term Temporal Convolutions for Action Recognition.» *IEEE Transactions on Pattern Analysis and Machine Intelligence* 40 (2018): 1510-1517.

VidalMata, Rosaura G., Walter J. Scheirer, et Hilde Kuehne. «Joint Visual-Temporal Embedding for Unsupervised Learning of Actions in Untrimmed Sequences.» *CoRR* abs/2001.11122 (2020).

Vrigkas, Michalis, Christophoros Nikou, et Ioannis A. Kakadiaris. «A Review of Human Activity Recognition Methods.» *Frontiers in Robotics and AI* 2 (2015).

Wan, Jun, Stan Z. Li, Yibing Zhao, Shuai Zhou, Isabelle Guyon, et Sergio Escalera. «ChaLearn Looking at People RGB-D Isolated and Continuous Datasets for Gesture Recognition.» *2016 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW).* 2016. 761-769.

Wang, Gaojing, Qingquan Li, Lei Wang, Wei Wang, Mengqi Wu, et Tao Liu. «Impact of Sliding Window Length in Indoor Human Motion Modes and Pose Pattern Recognition Based on Smartphone

Sensors.» *Sensors* 18 (2018).

Wang, Heng, Du Tran, Lorenzo Torresani, et Matt Feiszli. «Video Modeling with Correlation Networks.» *CoRR* abs/1906.03349 (2019).

Wang, Hongsong, et Liang Wang. «Modeling Temporal Dynamics and Spatial Configurations of Actions Using Two-Stream Recurrent Neural Networks.» *CoRR* abs/1704.02581 (2017).

Wang, Huogen, Zhanjie Song, W. Li, et P. Wang. «A Hybrid Network for Large-Scale Action Recognition from RGB and Depth Modalities.» *Sensors (Basel, Switzerland)* 20 (2020).

wang, Jiang, Xiaohan Nie, Yin Xia, Ying Wu, et Song-Chun Zhu. «Cross-view Action Modeling, Learning and Recognition.» *Cross-view Action Modeling, Learning and Recognition.* 2014.

Wang, Jiang, Zicheng Liu, Ying Wu, et Junsong Yuan. «Mining actionlet ensemble for action recognition with depth cameras.» *2012 IEEE Conference on Computer Vision and Pattern Recognition.* 2012. 1290-1297.

Wang, Keze, Xiaolong Wang, Liang Lin, Meng Wang, et Wangmeng Zuo. «3D Human Activity Recognition with Reconfigurable Convolutional Neural Networks.» *Proceedings of the 22nd ACM International Conference on Multimedia.* New York, NY, USA: Association for Computing Machinery, 2014. 97–106.

Wang, Limin, et al. «Temporal Segment Networks: Towards Good Practices for Deep Action Recognition.» *CoRR* abs/1608.00859 (2016).

Wang, Limin, Yu Qiao, et Xiaoou Tang. «Action Recognition with Trajectory-Pooled Deep-Convolutional Descriptors.» *CoRR* abs/1505.04868 (2015).

Wang, Limin, Yuanjun Xiong, Dahua Lin, et Luc Van Gool. «UntrimmedNets for Weakly Supervised Action Recognition and Detection.» *CoRR* abs/1703.03329 (2017).

Wang, Limin, Yuanjun Xiong, Zhe Wang, et Yu Qiao. «Towards Good Practices for Very Deep Two-Stream ConvNets.» *CoRR* abs/1507.02159 (2015).

Wang, P., W. Li, Jun Wan, P. Ogunbona, et Xinwang Liu. «Cooperative Training of Deep Aggregation Networks for RGB-D Action Recognition.» *AAAI.* 2018.

Wang, P., W. Li, Song Liu, Yuyao Zhang, Zhimin Gao, et P. Ogunbona. «Large-scale Continuous Gesture Recognition Using Convolutional Neural Networks.» *2016 23rd International Conference on Pattern Recognition (ICPR)*, 2016: 13-18.

Wang, P., W. Li, Zhimin Gao, Chang Tang, et P. Ogunbona. «Depth Pooling Based Large-Scale 3-D Action Recognition With Convolutional Neural Networks.» *IEEE Transactions on Multimedia* 20 (2018): 1051-1061.

Wang, Pichao, Shuang Wang, Zhimin Gao, Yonghong Hou, et Wanqing Li. «Structured Images for RGB-D Action Recognition.» *2017 IEEE International Conference on Computer Vision Workshops (ICCVW).* 2017. 1005-1014.

Wang, Pichao, Wanqing Li, Chuankun Li, et Yonghong Hou. «Action Recognition Based on Joint Trajectory Maps with Convolutional Neural Networks.» *IEEE TRANSACTIONS ON CYBERNETICS*, 2016.

Wang, Pichao, Wanqing Li, Philip Ogunbona, Jun Wan, et Sergio Escalera. «RGB-D-based human motion recognition with deep learning: A survey.» *Computer Vision and Image Understanding* (Elsevier) 171 (2018): 118-139.

Wang, Pichao, Wanqing Li, Song Liu, Zhimin Gao, Chang Tang, et Philip Ogunbona. «Large-scale Isolated Gesture Recognition using Convolutional Neural Networks.» *2016 23rd International Conference on Pattern Recognition (ICPR).* 2016. 7-12.

Wang, Pichao, Wanqing Li, Zhimin Gao, Jing Zhang, Chang Tang, et Philip O. Ogunbona. «Action Recognition From Depth Maps Using Deep Convolutional Neural Networks.» *IEEE Transactions on Human-Machine Systems* 46 (2016): 498-509.

Wang, Pichao, Wanqing Li, Zhimin Gao, Yuyao Zhang, Chang Tang, et Philip Ogunbona. «Scene Flow to Action Map: A New Representation for RGB-D Based Action Recognition With Convolutional

Neural Networks.» *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR).* 2017.

Wang, Xiaolong, Ali Farhadi, et Abhinav Gupta. «Actions ~ Transformations.» *CVPR.* 2016.

Wang, Xuanhan, Lianli Gao, Jingkuan Song, et Hengtao Shen. «Beyond Frame-level CNN: Saliency-Aware 3-D CNN With LSTM for Video Action Recognition.» *IEEE Signal Processing Letters* 24 (2017): 510-514.

Wang, Xuanhan, Lianli Gao, Peng Wang, Xiaoshuai Sun, et Xianglong Liu. «Two-Stream 3-D convNet Fusion for Action Recognition in Videos With Arbitrary Size and Length.» *IEEE Transactions on Multimedia* 20 (2018): 634-644.

Wei, Shih-En, Varun Ramakrishna, Takeo Kanade, et Yaser Sheikh. «Convolutional pose machines.» *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.* 2016. 4724-4732.

Weinzaepfel, Philippe, Zaïd Harchaoui, et Cordelia Schmid. «Learning to track for spatio-temporal action localization.» *CoRR* abs/1506.01929 (2015).

Wen, Yuhui, L. Gao, Hongbo Fu, Fang-Lue Zhang, et Shi-hong Xia. «Graph CNNs with Motif and Variable Temporal Block for Skeleton-Based Action Recognition.» *AAAI.* 2019.

Witten, Ian H., Eibe Frank, et Mark A. Hall. Dans *Data Mining: Practical Machine Learning Tools and Techniques (Third Edition)*, xxi-xxvii. Boston: Morgan Kaufmann, 2011.

Wu, Cong, Xiao-Jun Wu, et Josef Kittler. «Spatial Residual Layer and Dense Connection Block Enhanced Spatial Temporal Graph Convolutional Network for Skeleton-Based Action Recognition.» *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW).* 2019. 1740-1748.

Wu, Zuxuan, Xi Wang, Yu-Gang Jiang, Hao Ye, et Xiangyang Xue. «Modeling Spatial-Temporal Clues in a Hybrid Deep Learning Framework for Video Classification.» 4 2015.

Xia, Lu, Chia-Chih Chen, et J. K. Aggarwal. «View invariant human action recognition using histograms of 3D joints.» *2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops.* 2012. 20-27.

Xiao, Yang, Jun Chen, Z. Cao, Joey Tianyi Zhou, et X. Bai. «Action Recognition for Depth Video using Multi-view Dynamic Images.» *ArXiv* abs/1806.11269 (2019).

Xie, Chunyu, Ce Li, Baochang Zhang, et Chen Chen. «Memory Attention Networks for Skeleton-based Action Recognition.» *International Joint Conference on Artificial Intelligence (IJCAI-18)*, 2018.

Xiu, Yuliang, Jiefeng Li, Haoyu Wang, Yinghong Fang, et Cewu Lu. «Pose Flow: Efficient Online Pose Tracking.» *BMVC.* 2018.

Xu, Mingze, Mingfei Gao, Yi-Ting Chen, Larry S. Davis, et David J. Crandall. «Temporal Recurrent Networks for Online Action Detection.» *CoRR* abs/1811.07391 (2018).

Xu, Yangyang, Jun Cheng, Lei Wang, Haiying Xia, Feng Liu, et Dapeng Tao. «Ensemble One-Dimensional Convolution Neural Networks for Skeleton-Based Action Recognition.» *IEEE Signal Processing Letters* 25 (2018): 1044-1048.

Yan, Sijie, Yuanjun Xiong, et Dahua Lin. «Spatial Temporal Graph Convolutional Networks for Skeleton-Based Action Recognition.» *AAAI 2018*, 2018.

Yang, Xiaodong, et Ying Li Tian. «Eigenjoints-based action recognition using naive-bayes-nearest-neighbor.» *2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops.* 2012. 14-19.

Yang, Xitong, Haoqi Fan, Lorenzo Torresani, Larry Davis, et Heng Wang. «Beyond Short Clips: End-to-End Video-Level Learning with Collaborative Memories.» *Beyond Short Clips: End-to-End Video-Level Learning with Collaborative Memories.* 4 2021.

Yin, Jun, et al. «MC-LSTM: Real-Time 3D Human Action Detection System for Intelligent Healthcare Applications.» *IEEE Transactions on Biomedical Circuits and Systems* 15 (2021): 259-269.

You, Quanzeng, et Hao Jiang. «Action4D: Real-time Action Recognition in the Crowd and Clutter.» *CoRR* abs/1806.02424 (2018).

Yun, Kiwon, Jean Honorio, Debaleena Chattopadhyay, Tamara L. Berg, et Dimitris Samaras. «Two-person interaction detection using body-pose features and multiple instance learning.» *2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*. 2012. 28-35.

Zhang, Bowen, Limin Wang, Zhe Wang, Yu Qiao, et Hanli Wang. «Real-time Action Recognition with Enhanced Motion Vector CNNs.» *CoRR* abs/1604.07669 (2016).

Zhang, Haochen, Dong Liu, et Zhiwei Xiong. «Two-Stream Action Recognition-Oriented Video Super-Resolution.» *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*. 2019. 8798-8807.

Zhang, Pengfei, Cuiling Lan, Junliang Xing, Wenjun Zeng, Jianru Xue, et Nanning Zheng. «View Adaptive Neural Networks for High Performance Skeleton-Based Human Action Recognition.» *IEEE Transactions on Pattern Analysis and Machine Intelligence* 41 (2019): 1963-1978.

Zhang, Pengfei, Cuiling Lan, Junliang Xing, Wenjun Zeng, Jianru Xue, et Nanning Zheng. «View Adaptive Recurrent Neural Networks for High Performance Human Action Recognition from Skeleton Data.» *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017: 2136-2145.

Zhang, Songyang, et al. «Fusing Geometric Features for Skeleton-Based Action Recognition Using Multilayer LSTM Networks.» *IEEE Transactions on Multimedia* 20 (2018): 2330-2343.

Zhao, Chong, Minglin Chen, Jinhao Zhao, Qicong Wang, et Yehu Shen. «3D Behavior Recognition Based on Multi-Modal Deep Space-Time Learning.» *Applied Sciences* 9 (2019).

Zhao, Yang, Zicheng Liu, Lu Yang, et Hong Cheng. «Combing rgb and depth map features for human activity recognition.» *Proceedings of The 2012 Asia Pacific Signal and Information Processing Association Annual Summit and Conference*. 2012. 1-4.

Zhu, Hongyuan, Romain Vial, et Shijian Lu. «TORNADO: A Spatio-Temporal Convolutional Regression Network for Video Action Proposal.» *2017 IEEE International Conference on Computer Vision (ICCV)*. 2017. 5814-5822.

Zhu, Wentao, et al. «Co-occurrence Feature Learning for Skeleton based Action Recognition using Regularized Deep LSTM Networks.» *Computing Research Repository (CoRR)* abs/1603.07772 (2016).

Titre: Reconnaissance d'actions humaines par apprentissage profond et génération de données étiquetées basées sur le jumeau numérique de poste cobotique industriel.

Mots clés: Reconnaissance d'actions humaines industrielles (HAR), Industrie 4.0, Collaboration humain-robot (HRC), Jumeaux numériques (DT), Réseaux convolutionnels à graphes spatio-temporels (ST-GCN).

Résumé: La reconnaissance d'actions humaines (HAR) permet de faciliter les interactions et la collaboration humain-robot (HRC) au sein de l'industrie 4.0. En effet, les robots collaboratifs sont de plus en plus présents dans notre quotidien et induisent une interaction de plus en plus étroite entre l'humain et le robot, concept rassemblé dans le terme « cobotique ». Afin de garantir une collaboration efficace, les robots doivent être capables de comprendre leur environnement et doivent pouvoir communiquer sur leurs tâches en cours et leurs intentions. Cette communication et ces interactions représentent un enjeu majeur de performance et de sécurité. Dans ce contexte, cette thèse aborde le problème de reconnaissance d'actions humaines dans un milieu industriel pour répondre aux exigences de ses applications et aborder les problématiques de traitement en temps réel en impliquant un cas d'utilisation industriel lié à l'assemblage sur un poste cobotique d'un produit manufacturé.
Dans un premier temps, nous avons réalisé un état de l'art sur la collaboration humain-robot, les jeux de données de HAR et les méthodes associées. Cette étude a mis en évidence le manque des jeux de données de HAR dans un contexte industriel et nous a amené à proposer le jeu de données d'actions humaines industrielles nommé InHARD portant sur l'assemblage sur poste cobotique. L'introduction de ce jeu de données a révélé que l'entrainement des algorithmes de HAR pouvait bénéficier de l'apport des outils de Réalité Virtuelle (RV) permettant de simuler les interactions humains robots afin de surmonter les problèmes liés à la labélisation et au manque de données. Ainsi, nous avons proposé une méthodologie couplant jumeau numérique (DT) et réalité virtuelle pour extraire un modèle numérique des humains et permettre la génération automatique de données labélisées. Cette méthodologie a été appliquée pour créer le jeu de données InHARD-DT et nous avons évalué la robustesse et la généralisation de notre méthode en entrainant l'algorithme de HAR avec les données du jumeau numérique et en validant sur des données du jumeau physique. Les résultats montrent une généralisation atteignant les 89% de précision et de F1-score, prouvant la pertinence de l'approche proposée.
Nos études sur les algorithmes d'apprentissage profond basés sur des données squelettes ont été approfondies et ont permis de proposer une nouvelle méthode utilisant les réseaux de neurones convolutionnels à graphes spatio-temporel avec une fenêtre glissante et un vote majoritaire nommé STGCN-SWMV. Cette approche permet une détection en temps réel sur des données en flux continu. Nous avons montré l'efficacité de la méthode présentée qui, en comparaison avec les méthodes de HAR de l'état de l'art, a obtenu de meilleures performances de classification sur les jeux de données OAD et UOW. Les travaux de cette thèse ouvrent différentes possibilités et applications pour améliorer la collaboration humain-robot, qui est en adéquation avec la transition de l'industrie 4.0 vers l'industrie 5.0 plaçant l'humain au cœur de l'industrie.

Title: Human Action Recognition using Deep Learning and generation of label data based on the digital twin of an industrial cobotic workstation.

Keywords: Industrial Human Action Recognition (HAR), Industry 4.0, Human-Robot Collaboration (HRC), Digital Twins (DT), Spatial-Temporal Graph Convolutional Networks (ST-GCN).

Abstract: Human Action Recognition (HAR) facilitates human-robot (HRC) interactions and collaboration within Industry 4.0. Indeed, collaborative robots are increasingly present in our daily lives and induce an increasingly close interaction between humans and robots, a concept brought together in the term cobotics. In order to ensure effective collaboration, robots must be able to understand their environment and must be able to communicate about their ongoing tasks and intentions. This communication and these interactions represent a major performance and security challenge. In this context, this thesis addresses the problem of recognizing human actions in an industrial environment to meet the requirements of its applications and to address the problems of real-time processing by involving an industrial use case related to assembly on a cobotic station of a manufactured product.
First, we carried out a state of the art on human-robot collaboration, HAR datasets and associated methods. This study highlighted the lack of HAR datasets in an industrial context and led us to propose the Industrial Human Action Recognition dataset named InHARD relating to assembly on a cobotic station. The introduction of this dataset revealed that the training of HAR algorithms could benefit from the contribution of Virtual Reality (VR) tools allowing to simulate human-robot interactions in order to overcome the problems related to labeling and lack of data. Thus, we have proposed a methodology coupling Digital Twins (DT) and Virtual Reality to extract a digital model of humans and allow automatic labeled data generation. This methodology was applied to create the InHARD-DT dataset and we evaluated the robustness and generalizability of our method by training the HAR algorithm with data from the digital twin and validating on data from the physical twin. The results show a generalization reaching 89% of Accuracy and F1-score, proving the effectiveness of the proposed approach. Our studies on Deep Learning (DL) algorithms based on skeleton data have been deepened and allowed to propose a new method using Spatial-Temporal Graph Convolutional Neural Networks with a Sliding Window and a Majority Voting named STGCN-SWMV. This approach allows real-time detection on continuous data streams. We have shown the efficiency of the presented method, which, in comparison with state-of-the-art HAR methods, obtained better classification performance on the OAD and UOW datasets. The work of this thesis opens up different possibilities to improve human-robot collaboration, which is in line with the transition from industry 4.0 to industry 5.0, placing humans at the heart of industry.