Titre: Title:	Reconnaissance de la gestuelle statique de la main par apprentissage profond pour des plateformes interactives de procédés de fabrication
Auteur: Author:	Corentin Hubert
Date:	2023
Type:	Mémoire ou thèse / Dissertation or Thesis
Référence: Citation:	Hubert, C. (2023). Reconnaissance de la gestuelle statique de la main par apprentissage profond pour des plateformes interactives de procédés de fabrication [Mémoire de maîtrise, Polytechnique Montréal]. PolyPublie. https://publications.polymtl.ca/56782/

POLYTECHNIQUE MONTRÉAL
affiliée à l'Université de Montréal

Reconnaissance de la gestuelle statique de la main par apprentissage profond pour des plateformes interactives de procédés de fabrication

CORENTIN HUBERT
Département de génie informatique et génie logiciel

Mémoire présenté en vue de l'obtention du diplôme de *Maîtrise ès sciences appliquées*
Génie informatique

Décembre 2023

POLYTECHNIQUE MONTRÉAL
affiliée à l'Université de Montréal

Ce mémoire intitulé :

Reconnaissance de la gestuelle statique de la main par apprentissage profond pour des plateformes interactives de procédés de fabrication

présenté par **Corentin HUBERT**
en vue de l'obtention du diplôme de *Maîtrise ès sciences appliquées*
a été dûment accepté par le jury d'examen constitué de :

Thomas HURTUT, président
Lama SÉOUD, membre et directrice de recherche
Amir HAJZARGARBASHI, membre externe

DÉDICACE

À tous mes amis du labos,
vous me manquerez...

REMERCIEMENTS

Je tiens à exprimer ma gratitude envers toutes les personnes qui ont contribué de près ou de loin à la réalisation de ce mémoire de maîtrise. Leur soutien et leur implication ont grandement enrichi cette expérience de recherche.

Je tiens tout particulièrement à remercier ma directrice de recherche, Lama Seoud, pour sa guidance, ses conseils éclairés et son soutien constant tout au long de ce projet.

Un grand merci à Marie Noël, dont l'aide précieuse dans la création de la base de données a été déterminante.

Je souhaite exprimer ma reconnaissance envers Yoann, Julien-Mathieu et Amir qui ont généreusement ouvert les portes de la cellule cobotique, rendant ainsi possible la collecte de données essentielle à cette recherche.

Un grand merci à tous les participants de la base de données qui ont consacré leur temps sans contrepartie. Leur contribution volontaire joue un rôle fondamentale dans la réussite du projet.

Mes amis du laboratoire méritent également une mention spéciale. Leurs corrections avisées et leur aide précieuse pour mettre en place la configuration de l'ordinateur du laboratoire ont grandement facilité le processus de travail.

Enfin merci aux membres du jury pour leur temps et leur lecture de ce mémoire.

RÉSUMÉ

Les procédés de fabrication tels que l'ébavurage ou le polissage sont des tâches de parachèvement qui peuvent être longues et fastidieuses à réaliser pour les ouvriers. De nombreuses industries utilisent des robots pour réaliser ces tâches mais cela nécessite le réglage de paramètres que seuls les ouvriers avec leur expertise peuvent gérer. Nous proposons une méthode de communication entre le robot et l'opérateur pour que le réglage de ces paramètres soit le plus simple et efficace possible.

Cette communication se fait avec les gestes de la main, ce qui permet également à l'opérateur de montrer directement les zones qui doivent être retouchées sur la pièce à parachever. Notre solution utilise un système multi-caméra RGB-D : 6 caméras sont réparties dans la cellule cobotique où le robot et l'ouvrier vont collaborer pour corriger les pièces.

Nous avons développé un pipeline composé d'un modèle de détection de la main dans une image et d'un modèle de reconnaissance par classification de la gestuelle de la main. Nous avons testé ces modèles séparément et en chaîne, sur une base de données publique et détenons une meilleure performance sur ces données que l'article original. Cependant, cette base de données a été créée dans un contexte idéal, sans occlusion et avec une faible variabilité de points de vues. C'est pourquoi nous avons créé une nouvelle base de données, plus large, avec plusieurs caméras qui multiplient les angles de vues et la présence d'un robot qui crée des occlusions.

Nous montrons que la base de données publique ne permet pas d'entraîner des réseaux suffisamment robustes pour des cas réels. Nous montrons également que l'utilisation d'un système multi-caméra permet d'améliorer la robustesse du réseau et de le rendre plus performant.

ABSTRACT

Manufacturing processes such as deburring or polishing are finishing tasks that can be time-consuming and tedious for workers. Many industries use robots to carry out these tasks, but this requires the setting of parameters that only workers with their expertise can manage. We propose a method of communication between the robot and the operator to make the parameters settings as simple and efficient as possible.

This communication is envisionned through hand gestures, which also enables the operator to show the areas to be corrected directly on the part to be finished. Our solution uses a multi-camera RGB-D system: 6 cameras are distributed in the cobotic cell where the robot and the operator work together to correct the parts.

We have developed a pipeline composed of a hand detection model followed by a hand gesture recognition model. We have tested the models separately and in series, on a public database and found it to perform better than the state of the art on the same dataset. However, this dataset was created in an ideal context, with no occlusion and little variability in viewing angles. This is why we have create a new and richer database with several cameras that multiply the angles of view and the presence of a robot that creates occlusions.

We show that the existing database cannot be used to train networks that are sufficiently robust for real contexts. We also show that the use of a multi-camera system improves the robustness of the network and makes it more efficient.

TABLE DES MATIÈRES

LISTE DES TABLEAUX

LISTE DES FIGURES

LISTE DES SIGLES ET ABRÉVIATIONS

CNRC	Conseil national de recherches du Canada
CTFA	Centre de technologies de fabrication en aérospatiale
BR	Bloc résiduel
TOF	Time-of-flight
IoU	Intersection over union

LISTE DES ANNEXES

CHAPITRE 1 INTRODUCTION ET MISE EN CONTEXTE

De plus en plus d'industriels intègrent des systèmes cobotiques dans leurs usines, l'objectif étant de réduire le nombre de tâches pénibles et répétitives qui causent des problèmes de santé et de sécurité pour les opérateurs dans la chaîne de production.

Pour arriver à ces fins, les industries utilisent généralement des robots préprogrammés avec des codes rigides qui ne peuvent pas coopérer avec les humains. Cependant, certains procédés de fabrication tels que le polissage et l'ébavurage, utilisés pour le parachèvement, impliquent le réglage adéquat de plusieurs paramètres, tels que la géométrie de la pièce à usiner, l'outil requis et la trajectoire de l'outil. Les systèmes cyber-physiques interactifs offrent une solution élégante, donnant à l'opérateur humain essentiellement un rôle de supervision et de contrôle de la qualité. Le concept de collaboration homme-robot ne met pas l'accent sur le remplacement des humains par des robots sur les lieux de travail industriels, mais sur une collaboration entre humains et robots dans un espace de travail commun : la cellule cobotique. Pour que cette collaboration soit sûre et efficace, l'interactivité entre le robot et l'opérateur doit être bien conçue. Dans le contexte de l'industrie 4.0, l'interactivité la plus naturelle est imaginée par la vision par ordinateur et la reconnaissance des gestes.

L'idée est donc de permettre à un ouvrier de choisir le réglage des paramètres de la machine pour que celle-ci s'occupe par la suite de réaliser les procédés de parachêvement sur la pièce fabriquée. Ce dernier sera alors en mesure de vérifier si la pièce a été parachevée correctement et ordonner de nouvelles commandes s'il le juge nécessaire. Pour que cela soit fait de manière optimale, il est souhaitable que l'ouvrier puisse préciser les modifications à faire directement sur la pièce plutôt qu'à travers une application sur un ordinateur. Aussi, il faut le laisser circuler librement dans la cellule cobotique et donc autour de la pièce afin qu'il puisse demander à faire les mêmes modifications que dans un contexte standard où aucun robot n'est utilisé.

Ce projet de recherche est réalisé en collaboration avec le Centre de Technologies de Fabrication en Aérospatiale (CTFA) du Conseil National de Recherches du Canada (CNRC). Celui-ci dispose d'une cellule cobotique composée d'un robot UR10 de Universal Robots, monté sur une table de parachèvement et un ensemble de six caméras RGB-D placées en hauteur tout autour de celle-ci (Figure 1.1). Le projet s'articule autour de 4 modules. Un premier module porte sur la gestion optimale du système multi-caméras, notamment la sélection d'un sous-ensemble de vues optimales en fonction de la localisation de l'opérateur. Un second module porte sur le suivi et la segmentation en continu de l'opérateur avec mise en registre dans le repère du robot pour vérifier des contraintes de sécurité comme la distance minimale entre

l'humain et l'outil placé sur la tête du robot. Un troisième module porte sur la détection d'un signe de pointage du doigt et la détermination subséquente d'une surface à sélectionner sur la pièce à usiner. Enfin, le dernier module porte sur la mise en place d'un système de communication entre le robot et l'opérateur basé sur la gestuelle de l'opérateur.

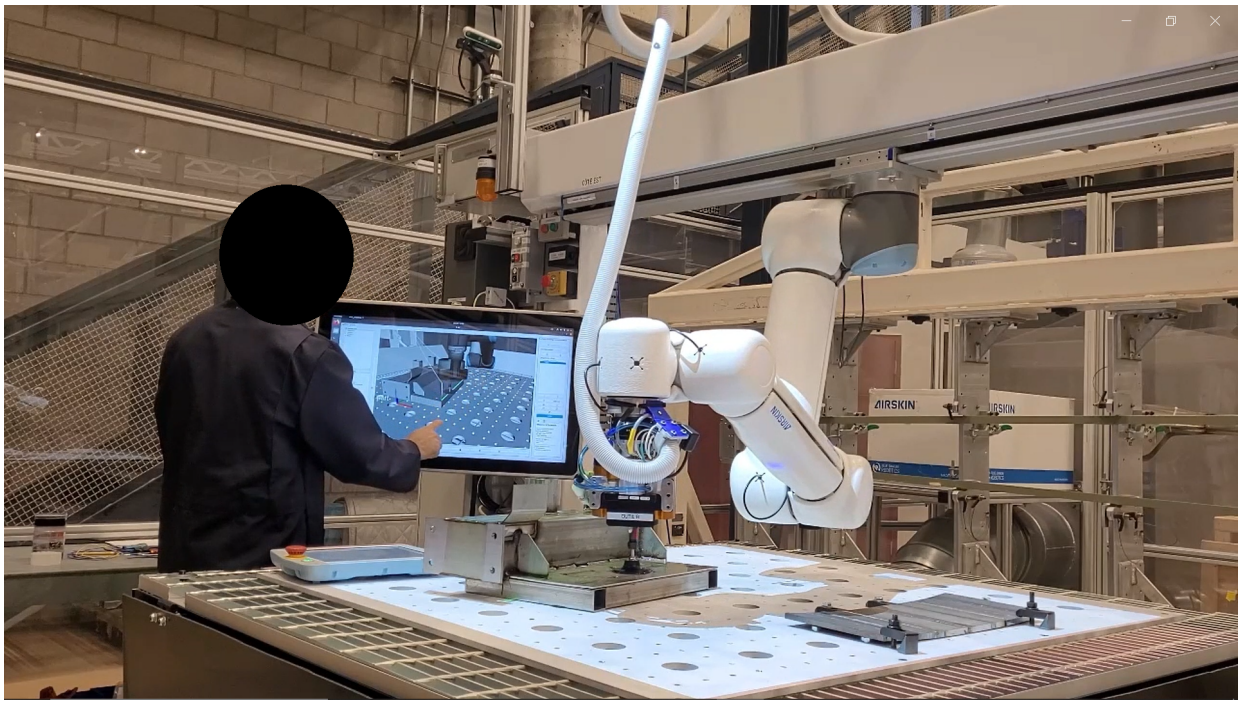

FIGURE 1.1 Cellule cobotique du CTFA.

Le travail de maîtrise présenté dans ce mémoire porte sur le quatrième et dernier module. Plus précisément, l'objectif est de développer un système intelligent capable de reconnaître automatiquement, à travers les flux vidéo des caméras, des gestes de mains de l'opérateur, permettant d'encoder des commandes pour le robot. Ces commandes sont plus précisément en lien avec le choix de l'outil à utiliser et sa taille ainsi que la force à appliquer sur la pièce à retravailler. De plus, le système doit pouvoir reconnaître des gestes indiquant le début et la fin d'une communication avec le robot, ainsi qu'un arrêt d'urgence du robot et l'annulation de certaines modifications demandées.

Le mémoire sera structuré de cette manière : d'abord, une présentation du travail déjà existant dans la littérature scientifique. Ensuite les objectifs de recherche seront expliqués et comparés à ceux de la littérature. La méthode mise en place pour répondre au besoin sera alors exposée. Enfin, les expériences réalisées et les résultats seront présentés avant la conclusion du mémoire.

CHAPITRE 2 REVUE DE LITTÉRATURE

Ce chapitre présente une revue des travaux existants en lien avec le sujet de recherche. La première section porte sur le développement des moyens de communication entre robots et humains. La seconde traite des différentes caméras pour obtenir des images en 3D. La troisième présente les modèles d'intelligence artificielle utilisés pour détecter un objet dans une image et donc par extension, détecter une main. La quatrième s'intéresse à la prédiction de la pose de la main. Enfin, les deux dernières sections présentent des méthodes de reconnaissance de la gestuelle de la main, respectivement des gestes dynamiques et statiques.

2.1 Interactions humains-robots

Le développement des technologies automatiques dans l'industrie ces dernières années a poussé les chercheurs à trouver des solutions pour permettre aux ouvriers d'utiliser ces technologies à travers des ordres donnés directement à la machine. L'objectif étant principalement de développer des moyens de communication avec les machines qui soient à la fois simples et rapides pour augmenter l'efficacité de la production sans pour autant demander de nouvelles compétences aux ouvriers. Entre autres, la communication verbale et gestuelle ainsi que l'analyse de signaux cérébraux font partie des solutions potentielles pour faciliter cette communication [1]. En revanche, la communication verbale peut être compliquée à mettre en place dans certaines industries où le bruit est très présent. Aussi, le fait que les ouvriers portent de l'équipement supplémentaire, dédié uniquement à l'interaction, tel que des gants dotés de capteurs inertiels (accéléromètres) ou une montre connectée est généralement à écarter, car ces équipements créent une contrainte pour l'ouvrier [2].

Malgré les nombreuses recherches dans ce domaine, les interactions humains-machines dans l'industrie doivent toujours être perfectionnées. Il faut également trouver une manière d'évaluer la communication entre les opérateurs et les machines pour convaincre les entreprises d'utiliser de tels outils [3]. Notamment, l'évaluation de la communication devrait comprendre la précision avec laquelle le robot répond aux commandes de l'ouvrier, le temps requis pour que le robot réagisse aux ordres de l'opérateur ainsi que le temps de formation requis par les ouvriers pour apprendre à utiliser ces technologies de manière optimale.

2.2 Modalité de l'imagerie 3D

Cette section traite des différentes caméras permettant l'acquisition d'images RGB-D. Une image RGB-D est une image comprenant à la fois une image en couleurs, stockée sur 3 canaux de 8 bits (rouge, vert et bleu) et une image représentant la profondeur. Les caractéristiques de l'image de profondeur peuvent dépendre de la caméra utilisée. Il existe 3 principaux procédés de mesure pour obtenir de telles images : la triangulation par temps de vol (TOF pour Time-Of-Flight), la triangulation par stéréoscopie et la triangulation par LiDAR [4]. Sont présentées ci-après des caméras utilisant chacun de ces procédés.

2.2.1 Les caméras TOF

Les caméras temps de vol ou « time of flight » permettent de créer une image de profondeur en émettant une impulsion lumineuse infrarouge (ordre de 850nm) réfléchie par les objets à détecter. À l'aide d'un calcul liant la vitesse de la lumière et le temps entre l'émission et la réception de la lumière émise, il est possible d'estimer la distance à laquelle se situent les objets.

Les caméras TOF sont les premiers appareils à prix abordable à être arrivés sur le marché, notamment avec la caméra Kinect v2 de Microsoft en 2013. Cette caméra a été rapidement adoptée afin d'améliorer les résultats pour la navigation de robots autonomes [5]. De plus, plusieurs études ont depuis été réalisées pour améliorer la calibration de la caméra et augmenter la précision de l'image de profondeur [6]. Une nouvelle version de cette caméra a été développée par Microsoft : la Kinect Azure (entre 400$ et 500$), celle-ci ayant été largement testée également [7–9].

Un des problèmes des caméras TOF est leur limitation à être utilisées conjointement, dans un système multicaméras. En effet, si plusieurs caméras TOF sont disposées dans une même pièce, les rayons lumineux émis par chaque caméra vont interférer entre eux, diminuant ainsi la précision de l'image de profondeur de chaque caméra. Cependant, des travaux sont proposés pour réduire ces interférences par des méthodes de preprocessing [10].

2.2.2 Les caméras stéréoscopiques

Les caméras stéréo ont un principe de fonctionnement très différent des caméras TOF puisqu'elles fonctionnent sur le même principe que celui de la vision humaine. Une caméra stéréo comprend donc deux capteurs photographiques qu'il faut comparer pour obtenir la profondeur. Nous noterons que plus la distance entre les deux capteurs est élevée, plus la précision sur les objets éloignés sera grande, en revanche, la distance minimale à laquelle les objets

doivent se trouver augmente également. Il est donc important de bien choisir la caméra stéréo dépendamment de l'application [4].

Les caméras stéréo ont largement été développées avec la série de caméras D400 de la compagnie Intel RealSense. Le prix de telles caméras étant aussi abordables (entre 250$ et 450$), elles ont donc fait concurrence aux caméras TOF. Dans cette série de caméras, nous trouvons les caméras D415, D435 et D455 dont la principale différence est la distance entre les deux capteurs de profondeurs et donc la distance à laquelle la précision est optimale. Ces appareils ont largement été comparés, des techniques avancées ont été mises en place pour comparer ces différentes caméras. Finalement, la caméra D455 serait tout simplement meilleure en termes de précision que la D435. La D415 serait en revanche plus stable que les deux précédentes. Cependant, différentes zones dans une région d'intérêt ne sont pas équitablement détectées par la D415, en conséquence, certaines zones de l'image de profondeur sont plus précises que d'autres [11].

La série D400 de Intel permet de configurer les caméras en mode passif ou actif. Le mode actif utilise une source de lumière infrarouge pour noyer la scène dans une texture synthétique permettant de faciliter la mise en correspondance entre les images des deux caméras. Le mode passif n'utilise aucune projection de lumière et seule la texture naturelle de la scène permet la mise en correspondance. Le mode actif permet donc d'obtenir des nuages de points plus denses et plus fidèles de la géométrie de la scène. Le fait que la texture projetée ne suit pas un pattern précis pour la triangulation, l'utilisation combinée de plusieurs caméras D400 ne limite pas la précision des nuages de points, au contraire elle enrichit la texture synthétique par superposition de plusieurs sources de lumière [12].

2.2.3 Les caméras LiDAR

Le fonctionnement des caméras LiDAR est similaire à celui d'une caméra TOF puisqu'il repose sur le principe de la mesure du temps de vol de la lumière. Contrairement aux caméras TOF, les LiDAR envoient des milliers d'impulsions laser dans différentes directions. Les résultats sont donc assez différents des caméras TOF ou stéréo puisque cette fois, nous obtenons un nuage de points et leurs coordonnées dans un repère ayant souvent pour origine la caméra. Ces caméras ne retournent pas d'images RGB.

Ces caméras n'ont pas de problèmes d'interférences grâce à l'utilisation de lasers. Leur utilisation est donc plus simple pour des dispositifs multicaméras. La précision de ces caméras est également meilleure que les caméras TOF ou stéréo. Ce sont les raisons pour lesquelles ces caméras sont largement utilisées pour l'élaboration de voitures autonomes où la robustesse de la vision est primordiale [13–15]. Cependant, les matériaux nécessaires pour la fabrication

d'une telle caméra sont souvent plus coûteux. Le prix final est donc plus élevé que pour les caméras précédentes.

Les nuages de points obtenus avec les LiDAR ne sont pas facilement interprétables et des études ont montré que fusionner les images de couleur avec les nuages de points permettait non seulement d'interpréter la profondeur captée par la caméra mais aussi d'améliorer les résultats [16].

2.3 Détection de la main dans une image

Cette section s'intéresse à la détection de la main dans une image, cas particulier de la détection d'objets dans une image. Ce problème peut être traité de deux manières différentes, soit par la détection directe de l'objet dans l'image, soit par le suivi d'un objet d'une image à l'autre.

2.3.1 Détection de l'objet à partir d'une image seule

Depuis 20 ans, la détection d'objet suscite l'intérêt des chercheurs et le nombre d'articles publiés autour de la détection d'objets n'a cessé d'augmenter depuis. De nombreuses architectures de réseaux de neurones ont été développées et testées sur les bases de données [17]. Aujourd'hui, la recherche sur la détection d'objets a pour objectif d'améliorer la précision de la détection et la rapidité d'exécution. Certaines architectures tendent plutôt à favoriser la rapidité d'exécution : c'est le cas de l'architecture YOLO [18]. D'autres se concentrent sur la précision comme faster RCNN [19]. De plus, certains utilisent la profondeur pour améliorer les résultats [19].

Un autre axe d'amélioration visé par les chercheurs est de créer des réseaux ne nécessitant que peu de données d'entraînements. Cela revient à utiliser des algorithmes de meta-learning appliqués à la détection d'objets [20]. L'avantage de tels algorithmes est qu'ils ne nécessitent pas de créer une nouvelle base de données à chaque fois que l'on veut suivre un nouvel objet. Cependant, des bases de données annotées de mains existent ainsi que des appareils permettant la détection de la main automatiquement. Nous présenterons ces bases de données et appareils dans la section 5 de ce chapitre.

2.3.2 Suivi de l'objet dans une séquence vidéo

Le suivi d'objets dans une image peut s'avérer complexe à cause de la présence de plusieurs objets d'intérêts qu'il faut associer entre les différentes images et cela implique souvent de

découper le problème en plusieurs processus [21]. Dans le cas des mains par exemple, il est souvent nécessaire de bien faire la distinction entre les mains droite et gauche, il ne faut pas non plus confondre les mains d'un utilisateur avec celles d'un autre. Par ailleurs, un autre problème du suivi d'objets est qu'il est souvent nécessaire d'avoir la position initiale de l'objet pour ensuite pouvoir le suivre, ce qui implique soit de commencer par une détection d'objets, soit d'indiquer manuellement la position de l'objet sur la première image de la séquence.

Dernièrement, le suivi d'objets en 3D multi-vues a connu beaucoup d'avancées notamment car il est très utilisé pour développer les voitures autonomes qui est un axe de recherche très investigué [22]. Une méthode régulièrement utilisée consiste à analyser le mouvement des objets pour pouvoir ajuster leur position dans les images successives [23].

Enfin, il existe des méthodes de suivi de points qui peuvent également s'avérer utiles pour suivre un objet. A partir de points posés préalablement sur la main, des algorithmes utilisant des transformers comme TAPIR retrouvent les points correspondants dans l'intégralité de la séquence vidéo fourni [24]. Puisque ce type d'algorithme fonctionne dans le cas général pour n'importe quel point appartenant à n'importe quel objet, un des avantages est qu'il ne nécessite pas de réentraîner le modèle sur des données adaptées. Toutefois, le modèle TAPIR est lourd et complexe.

2.4 Estimation de la pose de la main

Dans cette section, les méthodes pour estimer la pose de la main seront couvertes. L'estimation de la pose de la main consistant à prédire à l'aide d'algorithmes ou outils, la position des différentes articulations de la main dans une image en 2D ou en 3D. L'intérêt étant de réduire l'information d'une image à plusieurs canaux et des centaines de pixels à un vecteur de coordonnées qui est plus facile à traiter pour prédire la gestuelle de la main. L'estimation de pose dans un contexte général est un problème utile dans de nombreux domaines tel que le sport, le médical et la robotique [25].

On notera que pour les méthodes présentées ci-après, si elles ne requièrent pas de trouver la main dans une image avant de prédire la pose, alors, ces méthodes résolvent également le problème de détection de la main.

2.4.1 Estimation de pose à l'aide de gants

Beaucoup d'études ont cherché à estimer la pose de la main en temps réel grâce à des dispositifs portables, notamment des gants électroniques [26]. Cependant, un des problèmes majeur des gants est qu'il est nécessaire de trouver un moyen de le relier à l'ordinateur sans

gêner le porteur et sans diminuer les performances de l'outil [27]. De plus, même si les gants obtiennent de bons résultats, lorsque le mouvement est trop rapide, les capteurs ont tendance à voir leur précision diminuer [28].

2.4.2 Estimation de pose avec la Leap Motion Controller

En 2012, l'entreprise Leap Motion a présenté un dispositif de capture de mouvements des mains. Cette caméra, initialement créer pour interagir avec un ordinateur en réalité virtuelle, a l'avantage de prédire l'estimation de la pose de la main. De plus cette caméra a un coût (entre 100$ et 200$ [29]) plus faible que celui des autres caméras mentionnées plus tôt. Cependant, la Leap Motion Controller garantit une précision dans la pose estimée de la main que lorsque celle-ci se trouve à 25cm de la caméra [30]. Par conséquent, pour utiliser un tel appareil, il est donc nécessaire de le porter sur soi ou bien de rester en face de la caméra sans se déplacer.

2.4.3 Algorithmes de prédiction de la pose de la main

Il est possible d'estimer la pose de la main à partir d'images, avec ou sans profondeur. De nombreux algorithmes ont été développés pour prédire la pose de la main. Mediapipe fournit par exemple un modèle pré-entraîné (mediapipe hands) estimant la pose de la main avec une image RGB avec une précision élevée [31]. Mediapipe hands est divisé en deux réseaux différents : un détecteur de la paume de la main et réseau qui estime la pose de la main à partir de la position de la paume.

D'autres auteurs cherchent à retrouver la position des articulations et ensuite à les identifier [32]. Un problème récurrent dans ces travaux est l'occlusion des mains sur l'image, notamment quand les personnes manipulent des objets. Des études cherchent à exploiter les objets qui créent l'occlusion pour prédire la pose de la main [33].

2.5 Reconnaissance de la gestuelle de la main

Cette section présente les recherches qui ont été menées dans la littérature sur la reconnaissance de la gestuelle de la main dans le cas où le mouvement est dynamique ou statique. Les travaux incluent non seulement les algorithmes qui servent à prédire la gestuelle mais aussi les bases de données qui sont créées pour répondre à ce problème.

Par la reconnaissance de la gestuelle de la main, nous entendons ici la classification des gestes, ce qui veut dire que chaque réseau de neurones n'est entraîné à reconnaître que certains gestes

prédéfinis.

2.5.1 Reconnaissance de la gestuelle dynamique

Algorithmes et méthodes utilisés

La prédiction de la gestuelle de la main lorsque le mouvement est dynamique implique l'utilisation de réseaux de neurones prenant en considération l'échelle temporelle. C'est la raison pour laquelle dans la littérature, les réseaux utilisés sont souvent des réseaux de neurones récurrents (RNN) ou des réseaux long short-term memory (LSTM). Or, ces réseaux ne sont pas adaptés pour recevoir en entrée des images, c'est la raison pour laquelle nous travaillons généralement à partir de l'estimation de pose pour la gestuelle dynamique [34].

La prédiction de la gestuelle de la main à partir de l'estimation de pose de mediapipe a été testée [35]. Selon les auteurs, la précision de la classification des différents gestes est globalement satisfaisante. Toutefois, les auteurs démontrent que la performance diminue dans des cas d'auto-occlusions dépendamment du point de vue de la caméra par rapport à la main.

D'autres ont testé à partir de l'estimation de pose de la Leap Motion Controller. Là aussi, l'estimation de pose est suffisamment précise pour permettre la classification des gestes avec cette fois un RNN bidirectionnel. Nous noterons qu'il est néanmoins nécessaire de garder la main dans la zone de détection de la Leap Motion qui est assez réduite [36]. Pour augmenter la précision dans des zones plus larges, certains ont essayé d'utiliser plusieurs caméras Leap Motion Controller répartis dans l'espace. La zone reste néanmoins plus restreinte qu'avec des algorithmes comme mediapipe [37].

Base de données

Une des raisons principales pour laquelle la reconnaissance de la gestuelle de la main est recherchée est parce qu'elle pourrait permettre de traduire en temps réel le langage des signes et donc permettrait une meilleure inclusion dans la société pour les personnes sourdes ou muettes. Par conséquent, nous pouvons trouver des bases de données comprenant les signes des différentes langues gestuelles américaine, chinoise et allemande par exemple. Ces bases de données peuvent être composées de séquences vidéo avec profondeur entièrement annotées avec le texte associé aux vidéos [38–40].

On peut également trouver des bases de données contenant des signes destinés à être utilisés pour permettre à un conducteur de communiquer avec une voiture autonome [41].

2.5.2 Reconnaissance de la gestuelle statique

Algorithmes et méthodes utilisés

La reconnaissance de la gestuelle statique est un problème plus simple que celui de la gestuelle dynamique car la composante temporelle n'est plus à prendre en compte en amont. Elle peut toutefois être exploitée en aval pour corriger la prédiction en fonction des prédictions sur les images voisines. C'est la raison pour laquelle des réseaux de convolutions (CNN) peuvent suffire à résoudre le problème.

Il est possible d'avoir une approche similaire au cas de la reconnaissance dynamique mais la recherche montre qu'estimer la pose de la main n'est plus un prérequis : l'article ME-GURU présente une méthode pour prédire les gestes simplement en détectant les mains [42]. Aussi, certains essaient d'utiliser le fait que le problème soit plus simple pour tester d'autres méthodes. Par exemple en utilisant seulement l'image de profondeur fournie par la caméra Kinect et en cherchant des caractéristiques de l'image tel que le nombre de doigts levés ou la distance entre le bout des doigts et la paume [43]. En revanche, sur des gestes plus complexes, il est probable que ce type de méthode voit sa précision diminuer.

Base de données

Comme pour la reconnaissance dynamique, nous trouvons des bases de données liées à la traduction du langage des signes américain [44] mais aussi dans d'autres langues comme le malaysien [45]. Nous trouvons également une base de données, HGM-4, utilisant 4 angles de vues différents pour permettre de développer des modèles plus robustes [46]. Cependant, la base de données HGM-4 ne comprend que des images dans lesquelles a été réalisé une soustraction de background. Par conséquent, l'entraînement d'un algorithme pour la détection de la main n'est pas possible avec ces données.

Un défi dans la création de base de données est l'annotation des données qui peut être lente et fastidieuse surtout si elle est destinée à entraîner des modèles pour l'estimation de pose ou la détection de la main. Il existe cependant des astuces pour annoter les bases de données de manière semi-automatique, par exemple en utilisant des gants [45].

Dans le cadre de la robotique, nous pouvons également trouver des bases de données ayant pour objectif la communication avec un robot [47].Cependant, pour toutes les bases de données présentées, celles-ci sont souvent créées dans un contexte simplifié où l'utilisateur est placé de manière idéale, c'est à dire exactement face à la caméra, à une distance fixe et où la variabilité dans les données est de manière générale assez faible.

CHAPITRE 3 OBJECTIF DE RECHERCHE

Ce chapitre revient brièvement sur les articles présentés précédemment dans la revue de littérature pour appuyer les objectifs du projet de recherche. Le but étant d'expliciter ce que le travail réalisé dans le cadre de ce projet apporte au niveau scientifique. Par conséquent, ce chapitre consiste en une critique plus approfondie du contenu scientifique actuel pour justifier les choix faits et le travail exposé dans la suite du mémoire.

Dans la mesure où nous souhaitons laisser l'opérateur libre de se déplacer dans la cellule cobotique et de la manière la plus naturelle possible, les gants électroniques dotés de capteurs pour détecter la main ou estimer la pose ne seront pas considérés puisque ceux-ci ajoutent une contrainte physique pour l'ouvrier [2].

De plus, nous avons vu que l'utilisation de plusieurs caméras permettait d'augmenter le champ de visibilité. Dans notre contexte, l'utilisation de plusieurs caméras pourrait donc être nécessaire pour maximiser la liberté de mouvement dans la cellule cobotique. Parmi les trois types de caméras RGB-D présentés à la section 2.2, le type de caméras qui paraît le plus adapté à un montage multi-caméra est la stéréo active [10]. Les caméras de la série D400 de la compagnie Intel Real Sense semblent être un bon choix, leur prix étant très abordable. Nous retiendrons dans ce projet la D455 qui convient le mieux à la distance à laquelle les ouvriers se situeront par rapport aux caméras [11].

Dans notre contexte où les commandes à donner au robot consistent à lui préciser l'outil à utiliser, la force à appliquer sur la pièce ou le nombre de passages à exercer, il n'est pas nécessaire d'utiliser des gestes dynamiques. En effet, pour faire de tels gestes, il n'y a pas besoin de distinguer différentes formes tracées à partir d'un mouvement ou d'une direction que l'ouvrier voudrait pointer. D'autant plus que la reconnaissance de la gestuelle dynamique est plus lourde en ressources computationnelles et en temps de calcul. Par conséquent, l'estimation de pose de la main peut être contournée.

L'objectif principal du sujet de recherche étant, au final, de reconnaître la gestuelle statique de la main pour permettre à des ouvriers de communiquer avec un robot, il est très similaire à celui de l'article MEGURU [42]. Pour résumer, dans cet article, les auteurs ont créé un dictionnaire de gestes ainsi qu'une méthode de communication particulière : la signification des gestes est différente selon l'état dans lequel se trouve le robot. Ils ont donc entraîné un réseau de neurones pour reconnaître les différents gestes effectués à partir d'une base de données (HANDS) dont nous pouvons trouver les détails dans l'article associé à MEGURU [47]. Cependant, comme expliqué dans la revue de littérature, cette base de données a été

conçue dans un contexte idéal ou le sujet se trouve face à la caméra, et avec peu de variabilité dans les données. Nous pouvons notamment préciser qu'il n'y a que 5 sujets différents dans la base de données publiée. Si dans l'article, il est précisé qu'ils ont utilisé de l'augmentation de données pour éviter l'influence des changements de luminosité par exemple, l'augmentation de données a des limites. Il semble donc peu probable d'arriver à entraîner un réseau à partir de ces données qui soit résistant aux variations d'orientation hors du plan par exemple. De plus, cette base de données ne permet de reconnaître les gestes que lorsque l'opérateur se trouve exactement devant la caméra, ce qui n'est jamais le cas dans notre scénario où l'ouvrier doit être libre de se déplacer autour de la cellule cobotique.

Les hypothèses que nous souhaitons vérifier sont donc les suivantes :

1. La base de données HANDS créée dans des conditions idéales ne permet pas une généralisation à des conditions réelles plus complexes.
2. L'utilisation d'un systéme multi-caméra permet l'augmentation des performances dans des conditions réelles notamment en présence d'occlusions visuelles.

Par conséquent, le travail vise à atteindre les objectifs suivants :

1. Développer un pipeline permettant la détection de la main et la reconnaissance de la gestuelle de la main.
2. Évaluer les performances du modèle développé à l'objectif 1 sur la base de données HANDS et comparer ces performances avec les résultats de l'article MEGURU.
3. Créer une nouvelle base de données avec une variabilité plus importante d'angles de vue, de déplacements, du nombre de participants et avec plusieurs caméras réparties dans l'espace.
4. Déterminer une méthode pour fusionner les prédictions de gestes des différentes caméras.
5. Évaluer les performances du modèle développé à l'objectif 1 sur la nouvelle base de données découlant de l'objectif 3.

CHAPITRE 4 MÉTHODE

Ce chapitre présente les méthodes proposées d'une part pour la reconnaissance de la gestuelle de la main et d'autre part, pour la création d'une base données d'images riche en variabilité pour augmenter la robustesse de la reconnaissance de la gestuelle.

4.1 Architecture proposée pour l'interaction humain-robot basée sur la reconnaissance de la gestuelle

Cette section discute des solutions techniques proposées pour que l'opérateur puisse communiquer avec le robot par la gestuelle. Une première sous-section portera sur la méthode de communication, ensuite la deuxième sous-section définit l'environnement de la cellule et le positionnement des caméras. Enfin la dernière sous-section porte sur le développement des différentes étapes du pipeline proposé pour la reconnaissance de la gestuelle....

4.1.1 Méthode de communication

Cette sous partie présente les différents gestes définis et leurs séquences qui permettent de donner des commandes au robot dans le contexte précis de parachèvement cobotique.

Dictionnaire de gestes

Pour rappel, la communication gestuelle entre l'opérateur et le robot doit permettre à l'ouvrier de commencer une discussion, de donner au robot l'ordre d'exécuter une tâche, d'arrêter le robot et de choisir des paramètres tels que l'outil à utiliser, la taille de l'outil ou la force à appliquer. Il est donc important que l'ensemble des gestes intégrés dans le système de communication permette ces actions. Les gestes qui ont été retenus pour la communication dans la cellule cobotique sont illustrés dans la figure 4.1.

Aussi, il est souhaitable que ce mode de communication puisse s'adapter facilement à différentes industries. L'utilisation de nouveaux outils, l'apparition de nouveaux paramètres ne doit pas compromettre le dictionnaire de gestes établi. Autrement, l'ajout d'un nouveau geste nécessiterait d'entraîner à nouveau un modèle de reconnaissance avec une nouvelle classe. C'est la raison pour laquelle les gestes choisis pour la sélection de paramètres sont des chiffres et n'ont donc pas de significations particulières liées à un contexte industriel.

Les chiffres (incluant le poing qui peut être considéré comme un 0 ou un chiffre additionnel

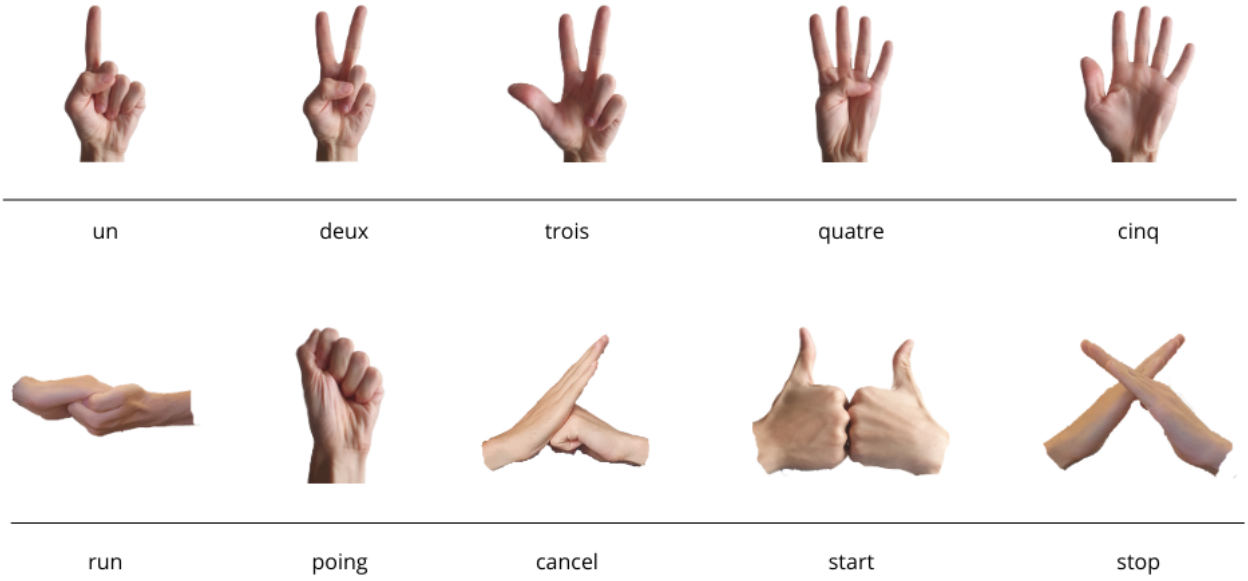

FIGURE 4.1 Ensemble des gestes utilisés pour la communication avec le robot.

au besoin) seront donc utilisés pour la sélection de paramètres tels que l'outil à utiliser ou sa taille. Les autres gestes sont des gestes nécessitant les deux mains. L'intérêt d'utiliser des gestes à deux mains est qu'il n'y a que peu de chances qu'un opérateur qui passe dans la cellule cobotique fasse ces gestes par inadvertance. Ces gestes seront donc utilisés pour les commandes les plus critiques comme démarrer la communication (geste : `start`), annuler les gestes précédents (geste : `cancel`), lancer l'exécution d'une tâche (geste : `run`) ou arrêter le robot lorsque celui-ci travaille (geste : `stop`).

Exemple de communication

Afin d'expliquer le mode de communication imaginé entre l'opérateur et le robot, il faut d'abord définir les états dans lequel le robot peut se trouver lorsque l'utilisateur commence la conversation. Nous considérons qu'il existe deux cas différents où l'opérateur souhaiterait communiquer avec le robot :

- Premièrement, lorsque le robot est à l'arrêt et que l'opérateur souhaite lui donner un ordre pour parachever une pièce qu'il aurait préalablement déposée sur la table de la cellule cobotique.
- Deuxièmement, lorsque le robot travaille déjà sur une pièce mais que l'opérateur souhaite l'arrêter pour vérifier la tâche en cours.

Nous définissons donc deux états différents : l'état d'attente du robot et l'état de fonctionnement du robot.

Dans le premier cas où le robot est en état d'attente, la figure 4.2 montre la séquence de gestes qui permet de lancer l'exécution d'une tâche, tout en choisissant les différents paramètres. Initialement, le robot ne répond qu'à un seul geste : le geste `start`. Si un autre signe est fait, le robot l'ignore. Ensuite, un signe parmi les chiffres permet de choisir l'outil (pour chaque industrie, nous pouvons associer un chiffre à un outil, par exemple au CTFA, il y a 3 outils possibles : un outil d'ébavurage, de polissage et un numériseur optique qu'on peut identifier par les chiffres 1, 2 et 3). Par la suite, l'opérateur montre directement sur la pièce la zone à retravailler, puis il peut choisir d'autres paramètres à l'aide de chiffres. Enfin il peut demander à exécuter la tâche avec le geste `run`. Nous noterons qu'à tout moment, l'opérateur peut annuler le dernier geste qu'il a fait avec le geste `cancel` ou annuler toute la communication qui a eu lieu depuis le geste `start` avec le geste `stop`.

Aussi, pour que l'opérateur soit certain que le robot ait bien compris les gestes qu'il a réalisés, un vidéo projecteur installé dans la cellule cobotique pourrait être utilisé afin d'indiquer directement les paramètres qui ont été sélectionnés sur la table. La zone à retravailler sur la pièce pourrait également être projetée pour que l'opérateur puisse vérifier la précision de la détection du geste de pointage. Toutefois, cette partie de retour d'information sort du cadre de ce travail de maîtrise.

Dans le cas où le robot est en état de fonctionnement, la figure 4.3 montre les séquences de gestes possibles. Lorsque le robot est en fonctionnement, le seul geste que le robot s'attend à recevoir est le geste `stop`. Ce geste met le robot en pause, ce qui permet à l'ouvrier de pouvoir vérifier la tâche en cours en toute sécurité. L'ouvrier peut alors demander de continuer la tâche en cours avec le geste `run` ou l'arrêter avec le geste `cancel` s'il y a un problème.

4.1.2 Structure physique

La cellule cobotique est composée d'une table en son centre sur laquelle nous pouvons déposer manuellement les pièces à parachever. Le robot est positionné au dessus de cette table. Un schéma d'une vue de haut de la cellule cobotique est présenté à la figure 4.4. La zone hachurée à droite de la table sur le schéma est inaccessible à l'opérateur, cette zone contient, entre autre, le système d'alimentation du robot et la circulation n'y est pas aisée. Les 6 caméras RGB-D sont disposées, en hauteur, autour de la table de sorte à couvrir de manière optimale les zones accessibles par l'opérateur. Il y a deux hauteurs auxquelles les caméras sont placées. Le premier niveau se situe à un peu plus de 2 mètres de haut et le second à près de 2,3 mètres. Seulement 2 caméras sont disposées sur le niveau le plus haut. Nous pouvons voir les angles de vue obtenus sur la figure 4.5.

On définit quatre zones autour de la table. Chaque zone est couverte par au moins deux

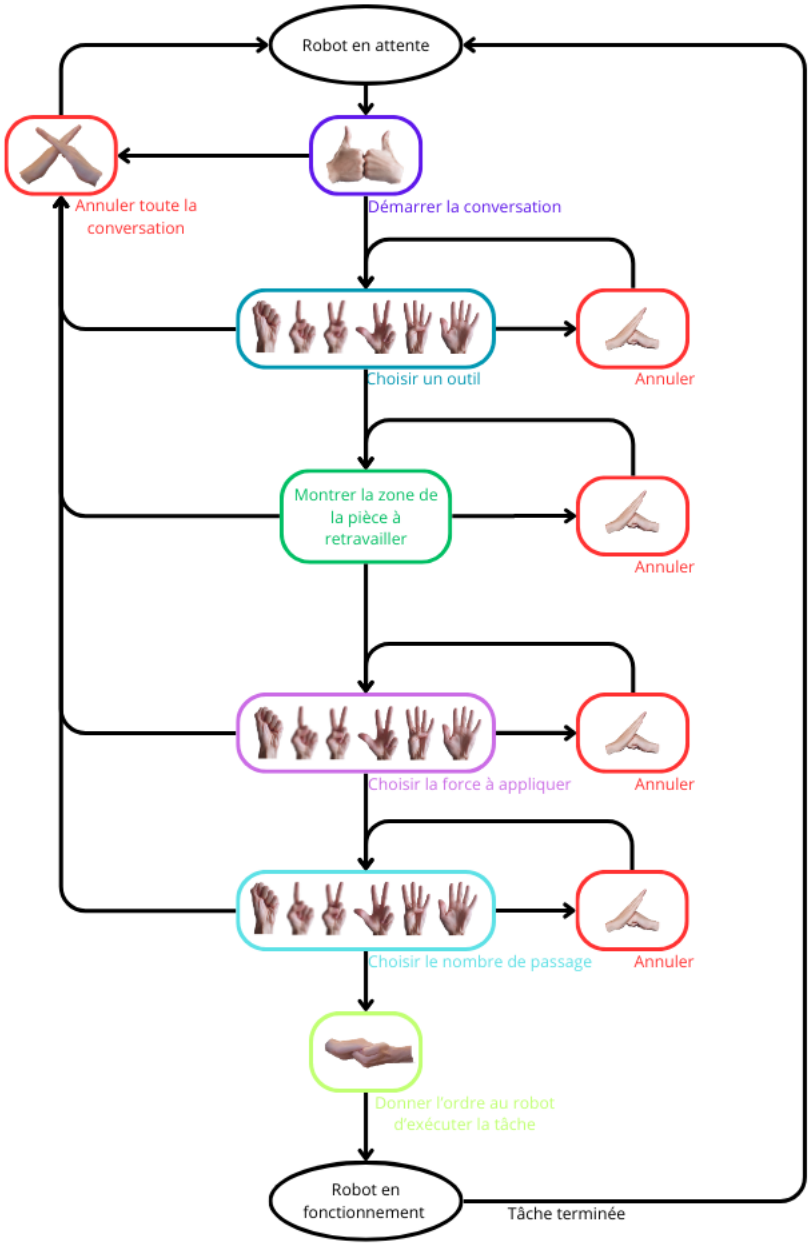

FIGURE 4.2 Diagramme de communication entre un opérateur et le robot lorsque le robot est initialement en attente.

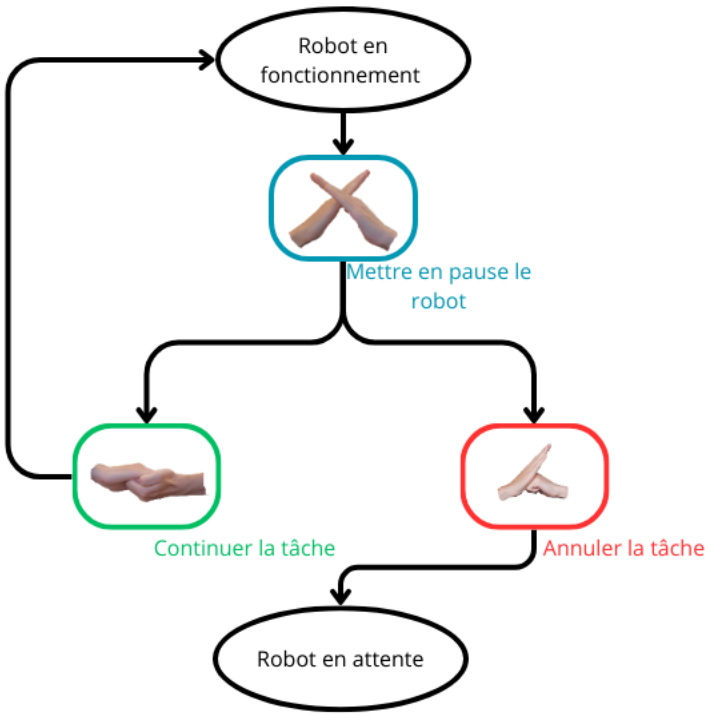

FIGURE 4.3 Diagramme de communication entre un opérateur et le robot lorsque le robot est initialement en fonctionnement.

caméras dont deux sont à un niveau de hauteur différent. Cela permet d'augmenter le nombre de points de vue dans chaque zone. Ainsi, les zones 1, 2, 3 et 4 sont respectivement couvertes par les caméras (c4, c6), (c2, c4, c5), (c2, c3, c5) et (c1, c3).

4.1.3 Structure informatique

Cette sous partie présente l'architecture informatique mise en place pour reconnaître les gestes de la main de l'opérateur à partir des images des six caméras de la cellule cobotique.

Structure globale

Dans la cellule cobotique, un seul ordinateur est disponible pour traiter les images provenant des six caméras, c'est la raison pour laquelle il n'est pas souhaitable de travailler avec les images des six caméras à un débit aussi élevé que 30 FPS. En effet, cela ne permettrait pas la reconnaissance des gestes en temps réel.

Pour éviter ce problème, les caméras auront deux états de fonctionnement à deux frame rate différents : le premier est un état de repos où la caméra fonctionne à un frame rate faible en vérifiant qu'il n'y ait pas d'opérateur dans la cellule. Le second est un état de fonctionnement rapide déclenché par la présence d'un utilisateur dans la cellule. Lorsqu'un opérateur entre dans une des quatre zones, les caméras associées à cette zone entrent donc en état de fonctionnement.

L'architecture informatique globale comprend donc quatre parties distinctes présentées sur la figure 4.6 :

1. La détection d'un opérateur dans une zone et donc l'activation des caméras les mieux placées.
2. La détection de la main dans les images R-GBD des caméras bien placées.
3. La reconnaissance du geste à partir des mains détectées par chaque caméra.
4. L'association des différentes prédictions de gestes provenant des différentes caméras.

La première étape de cette architecture est incluse dans un projet de recherche distinct. Ainsi, dans la suite du mémoire, nous ferons donc l'hypothèse que nous disposons seulement des images provenant des caméras bien placées (comme indiqué sur la figure 4.4), permettant une vision optimale de l'opérateur en fonction de la zone ou il se trouve. Les étapes suivantes sont détaillées ci-après.

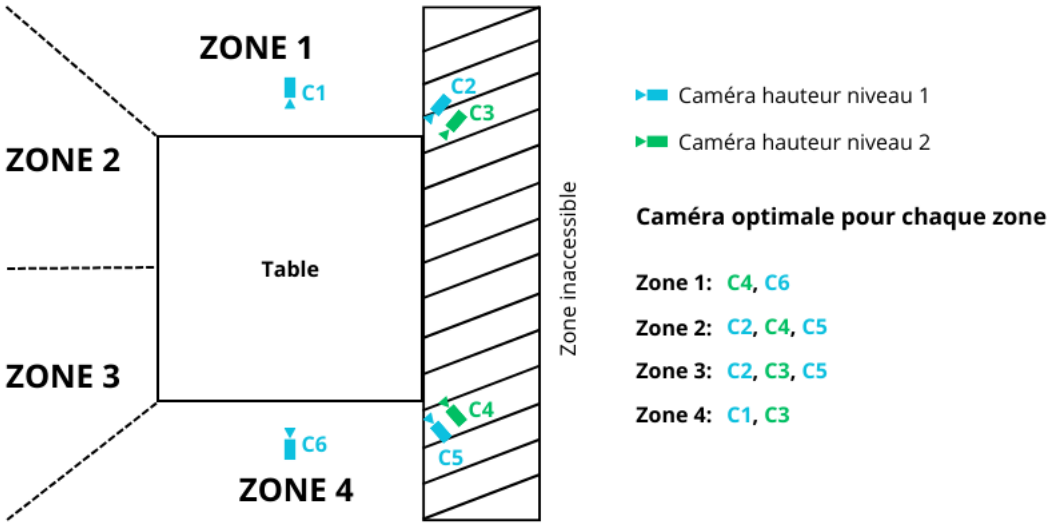

FIGURE 4.4 Schéma de la cellule cobotique avec la disposition des caméras

Détection de la main dans l'image

Pour cette étape, l'objectif est d'obtenir, à partir des images en sortie des caméras RGB-D, les boîtes englobantes dans lesquelles se trouvent les mains de l'opérateur.

Pour cela, nous proposons d'utiliser l'outil Mediapipe [31]. Mediapipe fonctionne seulement à partir d'images RGB, mais cela n'est pas un problème puisqu'il suffit de retirer le dernier canal de profondeur de nos images RGB-D pour obtenir des images RGB analysables par le modèle. Mediapipe a été conçu pour l'estimation de pose de la main et contient deux modèles différents : un premier permet d'obtenir la position de la paume de la main, à partir de cette position, le second modèle prédit la pose des différentes articulations de la main. Au total, 21 points sont détectés par le modèle. Un pour la paume et quatre pour chaque doigt. Un exemple d'estimation de pose permis par Mediapipe est montré en figure 4.7.

Une manière simple d'obtenir la boîte englobante à partir de l'estimation de pose de la main est de prendre les abscisses ou ordonnées des articulations les plus petites (resp. les plus grandes) en valeur et de les utiliser comme nos points en haut à gauche (resp. en bas à droite) du rectangle englobant. Un exemple du résultat qu'on obtient avec cette technique est montré en figure 4.8.

Cependant, les articulations sont souvent à l'intérieur des doigts de la main et les boîtes englobantes obtenues avec cette première solution sont souvent trop petites et coupent les bords de la main. C'est pourquoi, nous proposons d'agrandir ces boîtes englobantes d'un

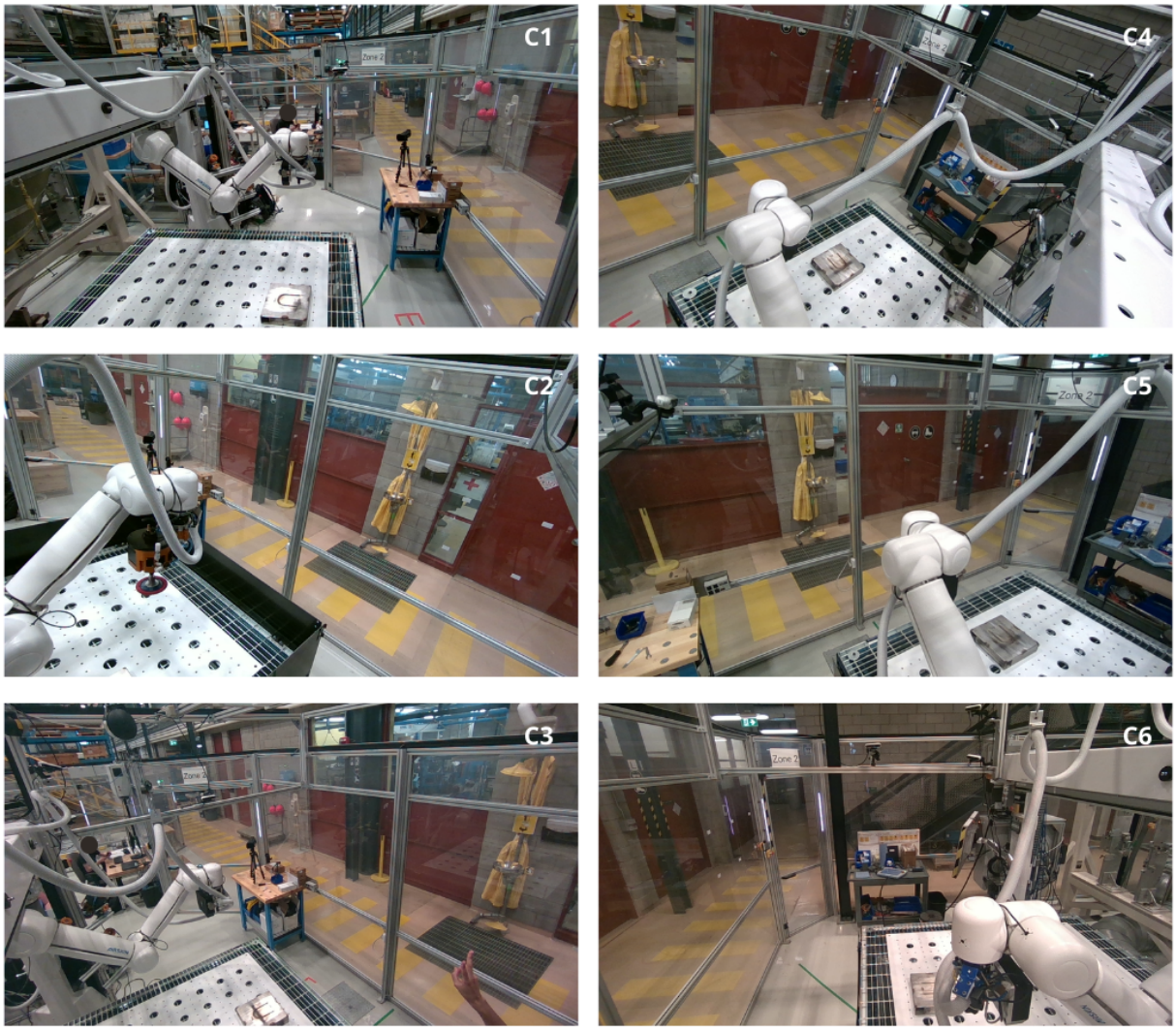

FIGURE 4.5 Angles de vue des six caméras autour de la cellule cobotique.

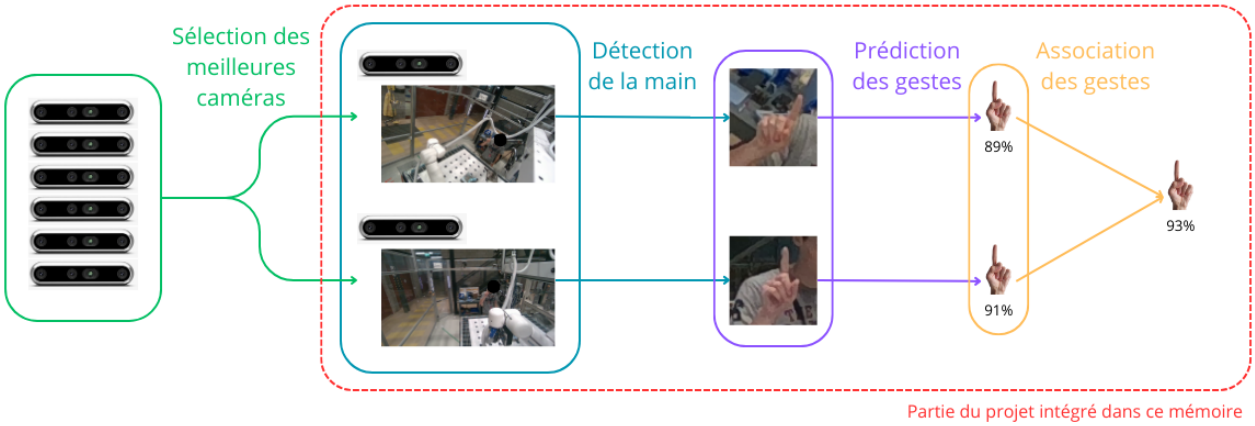

FIGURE 4.6 Architecture informatique globale du projet.

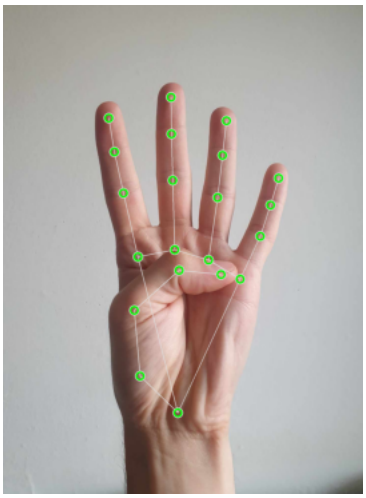

FIGURE 4.7 Exemple d'estimation de pose donnée par Mediapipe.

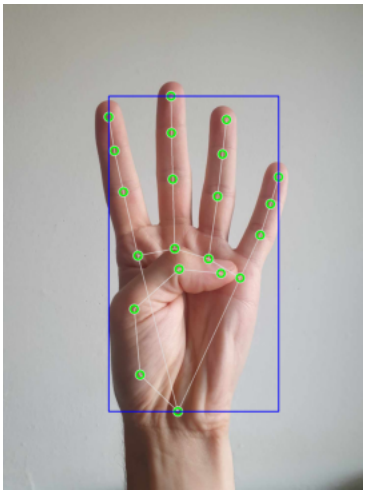

FIGURE 4.8 Exemple de boîte englobante obtenue à partir de l'estimation de pose de Mediapipe.

facteur *coef* à déterminer et en gardant le même centre du rectangle. Cela permet d'obtenir des rectangles englobants qui ne coupent plus la main, la figure 4.9 montre les rectangles englobants pour des coefficients différents.

Reconnaissance du geste

Pour cette sous section, nous considérons maintenant que nous avons la boîte englobante entourant la main. Cela revient à considérer que nous possédons une image de taille variable au format RGB-D contenant la main.

L'objectif est de reconnaître le geste réalisé parmi ceux du dictionnaire de données. Il faut donc un modèle capable de prendre en entrée une image de taille variable et de retourner un vecteur $[p_1, p_2, \ldots, p_{10}]$ avec $\sum_{i=1}^{10} p_i = 1$, p_i représentant la probabilité que le geste i soit réalisé en sortie.

Pour traiter ces images, il est nécessaire de passer par une méthode de prétraitement pour que les images aient toutes le même format en entrée du réseau. Le réseau développé prend en entrée des images de taille 100x100. Une fois que l'image initiale est rognée pour retirer tout ce qui est hors de la boîte englobante, nous redimensionnons l'image par interpolation bilinéaire de sorte qu'elle soit de taille 100x100. Nous ne créons pas de déformations dans l'image : si l'image initiale n'est pas un carré, nous préférerons laisser des bandes noires sur le côté plutôt que de la déformer. La figure 4.10 montre un exemple de redimensionnement d'images.

Aussi, puisque la profondeur est encodée sur un canal de 16 bits et la couleur sur trois canaux

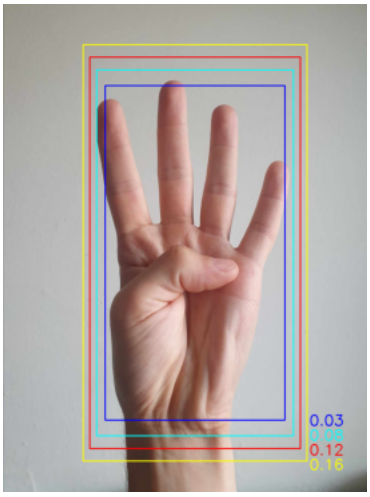

FIGURE 4.9 Exemple de plusieurs boîtes englobantes calculées avec divers coefficients à partir de l'estimation de pose de Mediapipe.

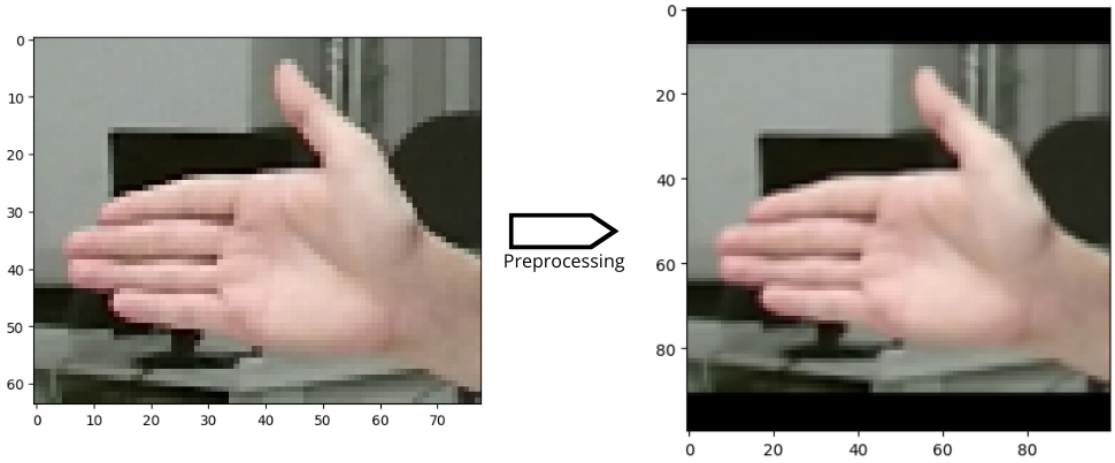

FIGURE 4.10 Exemple de redimensionnement d'images effectué permettant au réseau d'avoir des images de même format en entrée.

de 8 bits chacun, nous redistribuons les valeurs de l'image de profondeur et de l'image de couleurs entre 0 et 1. Alors seulement, nous concatènons les deux images pour avoir une image à quatre canaux.

Lorsque le prétraitement est terminé, l'image peut être envoyée dans le réseau de neurones. Ce réseau est un réseau de convolution largement inspiré de l'architecture ResNet [48]. Un réseau ayant une architecture ResNet est appelé un réseau résiduel. Ceci signifie qu'à certaines étapes du réseau choisies, nous sauvegardons les images caractéristiques (ou feature maps) du réseau. Après quelques convolutions, nous additionnons le résultat de ces convolutions avec les images caractéristiques sauvegardées. Nous appellons bloc résiduel, un bloc du réseau dans lequel nous sauvegardons les images caractéristiques au début et nous les additionnons au résultat des convolutions successives à la fin. Un exemple de bloc résiduel est montré en figure 4.11.

Le modèle utilisé pour la reconnaissance du geste de la main peut être séparé en deux parties distinctes. La première partie du réseau est un réseau de convolution résiduel qui réduit la taille des images pour en retenir les informations principales. La seconde est un réseau entièrement connecté qui classifie les images.

Le réseau résiduel contient six blocs résiduels qu'on note BR. L'architecture précise des blocs est illustrée à la figure 4.12. Le classificateur contient 3 couches linéaires, l'architecture du réseau en entier est montrée en figure 4.13. En sortie de ce réseau, nous obtenons le vecteur de probabilités $[p_1, p_2, \ldots, p_n]$ avec n le nombre de gestes.

Association des différentes prédictions

À ce stade, nous avons autant de vecteurs de probabilités que de caméras qu'on considère comme bien placées à chaque instant. Il reste maintenant à associer ces différentes prédictions

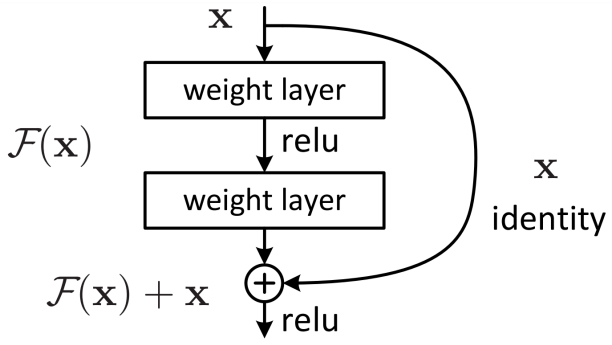

FIGURE 4.11 Exemple de bloc résiduel ©2016 IEEE.

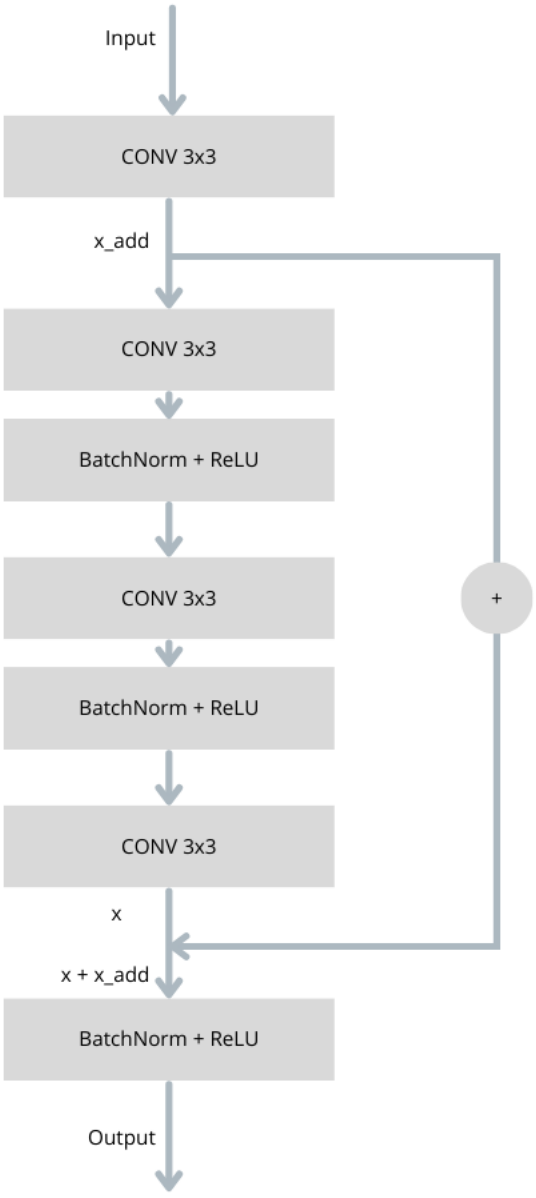

FIGURE 4.12 Architecture d'un bloc résiduel dans notre modèle

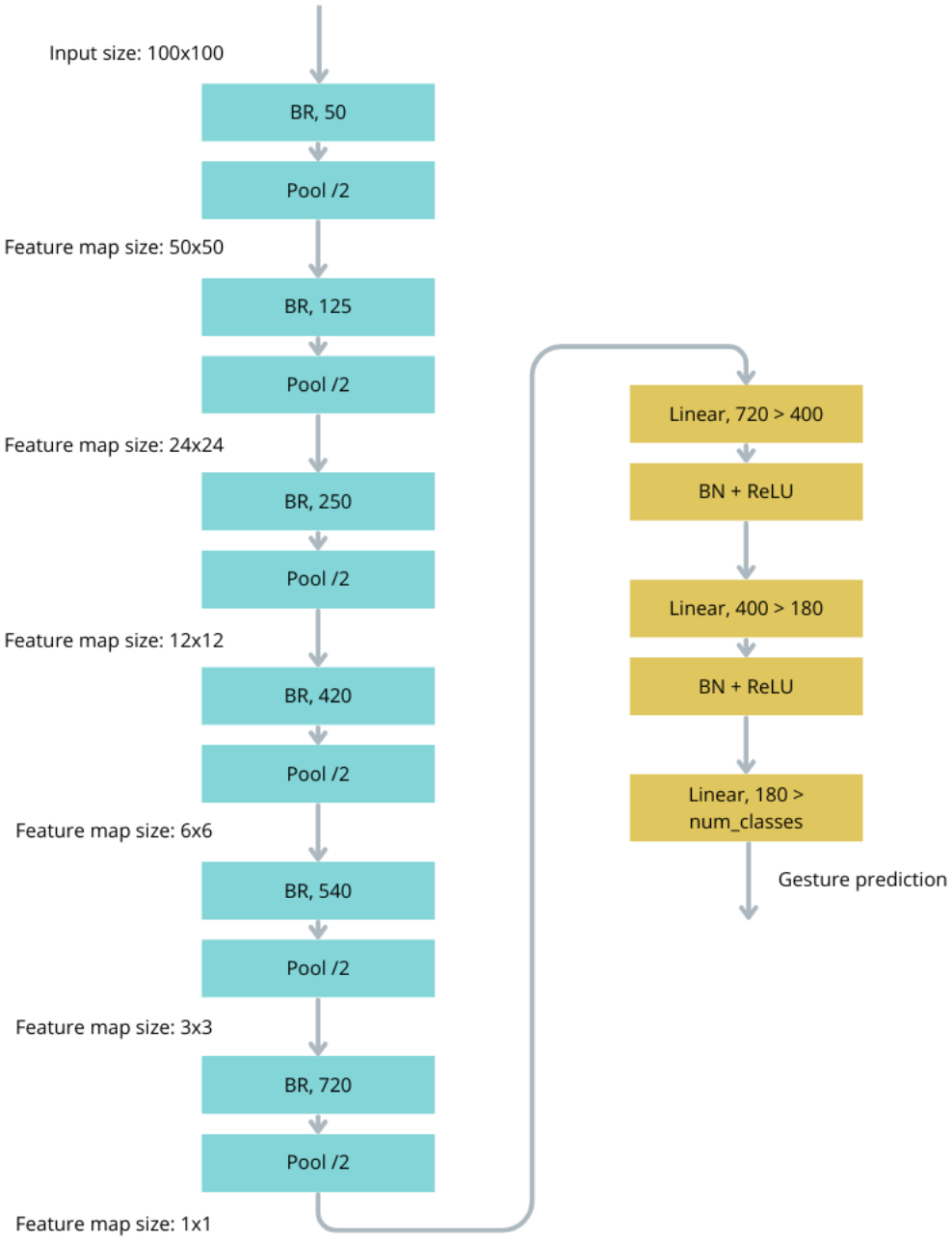

FIGURE 4.13 Architecture du réseau de classification au complet : en bleu la partie convolutive du réseau et en jaune la partie entièrement connectée. "BR, 100" signifie qu'il y a 100 feature maps pour ce bloc résiduel.

pour déterminer le geste qui est réalisé.

La méthode utilisée est de moyenner les différents vecteurs de probabilités. Au final, le geste ayant la plus grande probabilité après moyenne est le geste qu'on estime être réalisé. Un des avantages de cette méthode est qu'elle peut se généraliser facilement : peu importe la position des caméras ou leur nombre, faire la moyenne des probabilités devrait toujours avoir le même impact sur les résultats obtenus.

Cependant il est possible qu'il y ait des erreurs dues aux occlusions visuelles, comme par exemple le bras du robot qui serait dans le champ de vue d'une des caméras. Il est alors important de minimiser l'impact de ces erreurs. C'est pourquoi il est nécessaire de mettre en place un seuil de probabilité : si le geste ayant une probabilité maximale est inférieur à ce seuil, alors nous considérerons qu'aucun geste n'est effectué. Les gestes étant statiques, nous possèdons plusieurs frames successives pour un même geste que l'on peut utiliser à des fins de vérification. Nous proposons d'utiliser la méthode de l'article MEGURU [42] : lorsque le réseau prédit le même geste pendant x frames successives, nous considérerons que l'opérateur a bien réalisé le geste. Le seuil et le nombre x étant à déterminer pour maximiser la robustesse du réseau sans pour autant rendre la prédiction de geste impossible.

4.2 Création d'une nouvelle base de données

L'entraînement du réseau développé est supervisé. C'est pourquoi il est nécessaire d'avoir une base de données conséquente permettant d'entraîner le réseau. Comme discuté précédemment, les bases de données déjà existantes permettant d'entraîner ce type de réseau pour la reconnaissance de la gestuelle ne sont pas assez grandes et diverses. De plus, elles sont toujours faites dans des conditions idéales qui ne sont pas représentatives du contexte à l'étude. Cette section traite de la création d'une nouvelle base de données pour l'entraînement du réseau. Elle explique où et comment les données sont recueillies.

Comme la base de données sera également utilisée pour la sélection optimale d'un sous ensemble de caméras en fonction de la localisation de l'opérateur dans la pièce, il était important d'utiliser les six caméras simultanément.

Pour augmenter la capacité de généralisation du modèle, il est important de considérer une bonne variabilité dans l'ensemble d'entraînement. Cette variabilité peut s'exprimer en termes de configuration de la caméra (angle de vue et distance à l'opérateur), de caractéristiques humaines (morphologie, sexe, couleur de peau, préférence manuelle) et d'environnement (localisation de l'opérateur dans la cellule, présence d'occlusions, présence de plus d'un humain).

La variabilité de points de vue est rendue possible par le fait même d'utiliser les 6 caméras

simultanément dans la collecte. En terme de caractéristiques humaines, nous avons établi qu'un total de 20 participants serait suffisant pour ce travail, en comparaison aux bases de données publiques similaires [44, 46, 47].

La collecte de données est réalisée au Centre de Technologies de Fabrication en Aérospatiale (CTFA) du Centre National de Recherche du Canada (CNRC) : 2107 chemin de Polytechnique, Montréal, dans la cellule cobotique présentée à la Figure 1.1. Elle a eu lieu dans la semaine du 8 au 10 Août 2023.

Chaque participant doit faire les 10 gestes de la base de données, certains gestes seront réalisés deux fois : les gestes à une main seront fait avec la main droite et la main gauche. Le geste `cancel` sera également réalisé deux fois puisqu'il est le seul geste à deux mains à être véritablement asymétrique. Au total, il y a donc 17 séquences d'enregistrement d'une durée de une minute chacune. Pour chaque séquence, le participant se met dans une unique zone prévue à l'avance. Le fait qu'il ne change pas de zone permet de connaître les caméras bien placées automatiquement. Il n'y a pas besoin d'annoter par la suite les données pour connaître les caméras bien placées. Les participants font tous au minimum 4 gestes par zone et ne font pas les mêmes gestes dans chaque zone. Ainsi, les gestes (resp. participants) sont équitablement réalisés (resp. présents) dans chaque zone.

La durée d'une minute pour chaque séquence permet au participant de se déplacer dans la zone, ce qui permet d'augmenter la diversité des angles de vue et des occlusions. Chaque jour de la collecte de données, le robot est déplacé à une nouvelle position pour augmenter encore la variabilité (pour des raisons de sécurité, il n'est pas possible de déplacer le robot lorsque les participants se trouvent dans la cellule cobotique).

Pour la collecte de données, un script a été écrit en c++. Il permet d'enregistrer les images et d'annoter les gestes et les zones de manière automatique. Le frame rate maximal qui a été choisi est de 30 FPS mais en pratique, ce frame rate n'est jamais atteint car le fonctionnement de 6 caméras en simultané et la vitesse d'écriture de 6 images RGB-D sur le disque dur l'empêchent. La résolution des images est de 1280px720p.

Les caméras fonctionnent à un débit maximal lors de la collecte et ne sont pas synchrones. Cela implique d'enregistrer les `timestamp` de chaque image enregistrée pour pouvoir regrouper les images synchrones *a posteriori* en déterminant une durée Δt maximale d'écart entre chaque `timestamp` des images du groupe. Une fois que les données sont synchrones, nous vérifierons manuellement que, sur les images des caméras bien placées, nous puissions voir une main faisant le geste au moins sur l'une d'entre elles (autrement aucun apprentissage n'est possible à partir de ces images). Au final, nous retiendrons 100 groupes d'images synchrones par séquence. Au total, il devrait donc y avoir 1700 groupes d'images par participant, un groupe

d'images comprenant deux ou trois images dépendement de la zone dans laquelle le geste a été effectué.

Les participants sont invités parmi les employés du CTFA, les membres du groupe de recherche et des personnes proches de ces membres. Un formulaire d'information et de consentement a été proposé aux participants (disponible en annexe A). Nous précisons que la création de cette base de données a été approuvée par le comité d'éthique de la recherche à Polytechnique Montréal et par celui du Conseil National de Recherche du Canada.

CHAPITRE 5 EXPÉRIMENTATIONS

Ce chapitre décrit les différentes expériences qui ont été réalisées pour valider ou invalider les hypothèses de recherche. Aussi, ce chapitre présente les méthodes et métriques utilisées pour évaluer les modèles proposés. Enfin, les expériences qui ont été réalisées dans le but d'optimiser le modèle ou de tester certaines de ses limites seront présentées.

5.1 Évaluation et validation des modèles

Pour tester et évaluer un modèle entraîné de manière supervisée sur une base de données, il n'est pas souhaitable de le tester sur des données d'un participant qui est utilisé lors de l'entraînement. C'est pourquoi nous utilisons 80% des participants de chaque base de données pour l'entraînement et 20% pour le test et la validation. Pour chaque base de données, nous réalisons 5 entraînements différents de sorte à utiliser 100% des données comme sujet de test. Cela permet de faire une validation croisée des résultats.

Détection de la main

Le premier moyen pour évaluer la détection de la main est de compter le nombre d'images dans la base de données pour lesquelles le réseau a trouvé une main dans l'image. Cela donne un premier aperçu de la qualité du modèle.

Pour évaluer la détection de la main, nous devons aussi évaluer la précision des rectangles englobants. La métrique la plus utilisée dans ce cas est l'intersection sur l'union (IoU pour Intersection over Union). La figure 5.1 représente ce qu'est l'IoU. Le résultat est logiquement compris entre 0 et 1. Plus sa valeur est proche de 1, plus la précision de la détection est bonne.

Dans notre cas, nous ferons deux calculs d'IoU : un premier pour lequel nous ne prenons en compte que les images où une main a été trouvée dans l'image par le réseau, et un second où l'on donne une valeur de 0 pour les images où aucune main n'a été trouvée. Le premier résultat permet d'avoir une estimation de la précision des boîtes englobantes lorsqu'elles existent, le second permet une évaluation plus globale.

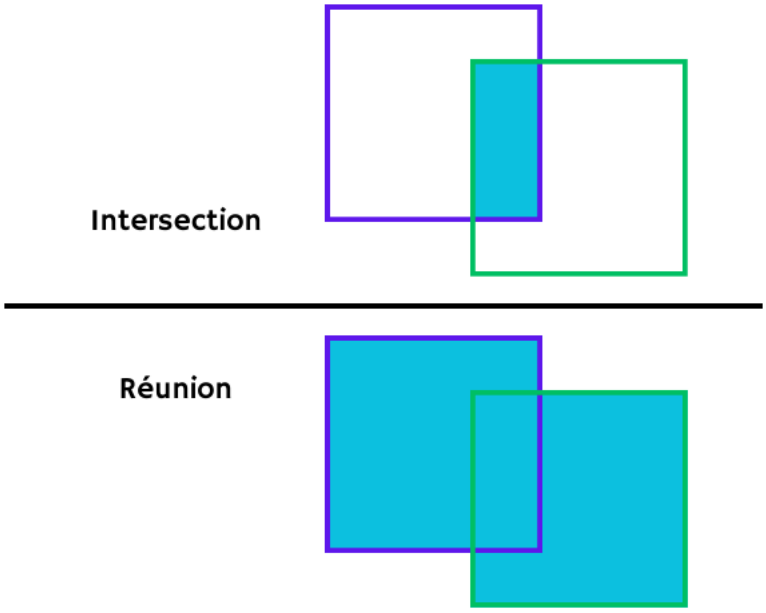

FIGURE 5.1 Schéma représentant l'IoU.

Reconnaissance du geste

Pour évaluer seulement la reconnaissance du geste sans prendre en compte la détection de la main, nous utilisons les rectangles englobants provenant des annotations de référence de la base de données. A partir de ces annotations, nous découpons la main dans l'image, nous appliquons les méthodes de prétraitement et nous prédirons le geste avec le modèle entraîné. Nous quantifierons ensuite la précision avec un pourcentage de gestes qui ont été correctement prédits.

Évaluation du pipeline complet

L'évaluation du pipeline complet proposé est la même que pour la reconnaissance de la gestuelle, à la différence près que l'on utilise le réseau de détection de la main pour obtenir les boîtes englobantes et non les annotations de référence des bases de données. Ainsi, nous calculerons également la précision de classification uniquement dans le cas où une main a été détectée.

L'intérêt des résultats qui ne prennent pas en compte le cas où les mains ne sont pas détectées est que dans la pratique, puisque nous considérons qu'un geste est réalisé seulement dans le cas où le modèle le prédit de manière successive, nous pouvons ignorer les frames où aucune main n'est détectée. Par exemple : imaginons une suite de 10 images dans lesquelles le geste

`start` est réalisé. Notons,

$$[\texttt{start}, \texttt{start}, \texttt{start}, \texttt{start}, \texttt{start}, \texttt{start}, \texttt{start}, \texttt{start}, \texttt{start}, \texttt{start}]$$

cette séquence de 10 images. La prédiction du modèle pourrait alors être :

$$[\texttt{cancel}, \texttt{start}, \texttt{start}, \texttt{NaN}, \texttt{start}, \texttt{start}, \texttt{start}, \texttt{start}, \texttt{NaN}, \texttt{start}]$$

où `NaN` correspond au cas où aucune main n'est détectée. Alors dans ce cas, nous proposons de considérer que le geste `start` a été réalisé 7 fois consécutivement malgré la présence des `NaN`.

5.2 Expérimentations avec la base de données HANDS

Tout d'abord, comme nous l'avons expliqué à la section 4.1.3, il faut ajuster les boîtes englobantes à partir de l'estimation de la pose de la main pour pouvoir utiliser correctement Mediapipe. Cela se fait par la recherche d'un coefficient optimal avec lequel nous allons agrandir les boîtes englobantes initiales obtenues avec Mediapipe. Pour cela, nous avons testé différentes valeurs de coefficient et comparé les boîtes englobantes obtenues et les annotations avec un calcul d'IoU sur l'intégralité de la base de données HANDS. Il suffit ensuite de prendre le coefficient maximisant les résultats.

Ensuite, le pipeline complet a été testé sur l'intégralité de la base de données en suivant les méthodes d'évaluation et de validation décrites dans la section précédente. Nous rappellons que Mediapipe a été utilisé pour la détection de main et notre modèle personnalisé, pour la reconnaissance de gestes.

En outre, deux autres expériences ont été réalisées sur la base de données HANDS pour évaluer :

— l'influence de la profondeur sur la performance de la reconnaissance de la gestuelle,
— l'impact de la diminution de la résolution des images en entrée sur la performance du modèle.

5.2.1 Influence de la profondeur sur les résultats

Cette expérience consiste à entraîner et tester le réseau en retirant le canal de profondeur. Ensuite, il faut comparer les résultats de ce modèle avec celui utilisant la profondeur. Aussi, nous ferons le test suivant : nous fournirons une image sur laquelle le canal de profondeur a

été mis à 0 en entrée du réseau entraîné avec la profondeur. Ceci nous permettra de vérifier l'utilisation de ce canal par le réseau.

On précise que cette expérience ne peut être réalisée que pour la classification de geste et non la détection puisque Mediapipe n'est pas conçu pour utiliser le canal de profondeur.

5.2.2 Impact de la résolution sur les résultats

Pour cette expérience, nous avons testé un même réseau sur différentes versions du jeu de test, simulant des résolutions spatiales différentes, variant entre 0.1 et 1. Sachant que, par exemple, une résolution de 0.2 correspond à une diminution du nombre de pixels par un facteur de 5. Nous noterons que la méthode pour diminuer la résolution est de prendre l'image de base et de réduire son nombre de pixels en utilisant la méthode d'interpolation d'Open CV : `INTER_AREA`.

Nous avons fait cette expérience cinq fois avec les modèles entraînés sur les différents sujets pour pouvoir également faire ces tests avec tous les sujets.

Cette expérience a comme intérêt d'observer la robustesse du réseau sur les changements de résolution. En fonction des résultats de l'expérience, nous pouvons ajouter ou non de la modification de résolution dans l'augmentation de données afin d'optimiser le réseau.

On notera que cette expérience n'a été réalisée que pour le modèle de classification des gestes et non pour la détection de main.

5.3 Expérimentations avec la nouvelle base de données

5.3.1 Détection et reconnaissance

Après avoir présenté les caractéristiques de la nouvelle base de données nommée COBOTIC, nous testerons le modèle dessus. Nous testerons, comme pour la base de données HANDS, à la fois la partie de détection de la main et la partie de classification.

Le modèle de classification est entraîné à partir des annotations de détection de la base de données. Pour la phase d'entraînement, nous ne prenons ni en compte la caméra d'où provient l'image, ni le fait que plusieurs images de plusieurs caméras ont été prise au même instant. Autrement dit, nous prenons l'intégralité des images de la base de données, nous les mélangeons sans retenir la position de l'opérateur, ni la caméra d'origine, ni l'instant auquel l'image a été prise. Nous retenons seulement la position de la main dans l'image et le geste effectué. Ensuite, nous entraînons le réseau sur ces données. Par conséquent, un même modèle avec les mêmes poids est utilisé pour chaque caméra.

Pour la détection de la main, nous nous contenterons d'évaluer la performance de Mediapipe, nous pourrons ensuite comparer les résultats obtenus sur cette base de données par rapport à ceux obtenus sur la base HANDS.

En outre, nous proposerons de tracer les courbes de calibration des modèles entraînés sur les jeux de tests différents. Les courbes de calibration permettent de vérifier que les différentes probabilités données pour la prédiction de gestes sont fiables. En fait, cela revient à vérifier que, lorsque le réseau donne une probabilité de 10% pour un geste, il s'agisse bien de ce geste dans 10% des cas. Il est intéressant de vérifier la calibration du modèle puisque nous allons associer plusieurs prédictions de plusieurs caméras. Concrètement, nous nous demandons s'il n'est pas plus rentable de sacrifier un peu la précision du modèle monoculaire dans le but d'améliorer sa calibration et d'améliorer la précision dans une configuration multi-caméra.

5.3.2 Croisement modèles/tests

Cette expérience consiste à tester le modèle entraîné sur la base de données HANDS sur la base de données COBOTIC et inversement. L'objectif de cette expérience est de montrer que la nouvelle base de données a une diversité suffisante qui permet aux modèles d'obtenir également de très bons résultats sur la base de données HANDS. A l'inverse, cela devrait aussi permettre de montrer que le modèle entraîné sur la base de données HANDS offrant peu de variabilité, a peu de robustesse par rapport aux changements de point de vue, à la présence d'occlusions visuelles et à la distance entre l'opérateur et la caméra.

5.3.3 Comparaison de la prédiction multi-caméra face à la prédiction à partir d'une unique caméra

Une fois que le réseau est entraîné et testé dans un contexte monoculaire sur la base COBOTIC, il s'agira de savoir si la précision du modèle permet d'améliorer la précision lorsque nous prenons en considération plusieurs images synchronisées. Cette fois, nous allons tester le modèle sur des groupes d'images synchronisées. Nous allons assembler les prédictions pour ce groupe d'images afin d'obtenir une prédiction d'un geste pour ce groupe d'images. La précision pour cette expérience sera comparée à celle obtenue dans l'expérience décrite dans la section 5.3.1 dans le but de savoir si le contexte multi-caméra permet bien d'améliorer les résultats de reconnaissance de la gestuelle.

CHAPITRE 6 RÉSULTATS ET DISCUSSION

Ce chapitre présente les résultats pour chaque expérience décrite dans le chapitre précédent. Ces résultats comprennent notamment les courbes d'entraînement, les métriques d'évaluation ainsi que des résultats qualitatifs. Chaque résultat présenté est ensuite analysé et discuté afin de mettre en évidence les particularités des modèles et données utilisés ainsi que de permettre leur comparaison.

6.1 Résultats sur la base de données HANDS

Cette section a pour objectif de présenter et discuter les résultats des différents éléments du pipeline ainsi que du pipeline complet sur la base de données HANDS. Une synthèse des résultats est disponible au tableau 6.1. Ces résultats sont analysés dans les sous-sections qui suivent. La lecture de tableau est décrite ci-après :

- Les colonnes du tableau correspondent à des résultats sur différents jeux de test. Pour rappel, la base de données comprend 5 sujets différents, chacun a été utilisé comme sujet de test, ce qui veut dire que pour la colonne "Sujet 1", l'entraînement du réseau a été fait sur les sujets 2 à 5 et les tests sur le sujet 1. La dernière colonne est la moyenne des 5 colonnes qui précèdent.
- La ligne "pourcentage de mains détectées" correspond au nombre de mains détectées par Mediapipe sur le nombre de mains dans les annotations. Nous précisons que même si nous obtenons un IoU égale à 0, nous considérons qu'une main a été détectée. Nous comptons en fait le nombre de boîtes englobantes données par Mediapipe sur le nombre total de mains.
- "L'IoU totale" est la valeur de l'IoU sur toutes les données. Autrement dit, une main non détectée compte pour 0 dans le calcul de l'IoU totale contrairement à la ligne suivante "IoU seulement pour les mains détectées" : cette fois les mains non détectées par Mediapipe ne sont pas prises en compte dans le calcul de l'IoU.
- La "précision pour IoU à 100%" correspond à la précision de la seconde partie du modèle (la classification des gestes) lorsque nous calculons la précision à partir des boîtes englobantes fournies en annotation. En d'autres termes, cette ligne correspond à la précision de la classification lorsque la détection de la main est parfaite.
- La "précision totale" correspond à la précision du pipeline au complet (détection + classification)
- La "précision seulement pour les mains détectées" correspond au même résultat qu'à

la ligne précédente mais en ne prenant pas en considération les cas où aucune main n'est détectée.

6.1.1 Coefficient optimal pour les rectangles englobants

Nous rappelons que les boîtes englobantes obtenues par Mediapipe ne sont pas immédiatement optimales, il faut les agrandir d'un coefficient à rechercher pour maximiser le résultat de la détection de la main en terme d'IoU.

Nous avons testé les coefficients suivants sur l'ensemble de la base de données HANDS : $[0.05, 0.1, 0.15, 0.18, 0.19, 0.2, 0.21, 0.22, 0.25]$. Les résultats sont affichés dans le tableau 6.2. Le coefficient pour lequel l'IoU est maximale est 0.2. C'est la valeur que nous utiliserons dans toutes les expériences du mémoire.

6.1.2 Détection de la main

Dans cette partie, nous nous intéressons uniquement aux résultats concernant la détection de la main par MediaPipe dans l'image. C'est à dire les trois premières lignes du tableau de synthèse 6.1.

Tout d'abord, concernant le nombre de mains détectées, nous observons que 81% des mains sont détectées par Mediapipe en moyenne. Cependant, ce résultat est fortement diminué par le sujet 3 sur lequel seulement 30% des mains sont détectées. Nous reviendrons dans la sous-section 6.1.4 sur le sujet 3 et les raisons pour lesquelles ces résultats peuvent être moins bons.

Même en l'absence d'occlusion visuelle et quand la main est parfaitement visible, il est possible que le modèle ne parvienne pas à détecter la main comme le montre la figure 6.1. Cela peut s'expliquer par l'utilisation d'un seuil trop élevé sur la confiance du modèle dans la détection. Cependant, comme nous l'avons déjà expliqué dans la section 5.1, il est important d'avoir un seuil de confiance élevé car le fait de ne pas détecter certaines mains n'est pas un véritable problème dans notre cas d'application (tant que les manquements ne sont pas trop nombreux).

Les résultats de l'IoU totale sont fortement corrélés (R=0.99) au pourcentage de mains détectées. Cela s'explique d'une part par le fait que, par définition, l'IoU totale est nulle lorsqu'une main n'est pas détectée ; et d'autre part par le fait que pour chaque sujet, l'IoU seulement pour les mains détectées est toujours proche de la moyenne globale qui est de 80.5%. Ceci démontre qu'une fois la main détectée, l'algorithme de localisation de la main (création du rectangle englobant) est très stable d'un sujet de test à l'autre.

TABLEAU 6.1 Synthèse des résultats (en %) obtenus par entraînement et test sur la base de données HANDS

	Sujets de test					**Moyenne**
	Sujet 1	Sujet 2	Sujet 3	Sujet 4	Sujet 5	
Pourcentage de mains détectées	95.1	80.4	30.0	99.4	100.0	81.0
IoU totale	77.4	64.2	25.0	78.3	79.5	64.9
IoU seulement pour les mains détectées	81.3	79.8	83.3	78.7	79.5	80.5
Précision pour IoU à 100%	93.5	97.5	86.9	92.7	97.8	93.7
Précision totale	88.8	70.2	28.6	97.6	93.2	75.7
Précision seulement pour les mains détectées	93.3	87.3	95.5	98.2	93.2	94.0

Coefficient	0.05	0.10	0.15	0.18	0.19	0.20	0.21	0.22	0.25
IoU	46.1	54.9	62.0	64.3	64.6	64.9	64.8	64.8	63.7

TABLEAU 6.2 IoU (en %) sur la base de données HANDS en fonction des différentes coefficients.

FIGURE 6.1 Exemple de mauvaise détection par Mediapipe avec une main non détectée.

Des résultats qualitatifs 6.2 permettent d'illustrer des cas d'échec de détection de la main par Mediapipe. Sur cet exemple, l'estimation de pose de la main par Mediapipe n'est pas complète : il manque un doigt. Ceci entraîne donc une faible valeur d'IoU associée à l'image puisque celle-ci est égale à 47.4%.

6.1.3 Reconnaissance de la gestuelle

Pour évaluer la classification de geste uniquement, c'est à dire sans la partie détection de la main, nous nous intéressons à la quatrième ligne du tableau de synthèse (6.1), intitulée "Précision pour IoU à 100%", nous ne regardons que la précision lorsque la détection est parfaite, c'est à dire quand les rectangles englobants de référence, fournis par la base de données, sont utilisés pour limiter la région d'intérêt.

Le modèle de classification a de très bon résultats puisque la moyenne est de 93.7%, nous remarquons que, comme pour la détection de main, le modèle a plus de difficulté avec le sujet 3 qui engendre une diminution de la moyenne globale. Nous reviendrons un peu plus loin sur les possibles raisons pour lesquelles les résultats sur le sujet 3 sont moins bons.

Influence de la profondeur

Les résultats pour la précision des réseaux de classification entraîné avec et sans profondeur sont présentés dans le tableau 6.3.

Ces résultats mettent en évidence le fait que le réseau ne parvient pas à utiliser l'information de profondeur pour améliorer les résultats. La différence de résultat entre les deux modèles n'est en revanche pas assez significative pour dire que la profondeur a un effet négatif sur la précision.

FIGURE 6.2 Exemple d'une détection incomplète par Mediapipe avec un doigt coupé. L'IoU pour cette détection de main est de 47.4%.

Tableau 6.3 Précision obtenue sur la base de données HANDS pour un même réseau de classification entraîné avec et sans images de profondeur (en pourcentage).

	Sujets de test					**Moyenne**
	Sujet 1	Sujet 2	Sujet 3	Sujet 4	Sujet 5	
Modèle sans profondeur	93.5	97.5	86.9	92.7	97.8	93.7
Modèle avec profondeur	92.4	98.3	83.7	94.1	98.2	93.3

Cependant, ajouter un canal à analyser augmente nécessairement le temps de calcul, même si cela est très léger, il ne sert à rien de garder le canal de profondeur des images puisqu'elles n'ont aucun intérêt. C'est la raison pour laquelle, nous pensons qu'il est préférable de ne pas utiliser le canal de profondeur pour classifier les gestes avec notre modèle.

Pour appuyer ce résultat, la précision du réseau entraîné avec des images de profondeur sur des images sans profondeur sont mis dans le tableau 6.4. Ces résultats ont une tendance à confirmer également la non utilisation du canal de profondeur pour la classification des gestes, hormis pour les tests sur le sujet 5 où la précision est inférieure de près de 3% lorsqu'on retire la profondeur.

Pour interpréter ce résultat, nous revenons à la configuration géométrique de la scène dans HANDS. Rappelons que dans cette base de données, les sujets font face à la caméra et les gestes de la main le sont également. La profondeur sur la main est donc quasiment constante ce qui pourrait expliquer qu'elle n'amène aucune information discriminante pour le modèle. Pour illustrer ce propos, un exemple d'image de profondeur en entrée du réseau est montré sur la figure 6.3.

Tableau 6.4 Précision (en %) obtenue sur la base de données HANDS pour un réseau de classification entraîné avec profondeur sur des images de test avec et sans profondeur.

	Sujets de test					**Moyenne**
	Sujet 1	Sujet 2	Sujet 3	Sujet 4	Sujet 5	
Jeu de test avec profondeur	92.4	98.3	83.7	94.1	98.2	93.3
Jeu de test sans profondeur	92.5	98.5	82.2	94.2	95.5	92.6

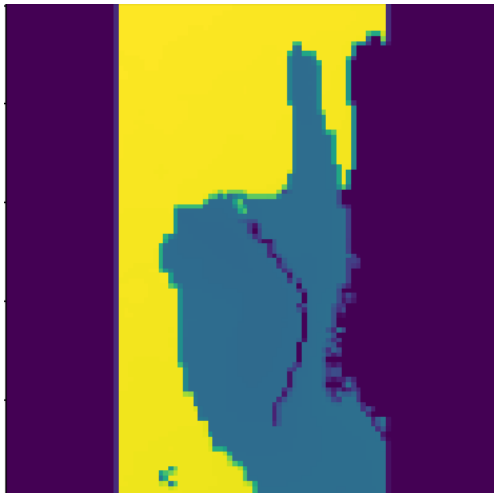

FIGURE 6.3 Exemple d'image de profondeur en entrée du réseau

Robustesse à la résolution spatiale

Nous avons tracé les courbes montrant l'évolution de la précision de la classification en fonction de la résolution des images dans les jeux de test. Les courbes pour chaque jeu de test sont visibles sur la figure 6.4 ainsi que la courbe moyenne. En abscisse, nous rapportons le facteur d'échelle utilisé pour réduire la résolution originale des rectangles englobants.

On observe que le réseau de classification, entraîné sans augmentation de données par mise à l'échelle, est robuste à une réduction de la résolution spatiale jusqu'à 20% à 30% de la résolution d'origine. Cela peut donc signifier que la distance à laquelle se trouve le sujet pourrait ne pas avoir d'impact sur la qualité de la classification.

Par ailleurs, nous remarquerons que le sujet 4 résiste plus que les autres. Cela est probablement dû au fait qu'il est la personne la plus proche de la caméra parmi les 5 sujets. Cela se voit sur la figure 6.5 où l'on peut estimer la distance à laquelle se situe chaque sujet. À l'inverse, le sujet 3 semble être la personne la plus éloignée. Ceci se traduit par une courbe qui commence à atteindre son plateau plus tard, à partir de 30% à 40% de facteur d'échelle. Enfin, il est assez étonnant que pour le sujet 5, la diminution de la résolution permette même dans une certaine mesure d'augmenter les résultats.

6.1.4 Pipeline complet

Pour évaluer le pipeline au complet, c'est à dire en juxtaposant le modèle de reconnaissance ou classification du geste au modèle de détection automatique de la main, nous nous intéressons maintenant aux 2 dernières lignes du tableau de synthèse (6.1).

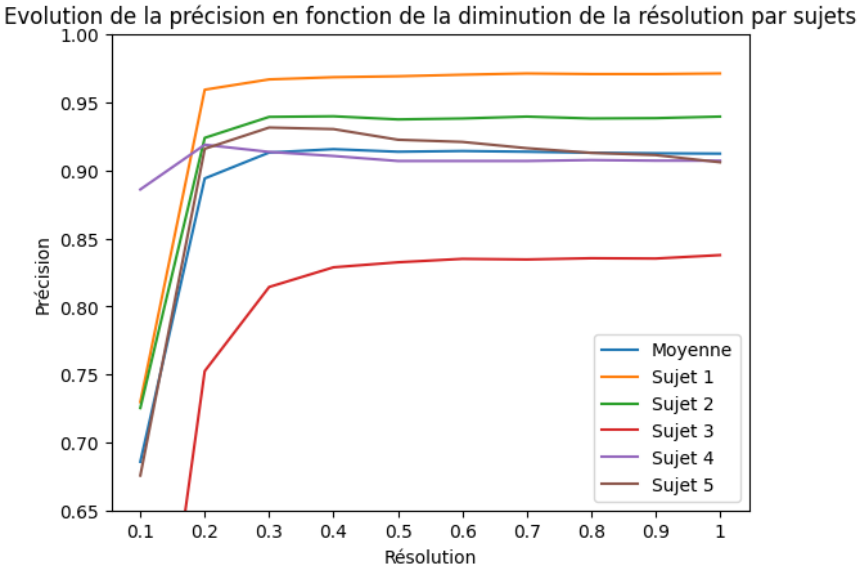

FIGURE 6.4 Évolution de la précision de classification par rapport à la résolution des données pour chaque sujet.

La précision totale moyenne est de 75.7%. En d'autres termes : notre pipeline est capable de prédire correctement le geste réalisé dans 3/4 des images de la base de données HANDS. Toutefois, si nous n'évaluons cette même métrique que sur les images dans lesquelles les mains sont détectées (peu importe l'IoU), alors la précision est de 94%. Ces résultats confirment que le maillon le plus faible du pipeline est bien le module de détection de la main.

Pour les sujets 3 et 4, la précision seulement pour les mains détectées (95.5% et 98.2% respectivement) est plus élevée que la précision dans le cas idéal, c'est à dire sans utiliser le modèle de détection mais les rectangles englobants fournis, (86.9% et 92.7% respectivement). Pourtant, l'IoU n'est pas particulièrement plus élevée que pour les autres sujets. Ces résultats pourraient s'expliquer par le fait que Mediapipe ne détecte pas certaines mains que le réseau de classification n'arrive de toute manière pas à classifier correctement, même lorsque nous lui donnons les boîtes englobantes des annotations.

Ensuite, nous pouvons remarquer que l'inverse se produit pour les sujets 2 et 5. Notamment pour le sujet 2 avec une précision dans le cas idéal qui est 10% plus élevée que la précision sur les mains détectées (97.5% contre 87.3%). Pourtant, l'IoU n'est pas inférieure comparée à celle des autres sujets. Cela voudrait dire que certaines mains sont mal détectées sans pour autant provoquer une diminution importante de l'IoU. Cela peut être dû au fait qu'un doigt est souvent coupé comme sur la figure 6.2.

Enfin, nous observons un résultat très surprenant pour le sujet 4. En effet, c'est le seul jeu

FIGURE 6.5 Image pour chaque sujet de la base de données HANDS permettant de visualiser la distance à laquelle se trouve les sujets.

de test pour lequel la précision totale obtenue par le pipeline complet proposé (97.6%) et sans éliminer les mains non détectées est supérieure à la précision à partir des rectangles englobants fournis par la base de données (92.7%). Pourtant le modèle de détection de la main ne semble pas avoir beaucoup mieux performer sur ce sujet que sur l'ensemble des sujets de test (IoU total de même ordre que pour les sujets 1 et 5). Il est difficile de trouver avec confiance une explication à ce résultat. Nous pourrions penser à une annotation manuelle des rectangles englobants, fournie par la base de données, de moindre qualité pour ce sujet.

Discussion autour du sujet 3

Comme nous le voyons depuis le début de cette section, le sujet 3 représente une véritable difficulté pour le modèle, aussi bien pour la détection de la main avec Mediapipe que pour le pipeline complet. En effet, comme le montre le tableau 6.5, si l'on retire le sujet 3, la précision du modèle est grandement améliorée.

Les seuls résultats améliorés lorsqu'on ajoute le sujet 3 sont celui sur la précision seulement pour les mains détectées (augmentation de 1%) et celui sur l'IoU seulement pour les mains détectés. Cela est dû au fait que Mediapipe a des difficultés à détecter une main sur les mêmes images que le réseau de classification ne parvient pas à classifier. Ce résultat est en fait biaisé par le fait que Mediapipe a trié les images en gardant les plus "faciles" car il n'a pas détecté de mains sur les images les plus "difficiles".

Par ailleurs, nous remarquons que la précision pour IoU à 100%, c'est à dire la précision de la classification quand la détection est idéale, n'est pas beaucoup plus faible. Si les deux parties du pipeline ont plus de mal avec le sujet 3, c'est principalement Mediapipe qui peine à obtenir de bons résultats.

En analysant les données manuellement, nous pouvons trouver une explication derrière ces baisses de résultats. En effet, comme nous l'avons vu précédemment, le sujet 3 est plus éloigné que les autres. Ceci peut expliquer que Mediapipe ne parvienne pas à détecter les mains correctement. Il y a aussi un autre facteur à prendre en compte : les sujets 1, 2, 4 et 5 sont quasi immobiles tout le long de l'enregistrement : seul les bras et les mains sont en mouvement. Alors que le sujet 3 est bien plus en mouvement (Figure 6.6). Cela augmente la variabilité des données et rend donc la tâche plus difficile. Lorsque nous entraînons un réseau avec les sujets 1, 2, 4 et 5, le réseau n'apprend pas ces variabilités. Ensuite, lorsque nous le testons sur le sujet 3 il peine donc davantage. Au final, cela remet en lumière l'insuffisance de la variété des données dans cette base de données, du moins pour les sujets 1, 2, 4 et 5.

TABLEAU 6.5 Comparaison des moyennes des résultats sur la base de données HANDS avec et sans le sujet 3 (en pourcentage).

	Moyenne avec sujet 3	**Moyenne sans sujet 3**
Pourcentage de mains détectées	81.0	93.7
IoU totale	64.9	75.0
IoU seulement pour les mains détectées	80.5	79.8
Précision pour IoU à 100%	93.7	95.4
Précision totale	75.7	87.5
Précision seulement pour les mains détectées	94.0	93.0

FIGURE 6.6 Quatre images illustrant les mouvements du sujet 3 de la base de données HANDS.

Comparaison avec les résultats de l'article MEGURU

Il serait intéressant de comparer nos résultats avec ceux obtenus dans l'article [42] qui utilise la base de données HANDS. Toutefois il est important de prendre en compte que la comparaison ne peut être directe. En effet, dans leur article, les auteurs ont eu accès aux données d'un 6ème sujet dont les données n'ont pas été publiées. De plus, les auteurs ont utilisé les sujets 1, 2 et 3 ainsi que le 6ème sujet pour leur jeu d'entraînement. Les sujets 4 et 5 sont donc réservés pour les tests. Par conséquent, ils n'ont pas fait de validation croisée de leurs résultats.

Pour comparer nos résultats, nous proposons donc de comparer leurs résultats de tests sur les sujets 4 et 5 avec la moyenne de nos résultats obtenus avec les deux réseaux ayant été testés sur ces mêmes sujets. Dans tous les cas, leur modèle comme les nôtres ont été entraînés sur un même nombre de sujets (4) et testés sur les mêmes également. La comparaison est donc pertinente.

En moyenne, sur les sujets 4 et 5, la précision totale que nous obtenons avec notre réseau est de 95,4% (97,6% pour le sujet 4 et 93,2% pour le sujet 5) tandis que dans l'article MEGURU, la précision sur ce jeu de test est de 90,0%.

6.2 Présentation de la base de données COBOTIC

Cette section a pour objectif de présenter la nouvelle base de données, intitulée COBOTIC, par des statistiques mais aussi les annotations qu'elle contient. Cette section donne également des exemples d'images qu'on peut y trouver. Nous y expliquons également comment ont été associé les groupes d'images.

Le consentement de tous les participants a été recueilli conformément aux directives éthiques. De plus, puisque les images de couleur permettent d'identifier les participants, un rond noir a été ajouté sur chaque visage dans les images qui sont publiées dans ce mémoire. En revanche, lors de la phase d'entraînement et de test, ces blocs ne sont pas présents. Enfin, nous préciserons que les images de profondeur ne sont jamais modifiées puisqu'elles ne permettent pas l'identification des participants.

6.2.1 Statistiques et exemples d'images

La base de données COBOTIC contient les images de 20 participants. Parmi ces 20 participants : 13 sont des hommes, 7 sont des femmes. Ce groupe de 20 personnes a une taille moyenne de 172,5 cm avec une certaine variabilité individuelle (écart-type de 9,31 cm), et un poids moyen de 74,6 kg avec une variabilité plus importante (écart-type de 16,3 kg) ce

qui témoigne d'une bonne variabilité de corpulence dans les données. Aussi, nous montrons dans la figure 6.7 un ensemble d'images qui montre comment le participant se déplace dans la cellule afin de couvrir la cellule cobotique dans son ensemble. Nous montrons aussi en figure 6.8 une image montrant un cas d'occlusions notamment par le robot. Enfin, nous montrons dans la figure 6.9 un exemple d'image de profondeur de la cellule cobotique avec son image de couleur associée.

Pour chaque personne, nous avons en moyenne 4250 images (4200 ou 4300 dépendamment des zones dans lesquelles le participant a fait les gestes). Au total, la base de données compte donc 85000 images RGB-D ou 34000 groupes d'images synchrones. Pour chaque groupe d'images, la main qui réalise le geste est visible sur au moins une des images du groupe (on rappelle qu'un groupe d'images est un ensemble d'images provenant des caméras bien placées et non de toutes les caméras). Cela permet de pouvoir entraîner et tester un modèle sur chaque groupe de la base de données.

6.2.2 Annotation des données

Toutes les images ont automatiquement été annotées en terme de geste effectué au moment même de l'acquisition. En revanche, la position de la main dans l'image, délimitée par des boîtes englobantes, a dû se faire manuellement. Ainsi, par contrainte de temps, seules les images de 10 des 20 participants ont pu être annotées pour la détection de la main. Finalement, l'objectif est d'annoter l'intégralité de la base de données mais pour ce mémoire, les résultats présentés autour de cette base de données ne sont calculés que sur la moitié de la base de données COBOTIC.

6.2.3 Synchronisation des caméras

La synchronisation des caméras a été effectuée *a posteriori*, moyennant les timestamps des images de la manière suivante : pour chaque geste de chaque participant, nous avons initialement en moyenne 180 images par caméra. Nous voulons synchroniser les images de sorte que nous ayons au moins 100 groupes d'images tout en minimisant le temps δt qui sépare les images. Pour cela, nous cherchons la valeur de δt de manière itérative. Au départ, nous fixons δt à 20ms et nous cherchons le nombre de groupe d'images synchrones qu'on peut obtenir avec ce δt. Si ce nombre est inférieur à 100, nous recommençons avec une valeur de δt plus grande jusqu'à un maximum de $\delta t = 150ms$. Ainsi, nous remarquerons que la valeur de δt varie par séquence d'acquisition.

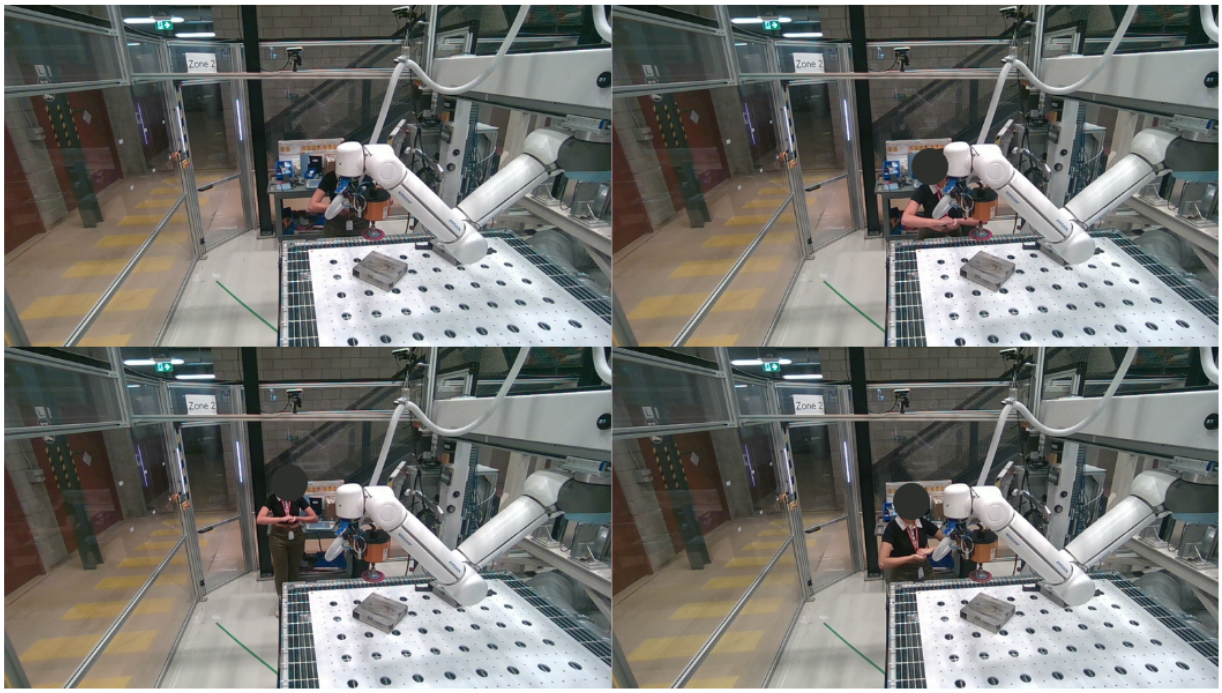

FIGURE 6.7 Exemple de déplacement d'un participant dans une zone

FIGURE 6.8 Exemple de cas d'occlusions, le geste effectué est un 4.

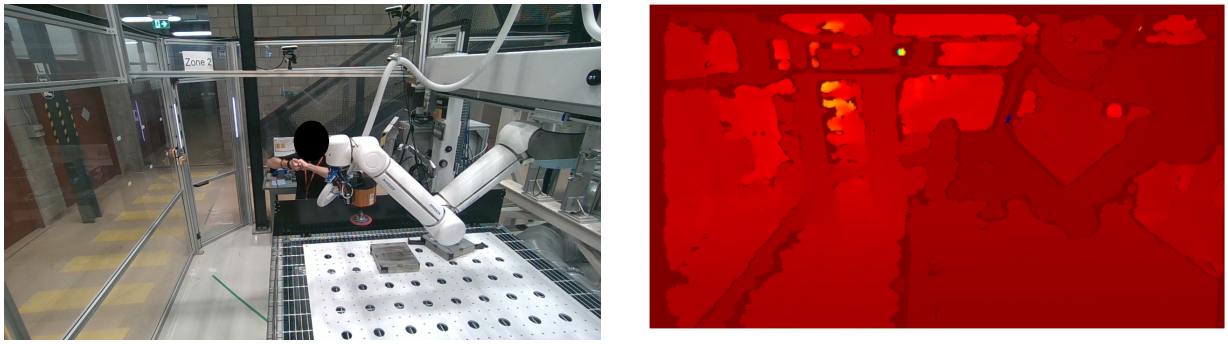

FIGURE 6.9 Exemple d'images de profondeur avec l'image de couleur associée.

6.3 Résultats sur la nouvelle base de données

Cette section traite des résultats du pipeline complet sur la base de données COBOTIC ainsi que des analyses et discussions autour de ces résultats. La première sous-section présente la performance de la détection de la main par Mediapipe. La seconde la performance pour la classification. Ensuite, les courbes de calibration des réseaux de classifications seront présentées. Enfin, nous reviendrons sur nos deux hypothèses de recherche, avec une section sur le croisement des jeux de données, puis une section sur la comparaison des performances de classification monoculaire et multi-caméra.

6.3.1 Résultats pour la détection de la main avec Mediapipe

La détection de la main avec Mediapipe fonctionnait relativement bien sur la base de données HANDS, avec un taux de détection de mains de 81% en moyenne sur les cinq sujets. Cependant, avec COBOTIC, la performance a drastiquement diminué : en effet, sur les 10 premiers sujets annotés, seulement 19,5% des mains visibles sont détectées par le réseau. Cette diminution de performance peut s'expliquer par deux facteurs :

1. La présence d'occlusions visuelles, inexistantes dans HANDS,
2. La diminution importante de la taille des mains (en pixels) dans les images, en fonction de la distance entre l'opérateur et la caméra (on notera que, dans la section 6.1.3, nous avons observé une robustesse à la résolution spatiale seulement pour le réseau de classification).

Initialement, Mediapipe a été entraîné pour obtenir l'estimation de pose de la main pour un utilisateur qui fait face à la caméra, ce qui implique d'avoir une résolution importante et une occlusion faible, voir inexistante. Malgré tout, une autre solution est envisagée pour détecter

la main et est présentée dans le chapitre 7.3.

6.3.2 Résultats pour la classification du geste

Contrairement à Mediapipe que nous ne pouvons pas entraîner à nouveau, le réseau de classification ou reconnaissance de geste proposé dans ce travail est ici ré-entraîné sur les données de COBOTIC. Comme discuté dans la section 6.1.3, nous avons documenté, par expérimentation, une robustesse du modèle de classification par rapport à une diminution de résolution spatiale. Nous nous attendons donc à obtenir pour ce modèle de meilleurs résultats que pour la détection de la main. Nous précisons qu'étant donné la faible qualité des résultats de détection, les résultats pour la classification ont été calculé à partir des boîtes englobantes obtenues manuellement (équivalent à ce qu'on définit par précision pour IoU à 100% dans le tableau 6.1).

Les résultats pour le modèle de classification entraîné et testé par validation croisée sur COBOTIC sont présentés dans le tableau 6.6. Comme pour la base de données HANDS, le tableau est séparé en plusieurs colonnes, chaque colonne correspondant à un jeu de test différent ou à la moyenne de tous les réseaux. La seule différence étant que les jeux de test correspondent aux images de deux participants mis de coté à chaque fois lors de l'entraînement. Ainsi les jeux de test sont composés des participants P1 et P2, puis P3 et P4, et ainsi de suite.

Concernant la classification des gestes, la diminution de la performance lorsque nous passons d'un modèle entraîné et testé sur HANDS (moyenne de 93.7%) au modèle entraîné et testé sur COBOTIC (moyenne de 86%) s'explique principalement avec l'occlusion. En effet, même si la main est détectée de manière optimale, il est difficile de dire quel geste est réalisé, même pour un humain lorsque l'entièreté de la main n'est pas visible. Un exemple de ces images est visible dans la figure 6.8. Par exemple, lorsque l'auriculaire et l'annulaire sont cachés par le robot, et que les autres doigts sont levés, il est impossible de savoir s'il s'agit d'un 3 ou d'un 5. Pour ce type d'images, nous espèrons avoir des vecteurs de probabilités avec des valeurs plus fortes pour le 3 et le 5 (ce qui devrait se traduire par une bonne calibration du réseau).

Tableau 6.6 Précision de la classification de geste (en %) obtenue par entraînement et test sur la base de données COBOTIC

	jeu de test					**Moyenne**
	P1-P2	P3-P4	P5-P6	P7-P8	P9-P10	
Précision pour IoU à 100%	88.7	86.3	84.9	84.8	85.6	86.0

Ainsi, lors de l'agrégation des données, les images provenant d'autres caméras permettront de trancher entre ces deux gestes.

Contrairement à la base de données HANDS, nous remarquons qu'il n'y a pas de jeu de test pour lequel la précision diminue fortement. Les performances des 5 modèles sont homogènes, avec un écart type de 1.6% seulement. Ainsi, la richesse des données de COBOTIC, en terme de variabilité, démontre une bonne robustesse inter-participants.

6.3.3 Courbe de calibration du modèle de classification

Les courbes de calibration ont été tracées sur la figure 6.10. Sur cette courbe nous pouvons voir la courbe pour chaque réseau entraîné sur les différents jeux de données ainsi qu'une courbe qui montre une calibration parfaite.

Plus les courbes de calibration sont proches de la courbe parfaite, plus la calibration est bonne. La courbe du réseau entraîné sur le troisième jeu de données (avec les images de tous les participants sauf P5 et P6) est la plus proche de la calibration parfaite. Alors que, la courbe du réseau entraîné sur le deuxième jeu de données (avec les images de tous les participants sauf P3 et P4) est la plus éloignée. Pour les cinq réseaux, la fréquence réelle observée est inférieure à la probabilité donnée par le réseau pour les probabilités prédites supérieures à 0,5. Cela signifie que le réseau a plutôt tendance à être trop confiant sur le geste qu'il prédit.

Il sera donc intéressant de voir si, dans une configuration multi-caméra, le réseau entraîné sur le troisième jeu de données (avec les images de tous les participants sauf P5 et P6) et testé sur P5-P6, voit sa performance augmenter plus que les autres. Cela pourrait s'expliquer par sa meilleure calibration.

6.3.4 Résultats pour le croisement des jeux de données

Cette section présente les résultats des réseaux testés sur la base de données sur laquelle ils n'ont pas été entraîné. Nous précisons que tous les tests ont été réalisés dans une configuration monoculaire même lorsque le réseau entraîné sur la base HANDS a été testé sur la base COBOTIC.

Le tableau 6.7 montre l'ensemble des résultats. Nous précisons aussi que les résultats dans ce tableau sont la moyenne de la précision sur les différents jeux de test des cinq réseaux entraînés.

Ces résultats montrent deux choses intéressantes :

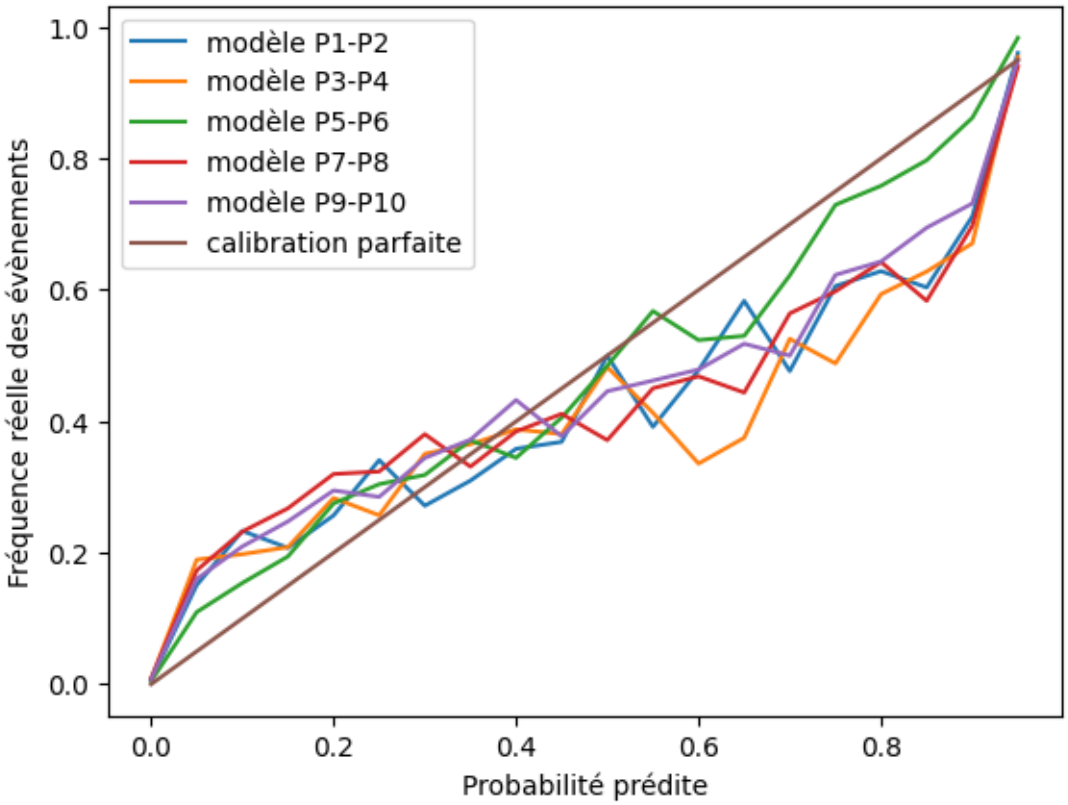

FIGURE 6.10 Courbes de calibrations pour chaque modèle de classification.

TABLEAU 6.7 Précision de classification moyenne (en %) obtenue pour le croisement des jeux d'entraînement et de test entre la base de données HANDS et COBOTIC.

	Entraînement sur HANDS	**Entraînement sur COBOTIC**
Test sur HANDS	93.7	93.4
Test sur COBOTIC	35.1	86.0

1. L'entraînement sur la base de données HANDS ne permet pas une bonne généralisation sur la base de données COBOTIC qui représente davantage la réalité pratique en conditions industrielles, avec un opérateur qui peut se déplacer dans la cellule.
2. À l'inverse, le réseau entraîné sur la base de données COBOTIC offre un meilleur pouvoir de généralisation à d'autres bases de données. D'autant plus, nos résultats indiquent que lorsqu'il est testé sur la base de données HANDS, il obtient une précision similaire (à moins de 1% de différence) à celle du modèle entraîné sur HANDS.

Ces résultats permettent d'une part, de confirmer notre première hypothèse de recherche, et d'autre part, de démontrer l'importance d'une base de données de qualité : riche en variabilité en terme de configuration de caméra (points de vues, distance à l'opérateur, entre autres), de caractéristiques humaines (corpulence, préférence manuelle, entre autres) et d'environnement (présence d'occlusions visuelles, entre autres).

6.3.5 Résultats de la classification multi-caméra par rapport à la classification monoculaire

Dans cette sous-section nous voulons maintenant comparer la performance du réseau de classification lorsqu'il prédit le geste à partir d'une vue monoculaire face à celle du même réseau lorsqu'on agrège les prédictions à partir de plusieurs vues synchronisées. Rappelons qu'ici nous n'exploitons que la base COBOTIC puisqu'acquise en mode multi-caméras. Le tableau 6.8 présente les résultats pour les deux réseaux et permet une comparaison quantitative.

Tout d'abord, nous pouvons noter que quelque soit le jeu de test, la précision est améliorée par le fait d'utiliser plusieurs caméras. En moyenne, la prédiction à partir de plusieurs caméras permet de gagner 5% en précision. Cela signifie que dans un contexte multi-caméra, les erreurs de classification que nous obtenions dans le mode monoculaire sont compensées par l'ajout d'autres caméras qui ont probablement un meilleur angle de vue (sans occlusion par exemple). Cela permet ainsi de confirmer notre deuxième hypothèse de recherche. La figure 6.11 illustre

Tableau 6.8 Précision de classification (en %) obtenue par entraînement et test sur la base de données COBOTIC en modes monoculaire et agrégation multi-caméra

	jeu de test					**Moyenne**
	P1-P2	P3-P4	P5-P6	P7-P8	P9-P10	
Précision pour le modèle monoculaire	88.7	86.3	84.9	84.8	85.6	86.0
Précision pour le modèle multi-caméra	91.7	92.6	88.7	89.6	92.4	91.0

un exemple d'un cas d'occlusion, réglé par l'agrégation des prédictions multi-caméras. Dans cet exemple, le geste de l'image de gauche est plus dur à prédire car la main passe derrière un tube du robot alors qu'au même instant, une autre caméra a pris une image parfaite du geste.

Pour un même modèle, entraîné sur le même jeu de données (COBOTIC), le mode multi-caméra permet même de s'approcher des résultats obtenus en mode monoculaire sur des images en contexte idéal (opérateur face à la caméra, absence d'occlusion visuelle), comme la base HANDS dont la précision était de 93.4% (6.7).

Par ailleurs, nous remarquons que le réseau entraîné sur le troisième jeu de données et testé sur P5 et P6 (qui, rappelons le, était le mieux calibré, voir section 6.3.3) ne voit pas sa précision augmenter plus fortement que les autres (augmentation de 3.8% alors que les autres gagnent en moyenne 5.2%). L'augmentation est même plus faible. *A priori*, sur la base de ces résultats préliminaires, nous pouvons indiquer qu'une meilleure calibration du réseau n'a pas un impact direct sur l'agrégation des données.

Nous noterons que sur certaines images de la base de données, la main qui fait le geste n'est pas visible. En mode de prédiction multi-caméra, nous obtenons tout de même une prédiction pour cette image. En effet, celle-ci fait partie d'un groupe d'images sur lequel nous pouvons voir la main sur au moins une image du groupe. A l'inverse, dans le cadre d'une prédiction monoculaire, cette image n'est pas prise en compte dans les pourcentages des tableaux puisque le rectangle englobant de la main n'est pas existant. Par conséquent, les pourcentages relevés dans le tableau 6.8 pour la précision monoculaire ne sont pas calculés sur toutes les images de test, contrairement à la précision multi-caméra. Cela signifie que l'utilisation de plusieurs caméras permet en réalité une augmentation de la performance encore plus intéressante que l'augmentation de 5% rapportée par nos expérimentations.

FIGURE 6.11 Exemple de cas où la configuration multi-caméra peut permettre de corriger des problèmes d'occlusion. Dans l'image de gauche nous sommes capable de localiser la main mais pas de reconnaître le geste, alors que dans celle de droite, obtenue d'un autre point de vue, nous pouvons reconnaître le geste trois.

CHAPITRE 7 CONCLUSION

Ce chapitre conclut le mémoire, dans un premier temps nous reviendrons sur les travaux réalisés et validerons ou invaliderons les hypothèses émises au chapitre 3. Ensuite nous présenterons les limitations de la solution proposée avant d'énumérer les travaux futurs pour ce projet.

7.1 Synthèse des travaux

Pour le projet de reconnaissance de la gestuelle de la main, nous avons :

- Créé et annoté une nouvelle base d'images multi-caméras de sujets humains en contexte de cobotique industrielle. Cette base de données (COBOTIC) offre une riche variabilité de points de vues, de sujets et d'occlusions visuelles dans l'objectif d'entraîner un réseau performant dans un contexte d'application réel.
- Développé un modèle de reconnaissance de gestes statiques d'abord aussi performant que l'état de l'art sur un jeu de données public (HANDS), et tout aussi performant sur la base de données créée (COBOTIC).
- Proposé une méthode d'agrégation multi-caméra pour améliorer la reconnaissance de la gestuelle notamment en démontrant plus de robustesse face aux occlusions visuelles.

Nous avons également pu valider les deux hypothèses qui étaient :

1. La base de données HANDS créée dans des conditions idéales ne permet pas une généralisation à des conditions réelles plus complexes.
2. L'utilisation d'un système multi-caméra permet l'augmentation des performances dans des conditions réelles notamment en présence d'occlusions visuelles.

En effet, la première hypothèse est validée par l'expérience de croisement de modèles/tests dans laquelle nous avons entraîné un même modèle sur les jeux de données COBOTIC et HANDS et les avons testé sur l'autre (expérience décrite dans la section 5.3.2 et résultats présentés dans la section 6.3.4). La faible performance en test sur COBOTIC du modèle entraîné sur HANDS valide notre hypothèse et la bonne performance du réseau COBOTIC sur les deux jeux de données justifie la création d'une nouvelle base de données.

La seconde hypothèse est validée par l'amélioration des performances du modèle de reconnaissance de la gestuelle lorsqu'on utilise l'agrégation multi-caméra détaillée dans les sections 5.3.3 et 6.3.5.

7.2 Limitations de la solution proposée

Le projet comporte aussi certaines limitations. Tout d'abord, le maillon faible du pipeline proposé se situe au niveau de la détection de la main, plus précisément du choix du modèle Mediapipe utilisé à cette fin. Nos expérimentations semblent indiquer que le modèle de détection de main fonctionne moins bien, sinon pas du tout, pour des résolutions faibles (sujet très éloigné de la caméra) et en présence d'occlusions visuelles.

Ensuite, l'annotation manuelle des données de la base COBOTIC (en terme de rectangle englobant pour la détection de la main) est longue et fastidieuse. Ainsi, seulement la moitié de la base de données (10 participants) a été annotée ce qui implique que seulement la moitié a été exploitée pour l'entraînement et les tests du modèle de reconnaissance de la gestuelle.

Enfin, à cause du long processus d'évaluation éthique par les deux institutions (Polytechnique Montréal et le Conseil National de Recherches du Canada) qui a duré au total 5 mois, ainsi que quelques problèmes techniques et logistiques dans la mise en place du script d'acquisition simultanée par 6 caméras, la collecte de données n'a eu lieu qu'en Aout 2023. À cela s'ajoute le temps considérable pour l'annotation des données. Il en résulte que le temps restant pour faire les expérimentations sur COBOTIC a été plus court, nous laissant moins de temps pour réaliser une analyse encore plus approfondie.

7.3 Améliorations futures

Plusieurs travaux peuvent être envisagés à la suite de ce projet, au delà de l'annotation du reste des données. En voici quelques pistes :

— Analyser la latence du pipeline. À savoir si le temps requis par le modèle de reconnaissance de la gestuelle est convenable pour une utilisation en temps-réel dans la cellule cobotique.
— Exploiter de manière plus explicite les images de profondeurs pour améliorer les performances. Nous pourrions notamment jouer sur la dynamique des images de profondeur à l'acquisition.
— Évaluer l'impact de la résolution spatiale des images sur la détection de la main par Mediapipe.
— Intégrer la reconnaissance de la gestuelle de la main aux autres parties du projet plus global, afin d'offrir une interface opérateur-cobot plus naturelle qu'un simple écran tactile, limitant les déplacements de l'opérateur dans la cellule.

Pour finir, nous noterons que l'intégration de notre solution au projet global peut permettre,

en une pierre deux coups, d'obtenir une solution pour la détection de la main. En effet, d'autres parties du projet nécessitent de prédire l'estimation de pose de l'opérateur dans la cellule cobotique (partie sélection des caméras) ou de faire une segmentation des opérateurs dans les images (partie sécurité du projet). Un exemple de prédiction de l'estimation de pose avec OpenPose [49] est donné en figure 7.1. Prédire l'estimation de pose de l'opérateur implique de prédire où se situe les poignets de l'opérateur. Ainsi, à partir de la position des poignets, nous pouvons alors imaginer plusieurs méthodes pour obtenir les boîtes englobantes de la main.

FIGURE 7.1 Exemple d'estimation de pose obtenue avec OpenPose dans la cellule cobotique.

RÉFÉRENCES

[1] S. Mahmud, X. Lin et J.-H. Kim, "Interface for Human Machine Interaction for assistant devices : A Review," dans *2020 10th Annual Computing and Communication Workshop and Conference (CCWC)*, janv. 2020, p. 0768–0773.

[2] E. Nazarova, O. Sautenkov, M. Altamirano Cabrera, J. Tirado, V. Serpiva, V. Rakhmatulin et D. Tsetserukou, "CobotAR : Interaction with Robots using Omnidirectionally Projected Image and DNN-based Gesture Recognition," dans *2021 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, oct. 2021, p. 2590–2595, iSSN : 2577-1655.

[3] J. Berg et S. Lu, "Review of Interfaces for Industrial Human-Robot Interaction," *Current Robotics Reports*, vol. 1, n°. 2, p. 27–34, juin 2020. [En ligne]. Disponible : https://doi.org/10.1007/s43154-020-00005-6

[4] I. RealSense, "Beginner's guide to depth (Updated)," juill. 2019. [En ligne]. Disponible : https://www.intelrealsense.com/beginners-guide-to-depth/

[5] P. Fankhauser, M. Bloesch, D. Rodriguez, R. Kaestner, M. Hutter et R. Siegwart, "Kinect v2 for mobile robot navigation : Evaluation and modeling," dans *2015 International Conference on Advanced Robotics (ICAR)*, juill. 2015, p. 388–394. [En ligne]. Disponible : https://ieeexplore.ieee.org/abstract/document/7251485

[6] Y. He et S. Chen, "Recent Advances in 3D Data Acquisition and Processing by Time-of-Flight Camera," *IEEE Access*, vol. 7, p. 12 495–12 510, 2019, conference Name : IEEE Access.

[7] G. Kurillo, E. Hemingway, M.-L. Cheng et L. Cheng, "Evaluating the Accuracy of the Azure Kinect and Kinect v2," *Sensors*, vol. 22, n°. 7, p. 2469, janv. 2022, number : 7 Publisher : Multidisciplinary Digital Publishing Institute. [En ligne]. Disponible : https://www.mdpi.com/1424-8220/22/7/2469

[8] M. Tölgyessy, M. Dekan, Chovanec et P. Hubinský, "Evaluation of the Azure Kinect and Its Comparison to Kinect V1 and Kinect V2," *Sensors*, vol. 21, n°. 2, p. 413, janv. 2021, number : 2 Publisher : Multidisciplinary Digital Publishing Institute. [En ligne]. Disponible : https://www.mdpi.com/1424-8220/21/2/413

[9] M. Antico, N. Balletti, G. Laudato, A. Lazich, M. Notarantonio, R. Oliveto, S. Ricciardi, S. Scalabrino et J. Simeone, "Postural control assessment via Microsoft Azure Kinect DK : An evaluation study," *Computer Methods and*

Programs in Biomedicine, vol. 209, p. 106324, sept. 2021. [En ligne]. Disponible : https://www.sciencedirect.com/science/article/pii/S0169260721003989

[10] S. Shrestha, F. Heide, W. Heidrich et G. Wetzstein, "Computational imaging with multi-camera time-of-flight systems," *ACM Transactions on Graphics*, vol. 35, n°. 4, p. 33 :1–33 :11, juill. 2016. [En ligne]. Disponible : https://dl.acm.org/doi/10.1145/2897824.2925928

[11] T.-M. Wang et Z.-C. Shih, "Measurement and Analysis of Depth Resolution Using Active Stereo Cameras," *IEEE Sensors Journal*, vol. 21, n°. 7, p. 9218–9230, avr. 2021, conference Name : IEEE Sensors Journal.

[12] A. Grunnet-Jepsen, A. Takagi, J. Sweetser, T. Khuong et D. Tong, "External Synchronization of Intel® RealSense™ Depth cameras," sept. 2021. [En ligne]. Disponible : https://web.archive.org/web/20210906072141/https://dev.intelrealsense.com/docs/external-synchronization-of-intel-realsense-depth-cameras

[13] K. S. Arikumar, A. Deepak Kumar, T. R. Gadekallu, S. B. Prathiba et K. Tamilarasi, "Real-Time 3D Object Detection and Classification in Autonomous Driving Environment Using 3D LiDAR and Camera Sensors," *Electronics*, vol. 11, n°. 24, p. 4203, janv. 2022, number : 24 Publisher : Multidisciplinary Digital Publishing Institute. [En ligne]. Disponible : https://www.mdpi.com/2079-9292/11/24/4203

[14] S. Y. Alaba et J. E. Ball, "A Survey on Deep-Learning-Based LiDAR 3D Object Detection for Autonomous Driving," *Sensors*, vol. 22, n°. 24, p. 9577, janv. 2022, number : 24 Publisher : Multidisciplinary Digital Publishing Institute. [En ligne]. Disponible : https://www.mdpi.com/1424-8220/22/24/9577

[15] G. Yan, Z. Liu, C. Wang, C. Shi, P. Wei, X. Cai, T. Ma, Z. Liu, Z. Zhong, Y. Liu, M. Zhao, Z. Ma et Y. Li, "OpenCalib : A multi-sensor calibration toolbox for autonomous driving," *Software Impacts*, vol. 14, p. 100393, déc. 2022. [En ligne]. Disponible : https://linkinghub.elsevier.com/retrieve/pii/S2665963822000896

[16] Y. Li, A. W. Yu, T. Meng, B. Caine, J. Ngiam, D. Peng, J. Shen, Y. Lu, D. Zhou, Q. V. Le, A. Yuille et M. Tan, "DeepFusion : Lidar-Camera Deep Fusion for Multi-Modal 3D Object Detection," dans *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. New Orleans, LA, USA : IEEE, juin 2022, p. 17 161–17 170. [En ligne]. Disponible : https://ieeexplore.ieee.org/document/9878415/

[17] Z. Zou, K. Chen, Z. Shi, Y. Guo et J. Ye, "Object Detection in 20 Years : A Survey," *Proceedings of the IEEE*, vol. 111, n°. 3, p. 257–276, mars 2023, conference Name : Proceedings of the IEEE.

[18] T. Diwan, G. Anirudh et J. V. Tembhurne, “Object detection using YOLO : challenges, architectural successors, datasets and applications,” *Multimedia Tools and Applications*, vol. 82, n°. 6, p. 9243–9275, mars 2023. [En ligne]. Disponible : https://doi.org/10.1007/s11042-022-13644-y

[19] S. Rani, D. Ghai et S. Kumar, “Object detection and recognition using contour based edge detection and fast R-CNN,” *Multimedia Tools and Applications*, vol. 81, n°. 29, p. 42 183–42 207, déc. 2022. [En ligne]. Disponible : https://doi.org/10.1007/s11042-021-11446-2

[20] G. Han, S. Huang, J. Ma, Y. He et S.-F. Chang, “Meta Faster R-CNN : Towards Accurate Few-Shot Object Detection with Attentive Feature Alignment,” *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, n°. 1, p. 780–789, juin 2022, number : 1. [En ligne]. Disponible : https://ojs.aaai.org/index.php/AAAI/article/view/19959

[21] Z. Pang, Z. Li et N. Wang, “SimpleTrack : Understanding and Rethinking 3D Multi-object Tracking,” dans *Computer Vision – ECCV 2022 Workshops*, ser. Lecture Notes in Computer Science, L. Karlinsky, T. Michaeli et K. Nishino, édit. Cham : Springer Nature Switzerland, 2023, p. 680–696.

[22] J. Park, C. Xu, S. Yang, K. Keutzer, K. Kitani, M. Tomizuka et W. Zhan, “Time Will Tell : New Outlooks and A Baseline for Temporal Multi-View 3D Object Detection,” oct. 2022, arXiv :2210.02443 [cs]. [En ligne]. Disponible : http://arxiv.org/abs/2210.02443

[23] C. Zheng, X. Yan, H. Zhang, B. Wang, S. Cheng, S. Cui et Z. Li, “Beyond 3D Siamese Tracking : A Motion-Centric Paradigm for 3D Single Object Tracking in Point Clouds,” 2022, p. 8111–8120. [En ligne]. Disponible : https://openaccess.thecvf.com/content/CVPR2022/html/Zheng_Beyond_3D_Siamese_Tracking_A_Motion-Centric_Paradigm_for_3D_Single_CVPR_2022_paper.html

[24] C. Doersch, Y. Yang, M. Vecerik, D. Gokay, A. Gupta, Y. Aytar, J. Carreira et A. Zisserman, “TAPIR : Tracking Any Point with per-frame Initialization and temporal Refinement,” août 2023, arXiv :2306.08637 [cs]. [En ligne]. Disponible : http://arxiv.org/abs/2306.08637

[25] C. Zheng, W. Wu, C. Chen, T. Yang, S. Zhu, J. Shen, N. Kehtarnavaz et M. Shah, “Deep Learning-Based Human Pose Estimation : A Survey,” *ACM Computing Surveys*, p. 3603618, juin 2023. [En ligne]. Disponible : https://dl.acm.org/doi/10.1145/3603618

[26] L. Dipietro, A. M. Sabatini et P. Dario, “A Survey of Glove-Based Systems and Their Applications,” *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 38, n°. 4, p. 461–482, juill. 2008, conference Name : IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews).

[27] M. Oudah, A. Al-Naji et J. Chahl, “Hand Gesture Recognition Based on Computer Vision : A Review of Techniques,” *Journal of Imaging*, vol. 6, n°. 8, p. 73, août 2020, number : 8 Publisher : Multidisciplinary Digital Publishing Institute. [En ligne]. Disponible : https://www.mdpi.com/2313-433X/6/8/73

[28] Y. Si, S. Chen, M. Li, S. Li, Y. Pei et X. Guo, “Flexible Strain Sensors for Wearable Hand Gesture Recognition : From Devices to Systems,” *Advanced Intelligent Systems*, vol. 4, n°. 2, p. 2100046, 2022, _eprint : https ://onlinelibrary.wiley.com/doi/pdf/10.1002/aisy.202100046. [En ligne]. Disponible : https://onlinelibrary.wiley.com/doi/abs/10.1002/aisy.202100046

[29] “Leap motion controller 2 – Ultraleap.” [En ligne]. Disponible : https://leap2.ultraleap.com/leap-motion-controller-2/

[30] A. Vysocký, S. Grushko, P. Oščádal, T. Kot, J. Babjak, R. Jánoš, M. Sukop et Z. Bobovský, “Analysis of Precision and Stability of Hand Tracking with Leap Motion Sensor,” *Sensors*, vol. 20, n°. 15, p. 4088, janv. 2020, number : 15 Publisher : Multidisciplinary Digital Publishing Institute. [En ligne]. Disponible : https://www.mdpi.com/1424-8220/20/15/4088

[31] F. Zhang, V. Bazarevsky, A. Vakunov, A. Tkachenka, G. Sung, C.-L. Chang et M. Grundmann, “MediaPipe Hands : On-device Real-time Hand Tracking,” juin 2020, arXiv :2006.10214 [cs]. [En ligne]. Disponible : http://arxiv.org/abs/2006.10214

[32] S. Hampali, S. D. Sarkar, M. Rad et V. Lepetit, “Keypoint Transformer : Solving Joint Identification in Challenging Hands and Object Interactions for Accurate 3D Pose Estimation,” 2022, p. 11 090–11 100. [En ligne]. Disponible : https://openaccess.thecvf.com/content/CVPR2022/html/Hampali_Keypoint_Transformer_Solving_Joint_Identification_in_Challenging_Hands_and_Object_CVPR_2022_paper.html

[33] J. Park, Y. Oh, G. Moon, H. Choi et K. M. Lee, “HandOccNet : Occlusion-Robust 3D Hand Mesh Estimation Network,” 2022, p. 1496–1505. [En ligne]. Disponible : https://openaccess.thecvf.com/content/CVPR2022/html/Park_HandOccNet_Occlusion-Robust_3D_Hand_Mesh_Estimation_Network_CVPR_2022_paper.html

[34] Q. Gao, Y. Chen, Z. Ju et Y. Liang, “Dynamic Hand Gesture Recognition Based on 3D Hand Pose Estimation for Human–Robot Interaction,” *IEEE Sensors Journal*, vol. 22, n°. 18, p. 17 421–17 430, sept. 2022, conference Name : IEEE Sensors Journal.

[35] M. Wameed et A. M. Alkamachi, “Hand Gestures Robotic Control Based on Computer Vision,” *International Journal of Intelligent Systems and Applications in Engineering*, vol. 11, n°. 2, p. 1013–1021, févr. 2023, number : 2. [En ligne]. Disponible : https://ijisae.org/index.php/IJISAE/article/view/2984

[36] L. Yang, J. Chen et W. Zhu, “Dynamic Hand Gesture Recognition Based on a Leap Motion Controller and Two-Layer Bidirectional Recurrent Neural Network,” *Sensors*, vol. 20, n°. 7, p. 2106, janv. 2020, number : 7 Publisher : Multidisciplinary Digital Publishing Institute. [En ligne]. Disponible : https://www.mdpi.com/1424-8220/20/7/2106

[37] V. Kiselev, M. Khlamov et K. Chuvilin, “Hand Gesture Recognition with Multiple Leap Motion Devices,” dans *2019 24th Conference of Open Innovations Association (FRUCT)*, avr. 2019, p. 163–169, iSSN : 2305-7254.

[38] A. Duarte, S. Palaskar, L. Ventura, D. Ghadiyaram, K. DeHaan, F. Metze, J. Torres et X. Giro-i Nieto, “How2Sign : A Large-Scale Multimodal Dataset for Continuous American Sign Language,” 2021, p. 2735–2744. [En ligne]. Disponible : https://openaccess.thecvf.com/content/CVPR2021/html/Duarte_How2Sign_A_Large-Scale_Multimodal_Dataset_for_Continuous_American_Sign_Language_CVPR_2021_paper.html?ref=https://githubhelp.com

[39] T. Hanke, M. Schulder, R. Konrad et E. Jahn, “Extending the Public DGS Corpus in Size and Depth,” dans *Proceedings of the LREC2020 9th Workshop on the Representation and Processing of Sign Languages : Sign Language Resources in the Service of the Language Community, Technological Challenges and Application Perspectives*. Marseille, France : European Language Resources Association (ELRA), mai 2020, p. 75–82. [En ligne]. Disponible : https://aclanthology.org/2020.signlang-1.12

[40] J. Huang, W. Zhou, Q. Zhang, H. Li et W. Li, “Video-Based Sign Language Recognition Without Temporal Segmentation,” *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32, n°. 1, avr. 2018, number : 1. [En ligne]. Disponible : https://ojs.aaai.org/index.php/AAAI/article/view/11903

[41] P. Molchanov, X. Yang, S. Gupta, K. Kim, S. Tyree et J. Kautz, “Online Detection and Classification of Dynamic Hand Gestures with Recurrent 3D Convolutional Neural Networks,” dans *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Las Vegas, NV, USA : IEEE, juin 2016, p. 4207–4215. [En ligne]. Disponible : http://ieeexplore.ieee.org/document/7780825/

[42] C. Nuzzi, S. Pasinetti, R. Pagani, S. Ghidini, M. Beschi, G. Coffetti et G. Sansoni, “MEGURU : a gesture-based robot program builder for Meta-Collaborative workstations,” *Robotics and Computer-Integrated Manufacturing*, vol. 68, p. 102085, avr. 2021. [En ligne]. Disponible : https://www.sciencedirect.com/science/article/pii/S0736584520302957

[43] P. Sharma et R. S. Anand, “Depth data and fusion of feature descriptors for static gesture recognition,” *IET Image Processing*, vol. 14, n°. 5, p. 909–920, 2020, _eprint : https ://onlinelibrary.wiley.com/doi/pdf/10.1049/iet-ipr.2019.0230. [En ligne]. Disponible : https://onlinelibrary.wiley.com/doi/abs/10.1049/iet-ipr.2019.0230

[44] A. Mavi, “A New Dataset and Proposed Convolutional Neural Network Architecture for Classification of American Sign Language Digits,” févr. 2021, arXiv :2011.08927 [cs]. [En ligne]. Disponible : http://arxiv.org/abs/2011.08927

[45] A. H. Alrubayi, M. A. Ahmed, A. A. Zaidan, A. S. Albahri, B. B. Zaidan, O. S. Albahri, A. H. Alamoodi et M. Alazab, “A pattern recognition model for static gestures in malaysian sign language based on machine learning techniques,” *Computers and Electrical Engineering*, vol. 95, p. 107383, oct. 2021. [En ligne]. Disponible : https://www.sciencedirect.com/science/article/pii/S0045790621003529

[46] V. T. Hoang, “HGM-4 : A new multi-cameras dataset for hand gesture recognition,” *Data in Brief*, vol. 30, p. 105676, juin 2020. [En ligne]. Disponible : https://linkinghub.elsevier.com/retrieve/pii/S2352340920305709

[47] C. Nuzzi, S. Pasinetti, R. Pagani, G. Coffetti et G. Sansoni, “HANDS : an RGB-D dataset of static hand-gestures for human-robot interaction,” *Data in Brief*, vol. 35, p. 106791, avr. 2021. [En ligne]. Disponible : https://www.sciencedirect.com/science/article/pii/S2352340921000755

[48] K. He, X. Zhang, S. Ren et J. Sun, “Deep Residual Learning for Image Recognition,” 2016, p. 770–778. [En ligne]. Disponible : https://openaccess.thecvf.com/content_cvpr_2016/html/He_Deep_Residual_Learning_CVPR_2016_paper.html

[49] Z. Cao, G. Hidalgo, T. Simon, S.-E. Wei et Y. Sheikh, “OpenPose : Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields,” mai 2019, arXiv :1812.08008 [cs]. [En ligne]. Disponible : http://arxiv.org/abs/1812.08008

ANNEXE A FORMULAIRE D'INFORMATION ET DE CONSENTEMENT

POLYTECHNIQUE MONTRÉAL
UNIVERSITÉ D'INGÉNIERIE

Formulaire d'information et de consentement

Titre de l'activité de recherche

SUPERVISION DE LA SCÈNE ET RECONNAISSANCE GESTUELLE POUR DES PLATEFORMES INTERACTIVES DE PROCÉDÉS DE FABRICATION

Équipe de recherche

Responsable de l'activité de recherche
Alaleh Asaran Darban
Candidate au doctorat
Polytechnique Montréal – Département GIGL
Adresse courriel: alaleh.asaran-darban@polymtl.ca

Responsable de l'activité de recherche
Corentin Hubert
Étudiant à la maîtrise
Polytechnique Montréal – Département GIGL
Adresse courriel: corentin.hubert@polymtl.ca

Sous la direction de
Lama Séoud
Professeure adjointe
Polytechnique Montréal – Département GIGL
Numéro de téléphone : 1-514-340-4711 poste 3669
Adresse courriel: lama.seoud@polymtl.ca

Financement de l'activité de recherche

La présente activité de recherche est financée par une subvention recherche-développement coopérative du Conseil National de Recherches du Canada (CNRC), organisme fédéral de financement.

Conflits d'intérêts

L'équipe de recherche n'est pas en situation de conflit d'intérêts dans le contexte de la présente activité de recherche.

SUPERVISION DE LA SCÈNE ET RECONNAISSANCE GESTUELLE POUR DES PLATEFORMES INTERACTIVES DE PROCÉDÉS DE FABRICATION

Préambule

Nous vous invitons à participer à une activité de recherche qui vise à faciliter la communication entre un opérateur et un robot ainsi qu'à assurer la sécurité dans une cellule cobotique en surveillant les déplacements et gestes de l'opérateur à travers des caméras de profondeur.

Cependant, avant d'accepter de participer à cette activité et de signer le présent formulaire d'information et de consentement, veuillez prendre le temps de lire l'information présentée.

Nous vous invitons à poser toutes les questions que vous jugerez utiles à la responsable ou au responsable de l'activité de recherche ou à tout autre membre de l'équipe de recherche et à leur demander de vous expliquer tout mot ou renseignement qui ne serait pas clair. Nous vous invitons également à prendre conseil auprès de toute autre personne de qui vous aimeriez obtenir un avis à propos de votre éventuelle participation.

Présentation générale du projet de recherche

Les industries utilisent généralement des robots préprogrammés avec des codes rigides qui ne peuvent pas coopérer avec les humains. Cependant, certains procédés de fabrication tels que le polissage et l'ébavurage impliquent le réglage adéquat de plusieurs paramètres, tels que la géométrie de la pièce à fabriquer, l'outil requis et la trajectoire de l'outil. Les systèmes cyber-physiques interactifs offrent une solution élégante, donnant à l'opérateur humain essentiellement un rôle de supervision et de contrôle de la qualité. Le concept de collaboration homme-robot ne met pas l'accent sur le remplacement des humains par des robots sur les lieux de travail industriels, mais sur une collaboration entre humains et robots dans un espace de travail commun, la cellule cobotique. Pour que cette collaboration soit sûre et efficace, l'interactivité entre le robot et l'opérateur doit être bien conçue. Dans le contexte de l'industrie 4.0, l'interactivité la plus naturelle est imaginée par la vision par ordinateur et la reconnaissance des gestes.
Les principaux objectifs de l'étude que nous réalisons sont les suivants :

- Détecter de manière automatique la caméra qui offre la meilleure visibilité de l'opérateur dans la cellule cobotique.
- Identifier automatiquement les parties de la pièce de production pointées manuellement par l'opérateur afin d'indiquer un besoin de parachèvement.
- Reconnaitre les gestes de l'opérateur afin de commander le robot (choix de l'outil, nombre de passes, trajectoire, arrêt, début/fin de communication)
- Suivre en temps réel la position de l'opérateur par rapport à la tête de l'outil pour vérifier le critère de distance de séparation minimale.

Afin d'atteindre nos objectifs, nous effectuerons une trentaine d'enregistrements vidéo d'environ 5 minutes durant lesquelles les participants réaliseront des gestes et des mouvements prédéfinis afin d'entraîner et de valider les algorithmes d'apprentissage par ordinateur.

Critères d'inclusion et d'exclusion

Dans le cadre de cette activité, nous cherchons à recruter des volontaires le plus inclusivement possible. Il est nécessaire que les participants possèdent tous les doigts de leurs mains. De plus, les participants ne doivent pas avoir des problèmes de mobilité qui les empêcheraient de se déplacer librement dans la cellule cobotique.

Nature et durée de votre participation à l'activité de recherche

Figure 1 - Cellule cobotique au CTFA

Une seule rencontre est suffisante pour chaque participant et devrait durer entre 10 et 15 minutes. Les rencontres auront lieu au Centre de Technologies de Fabrication en Aérospatiale (CTFA) : 2107 chemin de Polytechnique, Montréal, au 1er étage, dans la cellule cobotique présentée à la Figure 1. Par soucis de protection, vous devrez porter des chaussures de sécurité ou une protection par-dessus vos chaussures, fournies sur place.

Vous devrez réaliser des gestes à plusieurs endroits dans la cellule cobotique pour améliorer la diversité des données. Majoritairement, ces gestes consisteront à faire des chiffres avec les doigts de la main, vous déplacer autour de la table de parachèvement et indiquer une ou des surfaces sur des pièces de fabrication placées sur la table. Vous serez donc filmé (y compris votre visage) par 6 caméras RGB-D, placées autour de la cellule. De plus, quelques informations vous seront

demandées : votre sexe, votre âge ainsi votre poids et taille. Vous n'entrerez pas en contact avec le robot; de plus, celui-ci sera éteint lors des acquisitions vidéo.

Risques pouvant découler de votre participation à l'activité de recherche :

La présente activité de ne devrait pas entraîner des risques plus grands que ceux que vous rencontrez dans votre vie de tous les jours. Étant donné que le robot sera éteint lors de la collecte de données, il n'y aura aucun risque lié à l'interaction humain/machine. Ceci sera vérifié par le personnel responsable avant la séance d'acquisition vidéo. Par conséquent, le robot n'effectuera aucun mouvement lui-même et ne présentera aucun danger pour les participants.

Inconvénients pouvant découler de votre participation à l'activité de recherche

Les principaux inconvénients liés à votre participation à l'activité de recherche seront le déplacement requis au CTFA et le temps nécessaire pour faire l'enregistrement de la séquence (environ 15 minutes).

Avantages pouvant découler de votre participation à l'activité de recherche

Vous ne retirerez aucun bénéfice personnel de votre participation à la présente activité de recherche. Toutefois, votre participation permettra de faire avancer l'état des connaissances dans le domaine de la reconnaissance de la gestuelle. L'interactivité par la gestuelle dans des environnements de production est encore peu explorée.
Une interactivité efficace entre humains et robots ouvrira la voie vers l'automatisation de plusieurs procédés industriels qui nécessitent l'intervention d'humains pour la réalisation des tâches. Cela représentera un avantage notable pour les compagnies canadiennes, en particulier les PMEs, en leur évitant d'avoir recours à des expertises pointues et à du capital considérable pour automatiser leurs procédés de fabrication.

Compensation financière

Aucune compensation financière n'est prévue pour ce projet.

Participation volontaire et possibilité de retrait :

Votre participation à la présente activité de recherche est volontaire. Vous êtes donc libre de refuser d'y participer et pouvez à tout moment décider de vous en retirer sans avoir à motiver votre décision et sans risquer d'en subir de préjudice. Vous n'avez qu'à en informer la personne-ressource de l'équipe de recherche et ce, par simple avis verbal.

En cas de retrait, vous pouvez demander la destruction des données vous concernant. Cependant, il sera impossible de retirer vos données ou votre matériel des analyses menées une fois ces dernières publiées ou diffusées.

Tout au long des activités de recherche, vous recevrez en temps opportun l'information pertinente en lien avec votre participation.

L'équipe de recherche veillera à ce que le processus de consentement soit mené dans un cadre privé et non coercitif. La participation est volontaire et le fait de refuser de participer n'affectera en aucun cas votre relation professionnelle ou personnelle avec l'équipe de recherche et/ou vos collègues et supérieurs à Polytechnique Montréal, au CTFA ou ailleurs.

L'équipe de recherche et le comité d'éthique de la recherche se réservent le droit de vous retirer de l'étude si vous ne respectez pas les consignes, s'il existe des raisons administratives d'abandonner l'activité, ou pour toutes autres raisons concernant la faisabilité de l'étude. Si une telle situation survient, l'équipe de recherche vous en informera dès que possible.

Confidentialité et protection de vos données

L'équipe de recherche recueillera et consignera toutes vos données de manière sécuritaire de façon à en protéger le caractère confidentiel. De plus, votre nom sera remplacé par un code de recherche dans les données stockées. Le code permettra de lier votre identité aux données de recherche mais le décodage ne pourra se faire que par le chercheur principal ou par une personne déléguée par ce dernier.

Les données recueillies lors de la collecte seront stockées sur un serveur de l'école Polytechnique Montréal et ne seront accessibles que par les membres du groupe de recherche. Il n'est pas envisagé de destruction des données immédiatement à la fin du projet, nous les conserverons pour une durée minimale de 10 ans à partir de la fin du projet.

Vos noms ou vos coordonnées de contact ne paraîtront dans aucun rapport et ne seront jamais divulgués à des personnes hors du groupe de recherche. La recherche fera l'objet de publications scientifiques. Dans le cas de l'utilisation d'une image dans un article sur laquelle vous apparaissez, votre visage sera flouté de sorte que vous ne pourrez pas être identifié. En revanche, les membres du groupe de recherche auront accès aux données originales et pourront voir votre visage.

Les seules ajouts/modifications qui seront autorisés sur vos images seront l'annotation de celle-ci. Par exemple, la pose de la main, la trajectoire de la main sur une vidéo ou la direction du visage. L'entraînement des algorithmes n'entraînera pas de modifications des images.

Concernant les informations qui vous seront demandées, à savoir le sexe, l'âge, le poids et la taille, celles-ci ne seront utilisées qu'à des fins statistiques. L'objectif ici est de connaître les moyennes, écarts-type et autre valeur statistiques pour chaque information afin d'évaluer l'hétérogénéité de la base de données. Ces données enregistrées ne seront pas reliées à votre identité.

Vous avez le droit de consulter votre dossier de recherche pour vérifier l'exactitude des renseignements recueillis aussi longtemps que l'équipe de recherche ou Polytechnique Montréal détiendront ces informations. Cependant, afin de préserver l'intégrité scientifique du projet de recherche, certaines informations seront accessibles seulement à la fin du projet de recherche.

Diffusion des résultats de la recherche

Les résultats issus des recherches menées grâce aux données récoltées dans cette étude pourront être publiés dans des thèses et mémoires de recherche et des revues scientifiques, ou communiqués dans des congrès scientifiques. Toutefois, aucune information ou image pouvant vous identifier ne sera alors dévoilée.
Les participants seront informés des résultats de l'étude en étant invités par courriel à participer aux présentations finales des résultats à Polytechnique Montréal.

Publication d'une banque de données

Nous envisageons d'éventuellement rendre publiques les données récoltées dans la présente étude, pour que d'autres groupes de recherche puissent les utiliser uniquement à des fins de recherche. Dans ce cas, vous serez prévenu de la publication de la banque de données et un nouveau formulaire de consentement vous sera transmis présentant les nouvelles modalités. Seules les données des personnes signataires du nouveau formulaire seront incluses dans la banque de données publique. Il vous sera alors possible de refuser que les données vous concernant soient partagées avec d'autres chercheurs.

Indemnisation en cas de préjudice et droits des participant(e)s

Si vous deviez subir quelque préjudice que ce soit par suite de votre participation à cette activité de recherche, vous ne renoncez à aucun de vos droits ni ne libérez les chercheurs, l'organisme de financement ou Polytechnique Montréal de leurs responsabilités légales et professionnelles.

Personnes-ressources

Si vous avez des questions sur les **aspects scientifiques** du projet de recherche ou pour vous **retirer de l'étude**, vous pouvez contacter Lama Séoud au (514) 340-4711, poste 3669 ou encore par courriel à lama.seoud@polymtl.ca

Pour toute préoccupation sur vos droits ou sur les responsabilités de l'équipe de recherche concernant votre participation à ce projet, vous pouvez contacter le Comité d'éthique de la recherche de Polytechnique Montréal au (514) 340-4711, poste 4420 ou encore par courriel à ethique@polymtl.ca

SUPERVISION DE LA SCÈNE ET RECONNAISSANCE GESTUELLE POUR DES PLATEFORMES INTERACTIVES DE PROCÉDÉS DE FABRICATION

Consentement à la participation au projet de recherche

1. J'ai pris connaissance de la documentation ci-jointe, décrivant la nature et le déroulement du projet de même que les risques et les inconvénients qui pourraient survenir.
2. Je comprends que j'ai droit à des réponses satisfaisantes aux questions que je poserais quant à mon implication dans ce projet tout au long de ma participation.
3. Je consens à participer librement à ce projet, après avoir obtenu et pris le temps d'y réfléchir à ma satisfaction et sans avoir subi de pression à cet effet.
4. Je comprends qu'en participant à ce projet de recherche, je ne renonce à aucun de mes droits ni ne dégage les chercheurs de leurs responsabilités.
5. Je comprends que je peux consulter le dossier que l'équipe de recherche constitue sur moi.
6. Je pourrai à tout moment, sur simple avis de ma part, revenir sur ma décision de participer et serai alors immédiatement libéré de mon engagement.
7. J'ai reçu une copie du présent document.
8. Je consens à donner les informations suivantes me concernant: sexe, âge, poids et taille à condition que celle-ci soit sauvegardées de manière anonyme.

Enregistrement vidéo

9. Je consens à ce que des photos et vidéos dans lesquelles on me voit soient utilisés dans le contexte d'entraînement d'algorithme comprenant l'annotation des données.
10. Je consens à ce que des images dans lesquelles on me voit soient utilisées dans des publications scientifiques à condition que mon visage soit flouté de manière à ce qu'on ne puisse pas m'identifier.

Contact à l'avenir

11. J'autorise le groupe de recherche à me recontacter pour me transmettre leurs résultats de recherche.
12. J'autorise le groupe de recherche à me recontacter pour me faire parvenir un nouveau formulaire de consentement dans l'éventualité du partage des données sous forme de banque publique. Je comprends que je serai alors en droit de refuser les nouvelles modalités.

Prénom et nom du participant (caractère d'imprimerie)	Signature du participant Date :

Information de contact :

Adresse courriel : _____ Numéro de téléphone: _____

SUPERVISION DE LA SCÈNE ET RECONNAISSANCE GESTUELLE POUR DES PLATEFORMES INTERACTIVES DE PROCÉDÉS DE FABRICATION

Engagement de l'équipe de recherche

Je confirme que moi ou mon représentant avons expliqué à la personne précitée la nature de sa participation à la présente activité de recherche, demandé si elle avait des questions, répondu à ses questions. Nous avons clairement indiqué qu'elle ou il demeurait libre de participer et de mettre un terme à sa participation à tout moment, par simple avis verbal. Je m'engage, avec l'équipe de recherche, à respecter les modalités décrites dans le présent formulaire d'information et de consentement et déclare en avoir remis une copie signée à la personne.

Lama SEOUD (caractère d'imprimerie)	Signature de la responsable Date :
Alaleh ASARAN (caractère d'imprimerie)	Signature de l'étudiante Date :
Corentin HUBERT (caractère d'imprimerie)	Signature de l'étudiant Date :