Localisation en intérieur à bande ultra large (UWB) et Reconnaissance d'actions industrielles

Mickael Delamare

Normandie Université

THÈSE

Pour obtenir le diplôme de doctorat

Spécialité Informatique

Préparée au sein de l'ESIGELEC-IRSEEM et l'université de Rouen Normandie

Localisation en intérieur à bande ultra large (UWB) et reconnaissance d'actions industrielles

Présentée et soutenue par
Mickaël DELAMARE

Thèse soutenue publiquement le 23 juin 2021 devant le jury composé de		
Mme Oyunchimeg SHAGDAR	R&D Team Lead, Project manager, PhD supervisor, HDR, VEDECOM Paris	Rapporteure
M. Nacer M'SIRDI	Professeur des Universités, Université de Aix-Marseille, laboratoire LIS UMR CNRS 7020.	Rapporteur
Enjie GHORBEL	Research Associate, Interdisciplinary Centre for Security, Reliability and Trust (SnT), Université du Luxembourg	Examinatrice
Rémi BOUTTEAU	Professeur des universités, Rouen Normandie, laboratoire LITIS.	Examinateur
Adnane CABANI	Enseignant-Chercheur HDR, ESIGELEC-IRSEEM Rouen.	Examinateur, Co-Encadrant
Houcine CHAFOUK	Professeur de l'ESIGELEC, ESIGELEC-IRSEEM Rouen.	Examinateur, Directeur de thèse

Thèse dirigée par Houcine CHAFOUK et co-encadrée par Adnane CABANI, laboratoire ESIGELEC-IRSEEM

UNIVERSITÉ DE ROUEN NORMANDIE — MIIS — ESIGELEC irseem — ANRT ASSOCIATION NATIONALE RECHERCHE TECHNOLOGIE — SIATECH run your world

Dédié

Remerciements

Je tiens à remercier chaleureusement Xavier Savatier et Rémi Boutteau d'avoir accepté de diriger cette thèse, ainsi que Houcine Chafouk et Adnane Cabani pour leur reprise sur la direction en fin de thèse. Merci à eux quatre de m'avoir accompagné, formé et fait grandir durant ces trois années.

Merci à Oyunchimeg Shagdar et Naser M'sirdi qui ont accepté le travail de rapporteur.e.et qui m'ont fourni des remarques pertinentes m'ayant permis d'améliorer mon manuscrit de thèse.

Merci à mon Comité de suivi de thèse (CSI), Fabrice Duval et Enjie Ghorbel pour vos conseils tout le long de ma thèse qui m'a permis d'aller jusqu'au bout.

Merci à mes collègues du département Systèmes Embarqués de l'ESIGELEC, qui m'ont permis de m'épanouir au sein du laboratoire. Avec une mention à Vincent Vauchey, sans qui aucune donnée n'auraient pu être récupérées. Merci à mes collègues du département Informatique de l'ESIGELEC qui m'ont permis de découvrir l'enseignement et de partager ma passion de l'informatique (Fadoua Bouzbouz, Samuel Grave, Christine Roueche, Chloe Cabot et Yoan Teboul). Merci aussi à ceux qui m'ont supporté au coin café et dans leur bureau, Louis Lecrosnier, Antoine Caillot, Antoine Mauri, Romain Rossi, Matthieu Pluvinage, Redouane Khemmar, Nicolas Ragot, Benoist Decoux, Yohan Dupuis et Jean-Jacque Delarue. Petite mention à Louis qui m'a donné les clés du « bon doctorant » les premiers mois et qui a toujours été de bons conseils dans ce marathon. Merci à Isabelle Riguidel pour son suivi et son soutien durant cette thèse. Merci au CESI en particulier le laboratoire LINEACT de m'avoir accueilli à bras ouvert dans leur laboratoire afin de travailler ensemble. Petite mention à Vincent Havard, David Baudry et Mejdi Dallel pour les très bons moments de recherche ensemble sur la reconnaissance d'actions. Une mention à Fabrice Duval également pour m'avoir intégré dans les phases de test au sein du CESI pour continuer ma recherche sur la localisation en intérieur. Cette expérience montre bien que les laboratoires travaillent en synergie et ce fût une expérience très enrichissante, merci à vous.

Merci à mon Frère, Alexandre Delamare qui a participé activement aux tests réalisés en laboratoire et qui à découvert le côté recherche, ses patisseries m'ont aidé en tant que carburant. Merci à mes parents, Catherine et Sylvain Delamare, qui ont tout fait pour me soutenir dans mes choix et mes études en plus d'un soutien moral. Merci à mes amis d'ESIGELECTRONIX d'avoir accompagné moralement dans cette thèse (Simon Martel et Alexandre Espriet), avec une pensée pour Thibault Clamens pour les paris "bière" pour chaque article effectué. Petite mention pour Alexandre Terrier pour les corrections anglophones. Petite mention a Loïc Rousseau pour les retours sur les soutenances blanches réalisées, j'espère que tu es au fait sur cette thèse à force de m'avoir écouté. Merci à Antonin Piquenot qui m'a soutenu pour réaliser cette thèse, et à nos pauses guitare pour décompresser en fin de thèse. Merci à Karine Manouvrier, ma compagne, pour m'avoir supporté ;), corrigé et soutenu moralement durant cette thèse. Tu m'as aidé à me surpasser et aller au bout de cette thèse surtout dans les moments de doute pour finir sur une note très positive, heureux d'avoir partagé ces moments avec toi.

Je remercie aussi tous ceux que j'ai pu oublier.

Une pensée pour Vanessa Picard.

Table des matières

Table des figures

Liste des tableaux

Abbreviations

AHRS système de référence d'attitude et de cap

ANN réseaux neuronaux artificiels

AOA angle d'arrivé

BVH hierarchie biovision

CA-GCN réseau convolutif de graphes conscient du contexte

CNN réseaux neuronaux convolutifs

CV validation croisée

DOP dilution de la précision

DTW distorsion temporelle dynamique

ECUs unités de composant électronique

EKF filtre de Kalman étendu

FCNN réseau de neurones entièrement connectés

GAN réseaux neuronaux adversariaux génératifs

GCN réseau convolutif à graphes

GNSS système mondial de navigation

GPS géo-Positionnement par satellite

GPU processeurs graphique

HMMs modèles de Markov cachés

IA intelligence artificielle

IHM interfaces homme machine

IMU centrale de mesure inertielle

LDV ligne de vue

LF-RFID système d'identification par radiofréquence à basse fréquence

LSTM réseaux à mémoire à long et court terme

MEMS systèmes micro-electro-mécanique

MIMU unité de mesure magnetique-inertielle

NLDV non ligne de vue

PEGCN réseaux de graphes convolutifs à codage prédictif

RBM machines de Boltzmann restreintes

RFID identification par radiofréquence

RNN réseaux neuronaux récurrents

RSSI indicateur d'intensité du signal reçu

RTLS systèmes de localisation en temps réel

RTT temps de trajet aller-retour

RVB rouge vert bleue

SNN réseaux neuronaux à impulsion

ST-GCN réseaux de graphes convolutifs spatio-temporel

TDOA différence de temps d'arrivée

TOA temps d'arrivée

LF-RFID système d'identification par radiofréquence à très hautes fréquence

UWB ultra large bande

WLAN réseau local sans fil

Introduction

La robotisation, en particulier dans les usines, induit une interaction de plus en plus étroite entre l'homme et la machine, concept rassemblé dans le terme « cobotique ». Cette évolution s'accompagne d'une demande croissante pour disposer d'interfaces homme machine (IHM) basées sur une interaction naturelle tout en restant intuitive et performante. Dans ce contexte, la start-up SIAtech conçoit et développe des dispositifs innovants basés sur la perception du geste avec comme objectif de séparer l'Homme des organes de commandes des machines pour pouvoir les contrôler directement avec son corps ; l'objectif étant de fournir un contrôle intuitif et peu intrusif de toutes les machines qui nous entourent. Pour cela, il est necessaire d'avoir des capteurs précis et discrets. L'opérateur pouvant lui-même évoluer dans un bâtiment ou un atelier, la capture du geste ou la posture doit être couplée à une notion de localisation précise et absolue dans un bâtiment (atelier, entrepôt) ; cette information va nous permettre de fournir une interaction non seulement naturelle mais aussi contextuelle entre l'opérateur et un ensemble de machines.

Cette thèse a pour objectif d'explorer, expérimenter et évaluer des approches innovantes couplant trois modalités pour répondre aux besoins de sécurité et d'interactivité en industrie :

- Positionnement en intérieur basée sur la technologie radio ultra large bande (UWB).
- Perception précise d'actions industrielles grâce à l'intelligence artificielle.
- Couplage des informations de locatisation et de reconnaissance d'actions.

Contexte et motivations

L'observation et la compréhension du geste et de la posture est un sujet très étudié depuis ces dernières années. De nombreux travaux existent sur la perception du geste depuis un capteur fixe, le plus connu étant le système Kinect. Cependant toutes ces solutions restreignent les déplacements d'un opérateur et limitent les possibilités d'interaction naturelle avec la machine. Une autre approche consiste à observer le mouvement depuis un dispositif

portable. Ces solutions sont généralement basées sur des capteurs inertiels, accéléromètres et gyroscopes ; leur assemblage forme ce qui est communément appelé une centrale de mesure inertielle (IMU). Leur diffusion s'est accentuée avec l'essor des technologies systèmes micro-electro-mécanique (MEMS) qui offrent un bon compromis coût-encombrement-performances et sont bien adaptées à la mesure sur l'homme [33] [43].

Ces solutions sont par exemple à la base des dispositifs de mesure de cap et de comportement d'un robot ou un aéronef (système de référence d'attitude et de cap (AHRS)). Ces systèmes sont en fait des centrales d'attitude car ils permettent une mesure de l'orientation du capteur dans le référentiel terrestre, mesure qui peut être sensiblement améliorée grâce à des algorithmes de fusion de données les plus connus étant Magdwick [90] ou Mahony [57]. Ces méthodes sont généralement basées sur un filtrage de type filtre de Kalman étendu (EKF) [18] couplé à une représentation des angles par les quaternions [40]. Une revue de ces méthodes de filtrage et des performances que l'on peut aujourd'hui atteindre avec des capteurs MEMS bas coût peut être trouvé dans [95].

Ces méthodes de fusion permettent de tenir compte des mesures fournies par un gyroscope et un accéléromètre et de résoudre le problème que pose leur utilisation en mode *strapdown* (le référentiel du capteur n'est plus galiléen puisque lié au référentiel du corps dont on veut caractériser le mouvement). La fusion de données peut être utilisée pour compléter et corriger les informations des capteurs. Une solution bien connue consiste à coupler une IMU avec un magnétomètre, l'information d'orientation du capteur pouvant aussi être déduite d'une mesure par rapport au champ magnétique terrestre (unité de mesure magnetique-inertielle (MIMU)). Si cette approche améliore la mesure d'altitude [95] avec une erreur résiduelle de 3 à 4 degrés [81], elle fait l'hypothèse d'un champ constant ce qui la rend très sensible aux perturbations magnétiques et donc peu adaptée en environnement intérieur [95] et a fortiori pour une utilisation en milieu industriel. D'autre part, les performances ne sont pas les mêmes suivant les rotations mesurées, les angles de lacet et de tangage pouvant être retrouvés dans la mesure de l'accélération gravitationnelle ce qui n'est pas le cas de la mesure de cap où l'on constate des erreurs résiduelles entre 10 degrés et 20 degrés en environnement intérieur [18].

Dans tous les cas, ces techniques permettent bien de compenser les effets de dérive des gyroscopes et de fournir une mesure d'angle suffisamment précise pour des intéractions homme-machines mais il n'est pas possible de déterminer directemment le déplacement du capteur dans l'espace. Une contrainte technologique existe actuellement sur la mesure de manière précise (centimétrique) d'un déplacement, d'un robot, d'un objet, ou d'une main en utilisant des capteurs inertiels. Le calcul de la position dépend de la position précédente car le bruit de mesure des accéléromètres induit une dérive non bornée par l'effet de la double intégration. Si on cherche à mesurer le déplacement absolu d'un mobile par rapport à un réferentiel fixe, la seule solution est alors de faire un recalage avec un capteur fournissant une réference absolue, typiquement un géo-Positionnement par satellite (GPS) ; néanmoins cette solution reste cantonnée à une utilisation en extérieur. Une autre solution est de faire une hypothèse statistique sur les signaux mesurés ; par exemple en détectant une séquence dans laquelle la personne ne bouge plus on peut faire un recalage du type *Zero Velocity Update* [131]. Cette méthode est ainsi utilisée pour estimer le

déplacement d'une personne en détectant l'accélération produite par ses pas. D'autres travaux ont été menés pour coupler cette mesure avec un modèle biomécanique de la personne. C'est le principe des solutions de mesure de mouvement portable tels que Xsens [117]. Dans ce dernier cas, elles nécessitent l'utilisation de nombreux capteurs pour reconstruire les mouvements du modèle poly-articulé sous-jacent.

On peut chercher à corriger les erreurs de dérives des mesures inertielles par un recalage avec d'autres types de capteur tels que l'ultrason, l'infrarouge, la vision... Une liste exhaustive peut être trouvée dans l'article [92]. On peut cependant les organiser en deux grandes familles, les méthodes ne nécessitant pas d'installations sur l'infrastructure (infrastructure-free) et celles nécessitant le deploiement d'équipements dans l'environnement (infrastructure-based), comme proposé dans [4]. Il faut aussi souligner que la précison de la localisation n'est pas le seul critère pour un déploiement d'une solution de localisation en intérieur, d'autres paramètres tels que le taux de disponibilité (avaibility), la facilité de déploiement (scalability) ou encore le coût (cost) sont des facteurs importants [4] et problématiques sur des solutions recourant à des caméras ou des bornes placées dans l'infrastructure. Cependant ces solutions doivent être regardées avec attention car elles se diffusent fortement dans le milieu industriel dans le contexte de l'usine 4.0.

En intérieur, on voit ainsi se développer des solutions de géolocalisations « équivalentes » à un GPS et basées sur des systèmes radiofréquence. Ces approches sont de plus en plus investiguées avec l'utilisation de méthodes de trilatération (utilisation de l'information de distance entre le transmetteur et le récepteur, mesure connue sous l'acronyme RSSI pour *indicateur d'intensité du signal reçu (*RSSI*)*, disponible dans de nombreux standards de télécommunication) et plus récemment de multilatération (mesure à partir des différences de temps de transmission appelé *différence de temps d'arrivée (*TDoA*)* entre au moins deux transmetteurs et un récepteur ; les transmetteurs étant parfaitement synchronisés). Le fait que le système à localiser soit lui-même actif augmente considérablement la précision de la localisation. L'offre dans ce domaine est pléthorique. Une revue de ces solutions montre qu'une localisation précise autour de 30cm est possible. Cependant cette performance semble fortement dégradée lorsqu'il n'y a pas un trajet direct (ligne de vue (LDV)) entre les bornes et le recepteur à localiser. La précision réelle de ces solutions reste difficile à affirmer. En effet, elle est très liée à l'environnement dans lequel le système sera déployé donc au cas d'usage. Mais, il faut noter une forte effervescence sur ce sujet avec l'arrivée sur le marché de solutions basées sur des technologies radio UWB qui s'annoncent encore plus performantes.

En robotique mobile, cette question du recalage des mesures inertielles a aussi été étudiée. Parmi elles, les solutions basées vision sont intéressantes car avec la miniaturisation de ces capteurs et leur très faible coût, il devient possible de les intégrer dans un dispositif porté par une personne. En robotique mobile, de nombreuses méthodes ont étés developpées afin de retrouver l'orientation (*pose estimation*) [42], le déplacement(*visual odometry*) et la localisation absolue (*SLAM, place recognition*) [13] du capteur. Plusieurs inconvénients peuvent être soulignés, la nécessité que le champ de vue de la caméra ne soit pas trop occulté au risque que la méthode soit mise en défaut, la dépendance aux conditions d'éclairage de la scène et la ressource embarquée peut s'avérer importante pour

traiter les données. Cependant, on voit émerger depuis quelques années des approches très intéressantes, souvent bio-inspirées, basées sur des capteurs à très faible résolution [103] ou détectant uniquement les changements de contraste dans l'image (event-based) [48]. Dans tous les cas, si le cadre théorique est aujourd'hui bien connu pour résoudre le calcul du mouvement par mesure visuo-inertielle, aucune solution fonctionnant de manière générique n'existe actuellement. Il est donc nécessaire de travailler par cas d'usage et d'évaluer la performance et la robustesse des méthodes pour faire la bonne sélection.

Liste des publications

- Delamare, M., Boutteau, R., Savatier, X., & Iriart, N. (2019, September). Evaluation of an UWB localization system in Static/Dynamic. In International Conference on Indoor Positioning and Indoor Navigation (IPIN). ;
- Delamare, M., Boutteau, R., Savatier, X., & Iriart, N. (2020). Static and Dynamic Evaluation of an UWB Localization System for Industrial Applications. Sci, 2(1),7. (MDPI) ;
- Delamare, M., Duval, F., Boutteau, R.(2020). A New Dataset of People Flow in an Industrial Site with UWB and Motion Capture Systems . Sensors, 20 (16),4511. (MDPI) ;
- Delamare, M., Laville, C., Cabani, A., Chafouk,H.(2021, February). Graph Convolutional Networks Skeleton-Based Action Recognition for Continuous Data Stream : A Sliding Window Approach. In International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISAPP) ;
- Falla, P., Fourre, J., Vauchey, V., Deshais, A., Delamare, M., Dupuis, Y. (2021,Submitted) Study geolocation of a player on a soccer field thanks to the ultra-wide band. International Journal of Performance Analysis in Sport , [Submitted] (Taylor & Francis Online) ;
- Dallel, M., Havard, V., Delamare, M., Baudry, D. (2021, Submitted) A Sliding Window Based Approach for Online Human Action Recognition using Spatial-Temporal Graph Convolutional Neural Networks. ICCV (International Conference Computer Vision) [Submitted] ;
- Delamare, M., Cabani, A., Chafouk, H. (2021) Combination of indoor localization and action recognition by deep learning with a sliding window approach. Signal Processing : Image Communication (Elsevier) [Submitted]

Plan de la thèse

SIAtech a développé plusieurs preuves de concept à partir de capteurs inertiels afin de détecter les mouvements et posture de la main et pouvoir ainsi interagir avec une machine. Dans cette thèse, nous souhaitons améliorer les performances de mesure de mouvement de ces dispositifs en fusionnant la mesure inertielle avec deux autres

modalités, la localisation par radiofréquence UWB et les techniques de localisation et estimation de mouvement basée vision. En effet, parmi les solutions permettant la localisation d'une personne dans un bâtiment et la capture de son geste, les technologies UWB et vision sont en forte diffusion et nous apparaissent compatibles avec une solution nomade. Les capacités de calcul embarqué augmentant dans le même temps, les unités de composant électronique (ECUs) ont donc de plus en plus de capacité ce qui permettrait d'embarquer des algorithmes d'apprentissage profond (Deep Learning) en temps réel. L'objectif de cette thèse est de disposer d'une solution complète couplant des informations de localisation « absolues » ainsi que les informations « d'actions instantanées » d'une personne pour pouvoir établir avec précision la position ainsi que l'action réalisé dans le référentiel terrestre et non dans le référentiel des capteurs inertiels.

Dans un premier temps, l'idée est d'exploiter les solutions de localisation par UWB qui se diffusent sur le marché et seront des technologies clés pour l'industrie du futur. Il s'agira d'étudier les méthodes de filtrage pour, d'une part, améliorer la localisation absolue d'une personne mais aussi, en considérant les performances de localisation annoncées sur certaines technologies UWB, d'autre part, rechercher les possibilités d'amélioration de la mesure des mouvements de translation fournis par une centrale inertielle par des méthodes de filtrage bayésien non-paramétriques. Au préalable, une étude sera menée afin d'objectiver la performance de la localisation par des modules UWB sur un plan théorique mais aussi dans des conditions expérimentales représentatives des scénarios d'utilisation en milieu industriel.

Dans un second temps, l'idée est l'amélioration des performances et les conditions de sécurité sur les sites industriels qui reste un objectif clé pour la plupart des entreprises. Actuellement, l'objectif principal est de pouvoir localiser de manière dynamique les personnes et les biens sur le site. La sécurité et la réglementation de l'accès aux zones d'accès restreintes sont souvent assurées par des portes ou des barrières à badges et celles-ci posent plusieurs problèmes lorsque des personnes se trouvent dans des endroits où elles ne sont pas sensées se trouver ou même lorsque des outils ou des objets sont utilisés de manière incorrecte. En outre, l'utilisation croissante de nouveaux dispositifs exige des informations précises sur leur emplacement dans l'environnement, comme les robots mobiles ou les drones. Il devient donc essentiel de disposer des outils permettant de gérer de manière dynamique ces flux de personnes et de biens. Des solutions à bande ultra-large et de capture de mouvement seront utilisées pour identifier rapidement les personnes qui pourraient se trouver dans des zones non autorisées ou qui effectuent des tâches pour lesquelles elles n'ont pas reçu d'instructions. En plus du suivi dynamique des personnes, cela permet également de surmonter certains problèmes liés au déplacement d'objets ou d'outils dans l'atelier de production. Nous proposons un nouvel ensemble de données qui fournissent des informations précises sur les mouvements des travailleurs. Cet ensemble de données peut être utilisé pour développer de nouvelles mesures concernant l'efficacité et la sécurité des travailleurs.

Dans un troisième temps, l'idée est de présenter une nouvelle approche de la reconnaissance de l'activité humaine basée sur l'apprentissage profond. La méthode consiste en un réseau convolutif de graphiques spatio-

temporels fonctionnant en temps réel grâce à une approche à fenêtre glissante. L'architecture proposée consiste en une fenêtre fixe pour l'entrainement, la validation et le processus de test avec un réseau convolutif de graphiques spatio-temporels pour la reconnaissance d'actions basée sur un squelette. Nous évaluons notre architecture sur deux ensembles de données disponibles de reconnaissance d'actions en continu, l'ensemble de données Online Action Detection (OAD) [78] et les ensembles de données UOW Online Action 3D [135]. Cette méthode est utilisée pour la détection temporelle et la classification de la reconnaissance d'action effectuée en temps réel.

Dans un quatrième temps l'idée est de fusionner les résultats obtenus en localisation en intérieur qui nous sert de localisation absolue avec la détection d'actions en temps réel. La fusion de ces deux modalités (UWB et données squelette) permettront une localisation précise à l'intérieur d'une pièce ou d'un bâtiment industriel et de connaître les actions réalisées par l'utilisateur dans un contexte industriel. Cette fusion permettra une meilleure compréhension de l'humain au sein de l'industrie 4.0 pour la collaboration Homme-machine. Le robot pourra alors savoir à quelle étape est le travailleur afin de venir lui porter assistance au bon endroit grâce à la localisation.

CHAPITRE 1

Etat de l'art sur la localisation en intérieur & la reconnaisance d'actions

Les applications dans le contexte de l'industrie 4.0 nécessitent une localisation précise. La localisation en intérieur reste un problème ouvert. Parmi les solutions possibles, dans ce chapitre, nous verrons l'émergence des méthodes à UWB qui permettent de répondre à cette problématique. Nous établirons aussi un état de l'art sur la reconnaissance d'actions en temps réel. Nous verrons que la reconnaissance en temps réel de l'action humaine à partir de flux de données squelettes est un point central dans plusieurs applications car elle permet une coordination sans faille entre l'homme et la machine et peut être utilisée pour améliorer la sécurité du lieu de travail en vérifiant les chutes ou les situations dangereuses [107]. Cependant, il s'agit d'une tâche difficile car l'algorithme doit être capable de détecter le début et la fin de chaque action sans aucune pause entre les actions et de différencier chaque action en temps réel [78]. Il existe différentes approches, par exemple une qui consiste en deux algorithmes travaillant ensemble ; un algorithme détecte quand une action est en cours d'exécution puis l'autre différencie entre toutes les différentes actions [69]. Les deux algorithmes travaillent en parallèle, ce qui augmente le coût de calcul et diminue la précision globale. Cette méthode repose également sur des pauses entre les mouvements et n'a jamais été testée sur des actions représentatives.

1.1 Localisation en intérieur

1.1.1 Étude des différentes technologies de localisation en intérieur

Pour l'interaction homme-machine dans le contexte de l'industrie 4.0, il est nécessaire de pouvoir localiser l'opérateur dans un environnement large (plus de 20 m de portée) et avec une bonne précision (avec une précision de 0,5 m). La localisation dans un environnement intérieur sera utilisée dans l'industrie 4.0. Pour cela, le système devra être non intrusif pour l'opérateur en industrie avec un moindre coût. Sur la base de la thèse de Mautz [92, 155], il existe 13 technologies présentées dans le tableau 1 qui peuvent répondre à la localisation dans un environnement intérieur. $rappelericilescontraintescommesurlepptx$

Systèmes basés sur des caméras

Les systèmes basés sur des caméras pour les approches de localisation à l'intérieur sont utilisés de différentes manières.

La première consiste à avoir un modèle de bâtiment en 3D comme référence. Il n'est pas nécessaire de remplacer les nœuds de référence par une liste de points de référence numériques. Ces systèmes ont le potentiel d'assurer une couverture à grande échelle sans augmentation significative des coûts et ont un niveau de précision de l'ordre du décimètre [65, 74].

Le second système est l'approche dite "basée sur la vue". Elle consiste à prendre la vue actuelle d'une caméra mobile et à la comparer avec des séquences de vues précédemment capturées. Ce système est arrivé à une précision centimétrique et peut couvrir un bâtiment [58, 108].

Le troisième système est constitué de cibles codées utilisées pour l'identification de points afin de localiser une personne. Le système peut savoir où la personne se trouve avec une précision centimétrique mais ne mémorise pas la trajectoire effectuée par la personne [80].

Le quatrième système est la projection de points de référence dans l'environnement. Ce système a besoin d'une vue directe de la même surface et peut être utilisé pour le suivi et la reconstruction de la scène avec une précision millimétrique [139].

Le dernier système consiste à utiliser une ou plusieurs caméras sans référence en observant le changement de position. Ce système peut atteindre une précision sub-centimétrique et peut couvrir 30 m^2 [12].

L'utilisation d'une caméra pour notre application est limitée à sa vision. La caméra peut être recouverte de vêtements ou de poussières et donc perdre sa position, ce qui peut affecter la sûreté et la sécurité. L'UWB est meilleure dans ce cas car elle peut fonctionner lorsqu'elle est recouverte (un tissu par exemple).

Systèmes infrarouges

Les systèmes infrarouges basés sur des balises actives ou utilisant le rayonnement naturel sont principalement utilisés pour l'estimation approximative de la position ou pour détecter la présence d'une personne dans une pièce. Ils ont un niveau de précision de l'ordre du centimètre et peuvent couvrir une distance de 1 à 5 m dans des conditions statiques. Ils constituent une alternative courante aux systèmes optiques opérant dans le spectre de la lumière visible. Une précision de 4 cm a été signalée et les personnes peuvent être suivies jusqu'à une distance de 5 m [62] et avec une précision centimétrique dans un magasin de détail [5]. Les erreurs dues aux trajets multiples réduisent considérablement la précision de localisation. La technologie IR nécessite une ligne de visée entre l'émetteur et le récepteur pour fonctionner correctement. Dans l'industrie, la LDV est inattendue, c'est pourquoi le système UWB est plus intéressant, car il utilise une fréquence radio et dispose d'une large bande passante qui peut gérer la LDV.

Systèmes tactiles et polaires

Les systèmes tactiles et polaires ont une précision de l'ordre du $\mu m - mm$ et peuvent couvrir une pièce entière. La méthode du point polaire utilise une mesure de distance et une mesure angulaire à partir de la même balise pour déterminer les coordonnées d'une station proche. Les systèmes tactiles sont des instruments mécaniques de haute précision qui mesurent les positions en touchant un objet avec un pointeur calibré. On ne peut pas suivre une trajectoire entière en 3D [92] avec un système tactile. Les systèmes polaires sont vraiment plus étendus que les systèmes UWB et nécessitent une ligne de visée directe pour avoir la plus grande précision.

Systèmes basés sur les ondes sonores

Les systèmes de localisation basés sur la propagation des ondes sonores ont une précision centimétrique et peuvent couvrir 2 à 10 mètres carrés, en utilisant la technique de localisation par temps de vol. Le son étant une onde mécanique, les systèmes de positionnement utilisent l'air et les matériaux de construction comme moyens de propagation [120, 146]. Les ondes acoustiques sont affectées par la pollution sonore, ce qui signifie que dans l'industrie où plusieurs machines provoquent des bruits sonores, la précision sera affectée. Les systèmes UWB sont immunisés en raison de leur grande largeur de bande (3,5-6,5 Ghz).

Systèmes WLAN/WIFI

Les systèmes WLAN/WIFI ont une précision d'un mètre et peuvent couvrir 20 à 50 m^2. L'estimation de la distance par réseau local sans fil (WLAN) est tout à fait possible à partir de l'RSSI, temps d'arrivée (ToA), TDoA et temps de trajet aller-retour (RTT). Les récents systèmes de localisation basés sur le WiFi [142] ont atteint une précision médiane de localisation pouvant atteindre 23 cm [149]. Les systèmes Wifi sont sujets au bruit et nécessitent des algorithmes de traitement complexes. Dans notre cas, la précision de ce type de systèmes ne suffit pas pour

permettre une estimation précise de la trajectoire dans les locaux industriels en non ligne de vue (NLDV) [49, 19]. L'UWB est intéressant grâce à sa large bande passante qui le rend plus précis dans les entrepôts industriels.

Systèmes RFID

L'identification par radiofréquence (RFID) pourrait être utilisée et a un niveau de précision dm-m et peut couvrir une distance de 1 à 50 m. La plupart des systèmes RFID reposent sur la détection de proximité d'étiquettes montées en permanence pour localiser une personne. La précision d'un système RFID est directement liée à la densité de déploiement des étiquettes et aux portées de lecture. Ce système est coûteux à utiliser dans une zone étendue. Les systèmes RFID ne peuvent pas suivre une trajectoire en 3D car la plupart d'entre eux dépendent de la détection de proximité d'étiquettes montées en permanence pour localiser les lecteurs mobiles [63, 125]. Le système UWB sera plus intéressant car il nécessite moins d'étiquettes et utilise une bande passante plus large (plus de 3,5 Ghz), ce qui permet une meilleure précision.

Pseudolithes

Les pseudolithes utilisent des méthodes de localisation similaires à celles du système mondial de navigation (GNSS), mais dans des environnements intérieurs. Plusieurs difficultés telles que l'atténuation des trajets multiples, la synchronisation temporelle et la résolution des ambiguïtés ont limité ce système à quelques applications dans des environnements où le GNSS est difficile à utiliser, comme les mines à ciel ouvert [110, 46]. Il peut couvrir de 10 à 1000 m^2 et ont une précision de l'ordre du cm-dm. L'UWB est plus adapté à la gestion des trajets multiples en raison de sa large bande et promet d'être moins expansif.

Systèmes radiofréquence

D'autres systèmes de radiofréquence tels que Zigbee, Bluetooth, la télévision numérique, les réseaux cellulaires, les radars, la radio FM, les téléphones basés sur la technologie sans fil numérique améliorée ont pour le mieux une précision de plusieurs mètres et peuvent couvrir 10 à 1000 mètres carrés. Toutefois, les niveaux de performance et l'applicabilité varient considérablement en fonction de plusieurs facteurs tels que l'utilisation d'une infrastructure de référence préexistante, l'omniprésence des appareils, la portée des signaux, les niveaux de puissance [92]. Les meilleurs systèmes ont une précision de 1 m et peuvent couvrir un bâtiment. L'UWB promet d'être plus précis en situation industrielle NLDV par rapport aux autres canaux de radiofréquence car il ne sera pas affecté par la propagation par trajets multiples grâce à sa large bande à 3,5 Ghz.

Systèmes de navigation inertielle

Les systèmes de navigation inertielle sont généralement associés à des capteurs complémentaires qui fournissent des informations de localisation absolue en raison de la dérive et ont une précision de quelques mètres [92]. Les systèmes montés sur le pied peuvent utiliser une vitesse nulle lorsque le pied est en phase d'appui et ont donc une dérive plus faible et peuvent améliorer la précision en dessous de 5% [116] de la distance parcourue. Par rapport aux IMU montées sur d'autres parties du corps [133], dont la dérive est généralement plus importante, près de 6% des mouvements irréguliers sont mal classés mais sont toujours similaires aux IMUs à pied.

Systèmes basés sur le champ magnétique

Le système basé sur le champ magnétique a une précision centimétrique et peut couvrir une zone de 10 m [11]. Les différentes approches vont des systèmes dédiés à des fins médicales utilisant un champ magnétique artificiel quasi-statique de moins de 1 m^3 volume fonctionnant au niveau de précision du millimètre. Dans les environnements intérieurs, avec la même approche, nous pouvons avoir une précision de quelques mètres couvrant les allées de stockage et un bâtiment [141, 3] mais nous pouvons être perturbés par le champ magnétique induit par les moteurs électriques à l'intérieur des bâtiments industriels.

Systèmes d'infrastructure

Les systèmes d'infrastructure sont des technologies qui utilisent l'infrastructure existante du bâtiment ou qui intègrent une infrastructure supplémentaire dans les matériaux de construction, comme le positionnement des lignes électriques, les dalles de sol, les lampes fluorescentes ou les câbles d'alimentation qui fuient, comme décrit dans [92]. Ces systèmes ont un niveau de précision de l'ordre du cm au m et peuvent couvrir une zone de bâtiment. Les systèmes développés peuvent être dissimulés aux utilisateurs dans les structures du bâtiment. L'utilisation de l'UWB sera indépendante de l'infrastructure et peut être installée dans toute infrastructure réelle.

1.1.2 Systèmes à bande ultra large (UWB)

Nous avons décidé de nous concentrer sur la technologie UWB car elle peut couvrir une superficie de 50 m et avoir une précision de 10 à 30 cm, comme indiqué dans [92]. L'UWB est moins chère que d'autres technologies et peut être précise même dans des conditions de NLDV. Elle est beaucoup plus résistante à la propagation par trajets multiples (multipath), qui est en télécommunications sans fil la propagation sur plusieurs chemins d'un signal radio reçue sur une antenne, car elle transmet de courtes impulsions sur une large bande passante (3,5-6,5 GHz) [142]. L'UWB est une technologie sans fil développée pour transférer des données à haut débit sur de très courtes distances. De plus, elle a la capacité de transporter des signaux à travers des portes et d'autres obstacles qui ont tendance à réfléchir les signaux avec une bande passante plus limitée et des niveaux de puissance plus élevés [35].

TABLE 1.1 – Indoor positioning technologies as described in [92].

Technologie	**Exactitude typique**	**Couverture classique**	**Principe de mesure classique**
Cameras	0.1 mm-dm	1–10	Codedmarkers
Infrared	cm-m	1–5 m	IR camera
Tactile & Polar Systems	um-mm	3–2000 m	Distance & angular measurement
Sound	cm	2–10 m	Multilateration
WLAN/WIFI	m	20–50 m	Fingerprinting
RFID	dm-m	1–50 m	Cell of Origin
Ultra WideBand	cm-m	1–50 m	ToA, TDoA
High Sensitive GNSS	10 m	'global'	Assisted GNSS
Pseudolites	cm-dm	10–1000 m	carrier phase ranges
Other Radio Frequencies	m	10–1000 m	Fingerprinting, cell of Origin, RSSI, RTT
Inertial Navigation	1%	10–100 m	WLAN RSSI, GNSS
Magnetic Systems	mm-cm	1–20 m	DCfield, coils , AC magnetic field
Infrastructure Systems	cm-m	building	Powerlines, floor tiles, fluorescent lamps

TABLE 1.2 – Comparison of Ultra WideBand (UWB) systems manufacturers.

UWB Technologies	**Decawave** [1]	**BlinkSight** [2]	**IIDRE** [3]	**BeSpoon** [4]
Accuracy	X-Y 10 cm	10 cm	10–30 cm	10 cm
Detecting Range	290 m	200+ m	150 m	600 m
Bandwith	3.5 Ghz–6.5 Ghz	7–8.5 Ghz	3.5 Ghz–6.5 Ghz	3.5–4.5 Ghz
Current Consumption	30 mA	30 mA	Max 30 mA	30 mA
Interface	Spi control	wifi	Bluetooth USB RS232	USB
Cost (euros)	300 (kit)	N/A	1140	650

[1] `https://www.decawave.com`; [2] `https://www.blinksight.com`; [3] `https://iidre.com`; [4] `https://bespoon.com`.

Nous avons comparé quatre fiches techniques de fabricants dans le tableau 1.2 pour choisir le système matériel le plus approprié. Nous voyons également dans [119, 60] que Decawave est le plus précis. C'est pourquoi nous avons choisi le système Decawave pour nos expériences.

Syberfeldt et al. [134] ont proposé un examen des techniques et systèmes existants pour localiser les opérateurs dans une usine intelligente. Dans cette comparaison, on peut voir que l'UWB a une grande précision par rapport aux autres systèmes de localisation en intérieur et a un coût moyen pour l'industrie. Alarifi et al. [4] ont établi une analyse des forces, faiblesses, opportunités et menaces (SWOT) des systèmes UWB. Les principaux avantages des systèmes UWB sont une faible consommation d'énergie et la capacité à pénétrer différents types de matériaux. Les basses fréquences du spectre UWB ont de grandes longueurs d'onde. Cela permet d'utiliser cette technologie avec succès dans les communications hors ligne de vue. La capacité à pénétrer différents matériaux a fait de la technologie UWB un choix intéressant pour les applications systèmes de localisation en temps réel (RTLS) et radar. L'UWB ressemble à du bruit, car le signal a une très faible puissance et s'étend uniformément sur un large spectre. C'est un point très positif car il est à la fois difficile d'intercepter un signal UWB mais permet également de le faire coexister en toute sécurité avec d'autres technologies radio. Le point faible de l'UWB est la synchronisation. Le récepteur se synchronise avec le signal reçu, ce qu'un système de positionnement basé sur les temps de

propagation (ToA, TDoA) exige en synchronisant différents nœuds [134]. Cet article [4] conforte notre choix de l'UWB comme le meilleur système de localisation en intérieur.

Kulikov et al. [67] ont présenté un système pour la navigation 2D précise à l'intérieur des véhicules. Il intègre un capteur inertiel à faible coût et un système Ultra WideBand (Decawave). Ses futurs travaux consisteront à le comparer à un système cinématographique de vérité terrain et à voir comment la précision est améliorée. Li et al. [73] ont proposé de fusionner l'IMU et l'UWB avec un EKF et de tester les performances de deux algorithmes : l'algorithme "EKF vanilla" (EKF originel) sans fusion et l'EKF fusion avec des résultats de 0,16 m d'erreur moyenne avec l'EKF fusion et de 0,30 m d'erreur moyenne avec l'EKF originel. La fusion de l'Ultra WideBand et de l'IMU est efficace. Antonio Ramón Jiménez Ruiz et al. [119, 60] ont comparé trois systèmes UWB disponibles dans le commerce, tels que Ubisense, Bespoon et Decawave, dans des conditions de LDV et NLDV et ont montré que les performances de Decawave sont légèrement meilleures. Ils testent la performance de positionnement d'un objet en mouvement en utilisant un filtre à particules. L'évaluation de ces deux systèmes [60] montre que le système Decawave est légèrement meilleur que le Bespoon. La portée maximale détectée pour Decawave est de 103,4 m, ce qui correspond à notre besoin. La position de l'objet en mouvement est obtenue en calculant la position moyenne pondérée de toutes les particules. Ils utilisent 61 points de test dans leur laboratoire. Ces trois comparaisons montrent que le Decawave est légèrement meilleur que les autres systèmes à bande ultra-large.

Gharat et al. [50] ont présenté un système de localisation intérieur appelé système d'identification par radiofréquence à basse fréquence (LF-RFID) et l'ont comparé à un système système d'identification par radiofréquence à très hautes fréquence (LF-RFID) et un système UWB. L'erreur de l'Ultra WideBand est de 0,58 m dans les conditions NLDV, ce qui est mieux que le LF-RFID et l'UHF-RFID. Ils ont utilisé une méthodologie de positionnement avec 352 points de test.

L'UWB est utilisé récemment comme réference pour réaliser du fingerprinting [32]. Le fingerprinting, ou « prise d'empreinte » est une technique probabiliste visant à identifier un utilisateur de façon unique.

Dotlic et al [34] ont fourni une description des principes de l'estimation de l'angle d'arrivé (AOA) du DW1000 de Decawave et ont prouvé que l'AOA a de petites erreurs d'estimation. Marcelo et al [123] ont fourni un système de localisation dans un robot mobile utilisant la technologie UWB. Les résultats obtenus indiquent que ce système peut fournir une grande précision, inférieure à 25 cm dans les conditions NLDV pour localiser un robot mobile en 2D. Aryan et al [6] ont effectué une évaluation statique dans des environnements intérieurs et ont également testé différents matériaux pouvant perturber le système UWB. Ces résultats ont servi de base à l'établissement d'un environnement industriel interne pour nos tests que nous verrons au chapitre 2.

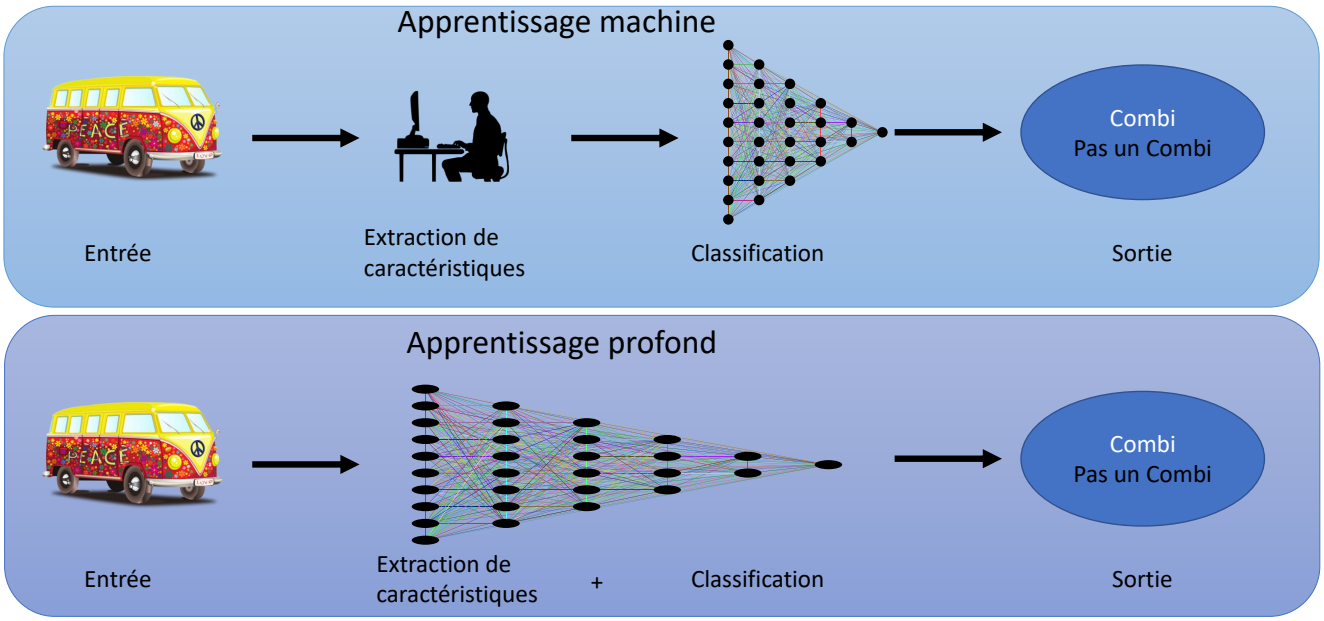


FIGURE 1.1 – Difference entre l'apprentissage machine et l'apprentissage profond

1.2 Principes de base des réseaux de neurones artificiels

Le domaine de l'intelligence artificielle est né lorsqu'une machine a pu automatiser une tâche qui nécessite l'intelligence humaine. De plus, dans l'apprentissage machine, la capacité d'apprendre par expérience et d'acquérir des compétences sans intervention humaine est primordiale. L'apprentissage profond est une sous-catégorie de l'apprentissage machine, dans lequel les réseaux neuronaux artificiels sont inspirés des neurones du cerveau humain ou des perceptrons qui apprennent essentiellement à partir d'une grande quantité de données. L'apprentissage profond est une classe d'algorithmes d'apprentissage machine qui utilisent plusieurs couches pour extraire progressivement des caractéristiques de niveau supérieur à partir de données brutes. Par exemple, dans le traitement d'images, les couches inférieures peuvent identifier des bords, tandis que les couches supérieures peuvent identifier des éléments significatifs pour l'homme tels que des chiffres/lettres ou des visages.

Dans la vie réelle, nous apprenons de l'expérience de la même manière que les réseaux neuronaux profonds exécutent chaque tâche de manière itérative, ce qui permet d'apprendre de chaque itération et d'obtenir un meilleur résultat. On parle d'apprentissage profond parce que les réseaux neuronaux fonctionnent sur une multitude de couches profondes qui stimulent l'apprentissage de l'algorithme, représenté sur la figure 1.1. À l'ère moderne, une quantité massive de données est générée. En plus de la création de données, l'algorithme DNN (Deep Neural Network) gagne en puissance de calcul, l'intelligence artificielle (IA) en tant que service est en pleine expansion.

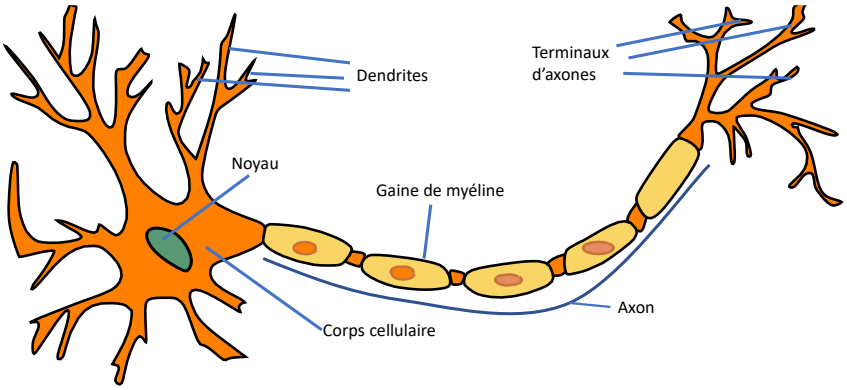


FIGURE 1.2 – Schéma d'un neurone biologique

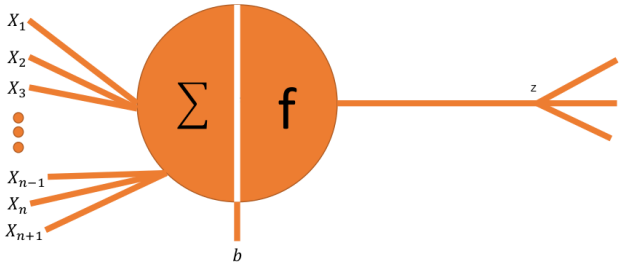


FIGURE 1.3 – Schéma d'un neurone artificiel

1.2.1 Le neurone formel

Les réseaux de neurones sont inspirés du cerveau [52] et sont traduits informatiquement. Il y a donc des neurones, des activations et des interconnexions qui ressemblent au fonctionnement du cerveau humain.

Un neurone artificiel par exemple, est directement inspiré d'un neurone biologique (figure 1.2), on constate des similarités avec celui-ci 1.3. Les entrées (information) sont représentées par des variables X_1 jusqu'à X_n sous formes de vecteurs, les dentrites pour un neurone biologique. Le noyau d'un neurone artificiel est representé par la somme des entrées qui en constituent une fonction f. Z représente les terminaux d'axones qui correspond à la sortie. Des paramètres w et b influencent le fonctionnement du neurone. La figure 1.4 présente le passage

d'information à travers un réseau de neurone formel où

$$Z = f(\langle w, x \rangle + b) \tag{1.1}$$

cette équation modélise le passage d'information du noyau d'un neurone jusqu'à sa dentrite. Les entrées sont variables. Les paramètres sont fixés par construction du modèle. La sortie est calculée en fonction des entrées et des paramètres. Pour résumer l'information d'entrée est passe dans un neurone qui est ensuite envoyé vers sa dentrite. Ce comportement est similaire pour chaque neurone du réseau [29].

1.2.2 Le réseau de neurones en couches

Un seul neurone ne permet pas de répondre à des problèmes complexes, il faudra pour cela réaliser un réseau complet. Pour constituer ce type de réseau il faut plusieurs couches, ceci est représenté par les lignes verticales (rond orange) de la figure 1.4. Pour passer de la première couche à la seconde couche notre réseau de neurones va devoir s'activer et envoyer des informations à la couche suivante. Pour cela, il existe plusieurs méthodes, c'est ce qu'on appelle fonction d'activation.

1.2.3 Fonction d'activation

Les fonctions d'activation sont des fonctions qui décident, compte tenu des entrées dans le nœud, quelle doit être la sortie du nœud. Comme c'est la fonction d'activation qui décide de la sortie réelle, nous appelons souvent les sorties d'une couche ses "activations".

L'une des fonctions d'activation les plus simples est la fonction de Heaviside (appelée marche d'escalier). Cette fonction renvoie un 0 si la combinaison linéaire est inférieure à 0. Elle renvoie un 1 si la combinaison linéaire est positive ou égale à zéro. Sa représentation est la suivante :

$$f(h) = \begin{cases} 0 \quad si \quad h \quad < \quad 0 \\ 1 \quad si \quad h \quad \geq \quad 0 \end{cases} \tag{1.2}$$

L'unité de sortie renvoie le résultat de $f(h)$, où h est l'entrée de l'unité de sortie. Il existe d'autres fonctions d'activation, comme la ReLU, TanH, Softmax. Elles sont à choisir selon le problème à résoudre.

Les modèles d'apprentissage profond fonctionnent par couches et un modèle typique comporte au moins trois couches. Chaque couche accepte les informations de la précédente et les transmet à la suivante comme on peut le voir sur la Figure 1.4.

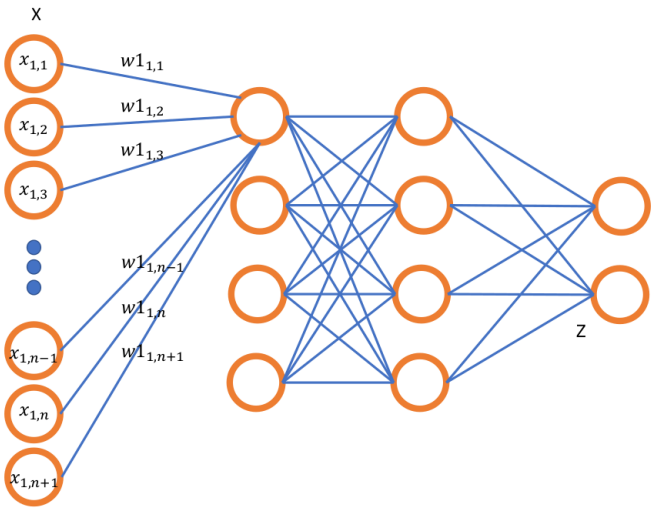


FIGURE 1.4 – Schéma d'un réseau de neurones artificiel

1.2.4 Les Poids et Biais

Lorsque les données d'entrée arrivent dans un neurone, elles sont multipliées par une valeur de poids qui est attribuée à cette entrée particulière. Par exemple, dans la figure 1.4 les poids sont représentés par $w1$ qui correspondent aux poids de la première couche, ce sont les mêmes que l'on retrouvait dans l'équation d'un neurone formel 1.1.

Les poids et les biais sont les paramètres à apprendre des modèles d'apprentissage profond. Le biais est représenté par b dans l'équation 1.4.

1.2.5 Phase d'entrainement

Pour que notre modèle de réseau de neurones puisse paramétrer ses poids et ses biais, il va devoir apprendre sur des jeux de données représentatifs du problème à résoudre. Pour un ensemble d'images de véhicles comprenant des combi-vans, nous avons besoin que le modèle distingue automatiquement celles qui sont liées à un combi-van des autres véhicles. Les modèles d'apprentissage automatique, comme les humains, doivent apprendre à différencier les deux catégories d'images en observant des images de combi-van et des images de vehicule différent. En conséquence, ils comprennent automatiquement des modèles qui décrivent mieux chaque catégorie. C'est ce que nous appelons la phase d'apprentissage.

Concrètement, un modèle est une combinaison pondérée de certaines entrées (images, parties d'images ou autres modèles). Par conséquent, la phase d'apprentissage n'est rien d'autre que la phase durant laquelle nous estimons les poids (également appelés paramètres) du modèle.

Lorsque nous parlons d'estimation, nous parlons d'une fonction objective que nous devons optimiser. Cette

fonction doit être construite de manière à refléter au mieux les performances de la phase d'apprentissage. Lorsqu'il s'agit de tâches de prédiction, cette fonction objective est généralement appelée fonction de perte et mesure le coût engendré par des prédictions incorrectes. Lorsque le modèle prédit quelque chose qui est très proche de la sortie réelle, la fonction de perte est très faible, et vice-versa.

En présence de données d'entrée, nous calculons une perte empirique (perte d'entropie croisée binaire en cas de classification et perte d'erreur quadratique moyenne en cas de régression) qui mesure la perte totale sur l'ensemble de notre jeu de données. Il existe ainsi plusieurs fonctions d'optimisation que nous verrons dans les prochains chapitres relatifs à la reconnaissance d'actions.

1.2.6 Propagation de l'information

En propageant les valeurs de la première couche (la couche d'entrée) à travers toutes les fonctions mathématiques représentées par chaque nœud, le réseau produit une valeur. Ce processus est appelé "forward pass".

1.2.7 Descente de gradient

La descente de gradient est un algorithme d'optimisation utilisé pour trouver les valeurs des paramètres (coefficients) d'une fonction (f) qui minimise une fonction de coût (cost). La descente de gradient est utilisée de préférence lorsque les paramètres ne peuvent pas être calculés analytiquement (par exemple en utilisant l'algèbre linéaire) et doivent être recherchés par un algorithme d'optimisation. La descente de gradient est utilisée pour trouver l'erreur minimale en minimisant une fonction "cost". Dans l'exemple de l'université (expliqué dans la section **??** sur les réseaux de neurones), les lignes correctes pour diviser l'ensemble de données sont déjà définies. Comme nous le savons, les poids sont ajustés pendant le processus d'entrainement. L'ajustement du poids permettra à chaque neurone de diviser correctement l'ensemble de données. Nous voulons que le réseau fasse des prédictions aussi proches que possible des valeurs réelles. Pour mesurer cela, nous avons besoin d'une mesure de l'erreur des prédictions, l'erreur. Une mesure commune est la somme des erreurs au carré (SSE) :

$$E = \frac{1}{2} \sum_{\mu} \sum_{j} [y_j^{\mu} - \hat{y}_j^{\mu}]^2 \tag{1.3}$$

où $\hat{y}$ est la prédiction et y est la valeur réelle, et la somme de toutes les unités de sortie j et une autre somme de tous les points de données μ L'SSE (Sum Squared Error) en français la somme des erreurs au carré est un bon choix pour plusieurs raisons. Le carré garantit que l'erreur est toujours positive et que les plus grandes erreurs sont plus pénalisées que les plus petites. En outre, il rend les calculs plus faciles, toujours un plus. La sortie d'un réseau de neurones, la prédiction, dépend des poids.

$$\hat{y}_j^\mu = f(\sum_i w_{ij} x_i^\mu) \tag{1.4}$$

et, par conséquent, l'erreur dépend des poids

$$E = \frac{1}{2} \sum_\mu \sum_j [y_j^\mu - f(\sum_i w_{ij} x_i^\mu)]^2 \tag{1.5}$$

Nous voulons que l'erreur de prédiction du réseau soit aussi faible que possible et les poids sont les boutons que nous pouvons utiliser pour y parvenir. Notre objectif est de trouver des poids qui minimisent l'erreur quadratique E. Pour ce faire, nous utilisons généralement un réseau de neurones avec descente de gradient. Avec la descente de gradient, nous faisons plusieurs petits pas vers notre objectif. Dans ce cas, nous voulons modifier les poids par étapes qui réduisent l'erreur. Pour poursuivre l'analogie, l'erreur est notre montagne et nous voulons atteindre le fond. Comme le chemin le plus rapide pour descendre d'une montagne est dans la direction la plus raide, les pas effectués doivent être dans la direction qui minimise le plus l'erreur.

1.2.8 Rétropropagation

Dans l'apprentissage machine, la rétropropagation [51] est un algorithme largement utilisé pour l'entrainement de réseaux neuronaux à action anticipée (FeedForward Network). Des généralisations de la rétropropagation existent pour d'autres réseaux neuronaux artificiels (ANN), et pour des fonctions en général. Ces classes d'algorithmes sont toutes appelées génériquement "rétropropagation" [140]. En ajustant un réseau neuronal, la rétropropagation calcule le gradient de la fonction de perte par rapport aux poids du réseau pour un seul exemple d'entrée-sortie.

Les pondérations commencent par des valeurs aléatoires, et à mesure que le réseau neuronal en apprend davantage sur le type de données d'entrée qui conduit à l'acceptation d'un étudiant dans une université (exemple ci-dessus), le réseau ajuste les pondérations en fonction de toute erreur de catégorisation que les pondérations précédentes ont entraîné. C'est ce que l'on appelle l'entrainement du réseau neuronal. Une fois le réseau formé, nous pouvons l'utiliser pour prédire la sortie pour une entrée similaire.

1.2.9 Erreurs

Pour savoir si notre modèle peut résoudre notre problème, nous avons besoin de connaître les erreurs de celui-ci. L'erreur quadratique moyenne est l'une des fonctions d'erreur les plus populaires. Il s'agit d'une version modifiée de l'erreur quadratique moyenne.

$$SSE = \sum_i (cible^{(i)} - sortie^i)^2$$
$$MSE = \frac{1}{n} \times SSE \tag{1.6}$$

Ou on peut écrire MSE comme :

$$E = \frac{1}{2m} \sum_\mu (y^\mu - \hat{y}^\mu)^2 \tag{1.7}$$

Il existe d'autres indicateurs que nous verrons dans les prochains chapitres.

1.3 Les principaux algorithmes de Deep Learning

Dans cette section, la majeure partie de l'algorithme d'apprentissage profond sera couvert. Il s'agira d'une explication concise point par point des algorithmes d'apprentissage profond supervisés et non supervisés les plus populaires.

1.3.1 Réseau neuronal entièrement connecté

Chaque neurone d'un réseau de neurones entièrement connectés (FCNN) est connecté à tous les autres neurones de la couche suivante. Pour la même raison, ces couches sont connues sous le nom de couches denses. Comme chaque neurone est connecté à n'importe quel autre neurone, ces couches sont très coûteuses à calculer.

Lorsque le nombre de neurones dans les couches est faible, cet algorithme est préférable ; dans le cas contraire, l'exécution des opérations nécessitera beaucoup de puissance de calcul et de temps. En raison de sa connectivité complète, il peut également contribuer à un surajustement [122] [82].

1.3.2 Réseau neuronal convolutif (CNN)

Les réseaux neuronaux convolutifs (CNN) sont une forme de réseau neuronal conçu pour travailler avec des images, des photos et des vidéos, par exemple. Ils sont donc utilisés pour diverses tâches de traitement d'images, telles que la reconnaissance optique de caractères (OCR), la localisation d'objets, etc. Les CNN peuvent également être utilisés pour reconnaître des vidéos, du texte et du son. Chaque pixel d'une image est une valeur que le réseau neuronal peut traiter. Une image 128 lignes par 128 colonnes, par exemple, signifie que l'image contient 16384 pixels ou attributs. Elle sera transmise au réseau neuronal sous la forme d'un vecteur de taille 16384. Il existe trois canaux (un pour chacun - rouge, bleu et vert) dans les images en couleur. Dans ce cas, la même image aura une résolution de 128x128x3 pixels.

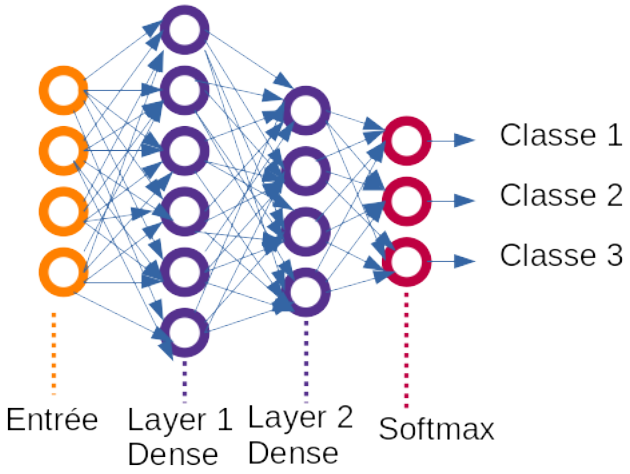


FIGURE 1.5 – Architecture d'un réseau de neurone entièrement connectée

Dans les couches du CNN, il y a une hiérarchie [87]. La première couche tente d'extraire les caractéristiques brutes des photographies, telles que les bords horizontaux et verticaux. À partir des caractéristiques extraites par la première couche, les secondes couches extraient encore plus d'informations. Les couches suivantes vont approfondir les détails pour définir des aspects particuliers d'une image, comme les cheveux, les yeux et le nez. La dernière couche qualifie l'image d'entrée d'humaine, d'animal de compagnie, de chien, etc.

Il y a trois terminologies importantes à savoir pour les réseaux de convolution :

— Convolutions : Les convolutions sont la somme des propriétés des éléments des deux matrices. La première matrice fait partie des données d'entrée, tandis que la deuxième matrice est un filtre qui extrait les caractéristiques de l'image.

— Pooling Layers (Couches de mise en commun) : Les couches de mise en commun sont responsables de l'agrégation de la fonctionnalité dérivée. En général, ces couches calculent une statistique agrégée (max, moyenne, etc.) et rendent le réseau insensible aux transformations locales. Un exemple de Max-pooling est illustré à la figure 1.7.

— Feature Maps (Cartes de caractéristiques) : Une carte de caractéristiques est essentiellement un filtre de poids qui est appris par l'entrainement. Le champ réceptif de chaque neurone est la zone de l'entrée qu'il examine. Une carte de caractéristiques est un ensemble de ces neurones avec les mêmes poids qui examinent

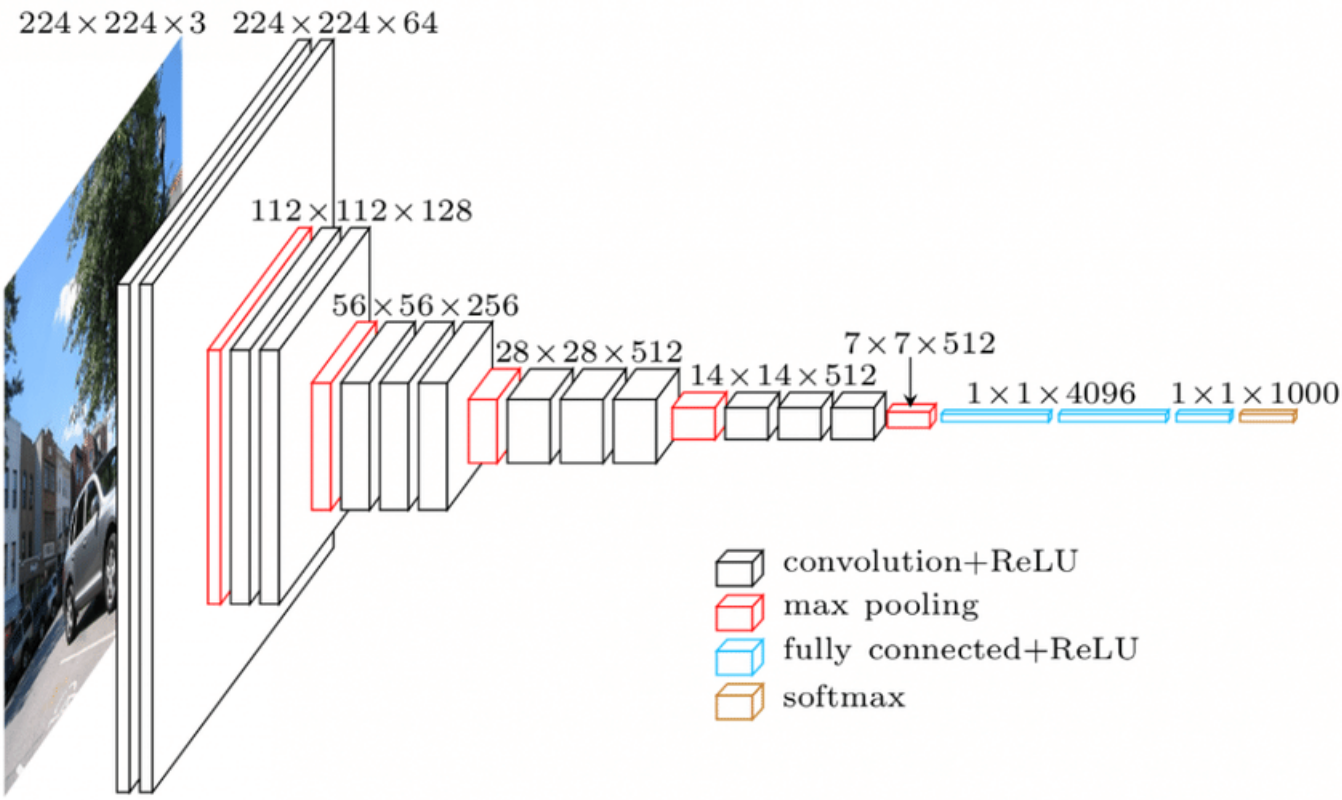


FIGURE 1.6 – Architecture d'un réseau neuronal convolutif particulier VGG-16 [87] [130]

diverses régions de l'image. Les neurones d'une carte de caractéristiques tentent tous d'éliminer la même caractéristique de diverses parties de l'image.

1.3.3 Réseaux neuronaux récurrents (RNN)

Les réseaux neuronaux récurrents (RNN) sont des réseaux neuronaux structurés pour traiter des résultats séquentiels. Les données qui font référence à des données précédentes, telles que du texte (séquence de textes, phrases, etc.) ou des vidéos (séquence d'images), de la voix, etc. sont appelées données séquentielles. Il est essentiel de tenir compte de la relation entre ces entités séquentielles ; sinon, il est inutile de mélanger l'ensemble

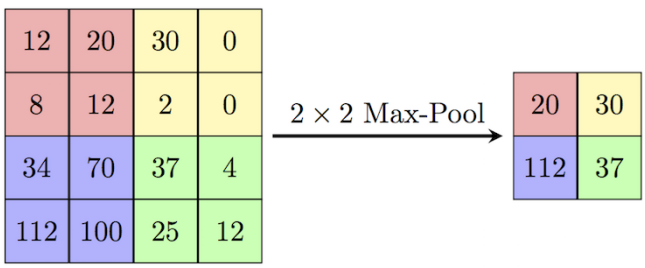


FIGURE 1.7 – Exemple Max-Pooling, Source : `https://computersciencewiki.org/index.php/Max-pooling_/_Pooling`

de l'essai et de tenter d'y trouver un sens. Ces entités séquentielles ont été prévues pour être traitées par des RNN. Un bon exemple d'utilisation des RNN est la génération automatique de sous-titres dans YouTube. Ce n'est rien d'autre que la reconnaissance automatique de la parole mise en œuvre à l'aide de RNN [17]. La principale différence entre les réseaux neuronaux normaux et les réseaux neuronaux récurrents est que les données d'entrée circulent selon deux dimensions : le temps (sur la longueur de la séquence pour en extraire des caractéristiques) et la profondeur (couches neuronales normales). Il existe différents types de RNN et leur structure change en conséquence. Les architectures sont :

— Plusieurs à un RNN : Dans cette architecture, l'entrée alimentant le réseau est une séquence et la sortie est une entité unique. Cette architecture est utilisée pour résoudre des problèmes tels que la classification des sentiments ou pour prédire le score de sentiment des données d'entrée (problème de régression). Elle peut également être utilisée pour classer les vidéos dans certaines catégories.

— Plusieurs à plusieurs RNN : - Dans cette architecture, l'entrée et la sortie sont toutes deux des séquences. Elle peut être classée en fonction de la longueur de l'entrée et de la sortie.

  • Même longueur : Le réseau produit une sortie à chaque pas de temps. Il y a une correspondance biunivoque entre l'entrée et la sortie à chaque pas de temps. Cette architecture peut être utilisée comme un marqueur de parties du discours où chaque mot de la séquence en entrée est marqué avec sa partie du discours en sortie à chaque pas de temps.

  • Longueur différente : Dans ce cas, la longueur de l'entrée n'est pas égale à la longueur de la sortie. L'une des utilisations de cette architecture est la traduction des langues. La longueur d'une phrase en anglais peut être différente de celle de la phrase correspondante en hindi.

— RNN un à plusieurs : - L'entrée est ici une entité unique tandis que la sortie est une séquence. Ces types de réseaux neuronaux sont utilisés pour des tâches telles que la génération de musique, d'images, etc.

— Un à un RNN : - Il s'agit d'un réseau neuronal traditionnel dans lequel l'entrée et la sortie sont des entités uniques.

Réseaux de mémoire à long et court terme (LSTM)

Un réseau de mémoire à long et court terme est un RNN spécifique mais l'un des inconvénients des réseaux neuronaux récurrents est le problème de la disparition du gradient. Ce problème est rencontré lors de la formation de réseaux neuronaux à l'aide de méthodes d'apprentissage basées sur le gradient, telles que la descente de gradient stochastique et la rétropropagation. Les gradients de la fonction d'activation sont responsables de la mise à jour des poids des réseaux.

Ils deviennent si petits que les poids des réseaux neuronaux sont à peine modifiés. Cela empêche les réseaux neuronaux de se former. Les RNN sont confrontés à ce problème lorsqu'ils ont des difficultés à apprendre les

dépendances à long terme.

Les réseaux à mémoire à long et court terme (LSTM) ont été conçus pour résoudre ce problème. Les LSTM sont constitués d'une unité de mémoire qui peut stocker les informations qui sont pertinentes pour les informations précédentes. Les unités récurrentes à déclenchement (GRU) sont également une variante des RNN qui aident à résoudre les problèmes de gradient évanescent.

Les deux réseaux ont un mécanisme de déclenchement pour résoudre ce problème. Les GRU (Gated Recurrent Unit) utilisent moins de paramètres de formation et donc moins de mémoire que les LSTM. Cela permet aux GRU de s'entraîner plus rapidement, mais les LSTM fournissent des résultats plus précis lorsque les séquences d'entrée sont longues.

1.3.4 Réseaux adversariaux génératifs (GAN)

Les réseaux neuronaux adversariaux génératifs (GAN) sont des algorithmes d'apprentissage non supervisés qui découvrent et apprennent automatiquement les modèles à partir des données. Après avoir appris ces modèles, il génère de nouvelles données en sortie qui ont les mêmes caractéristiques que les données d'entrée. Il crée un modèle qui est divisé en deux sous-modèles : le générateur et le discriminateur.

Le modèle générateur essaie de générer de nouvelles images à partir de l'entrée, tandis que le rôle du modèle discriminateur est de classer si les données sont une image réelle de l'ensemble de données ou des images générées artificiellement (images du modèle généré).

Le modèle discriminant agit généralement comme un classificateur binaire sous la forme d'un réseau neuronal convolutif. À chaque itération, les deux modèles essaient d'améliorer leurs résultats, car le but du modèle générateur est de tromper le modèle discriminateur dans l'identification de l'image et le but du discriminateur est d'identifier correctement les fausses images.

1.3.5 Machine de Boltzmann restreinte (RBM)

Les machines de Boltzmann restreintes (RBM) sont des réseaux neuronaux non déterministes dotés de capacités génératives et qui apprennent la distribution de probabilité sur l'entrée. Il s'agit d'une forme limitée de la machine de Boltzmann, limitée en termes d'interconnexions entre les nœuds de la couche.

Ils ne comportent que deux couches, à savoir la couche visible et la couche cachée. Il n'y a pas de couche de sortie dans la RBM et les couches sont entièrement connectées les unes aux autres. Les RBM sont désormais solennellement utilisés car ils ont été remplacés par les GAN. Plusieurs RBM peuvent également être assemblés pour créer un nouveau réseau qui peut être réglé à l'aide de la descente de gradient et de la rétropropagation comme les autres réseaux neuronaux. Ces réseaux sont appelés réseaux à croyance profonde.

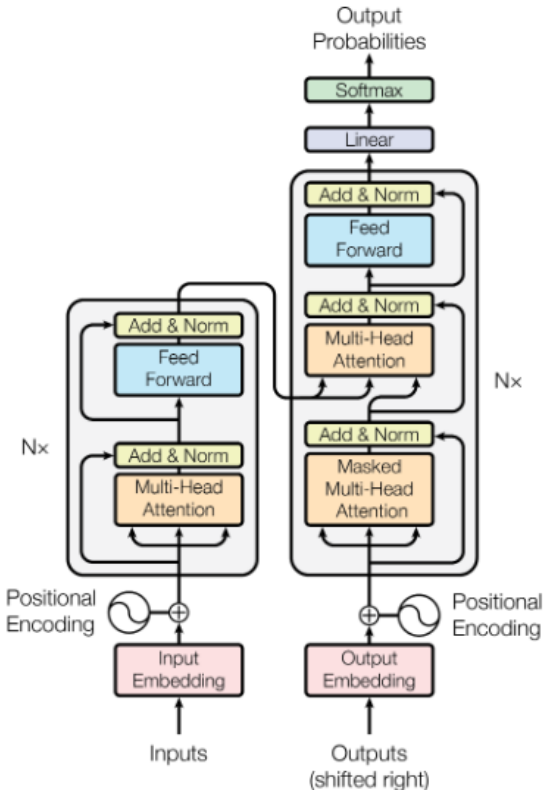


FIGURE 1.8 – Architecture d'un réseau de neurones transformateurs [143]

1.3.6 Transformers

Les transformateurs sont un type d'architecture de réseau neuronal qui a été conçu pour la traduction automatique neuronale. Ils impliquent un mécanisme d'attention qui se concentre sur une partie des informations fournies au réseau. Il comprend deux parties : Les encodeurs et les décodeurs. La partie gauche de la figure est l'encodeur, et la partie droite est le décodeur. L'encodeur et le décodeur peuvent être constitués de plusieurs modules qui peuvent être empilés les uns sur les autres. La même chose est représentée par Nx dans la figure. La fonction de chaque couche de l'encodeur est de déterminer quelles parties de l'entrée sont pertinentes les unes par rapport aux autres, ce que l'on appelle l'encodage. Ces codages sont ensuite transmis à la couche de codage suivante comme entrées. La couche décodeur prend ces codages et les traite pour générer la séquence de sortie. Le mécanisme d'attention pèse l'importance de chaque autre entrée et extrait des informations de ces relations pour prédire la séquence de sortie. Les couches de codage et de décodage se composent également de couches d'anticipation qui sont utilisées pour le traitement ultérieur des sorties.

Cette section a présenté brièvement le domaine de l'apprentissage profond, les composants utilisés dans les réseaux neuronaux, l'idée des algorithmes d'apprentissage profond, les hypothèses faites pour simplifier les réseaux neuronaux, etc. Cet article [126] fournit une liste restreinte d'algorithmes d'apprentissage profond, car il existe de nombreux algorithmes différents qui sont constamment créés pour surmonter les limites des algorithmes existants.

Les algorithmes d'apprentissage profond ont révolutionné la façon de traiter les vidéos, les images, les textes, etc. et ils peuvent être facilement mis en œuvre en important les modules nécessaires.

1.4 Reconnaissance d'actions

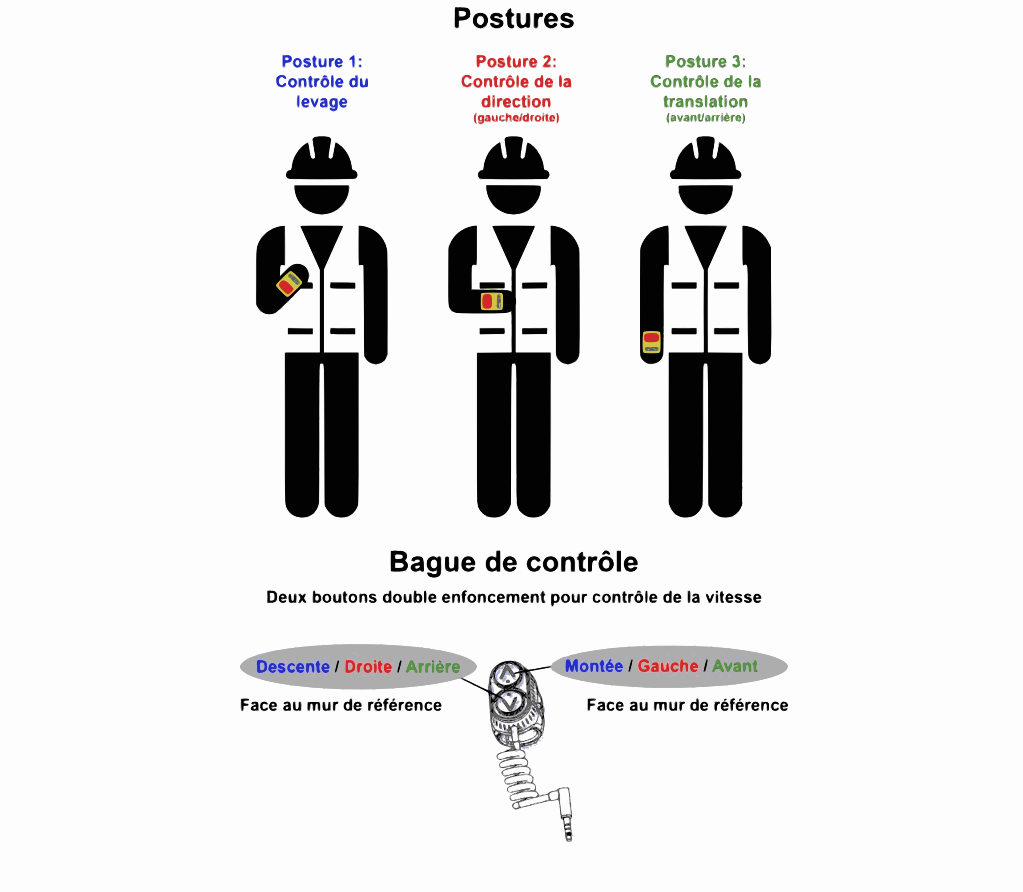
1.4.1 Reconnaissance de l'action continue en ligne

La reconnaissance en temps réel de l'action humaine à partir de flux de données squelettes est un point central dans plusieurs applications car elle permet une coordination transparente entre l'homme et la machine et peut être utilisée pour améliorer la sécurité du lieu de travail en vérifiant les chutes ou les situations dangereuses [107]. L'algorithme doit détecter le début et la fin de chaque action en temps réel ce qui rend la tâche plus complexe pour l'algorithme [78]. Une approche courante de ce problème est d'utiliser deux algorithmes travaillant ensemble, un algorithme de détection d'actions en cours d'exécution et un algorithme de différenciation entre toutes les actions [69], cela nécessite néanmoins un travail parallèle des deux algorithmes, ce qui augmente le coût de calcul et diminue la précision globale.

La détection d'actions en ligne s'est rapidement développée ces dernières années. Elle vise à localiser le segment d'action avec des séquences d'actions partiellement observées, qui peuvent être appliquées en temps réel. Les algorithmes de détection d'actions sont divisés en deux sous-parties : la détection d'action hors ligne et la détection d'action en ligne. Pour la détection d'action hors ligne, et pour nous, c'est une détection segmentée entrainé en hors ligne puis une détection en ligne avec ce pré-entrainement. La détection d'action en ligne, pour nous, signifie la détection en temps réel et pour notre méthode, c'est aussi la phase d'entrainement utilisant des données non segmentées mais un flux de données continu. La plupart des travaux [71] [106] ne considèrent que les images rouge vert bleue (RVB) comme entrée car les données RVB reflètent directement les informations originales, telles que la posture humaine, la pose des objets, etc. Cependant, les données d'entrée RVB nécessitent toujours une énorme quantité de calculs, qui sont généralement accélérés par le GPU. Un autre type d'entrée est basé sur les données du squelette, qui nécessitent moins de calculs et peuvent être extraites de la vidéo ou directement fournies par l'unité de mesure à inertie (IMU) [113] qui est plus pratique pour les travailleurs qui pourraient se déplacer beaucoup et se retrouver hors du champ de vision de la caméra.

Une des approches courantes de la reconnaissance des actions humaines se concentre sur la classification de différentes actions sur des flux de données segmentés [99], où le classificateur est fourni avec des actions individuelles manuellement segmentées et doit seulement identifier quelle action est effectuée. Les modèles de Markov cachés sont souvent utilisés à cette fin [136] mais ils sont lents et nécessitent un large ensemble de données. Un autre algorithme souvent utilisé est le filtrage particulaire. Cependant, une des principales limites de cette approche est le fait que pour l'appliquer à un scénario en temps réel, un autre algorithme est nécessaire pour segmenter le

FIGURE 1.9 – Schema Posture



flux de données [159]. L'ajout d'un autre algorithme augmente la complexité du système, ajoute une autre source d'erreurs potentielles et est coûteux en termes de calcul.

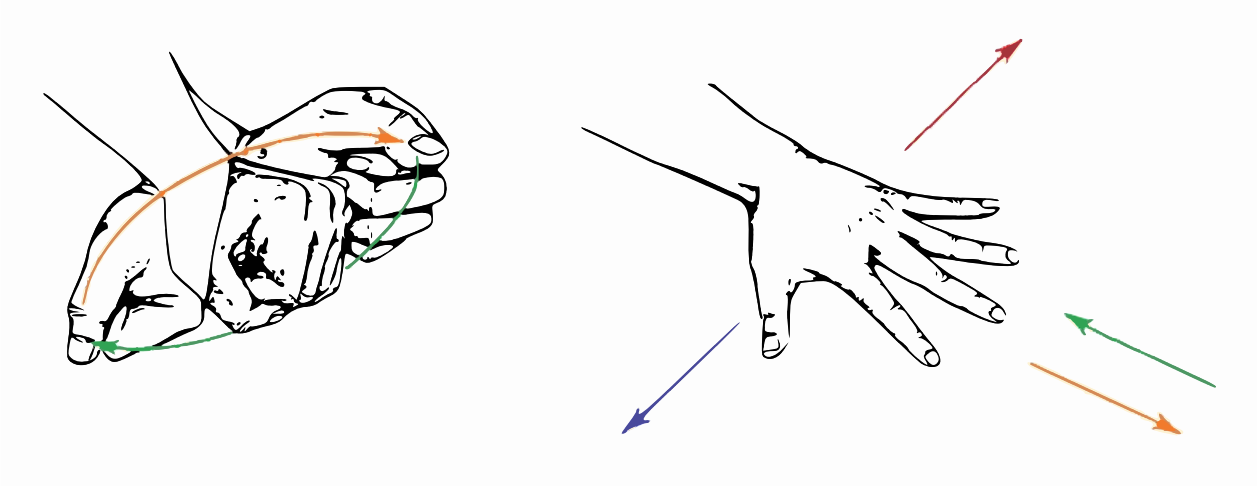
1.4.2 Mouvement des gestes segmenté

Un ensemble de caractéristiques ad hoc basé sur la position et l'orientation des doigts est calculé et introduit dans un classificateur SVM multi-classes afin de reconnaître les gestes effectués. Un ensemble de caractéristiques est également extrait de la profondeur calculée à partir du Kinect et combiné avec les caractéristiques des sauts afin d'améliorer les performances de reconnaissance [91].

La distorsion temporelle dynamique (DTW) est un algorithme de correspondance de modèles et est l'une des techniques utilisées dans la reconnaissance des gestes. Pour reconnaître un geste, le DTW déforme une séquence temporelle de positions d'articulations en séquences temporelles de référence et produit une valeur de similarité. Une méthode de DTW pondéré, qui permet d'optimiser un rapport en discriminant certaines articulations a été proposée [16].

La plupart des approches existantes pour la reconnaissance des actions basées sur le squelette modélisent

FIGURE 1.10 – Schema de la Gestuelle



l'évolution spatio-temporelle des actions en fonction de caractéristiques fabriquées à la main. En tant que banque de filtres à adaptation hiérarchique, l'algorithme de "Convolutionnal Neural Network" (CNN) est très performant dans l'apprentissage de la reconnaissance d'actions. Une architecture hiérarchique de bout en bout pour la reconnaissance d'actions basée sur un squelette avec CNN a été proposée [36].

Compte tenu des données de mouvement de nombreux chirurgiens de différents niveaux de compétence, un algorithme d'apprentissage d'un dictionnaire pour chaque geste ainsi qu'une grammaire de modèles de Markov cachés (HMM) décrivent les transitions entre les différents gestes [136].

Un système de reconnaissance des gestes rapide et très précis, basé sur la mémoire à long terme et à court terme (LSTM) et CNN, qui est formé pour traiter les séquences d'entrée des positions et des vitesses des mains en 3D acquises par des capteurs infrarouges pour la reconnaissance des gestes dynamiques de la main, a été proposé [104].

Une reconnaissance des gestes en temps réel utilisant des données sensorielles 3D telles que l'unité de mesure inertielle fournie par un gant de mouvement, et une lumière CNN a été proposée [31].

Ils ont montré que les gestes de mouvement segmentés sont vraiment précis pour chaque segmentation, mais dans la plupart des cas, ils nécessitent au moins deux algorithmes pour la reconnaissance d'actions en flux continu et la précision est dégradée en raison de la détection ou de la segmentation en ligne. C'est pourquoi nous nous concentrons sur les mouvements en ligne avec un seul algorithme.

Un modèle de Markov caché continu pour la reconnaissance d'actions en ligne basé sur la vision a été introduit [38]. Le système est capable de reconnaître des gestes dynamiques en mode indépendant de la personne et du contexte et fonctionne plusieurs fois plus vite que le temps réel. Une méthode basée sur les modèles de Markov cachés (HMMS) présentée pour la modélisation et la reconnaissance de trajectoires de gestes dynamiques a été proposée [145]. La méthode proposée permet d'obtenir une reconnaissance en ligne automatique des gestes de la

main et peut rejeter les gestes atypiques. Une version en ligne de l'algorithme de maximisation des attentes (EM) pour les HMM a été présentée [102]. L'algorithme en ligne est capable de traiter des environnements dynamiques, c'est-à-dire lorsque les statistiques des données observées changent avec le temps. La méthode HMM est la première méthode utilisée pour la reconnaissance des actions en ligne.

Une approche qui ajuste dynamiquement la taille de la fenêtre et le décalage à chaque étape a été proposée [68]. Une limitation est que les instances dépendent de la précision des capteurs. Si les capteurs ne captent pas un changement significatif dans l'environnement, le système ne détecte pas le changement d'état et ne crée pas l'instance correspondante. C'est pourquoi nous avons choisi d'utiliser une fenêtre fixe avec des données squelettes dans notre contexte.

Une nouvelle méthode d'extraction de caractéristiques efficace et performante qui utilise une approche de correspondance dynamique pour construire un vecteur de caractéristiques pour chaque trame et améliore la sensibilité aux caractéristiques des différents gestes et diminue la sensibilité aux caractéristiques des gestes au sein d'une même classe est proposée [159].

Une méthode de reconnaissance de gestes en temps réel à partir d'un flux de squelette bruité, tels que ceux extraits des capteurs de profondeur Kinect, a été introduite [97]. Cette méthode peut améliorer l'entrée du réseau convolutif à graphes (GCN).

Un système de reconnaissance dynamique en ligne des gestes de la main avec une caméra RGB-D, qui peut reconnaître automatiquement les gestes de la main sur un fond compliqué, est présenté [150].

Les auteurs [101] utilisent la classification temporelle connectionniste pour entraîner le réseau à prédire les étiquettes de classe à partir d'actions en cours dans des flux d'entrée non segmentés. Cette méthode fournit une détection et une classification en ligne des actions dynamiques de la main avec un réseau neuronal convolutif 3d récurrent.

Une approche de traitement des données par fenêtre glissante est utilisée [88], leur algorithme est adapté au traitement des données de flux pour la reconnaissance naturelle des actions de la main.

1.4.3 Étude sur les réseaux convolutionnels graphiques spatio-temporels

Les auteurs [148] fournissent une taxonomie qui regroupe les réseaux neuronaux de graphes en quatre catégories : réseaux neuronaux de graphes récurrents, réseaux neuronaux de graphes convolutifs, graphauto-codeurs et réseaux neuronaux de graphes spatio-temporels.

Un réseau convolutif de graphes conscient du contexte (CA-GCN) est proposé [158]. Outre le calcul de la convolution des graphes localisés, CA-GCN considère un terme de contexte pour chaque sommet en intégrant les informations de tous les autres sommets. Les dépendances à long terme entre les joints sont donc naturellement intégrées dans les informations de contexte, ce qui élimine ensuite la nécessité d'empiler plusieurs couches pour

élargir le champ de réception et simplifie grandement le réseau.

Les auteurs [160] montrent que lorsque la précision de l'extraction du squelette est suffisamment élevée, le modèle a une grande robustesse et précision. On utilise les données hierarchie biovision (BVH) qui sont des données squelettiques, en utilisant l'algorithme ST-GCN. Cet article montre que les données du squelette et le ST-GCN sont efficaces et renforcent notre choix en utilisant le ST-GCN.

La méthode de reconnaissance d'action basée sur le squelette bruyant et basée sur des réseaux de graphes convolutifs avec codage prédictif pour l'espace latent, appelés réseaux de graphes convolutifs à codage prédictif (PEGCN) est présentée [154]. Le PeGCN augmente la flexibilité du GCN et est mieux adapté aux tâches de reconnaissance d'action utilisant des caractéristiques squelettiques. Ce document renforce également notre choix en utilisant des données squelettiques.

Un nouveau réseau LSTM (Attention Enhanced Graph Convolutional LSTM) pour la reconnaissance de l'action humaine à partir de données squelettes est proposé [127]. L'AGC-LSTM proposé peut non seulement capturer des caractéristiques discriminantes dans la configuration spatiale et la dynamique temporelle mais aussi explorer la relation de co-occurrence entre les domaines spatiaux et temporels.

Un nouveau réseau convolutionnel de graphes adaptatifs à deux flux (2s-AGCN) pour la reconnaissance d'action basée sur le squelette est présenté [124] . Cette approche basée sur les données augmente la flexibilité du réseau convolutionnel de graphes et est plus adaptée à la tâche de reconnaissance d'actions.

Le ST-GCN pour la reconnaissance d'actions basée sur le squelette est étendu par l'introduction de deux nouveaux modules, à savoir, le GraphVertex Feature Encoder (GVFE) apprend les caractéristiques des sommets appropriés pour la reconnaissance d'actions en codant les données brutes du squelette dans un nouvel espace de caractéristiques. Et le DH-TCN (Dilated Hierarchical Temporal Convolutional Network) est capable de capturer les dépendances temporelles à court et long terme en utilisant un réseau convolutionnel hiérarchique dilaté [109].

Les études précédentes sont principalement basées sur des graphiques à squelette fixe, ne capturant que les dépendances physiques locales entre les articulations, ce qui peut manquer les corrélations implicites entre les articulations. Pour capturer des dépendances plus riches, [75] introduit une structure d'encodeur-décodeur, appelée module d'inférence A-link, pour capturer des dépendances latentes spécifiques à l'action, c'est-à-dire des liens actionnels, directement à partir des actions. Ils étendent également les graphes squelettes existants pour représenter les dépendances d'ordre supérieur, c'est-à-dire les liens structurels.

Un nouveau modèle de squelettes dynamiques appelé réseaux de graphes convolutifs spatio-temporel (ST-GCN) est proposé [151], qui va au-delà des limites des méthodes précédentes en apprenant automatiquement les modèles spatiaux et temporels à partir des données. Cette formulation conduit non seulement à une plus grande puissance d'expression mais aussi à une plus grande capacité de généralisation.

Le réseau de convolution Graph est une approche récente et démontre son efficacité, comme mentionné ci-dessus, pour détecter des actions avec des données squelettes.

Nous avons choisi cet algorithme ST-GCN [151] pour fournir une reconnaissance d'action avec une approche de fenêtres glissantes afin de pouvoir détecter une action de mouvement en temps réel et de se concentrer uniquement sur les fenêtres glissantes au lieu d'améliorer le ST-GCN. D'autres auteurs ont amélioré ce ST-GCN comme [75],[109] ou [160]

1.4.4 Étude sur l'approche des fenêtres glissantes

Les auteurs [68] ont utilisé une approche différente en utilisant des fenêtres dynamiques basées sur les événements. Leur approche ajuste dynamiquement la taille de la fenêtre et le décalage à chaque étape. Des expériences avec des ensembles de données publiques montrent que leur méthode, qui utilise des modèles plus simples, est capable de reconnaître les activités avec précision.

Le chevauchement des fenêtres glissantes dans les systèmes de reconnaissance de l'activité humaine (HAR) est associé aux limites sous-jacentes de la validation croisée (CV) en fonction du sujet. Lorsque la CV indépendante du sujet est utilisée, les fenêtres glissantes qui se chevauchent n'améliorent pas la performance des systèmes HAR mais nécessitent néanmoins beaucoup plus de ressources que les fenêtres qui ne se chevauchent pas [22]. Nous choisissons les fenêtres glissantes qui se chevauchent dans notre contexte pour avoir plus de données à caractériser, et l'algorithme peut être mis à jour plus fréquemment.

La détermination de l'heure de début et de fin de l'activité augmente la charge de calcul, de sorte que les résultats de la reconnaissance seront retardés [89]. C'est pourquoi nous avons choisi la méthode de la fenêtre glissante pour reconnaître les actions avec leur bruit ambiant sans début et fin reconnus dans notre algorithme. Cela permettrait de réduire la charge de calcul, et pourrait être déployé dans un système embarqué.

Au cours de la dernière décennie, l'étude théorique du modèle de la fenêtre glissante a été développée pour faire progresser les applications à très grande entrée et à sortie sensible au temps. Dans certaines situations pratiques, l'entrée peut être considérée comme une séquence ordonnée, et il est utile de limiter les calculs aux parties récentes de l'entrée, [21] a introduit le modèle de fenêtre glissante qui suppose que l'entrée est un flux d'éléments de données et divise les éléments de données en deux catégories : les éléments actifs et les éléments périmés. Nous désignons le flux D par une séquence d'éléments $\{P_i\}_{i=1}^{m}$ où $p_i \in \mathbb{N}$. Il est important de noter que m est incrémenté pour chaque nouvel arrivant. Un seau $B(x, y) = \{p_i, i \in [x, y]\}$ est l'ensemble de tous les éléments du flux entre p_x et p_y, inclusivement.

1.5 Perspectives : Spiking Neural Networks

La première génération de réseaux neuronaux était basée sur des neurones qui étaient des portes à seuil ou des perceptrons. Ces neurones n'avaient pas de fonction d'activation non linéaire, mais leur sortie était soit 1 soit

0, selon que la somme pondérée de leurs entrées était supérieure ou inférieure à un seuil "t". Une caractéristique importante de ces réseaux de neurones est qu'ils ne peuvent produire que des sorties numériques. Cependant, ils sont capables de modéliser toutes les fonctions booléennes et sont donc universels pour le calcul avec des entrées et des sorties numériques.

La deuxième génération de réseaux neuronaux est constituée de neurones qui appliquent une fonction continue non linéaire à la somme des entrées pondérées et produisent donc un ensemble continu de valeurs de sortie possibles. Les fonctions sigmoidales et tanh sont des exemples de fonctions d'activation. Un exemple typique de réseaux neuronaux de deuxième génération est le réseau neuronal de feed forward, dont nous avons parlé précédemment. D'autres exemples de réseaux de neurones de deuxième génération sont les réseaux de neurones récurrents et les réseaux de neurones récursifs, que nous aborderons dans les colonnes suivantes.

Les réseaux neuronaux de type feed forward sont ceux dans lesquels les signaux ne se déplacent que dans une seule direction, c'est-à-dire de l'entrée à la sortie. En d'autres termes, la sortie d'une couche antérieure de l'architecture n'alimente que les couches ultérieures et non l'inverse. La sortie d'une couche n'est pas transmise à elle-même ni à aucune autre couche antérieure dans les réseaux neuronaux à alimentation directe. Les réseaux neuronaux récurrents peuvent avoir la sortie d'une couche qui est réinjectée à travers le réseau dans une couche antérieure, au fil du temps.

Cela signifie essentiellement que le réseau neuronal comporte des boucles ou des cycles dans lesquels il permet à la sortie calculée à la couche "i" au moment "t" d'être renvoyée à une couche antérieure au moment "t+1". Cela permet aux réseaux neuronaux récurrents d'obtenir une forme de mémoire à court terme. Alors que les réseaux neuronaux de rétroaction sont utilisés pour la classification de la valeur d'entrée uniquement, les réseaux neuronaux récurrents sont généralement utilisés pour la classification des séquences et des séries chronologiques.

Les réseaux neuronaux de deuxième génération peuvent également prendre en charge des sorties numériques en appliquant un seuil à la sortie, et peuvent donc modéliser toute fonction booléenne arbitraire. Ils sont également universels pour les calculs analogiques, puisque toute fonction continue peut être approchée raisonnablement bien au moyen d'un réseau neuronal de deuxième génération avec une seule couche cachée elle-même.

Aujourd'hui, ces réseaux neuronaux de deuxième génération sont largement utilisés dans diverses applications telles que le traitement de la parole, la reconnaissance d'images et le traitement du langage naturel. Ces réseaux neuronaux sont généralement formés à l'aide d'algorithmes de descente de gradient et de propagation en retour. Afin d'appliquer les réseaux neuronaux à différentes tâches, l'un des défis importants est de savoir comment former efficacement et rapidement ces réseaux neuronaux. En fait, l'une des raisons de la popularité croissante des réseaux neuronaux artificiels est qu'ils peuvent être formés efficacement à l'aide de processeurs graphique (GPU).

La troisième génération de réseaux neuronaux utilise des neurones à impulsion (Spiking Neural Network). Ces réseaux de neurones modélisent plus étroitement l'activité des neurones biologiques par rapport aux neurones de première et deuxième générations. Ces réseaux de neurones à impulsions peuvent également être utilisés pour le

traitement de l'information, tout comme la deuxième génération de réseaux de neurones. Bien que leur efficacité ait été décrite en théorie, ils n'ont pas encore été largement utilisés dans des applications de la vie réelle en raison de leurs exigences élevées en matière de calcul. Nous n'aborderons donc pas la question des réseaux de neurones à impulsions même si cela reste une piste très intéréssante, si ce n'est pour souligner que de nombreuses recherches sont menées dans ce domaine. Par exemple, Qualcomm a récemment annoncé que les processeurs Qualcomm Zeroth sont basés sur le principe du dopage des réseaux de neurones, ce qui est un exemple d'architectures neuromorphes, autrement appelées architectures informatiques d'inspiration biologique.

[144] et al. introduit les réseaux de neurones à impulsions, les neurones biologiques utilisent des augmentations brèves et soudaines de la tension pour envoyer des informations. Ces signaux sont plus communément appelés potentiels d'action, pointes ou impulsions. Des recherches neurologiques récentes ont montré que les neurones encodent des informations dans le rythme des impulsions uniques, et pas seulement dans leur fréquence de déclenchement moyenne. Les réseaux à impulsions sont plus puissants que leurs prédécesseurs non impulsionnels, car ils peuvent coder des informations temporelles dans leurs signaux, mais ils ont également recours à des règles différentes et biologiquement plus plausibles pour la plasticité synaptique. Ce sont des réseaux de neurones qui sont les plus proches du cerveau biologiquement.

Tavanaei et al. [137] montre que la fonction de transfert des neurones à impulsions est généralement non-différenciable, ce qui empêche l'utilisation de la rétropropagation. Tavanaei et al. ont passé en revue les méthodes supervisées et non supervisées récentes pour former des réseaux neuronaux à impulsion (SNN) profonds, et les ont comparés en termes de précision et de coût de calcul. L'image qui se dégage est que les SNNs sont toujours à la traîne des ANNs en termes de précision, mais l'écart se réduit, et peut même disparaître pour certaines tâches, tandis que les SNNs nécessitent généralement beaucoup moins d'opérations et sont les meilleurs candidats pour traiter les données spatio-temporelles. C'est pour cela que dans cette thése nous n'utiliserons pas encore de SNN mais ce papier très récent montre l'interêt et le chalenge qui reste dans le monde de la recherche. Dans quelques années nous devrions arriver à entrainer ces algorithmes avec un même taux de précisions, ce qui rendra nos systèmes embarqués moins gourmands en ressources.

Lei Deng et al. [30] presente une reflexion sur la comparaison des performances entre les réseaux SNN et ANN. Ils ont donc répondu aux question de la "charge de travail" idéales pour un SNN et comment évaluer un SNN en ayant du sens. Il a été évalué que :

- Pour les charges de travail simples orientées ANN (par exemple, MNIST), le modèle 3 (SNN renforcé) est un meilleur choix avec une précision acceptable et un coût de calcul moindre (sans multiplications coûteuses et avec un peu plus d'additions).
- Sur les charges de travail plus complexes orientées ANN (par exemple CIFAR10), le modèle-1 (ANN naturel) est préféré pour maintenir la précision du modèle. Bien qu'il ait besoin de multiplications la quantité d'additions est la plus faible.

- Pour les charges de travail orientées SNN, le modèle 6 (SNN naturel) est le meilleur choix, car il offre à la fois une meilleure précision et un coût de décompilation plus faible.

Cet article est aussi très récent et prouve ce fort interêt, de plus les algoritmes SNN ont été testé sur les jeux de données très connues de l'état de l'art tel que MNIST ou CIFAR10. Une nouvelle librairie, performante permet de tester ces algorithmes bio-inspiré [111]. [39]

1.6 Conclusion

Dans ce chapitre, nous avons vu un état de l'art exhaustif de toutes les technologies qui répondent à la question de la localisation en intérieur. Dans le cadre d'une localisation en intérieur en milieu très perturbé, notre choix s'est arrété sur l'UWB. De nos jours, ce dispositif reste un choix abordable pour un industriel, nous verrons par la suite dans le chapitre 2 et le chapitre 3 une évaluation statique et dynamique en Ligne de Vue ainsi qu'une évaluation avec plusieurs personnes dans un milieu en Non Ligne de Vue industriel.

L'état de l'art sur les algorithmes de Deep Learning nous montre que le choix le plus pertinent pour la reconnaissance de gestes avec des centrales inertielles est d'utiliser des données squelettes et de surcroît utiliser des algorithmes de réseau à graph convolutionnel. Nous verrons une approche à fenêtre glissante qui permet à ce type d'algorithme d'acquérir une meilleure précision et robustesse pour des données en flux continu temps réel.

Dans le dernier chapitre nous utiliserons l'UWB et son filtrage amélioré en milieu perturbé pour garder une information de localisation et de détecter en même temps les actions industrielles d'une personne.

CHAPITRE 2

Évaluation statique et dynamique d'un système de localisation UWB pour les applications industrielles

De nombreuses applications dans le contexte de l'industrie 4.0 nécessitent une localisation précise. Cependant, la localisation en intérieur reste un problème ouvert, surtout dans des environnements complexes comme les environnements industriels. Ces dernières années, nous avons assisté à l'émergence de systèmes de localisation à bande ultra-large (UWB). L'objectif de ce chapitre est de montrer les performances d'un système UWB pour estimer la position d'une personne se déplaçant dans un environnement intérieur. Pour ce faire, nous avons mis en place un protocole expérimental pour évaluer la précision du système UWB à la fois statiquement et dynamiquement. Le système UWB est comparé à une vérité de terrain obtenue par un système de capture de mouvements avec une précision millimétrique.

2.1 Mise en place expérimentale et évaluation

Pour évaluer le système UWB, quatre points d'ancrage doivent d'abord être placés dans la pièce pour le positionnement interne. Nous alignons l'ancre avec la mesure laser. Pour être aussi précis que possible, le point d'ancrage doit être placé dans un rectangle. Une ancre est choisie comme référence (initialisation à x = 0 et y = 0), et nous devons obtenir la position de chaque ancre en fonction de l'ancre d'initialisation comme indiqué dans la figure 2.1a.

2.1.1 Installation expérimentale

Comme décrit dans [6], quatre ancres doivent d'abord être placées dans un rectangle pour la localisation intérieure. Nous calibrons les ancres avec les mesures laser afin d'être aussi précis que possible. Les ancres sont placées respectivement à (-2m ;5.03m),(3.10m ;5.03m),(-2m ;1m),(3.10m ;1m).

Nous avons obtenu la position de chaque ancre avec un système de capture de mouvement VICON [94]. La zone d'essai de la figure 2.1a se trouve dans le laboratoire avec une condition de LDV (Ligne De Vue) et dans un environnement industriel avec une structure métallique, des robots et une porte métallique à proximité de la zone d'essai, comme le montre la figure 2.1c. Le Tag est monté sur un support et placé sur un chariot en bois comme illustré sur la figure 2.1b avec une hauteur de 0,7 m pour vérifier la trajectoire en 3D dans la zone intérieure du système UWB. Pour limiter les interférences, nous avons placé le tag UWB sur un chariot en bois poussé par une personne suffisamment éloignée de l'étiquette. Nous attendons une précision de 10 cm ou 15 cm avec la technologie UWB dans des conditions de LDV. Nous utilisons l'algorithme standard de télémétrie bidirectionnelle (TWR) intégré par Decawave [1].

2.1.2 Méthode de comparaison

Nous examinons le comportement de l'UWB dans un état statique et évaluons la précision et l'exactitude dans un état de ligne de visée. Nous utilisons le système Vicon comme vérité de terrain en raison de sa capacité à mesurer une position avec une précision millimétrique [94].

L'erreur quadratique moyenne (RMSE) est calculée comme suit :

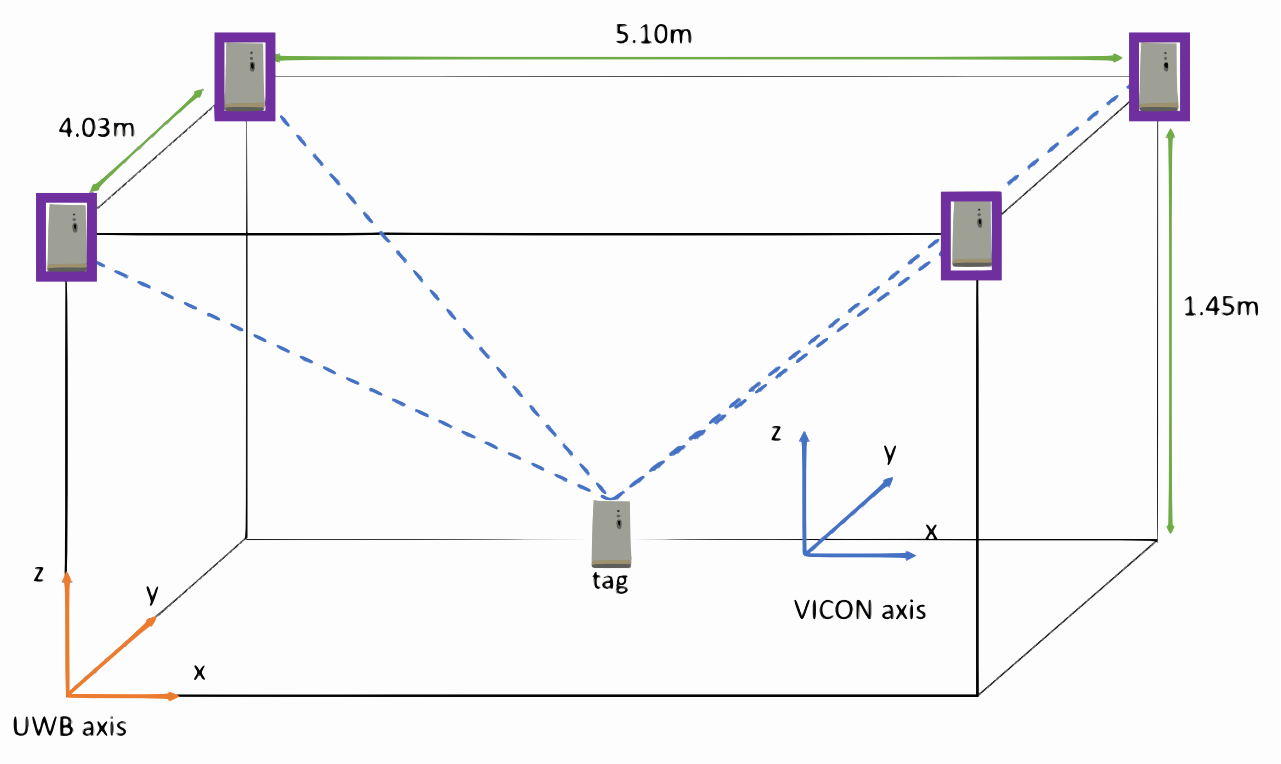
$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(\theta_i - \hat{\theta}_i)^2} \tag{2.1}$$

et représente la différence moyenne entre les valeurs de vérité de terrain du système Vicon, écrites θ_i, et la valeur estimée du système UWB représentée par $\hat{\theta}_i$. $\hat{\theta}_i$ est la valeur du point i^{th}.

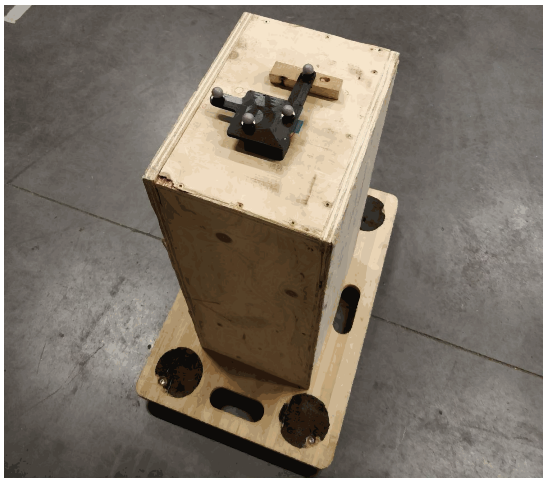
La fonction de distribution cumulative empirique de l'erreur (eCDF) est calculée comme suit :

$$eCDF = F_X(x) = \int_{-\infty}^{x} f_X(t)dt. \tag{2.2}$$

L'équation (2.2) permet de constater la distribution des valeurs autour de la moyenne, avec $f_X(t)$ l'intégrale de sa densité de probabilité fonction.



a Quatre ancres statiques en violet sont placées dans un rectangle. Le tag est placé dans la zone des ancres en LDV.



b Support avec les marqueurs Vicon sur un chariot en bois.

c Placement des ancres dans le laboratoire.

FIGURE 2.1 – Notre installation UWB.

2.1.3 Calibration

Dans cette étude, nous avons deux ensembles de données qui contiennent les positions (X, Y, Z) les données du capteur à bande ultralarge et les données du système VICON. Chaque ensemble de données étant dans son propre repère de référence, nous exprimerons donc les données du système Ultra WideBand dans le repère de référence VICON. Avant de comparer les données de l'Ultra WideBand avec les données VICON, une transformation rigide entre les deux repères, nommée $R_{Vicon->UWB}$ pour la rotation et $t_{Vicon->UWB}$ pour la translation, est appliquée. On trouve d'abord le centre de chaque nuage de points donné par l'UWB et le système Vicon. La fonction est appelée $Centroid$ et est définie par :

$$Centroid(Uwb) = \frac{1}{N}\sum_{i}^{N} P_i^{UWB} \tag{2.3}$$

$$Centroid(Vicon) = \frac{1}{N}\sum_{i}^{N} P_i^{Vicon} \tag{2.4}$$

P_i^{UWB} et P_i^{Vicon} sont les positions respectives de l'Ultra WideBand et du système VICON. Le but de cette étape est de centrer les deux ensembles de données avant d'établir la rotation entre les deux trames. Une matrice de covariance, appelée H, est calculée comme suit :

$$H = \sum_{i}^{N} (P_i^{UWB} - centroid(Uwb))\ (P_i^{Vicon} - centroid(Vicon))^T. \tag{2.5}$$

Dans une deuxième étape, nous utilisons une SVD (Singular Value Decomposition) tel que :

$$\Big[U, S, V\Big] = SVD(H) \tag{2.6}$$

pour trouver la rotation entre la trame Vicon et la trame UWB [37, 9] donné par :

$$R_{Vicon->UWB} = VU^T. \tag{2.7}$$

Parfois, la SVD renvoie une matrice de réflexion qui n'existe pas dans une situation réelle. La solution consiste à multiplier la troisième colonne de R (rotation) par -1 si le déterminant de R est négatif [9]. Il reste alors à trouver la translation entre les deux nuages de points comme ci-dessous :

$$t_{UWB} = Centroid(Vicon) - Centroid(Uwb). \tag{2.8}$$

Cela nous donne la translation entre les deux nuages de points :

$$P_i^{UWB} = R_{Vicon->UWB} \times P_i^{Vicon} + t_{Vicon->UWB}. \tag{2.9}$$

L'algorithme de recalage est présenté dans Algorithm 1 et sera utilisé pour recaler le système Vicon et le système UWB.

Algorithm 1 Fonction transformation entre deux nuages de points 3D

procedure RIGIDTRANSFORM(U, V) ▷ Uwb pour l'UWB et Vic pour le Vicon
  if Shape Uwb = Shape Vic **then**
    The function is available
  $UUwb \leftarrow Uwb - Centrod(Uwb)$ ▷ Recentre les points au centres des axes
  $VVic \leftarrow Vic - Centrod(Vic)$
  $H \leftarrow transpose(UUwb) * VVic$
  $U, S, Vt \leftarrow SVD(H)$
  $Rotation \leftarrow Vt.T * U.T$
  if Det(Rotation) < 0 **then**
    Vt[2, :]*=-1
    Rotation= Vt.T* U.T ▷ Detection de la matrice de reflexion
  $Translation \leftarrow -Rotation * centrod(Uwb) + centrod(Vic) \bmod b$
  return $Rotation, Translation$

2.2 Tests et évaluation

2.2.1 Précision de la mesure statique

D'un point de vue statistique, l'écart-type correspond à la précision, l'erreur moyenne correspond à la précision et à la marge moyenne appelée la plage de déviation [66]. Le premier test consiste à placer le tag UWB dans la zone intérieure des ancrages de l'UWB. Ce test nous donnera la distribution des points UWB lorsque le tag est statique.

Les résultats du test statique sont donnés dans le tableau 2.1. L'erreur moyenne sur les 3 axes XYZ est de 1 cm, et la portée moyenne, de 10 cm. Les valeurs sont réparties autour de la valeur moyenne avec un écart type de 0,011 m. Cela signifie que le système UWB n'est pas précis en situation statique (point par point), mais a une précision élevée de 10 cm en moyenne (une moyenne de plusieurs points).

Ces résultats confirment ceux de Jimenez et al. [119, 60] et les résultats de Dotlic et al. [34] en statique lors du calcul de la moyenne. L'UWB est précis jusqu'à 10 cm en statique et se comporte comme une sphère autour de la cible avec une valeur de portée de 10 cm, comme le montre la figure 2.2. Lorsqu'une personne ne bouge pas et l'environnement non plus, nous savons où se trouve la personne avec une précision de 10 cm. Nous pourrions utiliser cette information pour améliorer le calcul "à l'estime" appelé "dead reckoning" pendant la phase statique de celui-ci lorsqu'un IMU (Inertial measurements Unit) est porté au bras.

TABLE 2.1 – Comparaison des erreurs de localisation moyennes et de l'écart-type avec un test statique en condition de ligne de visée.

Static LDV Test	**X-Axis**	**Y-Axis**	**Z-Axis**	**2D**	**3D**
Erreur moyenne	0.01 m	0.01 m	0.01 m	0.01 m	0.01 m
Portée	0.09 m	0.10 m	0.11 m	0.09 m	0.10 m
Ecart type	0.01 m	0.01 m	0.01 m	0.01 m	0.01 m

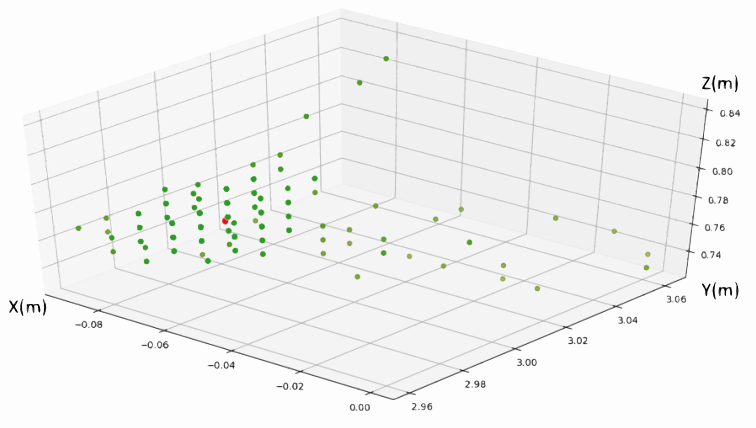


FIGURE 2.2 – Distribution des points UWB (en vert) par rapport à la vérité de terrain Vicon (en rouge).

2.2.2 Mesure dynamique Évaluation et précision d'une trajectoire

Grâce au système Vicon [94], nous avons comparé le point 3D exact du Vicon avec le point 3D de l'UWB en temps réel en utilisant le logiciel RTMaps (Real Time, Multisensor, Advanced Prototyping Software) (`https://intempora.com/products/rtmaps.html`) pour enregistrer nos données. Le premier test que nous avons effectué était une trajectoire dans la zone intérieure des ancres de l'UWB. Le test a été effectué en laboratoire, à LDV, dans des conditions industrielles.

Ce résultat montre que, dans la localisation dynamique, nous pouvons utiliser l'UWB pour le suivi de mouvement avec l'axe X-Y en temps réel mais pas en 3D car l'axe Z n'est pas fiable. La figure 2.3 montre que les mesures de l'axe Z sont ondulées. Les raisons possibles des erreurs de l'axe Z seront une dilution de la précision (DOP) verticale car tous les ancrages sont à la même hauteur.

En mesure XY, nous avons une précision de 21 cm et 78% des valeurs sont meilleures que 0,2 m comme le montrent la Figure 2.4 et le tableau 2.2. Nous avons 0,135 m d'écart-type en XY pour cette trajectoire en XY. Nous pouvons utiliser UWB pour la localisation dynamique en temps réel. Nous avons une précision de 0,24 m en 3D (XYZ) ; seuls 40% des valeurs de l'axe Z sont précises, comme le montre la figure 2.4. L'axe Z n'est pas fiable pour la localisation dynamique et pour la reconnaissance des mouvements.

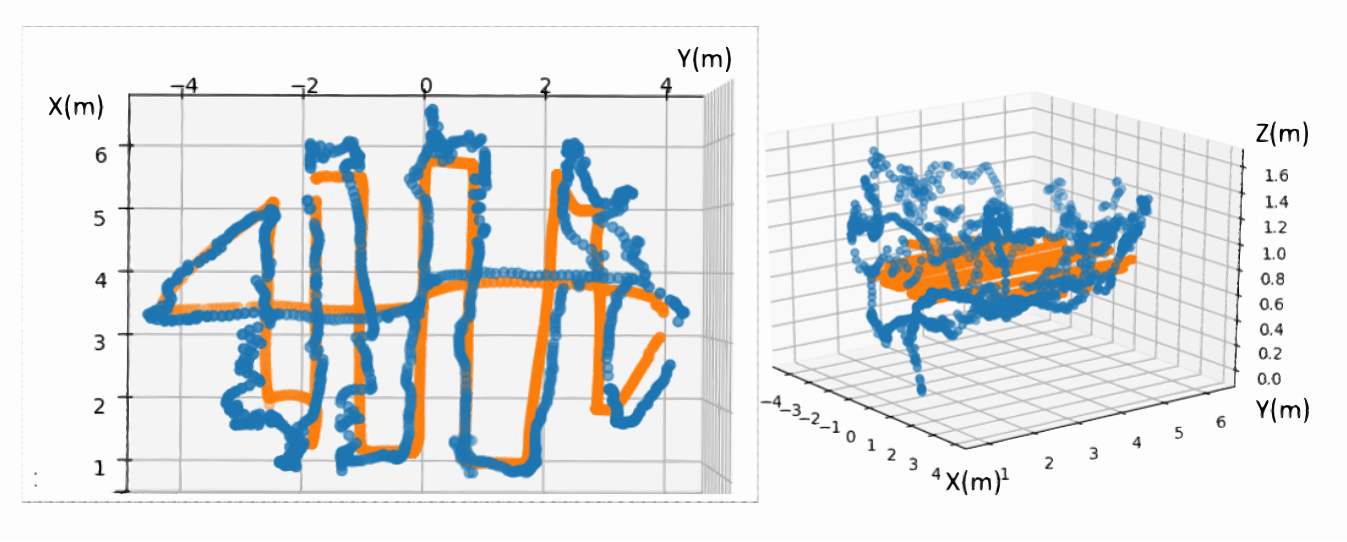


FIGURE 2.3 – Trajectoire réalisée en laboratoire dans des conditions de LDV en 2D et 3D avec VICON (orange) et UWB (bleu) en mètres.

TABLE 2.2 – Tableau de la dynamique trajectoire.

Mesure Dynamique	**X-Axis**	**Y-Axis**	**Z-Axis**	**2D**	**3D**
Erreur moyenne	0.20 m	0.22 m	0.32 m	0.21 m	0.24 m
Portée	0.73 m	0.64 m	0.87 m	0.65 m	0.75 m
Ecart type	0.13 m	0.14 m	0.29 m	0.13 m	0.18 m

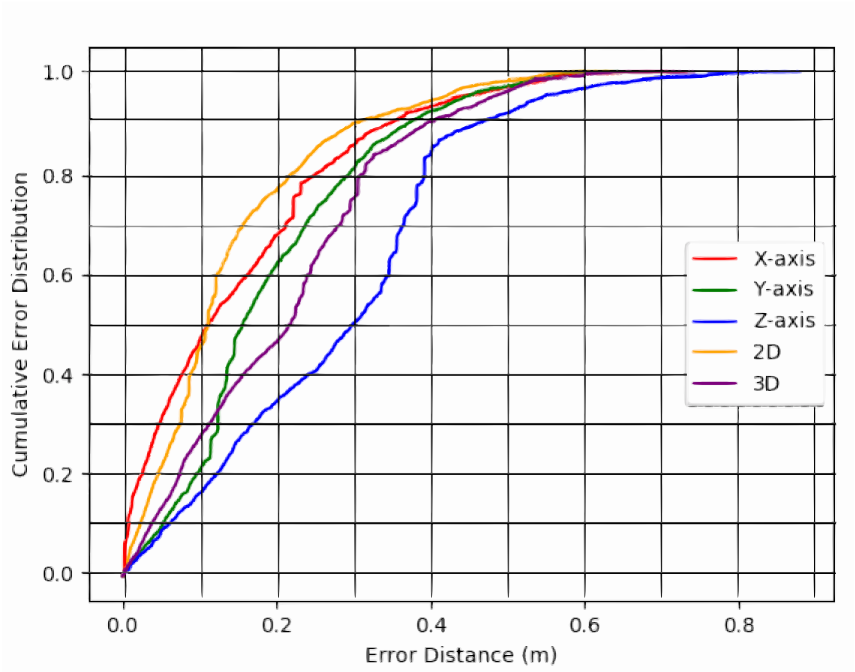
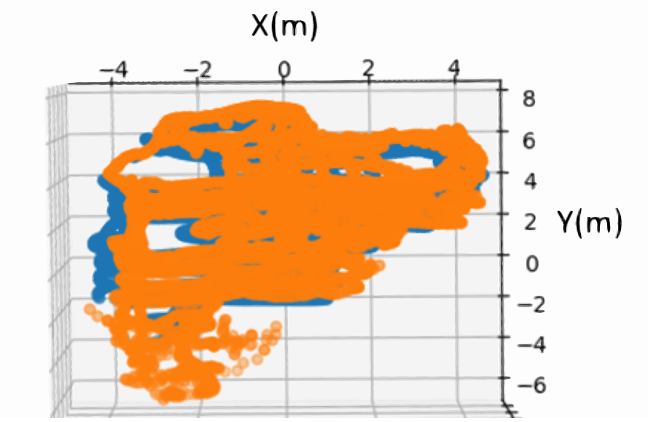
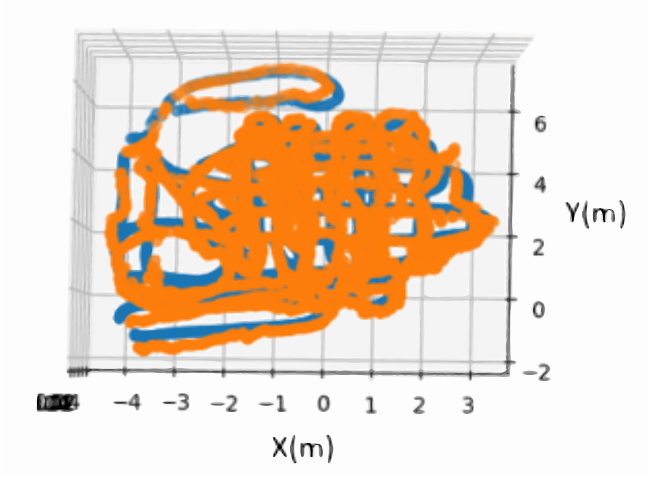


FIGURE 2.4 – Comparaison de la distribution des erreurs cumulées entre l'axe X, l'axe Y, l'axe Z, la 2D et la 3D de l'UWB dans une condition LDV industrielle réalisée en laboratoire.



a Trajectoire du déplacement dynamique dans la zone extérieure de l'UWB dans des conditions de vue directe



b Trajectoire du déplacement dynamique dans la zone intérieure en condition de vue directe.

FIGURE 2.5 – Trajectoire des deux déplacements dynamiques de l'UWB en orange et du Vicon en bleu comme vérité terrain.

2.2.3 Évaluation des mesures dynamiques et précision de la cartographie

Le troisième test consiste à réaliser une cartographie de la zone extérieure aux placements des ancres UWB (figure 2.5a) et de la zone intérieure aux placements des ancres UWB (figure 2.5b) du système UWB pour évaluer son comportement. Nous avons couvert la zone maximale et essayé de voir si la précision/exactitude a changé.

Mok et al. [100] décrivent tous les facteurs d'influence utilisant la bande ultra-large pour le positionnement, tels que le matériel d'obstruction, les trajets multiples, l'effet géométrique et l'atténuation du signal. Nous savons par cet article [100] que dans notre laboratoire nous avons une structure métallique proche de la zone d'essai et des robots métalliques qui peuvent avoir une influence sur les données. Par rapport à notre premier test, nous avons une précision de 23 cm dans la zone intérieure et de 25 cm dans la zone extérieure en 2D et 23 cm et 24 cm en 3D, ce qui est proche de notre premier résultat de localisation dynamique présenté dans le tableau 2.2. Ces deux tests montrent que l'UWB est homogène pour une zone couverte même en dehors de la zone définie par ses ancrages dans des conditions de LDV industrielles. L'UWB est vraiment bon pour la localisation dynamique dans des environnements intérieurs.

Nous avons moins de précision par rapport à nos résultats statiques. Nous avions une précision de 10 cm en mesure statique, alors que nous avions une précision de 0,24 m en localisation dynamique comme le montre le tableau 2.3. Dans le cas du test statique, nous avons calculé une erreur moyenne tandis que dans le cas dynamique, nous calculons une erreur instantanée (point par point) pour chaque point. C'est la principale raison de l'augmentation de l'erreur dans le cas dynamique.

TABLE 2.3 – Erreurs de la mesure dynamique de la cartographie.

	UWB Mapping	X-Axis	Y-Axis	Z-Axis	2D	3D
Inner	Erreur moyenne	0.30 m	0.17 m	0.23 m	0.23 m	0.23 m
	Portée	1.07 m	0.60 m	1.37 m	0.56 m	1.01 m
	Standard deviation	0.18 m	0.00 m	0.20 m	0.18 m	0.12 m
Outer	Erreur moyenne	0.23 m	0.27 m	0.23 m	0.25 m	0.24 m
	Portée	0.98 m	1.05 m	1.03 m	1.01 m	1.02 m
	Standard deviation	0.03 m	0.15 m	0.19 m	0.09 m	0.12 m

TABLE 2.4 – Résultat dynamique avec différentes hauteurs pour les ancres.

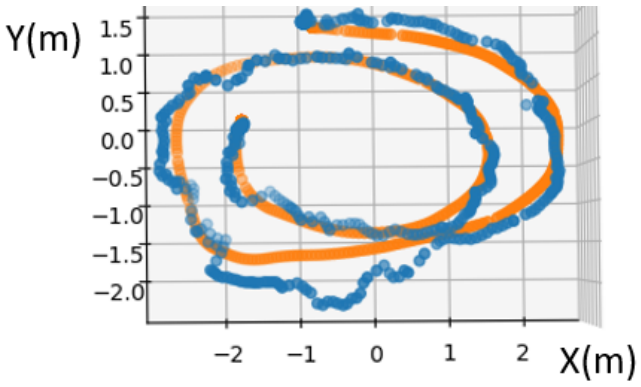
Anchors Z Change	X-Axis	Y-Axis	Z-Axis	2D	3D
Erreur moyenne	0.38 m	1.37 m	0.70 m	0.87 m	0.81 m
Portée	0.78 m	0.64 m	1.28 m	0.71 m	0.90 m
Ecart type	0.04 m	0.05 m	0.22 m	0.04 m	0.10 m

2.2.4 Test changement de hauteurs des ancres sur l'axe Z

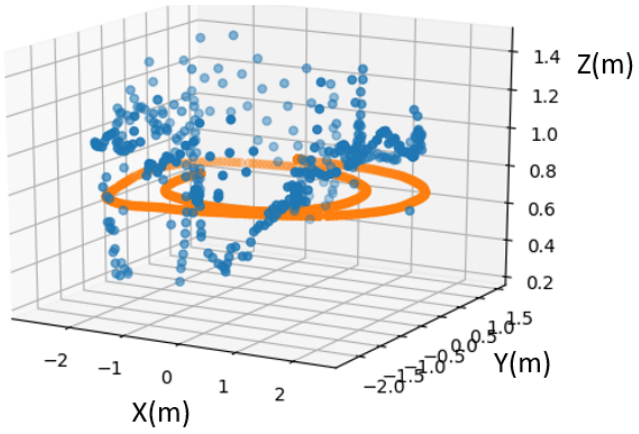
Ce test a été réalisé pour vérifier le comportement du système UWB lorsque la hauteur des ancres change. Ayan et al. [6] a déclaré que la hauteur des ancres peut modifier la précision de l'UWB. Dans ce test, nous voulons connaître le comportement dans une situation dynamique. Nous avons abaissé 2 ancres de 40 cm, et ce changement a été pris en compte dans les mesures d'étalonnage. Nous avons une mauvaise précision de 0,87 m en X–Y et de 0,81 m en 3D comme le montre le tableau 2.4 qui confirme que lorsque les ancres Z ne sont pas à la même hauteur, la précision et l'exactitude de la localisation dynamique sont affectées.

2.2.5 Étude de l'influence des ancres

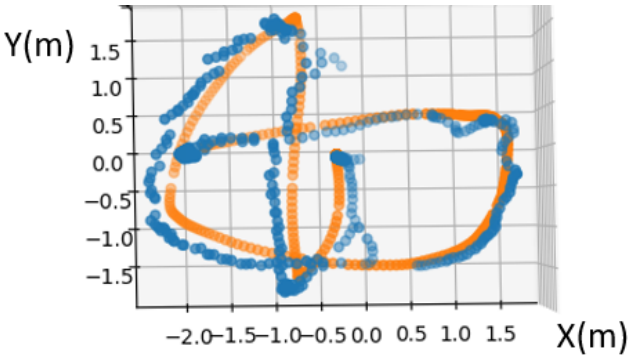
Nous avons effectué ce test pour vérifier le comportement de l'UWB dans la zone intérieure de l'UWB en dynamique avec cinq et six ancrages. Nous avons placé quatre ancrages exactement comme sur la figure 2.1a et un autre sur le sol dans un coin de notre cube (configuration à cinq ancrages). Ensuite, nous avons placé quatre ancres comme sur la figure 2.1 et deux autres sur le sol dans les coins (configuration à six ancres). L'objectif est de voir une meilleure performance de l'axe Z. Avoir 5 ou 6 ancres améliore la précision et l'exactitude pour l'axe 3, mais l'axe Z est encore trop ondulé comme le montre la figure 2.6b,d. L'axe Z présente une erreur moyenne de 0,22 m, comme le montre le tableau 2.5, ce qui est mieux que les 0,23 m indiqués dans le tableau 2.3 mais pas de manière significative. Par rapport à l'erreur de distribution cumulée, nous constatons une amélioration de l'axe Z d'environ 10 cm, 80% des valeurs sont inférieures à 0,3 m comme indiqué dans le tableau 2.5 avec cinq ancrages et 80% des valeurs sont inférieures à 0,4 m de précision avec quatre ancrages comme indiqué dans la figure 2.4. L'axe X a 80% des valeurs inférieures à 0,2 m comme illustré dans Figure 2.7a, ce qui est mieux que notre premier



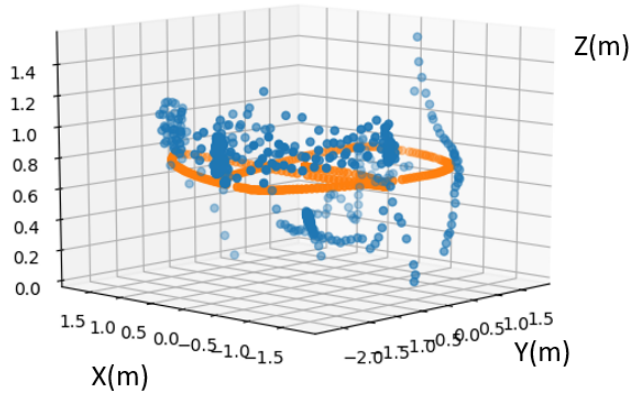
a Trajectoire dynamique de l'UWB en 2D avec 5 ancres N avec VICON (orange) et UWB (bleu).



b Trajectoire dynamique de l'UWB en 3D avec 5 ancres N avec VICON (orange) et UWB (bleu).



c Trajectoire dynamique de l'UWB en 2D avec 6 ancres N avec VICON (orange) et UWB (bleu).



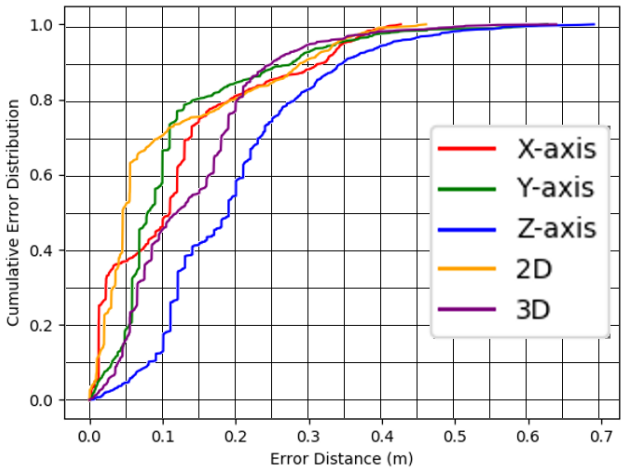
d Trajectoire dynamique de l'UWB en 3D avec 6 ancres N avec VICON (orange) et UWB (bleu).

FIGURE 2.6 – Trajectoire dynamique de l'UWB réalisée en laboratoire avec 5 et 6 ancres avec le système Vicon comme vérité de terrain dans des conditions de LDV industrielles.

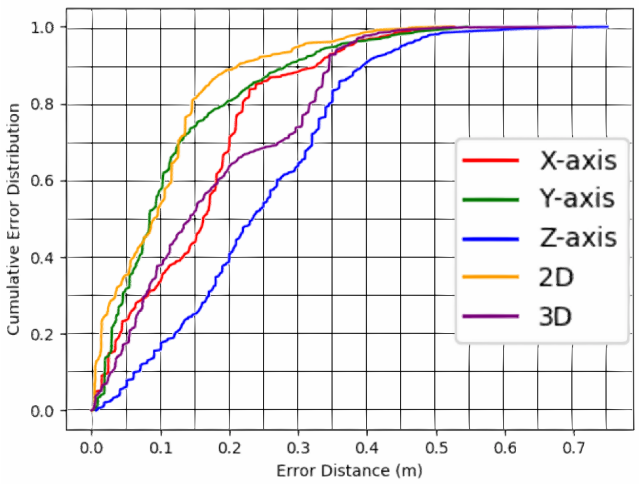
essai dynamique avec quatre ancrages comme le montre la figure 2.4. L'axe Y améliore également sa précision. Dans la figure 2.7b l'axe Z a une mauvaise précision, 80% des valeurs sont supérieures à 0,3 m. Nous améliorons l'exactitude et la précision de l'UWB en ajoutant une ou deux ancre(s). Selon les résultats du tableau 2.5, pour 5 ancrages nous avons 5 cm d'erreurs en 2D et 6 cm d'erreurs en 3D. Les performances sont plus faibles pour 6 ancres. Il est possible que la position de la sixième ancre conduise à augmenter l'influence d'un phénomène indésirable qui prend le dessus sur l'avantage d'utiliser plus d'ancres. Une hypothèse pourrait être que le signal reçu/transmis depuis/vers cette sixième ancre a plus de trajets multiples. Cependant, une chose qui est clairement visible sur la figure 2.7 est l'amélioration de l'écart-type. Cela signifie que la trajectoire est moins dispersée, mais reste centrée dans la même erreur en 2D.

TABLE 2.5 – Influence des ancrages UWB dans les conditions de LDV en milieu industriel.

	Influence of Anchors	X-Axis	Y-Axis	Z-Axis	2D	3D
4 anchors	Erreur moyenne	0.20 m	0.22 m	0.32 m	0.21 m	0.24 m
	Portée	0.73 m	0.64 m	0.87 m	0.65 m	0.75 m
	Standard deviation	0.13 m	0.14 m	0.29 m	0.13 m	0.18 m
5 anchors	Erreur moyenne	0.16 m	0.16 m	0.22 m	0.16 m	0.18 m
	Portée	0.42 m	0.60 m	0.66 m	0.51 m	0.56 m
	Standard deviation	0.02 m	0.10 m	0.22 m	0.06 m	0.11 m
6 anchors	Erreur moyenne	0.19 m	0.16 m	0.27 m	0.17 m	0.20 m
	Portée	0.52 m	0.54 m	0.74 m	0.53 m	0.6 m
	Standard deviation	0.10 m	0.01 m	0.26 m	0.05 m	0.12 m



a Comparaison de la précision de l'UWB avec 5 ancres.



b Comparaison de la précision de l'UWB avec 6 ancres..

FIGURE 2.7 – Fonction de distribution cumulative empirique des erreurs entre l'axe X, l'axe Y, l'axe Z, la 2D et la 3D de l'UWB dans les conditions industrielles LDV réalisées en laboratoire.

2.3 Conclusion

Dans ce chapitre, nous avons décrit le comportement d'un système Ultra WideBand dans des cas statiques et dynamiques par comparaison avec une vérité terrain obtenue avec un système de capture de mouvement. Nous avons une évaluation de la précision et de l'exactitude du système UWB qui est vraiment bonne pour les axes X-Y mais pas fiable le long de l'axe Z. Nous avons également montré que si nous modifions la hauteur des ancres, nous perdons de la précision et de l'exactitude dans la localisation statique et dynamique. Nous avons également confirmé que la précision et l'exactitude sont meilleures en ajoutant des ancres lors de la localisation dynamique. Les systèmes UWB peuvent être un très bon choix pour la localisation, même en dynamique, et peuvent être plus robustes si nous ajoutons plus d'ancres, sans trop en ajouter pour ne pas créer de multitrajet qui réduira la précision. L'axe Z doit être amélioré, surtout en termes de précision, et cela peut être réalisé par la fusion de données avec d'autres capteurs. Nos travaux futurs, dans le chapitre suivant, consisteront à améliorer la précision du système dans des conditions très perturbées en zone industrielle avec un suivi dynamique des personnes dans un atelier de production.

CHAPITRE 3

Un nouvel ensemble de données sur la circulation des personnes dans un site industriel grâce à l'UWB et aux systèmes de capture du mouvement

L'amélioration des performances et des conditions de sécurité sur les sites industriels reste un objectif clé pour la plupart des entreprises. Actuellement, l'objectif principal est de pouvoir localiser de manière dynamique les personnes et les biens sur le site. La sécurité et la réglementation de l'accès aux zones restreintes sont souvent assurées par des portes ou des barrières à badges et celles-ci posent plusieurs problèmes lorsque des personnes se trouvent dans des endroits où elles ne sont pas sensées se trouver ou même lorsque des outils ou des objets sont utilisés de manière incorrecte. En outre, l'utilisation croissante de nouveaux dispositifs exige des informations précises sur leur emplacement dans l'environnement, comme les robots mobiles ou les drones. Il devient donc essentiel de disposer des outils permettant de gérer de manière dynamique ces flux de personnes et de biens. Des solutions à bande ultra-large et de capture de mouvement seront utilisées pour identifier rapidement les personnes qui pourraient se trouver dans des zones non autorisées ou qui effectuent des tâches pour lesquelles elles n'ont pas reçu d'instructions. En plus du suivi dynamique des personnes, cela permet également de surmonter certains problèmes liés au déplacement d'objets ou d'outils dans l'atelier de production. Nous proposons un nouvel ensemble de données qui fournissent des informations précises sur les mouvements des travailleurs. Cet ensemble de données peut être utilisé pour développer de nouvelles mesures concernant l'efficacité et la sécurité des travailleurs.

3.1 Introduction

3.1.1 Suivi des personnes dans un atelier de fabrication manuelle

Dans une usine, ou un site industriel, différentes ressources (personnes, machines, etc.) coopèrent et interagissent pour effectuer diverses tâches dans le cadre de processus industriels et aboutir à la réalisation de différents cycles de production ou de soutien [118]. Cependant, certaines interactions involontaires entre ces ressources peuvent être contre-productives pour les objectifs de performance, de sûreté et/ou de sécurité de l'entreprise. Par exemple, certains articles peuvent représenter un danger pour certains opérateurs s'ils ne sont pas équipés d'une protection adéquate (questions de sécurité). D'autres éléments, qui peuvent être de nature confidentielle, ne doivent pas être accessibles à tous les opérateurs et ne doivent être accessibles qu'à quelques parties prenantes sur le site. D'autre part, la surveillance spatiale et temporelle peut être utilisée pour mieux contrôler la production, le stockage des produits ou les interventions concernant d'autres opérations telles que la maintenance, les mises à jour (mises à niveau) du matériel ou des logiciels, etc. Il existe un réel besoin de connaître et de contrôler l'emplacement des différents articles et des personnes dans une usine afin d'éviter les risques liés aux personnes et aux articles se trouvant parfois dans un endroit inapproprié (risque de dysfonctionnement, risque de fuite d'informations confidentielles, risque de problèmes de sécurité, etc.) Dans la pratique, les processus industriels, les "méthodes d'organisation" sont utilisées pour planifier les tâches de chaque opérateur et la présence de certains articles (matières premières, produits finis, produits en stock, outils, pièces de rechange, etc.) Même si ces informations sont intégrées dans les outils de planification des ressources de l'entreprise (ERP) [56], aucune procédure n'est en place pour la présence de personnes et d'articles dans des endroits inappropriés, ce qui n'exclut pas les risques mentionnés. En outre, ces méthodes sont statiques en ce sens que les informations relatives à la localisation spatio-temporelle des objets et des personnes ne sont pas mises à jour de manière systématique, rapide et fiable. L'usine du futur, qui incarne la 4ème révolution industrielle, est une usine dont la vocation est d'être une industrie flexible, connectée et intelligente [70]. Tous les composants (machines, opérateurs, objets, systèmes de transport, etc.) qui composent l'entreprise doivent pouvoir communiquer avec le système de supervision et entre eux. Plusieurs types de sources d'information peuvent être utilisés pour détecter les états et les emplacements des personnes et des objets, tels que les capteurs, déjà présents dans l'usine ou portés par les personnes, les horaires du personnel, etc. [53]. Cependant, les moyens technologiques actuels sont souvent exploités séparément par les systèmes d'information de l'entreprise, ce qui ne permet pas une analyse approfondie et précise, ni une décision pertinente sur les risques éventuels ; cela pourrait entraîner une faille dans la sécurité. Il est donc nécessaire de pouvoir fusionner toutes ces données provenant de plusieurs sources lorsqu'on relie les informations sur l'identité des personnes et la nature des objets concernant leurs positions. Les informations sur la position occuperont une place autour de laquelle toutes les autres informations seront structurées de manière dynamique.

Ce travail est constitué d'un ensemble complet de données de six travailleurs sur une chaîne de montage

industrielle en NLDV pendant trois heures, et de la création d'un algorithme pour lisser les données des travailleurs. Nous avons également proposé une classification des zones en fonction de l'environnement et une analyse de la plate-forme d'un travailleur. Deux modalités sont proposées : un système à bande ultra-large (UWB) et un système de capture de mouvement appelé MoCap.

Ce chapitre est divisé en cinq parties. Nous avons établi une enquête sur les ensembles de données existants dans la NLDV. Nous avons présenté notre dispositif expérimental. Nous avons étudié le comportement de nos données brutes synchronisées et de nos résultats. Nous avons proposé un moyen d'améliorer nos données. Nous avons proposé l'utilisation et l'interprétation de notre ensemble de données.

3.2 Mise en place expérimentale

Pour obtenir des données à partir des balises UWB, nous avons placé quatre ancres dans la pièce utilisée pour la localisation à l'intérieur. Pour être aussi précis que possible, nous avons placé les ancres dans une configuration rectangulaire recommandée par le fabricant, pour cela nous utilisons le MDEK1001 des systèmes de localisation en temps réel (RTLS) de Decawave, qui est représenté sur la figure 3.1 sous la forme d'un rectangle rouge.

Ce système (MDEK1001) a été amélioré récemment par un filte de kalman (EKF) robuste [59]. Comme évoqué dans le chapitre précédant, un ensemble de balises et d'ancres doit être associé au même réseau (cluster) afin de pouvoir obtenir des informations de localisation pour chacun d'entre eux. L'une des ancres du réseau doit être configurée en tant que maître, c'est-à-dire celui qui maintient le timing et la synchronisation de tous les modules d'un réseau et un réseau d'ancres est toujours composé de quatre ancres par cluster pour calculer la position. *Jimenez et al.* [59] ont utilisé huit ancres afin de refaire un algorithme de calcul de la position avec la connaisssance de 8 ancres et non quatre. Ils ont utilisés un EKF à six états, avec trois termes pour la position 3D, x, y et z, et trois pour la vitesse V_x, V_y et V_z :

$$X = (x, y, z, V_x, V_y, V_z)^T \tag{3.1}$$

où le modèle de mouvement est une fonction de vitesse constante f reliant l'évolution des états, plus un bruit additif $g(a)$ qui prend en compte les changements d'accélération de l'objet en mouvement. Le modèle de mesure dans l'EKF utilise les portées UWB enregistrées avec le Bluetooth avec chacun des tags d'une paire (huit portées au total, quatre pour chacune des deux tags). La trilatération est calculée implicitement dans le filtre de Kalman en définissant en conséquence la matrice de mesure H, qui est de taille $Na \times 6$, Na étant le nombre d'ancres connectées au tag à l'instant courant $(Na \leqslant 8)$. La ligne i de la matrice H est de la forme :

$$H_i = (\frac{x - x^i}{r^i} \quad \frac{y - y^i}{r^i} \quad \frac{x - x^i}{r^i} \quad 0 \quad 0 \quad 0) \tag{3.2}$$

avec r^i la distance entre les ancres a^i et la dérnière estimation de la position d'un tag. Ils ont choisit de supprimer les

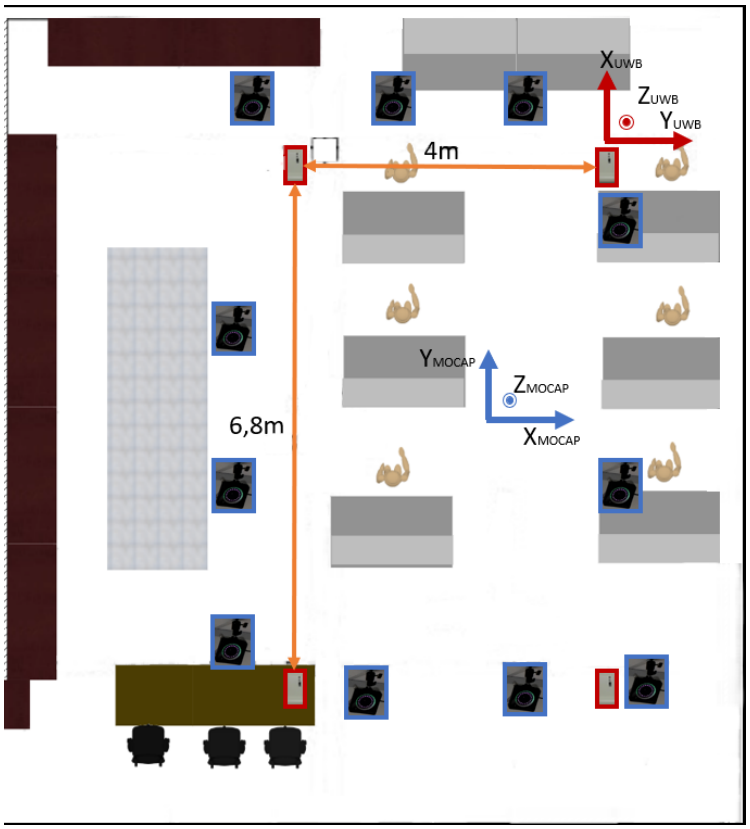


FIGURE 3.1 – Mise en place de la chaîne d'assemblage. Les ancres à bande ultra-large (UWB) sont placées dans une configuration rectangulaire en rouge. Les caméras MoCap sont placées autour de la zone en bleu.

valeurs de position au delà de $0.6m$ car considéré comme aberrantes, nous le verrons dans la suite de ce chapitre mais ces valeurs abérantes pourront être traitées par le filtre de Savitsky-Golay afin de renforcer cet algorithme.

Nous avons choisi une ancre comme référence (initialisation à x = 0 et y = 0 selon la référence UWB), et nous devons obtenir la position de chaque ancre selon l'ancre d'initialisation comme le montre la figure 3.1. Pour ce faire, nous utilisons un algorithme de télémétrie à deux voies (TWR) fourni par Decawave. L'algorithme TWR de la norme IEEE P802.15 [55] estime la distance entre chaque ancre et la balise selon la formule suivante,

$$T_{TOA} = (T_2^R - T_1^R) - (T_2^T - T_1^T) - E_2 - E_1 \tag{3.3}$$

où T_{TOA} est le temps de vol entre la station de référence et le tag UWB ; T_2^R et T_1^R sont les horodatages du récepteur ; T_2^T et T_1^T sont les horodatages du récepteur ; et E2 et E1 sont les erreurs affectées par la puissance du signal des horodatages T_1^T et T_2^T, respectivement. Si l'on suppose que la vitesse des ondes radio dans l'air est la même que la vitesse de la lumière c, alors la distance entre les ancres et le tag UWB peut être déterminée par

$$Distance = c \times T_{TOA} \tag{3.4}$$

La position du tag UWB peut être calculée par des mesures TOF provenant de différents capteurs. Le moteur de localisation utilise la probabilité maximale entre les quatre ancres pour donner la position. Une balise UWB est placée dans la poche de chaque personne.

3.2.1 Installation industrielle

L'objectif des scénarii de processus de la chaîne de montage est de construire six tricycles en trois heures. Chaque plate-forme a son propre processus d'assemblage. Il y a un ouvrier pour chaque plate-forme. Les stations sont fixées comme indiqué dans la figure 3.2b pour les stations un, deux et trois et la figure 3.2a pour les stations quatre, cinq et six. Au cours de tous les processus, chaque personne effectue la même tâche.

La personne qui se trouve à la station 1 construit le cadre inférieur du tricycle. Dès qu'elle a terminé, elle donne sa partie à la personne de la station 2. La personne à la station 2 reçoit la partie inférieure du tricycle et assemble l'essieu avec la partie inférieure. Dès qu'elle a terminé, elle donne sa partie à la personne de la station quatre. La personne qui se trouve à la troisième station assemble la selle et le pédalier et les donne à la personne qui se trouve à la quatrième station. La personne à la station quatre assemble les deux parties données par la station, qui sont l'arrière, les roues et l'unité d'essieu. Elle le donne ensuite à la personne de la station six. La personne se trouvant à la cinquième station assemble la roue avant du tricycle et l'unité d'essieu. Elle le remet ensuite à la personne qui se trouve à la station six. La personne se trouvant à la station six assemble les deux dernières parties du tricycle. Ces processus sont schématisés à la figure 3.4.

Au terme de la période d'enregistrement de trois heures, six tricycles seront construits, comme le montre la figure 3.3. Chaque personne devra se réapprovisionner, car le stock initial est de trois vélos. Les personnes auront un parcours qui s'écartera du protocole de la figure 3.4, par exemple, lorsqu'elles feront une pause. Cela peut signifier que le stock n'a pas été correctement planifié au départ et que l'opérateur devra le reconstituer. Il prendra donc un chemin différent de celui du protocole initial. Les personnes peuvent également aller aider d'autres personnes ou faire des pauses. Grâce aux données de nos capteurs, nous pouvons détecter quand une personne ne respecte pas le processus d'assemblage. Des actions peuvent également être entreprises pour améliorer le confort de l'opérateur.

3.2.2 Système de capture du mouvement

Nous avons utilisé un système de capture de mouvement d'Optitrack similaire à celui de [94] avec une précision millimétrique [47] comme vérité terrain.

Cet outil ne donne pas directement la position des personnes mais des points de réflexion. C'est en les combinant dans des formes déterminées par groupes de 4 ou 5 (appelés "corps solides") que l'identification devient unique et permet de suivre un objet ou un individu. L'inconvénient de ce type de localisation est le masquage d'un ou plusieurs points, rendant l'identification singulière ou impossible. Dans notre cas, les structures métalliques des

FIGURE 3.2 – Notre système UWB est installé dans une chaîne de montage industrielle NLDV. Dans les carrés bleus se trouve le système MoCap et dans le carré rouge le système UWB lorsqu'il n'est pas caché en raison des conditions NLDV. Installation industrielle dans une zone non visible (NLDV) avec six installations de montage. vue [A]. (**b**) Installation industrielle en NLDV avec six postes de montage. vue [B].

FIGURE 3.3 – Assemblage final des tricycles, et pendant le processus dans un état industriel NLDV fait dans un atelier. (**a**) Vue de face du processus d'assemblage. Dans le carré rouge se trouve le système UWB et dans le carré bleu, le système MoCap. (**b**) Résultat du processus d'assemblage.

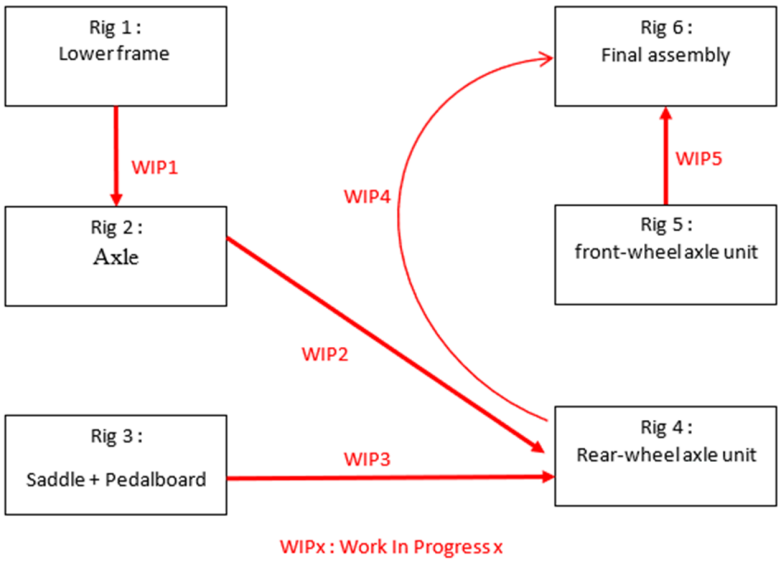


FIGURE 3.4 – Configuration du scénario de mouvements de six personnes correspondant aux six plates-formes dans une condition NLDV réalisée dans un atelier

stations manuelles bloquent de nombreuses lignes de vue et il arrive donc régulièrement que l'on perde la localisation. Dans la base de données, le résultat est simplement un trou dans les données jusqu'à ce que tous les points réfléchissants soient récupérés pour assurer l'unicité de l'objet ou de la personne détectée (à partir de 4 points). Nous avons choisi de placer des points réfléchissants sur la tête des personnes, ce qui limite la perte de données.

Le tableau est donc constitué d'une colonne de temps synchrone (toutes les 100 ms) et des coordonnées du centre de chaque corps solide. Nous avons choisi de mettre les données sous la forme d'un angle d'Euler pour leur lisibilité et les coordonnées x, y et z.

Nous avons dû appliquer les translations et les rotations nécessaires pour déplacer les axes et le centre comme le montre la figure 3.5. Ainsi, l'axe Z de l'expérience devient l'axe X de la base de données, et l'axe X devient l'axe Y. Une translation de 5,6 m sur l'axe X et de 5,5 m sur l'axe Y sont finalement appliquées. Nous faisons ce changement parce que le système MoCap a sa propre référence dans l'atelier et le système UWB a également sa propre référence dans l'atelier. Nous avons utilisé plusieurs positions statiques pour calibrer le système MoCap dans le temps et les positions. Au début de l'enregistrement, toutes les personnes commencent à la même position (position de départ) et finissent à une autre position connue (position final).

3.2.3 Système à bande ultra-large

Cet outil donne directement la position des badges portés à la ceinture ou dans la poche des personnes dans l'atelier. Pour exploiter les données, il a simplement fallu réaligner les données sur les axes indiqués dans la figure 3.1, nous avons donc inversé les deux axes et effectué une translation de 6,85 m sur l'axe x et de 8,19 m sur l'axe y. La zone de fonctionnement optimale est donnée par le cadre rouge de la figure 3.5. Les données restent accessibles en dehors de cette zone, mais il n'y a aucune garantie de précision. Les données ne sont pas reçues de manière synchrone. Chaque badge fournira sa position toutes les 100 ms plus ou moins 10 ms si ce badge est correctement situé. Aucune donnée ne sera transmise si le badge n'est pas correctement localisé.

3.2.4 Discussion sur les données brutes

Ces deux moyens de modalités (MoCap et UWB) ont chacun leurs moyens de localisation, il est donc difficile de garantir que les axes soient parfaitement alignés en raison de la taille du système (zone de travail de plus de 50 m^2). Nous savons qu'une erreur dans le positionnement des badges de référence UWB, ou le référencement de ces positions, induit directement une erreur de positionnement en fonction de la position des badges. Le MoCap garantit une précision de positionnement de quelques millimètres, mais seulement lorsqu'il est dans la capacité de repérer ces marqueurs. Le nombre très élevé de pertes de données lors des mesures montre les limites de ce système dans un atelier de fabrication dense en éléments de masquage. Toutes les comparaisons sont faites par rapport au MoCap et seulement lorsque l'emplacement est bon. Il n'y a pas d'interpolation sur les positions intermédiaires afin

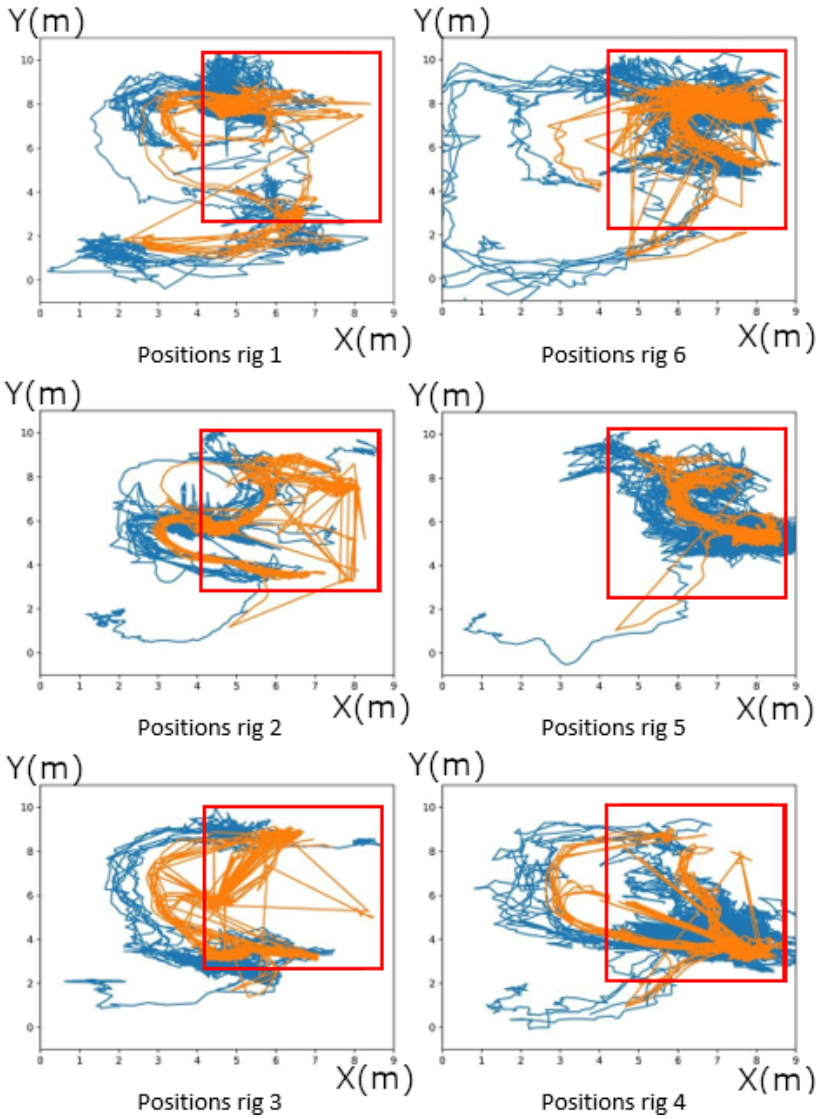


FIGURE 3.5 – Mouvement de chaque personne en fonction de son gréement en mètres réalisés dans un atelier. Le système de capture de mouvement est en orange et le système UWB est en bleu. Le carré rouge est la zone de l'ensemble de travail.

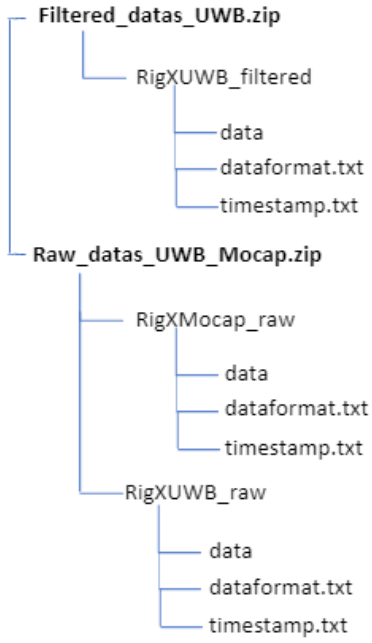


FIGURE 3.6 – Structure des fichiers Zip fournis. X signifie "Rig one to Rig six".

de ne pas fausser les résultats. La synchronisation des deux moyens a été validée pour un décalage de 38 minutes car le système UWB a démarré avant le système MoCap.

3.2.5 Jeu de données

Toutes les lectures des capteurs d'une séquence sont zippées dans deux fichiers uniques nommés "Filtered_datas_UWB.zip" et "Raw_datas_UWB_Mocap.zip". La structure des répertoires est illustrée dans la figure 3.6. Nous fournissons des données brutes du système de capture de mouvement, qui peuvent être comparées au travail de [94] comme système de capture de mouvement avec une précision millimétrique et avec une précision centimétrique dans le cas du système UWB [25, 112, 162]. Les horodatages sont stockés dans timestamps.txt et les formats de données sont stockés dans dataformat.txt. Chaque ligne du fichier timestamps.txt est composée de la date, de l'heure en heures, minutes et secondes. Chaque ligne du dossier de données fournit la position X et la position Y pour le système UWB et la position X, Y et Z pour le système de capture de mouvement. Un aperçu de l'ensemble de données peut être vu dans la figure 3.7, avec un exemple de la $Rig1Mocap\_raw$. Tous les fichiers zippés fournissent la position de six appareils correspondant à la figure 3.4. Nous fournissons également des données UWB filtrées qui sont des données filtrées par le filtre Savitsy–Golay [121].

L'ensemble de données est disponible à l'adresse suivante : IndoorIndustrialLocalisationDataset.

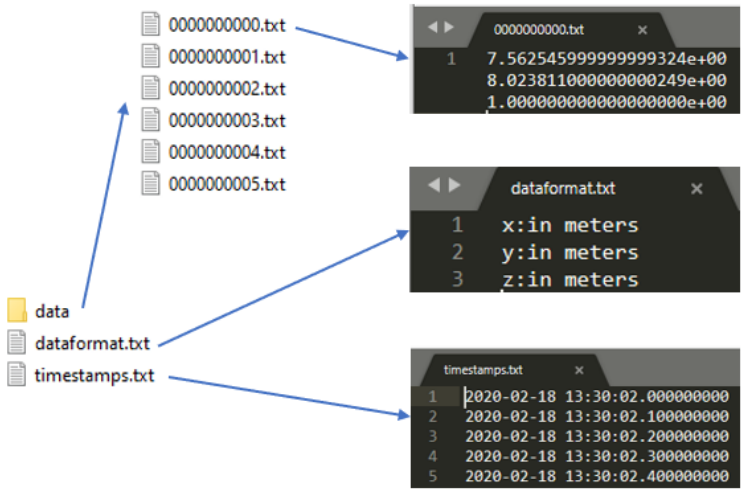


FIGURE 3.7 – Snapshot de l'ensemble de données pour $Rig1Mocap\_raw$.

3.3 Résultats et amélioration des résultats

Positions

La figure 3.5 montre toutes les mesures sur toute la durée de l'étude. Le cadre rouge représente approximativement la zone de confiance dans la localisation de l'UWB. Nous pouvons observer de nombreuses sorties de la zone par les acteurs de cette expérience. Elles ne seront pas prises en compte car le système MoCap ne peut détecter les mouvements de personnes que dans la zone du carré rouge visible sur la figure 3.5. Les stations 2–4 contournent régulièrement la zone en raison d'un équipement qui bloquait l'espace entre les stations 2 et 5. Dans ces graphiques, des segments sont dessinés entre les points non validés, ce qui rend le schéma lourd mais facilite la comparaison.

On peut voir que malgré le bruit de l'emplacement de l'UWB, les courbes se suivent très bien tout au long de l'expérience.

Précision

Nous avons rapidement admis qu'il était impossible d'avoir une simple expression de précision. En effet, plusieurs paramètres entrent en jeu et il est actuellement difficile de savoir quel paramètre est prédominant par rapport à l'autre. En phase statique, nous pouvons avoir une précision à différentes localisations.

A partir de cette observation, nous avons affiché sous forme d'image la précision en fonction de la position des personnes. Nous avons utilisé la RMSE calculée comme suit,

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(\hat{\theta}_i - \theta_i)^2}$$

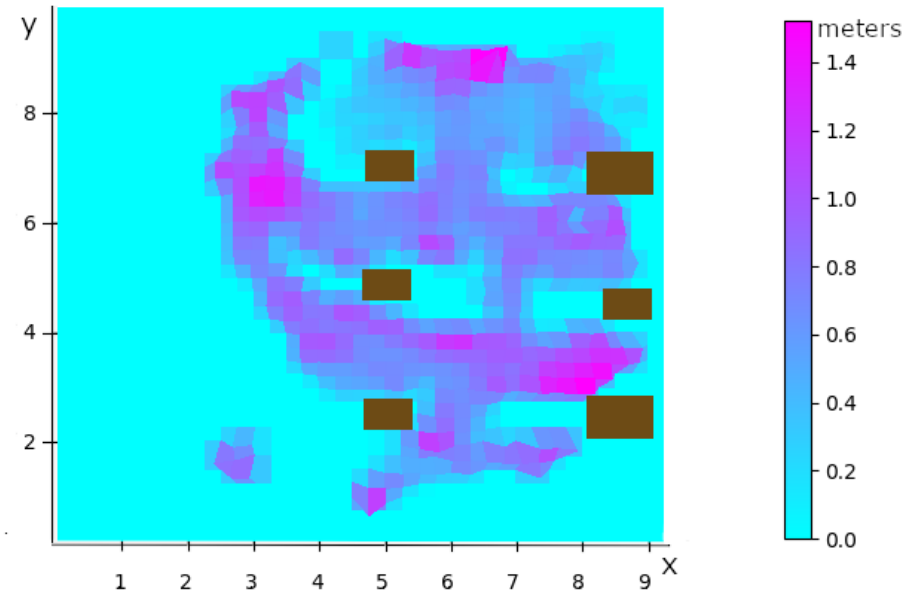


FIGURE 3.8 – Précision en mètre en fonction de la position. Les cyans sont des zones sans données (0,0 m). Violet sont des zones avec une erreur maximale de 1,5 m. Rectangle marron sont chaque rig.

où la différence moyenne entre les valeurs de vérité de terrain du système Mocap, écrites $\hat{\theta}_i$, et les valeurs estimées du système UWB exprimées par θ_i. Nous avons comparé les deux modalités UWB et Mocap après décalage temporel des données. Le résultat est présenté dans la figure 3.8.

Les points indiquant une erreur nulle sont ceux où, il n'y a pas de données disponibles. Pour les autres, nous constatons une forte relation entre le fait d'être sous les tags de référence (aux 4 coins de la zone rouge de la figure 3.5) et l'alignement des structures métalliques des postes. C'est la combinaison des deux qui présente les erreurs les plus fortes, comme les points de coordonnées (8, 3) et (6.5, 9) par exemple. L'erreur maximale dans la zone libre est de 40 cm, tandis que dans la zone métallisée, nous avons un mètre d'erreur.

Discussion des valeurs brutes

L'impact de la vitesse de déplacement semble également très intéressant à quantifier. La vitesse V est calculée par $V = \frac{d}{t}$, avec d le déplacement et t le temps où D_{n+1} est la position à t_{n+1}. D_n est la position à t_n et $t = t_{n+1} - t_n$ et correspond à la date et à l'heure de *n* et *n + 1*. Nous n'estimons la vitesse qu'à partir du système Mocap, car c'est notre vérité de base. L'objectif de qualifier l'UWB pour une localisation de personnes ou d'objets est atteint. La précision de la localisation des résultats brut est proche d'un mètre dans les pires cas, et une meilleure compréhension de celle-ci peut maintenant facilement l'améliorer. Les travaux antérieurs [25, 128, 23] montrent que l'UWB peut être très précis dans des conditions industrielles LDV, mais nous sommes dans des conditions industrielles NLDV. L'étude de la vitesse pendant un trajet peut améliorer les résultats. La dilution de la précision causée par le corps et l'environnement (béton et métal) [7] peut également avoir un impact sur le calcul de la

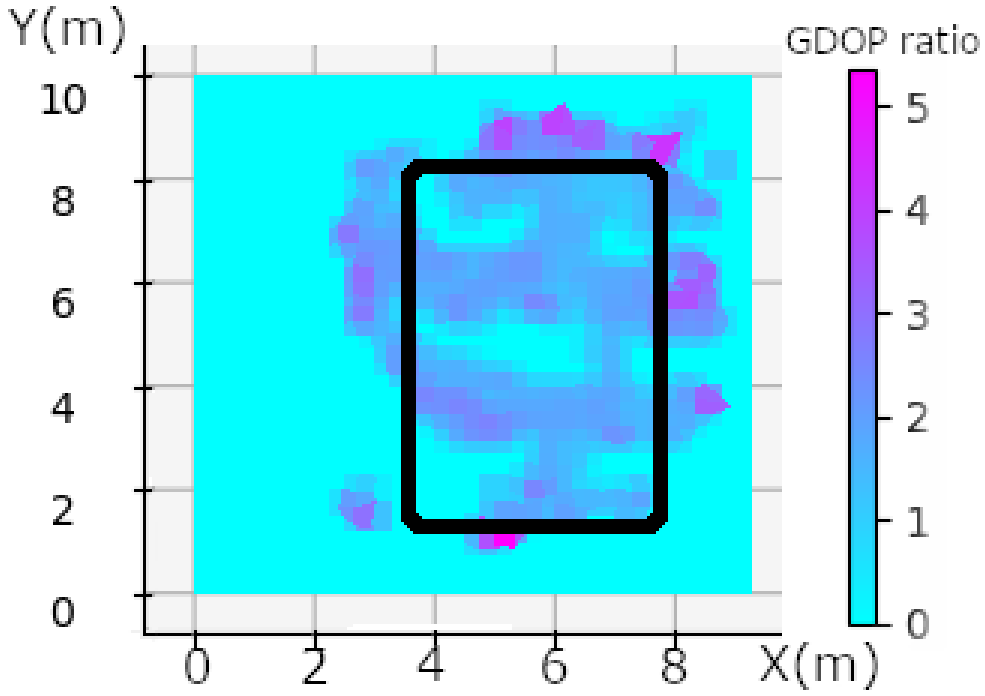


FIGURE 3.9 – Dilution géométrique du calcul de précision effectué en atelier dans les conditions industrielles en LDV. Des rectangles noirs montrent la zone où sont placés les ancrages UWB, un dans chaque coin.

triangulation.

Nous calculons la valeur de la dilution géométrique de la précision (GDOP) comme décrit dans [45],

$$GDOP = \frac{RMSE_{loc}}{RMSE_{range}} \tag{3.5}$$

avec $RMSE_{loc}$ qui correspond à l'erreur globale de position mesurée et $RMSE_{range}$ qui correspond à l'erreur théorique (10cm) de l'UWB, le résultat est indiqué dans la figure 3.9.

On peut voir une corrélation entre la vitesse de déplacement dans la figure 3.10 et le calcul de la dilution géométrique de la précision (GDOP) dans la figure 3.9. Les ancres sont placées au coin du rectangle noir de la figure 3.9. Nous pouvons voir que la pire GDOP se trouve dans un plan rectangulaire à la position des ancres, la GDOP maximale est en violet (5.5), et le cyan signifie qu'aucune donnée n'était disponible. La combinaison de la vitesse et du GDOP élevé donne toutes les explications des erreurs de calcul dans le cas NLDV. Le GDOP nous donne juste une indication de la géométrie, le calcul prend en compte la mesure du bruit parce que nous sommes dans NLDV et aussi un DOP dû à la personne qui porte le tag. Cependant, nous pouvons voir que la plus grande erreur de mesure se trouve dans le plan sur le côté du rectangle (voir la figure 3.9). Cela montre que malgré le bruit de mesure, l'environnement, la personne et la géométrie ont une influence sur le calcul de la position. La vitesse a également un impact sur le calcul de la position. Les deux combinés montrent la majorité des erreurs de position vues dans la figure 3.11. Les autres erreurs de position qui ne sont pas principalement dues à ces deux facteurs

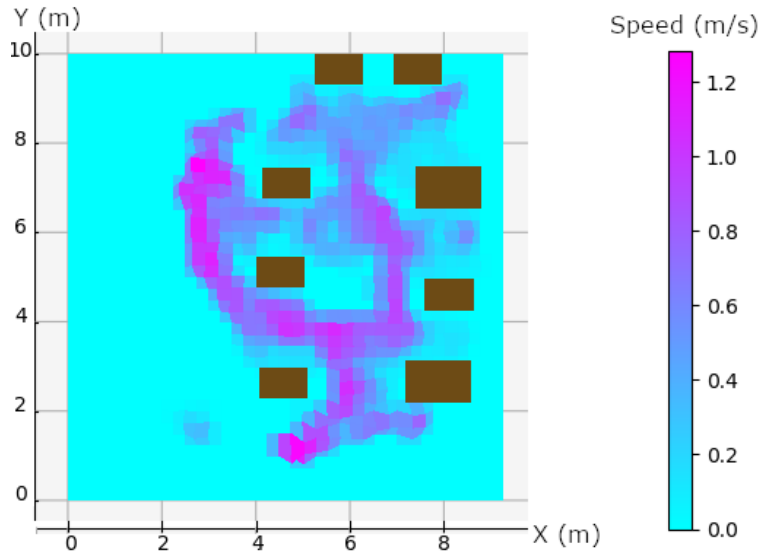


FIGURE 3.10 – Diagramme de vitesse en m/s. Les rectangles marron représentent chaque appareil de forage, les deux du haut représentent l'appareil de forage de ravitaillement. Les valeurs en violet correspondent à la zone de vitesse maximale et les valeurs en cyan à l'absence de données.

sont celles de l'environnement industriel. Les solutions qui pourraient améliorer le calcul seraient de prendre en compte la géométrie et donc le positionnement des balises, qui doivent être placées dans les coins des pièces, le plus haut possible. Lorsqu'il y a des zones où la personne va se déplacer assez vite, nous pouvons ajouter une ancre. Des algorithmes connaissant l'environnement pourront corriger les erreurs automatiquement. Ye et al. [152] ont proposé un moyen de placer des ancres et d'avoir un bon GDOP.

Amélioration des résultats

Nous proposons une première méthode pour écarter le calcul du biais dans la NLDV en utilisant le filtre Savitsy–Golay [121] qui lisse la trajectoire lorsque la vitesse est élevée. Le filtre Savitsky–Golay est défini par l'équation

$$\hat{Y}_j = \frac{\sum_{i=-m}^{i=m} C_i Y_{j+i}}{N}$$

où *Y* est la valeur de position initiale, $\hat{Y}_j$ est la valeur de position résultante, C_i est le coefficient pour la ième valeur de position du filtre, et *N* reste le nombre d'entiers convolutifs. *N* est égal à la taille de la fenêtre de lissage $(2m+1)$, avec $m \in \mathbb{N}$. L'indice *j* est l'indice courant de la table de données d'ordonnée d'origine. Le tableau de lissage est constitué de $2m+1$ points, où m est la demi-largeur de la fenêtre de lissage. Les coefficients d'un filtre Savitsky-Golay (C_i) peuvent être obtenus à partir de Steiner et al. [132]. Deux paramètres doivent être déterminés en fonction de la position comme indiqué sur la figure 3.11. Le premier paramètre est *m*, la demi-largeur de la fenêtre de lissage. Une valeur plus grande de *m* génère un résultat plus lisse. Nous avons choisi cinq points $(m = 5)$ en

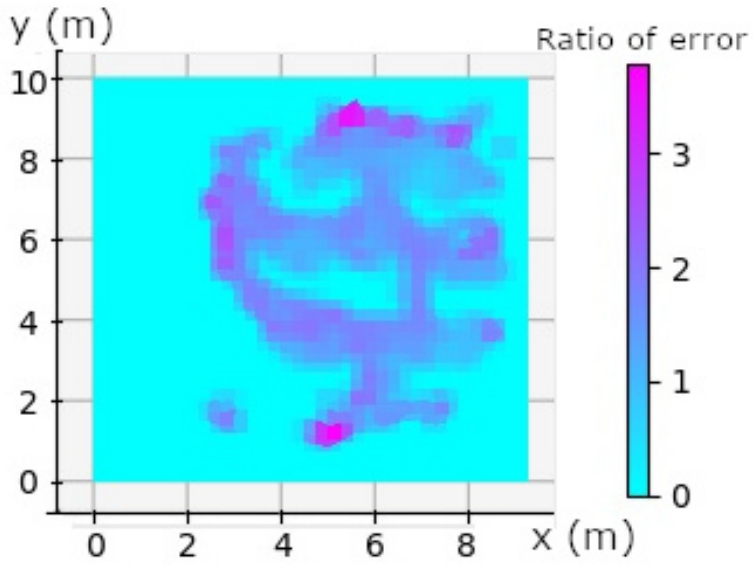


FIGURE 3.11 – Vitesse combinée et ratio GDOP. La vitesse maximale et le GDOP sont en violet (~4) ; cyan aucune donnée disponible.

TABLE 3.1 – Comparaison avec des données filtrées et des données brutes avec le système MoCap comme référence.

	Overall Experiment in Red Square Zone	**X-Axis**	**Y-Axis**	**2D**
Raw UWB data	Mean error	0.21 m	0.12 m	0.16 m
	Range	2.84 m	3.45 m	3.14 m
	Standard deviation	0.46 m	0.38 m	0.42 m
Filtered UWB data	Mean error	0.19 m	0.11 m	0.15 m
	Range	2.74 m	3.44 m	3.09 m
	Standard deviation	0.41 m	0.38 m	0.39 m

considérant que les nouvelles positions sont fournies par l'UWB à un taux de 10 Hz, la longueur combinée de la fenêtre sera de 11 points $(2m + 1)$, chaque seconde nous maintiendrons la trajectoire corrigée actuelle, ce qui est acceptable pour une application en temps réel. Le seconde paramètre représente un nombre entier (d) spécifiant le degré du polynôme de lissage. Nous choisissons $d = 1$ pour considérer une approximation linéaire et éliminer le biais. Comme le montre la figure 3.12a, lorsque le Savistsy–Golay n'est pas appliqué, on peut voir que la trajectoire est bruitée. La figure 3.12b correspondante montre la même trajectoire filtrée, celle-ci est lissée. Ce filtre améliore la précision globale de l'UWB vue dans le tableau 3.1 d'un centimètre et élimine les valeurs erronées. Nous comparons les données de l'UWB avec le système MoCap lorsque les données sont fiables. Cette méthode peut être utilisée en temps réel et ne nécessite pas d'autres capteurs (IMUs, cameras,...) tels que ceux de [138, 77, 41].

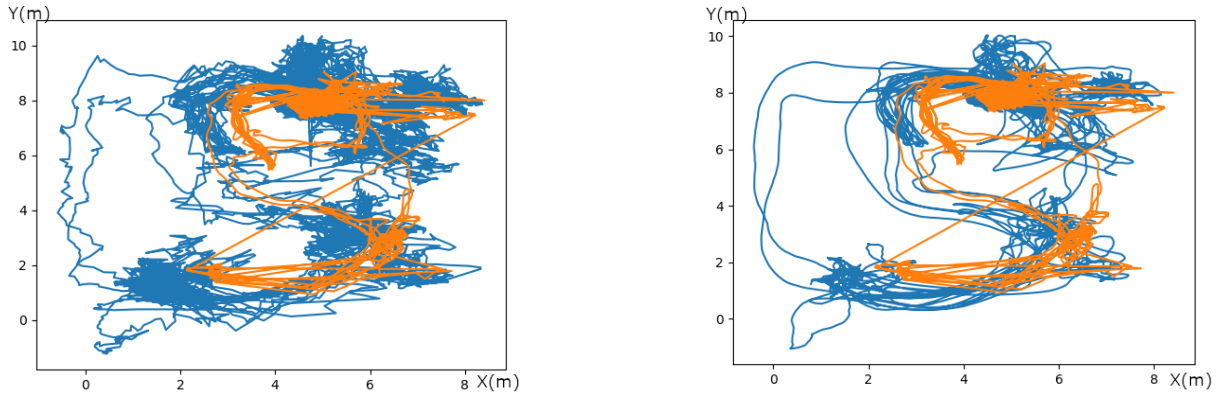


FIGURE 3.12 – Comparaison de la trajectoire du travailleur de la station 1 entre UWB filtré et non filtré en bleu. En orange, le système de capture de mouvement. (**a**) Système UWB sans filtre Sav–Gol en bleu et système de capture de mouvement en orange. (**b**) Système UWB avec filtre Sav–Gol en bleu et système de capture de mouvement en orange.

TABLE 3.2 – Comparaison des ensembles de données de localisation et de positionnement existants basés sur l'UWB et les nôtres.

Dataset	**Distance Est**	**Modalities**	**Number of Tag**	**Anchor Settings**	**Industrial Scenarii**	**UWB Node**
Cung et al.[79]	AltDS-TWR	UWB	1	4	No	DWM1000
Minne et al. [96]	ToF	UWB	6	8	No	DWM1000
Raza et al. [115]	ToF(TDOA)	UWB+BLE &UWB+MoCap	1	4	No	DWM1001
Queralta et al. [114]	ToF	UWB+MoCap	1–4	multiple	No	DWM1001
Barral et al.[8]	RSS	UWB+IMU +camera	1		No	Pozyx
Li et al. [72]	ToF	IMU+UWB +Mocap(VICON)	1	6	No	TimeDomain
Bernhard et al. [54]	ToF	UWB	1	1	No	DW1000
Le nôtre	ToF	UWB+MoCap	6	4	Yes	MDEK1001

3.4 Utilisation et interprétation

Avec cet ensemble de données, nous fournissons trois heures d'enregistrement. Six personnes en même temps sur six types de postes de travail différents avec des scénarios industriels représentatifs comparés à d'autres ensembles de données existants, comme indiqué dans le tableau 3.2.

Cette base de données peut être utilisée pour la maintenance prédictive ou le calcul du temps de déplacement (nombre de kilomètres parcouru par chaque personne). Nous pouvons également procéder à l'optimisation des locaux pour améliorer le confort des opérateurs et l'efficacité de la production. Toutes les données permettent de réaliser ou proposer de nouveaux algorithmes en utilisant UWB dans NLDV et MoCap comme vérité terrain.

Dans cette étude, nous avons défini quatre types d'erreurs potentielles d'UWB. Le premier est la dilution géométrique de la précision (GDOP), qui se caractérise par la géométrie du positionnement des ancres. Lorsque le tag se

trouve dans une zone plane de l'axe X ou de l'axe Y, une erreur plus importante sera caractérisée, comme on peut le voir sur la figure 3.9. On peut voir que l'erreur se situe autour du rectangle de positionnement des ancres et est d'environ un mètre par rapport à la figure 3.8 et avoir un GDOP compris entre 3 et 5. Selon Fevzi Aytaç Kaya et al, ils classent la notation des DOP (y compris les GDOP) entre le niveau 1 et 50. Avoir une valeur de GDOP comprise entre 3 et 5 selon [61] est bon et excellent lorsque la personne en mouvement se trouve à l'intérieur du rectangle noir sur la figure 3.9. Cette géométrie explique les erreurs proches des plans des axes X et Y.

Une deuxième cause des erreurs de position est la vitesse. Comme le montre la figure 3.10, les erreurs de position sont directement influencées par la vitesse de l'utilisateur.

La dilution de la précision (DOP), le troisième type d'erreur, provoque une erreur globale moins importante sur la position du tag ; elle est négligeable par rapport aux autres types d'erreurs. Richa Bharadwaj et al. [10] ont étudié l'effet du placement aléatoire des stations de base sur la localisation tridimensionnelle centrée sur le corps, et Andrew Fort et al. [44] montrent que le corps peut avoir un faible impact et la précision.

Le quatrième type d'erreur est la NLDV. On le voit clairement si on le compare à la figure 3.8, les erreurs de position sont bien corrélées en vitesse et en GDOP. Il y a un point d'erreur qui n'a pas été identifié par le GDOP et la vitesse, ce point d'erreur est l'endroit où il y a du métal dans la NLDV à la coordonnée (8,3).

3.5 Conclusions

Dans ce chapitre, nous avons proposé un nouvel ensemble de données pour la localisation en intérieur en Non Ligne De Vue (NLDV) avec six scénarios dynamiques dans un site industriel pendant une phase d'assemblage. Nous avons suggéré également des moyens d'améliorer notre ensemble de données afin de le comparer avec d'autres domaines de recherche. Nous avons montré que la géométrie et la vitesse ont une influence sur le calcul de la position. Nous introduisons une nouvelle façon de filtrer l'estimation de la position de l'UWB sans fusionner les données avec d'autres modalités.

Cette étude vise à mieux comprendre les erreurs d'estimation de la position en se basant sur un support UWB dans des conditions réelles d'utilisation.

Les deux modalités peuvent être mises en œuvre pour la localisation en intérieur. Le système MoCap présente une meilleure précision que l'UWB mais nécessite une infrastructure plus coûteuse et plus complexe. Pour les cas d'utilisation nécessitant une précision moindre, ou une installation plus rapide et moins coûteuse, l'UWB est un excellent choix pour accroître la sécurité dans les environnements industriels à faible coût.

CHAPITRE 4

Réseaux de graphes convolutionnels d'actions basées sur la reconnaissance de squelette pour les flux de données continus : une approche à fenêtre glissante

Ce chapitre présente une nouvelle approche de la reconnaissance de l'activité humaine basée sur l'apprentissage profond. La méthode consiste en un réseau convolutionnel de graphiques spatio-temporels fonctionnant en temps réel grâce à une approche de fenêtre glissante. L'architecture proposée consiste en une fenêtre fixe pour la formation, la validation et le processus de test avec un Réseau Convolutionnel Spatio-Temporel-Graphique pour la reconnaissance d'actions basée sur un squelette. Nous évaluons notre architecture sur deux ensembles de données disponibles de reconnaissance d'actions en continu : l'ensemble de données de détection d'actions en ligne (OAD) et les ensembles de données 3D d'actions en ligne de l'UOW (University Of Wollongong) Online Action3D. Cette méthode est utilisée pour la détection temporelle et la classification de la reconnaissance d'action effectuée en temps réel.

4.1 La méthode SW-GCN

Cette méthode est la combinaison entre un réseau de graphes spatio-temporels que nous verrons dans la section “Réseau convolutionnel de graphes spatio-temporels” et une fenêtre glissante que nous détaillerons dans la section “Une approche à fenêtre glissante”.

4.1.1 Réseau convolutionnel de graphes spatio-temporels

Le ST-GCN est un réseau neuronal qui prend en entrée les données du squelette et utilise un noyau spatio temporel pour détecter les mouvements du squelette. Cela permet au réseau de détecter et de classer différentes actions sans avoir besoin d'un algorithme lourd.

C'est pourquoi nous avons décidé de choisir le Spatial Temporal Graph Convolutional Network pour détecter l'action et caractériser le bruit autour de l'action en utilisant les fenêtres glissantes. Seules les données squelettiques ont été utilisées car elles peuvent être obtenues par des capteurs inertiels qui sont les moins chers du marché par rapport à la caméra ou au système de capture de mouvements. De notre point de vue, dans un contexte industriel, c'est le meilleur choix. Dans ce contexte les caméras seront intrusives pour les utilisateurs contrairement à des IMUs qui feront parti intégrante de la personne. Un système de capture de mouvement coutera très cher par rapport aux IMUs comme vu dans le chapitre 1.

4.1.2 Une approche à fenêtre glissante

On suppose que la taille de la fenêtre joue un rôle crucial dans cette méthode. C'est pourquoi nous choisissons une taille de fenêtre fixe qui correspond à la taille d'une action moyenne sur chaque ensemble de données. Pour l'étiquetage, nous prenons le cadre du milieu pour définir l'action correspondant à la fenêtre glissante, comme le montre la figure 4.1. Cela permet à l'algorithme de caractériser les morceaux d'actions pour chaque fenêtre avec le bruit de mesure induit par les autres actions. Le graph spatio-temporel est construit sur les séquences du squelette en deux étapes. Tout d'abord, les articulations à l'intérieur d'un squelette sont reliées par des bords selon la connectivité de la structure du corps humain, une matrice est donnée à l'algorithme pour définir cette connectivitée 4.2. Ensuite, chaque articulation sera connectée à la même articulation dans la frame consécutive. Les connexions dans cette configuration sont donc naturellement définies sans l'affectation manuelle des articulations. Cela permet également à l'architecture de notre réseau de travailler sur des ensembles de données avec un nombre différent d'articulations ou de connectivité d'articulation [151]. Dans le cadre de la méthode de la fenêtre glissante combinée avec le ST-GCN (Spatio Temporal-Graph Convolutionnal Network), il est possible de capturer en temps réel des informations de mouvement dans des séquences de squelette dynamiques. Notre fenêtre glissante est fixée à la même taille pendant la phase d'entrainement et la phase de test. La taille est déterminée par la durée moyenne d'une action pour chaque ensemble de données indiqué dans la figure 4.1. Le décalage de la fenêtre glissante est d'une frame par une frame.

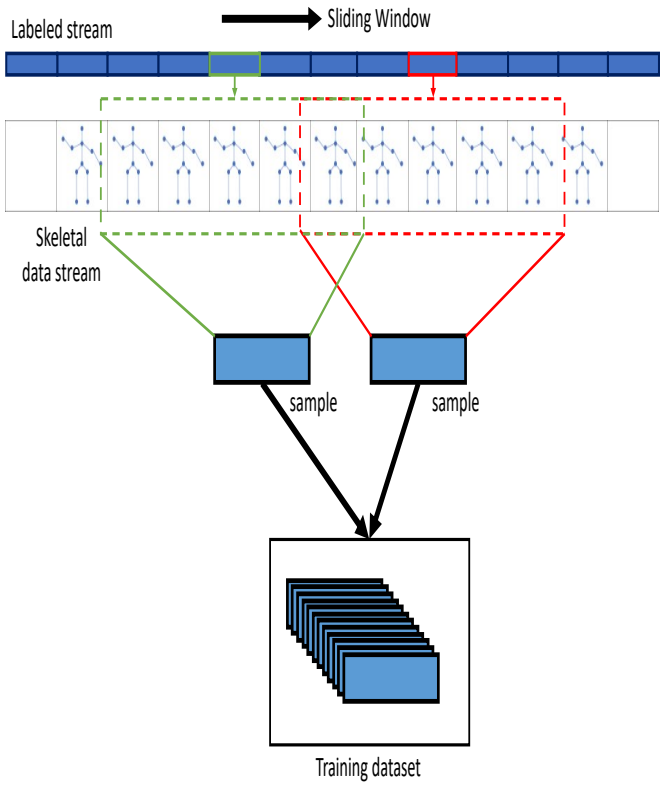


FIGURE 4.1 – Pré-traitement de l'agencement du Squelette avec chaque joint utilisé et la fenêtre glissante labélisée.

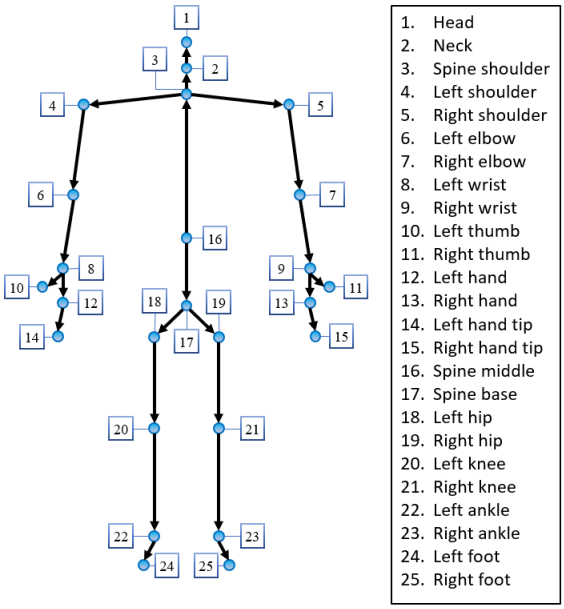


FIGURE 4.2 – Schéma squelette avec chaque articulation utilisée pour les deux ensembles de données (25 articulations)

Methode	Exactitude	F1-score
SW-CNN	0.680	0.680
SW-GCN	**0.755**	**0.750**

TABLE 4.1 – Résultat de l'action en ligne UOW Action 3D Comparaison entre le score F1 et l'exactitude

4.2 Expérimentations

La méthode proposée est évaluée sur deux ensembles de données complexes : l'ensemble de données de détection d'action en ligne [78] et l'ensemble de données 3D d'action en ligne de l'université Wollongong (UOW) [135] qui comportent des séquences en ligne entières et non segmentées. Dans tous les ensembles de données, les actions multiples sont contenues dans des séquences de vidéos avec des données squelettes. Nous avons donc utilisé les données BVH qui correspondent aux données squelettes et avons utilisé uniquement les données vidéos à des fin de démonstration visuelle que l'on peut retrouver sur GitHub [1].

La disposition du squelette avec chaque articulation utilisée est organisée comme le montre la figure 4.2. Les deux ensembles de données utilisant 25 articulations, nous avons donc utilisé la même matrice de proximité du graphique d'entrée pour le ST-GCN.

4.2.1 Expériences sur le jeu de données 3D de l'action en ligne de l'UOW

Le jeu de données 3D de l'action en ligne de l'UOW [135] contient 20 actions différentes, réalisées par 20 sujets différents avec jusqu'à 3 à 5 exécutions différentes. Pour chacune des 48 séquences, les 25 positions communes par image ont été utilisées comme entrées. Nous avons choisi cet ensemble de données au lieu de l'ensemble de données d'actions 3D du MSR [76] parce qu'il contient les mêmes actions mais il a des séquences continues d'actions qui correspondent à notre objectif.

L'ensemble de données 3D de l'action en ligne de l'UOW est récent et ne propose pas de méthode similaire à la nôtre. Nous avons donc créé une fenêtre glissante basée sur le réseau neuronal convolutif (SW-CNN)que nous avons comparé avec notre méthode SW-GCN. Le SW-CNN a été entrainé dans le cadre de l'optimiseur de ranger, qui se compose de deux éléments : Adam rectifié (RAdam) et Lookahead [86], pour 200 époques. Il se compose de quatre couches de convolution avec 40 à 160 filtres, ainsi que de 2 couches de max-pooling suivies d'une couche entièrement connectée de 100 neurones et de la fonction d'activation Mish [98]. La fonction de perte (loss function) utilisée est une fonction de perte d'entropie pondérée croisée (weighted cross-entropy loss function). Cela démontre l'efficacité de l'algorithme GCN à travers la fenêtre glissante qui est la même que les deux méthodes car si nous utilisions un optimiseur classique (descente stochastique) les résultats sont moins précis, l'algorithme n'arrive pas à classifier. C'est pourquoi nous avons été obligé de l'optimiser avec les derniers optimiseurs cités plus haut. Le SW-GCN a été entrainé sous descente stochastique de gradient pour 140 époques et se compose de 11 couches

1. https ://github.com/DelamareMicka/SW-GCN

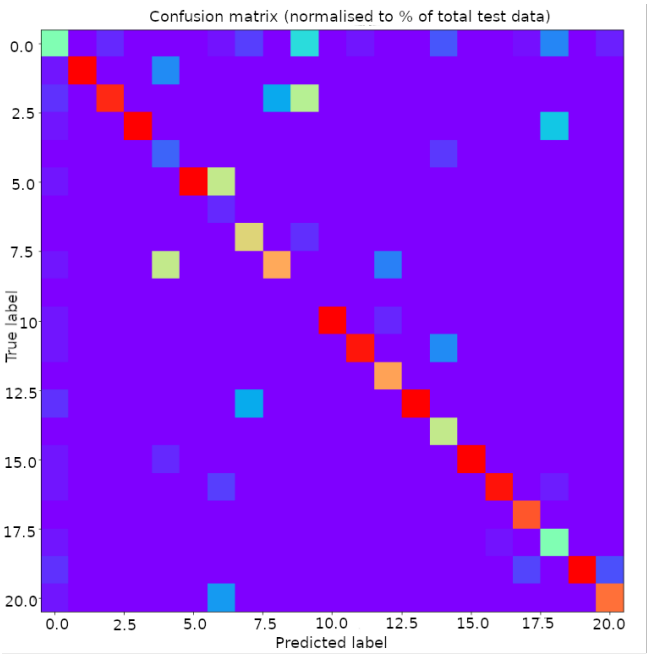


FIGURE 4.3 – Matrice de confusion de la méthode SW-GCN pour la validation

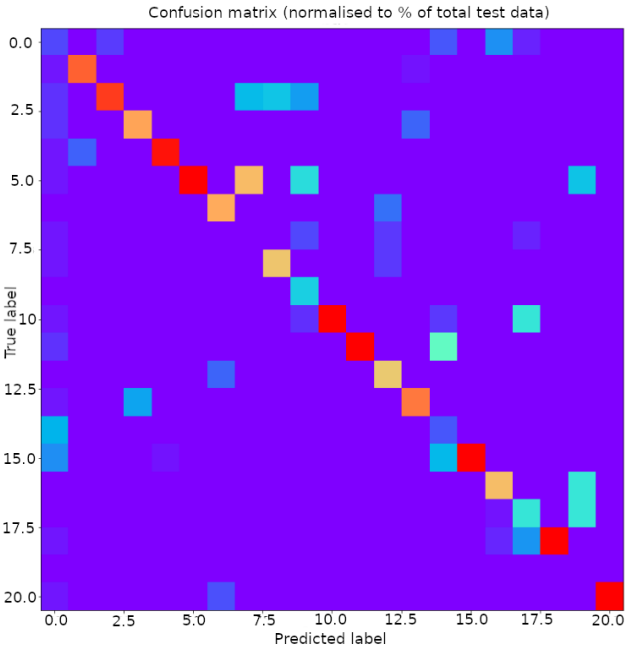


FIGURE 4.4 – Matrice de confusion de la méthode SW-CNN pour la validation

avec 32 à 128 filtres et une fonction d'activation linéaire réticente. La fonction de perte utilisée est une fonction de perte d'entropie croisée pondérée.

Les données ont été réorganisées en fenêtres de 50 trames, car il s'agit de la durée moyenne de toutes les actions dans cet ensemble de données, avec la méthode indiquée précédemment, puis chaque articulation a été séparée et les valeurs ont été recentrées autour de zéro, avant d'être divisées par trois pour les mettre entre -1 et 1 afin de les normaliser. Enfin, les données ont été séparées en un ensemble d'entraînement de 46 séquences, un ensemble de validation de 1 séquence continue, et un ensemble de test de 1 séquence continue qui comprennent toutes les actions à classifier. Cette répartition a été réalisée pour se concentrer sur l'entrainement et pour valider notre modèle nous avons gardé une séquence, cette séquence n'a jamais été utilisée dans l'algorithme. Ce parti pris de répartition de dataset n'est pas la norme, généralement il est préférable de répartir en 80% /20 % mais cela dépend du jeu de données, dans notre cas c'est un grand jeu de données. Le fait de séparer notre jeu de données de cette manière permet de vérifier que notre algorithme arrive à généraliser correctement, avec moins d'information en phase de test.

Cet ensemble de données nous permet de montrer l'efficacité de notre nouvelle approche. Le SW-GCN a un score F1 de 0.75 tandis que le SW-CNN a un score de 0,68 comme le montre le tableau 4.1. Cependant, le SW-CNN est 10 fois plus rapide que notre méthode (1,63 ms au lieu de 10 ms), mais l'utilisation du CNN peut donner lieu à une prédiction avec des faux positifs car les entrées ne sont qu'une simple matrice. L'utilisation du GCN fournit des informations sur les entrées sous forme de matrice squelette qui est beaucoup plus fiable et ne crée pas de faux positifs comme on peut le voir sur les matrices de confusions 4.3 et 4.4.

Actions	SVM-SW [78]	RNN-SW [161]	CA RNN [78]	JCR RNN [78]	**SW-GCN (Notre méthode)**
Boire	0.15	0.44	**0.58**	0.57	0.09
Manger	0.47	0.55	0.56	0.52	**0.84**
Ecrire	0.65	0.86	0.75	0.82	**0.92**
Ouvrir le placard	0.30	0.32	0.49	0.50	**0.89**
Se laver les mains	0.56	0.67	0.67	0.71	**0.78**
Ouvrir le micro-ondes	0.60	0.67	0.47	0.70	**0.78**
Balayer	0.46	0.59	0.60	0.64	**0.93**
Se gargariser	0.44	0.55	0.58	0.62	**0.95**
Jeter des ordures	0.55	0.674	0.43	0.46	**0.88**
Essuyer	0.86	0.75	0.76	0.78	**0.96**

TABLE 4.2 – Comparaison sur l'ensemble de données OAD F1-Score pour chaque classe

Methode	Exactitude
ST-LSTM [83]	0.77
AttentionNet [84]	0.75
JCR-RNN [78]	0.79
FSNet [85]	0.80
SSNet [85]	0.82
SW-GCN (Notre méthode)	**0.90**

TABLE 4.3 – Comparaison sur l'ensemble des données OAD Précision globale

4.2.2 Expériences sur l'ensemble de données OAD

L'ensemble de données OAD [78] contient de longues séquences correspondant à 700 séquences d'actions avec dix classes d'actions collectées avec une caméra Kinect v2. Les données ont été réorganisées exactement comme le dataset d'action en ligne UOW.

Le ST-GCN a été entrainé à la descente stochastique par gradient pour 140 époques, et se compose de 11 couches avec 32 à 128 filtres et une fonction d'activation linéaire réticente. La fonction de perte utilisée est une fonction de perte d'entropie croisée pondérée.

Les auteurs [85] ont obtenu une précision (accuracy) globale de 82%, avec notre méthode, nous avons une meilleure précision à 90% pour la détection d'action en ligne dans l'ensemble. Nous avons une meilleure précision en comparant le F1-score de chaque action sauf pour une seule action (Boire). Cela peut s'expliquer par le fait que l'action "Boire" n'est pas bien reconnue par notre méthode et ne peut être détectée autour du bruit. Cela est principalement dû à la taille de la fenêtre.

4.2.3 Évaluation de notre méthode

Les mesures ont été réalisées sur un ordinateur portable équipé d'un processeur i7-8750H et d'une carte graphique GTX 1070. Pour les deux ensembles de données, nous avons mesuré à la fois le temps d'inférence et le débit du réseau. Le temps d'inférence a été mesuré après le réchauffement d'un GPU et a été mesuré pour 300 répétitions. Le temps d'inférence moyen sur l'ensemble de données UOW OnlineAction 3D était de 10,1 ms et le temps d'inférence moyen sur l'ensemble de données Online Action Detection était de 11,2 ms

Le débit du réseau a été mesuré sur une seconde et était de 2544 répétitions pour l'ensemble de données UOW OnlineAction 3D et de 2515 répétitions pour l'ensemble de données Online Action Detection. Le temps d'inférence pour les deux ensembles de données est d'environ 10 ms, ce qui est acceptable pour la reconnaissance d'actions en ligne en temps réel [156].

Nous avons obtenu des résultats précis avec l'ensemble de données OAD dans la reconnaissance d'actions en ligne, comme le montrent le tableau 4.2 et le tableau 4.3. Notre méthode a une meilleure précision de résultat que le SW-CNN de l'ensemble des données 3D de l'action en ligne UOW présenté dans le tableau 4.1, ce qui prouve que notre méthode peut généraliser des séquences entières de reconnaissance d'action. L'algorithme est capable de caractériser une action même si l'action est bruyante.

Pour illustrer l'objectif de notre méthode de reconnaissance d'action, les résultats de la validation sont présentés sur la figure 4.5 pour le jeu de données OAD et sur la figure 4.6 qui montre les prédictions complètes de la séquence de validation par rapport à la séquence de vérité terrain avec une précision de 90 %. Il s'agit d'un graphique de toutes les actions détectées en temps réel. Pour valider notre méthode, sur la figure 4.7 une séquence de test a été réalisée et nous avons eu une précision de 91%. Sur la figure 4.8, la phase de test correspond à la séquence de vérité de terrain. La liste des classes est la suivante [0 : aucune action, 1 : boire, 2 : manger, 3 : écrire, 4 : ouvrir un placard, 5 : se laver les mains, 6 : ouvrir un four à micro-ondes, 7 : balayer, 8 : se gargariser, 9 : jeter des déchets, 10 : essuyer].

Pour le jeu de données UOW, les résultats de la validation sont visibles sur la figure 4.9 pour le jeu de données OAD et sur la figure 4.10 qui montre l'ensemble des prédictions de la séquence de validation par rapport à la séquence de vérité de terrain avec une précision de 75 %. Il s'agit d'un graphique de toutes les actions détectées en temps réel. Pour valider notre méthode, sur la figure 4.11, un test a été effectué, et sur la figure 4.12, la phase de test qui ajuste la séquence de vérité de sol en vert avec une précision de 73%. La liste des classes est la suivante : [0 : aucune action, 1 : bras levé, 2 : bras horizontal, 3 : marteau, 4 : réception de la main, 5 : coup de poing avant, 6 : lancer haut, 7 : tirage au sort, 8 : coche de tirage au sort, 9 : cercle de tirage au sort, 10 : frappe de la main, 11 : deux mains, 12 : boxe latérale, 13 : flexion, 14 : coup de pied avant, 15 : coup de pied latéral, 16 : jogging, 17 : swing de tennis, 18 : service de tennis, 19 : swing de golf, 20 : ramasser et lancer].

Pour les deux ensembles de données, nous avons une représentation pertinente de la reconnaissance d'action

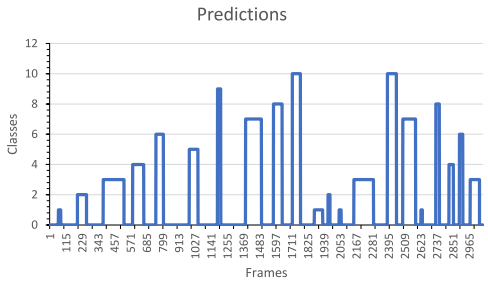


FIGURE 4.5 – Prédictions de la méthode SW-GCN avec la séquence de validation en bleu pour l'ensemble de données OAD avec une précision de 90 %.

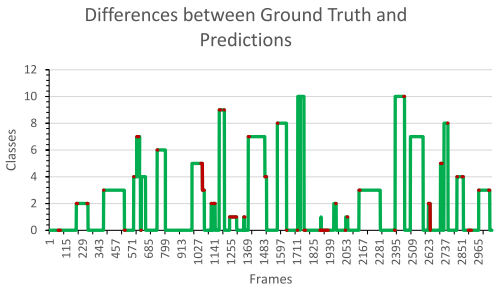


FIGURE 4.6 – Erreurs de prédictions de la méthode SW-GCN mises en évidence en rouge avec la séquence de validation pour le jeu de données OAD. En vert les prédictions qui correspondent à la vérité terrain.

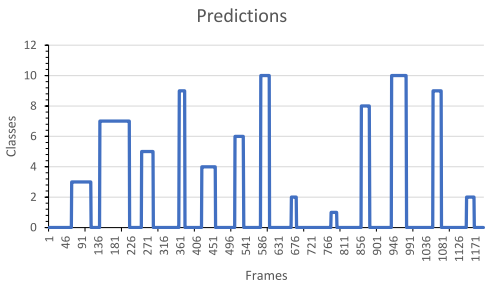


FIGURE 4.7 – Prédictions de la méthode SW-GCN avec la séquence de test en bleu pour le jeu de données OAD avec une précision de 91%.

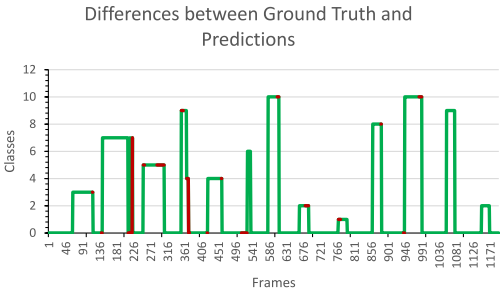


FIGURE 4.8 – Erreurs de prédictions de la méthode SW-GCN mises en évidence en rouge avec la séquence de validation pour le jeu de données OAD. En vert les prédictions qui correspondent à la vérité terrain.

continue en temps réel. Nous avons également produit une séquence vidéo pour montrer en temps réel notre solution SW-GCN. Toute la matrice de confusion est disponible dans le dépôt Github ainsi que le code pour reproduire notre méthode sur le site GitHub [2].

2. https ://github.com/DelamareMicka/SW-GCN

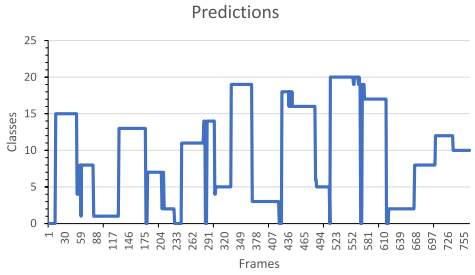


FIGURE 4.9 – Prédictions de la méthode SW-GCN avec la séquence de validation en bleu pour le jeu de données UOW avec une précision de 75 %.

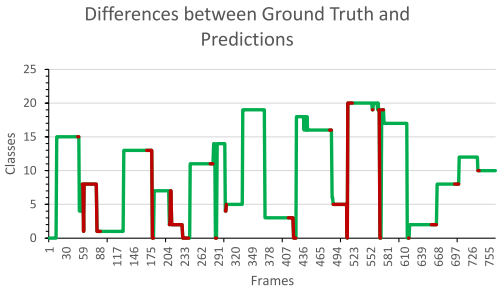


FIGURE 4.10 – Erreurs de prédictions de la méthode SW-GCN mises en évidence en rouge avec la séquence de validation pour le jeu de données UOW. En vert les prédictions qui correspondent à la vérité terrain.

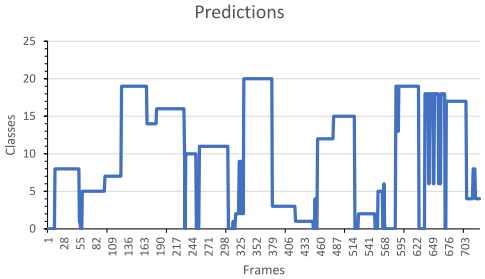


FIGURE 4.11 – Prédictions de la méthode SW-GCN avec la séquence de test en bleu pour le jeu de données UOW avec une précision de 73 %.

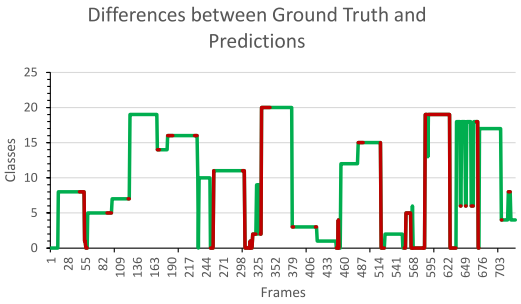


FIGURE 4.12 – Erreurs de prédiction de la méthode SW-GCN mises en évidence en rouge avec la séquence de test pour le jeu de données UOW. En vert les prédictions qui correspondent à la vérité terrain.

4.3 Conclusion

Dans ce chapitre, il a été montré que l'approche des fenêtres glissantes couplée aux réseaux spatio-temporels de graphes convolutifs permet de tirer parti du maximum de performances de ce réseau puisque celui-ci utilise les informations temporelles du squelette et peut caractériser le bruit autour de l'action pour déterminer avec précision l'action dans les fenêtres glissantes. Nous avons montré que la fenêtre glissante est une approche pertinente pour la reconnaissance d'action en ligne en temps réel avec des flux de données continus et qu'elle ne nécessite pas de processus puissant comme deux algorithmes de détections : un pour la segmentation du flux de données, un autre pour la reconnaissance d'action. Notre méthode ne fournit qu'un seul algorithme. Et il peut être intégré dans une petite unité de contrôle électronique (UCE) pour permettre une inférence rapide de l'action en cours.

L'une des limites est la taille de la fenêtre glissante, elle est efficace lorsque nous connaissons la durée moyenne d'une action. Nous avons validé notre méthode avec deux ensembles de données de pointe avec une action commune de mouvement en temps réel, et nous avons montré une meilleure performance par rapport à l'état de l'art.

Nos futurs travaux porteront sur une fenêtre glissante variable qui permet de connaître plusieurs actions de durée différente. Le principal défi est la quantité de données, l'ensemble de données InHard [20] correspond à l'objectif de reconnaissance d'actions dans les sites industriels. Mais, il faut beaucoup plus de données pour la détection d'actions généralisées. Ce sera une partie de notre travail d'élargir cet ensemble de données. Notre méthode peut également être améliorée en utilisant des ST-GCN améliorés, comme le montre notre état de l'art sur les ST-GCN. Dans le prochain chapitre, nous verrons comment notre méthode fonctionnera sur des données industrielles réelles.

CHAPITRE 5

Combinaison entre la localisation en intérieur et reconnaissances d'actions par apprentissage profond avec une approche à fenêtre glissante

Dans ce chapitre, la localisation intérieure à bande ultra-large est combinée à la reconnaissance des actions industrielles. Cet essai se concentre sur cette combinaison car elle répond à un certain nombre de problèmes industriels, tels que la protection d'un opérateur sur son lieu de travail. Dans notre cas, nous serons en mesure de détecter les mouvements d'une personne en temps réel et en continu. Cela améliorerait grandement l'interface entre un robot et un humain, permettant au robot de comprendre où se trouve l'opérateur dans le processus, lui permettant de l'assister et, par conséquent, de renforcer la synergie entre les deux. Nous avons mis notre approche SW-GCN à l'épreuve avec un ensemble de données déséquilibrées qui imite une situation d'assemblage du monde réel. Nous avons amélioré cette méthode en la combinant avec l'UWB, ce qui nous permet de fournir des informations de position et de comportement en temps réel.

5.1 Introduction

La robotique, en particulier dans les usines, a conduit à des interactions plus étroites entre les hommes et les machines.

Ce développement s'accompagne d'une demande croissante d'interfaces homme-machine (IHM) basées sur l'interaction naturelle tout en restant intuitives et efficaces. Dans ce cas, la start-up SIAtech conçoit et développe des dispositifs innovants basés sur la perception gestuelle. Son but est de séparer les organes de contrôle des

humains et des machines afin qu'ils puissent les contrôler directement avec leur corps. Un de nos objectifs est d'avoir un dispositif non intrusif tels que les capteurs MEMS, afin d'avoir une immersion complète de la collaboration homme-machine, comme un sixième sens.

Nous avons, par les chapitres précédent choisi une technologie qui permet une localisation précise et robuste de la personne dans un milieu industriel fortement perturbé [24] [25] [26]. C'est pourquoi nous allons utiliser l'UWB pour localiser la personne dans un milieu perturbé afin de connaitre son emplacement en temps réel. Nous allons aussi utiliser notre algorithme de détection d'actions appelée "SW-GCN" [27].

Nous avons donc continué les expérimentations de notre méthode de reconnaissance d'actions sur des données industrielles qui correspondent à notre objectif. Dans une première étape nous allons rappeler les résultats de notre méthode avec l'état de l'art pour justifier notre choix de continuer avec notre méthode. Dans une deuxième étape nous allons montrer les résultats sur un jeu de données réel industriel pour mettre en évidence les paramètres pour qu'un algorithme de Deep Learning puisse reconnaître des actions. Dans un troisième temps nous verrons les avancées pour réaliser une détection d'actions dans un milieu industriel et pourquoi nous avons besoin d'une localisation précise en intérieur.

5.2 La Méthode SW-GCN

Sur la base des travaux de [151], nous avons proposé un réseau neuronal convolutif à graphe spatio-temporel utilisant une approche à fenêtre glissante pour résoudre la problématique de détection d'actions avec des données continues en ligne. Plusieurs méthodes utilisent Open-pose ([15] [129] [14] [147]) lorsque des vidéos sont disponibles. Dans notre cas nous préférons utiliser des données squelettes qui proviennent d'IMU (Inertial Measurement Unit) qui ne sont pas intrusives par rapport aux caméras. Cependant notre méthode pourra elle aussi être utilisée par des caméras.

Généralement avant d'utiliser les ST-GCN, lorsqu'on se sert d'une camera, on applique Open-pose pour extraire un squelette, et plus précisément les caractéristiques de chaque noeud. Les caractéristiques envoyées en entrée de l'algorithme ST-GCN sont les noeuds et les vertices (nodes and edges) du squelette.

Le graphe est une collection de nœuds et de vertices comme on peut le voir sur la figure 5.2. Les noeuds stockent les données, Les vertices représentent la relation entre les noeuds. Les graphes sont utilisés dans la création de réseaux sociaux et de cartes. Nous les représentons en utilisant une matrice appelée matrice d'ajencement. Comme le montre la figure 5.1, nous avons 17 Joints dans notre matrice et cela correspond à notre IMU, avec 3 canaux (X,Y,Z).

Notre méthode permet donc de directement avoir les données squelettes du modèle sans passer par un algorithme de pré-traitement pour extraire les données squelettes. Comme décris dans [151] une pondération plus importante est faite pour les vertices (edge) du squelette utilisant trois décompositions de la matrice adjacente A.

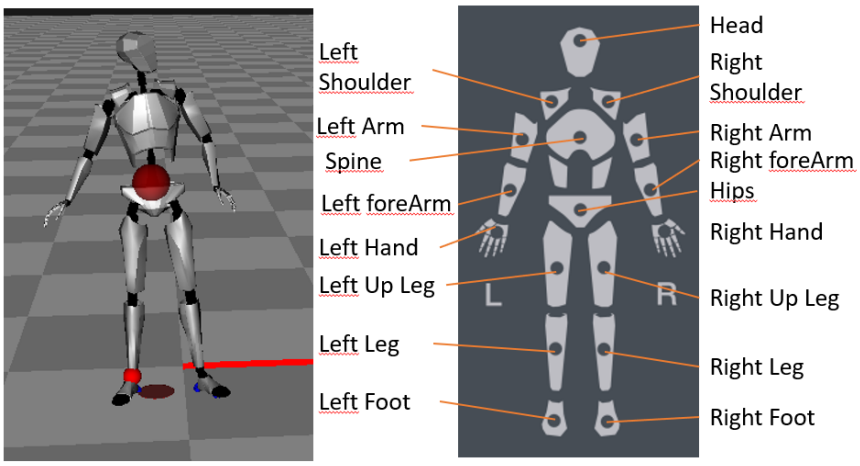


FIGURE 5.1 – Disposition du squelette avec chaque articulation utilisée pour les deux jeux de données (17 articulations).

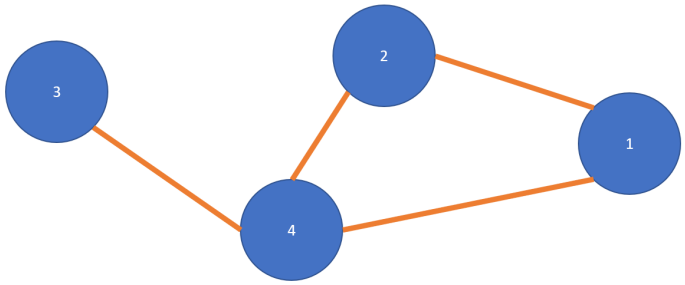


FIGURE 5.2 – Exemple d'un graph, les cercles bleus sont les noeuds, les traits oranges sont les arêtes qui correspondent aux liaisons.

La suite de l'explication du réseau ST-GCN se base sur l'article de Yan et al. [151].

L'ensemble des vertices (edges) E est composé de deux sous-ensembles, le premier sous-ensemble représente la connexion intra-squelette à chaque trame, dénommée $ES = v_{ti}v_{tj}|(i,j) \in H$, où H est l'ensemble des articulations du corps humain naturellement connectées. Le deuxième sous-ensemble contient les vertices inter-frame, qui relient les mêmes articulations dans des frames consécutifs, désignés par $EF = v_{ti}v_{(t+1)i}$. Par conséquent, toutes les arêtes dans EF, pour une articulation i particulière représenteront sa trajectoire dans le temps.

Le modèle de base pour un réseau de convolution graphique (Graph CNN model) est expliqué ici avec une seule trame pour simplifier la compréhension. Dans ce cas nous prenons une trame au temps τ, il y aura N noeuds conjoints Vt ainsi que les vertices (edges) du squelette noté $E_s(\tau) = V_{ti}Vtj|t = \tau, (i,j) \in H$. L'opération de convolution sur les images naturelles 2D peut être traitée comme des grilles 2D (Graph). L'entrée et la sortie ont la même taille, la taille du noyau est de dimension $K \times K$, une entrée caractérisée sera écrite f_{in} avec un nombre c de canaux.

$$f_{out}(x) = \sum_{h=1}^{K} \sum_{w=1}^{K} f_{in}(\mathbf{p}(\mathbf{x}, h, w)).\mathbf{w}(h, w), \tag{5.1}$$

Où la fonction d'échantillonage **p** : $Z^2 \times Z^2 \Rightarrow Z^2$ énumère les voisins du point x. La fonction de pondération **w** : $Z^2 \Rightarrow \mathbb{R}^c$ fournit un vecteur de poids dans l'espace réel de dimension c pour calculer le produit intérieur avec les vecteurs de caractéristiques d'entrée échantillonnés de dimension c.

Yan et al. [151] definissent la fonction d'échantillonage pour les Graphs en analogie avec les pixels voisins, ils définissent les noeuds voisins des Graphs tel que :

$$\mathbf{p}(V_{ti}, V_{tj}) = V_{tj}. \tag{5.2}$$

où v_{ti} est un noeud, pour une dimension $D = 1$. Le but de cette fonction est de trouver la longueur minimale entre deux noeuds.

La fonction de pondération w est définie par Yan et al. [151] :

$$\mathbf{w}(V_{ti}, vtj = \mathbf{w'}(l_{ti}(v_{tj})). \tag{5.3}$$

Où V_{ti} et V_{tj} sont toujours deux noeuds à une dimension D=1, l_{ti} qui fait correspondre un nœud voisin à son label de base. La force de cet algoritme est justement les stratégies de partitions de la fonction de pondération, plus de détails seront donnés dans la section suivante.

L'équation 5.1 devient donc :

$$f_{out}(V_{ti}) = \sum_{V_{tj} \in B(V_{ti})} \frac{1}{Z_{ti}(v_{tj})} f_{in}(v_{tj}) \dots \mathbf{w}(l_{ti}(v_{tj})), \tag{5.4}$$

Ce qui nous donne l'équation spatiale des Graphs convolutif. Où le terme normalisateur

$$Z_{ti}(V_{tj}) = |V_{tk}| l_{ti}(V_{tk}) = l_{ti}(V_{tj})| \tag{5.5}$$

est égal à la cardinalité du sous-groupe correspondant et **w** est similaire au noyau de la convolution 2D. Pour l'algorithme ST-GCN, le processus de la fonction de pondération est simplifié en trois stratégies. La première stratégie est de labelliser les nœuds qui sont identifiés en fonction de leur distance au centre de gravité du squelette (Distance partitioning). La deuxième est que chaque articulation est repérée par rapport à leur articulations voisines (Uni Labeling). Le squelette du corps est spatialement localisé, cette configuration spatiale spécifique est utilisée dans le processus de partitionnement en trois sous groupes : le noeud racine, le groupe centripète qui constitue les noeuds voisins les plus proches du centre de gravité du squelette, et le groupe centrifuge où la coordonnée moyenne de

toutes les articulations du squelette est traitée comme son centre de gravité (Spatial configuration). Ce qui donne :

$$l_{ti}(v_t j) = \begin{cases} 0, \text{si} r_j = r_i \\ 1, \text{si} r_j < r_i \\ 2, \text{si} r_j > r_i \end{cases}$$

où r_i est la distance moyenne entre le centre de gravité et l'articulation i sur toutes les frames de l'entrainement.

Cette méthode de fenêtre glissante a été améliorée en ajoutant le vote majoritaire pondéré, comme décrit dans le document [28] et peut s'écrire comme suit :

$$\hat{y} = \underset{\text{i}}{argmax} \sum_{j=1}^{N_v} \alpha_j C_j \tag{5.6}$$

où $\hat{y}$ est l'étiquette de la classe attendue à la trame t, Nv est le nombre de prédictions ou de votes, α_j est le coefficient de pondération du classificateur, et Cj est la sortie du vecteur du classificateur des étiquettes de classe K, et i correspond à la séquence d'action. Le nombre de prédictions ou de votes pour une image donnée, Nv, est déterminé comme suit :

$$N_v = \frac{W+1}{SW_{step}} \tag{5.7}$$

où W est la taille de la fenêtre glissante et SW_{step} est la fenêtre glissante actuelle.

5.3 Calcul des erreurs

Pour évaluer notre méthode, nous utiliserons ces paramètres comme suit : Le *support* est le nombre d'occurrences de chaque classe de référence (correctes).

L'exactitude est le pourcentage de cas où le modèle a prédit la valeur correcte. Elle est exprimée en pourcentage.

La *précision* est la fraction des vrais positifs (tp) sur le total de la somme des vrais positifs et des faux positifs (fp), ce qui donne :

$$Precision = \frac{tp}{tp+fp} \tag{5.8}$$

La *Rappel* est la fraction des vrais positifs (tp) sur le total de la somme des vrais positifs et des faux negatifs (fn), ce qui donne :

$$Rappel = \frac{tp}{tp+fn} \tag{5.9}$$

Le *F1-score* est la moyenne harmonique de la précision et du rappel. Ce qui donne :

$$F = 2 \times \frac{precision \times Rappel}{precision + Rappel} \tag{5.10}$$

5.4 Experimentation données industrielles sans localisation

Comme évoqué dans le chapitre précédent nous avons testé notre algorithme SW-GCN sur deux datasets de l'état de l'art [27]. Nous avons obtenu 90% de réussite sur le dataset OAD [78] et sur le dataset UOW. Si notre algorithme a des résultats pertinents, du fait de l'utilisation des Graphs pour représenter le squelette d'une personne ainsi que la stratégie de la matrice de pondération, nous n'avons pas encore vérifié notre algorithme sur un dataset plus réaliste et proche d'actions industrielles. C'est ce que propose Dallel et al.[20], avec un dataset effectué dans un environnement industriel. Ce dataset propose 4800 échantillons d'actions industrielles, réalisé avec 16 personnes distinctes qui comprend un ensemble de données RGB+ Squelette. Nous allons utiliser uniquement les données squelettes, pour rester dans la même optique de n'utiliser que des IMUs.

Ce dataset est composé de 13 actions [0 : No action, 1 : Consulter les fiches, 2 : Tourner les feuilles, 3 : Prendre un tournevis, 4 : Poser un tournevis, 5 : Prendre devant, 6 : Prendre à gauche, 7 : Prendre une règle de mesure, 8 : Poser une règle de mesure, 9 : Prendre un composant, 10 : Poser un composant, 11 : Assembler un système, 12 : Prendre un sous-système, 13 : Poser un sous-système].

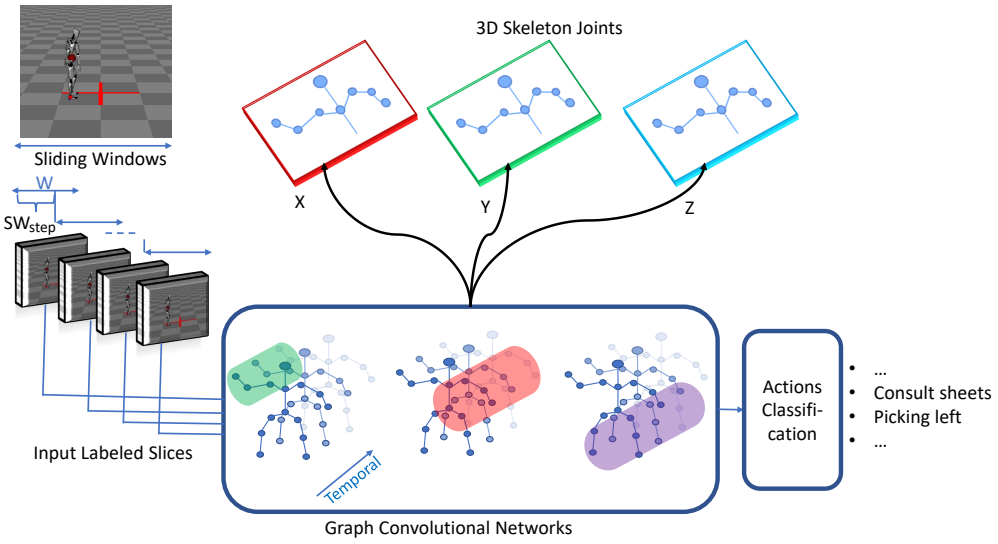


FIGURE 5.3 – Structure of SW-GCN for Online Action Recognition.

Le ST-GCN a été formé à la descente stochastique par gradient pendant 2000 époques et se compose de 11 couches avec 32 à 128 filtres et une fonction d'activation linéaire. La structure de cet algorithme est illustrée par la figure 5.3. La fonction de perte utilisée était une fonction de perte d'entropie croisée pondérée.

Comme on peut le voir dans la figure 5.4, la matrice de confusion ne comporte pas de diagonale, cela veut dire que le modèle n'a pas généralisé les actions. Cela est dû au fait que le dataset est réaliste, et il ne comporte pas le même nombre d'actions. 393 "no actions" ont été reconnue sur 1202 actions, 40% de ces "no actions" ont été re-connue". Sur la figure 5.4 c'est la première ligne. La seconde ligne sur la figure 5.4 corresponds à l'actions "Prendre

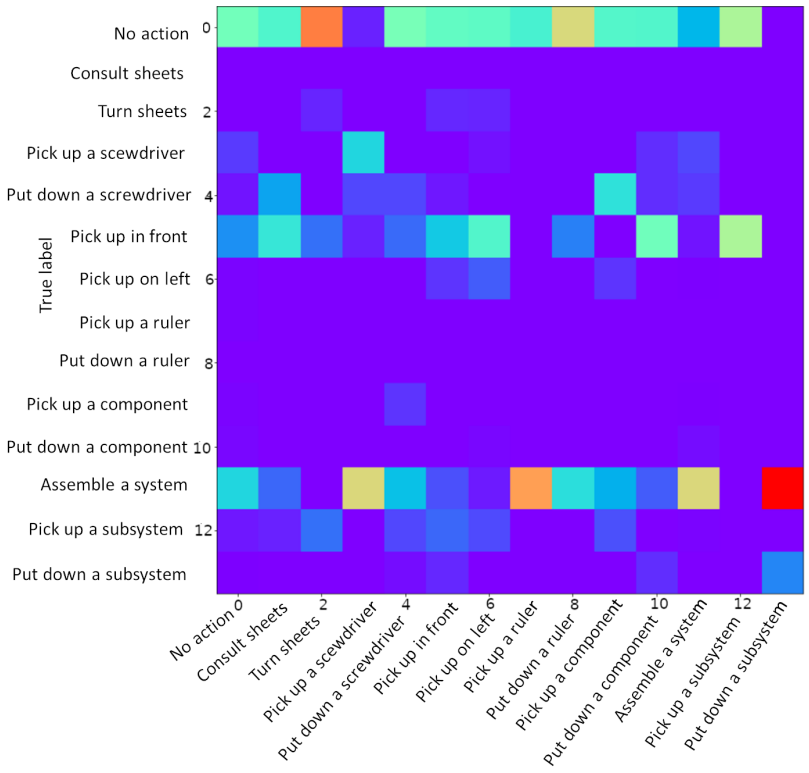


FIGURE 5.4 – Matrice de confusion réalisée sur le jeu de données InHARD à 35% d'exactitude.

devant" avec 166 actions, 25% de reconnaissances. La troisième ligne est "Prendre un système" qui comprend 404 actions reconnues soit 58% d'exactitude. On constate que ces trois lignes ont un pourcentage d'exactitude les plus élevé par rapport aux autres actions.

Ces résultats sont principalement dus au jeu de données qui comprend plus d'actions "Aucune action", "Prendre de l'avance" et "Prendre un système" par rapport aux autres actions. Nous avons utilisé notre algorithme car il est meilleur sur les jeux de données de l'état de l'art. Avec ces résultats (voir tableau 5.1), l'algorithme ne peut pas être industrialisé. Pour vérifier que le jeu de données est bien la cause d'une non-convergence de notre algorithme, nous avons réalisé un nouveau jeu de données avec les mêmes actions. Mais dans ce nouveau jeu de données, nous avons le même nombre d'actions afin que l'algorithme puisse mieux généraliser les actions. Il s'agit d'un nouveau jeu de données réalisé en réalité virtuelle, il comprend les mêmes actions que le jeu de données précédent à l'exception des "sans actions", mais ce jeu de données contient un nombre équilibré d'actions (il est "équilibré"). Cela permet en théorie à un algorithme de mieux caractériser les actions et de ne pas généraliser une seule action parce que c'est l'action prépondérante du jeu de données, comme sur l'expérience précédente.

Actions	Exactitude	F1-score	Support
No action	**40%**	**42%**	**393**
Consulter les fiches	0%	0%	0
Tourner les feuilles	4%	6%	9
Prendre un tournevis	27%	22%	73
Poser un tournevis	8%	7%	59
Prendre devant	**25%**	**14%**	**166**
Prendre à gauche	10%	16%	22
Prendre une règle de mesure	0%	0%	3
Poser une règle de mesure	0%	0%	0
Prendre un composant	0%	0%	8
Poser un composant	0%	0%	12
Assembler un système	**58%**	**53%**	**404**
Prendre un sous-système	0%	0%	43
Poser un sous-système	15%	20%	10
Actions total	35%	35%	1202

TABLE 5.1 – Tableau des résultats de chaque action industrielle sur le dataset InHARD.

Actions	Exactitude	F1-score	Support
Consulter les fiches	29%	16%	36
Tourner les feuilles	93%	94%	41
Prendre un tournevis	72%	81%	37
Poser un tournevis	93%	87%	51
Prendre devant	69%	67%	28
Prendre à gauche	68%	78%	25
Prendre une règle de mesure	42%	46%	26
Poser une règle de mesure	0%	0%	1
Prendre un composant	68%	79%	26
Poser un composant	97%	81%	47
Assembler un système	45%	44%	51
Prendre un sous-système	97%	66%	64
Poser un sous-système	0%	0%	11
Actions total	64%	65.87%	444

TABLE 5.2 – Tableau des résultats de chaque action industrielle labellisée avec la réalité virtuelle.

5.5 Expérience sur des données réelles avec localisation

Nous avons donc localisé des personnes dans un environnement industriel effectuant des actions sur la chaîne de montage.

La figure 5.5 nous montre que notre algorithme a reussi à généraliser, grâce à notre jeu de données équilibré. Cela montre que notre algorithme dans une situation réelle peut généraliser et reconnaitre des actions industrielles. Cependant on remarque que l'exactitude de la reconnaissance des gestes est de 64%, c'est certes supérieur à la moyenne mais pas suffisant dans ce cas pour industrialiser l'algorithme. Nous pouvons remarquer qu'il y a 5 actions sur 13 qui sont en dessous de 50% de reconnaissances. Tous les résultats peuvent être vus sur le tableau 5.2. Nous avons regardé en temps réel les données issues d'IMUs, et nous avons constaté une dérivation dans le temps de celles-ci. Ce phénomène n'est pas nouveau et est bien connu [2] [105]. La figure 5.6 nous

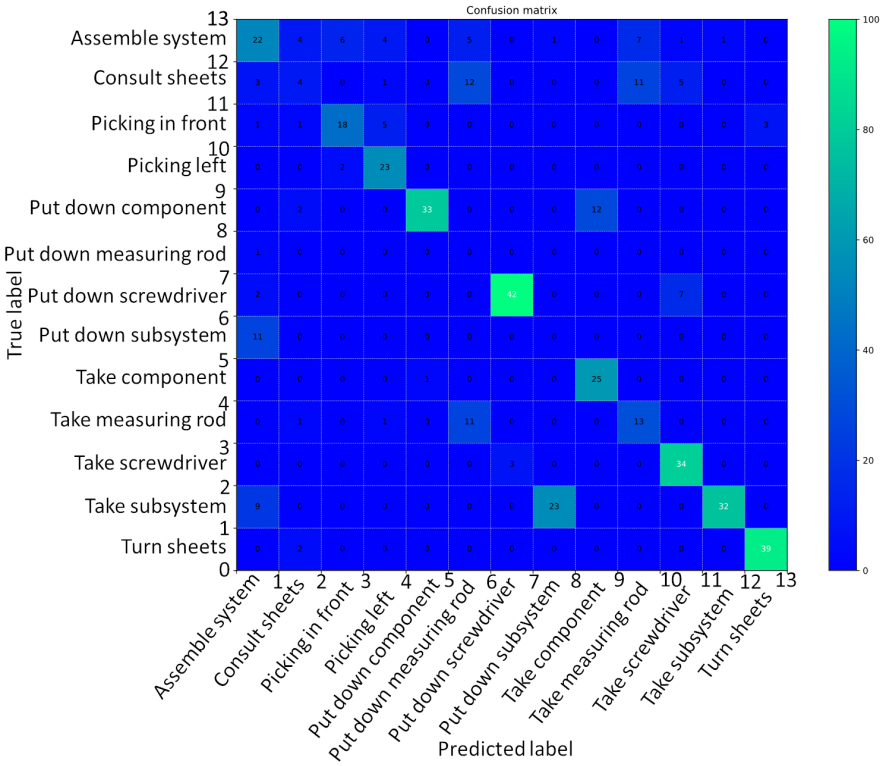


FIGURE 5.5 – Matrice de confusion des données IMUs dérivé avec 64% d'exactitude.

montre le déplacement du squelette dans le temps, et peut être constaté par une animation à ce lien [1]. Nous avons utilisé la librairy Pymo afin de manipuler les données squelettes BVH extraites des IMUs. Nous avons utilisé la fonction "RootCentricPositionNormalizer" pour centrer sur la racine (les hanches) toutes les positions des données squelettes et de les normaliser. La projection de la racine sur le plan du sol est la référence. Nous gardons cette disposition car notre but est que toutes les données squelettes ne dérivent plus. Les actions seront donc classifiées selon des mouvements statiques. Dans le tableau 5.3 on obtient un F1-score (robustesse) de 75.14%, ce qui prouve qu'on a bien enlevé l'effet du drift car on améliore de quasiment 10% la robustesse de l'algorithme avec une exactitude de 73% avec 3 actions sur 13 en dessous de 50% de détection. Ce qui veut dire que sur une chaîne de montage on peut détecter une dizaine d'actions. Cet algorithme ne peut être utilisé à des fins de sécurité pour détecter des "mauvaises actions" ou "action dangereuses" dans un contexte industriel. Pour la détection d'actions, on a donc amélioré la détection en enlevant la dérive des capteurs, cependant nous perdons une information de localisation qui est importante si on veut faire de la cobotique, ou de la sécurité industrielle. En effet la connaissance de la localisation d'une personne dans une industrie est importante, d'une part d'un point de vue de la sécurité pour intervenir rapidement sur un lieu à secourir. Et d'autre part si un robot doit intervenir pour aider un opérateur il pourra ainsi se déplacer dans l'environnement, connaître les déplacements des humains et ne pas les géner dans leurs tâches ou intervenir au bon moment, au bon endroit.

1. $https://github.com/DelamareMicka/STGCN_Fusion$

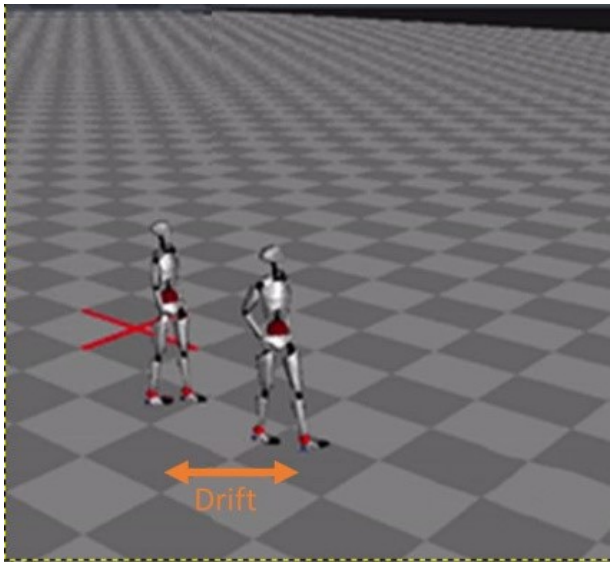


FIGURE 5.6 – Dérivation causée par les IMUs.

Actions	Exactitude	F1-score	Support
Consulter les fiches	36%	53%	5
Tourner les feuilles	100%	97%	45
Prendre un tournevis	83%	73%	60
Poser un tournevis	64%	71%	37
Prendre devant	38%	53%	12
Prendre à gauche	97%	78%	51
Prendre une règle de mesure	97%	62%	65
Poser une règle de mesure	0%	0%	1
Prendre un composant	68%	74%	31
Poser un composant	88%	77%	44
Assembler un système	63%	72%	37
Prendre un sous-système	94%	89%	37
Poser un sous-système	71%	79%	19
Actions total	73%	75.14%	444

TABLE 5.3 – Tableau des résultats de chaque action industrielle recalé selon le premier squelette.

Pour ne pas perdre la notion de localisation, dans les chapitres 2 et 3 nous avons démontré l'importance de l'UWB dans un milieu industriel très perturbé. Nous avons récuperé les données de localisation d'une personne lorsqu'elle effectuait les actions pour ce jeu de données. Nous avons appliqué le filtre de savitsy-Golay [121] afin d'avoir les données les plus fiables [26]. Nous avons donc une position statique de la personne autour de 2.16m en X et 2.18m en Y. Grâce à cette localisation nous connaissons l'emplacement précis de la personne dans l'industrie. Toutes les données squelettes ont été recalées par rapport à l'UWB. Les données squelettes sont stockées sous le format BVH (Biovision Hierarchical data) représentant les os du squelette. Ils se composent en deux parties, la première détaille la hiérarchie et la pose initiale du squelette et la deuxième décrit les canaux de données pour chaque armature (ce qui décris le mouvement). L'explication des equations BVH squelettes est tiré de [93]. On definit donc v et v' qui sont les sommets transformés et originaux respectivement, et M la matrice de transformation. On a

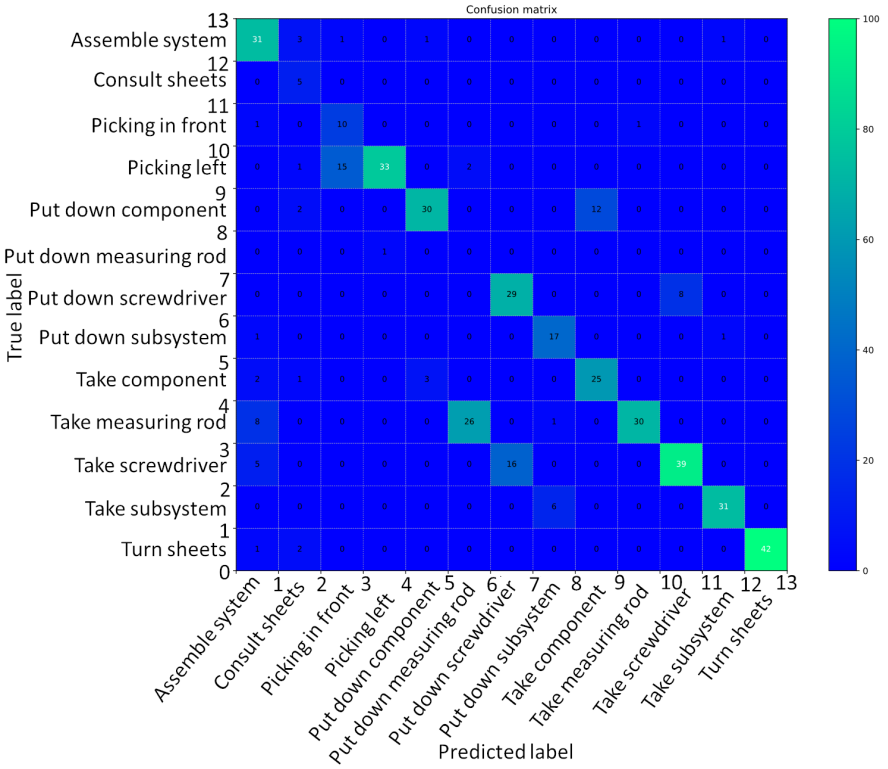


FIGURE 5.7 – Matrice de confusion des données IMUs recentré avec 73% d'exactitude.

alors :

$$v' = Mv \tag{5.11}$$

La matrice de rotation composée pour chaque os de droite à gauche (convention OpenGL), R, basée sur les matrices de rotations 3D : Rx, Ry, Rz est définit comme 5.12 :

$$Rv = R_x R_y R_z v \tag{5.12}$$

Les mouvements d'un os se compose de trois variables, une variable de translation T et de rotation R et une variable d'echelle S, La transformée d'un mouvement d'un os est définie par :

$$M = TRS \tag{5.13}$$

Dans la plupart des formats de fichiers BVH, les données sont présentées de manière hiérarchique et la formule dérivée de l'équation 5.13 ne donne que la transformation locale d'un os. La transformation locale d'un os décrit son orientation dans son système de coordonnées locale, qui à son tour est soumis aux orientations locales de son parent. Pour obtenir une transformation matricielle globale pour un os donné, la transformation locale doit être

Actions	Exactitude	F1-score	Support
Consulter les fiches	14%	25%	2
Tourner les feuilles	100%	95%	46
Prendre un tournevis	85%	81%	52
Poser un tournevis	64%	74%	33
Prendre devant	81%	76%	29
Prendre à gauche	91%	81%	43
Prendre une règle de mesure	94%	62%	62
Poser une règle de mesure	0%	0%	2
Prendre un composant	97%	77%	57
Poser un composant	35%	51%	13
Assembler un système	57%	64%	38
Prendre un sous-système	97%	65%	66
Poser un sous-système	4%	8%	1
Actions total	68%	72.95%	444

TABLE 5.4 – Tableau des résultats de chaque action industrielle recalé avec l'UWB.

pré-multipliée par la transformation globale de son parent transformée, qui est elle même dérivée de la multiplication locale et ainsi de suite. L'équation 5.14 décrit cette séquence de combinaison pour n qui est l'os actuel et l'os parent est $n-1$ et $n=0$ est l'os à la racine de la hierarchie.

$$M^n_{global} = \prod_{i=0}^{n} M^i_{local} \tag{5.14}$$

Dans notre cas, nous avons remplacé à $n=0$ les positions X et Y par ceux de l'UWB en X et en Y.Ce qui donne :

$$M^n_{global} = \prod_{i=0}^{1} M^i_{UWB} \prod_{i=1}^{n} M^i_{local} \tag{5.15}$$

où $\prod_{i=0}^{1} M^i_{UWB} = T_{UWB} R_{root} S_{root}$

Les résulats recalé avec l'UWB sont meilleurs que les résultats non recalés. L'UWB permet donc de recaler les IMUs dans le temps et de détecter des actions, l'algorithme généralise bien les actions 5.8. Nous avons dans ce test 4 actions sur 13 en dessous de 50 % de reconnaissance. On a 68% d'exactitude, c'est 5% de moins que les actions recalés sans UWB. Le système reste robuste avec seulement 2.19% de moins qu'avec les données récalés sans UWB 5.4.

Afin d'améliorer notre algorithme combinant l'UWB et la reconnaissance d'actions, nous avons utilisé un autre optimiseur différent de la descente de gradient, il s'apelle Ranger [153]. L'optimizer Ranger est un optimiseur synergique combinant RAdam (Rectified Adam) [64] et LookAhead [157], et maintenant GC (centralisation du gradient) dans un seul optimiseur [153].

Selon le tableau 5.5 nous avons 2 actions sur 13 en dessous de 50%, l'algorithme peut être utilisé et industrialisable mais ne permettra pas de savoir si l'opérateur consulte ses fiches de montage et ne saura jamais lorsqu'il pose sa règle de mesure. Cependant pour un enchainement d'actions, l'algorithme pourra détecter les autres ac-

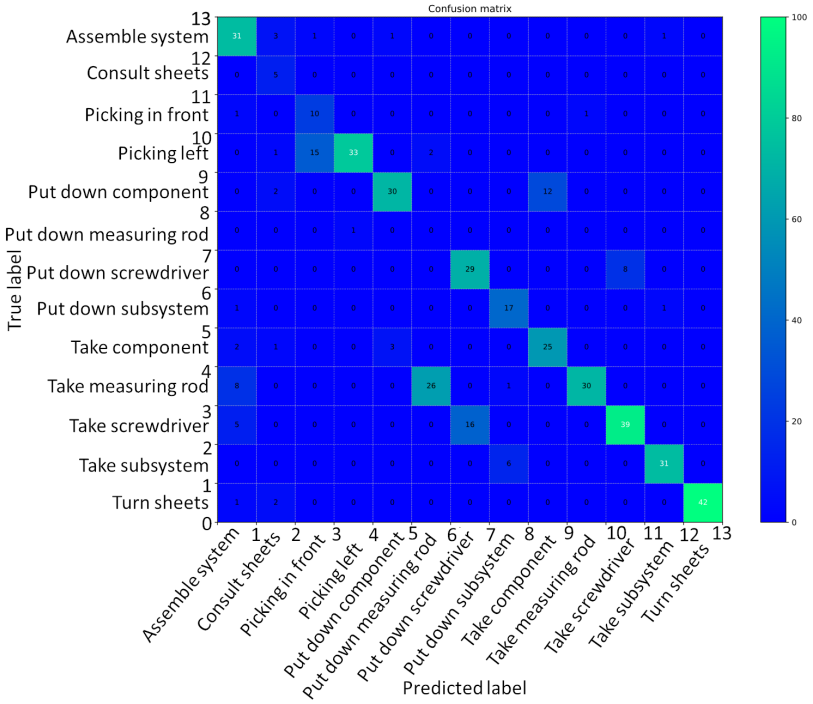


FIGURE 5.8 – Matrice de confusion des données IMUs recentré avec 68% d'exactitude.

Actions	Exactitude	F1-score	Support
Consulter les fiches	21%	22%	13
Tourner les feuilles	90%	92%	41
Prendre un tournevis	85%	81%	52
Poser un tournevis	96%	84%	60
Prendre devant	85%	72%	35
Prendre à gauche	85%	82%	37
Prendre une règle de mesure	74%	61%	44
Poser une règle de mesure	0%	0%	2
Prendre un composant	51%	64%	22
Poser un composant	94%	74%	53
Assembler un système	71%	78%	41
Prendre un sous-système	76%	75%	60
Poser un sous-système	88%	76%	31
Actions total	73%	74.76%	444

TABLE 5.5 – Tableau des résultats de chaque action industrielle recalé avec l'UWB et optimisé avec Ranger.

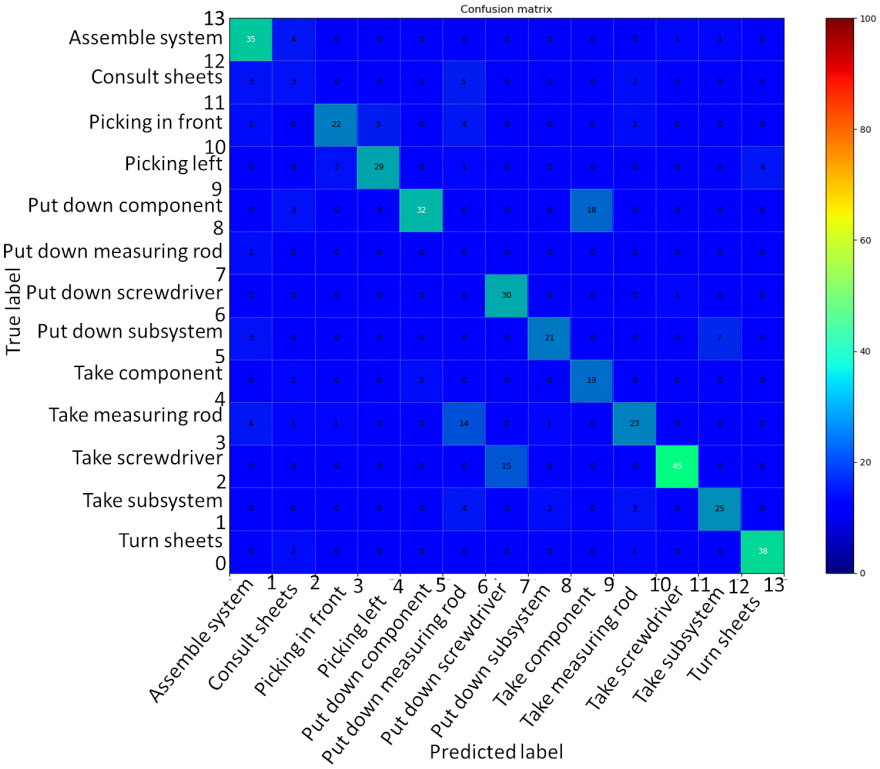


FIGURE 5.9 – Matrice de confusion des données IMUs recentré avec 73% d'exactitude.

Mesure	Sans re-calibration	Calibrage selon les hanches	Calibration avec UWB	Calibration avec UWB optimizé avec Ranger
F1-score	65.84%	**75.14%**	72.95%	74.76%
Accuracy	64%	**73%**	68%	**73%**
Information de localisation	**Oui**	Non	**Oui**	**Oui**
Actions réalisées à plus de 50%	8/13	10/13	9/13	**11/13**

TABLE 5.6 – Tableau comparatif de chaque expérience

tions avec un indice de robustesse (F1-score) de 61% pour l'action la moins bien reconnues en excluant les deux actions qui ne seront pas détectés. Sur le tableau 5.6 on voit clairement que cette combinaison est le meilleur compromis pour avoir des informations de localisation et une précision convenable pour l'application visée. Les deux actions (Consulter les fiches et poser une règle de mesure) ne sont pas reconnue car les actions sont proche d'actions parasites et ne sont pas bien détectées par l'algorithme. Dans l'ensemble, l'algorithme peut être utilisé dans un milieu complexe telle qu'une industrie avec des capteurs qui sont moins chers que les systèmes VICON [94].

5.6 Conclusion

Dans ce chapitre nous avons utilisé un nouveau jeu de données avec un système de réalité virtuelle pour améliorer l'équilibrage de données industrielles. Nous avons entraîné et testé notre algorithme sur un jeu de données en ligne industriel, qui est très réaliste et expérimental. Notre algorithme a les meilleurs résultats sur les jeux de données de l'état de l'art. C'est pourquoi nous avons gardé notre algorithme afin de le tester sur ce type de jeu de données. Nous avons constaté une évidence sur la dérive des catpteurs MEMS (IMUs), c'est pourquoi nous avons réalisé dans le chapitre 1, un état de l'art, dans le chapitre 2 et 3 une évaluation statique et dynamique en milieu très perturbé. C'est pourquoi nous avons utilisé nos résultats précédents pour les combiner à la détection d'actions. Nous avons changé la localisation statique des actions en milieu industriel avec comme référence l'UWB. Nous avons constaté que la précision de l'UWB couplé à notre méthode de filtrage permet une localisation précise qui ne détériore pas la détection d'actions, au contraire nous sommes passés de 5 actions non reconnues (en dessous de 50%) au final à 2 actions non reconnues (en dessous de 50%. Il reste néanmoins des pistes d'amélioration car nous avons une robustesse de notre algorithme de 74.76% (F1-score) avec la combinaison de l'UWB et d'un optimiseur (Ranger). Nous pouvons compléter l'algorihme SW-GCN en améliorant l'approche de répartition comme le montre [109] par exemple. Il restera aussi à prouver cette méthode en dynamique, mais cela implique d'avoir un nouveau jeu de données qui prend en compte le déplacement en dynamique des actions. Nous pouvons aussi utiliser cette combinaison afin de comptabiliser la pénibilité au travail, en connaissant l"emplacement et la répétition des actions au cours d'une journée.

CHAPITRE 6

Conclusion et perspectives

A travers cette thèse, nous avons abordé la problématique de localisation de personnes dans un milieu industriel perturbé avec pour solution technologique l'Ultra-Wide Band ainsi que la détection d'actions industrielles en temps réel sur un flux de données continu. Nous avons aussi abordé la problématique combinant ces deux aspects, pour pouvoir élaborer des solutions alliant : la sécurité d'un opérateur par rapport à son environnement, la cobotique qui est au coeur de l'industrie 4.0 et la pénibilité au travail.

Au sein du premier chapitre, nous avons vu un état de l'art exhaustif de toutes les technologies qui répondent à la question de la localisation en intérieur. Dans le cadre d'une localisation en intérieur en milieu très perturbé, notre choix s'est arrêté sur l'UWB. De nos jours, ce dispositif reste un choix abordable pour un industriel. L'état de l'art sur les algorithmes de Deep Learning nous montre que le choix le plus pertinent pour la reconnaissance d'actions avec des centrales inertielles est d'utiliser des données squelettes et de surcroît utiliser des algorithmes de réseau à graphe convolutionnel. Nous avons vu une approche à fenêtre glissante qui permet à ce type d'algorithme d'obtenir de meilleures performances en terme de précision pour des données en flux continu temps réel.

Dans un second chapitre, nous avons décrit le comportement d'un système Ultra WideBand dans des cas statiques et dynamiques. Nous avons réalisé une étude qualitative en nous appuyant sur une vérité terrain obtenue par un système de capture de mouvement. Nous avons réalisé une évaluation de la précision et de l'exactitude du système UWB qui est satisfaisant pour l'application souhaitée sur les axes X-Y mais ne répondant pas aux attentes d"une localisation en milieu industriel le long de l'axe Z. Nous avons également montré que si nous modifions le positionnement vertical des ancres, nous perdons en précision et en exactitude dans la localisation statique et dynamique. Nous avons également confirmé que la précision et l'exactitude sont meilleures en ajoutant des ancres

lors de la localisation dynamique. Les systèmes UWB peuvent être un choix pertinent pour la localisation, même en dynamique, et augmenter en robustesse si nous ajoutons plus d'ancres, sans trop en ajouter pour ne pas créer de multitrajet qui réduira la précision. La localisation selon l'axe vertical doit être améliorée, surtout en terme de précision. Une des pistes de résolution de cette problématique passe par la fusion de capteurs.

Dans un troisième chapitre, nous avons proposé un nouvel ensemble de données de type Non Ligne De Vue (NLDV), pour la localisation en intérieur avec six scénarii dynamiques dans un site industriel pendant une phase d'assemblage. Nous avons également suggéré des moyens d'améliorer notre jeu de données afin de le comparer avec d'autres domaines de recherche. Nous avons montré que la géométrie des ancres et la vitesse de l'utilisateur ont une influence sur le calcul de la position. Nous avons introduit une nouvelle méthode de filtrage de l'estimation de la position de l'UWB sans fusionner les données avec d'autres modalités. Cette étude vise à mieux comprendre les erreurs d'estimation de la position en se basant sur un support UWB dans des conditions réelles d'utilisation. Les deux modalités peuvent être mises en oeuvre pour la localisation en intérieur. Le système de capture de mouvement présente une meilleure précision que l'UWB mais nécessite une infrastructure plus coûteuse et plus complexe. Pour les cas d'utilisation nécessitant une précision moindre, ou une installation plus rapide et moins coûteuse,l'UWB est un choix pertinent pour accroître la sécurité dans les environnements industriels à faible coût.

Au sein du quatrième chapitre, il a été montré que l'approche à fenêtre glissante couplée aux réseaux spatio-temporels de graphes convolutionnels permet de tirer parti du maximum de performances de ce réseau. Celui-ci utilise les informations temporelles du squelette et peut caractériser le bruit autour de l'action pour déterminer la bonne action dans la fenêtre glissante. Nous avons montré que la fenêtre glissante est une bonne approche pour la reconnaissance d'actions en ligne en temps réel avec des flux de données continus. De plus, elle ne nécessite pas de processus lourd en terme de coût de calcul comme deux algorithmes de détections : un pour la segmentation du flux de données, un autre pour la reconnaissance d'action. Notre méthode n'emploie qu'un seul réseau de neurones. Il peut être intégré dans une petite unité de contrôle électronique (UCE) pour permettre une inférence de l'action en cours en temps réel.

L'une des limites est la taille de la fenêtre glissante. Elle est pertinente lorsque nous connaissons la durée moyenne d'une action. Nous avons validé notre méthode avec deux jeux de données connus de l'état de l'art avec une action commune de mouvement en temps réel, et nous avons montré une meilleure performance par rapport à cet état de l'art.

Nos travaux ont porté sur une fenêtre glissante variable qui permet de connaître plusieurs actions de durée différente. Le principal défi est la quantité de données, l'ensemble de données InHard [20] correspond à l'objectif de reconnaissance d'actions dans les sites industriels. Mais il faut augmenter le nombre de données pour atteindre le même nombre de grandeur pour chaque action (une cinquantaine dans notre cas) afin de généraliser. Notre méthode peut gagner en précision en utilisant des ST-GCN améliorés, comme le montre notre état de l'art sur les ST-GCN.

Dans le cinquième chapitre, nous avons utilisé un dispositif de réalité virtuelle pour exécuter un nouveau marquage des données afin d'améliorer l'équilibre des données industrielles. Nous avons utilisé un jeu de données industriel en ligne pour entraîner et valider notre algorithme, qui est à la fois pratique et expérimental. Sur les jeux de données de l'état de l'art, notre algorithme a généré les meilleurs résultats. Nous avons retenu notre algorithme afin de le tester sur ce jeu de données industrielles. Nous avons découvert des signes de dérive des capteurs MEMS (IMUs). C'est pourquoi nous avons recalé les actions industrielles avec la localisation UWB. Nous avons déterminé expérimentalement que la précision de l'UWB combinée à notre système de filtrage permet une localisation précise sans compromettre la détection des actions. Cependant, puisque notre algorithme a une robustesse de 74,76% (score F1) avec la combinaison de l'UWB et d'un optimiseur (Ranger), il y a des pistes d'amélioration, notamment en caractérisant mieux les gestes parasites grâces à une meilleure définition des graphes au niveau de la matrice d'entrée.

La technologie UWB, bien qu'émergente bascule dans le domaine de la grande consommation. On peut le constater avec les grandes marques de smartphones (Apple et Samsung) qui intègre l'UWB directement dans le téléphone. Les AirTags sont un tout nouveau gadget qui permet de retrouver des objets perdus en intérieur.

Dans cette thèse nous avons atteint les limites de la technologie UWB dans le domaine industriel. Il reste de nombreux axes de recherche, l'UWB devrait être une technologie qui permet de localiser précisement un objet ou une personne dans un milieu fortement perturbé. Mais elle devra s'appuyer sur d'autres technologies pour former un réseau ouvert pour que chaque utilisateur puisse se localiser dans n'importe quel lieu intérieur, comme l'utilisation du bluetooth par exemple.

Ce qui a été présenté dans cette thèse peut être utilisé pour localiser une personne et répondre à un besoin de sécurité en industrie. Le couplage UWB et détections d'actions peut servir pour la collaboration homme-machine en l'état. Il peut cependant être amélioré pour en faire un outil d'aide à la détection pour la pénibilité au travail comme localiser les zones de bruits par exemple et compter le nombre de fois qu'une personne se baisse ou effectue une action pénible.

L'utilisation d'algorithmes SNN (Spiking Neural Network) pour la détection d'actions serait avantageuse car elle permettrait de déployer notre algorithme en tirant parti des performances des réseaux de portes programmables (Field-Programmable Gate Array), tout en minimisant la consommation énergétique en comparaison actuelle sur carte graphique (GPU). En raison de sa forme d'impulsion naturelle, l'UWB pourrait être utilisée directement comme entrée de ces algorithmes sans aucun prétraitement. Cet axe est essentiel pour un système embarqué, et sera un complément très utile à la littérature scientifique.

Bibliographie

[1] Decawave. URL `https://www.decawave.com/`.

[2] N. Ahmad, R. A. R. Ghazilla, N. M. Khairi, and V. Kasi. Reviews on various inertial measurement unit (imu) sensor applications. *International Journal of Signal Processing Systems*, 1(2) :256–262, 2013.

[3] A. Al-Hamad, A. Ali, M. Elhoushi, and J. Georgy. Indoor Navigation using Consumer Portable Devices in Cart/Stroller. In *International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS+ 2017)*, pages 813–825, 2017.

[4] A. Alarifi, A. Al-Salman, M. Alsaleh, A. Alnafessah, S. Al-Hadhrami, M. A. Al-Ammar, and H. S. Al-Khalifa. Ultra wideband indoor positioning technologies : Analysis and recent advances. *Sensors*, 16(5) :707, 2016.

[5] T. Arai, T. Yoshizawa, T. Aoki, K. Zempo, and Y. Okada. Evaluation of Indoor Positioning System based on Attachable Infrared Beacons in Metal Shelf Environment. In *IEEE International Conference on Consumer Electronics (ICCE)*, pages 1–4, 2019.

[6] A. Aryan. Evaluation of ultra-wideband sensing technology for position location in indoor construction environments. Master's thesis, University of Waterloo, 2011.

[7] J. D. Bard and F. M. Ham. Time difference of arrival dilution of precision and applications. *IEEE transactions on Signal Processing*, 47(2) :521–523, 1999.

[8] V. Barral, P. Suárez-Casal, C. J. Escudero, and J. A. García-Naya. Multi-sensor accurate forklift location and tracking simulation in industrial indoor environments. *Electronics*, 8(10) :1152, 2019.

[9] P. J. Besl and N. D. McKay. Method for registration of 3-D shapes. In *Sensor Fusion IV : Control Paradigms and Data Structures*, volume 1611, pages 586–606. International Society for Optics and Photonics, 1992.

[10] R. Bharadwaj, S. Swaisaenyakorn, J. C. Batchelor, S. K. Koul, and A. Alomainy. Base-station random placement effect on the accuracy of ultrawideband body-centric localization applications. *IEEE Antennas and Wireless Propagation Letters*, 17(7) :1319–1323, 2018.

[11] J. Blankenbach and A. Norrdine. Position estimation using artificial generated magnetic fields. In *IEEE International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pages 1–5, 2010.

[12] B. Burki, S. Guillaume, P. Sorber, and H.-P. Oesch. DAEDALUS : A versatile usable digital clip-on measuring system for Total Stations. In *IEEE International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pages 1–10, 2010.

[13] C. Cadena, L. Carlone, H. Carrillo, Y. Latif, D. Scaramuzza, J. Neira, I. Reid, and J. J. Leonard. Past, present, and future of simultaneous localization and mapping : Toward the robust-perception age. *IEEE Transactions on Robotics*, 32(6) :1309–1332, 2016.

[14] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh. Realtime multi-person 2d pose estimation using part affinity fields. In *CVPR*, 2017.

[15] Z. Cao, G. Hidalgo Martinez, T. Simon, S. Wei, and Y. A. Sheikh. Openpose : Realtime multi-person 2d pose estimation using part affinity fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019.

[16] S. Celebi, A. S. Aydin, T. T. Temiz, and T. Arici. Gesture recognition using skeleton data with weighted dynamic time warping. In *VISAPP (1)*, pages 620–625, 2013.

[17] C.-C. Chiu, A. Narayanan, W. Han, R. Prabhavalkar, Y. Zhang, N. Jaitly, R. Pang, T. N. Sainath, P. Nguyen, L. Cao, et al. Rnn-t models fail to generalize to out-of-domain audio : Causes and solutions. In *2021 IEEE Spoken Language Technology Workshop (SLT)*, pages 873–880. IEEE, 2021.

[18] C. Combettes and V. Renaudin. Delay kalman filter to estimate the attitude of a mobile object with indoor magnetic field gradients. *Micromachines*, 7(5) :79, 2016.

[19] Y. Cui, Y. Zhang, Y. Huang, Z. Wang, and H. Fu. Novel WiFi/MEMS Integrated Indoor Navigation System Based on Two-Stage EKF. *Micromachines*, 10(3)(3) :198, 2019.

[20] M. Dallel, V. Havard, D. Baudry, and X. Savatier. Inhard - an industrial human action recogniton dataset in the context of industrial collaborative robotics. In *IEEE International Conference on Human-Machine Systems ICHMS*, 2020. URL `https://recherche.cesi.fr/inhard-industrial-human-action-recognition-dataset/`.

[21] M. Datar, A. Gionis, P. Indyk, and R. Motwani. Maintaining stream statistics over sliding windows. *SIAM journal on computing*, 31(6) :1794–1813, 2002.

[22] A. Dehghani, O. Sarbishei, T. Glatard, and E. Shihab. A quantitative comparison of overlapping and non-overlapping sliding windows for human activity recognition using inertial sensors. *Sensors*, 19(22) :5026, 2019.

[23] M. Delamare, R. Boutteau, X. Savatier, and N. Iriart. Evaluation of a UWB localization system in static and dynamic. In F. Potortì, V. Renaudin, K. O'Keefe, and F. Palumbo, editors, *Work-in-Progress Papers (IPIN-WiP 2019) (IPIN 2019), Pisa, Italy, September 30th - October 3rd, 2019*, volume 2498 of *CEUR Workshop Proceedings*, pages 80–86. CEUR-WS.org, 2019. URL `http://ceur-ws.org/Vol-2498/short11.pdf`.

[24] M. Delamare, R. Boutteau, X. Savatier, and N. Iriart. Evaluation of an uwb localization system in static/dynamic. In *International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, 2019.

[25] M. Delamare, R. Boutteau, X. Savatier, and N. Iriart. Static and dynamic evaluation of an uwb localization system for industrial applications. *Sci*, 2(1) :7, 2020.

[26] M. Delamare, F. Duval, and R. Boutteau. A new dataset of people flow in an industrial site with uwb and motion capture systems. *Sensors*, 20(16) :4511, 2020.

[27] M. Delamare., C. Laville., A. Cabani., and H. Chafouk. Graph convolutional networks skeleton-based action recognition for continuous data stream : A sliding window approach. In *Proceedings of the 16th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications - Volume 5 VISAPP : VISAPP,*, pages 427–435. INSTICC, SciTePress, 2021. ISBN 978-989-758-488-6. doi : $10.5220/0010234904270435$.

[28] J. Delgado and N. Ishii. Memory-based weighted majority prediction. In *SIGIR Workshop Recomm. Syst. Citeseer*, page 85. Citeseer, 1999.

[29] L. Deng. A tutorial survey of architectures, algorithms, and applications for deep learning. *APSIPA Transactions on Signal and Information Processing*, 3, 2014.

[30] L. Deng, Y. Wu, X. Hu, L. Liang, Y. Ding, G. Li, G. Zhao, P. Li, and Y. Xie. Rethinking the performance comparison between snns and anns. *Neural Networks*, 121 :294–307, 2020.

[31] N. Diliberti, C. Peng, C. Kaufman, Y. Dong, and J. T. Hansberger. Real-time gesture recognition using 3d sensory data and a light convolutional neural network. In *Proceedings of the 27th ACM International Conference on Multimedia*, pages 401–410, 2019.

[32] S. Djosic, I. Stojanovic, M. Jovanovic, T. Nikolic, and G. L. Djordjevic. Fingerprinting-assisted uwb-based localization technique for complex indoor environments. *Expert Systems with Applications*, 167 :114188, 2021.

[33] B. H. Dobkin. Wearable motion sensors to continuously measure real-world physical activities. *Current opinion in neurology*, 26(6) :602, 2013.

[34] I. Dotlic, A. Connell, H. Ma, J. Clancy, and M. McLaughlin. Angle of arrival estimation using decawave DW1000 integrated circuits. In *Positioning, Navigation and Communications (WPNC), 2017 14th Workshop on*, pages 1–6. IEEE, 2017.

[35] D. Dragomirescu, M. Kraemer, M. Jatlaoui, P. Pons, H. Aubert, A. Thain, and R. Plana. 60ghz Wireless Nano-Sensors Network for Structure Health Monitoring as Enabler for Safer, Greener Aircrafts. *Proceedings of SPIE - The International Society for Optical Engineering*, 7821, 2010.

[36] Y. Du, Y. Fu, and L. Wang. Skeleton based action recognition with convolutional neural network. In *2015 3rd IAPR Asian Conference on Pattern Recognition (ACPR)*, pages 579–583. IEEE, 2015.

[37] D. Eggert, A. Lorusso, and R. Fisher. Estimating 3-D rigid body transformations : a comparison of four major algorithms. *Machine Vision and Applications*, 9(5-6) :272–290, Mar. 1997. ISSN 0932-8092, 1432-1769. doi : $10.1007/s001380050048$. URL `http://link.springer.com/10.1007/s001380050048`.

[38] S. Eickeler, A. Kosmala, and G. Rigoll. Hidden markov model based continuous online gesture recognition. In *Proceedings. Fourteenth International Conference on Pattern Recognition (Cat. No. 98EX170)*, volume 2, pages 1206–1208. IEEE, 1998.

[39] R. El-Allami, A. Marchisio, M. Shafique, and I. Alouani. Securing deep spiking neural networks against adversarial attacks through inherent structural parameters. *arXiv preprint arXiv :2012.05321*, 2020.

[40] J. Favre, B. Jolles, O. Siegrist, and K. Aminian. Quaternion-based fusion of gyroscopes and accelerometers to improve 3d angle measurement. *Electronics Letters*, 42(11) :612–614, 2006.

[41] A. G. Ferreira, D. Fernandes, A. P. Catarino, A. M. Rocha, and J. L. Monteiro. A loose-coupled fusion of inertial and uwb assisted by a decision-making algorithm for localization of emergency responders. *Electronics*, 8 (12) :1463, 2019.

[42] D. Fleer and R. Möller. Comparing holistic and feature-based visual methods for estimating the relative pose of mobile robots. *Robotics and Autonomous Systems*, 89 :51–74, 2017.

[43] D. T.-P. Fong and Y.-Y. Chan. The use of wearable inertial motion sensors in human lower limb biomechanics studies : a systematic review. *Sensors*, 10(12) :11556–11565, 2010.

[44] A. Fort, C. Desset, J. Ryckaert, P. De Doncker, L. Van Biesen, and P. Wambacq. Characterization of the ultra wideband body area propagation channel. In *2005 IEEE International Conference on Ultra-Wideband*, pages 6 pp.–, 2005.

[45] S. Frattasi and F. Della Rosa. *Mobile positioning and tracking : from conventional to cooperative techniques*. John Wiley & Sons, 2017.

[46] K. Fujii, Y. Sakamoto, W. Wang, H. Arie, A. Schmitz, and S. Sugano. Hyperbolic Positioning with Antenna Arrays and Multi-Channel Pseudolite for Indoor Localization. *Sensors*, 15(10)(10) :25157–25175, 2015.

[47] J. S. Furtado, H. H. Liu, G. Lai, H. Lacheray, and J. Desouza-Coelho. Comparative analysis of optitrack motion capture systems. In *Advances in Motion Sensing and Control for Robotic Applications*, pages 15–31. Springer, 2019.

[48] G. Gallego and D. Scaramuzza. Accurate angular velocity estimation with an event camera. *IEEE Robotics and Automation Letters*, 2(2) :632–639, 2017.

[49] S. Gansemer, U. Grossmann, and S. Hakobyan. RSSI-based Euclidean Distance algorithm for indoor positioning adapted for the use in dynamically changing WLAN environments and multi-level buildings. In *IEEE International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pages 1–6, 2010.

[50] V. Gharat, E. Colin, G. Baudoin, and D. Richard. Indoor performance analysis of LF-RFID based positioning system : Comparison with UHF-RFID and UWB. In *Indoor Positioning and Indoor Navigation (IPIN), 2017 International Conference on*, pages 1–8. IEEE, 2017.

[51] I. Goodfellow, Y. Bengio, and A. Courville. 6.5 back-propagation and other differentiation algorithms. *Deep Learning*, pages 200–220, 2016.

[52] I. Goodfellow, Y. Bengio, A. Courville, and Y. Bengio. *Deep learning*, volume 1. MIT press Cambridge, 2016.

[53] D. Gorecky, M. Schmitt, M. Loskyll, and D. Zühlke. Human-machine-interaction in the industry 4.0 era. In *2014 12th IEEE international conference on industrial informatics (INDIN)*, pages 289–294. IEEE, 2014.

[54] B. Großwindhager, M. Rath, J. Kulmer, M. S. Bakr, C. A. Boano, K. Witrisal, and K. Römer. Dataset : single-anchor indoor localization with decawave dw1000 and directional antennas. In *Proceedings of the First Workshop on Data Acquisition To Analysis*, pages 21–22, 2018.

[55] I. . W. Group et al. Ieee standard for local and metropolitan area networks—part 15.4 : Low-rate wireless personal area networks (lr-wpans). *IEEE Std*, 802 :4–2011, 2011.

[56] M. Haddara and A. Elragal. The readiness of erp systems for the factory of the future. *Procedia Computer Science*, 64 :721–728, 2015.

[57] T. Hamel and R. Mahony. Attitude estimation on SO [3] based on direct inertial measurements. In *Robotics and Automation, 2006. ICRA 2006. Proceedings 2006 IEEE International Conference on*, pages 2170–2175. IEEE, 2006.

[58] J. Ido, Y. Shimizu, Y. Matsumoto, and T. Ogasawara. Indoor Navigation for a Humanoid Robot Using a View Sequence. *The International Journal of Robotics Research (IJRR)*, 28(2)(2) :315–325, 2009.

[59] A. R. Jiménez and F. Seco. Improving the accuracy of decawave's uwb mdek1001 location system by gaining access to multiple ranges. *Sensors*, 21(5) :1787, 2021.

[60] A. R. Jiménez and F. Seco. Comparing Decawave and Bespoon UWB location systems : Indoor/outdoor performance analysis. In *Indoor Positioning and Indoor Navigation (IPIN), 2016 International Conference on*, pages 1–8. IEEE, 2016.

[61] F. A. Kaya and M. Saritas. A computer simulation of dilution of precision in the global positioning system using matlab. In *Proceedings of the 4th International Conference on Electrical and Electronic Engineering, Bursa, Turkey*, volume 711, 2005.

[62] K. Khoshelham and S. O. Elberink. Accuracy and Resolution of Kinect Depth Data for Indoor Mapping Applications. *Sensors*, 12(2)(2) :1437–1454, 2012.

[63] M. Kiers, W. Bischof, E. Krajnc, and M. Dornhofer. Evaluation and improvements of an rfid based indoor navigation system for visually impaired and blind people. In *2011 International Conference on Indoor Positioning and Indoor Navigation ; Paper, Guimarães, Portugal (September 2011)*, volume 16, 2011.

[64] D. P. Kingma and J. Ba. Adam : A method for stochastic optimization. *arXiv preprint arXiv :1412.6980*, 2014.

[65] T. K. Kohoutek, R. Mautz, and A. Donaubauer. Real-time indoor positioning using range imaging sensors. In *Real-Time Image and Video Processing*, 2010.

[66] H. H. Ku. Precision measurement and calibration. volume 1. statistical concepts and procedures. Technical report, NATIONAL BUREAU OF STANDARDS GAITHERSBURG MD, 1969.

[67] R. S. Kulikov. Integrated UWB/IMU system for high rate indoor navigation with cm-level accuracy. In *Electronic and Networking Technologies (MWENT), 2018 Moscow Workshop on*, pages 1–4. IEEE, 2018.

[68] J. O. Laguna, A. G. Olaya, and D. Borrajo. A dynamic sliding window approach for activity recognition. In *International Conference on User Modeling, Adaptation, and Personalization*, pages 219–230. Springer, 2011.

[69] O. D. Lara and M. A. Labrador. A survey on human activity recognition using wearable sensors. *IEEE communications surveys & tutorials*, 15(3) :1192–1209, 2012.

[70] H. Lasi, P. Fettke, H.-G. Kemper, T. Feld, and M. Hoffmann. Industry 4.0. *Business & information systems engineering*, 6(4) :239–242, 2014.

[71] P. Lei and S. Todorovic. Temporal deformable residual networks for action segmentation in videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6742–6751, 2018.

[72] J. Li, Y. Bi, K. Li, K. Wang, F. Lin, and B. M. Chen. Accurate 3d localization for mav swarms by uwb and imu fusion. In *2018 IEEE 14th International Conference on Control and Automation (ICCA)*, pages 100–105. IEEE, 2018.

[73] J. Li, Y. Bi, K. Li, K. Wang, F. Lin, and B. M. Chen. Accurate 3d Localization for MAV Swarms by UWB and IMU Fusion. In *IEEE International Conference on Control and Automation (ICCA)*, pages 100–105, 2018.

[74] J. Li, C. Wang, X. Kang, and Q. Zhao. Camera localization for augmented reality and indoor positioning : a vision-based 3d feature database approach. *International Journal of Digital Earth*, pages 1–15, 2019.

[75] M. Li, S. Chen, X. Chen, Y. Zhang, Y. Wang, and Q. Tian. Actional-structural graph convolutional networks for skeleton-based action recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3595–3603, 2019.

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

[77] X. Li, Y. Wang, and K. Khoshelham. Comparative analysis of robust extended kalman filter and incremental smoothing for uwb/pdr fusion positioning in nlos environments. *Acta Geodaetica et Geophysica*, 54(2) :157–179, 2019.

[78] Y. Li, C. Lan, J. Xing, W. Zeng, C. Yuan, and J. Liu. Online human action detection using joint classification-regression recurrent neural networks. In *European Conference on Computer Vision*, pages 203–220. Springer, 2016.

[79] C. Lian Sang, B. Steinhagen, J. D. Homburg, M. Adams, M. Hesse, and U. Rückert. Identification of nlos and multi-path conditions in uwb localization using machine learning methods. *Applied Sciences*, 10(11) :3980, 2020.

[80] X. Liao, R. Chen, M. Li, B. Guo, X. Niu, and W. Zhang. Design of a Smartphone Indoor Positioning Dynamic Ground Truth Reference System Using Robust Visual Encoded Targets. *Sensors*, 19(5) :1261, 2019. URL `https://www.mdpi.com/1424-8220/19/5/1261`.

[81] G. Ligorio, E. Bergamini, I. Pasciuto, G. Vannozzi, A. Cappozzo, and A. M. Sabatini. Assessing the performance of sensor fusion methods : Application to magnetic-inertial-based human body tracking. *Sensors*, 16 (2) :153, 2016.

[82] M. Lin, Q. Chen, and S. Yan. Network in network. *arXiv preprint arXiv :1312.4400*, 2013.

[83] J. Liu, A. Shahroudy, D. Xu, A. C. Kot, and G. Wang. Skeleton-based action recognition using spatio-temporal lstm network with trust gates. *IEEE transactions on pattern analysis and machine intelligence*, 40(12) :3007–3021, 2017.

[84] J. Liu, G. Wang, P. Hu, L.-Y. Duan, and A. C. Kot. Global context-aware attention lstm networks for 3d action recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1647–1656, 2017.

[85] J. Liu, A. Shahroudy, G. Wang, L.-Y. Duan, and A. C. Kot. Skeleton-based online action prediction using scale selection network. *IEEE transactions on pattern analysis and machine intelligence*, 42(6) :1453–1467, 2019.

[86] L. Liu, H. Jiang, P. He, W. Chen, X. Liu, J. Gao, and J. Han. On the variance of the adaptive learning rate and beyond. *arXiv preprint arXiv :1908.03265*, 2019.

[87] M. Loukadakis, J. Cano, and M. O'Boyle. Accelerating deep neural networks on low power heterogeneous architectures. 01 2018.

[88] G. Luzhnica, J. Simon, E. Lex, and V. Pammer. A sliding window approach to natural hand gesture recognition using a custom data glove. In *2016 IEEE Symposium on 3D User Interfaces (3DUI)*, pages 81–90. IEEE, 2016.

[89] C. Ma, W. Li, J. Cao, J. Du, Q. Li, and R. Gravina. Adaptive sliding window based activity recognition for assisted livings. *Information Fusion*, 53 :55–65, 2020.

[90] S. Madgwick. An Efficient Orientation Filter for Inertial and Inertial. *Magnetic Sensor Arrays*, 2010.

[91] G. Marin, F. Dominio, and P. Zanuttigh. Hand gesture recognition with leap motion and kinect devices. In *2014 IEEE International conference on image processing (ICIP)*, pages 1565–1569. IEEE, 2014.

[92] R. Mautz. *Indoor positioning technologies*. PhD Thesis, ETH Zurich, Department of Civil, Environmental and Geomatic Engineering, Institute of Geodesy and Photogrammetry, 2012.

[93] M. Meredith, S. Maddock, et al. Motion capture file formats explained. *Department of Computer Science, University of Sheffield*, 211 :241–244, 2001.

[94] P. Merriaux, Y. Dupuis, R. Boutteau, P. Vasseur, and X. Savatier. A Study of Vicon System Positioning Performance. *Sensors*, 17(7) :1591, 2017.

[95] T. Michel, P. Genevès, H. Fourati, and N. Layaïda. On attitude estimation with smartphones. In *Pervasive Computing and Communications (PerCom), 2017 IEEE International Conference on*, pages 267–275. IEEE, 2017.

[96] K. Minne, N. Macoir, J. Rossey, Q. Van den Brande, S. Lemey, J. Hoebeke, and E. De Poorter. Experimental evaluation of uwb indoor positioning for indoor track cycling. *Sensors*, 19(9) :2041, 2019.

[97] L. Miranda, T. Vieira, D. Martínez, T. Lewiner, A. W. Vieira, and M. F. Campos. Online gesture recognition from pose kernel learning and decision forests. *Pattern Recognition Letters*, 39 :65–73, 2014.

[98] D. Misra. Mish : A self regularized non-monotonic activation function. *arXiv preprint arXiv :1908.08681*, pages 1–14, 2020.

[99] S. Mitra and T. Acharya. Gesture recognition : A survey. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 37(3) :311–324, 2007.

[100] E. Mok, F. Lau, L. Xia, G. Retscher, and H. Tian. Influential factors for decimetre level positioning using ultra wide band technology. *Survey Review*, 44(324) :37–44, 2012.

[101] P. Molchanov, X. Yang, S. Gupta, K. Kim, S. Tyree, and J. Kautz. Online detection and classification of dynamic hand gestures with recurrent 3d convolutional neural network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4207–4215, 2016.

[102] G. Mongillo and S. Deneve. Online learning with hidden markov models. *Neural computation*, 20(7) :1706–1716, 2008.

[103] M. Mérida-Floriano, F. Caballero, D. García-Morales, F. Casares, and L. Merino. Bioinspired vision-only UAV attitude rate estimation using machine learning. In *Unmanned Aircraft Systems (ICUAS), 2017 International Conference on*, pages 1476–1482. IEEE, 2017.

[104] C. R. Naguri and R. C. Bunescu. Recognition of dynamic hand gestures from 3d motion data using lstm and cnn architectures. In *2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pages 1130–1133. IEEE, 2017.

[105] M. Narasimhappa, A. D. Mahindrakar, V. C. Guizilini, M. H. Terra, and S. L. Sabat. Mems-based imu drift minimization : Sage husa adaptive robust kalman filtering. *IEEE Sensors Journal*, 20(1) :250–260, 2019.

[106] P. Nguyen, T. Liu, G. Prasad, and B. Han. Weakly supervised action localization by sparse temporal pooling network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6752–6761, 2018.

[107] P. Ni, S. Lv, X. Zhu, Q. Cao, and W. Zhang. A light-weight on-line action detection with hand trajectories for industrial surveillance. *Digital Communications and Networks*, 2020.

[108] Q. Niu, M. Li, S. He, C. Gao, S. H. Gary Chan, and X. Luo. Resource-efficient and Automated Image-based Indoor Localization. *ACM Transactions on Sensor Networks (TOSN)*, 15(2)(2) :19, 2019.

[109] K. Papadopoulos, E. Ghorbel, D. Aouada, and B. Ottersten. Vertex feature encoding and hierarchical temporal modeling in a spatial-temporal graph convolutional network for action recognition. *arXiv preprint arXiv :1912.09745*, 2019.

[110] L. Patiño-Studencka, U. Batzer, and J. Thielecke. Phase smoothing in a virtually synchronized pseudolite system using stochastic clock modelling. In *Ubiquitous Positioning Indoor Navigation and Location Based Service*, pages 1–5, 2010.

[111] C. Pehle and J. E. Pedersen. Norse - A deep learning library for spiking neural networks, Jan. 2021. URL `https://doi.org/10.5281/zenodo.4422025`. Documentation : https ://norse.ai/docs/.

[112] M. M. Pietrzyk and T. von der Grün. Ultra-wideband technology-based ranging platform with real-time signal processing. In *Signal Processing and Communication Systems (ICSPCS), 2010 4th International Conference on*, pages 1–5. IEEE, 2010.

[113] R. Polfreman. Hand posture recognition : Ir, semg and imu. 2018.

[114] J. P. Queralta, C. M. Almansa, F. Schiano, D. Floreano, and T. Westerlund. Uwb-based system for uav localization in gnss-denied environments : Characterization and dataset. *arXiv preprint arXiv :2003.04380*, 2020.

[115] U. Raza, A. Khan, R. Kou, T. Farnham, T. Premalal, A. Stanoev, and W. Thompson. Dataset : Indoor localization with narrow-band, ultra-wideband, and motion capture systems. In *Proceedings of the 2nd Workshop on Data Acquisition To Analysis*, pages 34–36, 2019.

[116] V. Renaudin, B. Merminod, and M. Kasser. Optimal data fusion for pedestrian navigation based on UWB and MEMS. In *Position, Location and Navigation Symposium, 2008 IEEE/ION*, pages 341–349. IEEE, 2008.

[117] D. Roetenberg, H. Luinge, and P. Slycke. Xsens mvn : Full 6dof human motion tracking using miniature inertial sensors. *Xsens Motion Technologies BV, Tech. Rep*, 1, 2009.

[118] A. Rojko. Industry 4.0 concept : background and overview. *International Journal of Interactive Mobile Technologies (iJIM)*, 11(5) :77–90, 2017.

[119] A. R. J. Ruiz and F. S. Granja. Comparing Ubisense, BeSpoon, and DecaWave UWB location systems : indoor performance analysis. *IEEE Transactions on instrumentation and Measurement*, 66(8) :2106–2117, 2017.

[120] T. Sato, S. Nakamura, K. Terabayashi, M. Sugimoto, and H. Hashizume. Design and implementation of a robust and real-time ultrasonic motion-capture system. In *IEEE International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pages 1–6, 2011.

[121] A. Savitzky and M. J. Golay. Smoothing and differentiation of data by simplified least squares procedures. *Analytical chemistry*, 36(8) :1627–1639, 1964.

[122] A. G. Schwing and R. Urtasun. Fully connected deep structured networks. *arXiv preprint arXiv :1503.02351*, 2015.

[123] M. Segura, H. Hashemi, C. Sisterna, and V. Mut. Experimental demonstration of self-localized ultra wideband indoor mobile robot navigation system. In *Indoor Positioning and Indoor Navigation (IPIN), 2010 International Conference on*, pages 1–9. IEEE, 2010.

[124] L. Shi, Y. Zhang, J. Cheng, and H. Lu. Two-stream adaptive graph convolutional networks for skeleton-based action recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 12026–12035, 2019.

[125] W. Shi, J. Du, X. Cao, Y. Yu, Y. Cao, S. Yan, and C. Ni. IKULDAS : An Improved kNN-Based UHF RFID Indoor Localization Algorithm for Directional Radiation Scenario. *Sensors*, 19(4)(4) :968, 2019.

[126] A. Shrestha and A. Mahmood. Review of deep learning algorithms and architectures. *IEEE Access*, 7 : 53040–53065, 2019.

[127] C. Si, W. Chen, W. Wang, L. Wang, and T. Tan. An attention enhanced graph convolutional lstm network for skeleton-based action recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1227–1236, 2019.

[128] B. Silva, Z. Pang, J. Åkerberg, J. Neander, and G. Hancke. Experimental study of uwb-based high precision localization for industrial applications. In *2014 IEEE International Conference on Ultra-WideBand (ICUWB)*, pages 280–285, 2014.

[129] T. Simon, H. Joo, I. Matthews, and Y. Sheikh. Hand keypoint detection in single images using multiview bootstrapping. In *CVPR*, 2017.

[130] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv :1409.1556*, 2014.

[131] I. Skog, P. Handel, J.-O. Nilsson, and J. Rantakokko. Zero-velocity detection—An algorithm evaluation. *IEEE Transactions on Biomedical Engineering*, 57(11) :2657–2666, 2010.

[132] J. Steinier, Y. Termonia, and J. Deltour. Smoothing and differentiation of data by simplified least square procedure. *Analytical chemistry*, 44(11) :1906–1909, 1972.

[133] M. Susi, V. Renaudin, and G. Lachapelle. Motion mode recognition and step detection algorithms for mobile phone users. *Sensors*, 13(2) :1539–1562, 2013.

[134] A. Syberfeldt, M. Ayani, M. Holm, L. Wang, and R. Lindgren-Brewster. Localizing operators in the smart factory : A review of existing techniques and systems. In *Flexible Automation (ISFA), International Symposium on*, pages 179–185. IEEE, 2016.

[135] C. Tang, W. Li, P. Wang, and L. Wang. Online human action recognition based on incremental learning of weighted covariance descriptors. *Information Sciences*, 467 :219–237, 2018.

[136] L. Tao, E. Elhamifar, S. Khudanpur, G. D. Hager, and R. Vidal. Sparse hidden markov models for surgical gesture classification and skill evaluation. In *International conference on information processing in computer-assisted interventions*, pages 167–177. Springer, 2012.

[137] A. Tavanaei, M. Ghodrati, S. R. Kheradpisheh, T. Masquelier, and A. Maida. Deep learning in spiking neural networks. *Neural networks : the official journal of the International Neural Network Society*, 111 :47–63, 2019.

[138] Q. Tian, I. Kevin, K. Wang, and Z. Salcic. A low-cost ins and uwb fusion pedestrian tracking system. *IEEE Sensors Journal*, 19(10) :3733–3740, 2019.

[139] S. Tilch and R. Mautz. Current investigations at the ETH Zurich in optical indoor positioning. In *IEEE Workshop on Positioning, Navigation and Communication*, pages 174–178, 2010.

[140] D. T$\rho\iota\acute{\alpha}\nu\tau\eta\varsigma$. Functionally weighted convolutional neural networks. 2017.

[141] D. Vandermeulen, C. Vercauteren, and M. Weyn. Indoor localization Using a Magnetic Flux Density Map of a Building. In *International Conference on Ambient Computing, Applications, Services and Technologies*, pages 42–49, 2013.

[142] D. Vasisht, S. Kumar, and D. Katabi. Decimeter-level localization with a single wifi access point. In *13th {USENIX} Symposium on Networked Systems Design and Implementation ({NSDI} 16)*, pages 165–178, 2016.

[143] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. Attention is all you need. *arXiv preprint arXiv :1706.03762*, 2017.

[144] J. Vreeken. Spiking neural networks, an introduction. 2003.

[145] X. Wang, M. Xia, H. Cai, Y. Gao, and C. Cattani. Hidden-markov-models-based dynamic hand gesture recognition. *Mathematical Problems in Engineering*, 2012, 2012.

[146] Y.-T. Wang, J. Li, R. Zheng, and D. Zhao. ARABIS : an Asynchronous Acoustic Indoor Positioning System for Mobile Devices. In *IEEE International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pages 1–8, 2017.

[147] S.-E. Wei, V. Ramakrishna, T. Kanade, and Y. Sheikh. Convolutional pose machines. In *CVPR*, 2016.

[148] Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, and S. Y. Philip. A comprehensive survey on graph neural networks. *IEEE Transactions on Neural Networks and Learning Systems*, 2020.

[149] J. Xiong and K. Jamieson. Arraytrack : A fine-grained indoor location system. In *Presented as part of the 10th {USENIX} Symposium on Networked Systems Design and Implementation ({NSDI} 13)*, pages 71–84, 2013.

[150] D. Xu, X. Wu, Y.-L. Chen, and Y. Xu. Online dynamic gesture recognition for human robot interaction. *Journal of Intelligent & Robotic Systems*, 77(3-4) :583–596, 2015.

[151] S. Yan, Y. Xiong, and D. Lin. Spatial temporal graph convolutional networks for skeleton-based action recognition. In *Thirty-second AAAI conference on artificial intelligence*, 2018.

[152] R. Ye and H. Liu. Uwb tdoa localization system : Receiver configuration analysis. In *2010 International Symposium on Signals, Systems and Electronics*, volume 1, pages 1–4. IEEE, 2010.

[153] H. Yong, J. Huang, X. Hua, and L. Zhang. Gradient centralization : A new optimization technique for deep neural networks. In *European Conference on Computer Vision*, pages 635–652. Springer, 2020.

[154] J. Yu, Y. Yoon, and M. Jeon. Predictively encoded graph convolutional network for noise-robust skeleton-based action recognition. *arXiv preprint arXiv :2003.07514*, 2020.

[155] F. Zafari, A. Gkelias, and K. K. Leung. A survey of indoor localization systems and technologies. *IEEE Communications Surveys & Tutorials*, 2019.

[156] B. Zhang, L. Wang, Z. Wang, Y. Qiao, and H. Wang. Real-time action recognition with enhanced motion vector cnns. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2718–2726, 2016.

[157] M. R. Zhang, J. Lucas, G. Hinton, and J. Ba. Lookahead optimizer : k steps forward, 1 step back. *arXiv preprint arXiv :1907.08610*, 2019.

[158] X. Zhang, C. Xu, and D. Tao. Context aware graph convolution for skeleton-based action recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14333–14342, 2020.

[159] X. Zhao, X. Li, C. Pang, X. Zhu, and Q. Z. Sheng. Online human gesture recognition from motion data streams. In *Proceedings of the 21st ACM international conference on Multimedia*, pages 23–32, 2013.

[160] W. Zheng, P. Jing, and Q. Xu. Action recognition based on spatial temporal graph convolutional networks. In *Proceedings of the 3rd International Conference on Computer Science and Application Engineering*, pages 1–5, 2019.

[161] W. Zhu, C. Lan, J. Xing, W. Zeng, Y. Li, L. Shen, and X. Xie. Co-occurrence feature learning for skeleton based action recognition using regularized deep lstm networks. *arXiv preprint arXiv :1603.07772*, 2016.

[162] D. Zito and D. Morche. UWB Radios—The maturity age ? In *New Circuits and Systems Conference (NEWCAS), 2016 14th IEEE International*, pages 1–4. IEEE, 2016.

Titre : Localisation en intérieur à bande Ultra large (UWB) et reconnaissance d'actions industrielles

Mots clés : Localisation en intérieur,technologie à bande ultra large (UWB),estimation de distance,Réseaux convolutifs à graphes spatio-temporels,Reconnaissance d'actions industrielles.

Résumé : La robotisation, en particulier dans les usines, induit une interaction de plus en plus étroite entre l'homme et la machine, concept rassemblé dans le terme « cobotique ». Les objectifs de ces travaux de thèse est le développement d'un système de localisation et de reconnaissance d'actions d'un opérateur pour le pilotage de machines industrielles dans le contexte de l'industrie 4.0. Les travaux ont consisté à faire un état de l'art sur les méthodes et technologies de localisation en environnement intérieur. La technologie Ultra WideBand (UWB) a été retenue pour avoir une estimation de la position absolue d'une personne en milieu industriel. Le développement d'une méthode d'acquisition des données a été réalisé afin d'évaluer la précision d'un système UWB en statique et en dynamique au sein du laboratoire IRSEEM. Des tests complémentaires ont été également effectués sur une chaîne de production. Ces tests nous ont permis d'évaluer les performances du système UWB dans un environnement industriel complexe représentatif des applications visées avec une vérité terrain fiable. Suite à un état de l'art sur les différents algorithmes de classification d'actions, nous en avons conclu qu'il y a un fort intérêt à avoir des données représentatives des actions d'un opérateur au sein d'une entreprise. Ainsi, le couplage entre l'UWB et les centrales inertielles permettra d'avoir une localisation précise et de classifier l'action effectuée par l'opérateur. Nous avons approfondi nos recherches sur des algorithmes d'apprentissage profond basé sur le squelette appelé Spatio-Temporal Graph Convolutionnal Networks (ST-GCN). Nous avons établis une nouvelle méthode appelée SW-GCN avec une fenêtre glissante qui permet une détection temps réel sur des données en flux continue, qui nous ont permis une meilleure classification comparée à l'état de l'art, et ensuite avec des données représentatives (industrielles) et donc plus complexe. Cette dernière méthode a été combinée avec l'UWB pour recaler les IMUs et ne pas perdre l'information de localisation. Ces travaux de thèse ouvrent sur plusieurs applications, comme la réduction de la pénibilité au travail, une meilleure sécurité en entreprise, et une collaboration cobotique accrus, ce qui est en adéquation avec l'industrie 4.0 vers l'industrie 5.0 qui remet l'homme au cœur de l'industrie.

Title : Ultra Wide Band (UWB) indoor localization and industrial action recognition

Keywords : Indoor localization, Ultra Wide Band (UWB) technology, range estimation, Spatial-temporal Graph Convolutional Networks, industrial action recognition.

Abstract : Robotization, especially in factories, induces an increasingly close interaction between man and machine, a concept gathered in the term « cobotics ». The objectives of this thesis work is the development of a system of localization and recognition of operator's actions for the piloting of industrial machines in the context of the industry 4.0. The work consisted in making a state of the art on the methods and technologies of localization in indoor environment. The Ultra WideBand (UWB) technology was selected to have an estimation of the absolute position of a person in an industrial environment. The development of a data acquisition method was carried out in order to evaluate the accuracy of a UWB system in static and dynamic mode within the IRSEEM laboratory. Complementary tests were also carried out on a production line. These tests allowed us to evaluate the performance of the UWB system in a complex industrial environment representative of the targeted applications with a reliable ground truth. Following a state of the art on the different action classification algorithms, we concluded that there is a strong interest in having data representative of the actions of an operator within a company. Thus, the coupling between the UWB and the inertial units will allow to have a precise localization and to classify the action performed by the operator. We have deepened our research on deep learning algorithms based on the skeleton called Spatio-Temporal Graph Convolutional Networks (ST-GCN), we have established a new method called SW-GCN with a sliding window that allows a real-time detection on continuous flow data, which allowed us a better classification compared to the state of the art, and then with representative data (industrial) and therefore more complex. This last method has been combined with UWB to re-calibrate the IMUs and not to lose the location information. This thesis work opens on several applications, such as work drudgery, a better safety in companies, and an increased cobotic collaboration, that is in adequacy with the industry 4.0 towards the industry 5.0 which puts the man in the heart of the industry.

Université de Rouen Normandie
1 rue Thomas Becket 76821 Mont-Saint-Aignan, France