Democratic and Popular Republic of Algeria

Ministry of Higher Education and Scientific Research

University Mohamed Khider of Biskra

Faculty of Exact Sciences and Science of Nature and Life

Department of Computer Sciences

Ref : ……..

Thesis Presented to obtain the degree of

Doctorate in Computer Science

Option: Image and Artificial Life

Entitled:

Utilisation des techniques de deep learning pour l'amélioration de la gestion des occultations pour la réalité augmentée

Presented by:

BEKIRI Roumaissa

Publicly defended on: 04/03/ 2024

In front of the Jury committee composed of:

Mr. Foudil Cherif	Professor	University of Biskra	President
Mr. Mohamed Chaouki Babahenini	Professor	University of Biskra	Supervisor
Mr. Abdel Ouaheb Moussaoui	Professor	University of Setif 1	Examiner
Mr. Nadjib Kouahla	Professor	University of Guelma	Examiner
Mr. Ahmed Tibermacine	MCA	University of Biskra	Examiner

. . . .. To my Parents,
Who have taught me the love of learning and research as the most crucial thing in life. I am forever thankful to them for being so supportive, without their endless love and encouragement; I would never have been able to complete the way.
This dissertation is also dedicated to my dear sisters Naouel, Roufia, Soulef, Yasmine, Moufida, Manel, my dear nephew Ahmed Amine.
I would like to dedicate this work to my best friend, who was there for me and gave me lots of support and continuous encouragement throughout this very hard and long way. I am truly thankful for having you in my life.

Thank you
Roumaissa

Acknowledgement

First, I would like to express my gratitude to Allah, the Almighty, who guided me through this path and gave me the courage and patience to complete this dissertation.

I would like to thank my supervisor **Prof. Babahenini Mohamed Chaouki** sincerely, I am extremely grateful for his support and the loose guidance I had during these long years, that allowed me to identify my own path without drifting too far away from it. Moreover, I learned several secondary skills from him, in particular, how to be patient and ambitious.

My hearty thanks go to my jury members for their time necessary to read and understand the manuscript. Their expertise and critical evaluation have significantly contributed to the quality and credibility of this research.

My thanks go to my colleagues at the University Lab, particularly, Asma Besbes, Boudouh Naouara, Rima Benelmir, Ouamene Fatima Zahra,.... and last but not least laboratory administrator Rahim Affaf.

BEKIRI Roumaissa

ملخص

يمثل الواقع المعزز جبهة تكنولوجية مبتكرة تمزج بسلاسة بين العالمين الرقمي والفعلي. يمنح المستخدمين تجربة بيئات محسنة وتفاعلية ومشبعة حيث تتعايش المعلومات المولدة بواسطة الحاسوب والكائنات والتجارب مع العالم الحقيقي. تقدير وضع اليد هو مجال في رؤية الحاسوب وتعلم الآلة يركز على استنتاج التكوين الثلاثي الأبعاد الدقيق وحركة اليد استناداً إلى البيانات ثنائية الأبعاد أو ثلاثية الأبعاد من الكاميرات أو الأجهزة الاستشعار. بالإضافة إلى ذلك، يعتبر تحديا أكبر من تقدير أجزاء الجسم البشري الأخرى بسبب حجم اليد الصغير وتعقيدها الأكبر وتحجيم ذاتي مهم. في هذا السياق، نقوم بفحص مشكلة التغطية طوال التفاعل.

تقدم هذه الرسالة أسلوباً كلاسيكياً لحل مشكلة التغطية في نظام واقع معزز ديناميباستخدام خوارزمية التصوير الفوتوغرافي عن كثب. بالإضافة إلى ذلك، نقوم بإنشاء مجموعات بيانات واقعية مكونة من مشاهد فيزيائية من كاميرات زوايا مختلفة. بالإضافة إلى ذلك، نستخدم بيانات خرائط العمق التي تثبت أنها استراتيجية قيمة لإدارة التغطية بشكل فعال في سيناريوهات الواقع المعزز، حيث تقدم معلومات أساسية حول العلاقات المكانية والمسافات بين الكائنات في المشهد وتمكن من التمييز بدقة حول الكائنات التي يجب أن تظهر أمام الأخرى أو خلفها. لقد أثبت هذا النهج أنه أداة حيوية في معالجة تحدي التغطية المستمر، مما يتيح خلق تجارب واقع معزز أكثر تميزاً وسياقاًياًا. ثم نمد دراستنا في العملية عبر الإنترنت. نتناول مشكلة تقدير وضع اليد ونقدم أسلوب تصاعدي جديد من الصور اللونية ثنائية الأبعاد، الذي يهدف إلى التعامل مع مشكلات التغطية خلال تفاعل اليد مع الكائنات في الوقت الحقيقي. مع ظهور التعلم العميق، هناك تحول نحو استخدام شبكات عصبية عميقة لتعلم والتقاط وتلاعب الكائنات بدقة. يمكنه اكتشاف وتنبؤ بوضع اليد ثنائي الأبعاد و ْضسونت إطارنا المسمى ْشبكة ثلاثي الأبعاد بكفاءة من خلال استخدام ثلاث وحدات رئيسية: استخراج الميزات الذي يستخدم استراتيجية التعلم النقل لاستخراج خرائط الميزات، وتصوير وضع اليد ثنائي الأبعاد، وتقدير وضع اليد ثلاثي الأبعاد. النتائج الكمية والنوعية على ثلاث مجموعات بيانات، تظهر باستمرار أن أسلوب التصاعد الخاص بنا يفوق أساليب تقدير وضع اليد الحالية الرائجة.

الكلمات المفتاحية: الواقع المعزز، رؤية الحاسوب،شبكات عصبية عميقة،تقدير وضع اليد،ثنائي الأبعاد،ثلاثي الأبعاد

Abstract

Augmented Reality (AR) represents a groundbreaking technological frontier that seamlessly merges the digital and physical worlds. At the core of this technology lies the need for precise and intuitive interactions, and hand pose estimation has emerged as a crucial component in achieving this goal. Besides, it is considered more challenging than other human part estimations due to the small size of the hand, its greater complexity, and its important self-occlusions. In this context, we investigate the occlusion issue throughout the interaction.

This dissertation proposes a classical method for resolving occlusion in a dynamic augmented reality system by employing a close-range photogrammetry algorithm. Additionally, we create realistic datasets composed of physical scenes from different viewpoint cameras. Further, we apply depth map data that proves to be a valuable strategy for effectively managing occlusion in augmented reality scenarios, which provides essential information about the spatial relationships and distances between objects in the scene and can accurately discern which objects should appear in front of or behind others. This approach has proven instrumental in addressing the persistent challenge of occlusion, allowing for seamless and contextually creating more immersive AR experiences. Then, we extend our study in the online process. We address the problem of hand pose estimation and present a new regression method from monocular RGB images, which aims to tackle occlusion issues during hand-object interaction in real-time. With the advent of deep learning, there has been a shift towards using deep neural networks to learn, grasp, and manipulate objects accurately. The proposed framework, defined as the "ResUnet network," provides effective capabilities in detecting and predicting both 2D and 3D hand pose. This is achieved by utilizing three primary modules: feature extraction, which employs a transfer learning technique to extract feature maps; 2D pose regression; and 3D hand estimate. Our regression methodology consistently outperforms the current state-of-the-art hand pose estimation approaches, as demonstrated by the quantitative and qualitative findings obtained from three datasets.

Keywords: Augmented Reality, Occlusion, hand pose estimation, deep learning, Human-computer interaction, 2D pose, 3D pose

Résumé

La Réalité Augmentée(RA) représente une frontière technologique révolutionnaire qui fusionne de manière transparente les mondes numérique et physique. Au cœur de cette technologie transformative réside le besoin d'interactions précises et intuitives, et l'estimation de la pose de la main s'est imposée comme un composant crucial pour atteindre cet objectif. De plus, Il est considérée comme plus complexe que d'autres estimations de parties du corps humain en raison de la petite taille de la main, de sa plus grande complexité et de ses auto-occultations importantes. Dans ce contexte, nous examinons la question de l'occultation tout au long de l'interaction.

Cette thèse propose une méthode classique pour résoudre le problème de l'occultation dans un système de réalité augmentée dynamique en utilisant un algorithme de photogrammétrie à courte portée. De plus, nous créons des ensembles de données réalistes composés de scènes physiques prises depuis différentes perspectives de caméras. Nous utilisons des données de cartes de profondeur qui se révèlent être une stratégie précieuse pour gérer efficacement l'occultation dans les scénarios de réalité augmentée, fournissant des informations essentielles sur les relations spatiales et les distances entre les objets dans la scène et permettant de discerner avec précision quels objets doivent apparaître devant ou derrière d'autres. Ensuite, nous étendons notre étude dans le processus en ligne. Nous abordons le problème de l'estimation de la pose de la main et présentons une nouvelle méthode de régression à partir d'images RVB monoculaires, visant à résoudre les problèmes d'occultation lors de l'interaction main-objet en temps réel. Avec l'avènement de l'apprentissage en profondeur, il y a eu un passage à l'utilisation de réseaux neuronaux profonds pour apprendre, saisir et manipuler des objets avec précision. Le cadre proposé, défini comme le "réseau ResUnet", offre des capacités efficaces pour détecter et prédire à la fois la pose de la main en 2D et en 3D. Cela est réalisé en utilisant trois modules principaux : l'extraction de caractéristiques, qui utilise une technique d'apprentissage par transfert pour extraire des cartes de caractéristiques ; la régression de la pose en 2D ; et l'estimation de la main en 3D. Notre méthodologie de régression dans cette recherche surpasse de manière constante les approches actuelles de l'estimation de la pose de la main, comme le démontrent les résultats quantitatifs et qualitatifs obtenus à partir de trois ensembles de données.

Mots Clée : Réalité augmentée, Occultation, Estimation de la pose de la main, Apprentissage profond, Interaction homme-machine, 2D position, 3D position.

Contents

List of Figures

List of Tables

Chapter 1

General introduction

1.1 Overview

The innate human desire to explore and engage with alternative and reciprocal realities has deep roots in human beings. Figure 1.1 provides a concise overview of the history of Augmented Reality(AR). In the late 1960s, Ivan Sutherland [38] pioneered the creation of the first head-mounted three-dimensional display device, famously known as "The Sword of Damocles." This groundbreaking invention laid the initial foundation for today's AR technology. Users of this head-mounted device could immerse themselves in an alternate reality where computer-generated images could interact with them by tracking their head movements.
The term "Augmented Reality" was formally introduced in the early 1990s by Tom Caudell. Shortly after that, L. Rosenberg [1] developed the first functional AR system, known as "Virtual Fixtures," which aimed to enhance the efficiency of operators in various tasks.

Since the late 1990s, AR technology began to find its way into diverse applications, including space navigation, live sports event broadcasts, and battlefield simulations [4,6]. Over time, modern AR applications have expanded into a wide array of domains. For instance, AR has assisted surgeons during complex procedures in the medical field.

Additionally, in education, AR-based learning materials, as advocated by I. Radu [39], have been shown to enhance students motivation, improve content comprehension, and promote long-term memory retention, among other benefits.

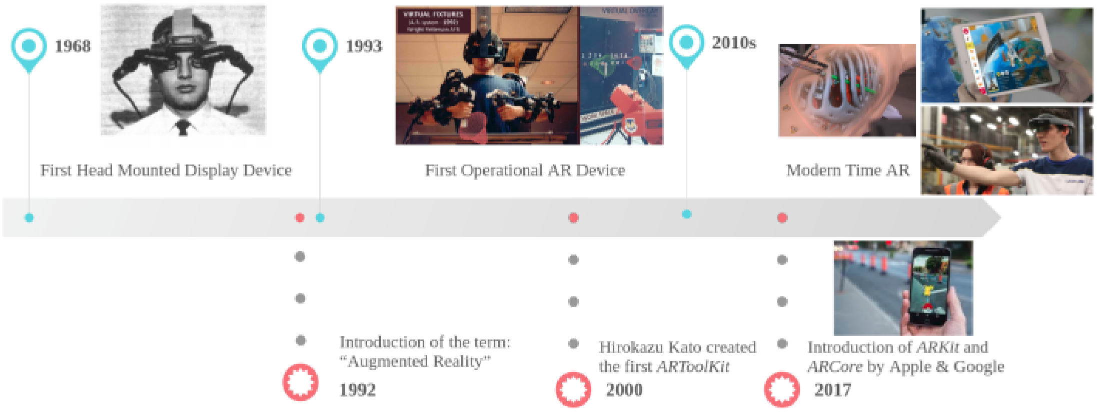

1968
First Head Mounted Display Device
1993
First Operational AR Device
2010s
Modern Time AR
Introduction of the term:
"Augmented Reality"
1992
Hirokazu Kato created
the first *ARToolKit*
2000
Introduction of *ARKit* and
ARCore by Apple & Google
2017

Figure 1.1: A Concise History of Augmented Reality(AR) [1]

Augmented reality(AR) has seen a surge in growth, benefiting from the widespread use of the Internet and mobile devices. In 2017, Apple introduced ARKit for iOS, and Google rolled out ARCore for Android, which greatly simplified the creation of AR applications for mobile devices. From users' perspective, mobile phones have significantly reduced the need for additional equipment to experience AR, making it more accessible. Consequently, the AR industry is experiencing rapid expansion. According to a recent projection by Goldman Sachs in 2016 [40], the AR and Virtual Reality(VR) industries are anticipated to reach a combined market value of 95 billion by 2025 [41].

The pose estimation problem in augmented reality(AR) is crucial to creating realistic and immersive AR experiences. Pose estimation refers to determining a physical object's precise position and orientation, typically a camera or a user's device, relative to the surrounding environment. In this thesis, we focus on on hand pose estimation that considered as a pivotal process that involves deducing the three-dimensional coordinates of each joint in the human hand from visual inputs. While the concept of hand pose estimation bears some resemblance to the task of estimating the poses of the human body, and indeed, some hand pose algorithms draw inspiration or techniques from body pose estimation methods, it is essential to recognize the nuanced disparities that render hand pose estimation a more demanding endeavor. These distinctions arise from factors such as the similarity between the fingers, the intricate dexterity exhibited by the hand, and the frequent occurrence of self-occlusion –wherein parts of the hand obscure one another in the visual input. These challenges underscore the complexity of hand pose estimation and prompt the development of specialized approaches tailored to the unique intricacies of the human hand.

Markerless hand pose estimation is primarily attributed to two distinct approaches: discriminative and generative approaches. The discriminative approach, characterized as appearance-based, directly derives the hand pose from the input data. On the other hand, the generative approach employs a hand model and tries to align this model with the observed data to estimate the hand pose. Some exclusively use the discriminative approach [42, 43] while others solely rely on the generative approach [44, 45]. In contrast, some pipelines adopt a hybrid strategy [46], effectively combining discriminative and generative methods elements within a single framework.

1.2 Challenges

In this section, we outline four frequently encountered challenges that arise when attempting to tackle the task of 2D-3D hand pose estimation under demanding conditions. These challenges serve as the foundation for understanding the complexity inherent in these issues.

- **Occlusion:** The occlusion challenge within hand pose estimation presents a significant issue for computer vision systems striving to accurately estimate the positions and orientations of a hand or fingers in images and videos. This challenge encompasses various scenarios, including partial occlusion, where objects, other body parts, or the hand itself obscure portions of the hand, as well as full occlusion, where the entire hand is concealed from view, often due to external objects or movements out of the camera field of vision. Adding complexity to the issue, dynamic occlusion occurs when objects or the user's other hand intermittently obstruct parts of the hand during motion. Some applications rely on data from multiple sensor modalities, necessitating consistent depth information across different sensors. Ambiguity arises from occlusion, as multiple plausible hand configurations may be consistent with observed data, making distinguishing between them a complex task. In scenarios where hands interact with objects, such as in human-computer interaction, occlusion poses an additional layer of intricacy. Furthermore, addressing occlusion swiftly and accurately is essential for real-time applications like augmented and virtual reality.

  Researchers and developers address the occlusion challenge through various techniques, including advanced neural network architectures that can learn to handle occluded data, multi-sensor fusion to mitigate occlusion effects, and data augmentation strategies that introduce occlusion patterns into training datasets. Overcoming occlusion is critical to successfully deploying hand pose estimation systems in real-world applications, where hands are often in dynamic and occluded environments.

- **Self-occlusion:** Self-occlusion occurs when parts of the hand obscure other parts, such as fingers hiding behind the palm or one finger partially obstructing another. This phenomenon introduces a high level of complexity into the hand pose estimation process. Thus, this challenge not only limits the visibility of crucial hand landmarks but also creates ambiguity in interpreting the hand configuration accurately. Consequently, distinguishing between different possible hand poses becomes challenging when parts of the hand are hidden from view. Many studies address the self-occlusion challenge by employing advanced algorithms and models that can handle such occluded scenarios [26, 27, 47], often leveraging deep learning techniques and multi-modal sensor data to enhance accuracy. Successfully overcoming self-occlusion is crucial for various applications, from sign language recognition to human-computer interaction, where the hand's precise pose is integral to system performance and user experience. Furthermore, model-based approaches address occlusions to some extent by calculating visibility but fall short of offering a comprehensive solution, particularly when dealing with single-camera systems.

- **Data Variability:** The data variability presents a challenge in hand pose estimation, where the goal is to accurately determine the positions and orientations of a human hand or fingers in images and videos. This challenge encompasses diverse dimensions, including variations in hand shapes and sizes, disparities in skin tones and lighting conditions, and differences in background and environmental settings. Moreover, the dynamic nature of hand gestures and movements and variations in viewpoint and camera distances add further complexity. Using different sensor types and data modalities in hand pose estimation systems introduces another layer of variability. Many researchers rely on extensive and diverse training datasets to address this challenge effectively, employ data augmentation techniques to simulate various conditions, and leverage robust feature extraction methods, such as deep learning architectures. Successful hand pose estimation models must demonstrate resilience and adaptability to the inherent variability encountered in real-world data, ensuring accurate and reliable performance across various scenarios.

- **Real-Time Processing:** The real-time processing challenge in hand pose estimation is a pivotal obstacle in developing systems that can swiftly and accurately determine the positions and orientations of a person's hand or fingers in images or videos. It encompasses a multifaceted set of demands, including the imperative for extremely low latency, particularly in applications such as virtual reality, augmented reality, and human-computer interaction, where any perceptible delay in hand pose estimation can significantly degrade the user experience. High frame rates are essential, requiring systems to process a continuous stream of images at 30 frames per second or more, all while delivering pose estimates within the same time frame. Achieving computational efficiency is paramount, often entailing intricate computations, such as deep learning inference or 3D model fitting, and necessitating algorithmic and hardware optimizations. Coordinating data from multiple sensors and ensuring synchronization is critical to sensor integration. Developers often employ parallel processing techniques to meet real-time constraints, distributing the computational load in various processing units. Additionally, energy efficiency is vital for mobile or battery-powered devices, necessitating minimal power consumption. Addressing these multifarious demands involves a blend of algorithmic refinements, efficient hardware and software implementations, and strategic model optimizations to ensure that hand pose estimation systems can deliver seamless and responsive interactions in real-time applications.

To conclude, addressing the issue of hand pose estimation through computer vision methods poses significant challenges from a technical and practical perspective. In this thesis, we have endeavored to address certain limitations in current approaches with two distinct contributions.

1.3 Problem statement

Augmented Reality(AR) is a groundbreaking technology that has garnered substantial interest from researchers in diverse fields and sectors. Through its seamless fusion of digital content with the tangible world, AR provides users with enriched encounters, context-aware knowledge, and interactive functionalities. With the increasing number of companies dedicated to advancing the commercial applications of Augmented Reality(AR), Virtual Reality(VR), and wearable devices, the necessity for a hand-based input mechanism has become paramount. This need arises from the pursuit of creating user experiences that are not only technologically sophisticated but also seamlessly integrated and genuinely immersive.

Hands are regarded as the most efficient and intuitive tools for human-computer interaction(HCI) among various body parts. Pose estimation can also be incorporated into the pipeline for gestures and sign language recognition tasks. It represents a set of techniques

designed to provide precise 3D coordinates of keypoints, facilitating the determination of the accurate hand pose. This computational task presents a significant challenges owing to several factors. However, the accurate estimation of hand poses in the presence of occlusions is a persistent obstacle to the seamless incorporation of augmented reality in various applications. Firstly, the sheer diversity of potential hand poses, encompassing intricate configurations and variations, adds complexity to the estimation process. Secondly, self-occlusion, where parts of the hand obscure others, further compounds the challenge by limiting the visibility of critical keypoints. Additionally, the resemblance of finger appearances, especially in specific orientations, introduces ambiguity into the estimation, making it challenging to discern between similar poses.
Hand Pose Estimation challenge can be tackled from various angles, depending on available data, project budget, required prediction precision, or any other constraints.

This dissertation uses deep learning techniques to address occlusion issues in the hand pose estimation field. The research aims to develop novel deep-learning models and techniques that can effectively manage occluded hand poses, thereby enhancing the accuracy and dependability of augmented reality systems and creating new opportunities for natural and immersive interactions in enriched environments.

1.4 Thesis Contributions

All the above considerations lead this thesis to address the challenge of hand pose estimation. The research explores two main categories of approaches: handcrafted methods and deep learning techniques. Hence, we investigate in the first part a classical method for handling occlusion problems by employing two technologies: AR technology and photogrammetry algorithm. In the second part, we focus on hand pose estimation, a challenge significantly benefiting from using state-of-the-art deep learning-based algorithms. It is a crucial topic in many computer vision applications that flourish in augmented reality and offer the possibility of engaging with virtual reality devices. Due to the complicated anatomy and dexterous motion of human hands, estimating hand pose is a significant academic and technical issue. A complete study of the impact of the hand pose estimator research process will be considered. Thus, we extend the study to online dynamic hand pose estimation, taking over the whole pipeline destined to handle occlusion during the interaction with the virtual objects in real-time. The main contributions of this thesis can be summarized as follows:

- **Real-time handling occlusion in augmented reality based on photogram-**

metry: We propose an original framework based on a photogrammetry algorithm and AR technology for handling occlusion that occurs during interaction in real-time. We apply close-range photogrammetry for 3D reconstruction of real scenes based on imagery acquisition to get a highly accurate 3D model. Additionally, we employ markerless tracking used in AR applications to track the position and orientation of objects without the need for physical or fiducial markers. This approach enables a more natural and immersive AR experience. We validated our approach by creating small datasets representing realistic scenes from different viewpoints. The obtained results show the efficiency and accuracy of the proposal for solving the occlusion challenge in AR systems.

- **Hand pose estimation based on regression method from monocular RGB cameras for handling occlusion:** We propose a novel architecture based on deep-learning-based aims to gather the previous research advances systems that tackled the most challenges with discussing each existing research addressing its advantages and drawbacks. Thus, we identify the numerous benchmark datasets with their characteristics and the famous evaluation metrics used to evaluate these methods. Further, we discuss the results of potential research direction in terms of time speed, precision, and type of Convolutional Neural Network(CNN) architecture in this rapidly expanding field. Additionally, the main contribution of our research is developing a deep learning framework to learn full 2D and 3D hand pose estimation from an egocentric image. The proposed architecture, "ResUnet network," aims to handle occlusion challenges during hand interaction with objects through training. Finally, both information of 2D and 3D prediction are merged to perform an accurate estimation of hand pose.

1.5 Thesis outline

The thesis is organized as follows: Chapter 1 presents the theoretical chapter that provides the foundations of augmented reality(AR). The reader will find the purpose of augmented reality, its technical definition, its functional taxonomy, different applications, and the most known devices. In Chapter 2, we lay out the issues of hand pose estimation. We delve into a comprehensive literature review that proposes explicitly a new taxonomy that is grouped with the recent research depending on the type of input data as well as existing solutions in the literature. We identify the numerous benchmark datasets with their characteristics and the popular evaluation metrics used to evaluate these methods. Chapter 3 introduces the classical contribution of our thesis based on resolving occlusion in an Augmented Reality environment using a close-range photogrammetry algorithm. In Chapter 4, we propose a deep learning framework for hand estimation by performing 2D and 3D hand poses. Finally, we conclude the manuscript in Chapter 6 by summarizing the contributions of the thesis and proposing several directions for future research.

Chapter 2

Augmented Reality: Definition, Applications, Interaction

2.1 Introduction

Augmented Reality(AR) is an enhanced version of the real world, achieved through the use of computer-generated digital information. These include visual, sound, and other sensory elements. AR uses computer hardware and software, such as apps, consoles, screens, or projections, to combine digital information with the real-world environment [48].

This chapter presents different definitions of Augmented Reality articulated by various researchers within the domain. Further, we introduce Milgram's Reality-Virtuality Continuum, which outlines a spectrum ranging from the authentic physical environment to the entirely virtual realm. This framework provides a structured understanding of the varying degrees of digital augmentation within reality and virtuality.

To enhance the visibility of Augmented Reality (AR), we conduct a comparative analysis between AR and Virtual Reality (VR). The context of AR introduces virtual components into the real world, allowing users to engage with both the digital and physical realms concurrently. In contrast, VR creates immersive environments that disconnect users from their physical surroundings, focusing solely on the virtual experience.

The physical world and virtual environments are seamlessly blended in the AR application. These applications have found extensive utilization in various aspects of human existence, encompassing work, education, training, leisure, travel, and more. This study categorizes AR applications into four main groups: "Training & Education," "Entertainment & Commerce," "Navigation & Tourism," and "Medical & Construction."

On the other hand, AR devices represent a technological revolution that has seamlessly integrated into our daily existence. These devices include displays, computers, tracking systems, and input devices, among other components. Each component contributes to the immersive AR experience, connecting the real and virtual worlds, that enables users to interact with digital content in real-world environments, which creates novel interaction, educational entertainment, and problem-solving opportunities. Additionally, software and hardware frameworks are pivotal in shaping the user experience. AR software frameworks provide the foundation for creating and deploying AR applications, offering tools and libraries to integrate virtual elements into the real world. On the other hand, hardware frameworks involve the physical components that enable AR interactions, encompassing devices such as smart glasses, smartphones, tracking sensors, and input interfaces.

2.2 Augmented Reality: Definition and Concepts

AR is Augmented Reality (en français: réalité augmentée) was first used in 1992 by Boeing researchers Thomas Caudell and David Mizell to describe a semi-transparent helmet used by aviation electricians that displayed virtual information overlaid on the real-world image.

The dictionary Petit Robert serves as the following etymological definitions:

- Reality n. f. (female noun) Lower Latin: realitas ->"nothing" Refers to the nature of what is real, that which is not merely a concept but rather an actual substance or fact. It refers to what is genuine, factual, and presented to the mind as such.
- Enhanced (v. tr.) Latin: augmentare -> "to increase." It means to increase the size, substance, or significance by adding something of the exact nature.

These definitions highlight the essence of "reality" and "augmented." Reality means something in the physical world, whereas augmentation refers to enhancing or adding to reality. In augmented reality, virtual elements are added to the real world, resulting in an enhanced and interactive experience that combines the physical and digital realms.

The first definitions of augmented reality (AR) were limited to using semi-transparent Head-Mounted Displays(HMDs) for visualizing virtual information overlaid on the real world. There are several definitions of AR [2] are provided. Still, the most commonly used is that offered by Paul Milgram in 1994(University of Toronto's Department of Industrial Engineering) and Fumio Kishino (Osaka University's Department of Electronics, Information Systems, and Energy Engineering): "Augmented Reality aims to enhance the natural feedback of the operator with the real world using virtual cues." Milgram et al. [2] proposed the virtuality continuum or reality-virtuality continuum concept as shown in Figure 2.1.

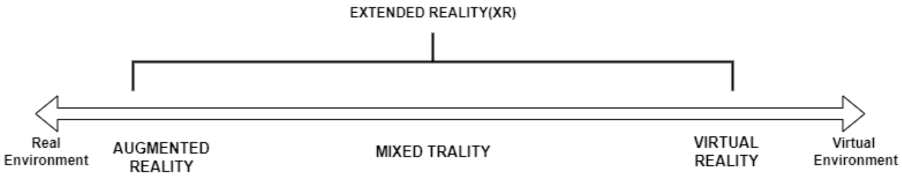


Figure 2.1: The Virtuality Continuum (Figure from [2]).

- **Real Environment (RE):** consists entirely of real objects without augmentation.
- **Augmented Virtuality (AV):** This category includes incorporating real-world components into a virtual environment.
- **Augmented Reality(AR):** AR technology integrates virtual objects into the real world, providing users with an interactive and immersive experience.

To create effective AR. The three requirements that follow must be fulfilled:

- **Combining real and virtual objects:** AR technology is designed to incorporate virtual objects into real-world environments. This requires using sensors and cameras to detect and monitor the user's surroundings, then augmenting the real world with virtual content.
- **Aligning real and virtual objects:** for virtual objects to appear seamlessly integrated with the real world. This is achieved by computer vision algorithms that analyze the surrounding environment and adapt the virtual content accordingly.
- **Providing dynamic interactions:** For augmented reality to be genuinely immersive, it must provide dynamic interactions between the real and virtual worlds. Users must be able to manipulate virtual objects.

In 1997, Ronald Azuma [48] developed the concept of Augmented Reality, which is characterized by the combination of real-world and virtual objects generated by computers in a manner that gives the illusion of coexistence within the same space as the physical reality. Thus, Azuma identifies three essential rules to define the AR system:

- Integrating virtual objects into the physical world.
- Real-time interactivity.
- Alignment (registration) of real and virtual objects.

This definition has faced criticism from researchers like Didier et al. [49] and Hugues et al. [50]. According to Didier [49], it ought to be enough for a virtual object to be semantically linked to a real-world environment, even without precise alignment, for it to be considered an augmentation. Therefore, it suggests removing the third condition proposed by Azuma et al. [48].

Fuchs and Moreau et al. [51] presented a broader definition of augmented reality, which encompasses a range of techniques that involve combining the real world with the virtual world, particularly through the integration of Real Images (RI) with Virtual Entities (VE), including computer-generated images, virtual objects, texts, symbols, diagrams, graphics, and more. Also, it mentions that other types of associations between the real and virtual worlds are possible through sound or haptic feedback.

According to Dubois et al. [52], augmented reality (AR) is an interaction paradigm that arose from the need for integrating computational processing capabilities with the real world. The goal is to eliminate the gap between the computer and the real world, allowing users to use computational resources while remaining connected to their environment. AR intends to extend the use of computers beyond their conventional framework (screen, keyboard, and mouse) and emphasize interaction with the user's physical environment.

Otmane et al. [53] defined augmented reality(AR) depending on its purpose: "The purpose of augmented reality is to enable one or more persons to have multisensory interactions (audio, video, and haptic) with an environment that coexists the two worlds, the virtual and the real."

Hugues et al. [50] defined an alternative term, "Augmented Perception." This new terminology transfers the emphasis from attempting to enhance an already complete reality to improving how individuals perceive and interact with their world. Employing the term "Augmented Reality " emphasizes how technology and digital elements can supplement and enhance human perception, as opposed to altering reality directly.

AR technology enhances and enriches human experiences by augmenting the natural environment or situations. This is made possible through computer vision, object recognition, and AR cameras inside mobile devices. These technologies make the surrounding real world interactive and manipulative, allowing users to engage with virtual content seamlessly and intuitively.

The association between the real and the virtual in augmented reality can vary based on the specific implementation. It may involve a semantic connection, where the virtual elements are meaningfully linked to the real-world context, adding valuable information and context to the user's experience. The fusion can also consider the precise alignment or recalibration of the real and virtual objects, ensuring a seamless and coherent integration. The choices made in the design and implementation of augmented reality systems profoundly impact user immersion in their augmented environment. A well-crafted augmented reality experience can make users feel fully immersed and connected to the blended world, elevating the overall quality of the interaction and user satisfaction.

2.3 Definition of Virtual Reality

In 1989, Jaron Lanier [54], the founder of VPL Research, coined the term "Virtual Reality" (VR) for the first time. However, the term presents a contradiction since something cannot exist simultaneously as real and virtual. A more suitable alternative could be "Real Virtuality," considering the outcomes of modern technologies. Alongside VR, related terms include "Artificial Reality" (credited to Myron Krueger in the 1970s), "Cyberspace" (introduced by William Gibson in 1984), and more recently, "Virtual Worlds" and "Virtual Environments" (from the 1990s). The foundations of Virtual Reality can be traced back to the mid-1960s, attributed to the pioneering work of Van Sutherland at the University of Utah [55]. Furthermore, Virtual Reality (VR) is a computer-generated simulation or representation of a three-dimensional environment or experience that a person can explore and interact with, typically through specialized electronic devices. VR aims to create a convincing and immersive digital experience that can mimic real-world scenarios or entirely fictional environments. The primary challenge of VR lies in creating a convincing and realistic experience for users. This involves ensuring that virtual objects appear real in various aspects, including their appearance, behavior, and the quality of their interaction with the user/environment.

To accomplish this level of realism, VR uses various multimedia technologies, including images, videos, audio, and text, which are utilized frequently in virtual reality experiences. Nonetheless, as VR evolves, it also investigates emerging media types such as e-touch (haptic feedback), e-taste (simulation of taste sensations), and e-smell (simulation of olfactory sensations). By incorporating these emerging technologies, VR developers can enhance users' sense of presence and immersion. For that, Heim et al. [56] defined the characteristics of VR using three key elements "I":

- **Immersion:** VR intends to immerse users in a digital environment, providing a sense of presence in which users feel physically present in the virtual world, employing immersive technologies, such as Head-Mounted Displays(HMDs) and spatial audio, which play a significant role in achieving this sensation of presence and realism.

- **Interactive Experience:** interactivity in VR is a crucial aspect that allows users to engage with the virtual environment actively. The computer system supporting the VR experience can rapidly update the scene and the user's perspective based on their physical movements and interactions. As users move or change their physical position, the virtual scene responds accordingly, providing a seamless and responsive experience. Interactivity enhances the feeling of agency, empowering users to influence and control their virtual surroundings.

- **Information Intensity:** Information intensity in VR refers to the richness and complexity of the virtual world presented to users. It encompasses the idea that a virtual environment can offer unique qualities, such as telepresence (the feeling of being present in a remote location) and artificial entities that exhibit a certain degree of intelligent behavior. VR can provide vast amounts of data and sensory inputs, enabling users to experience complex and intricate virtual scenarios that feel remarkably authentic and engaging.

2.4 Difference between Augmented Reality and Virtual Reality

Virtual Reality (VR) is regarded as a subset of Augmented Reality (AR), according to the research in [57] that first introduced Augmented Reality as the antithesis of VR. Whereas VR immerses the user in a fictional environment, AR aims to enhance the real world by integrating information processing capabilities. AR is a system that allows users to perceive the actual world and computer-generated data simultaneously. Unlike virtual reality, which isolates the user from the real world, AR will enable users to maintain their view of reality while incorporating computer-generated elements. Consequently, VR replaces the real world with a fictional reality, while AR enhances both the virtual and real world [58].

Table 2.1 provides an overview of the main distinctions between AR and VR technologies. Depending on the degree of reality, AR is composed of 75% real, while VR is 75% virtual and 25% real. Unlike VR, AR does not require a headset device. Moreover, AR enhances both the virtual and real worlds, whereas VR replaces the real world entirely with a fictional reality.

Virtual Reality (VR) applications encompass a wide range of uses and industries, taking advantage of immersive technology to create simulated environments in various domains, as on gaming [59] that allows players to immerse themselves in interactive and lifelike virtual worlds, enhancing the gaming experience. VR also enables users to watch movies, concerts, and other entertainment content in a more immersive way; Education

	Augmented Reality	Virtual Reality
Definition	overlays computer-generated content onto the real world	creates a fully immersive and simulated digital environment
Blocking out the real world	No	Yes
Device	Smartphones, tablets, laptops	VR headsets
Immersion	75% real-world	75% virtual-world, 25% real-world
Enhancement	Improvement of actual and virtual worlds	Replaces the real world with a fictional reality
Interaction	Touchscreen or voice commands	Controllers or body movements

Table 2.1: Differences between Augmented Reality(AR) and Virtual Reality(VR).

and Learning, Healthcare and Therapy [60] such as pain management, exposure therapy for phobias, and cognitive rehabilitation. It can also assist medical professionals in planning surgeries and visualizing patient data in three-dimensional space, architectural design [61] to enable architects and designers to create virtual models of buildings and spaces, allowing clients to experience and interact with designs before they are constructed; virtual tourism [62] due to VR that can transport users to different locations and landmarks around the world, providing a realistic travel experience without leaving their homes.

The work of **Slater et al.** [63] conducted a thorough and recent study on the primary applications of virtual reality (VR) as well as its strengths and limitations in various research domains. These domains include science, learning, instruction, health, social structures, ethical conduct, and the potential for application in various disciplines. Moreover, a study offered by Freeman et al. [64] focused on using virtual reality (VR) in mental health. This review demonstrated the effectiveness of VR as a tool for evaluating and addressing various psychological conditions, including anxiety, schizophrenia, depression, and eating disorders.

2.5 Augmented Reality Applications

Augmented Reality (AR) is a technology that blends the virtual and real worlds, superimposing computer-generated elements onto our physical environment. This integration has opened up many exciting and practical applications across numerous industries. AR has evolved rapidly in recent years, driven by advancements in hardware and software capabilities, making it more accessible and practical for widespread adoption. This section divides the AR applications into four major categories: Training & Education, Entertainment & Commerce, Navigation & Tourism, and Medical & Construction.

2.5.1 Education & Training

By seamlessly blending virtual content with the real world, AR offers immersive, interactive, and engaging experiences that significantly enhance the learning process and skill development. Educators and trainers can leverage AR to present complex concepts and practical skills in a more accessible and impactful manner. One of the critical applications of AR in education is through interactive learning materials such as textbooks, flashcards, and other educational materials, whereby students can readily understand the detailed concepts of physics, anatomy, astronomy, mathematics, and geometry as the work of Kaufmann et al. [3] used the Construct3D tool for learning mathematics and geometric concepts. This tool was built on the StudierStube framework, allowing for an interactive and immersive learning experience(Figure 2.2a). In MARIE(Multimedia Augmented Reality Interface for E-learning)(Figure. 2.2b) especially its application for engineering education. Its primary focus is the potential of AR by superimposing Virtual Multimedia Content (VMC) information in an AR tabletop environment [4].

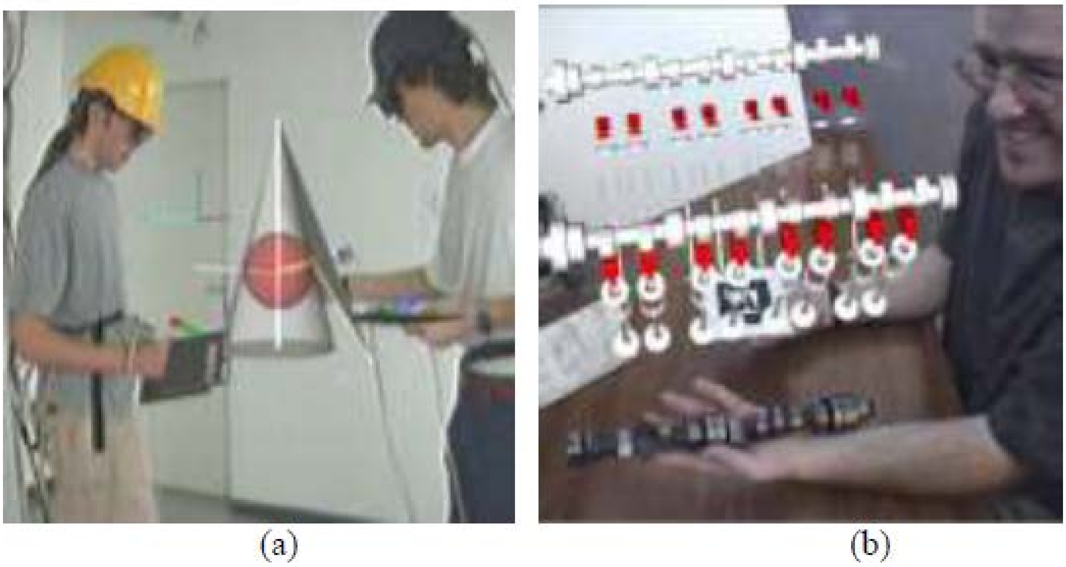


Figure 2.2: (a) using Construct3D tool [3] and (b) MARIE [4].

Students can use their smartphones or AR devices to scan textbooks and worksheets, unlocking additional multimedia content such as 3D models, videos, animations, and interactive quizzes, making the learning experience dynamic and captivating. AR Flashcards [65] serve as a valuable tool in utilizing AR technology to aid students in comprehending fundamental concepts such as the alphabet, colors, shapes, and space. SkyView [66] is a user-friendly stargazing application that leverages the smartphone camera to identify stars, constellations, and satellites in the sky. For Arloon Geometry [14] is an educational application that harnesses Augmented Reality (AR) to present students with three-dimensional models of various geometric shapes. Anatomy 4D [67] utilizes an AR system to showcase the human body, enabling students to delve into the various systems and study human anatomy. The Elements 4D [68] encourages students to engage in an interactive and immersive exploration of chemistry, covering topics from the basic elements to the complexities of life.

AR-based training offers a more engaging and practical learning experience, enabling trainees to practice real-life situations and learn from their mistakes without consequences. AR can guide technicians through vehicle repair and maintenance processes, highlighting critical components and providing step-by-step instructions. An AR system also provides military and defense training for combat scenarios, tactical exercises, and equipment operations. Soldiers can receive real-time information and situational awareness through AR devices.
Another application of The AR-based system, Hyundai Motor Group, has unveiled a

virtual guide [5] designed as AR technology to showcase the different components of the car's dashboard and engine area. With the virtual guide, users can intuitively receive step-by-step instructions to address minor car repair issues independently. From [5], the Volkswagen Mobile App MARTA allows users to access comprehensive information about their car, including its components and functions. Additionally, the application provides guidance on troubleshooting and resolving specific issues that may arise with the vehicle, as shown in Figure 2.3, which illustrates a snapshot from the Educational and Training AR application.

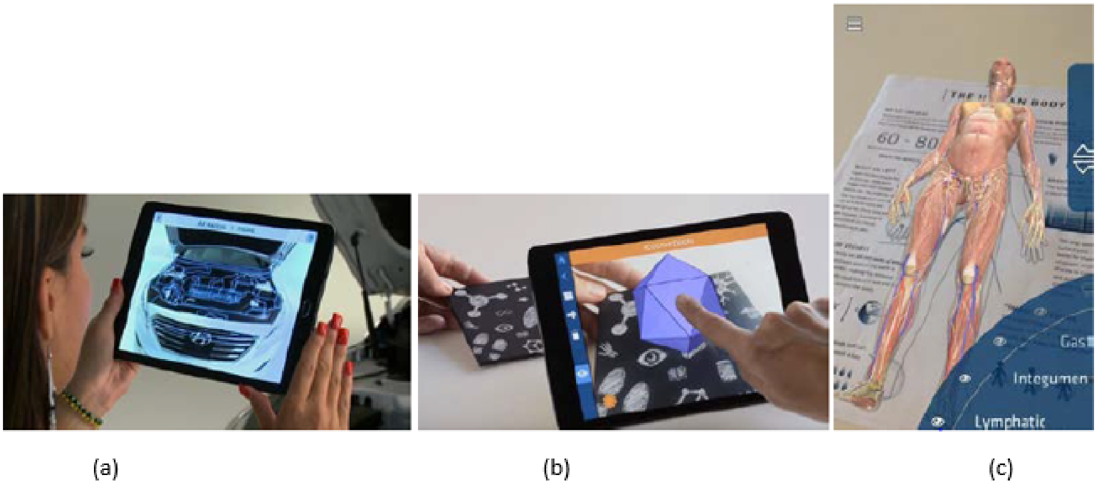


Figure 2.3: Screenshot of training and education application: (a) Hyundai Virtual Guide; Mathematics Arloon Geometry(b);(c) Anatomy 4D for human-body [5].

2.5.2 Entertainment & Commerce

AR utilized in the entertainment industry through the development of AR video games and the enhancement of the visibility of crucial aspects in live sports broadcasting like swimming pools, football fields, race tracks, and others are familiar and readily prepared, making them ideal for incorporating video see-through augmentation using tracked camera feeds. For instance, the Fox-Trax system [6] is employed in ice hockey broadcasts to highlight the puck's position, making it more visible to viewers as it moves swiftly across the ice. As shown in Figure 2.4, the ball trajectory is visually emphasized in golf broadcasts, providing viewers with a clear and dynamic understanding of each shot's path. Similarly, in football broadcasts, the first-down line is prominently displayed on the playing field, making it easier for audiences to track the game progress and comprehend the distance needed for a first-down conversion [69].

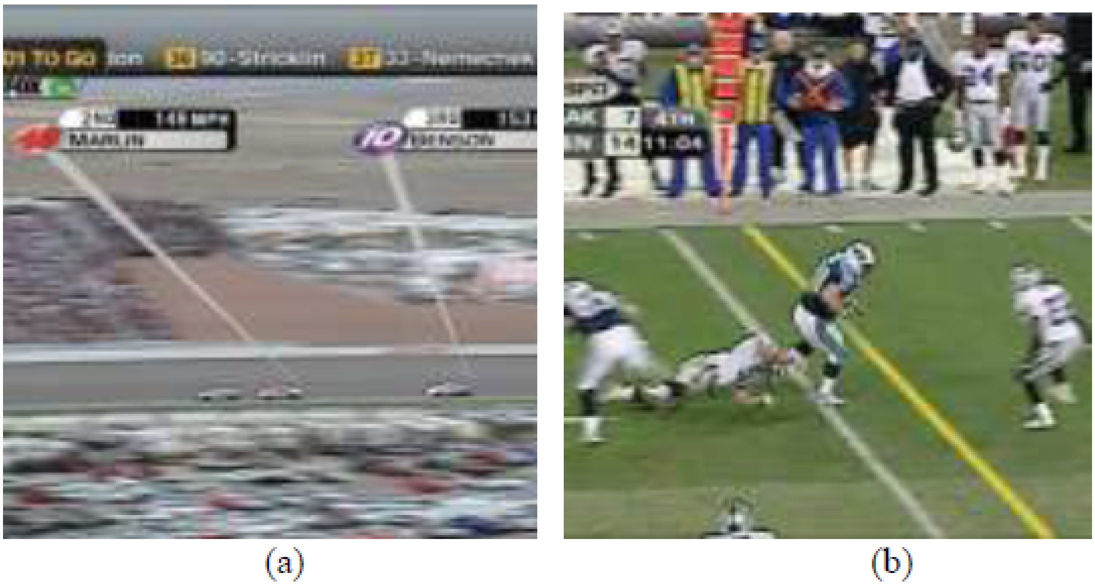


Figure 2.4: AR application for sports broadcasting:(a) racing and (b) football [6]

AR games provide immersive and interactive experiences by combining virtual and real-world elements. Recently, AR-based games have gained significant popularity, particularly with the introduction of the immensely successful "Pokémon GO" [7] game. Developed by Niantic in collaboration with Nintendo, "Pokémon GO" employs AR technology through smartphones, relying on GPS to track user location. Players are tasked with capturing Pokémon characters, which are superimposed into real-world locations, encouraging competition among players.

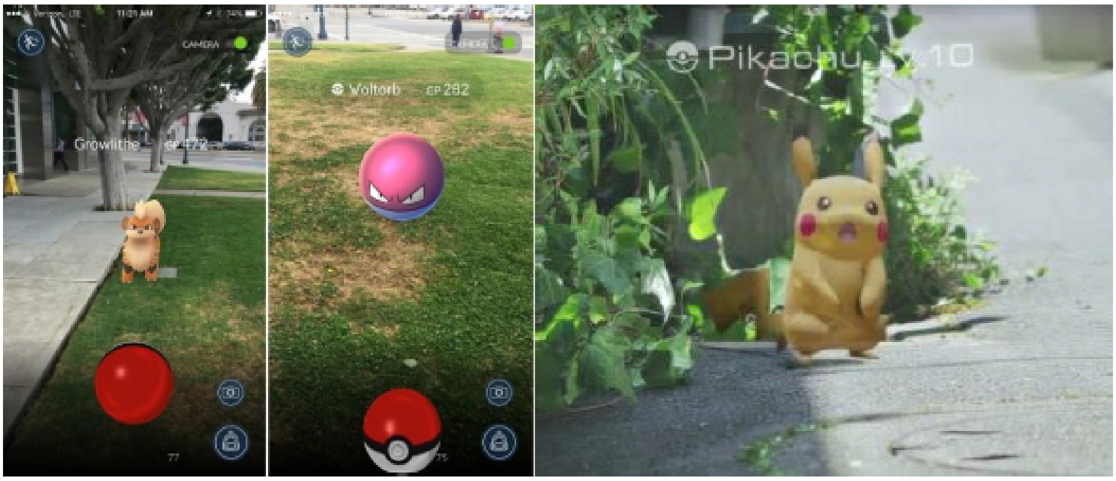
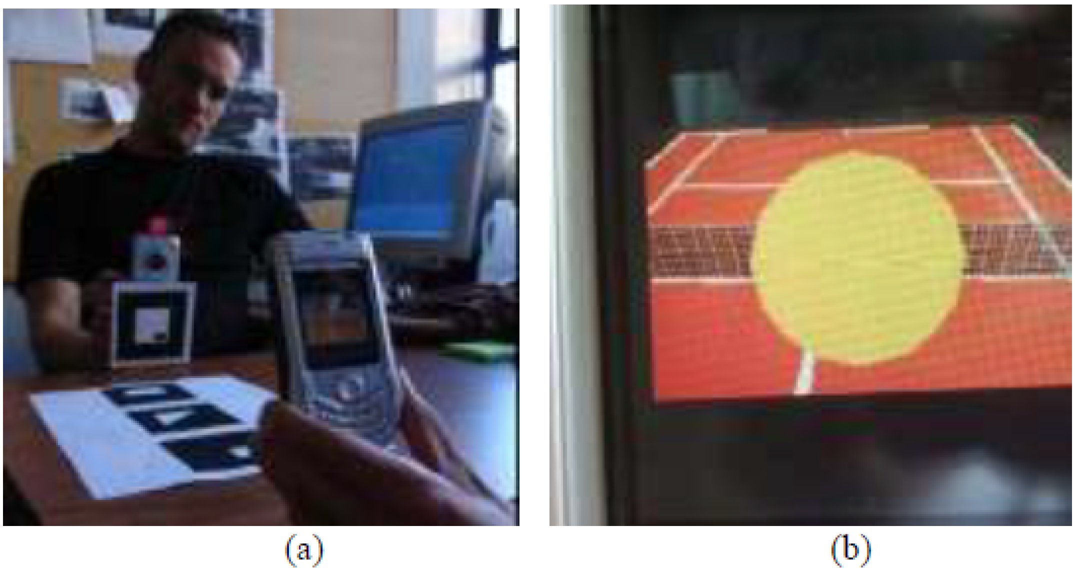


Figure 2.5: Pokémon GO application [7].

Further, "Real Strike" is a 3D first-person shooting-gun game developed by Yii International. The players experience an immersive military setting, with various sensitive support weapons available, all seamlessly integrated into their real-world surroundings using AR technology.
Several games have been developed specifically for controlled indoor environments, including the action-packed "Aqua-Gauntlet" [70]; players use AR technology to interact with the virtual aliens and engage in thrilling combat scenarios. The entertaining "Contact-Water" is where players juggle virtual dolphins indoors. The competitive "ARHockey" allows players to engage in virtual hockey matches indoors. The game simulates the sport, providing a competitive and interactive hockey experience, and the immersive "2001 AR Odyssey" [71], inspired by the classic movie "2001: A Space Odyssey".

In the case of "AR-Bowling," Matysczok et al. [72] conducted a study on the gameplay mechanics of AR bowling, which enables players to enjoy a virtual bowling experience in an indoor environment, interacting with AR-rendered elements to knock down virtual pins. In contrast, Henrysson et al. [8] created an AR tennis game for Nokia mobile phones (Figure 2.6). For "AR air hockey" [73], where players compete in a virtual air hockey match.


(a)

(b)

Figure 2.6: AR mobile tennis game where players used their phones as virtual rackets [8].

AR has become increasingly prevalent in commerce, offering various applications to enhance customer experiences and drive business growth. For example, AR enables customers to virtually try on products such as clothing, eyewear, makeup, and accessories,

allowing them to see how items look on themselves before making a purchase. This technology enhances the online shopping experience and reduces the need for physical store visits. To address the issue of furniture returns resulting from incorrect sizing for the intended location, IKEA implemented an AR-based smartphone catalog application.
With this innovative solution, users can virtually try out selected products in their homes using 3D virtual furniture. By using AR technology, customers can preview how the furniture will look and fit in their living spaces before making a purchase, reducing the likelihood of buying items that do not suit their needs.

Another AR application, EZface Inc. [74], introduced a virtual-testing platform for makeup products, allowing users to try on a wide range of their makeup offerings virtually. This AR-based platform enables users to see how the makeup products would look on their faces without needing to apply the makeup physically. This technology has proven to be a game-changer in the beauty industry, as it empowers customers to make more confident and informed choices when purchasing makeup products.

Furthermore, "Lululemon Mirror" [75], an athletic apparel brand, designed the "Mirror" feature in their app, enabling customers to try on different workout outfits using AR technology virtually. AR allows businesses to create virtual showrooms, showcasing their products in a digital space accessible from anywhere, enabling remote shopping experiences.

During the Christmas holiday season, Starbucks introduced the "Starbucks Cup Magic" [9] application, leveraging Augmented Reality (AR) technology to engage its customers uniquely and interactively. With this AR-based smartphone, Starbucks patrons could bring their coffee cups to life and experience an entertaining advertisement. This app cleverly blended the real-world coffee cup with virtual elements, creating a seamless and immersive AR experience for users. As demonstrated in Figure 2.7, there are some examples of entertainment and Commerce applications.

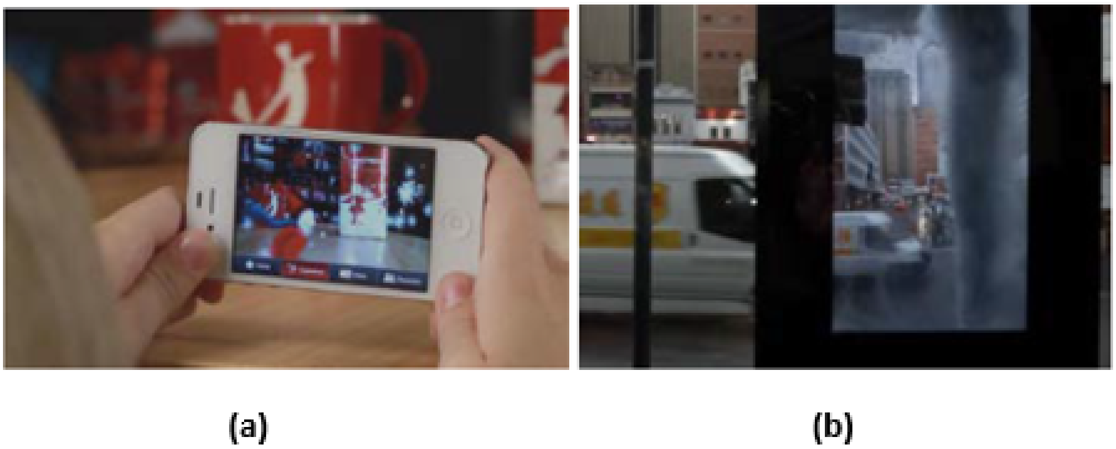


Figure 2.7: Screenshot of entertainment and Commerce applications: (a)Starbucks Cup Magic [9] and (b) Into the Storm.

2.5.3 Navigation & Tourism

AR navigation applications have emerged as cutting-edge tools that blend real-world environments with virtual overlays, transforming how users navigate and explore unfamiliar places. For drivers and travelers, new equipment has been explored to enhance the effectiveness of navigating in new areas, such as GPS technology. This integration allows navigation apps to determine the user's precise coordinates, enabling real-time tracking of their movements and providing accurate location-based services. AR navigation on smartphones offers real-time mapping capabilities, displaying interactive maps that users can zoom in and out of, pan, and rotate. These maps provide an intuitive and dynamic view of the surrounding area, making it easy for users to understand their location and plan their routes.

Indeed, AR proves to be exceptionally well-suited for navigation-based services, both indoors and outdoors. By leveraging AR technology, relevant information about user surroundings can be seamlessly inserted into a camera image, providing an intuitive and informative navigation experience based on user position and orientation. For indoor navigation

Starner et al. [76] explore the potential uses and constraints of AR in wearable computing, examining aspects such as finger tracking and facial recognition challenges. Another AR application, Field Trip [11], offers an extensive range of information about the current environment by displaying location-specific details on a card without requiring user interaction. It encompasses many interesting places and experiences, catering to a wide range of interests. Users can explore captivating content related to architecture, historic landmarks, significant events, local lifestyle, exclusive offers and deals, culinary delights, iconic

movie locations, captivating outdoor art installations, and lesser-known hidden gems of interest.
For outdoor environments and drivers, Narzt et al. [10] proposed a system that overlays various useful information directly onto their view, enhancing their navigation experience. Users can see routes, highway exits, follow-me cars, potential dangers, and real-time fuel prices all superimposed on their real-world surroundings. This information-rich display enables pedestrians to make informed decisions and navigate more efficiently, whether they are walking through city streets or exploring unfamiliar areas, as shown in (Figure 2.8(a)). Tönnis et al. [77] delve into the effectiveness of employing Augmented Reality (AR) warnings to draw a car driver's attention to potential hazards (Figure 2.8(b)), which aims to alert drivers to dangers and critical situations on the road.

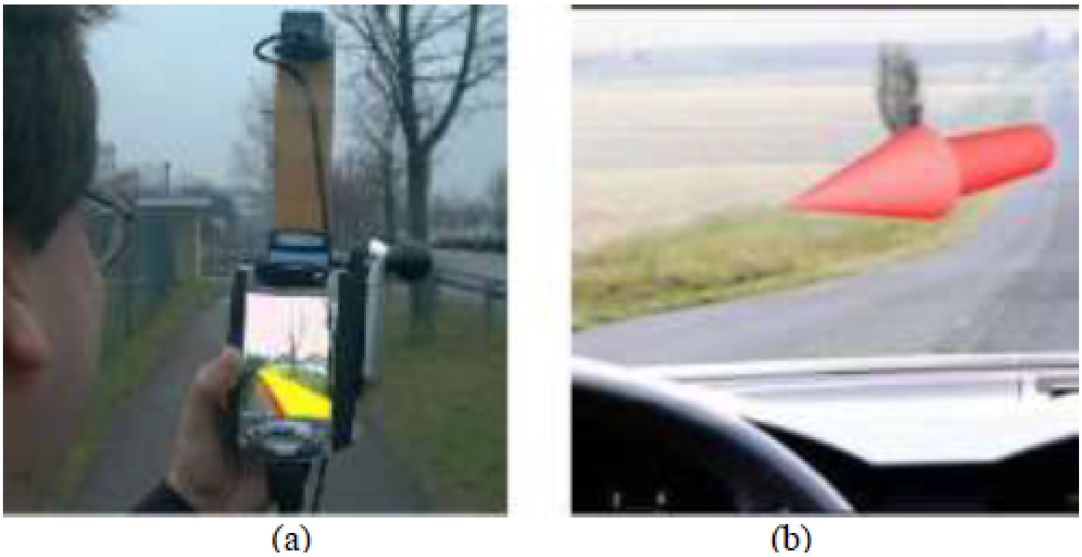
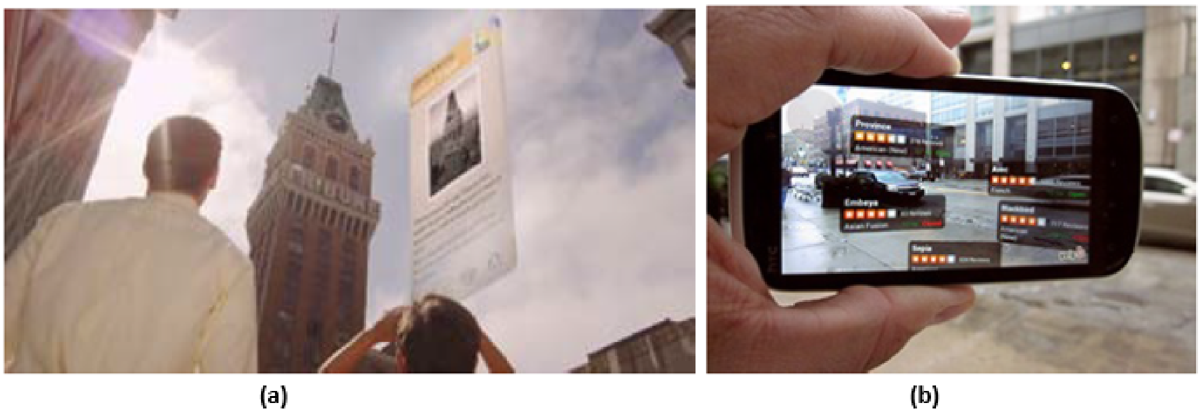


Figure 2.8: Navigation for pedestrians [10]and traffic alerts.

AR-based plays a significant role in ensuring a seamless and stress-free travel experience. It can provide real-time navigation guidance, helping tourists navigate complex transportation hubs, airports, or bustling tourist areas. In this context, travel applications have become invaluable tools for modern travelers, providing convenient access to a wealth of trip-related information at their fingertips. Travel apps provide interactive maps with GPS functionality, helping tourists navigate unknown locations, find directions to their destinations, and discover nearby attractions. The Metro AR Pro [78] is a mobile application that utilizes AR technology to enhance the user's experience with public transportation, specifically focusing on subway and metro systems. It used the Wikitude AR SDK that supports image recognition (identifying objects or images in the real world), tracking (tracking the user's position and orientation), 3D-model rendering (displaying 3D objects in the AR view), video overlay, and geolocation-related functions. The ARCHEOGUIDE [79], AR project is based on a cultural heritage on-site guide that is designed to provide visitors with cultural-heritage sites with valuable archaeological information. The project aims to enhance the visitor experience by leveraging AR technology to offer interactive and immersive content directly on-site. The "Augmented City" [80], an AR app, was proposed as a guide to enhance the tourist experience by providing information sharing and filtering functionalities. Diez et al. [12] offered an AR platform based on technology and mobile devices to provide an accessible and collaborative tourist guide. This platform aims to improve the tourist experience by providing a more engaging and interactive means to investigate and learn about their destination. Yelp Monocle mobile apps leverages AR technology to display nearby businesses and their reviews based on the direction the user points their smartphone camera. Users can see digital information superimposed on real-world surroundings using the smartphone's camera view. Figure 2.9 shows AR-based applications for tourism.


(a)

(b)

Figure 2.9: Some examples from AR-based tourism application:(a)Field Trip [11], and (b) Yelp Monocle [12]

2.5.4 Medical & Construction

Medical Augmented Reality (MAR) is a rapidly developing discipline integrating AR technology into numerous medical and healthcare domains. It provides novel methods for enhancing medical procedures, training, visualization, and patient care. For surgical Navigation, AR assists surgeons during procedures by providing real-time, 3D visualizations of the patient's anatomy. Surgeons can overlay important information, such as Computed Tomography scans(CT), Magnetic Resonance Imaging(MRI scans), or ultrasound imaging directly onto the patient's body, enabling more precise and minimally invasive surgeries. This would effectively give a physician an "X-ray vision" inside a patient. Consequently, it is significant during minimally invasive surgery. While these procedures offer benefits like reduced trauma and faster recovery for patients, they can limit a surgeon's visibility inside the patient's body due to the use of small incisions or no incisions at all. This reduced visibility can make surgical maneuvers more challenging and increase the risk of complications.

Sutherland et al. [81] introduced a novel human-computer interface called a "tracked Head-Mounted Display" (HMD) that enables viewpoint-dependent visualization of virtual objects, which has laid the foundation for modern virtual reality(VR) and augmented reality(AR) technologies. Other medical AR apps, including "EyeDecide" [82], aims to facilitate the rapid diagnosis of pathology by providing real-time, three-dimensional visualizations of medical data, such as X-rays and MRI scans overlaid onto the patient's anatomy. This allows medical professionals to gain a more comprehensive understanding of the pathology, leading to faster and more accurate diagnoses potentially improving patient outcomes.

Several projects are exploring this application area. At UNC Chapel Hill, a research group has conducted trial runs of scanning the womb of a pregnant woman with an ultrasound sensor, generating a 3-D representation of the fetus inside the womb and displaying that in a see-through HMD [13] (Figure 2.10).

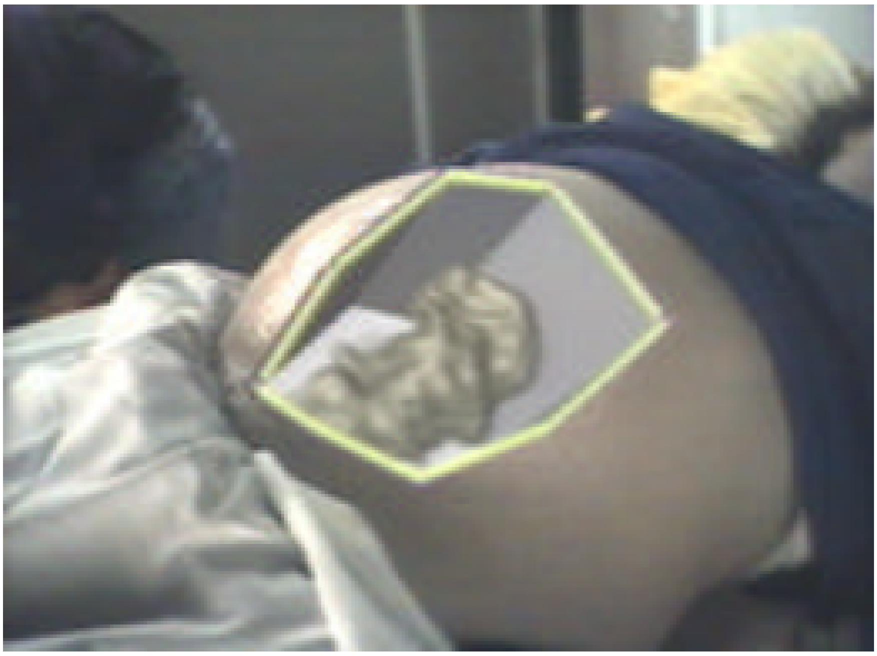


Figure 2.10: Virtual fetus inside womb of patient [13]

Vogt et al. [83] have employed video see-through Head-Mounted Displays (HMDs) to integrate Magnetic Resonance (MR) scans onto patients' heads. This advanced technology enhances surgical planning and intraoperative guidance, improving precision and patient outcomes. Similarly, Merten [2] has demonstrated the overlay of MR scans on feet using HMDs, showcasing the potential of augmented reality in podiatry and orthopedics, as shown in Figure 2.11. Additionally, Kotranza and Lok [84] observed that augmented patient dummies with haptic feedback elicited similar responses from medical specialists as they would with actual patients.

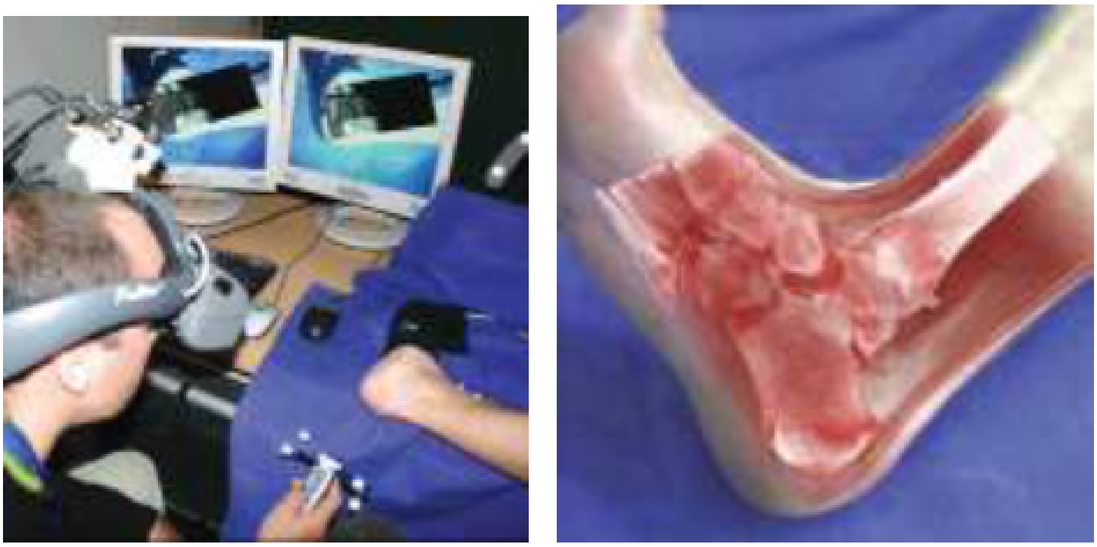


Figure 2.11: AR overlay of a medical scan [2].

AR is making significant strides in the construction industry, transforming how projects are planned, designed, and executed. It allows architects, engineers, and clients to view 3D models of buildings and infrastructure overlaid onto the physical construction site. This real-time visualization helps stakeholders understand the design's intended context, facilitating more informed decision-making and refinements. AR can play a significant role in the construction teams to identify clashes and conflicts between different building systems and components during the design phase. By overlaying virtual models onto the construction site, teams can detect potential issues early, leading to reduced errors and rework [85].

AR mobile application, CityViewAR [86] aims to offer users a unique glimpse into the pre-earthquake city of Christchurch. Also, it can explore the current cityscape while simultaneously witnessing the past by overlaying full-scaled 3D virtual models of buildings that once stood but were later demolished due to seismic events. Bentley Systems [87] has pioneered the development of a groundbreaking prototype application designed to revolutionize the visualization of underground infrastructure, which enables users to measure the distances between pipes within the underground network accurately. Furthermore, the application empowers users to incorporate 3D pipeline data using ground-penetrating radar technology. In [88], an AR-based application for urban planning eliminates the need for calibration and simplifies the AR setup process. The affine representation technique allows for the accurate overlay of digital content onto the physical urban environment, enabling urban planners and stakeholders to visualize proposed developments and changes in real time. Smart Reality [14]is a cutting-edge application that uses a smartphone camera and a printed target(Figure 2.12). Users can overlay 3D virtual objects onto the real world, and it is compatible with multiple devices, including Oculus Rift, a popular virtual reality headset, and the Epson Moverio BT-200 smart glasses.

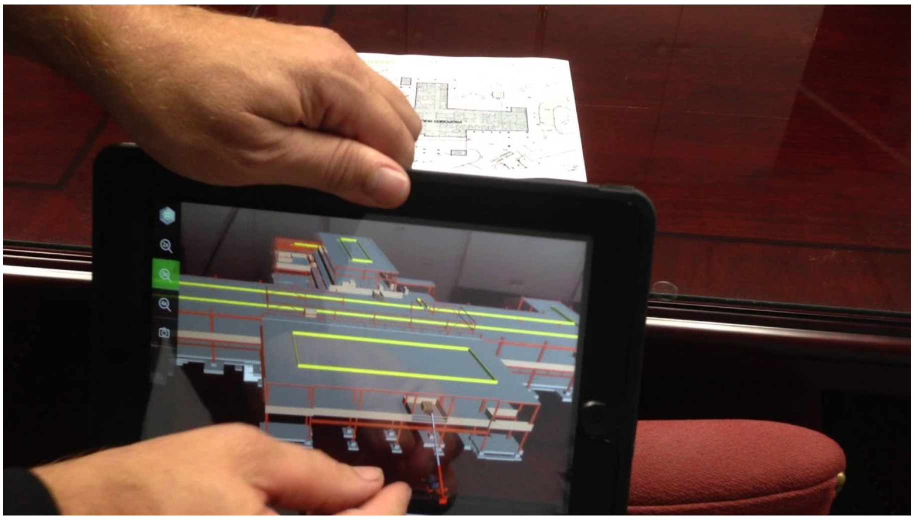


Figure 2.12: Smart Reality apps [14].

2.6 Augmented Reality Devices

Augmented Reality (AR) devices are technological devices that blend digital information and virtual content with the real-world environment to offer consumers an enhanced and interactive experience. These devices seamlessly combine the virtual and real worlds by superimposing computer-generated graphics, text, or images on the user's view of the physical environment. The components and primary devices for AR systems can be categorized into four types:

2.6.1 Displays

The display is a crucial component in AR devices as it overlaps virtual content onto the user's view of the real-world environment. Numerous approaches exist for presenting information to a person on the move, and a diverse range of display options can be utilized for this objective. These include personal hand-held devices(Smartphones and tablets), wrist-worn (smartwatches and wristbands), and head-worn displays(HMDs). Screens and directed loudspeakers integrated into the environment, as well as image projection on various surfaces, are also among the available choices. Moreover, several display techniques can be combined to create more comprehensive and immersive experiences. To present virtual content, there are five types of display:

- **Head-Mounted-Displays(HMDs):** is a device designed to be worn on the head, often in the form of a helmet or goggles, which presents a combination of real-world images and virtual content to the user's field of view (as depicted in Figure 2.13). It is mostly used for AR, VR, and MR applications and can be categorized into Optical and Video see-through displays. In an optical see-through HMD, the user looks through transparent lenses(such as Micro-Vision Nomad, TekGear Icuiti, or EyeTop) to see the physical world directly with their eyes. Virtual content is superimposed on the user's view using partially reflective or partially transparent mirrors. On the other hand, Video see-through displays with which the user wears cameras on the front of the device that capture the real-world environment. Then, it enhances the real-world view with virtual data and combines the altered images, making this method more computationally intensive.
  Furthermore, human factors like social acceptance and safety are also considered. A current trend in Mobile Augmented Reality (MAR) involves combining various display technologies with wearable computing, as noted by Hol [89].

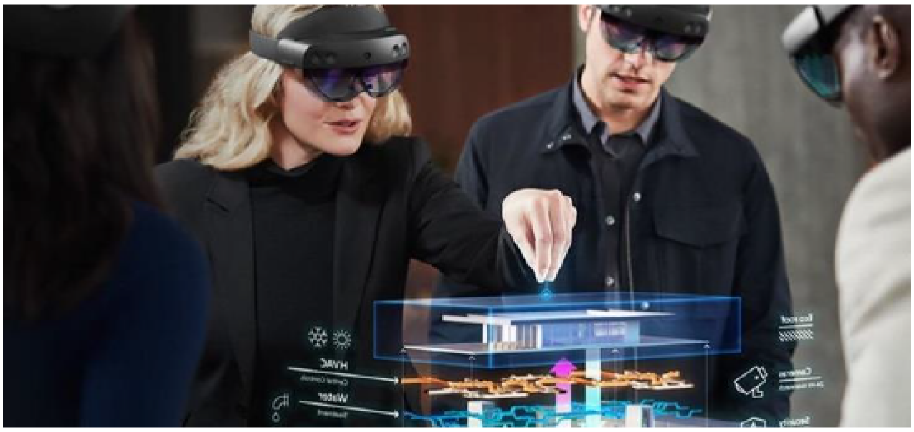


Figure 2.13: HMD Microsoft Hololens device used for augmented reality [15].

- **Spatial Augmented Reality(SAR)**, also known as Projection Mapping, is a technology that enhances real-world objects and scenes by projecting digital content onto them using video-projectors [90] or through holography, as seen with devices like the Microsoft HoloLens, creates virtual objects that appear to exist in the same space as the physical environment. Both methods depend on markless tracking, which aims to detect and track real-world objects and surfaces without needing predefined markers or fiducial points. Ridal et al. [91] presented the "Revealing Flashlight" project within the Cultural Heritage (CH) domain that offered projected displays in SAR. This project involves projection mapping to enhance the viewer's experience of cultural artifacts or historical spaces. On the other hand, holographic applications can expect new and innovative use cases for holographic AR to emerge across various industries. There are three main types of (SAR), each of which provides a unique way to augment the real environment: video-see-through, optical-see-through, and direct augmentation. In the video-see-through approach, real-world scenes are captured using cameras, and the captured video feed is then combined with virtual content. This composite view is then projected back onto the physical environment using projectors. Users can view both the real world and the overlaid virtual elements simultaneously. Spatial optical-see-through involves users looking through a transparent display, such as a visor or glasses, and optical holograms that allow them to see the real world directly.

Virtual content is superimposed onto the user view of the physical environment. This approach often requires precise alignment of virtual elements with the user's line of sight, which can be achieved through sophisticated tracking systems. It is not suitable for mobile applications because of the alignment challenges posed by spatial optics and display technology. Direct augmentation offers a seamless integration of virtual content with physical objects and spaces. It can be highly immersive and attention-grabbing, but calibration and environmental lighting conditions must be carefully managed. As mentioned in Figure 2.14.

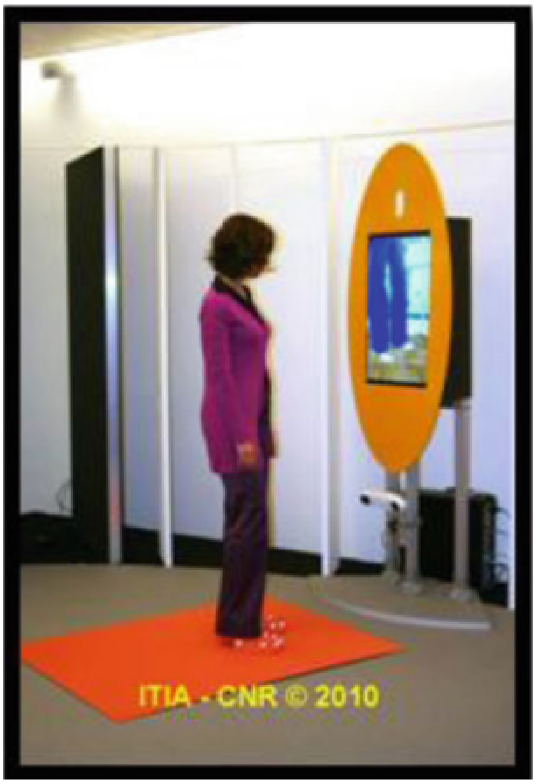


Figure 2.14: Example of Spatial Augmented Reality(SAR) application from [15].

- **Hand-held devices(HHD)** utilize video-see-through technique to superimpose graphical elements onto the actual surroundings. It incorporates sensors, including digital compasses inertial and GPS units, to facilitate six-degree-of-freedom tracking. Most AR projects in the (CH) domains employ Hand-held displays to merge the digital content over the real world as the work of [92, 93]. Additionally, tracking strategies such as fiducial marker systems, including ARToolKit, and computer vision technologies such as SLAM. There are three distinct categories of commercially available handheld displays extensively employed for augmented reality systems: smartphones, personal digital assistants (PDAs), and Tablet PCs [16]. Smartphones are highly portable and widely used, benefiting from recent technological advances that have equipped them with powerful CPUs, advanced cameras, accelerometers, GPS capabilities, and solid-state compasses. These features make smartphones an up-and-coming platform for augmented reality (AR). However, their drawback lies in their small display size, which is less than ideal for rendering intricate 3D user interfaces within AR experiences. (PDAs) offer many of the same advantages and drawbacks as smartphones. It shares characteristics and advanced features such as powerful CPUs, cameras, and accelerometers. However, PDAs have become less prevalent compared to smartphones due to the rapid progress made in recent times, especially with the dominance of Android-based phones and iPhones. These advancements have contributed to the decline in the popularity of PDAs. Smartphones are considerably more expensive and too heavy for single-handed and even prolonged two-handed use. Nevertheless, with the recent release of the iPad, we believe that Tablet PCs could become a promising platform for handheld AR displays. As demonstrated in Table 2. A comparison of different types of display techniques for Augmented Reality.

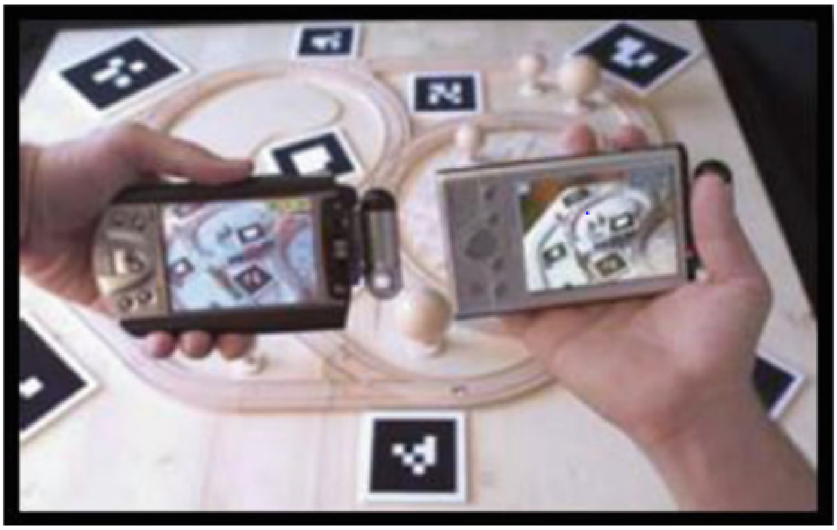


Figure 2.15: Example of Hand-Held displays from [16].

Types of displays	HMD		Handheld			Spatial		
	Video-see-through	Optical-see-through	Types of displays	HMD	Handheld	Video-see-through	Optical-see-through	Direct Augmentation
Advantages	Immersive Experiences, High-Quality Graphics, Visual content integration.	Improved Interaction with Physical Objects, Natural perception of the world, more comfortable, portable, ubiquitous high-performance CPU, camera, accelerometer, GPS	Wide Field of View, enable hands-free interaction, powerful	Larger displays with more expansive fields of view, enhancing immersion	Enhanced situational awareness, seamless integration with the physical environment, realistic visualization	Accurate visualization of real-world	Enhanced perception of surroundings, Real-time information overlay, Improved interaction with the physical world	
Drawbacks	Limited real-world interaction, Isolation.	Limited Graphics Quality, require accurate calibration and time-consuming	Small display	modest display, less widespread	Less common and more expensive, require specific software compatibility	Limited field of view, dependency on hardware, potential for motion sickness	Potential for visual distortion, does not support mobile system	Hardware complexity required, user discomfort

Table 2.2: Comparison between types of AR display.

2.6.2 Computers

AR systems often involve real-time processing of camera input, tracking user movement, and rendering virtual content seamlessly. This requires a powerful(CPU) and a significant amount of random-access memory (RAM) to handle the computational load. Historically, mobile computing setups have involved laptops integrated into backpacks. However, the advancement of smartphone and tablet technology and devices such as iPads indicates a potential shift from this cumbersome backpack configuration to a more sleek and sophisticated system. On the other hand, stationary setups can opt for traditional workstations equipped with powerful graphics cards to meet the computational demands of AR processing.

2.6.3 Tracking

Tracking devices and cameras are essential components in augmented reality(AR) and mixed reality(MR) systems to determine the user's position and orientation about the environment. For that, cameras have a crucial role in various applications, especially those that rely on marker-based or markerless tracking methods. Marker-based tracking involves detecting predefined markers in the environment, while markerless tracking uses computer vision techniques to identify and track features without relying on specific markers. In [94], the author outlined several tracking devices used for AR systems, mechanical, magnetic sensing, GPS, ultrasonic, inertia, and optics. Another CH application known as "AR-Teleport," developed by Kang et al. [93], uses the built-in inertial sensors and camera of a smartphone to monitor the user's location and position.

2.6.4 Input Devices

Various input devices are employed in augmented reality (AR) systems, as Reitmayr et al. [95] presented a mobile augmented system that incorporates gloves for interaction. Alternatively, ReachMedia [96] implements wireless wristbands. For instance, Google Sky Map on Android phones necessitates users to orient their phones toward celestial bodies they wish to identify, such as stars or planets. Nonetheless, selecting an input device should be contingent upon both the application domain and the system characteristics. The TOOTEKO AR application, as introduced by D'Agnano et al. [97], employs Near-Field Communication (NFC) sensors affixed to a 3D-printed reproduction of an artifact as an input mechanism. In the case of using AR systems for mobile devices, interaction and input mechanisms can harness various features, including the touch screen, microphone, and tracking sensors.

2.7 Hardware and software platforms for AR

2.7.1 Hardware Platforms

An important aspect of augmented reality (AR) systems is the integration of software and hardware platforms with the actual world, including the following devices: 1) webcam-equipped computers, 2) smartphones, 3) head-mounted displays, 4) spectacles, and 5) haptic devices.

The personal computer equipped with a webcam is the most extensively adopted platform for AR applications. Given the stationary nature of computers, a marker is positioned within the webcam's field of view, enabling a live video feed. A pertinent illustration is the Shiseido Makeup Mirror [98], which exemplifies the PC-based AR platform. This mirror features a touch-sensitive screen, permitting users to select from an array of eye colors, lip shades, and blush options. The kiosk is another example of a PC-based AR platform. The kiosk is a physical station that provides customers with the opportunity to utilize AR data to learn more about the products they carry with them. Personal computers are crucial in delivering interactive and informative AR experiences that accommodate user preferences and product exploration in these instances. The AR SandBox [99] initiative encompassed creating an integrated real-time augmented reality(AR) system for crafting physical topography models. These models are instantaneously scanned into a computer as they are formed. Using a projector, these scans of the models are subsequently employed as backdrops for an assortment of graphical effects and simulations.

Smartphones or tablets are arguably the most prevalent way to access AR content today. The two categories of smartphone usage are pervasive and continuously held. The Wikitude World Browser [100], one of the most user-friendly augmented reality(AR) tools, is illustrative of the ubiquitous category; information is displayed continuously with real-world images using a Web plug-in. The Microsoft omniTouch [101] is a wearable computer, depth-sensing camera, and projection system for interacting with common surfaces. For that, smartphone and tablet-based access to AR content has become a dominant mode, accommodating various usage contexts and preferences.

A Head-Mounted Display (HMD) integrates a screen with a headset, projecting information or images onto the user's view with six degrees of freedom (6 DOF). This augmentation aligns with the user's head movements in any direction or angle. An exemplary illustration is the SKULLY Helmet [102], notable for being the first helmet incorporating a built-in 180°Blindspot Camera and a Heads-Up Display. This combination delivers exceptional situational awareness and safety to users. Furthermore, the Microsoft

HoloLens [103] introduces an immersive experience where 3D models are seamlessly visualized within the real world. The HoloLens is a self-contained, holographic computer that allows users to interact with digital content and holograms within their real environment. It employs various sensors, advanced optics, and a dedicated holographic processing unit. The DAQRI smart helmet [102] is designed to enhance human capabilities across industries by establishing a seamless connection between users and their work environments. This helmet facilitates unparalleled interaction and engagement with the surroundings, contributing to improved performance and situational awareness.
The Google Glass [104] display is the most well-known AR-glasses product, with Vuzix [105] and other companies offering comparable products. Over time, technological advancements are likely to enhance the user experience and reduce expenses. The focus of Meta [106] is distinct from that of Google Glass. Meta method involves superimposing AR on the user's reality. The system recognizes user gestures, enabling users to manipulate 3D objects with a clay-like texture. Thus, Meta users are granted an almost infinite number of displays, as video content can be incorporated seamlessly into their AR experiences. The Icis [107] appears akin to regular eyeglasses, with no prominent component such as a visible camera. In contrast, the Atheer One smart glasses emphasize intuitive interaction, enabling users to control the device through hand gestures. This model incorporates dual displays—one for each eye—positioned in a manner resembling the placement of a landscape-oriented tablet measuring about 26 inches, situated immediately in front of the user's face.

Another AR device known as AR haptic allows users to immerse themselves in an augmented real environment enriched with synthetic haptic interactions, including the game-pads of Nintendo Wii U/PS4/Xbox [108], which furnishes users with real-time tactile feedback. Combining a (HMD) with the PHANTOM Stylus allows users to touch and feel visualized objects directly. In surgical contexts, integrating the VHB (Virtual Haptic Back) system [109] with the PHANTOM Stylus brings forth a remarkable application. Surgeons employing this setup can visualize human organs and entire body parts during surgery while simultaneously receiving tactile feedback through their fingers as they perform cutting, removal, and device manipulation procedures within the body.

2.7.2 Software Platform

Several software frameworks have emerged to facilitate the development of immersive reality applications. In this section, we present an overview of frameworks particularly well-suited for developed AR systems. The initial differentiating factor is the selection of the operating system (OS). This decision holds significance as not all available frameworks are compatible with the most commonly used operating systems. Further, the most widely commercial software development Kit(SDK) are Metaio, Vuforia, Wikitude, and ARToolkit, while PanicAR, DroidAR [110], and ARToolkit are available for free use.

Wikitude SDK, a commercial framework introduced in 2008, capitalizes on both location-based and vision-based tracking techniques. It has found application in various contexts, including a museum environment, as described by Caggianese et al. The Kudan [111] represents an exceptional AR SDK, standing out prominently from its counterparts. It supports highly robust single-camera Simultaneous Localization and Mapping(SLAM) technology. This empowers the SDK to deliver adaptable tracking capabilities, enhancing its ability to track targets in various scenarios effectively.

The Metaio SDK [112] represents a comprehensive software framework comprising components such as the Mobile, PC, Web SDK, Design, Creator, Engineer, and the Junaio [100] browser plugin. The fundamental vision driving the Metaio framework is seamlessly incorporating virtual elements into the physical world. The Vuforia SDK [112] is designed to facilitate the development of augmented reality applications specifically tailored for mobile devices. Vuforia is equipped to recognize a diverse array of elements, including intricate objects, images defined by the user, cylinders, text, boxes, and frame markers, all of which can be linked with cloud data. It is capabilities encompass an extensive spectrum of marker types, extending to compatibility with the Microsoft HoloLens, allowing for AR experiences to be crafted across an expansive range of devices and contexts.

ARToolKit [112] stands as an additional software development kit (SDK) designed to facilitate the creation of AR applications. This SDK employs square marker patterns, which are tracked through the position and orientation of a single camera. ARToolKit encompasses assistance for three primary tracker categories: natural feature tracking, traditional square markers based on templates, and 2D-barcode markers. It offers multi-camera capabilities and is compatible with the Windows Phone platform. Additionally, it excels in providing sturdy marker tracking across various distances, ensuring reliability and accuracy in augmented reality experiences. Layar stands as the most extensively employed option for location-centered services. Its capability to store Points of Interest(POIs) within a distant database and fetch relevant data according to the user's location renders this system especially suitable for outdoor navigation encounters. Table 3 compares the most widely used AR framework depending on appropriate characteristics.

SDK	Purpose	Platforms	Graphics	Tracking Sensors
Wikitude	Indoor, Outdoor	IOS, Android	Unity3D, 2D images, text, 3D models	Cameras, GPS, IMU
Vuforia	Indoor	IOS, Android	Unity3D, OpenGL, 3D models	Camera
Layar	Outdoor	IOS, Android	2D images, 3D models	GPS,IMU
AR Toolkit	2D images, Markers	IOS, Android	Unity3D, Android	GPS, IMU
Metaio SDK	Indoor, Outdoor	iOS, Android	Unity3D, Web	GPS, IMU
PanicAR	Outdoor	IOS	2D images, Labels	GPU,IMU

Table 2.3: A Comparison of the Most Widely Adopted Augmented Reality Frameworks

2.8 Conclusion

In this chapter, we have presented the foundations of augmented reality. After discussing the various definitions proposed in the literature and the etymology of the term, we advocate for both a conceptual and technological approach to augmented reality. Firstly, we introduced the purpose of augmented reality, based on observations of different systems, to enable individuals to engage in sensorimotor and cognitive activities within a mixed space that combines the real and virtual environment. Then, we define the most widely used AR application developed in various domains by employing AR devices that include displays, computers, tracking, and input devices. We conclude by providing the popular software and hardware utilized during the implementation. This chapter is considered as a comprehensive review that collects the necessary concepts overall in the AR domain.

Chapter 3

Handling occlusion in augmented reality: Literature review

3.1 Introduction

In this chapter, we focus on handling occlusion, which is a fundamental challenge in computer vision, and it has garnered significant attention from both classical and modern deep learning approaches. We introduce classical methods that employ object tracking, background subtraction, optical flow, and contour analysis to address occlusion issues by predicting object movements, segmenting foreground objects, estimating motion, and inferring shapes from visible parts. In the second part, we present a new taxonomy based on modern deep learning algorithms applied for hand pose estimation, particularly Convolutional Neural Networks (CNNs), which have revolutionized occlusion handling by enabling robust object detection and recognition. Recurrent Neural Networks (RNNs), Gated Feedback Networks, and Generative Adversarial Networks (GANs) extend the capabilities of deep learning in addressing occlusion issues. We present related works about hands. Benchmark datasets of depth images and 3D skeletal data collected for hand pose estimation are then presented. Further, evaluation metrics are provided to measure the effectiveness and robustness of the proposal methods. We review the main existing state-of-the-art approaches, which provide a methodology to tackle the problem of hand pose estimation.

Using bibliometric analysis, this chapter aims to define the hand pose estimation field and its parametric to understand their contribution as passive solutions for handling occlusion using the VosViewer program. Finally, we discuss the results of potential research direction regarding time speed, precision, and type of CNN architecture in this rapidly expanding field.

3.2 Classical Methods for Handling Occlusion in Augmented Reality

Within the realm of academic studies, numerous researchers have developed approaches to address real-time mutual occlusion challenges in augmented reality. As defined by Yuan Tian et al. [113], various techniques have been presented, broadly classified into three categories: contour-based, depth-based, and based on 3D reconstruction.

3.2.1 Contour-based methods

This method represents a fundamental category within augmented reality (AR) that is pivotal in addressing mutual occlusion issues, object recognition, and spatial interaction. These innovative techniques capitalize on the intricate details of object boundaries to enhance the user's AR experience.

These approaches begin with accurately delineating real-world object boundaries, often involving image segmentation techniques. Once the contours are identified, tracking and rendering mechanisms come into play, allowing for the dynamic superimposition of virtual objects onto the real environment. This integration is not just about visual fidelity but also about preserving the correct spatial relationships between objects. **Tian et al.** [113]introduced a segmentation technique to delineate the contours of actual objects. Subsequently, they tracked these contour entities in subsequent frames, enabling the generation of augmented images through pixel redraws to faithfully represent spatial relationships. Furthermore, they refined their initial framework by proposing an automatic occlusion handling method [114], which relies on computing the disparity map of real objects in the initial frame. It is worth noting that the efficiency of this method is dependent on the precision of contour extraction in the initial frame.

In contrast, **Fukiage et al.** [115] presented an innovative solution to the occlusion challenge that doesn't necessitate an exact foreground-background segmentation technique. Their approach considers the characteristics of human transparency perception, as observed in psychophysical experiments. This alternative method delivers real-time accuracy even in complex scenes.

Separately, **Sanches et al.** [116] introduced a method that empowers augmented reality environments based on fiducial markers. This approach facilitates mutual occlusion support between a real element and multiple virtual ones by leveraging fiducial markers. The method adjusts based on the element position(depth) within the environment.

3.2.2 Depth-based method

In this section, we delve into the realm of depth-based methods, which primarily revolve around comparing pixel values of real and virtual objects to unveil unoccluded regions of virtual objects, thus ensuring accurate occlusion representation As the research of **Schmidt et al.** [117], focuses on calculating dense disparity maps within stereo images. These maps are instrumental in detecting and managing mutual occlusion in augmented reality scenarios. Conversely, **Hayashi et al.** [118] employ a contour-based stereo-matching approach to acquire precise depth information of real objects, aiding in occlusion detection. The primary motivation here is to tackle challenges posed by the occlusion of a user's hands by virtual objects on an AR tabletop. However, this approach has limitations in accurately determining occlusion between real and virtual entities.

Additionally, **Setohara et al.** [119] propose a method aimed at detecting moving objects in front of a predefined marker pattern, using a pattern as a background image. Nevertheless, accurate occlusion resolution becomes impossible if the depth values of moving objects in the real environment are not obtained. Addressing this issue, **Kim et al.** [120] leverage stereo matching to estimate the depth of the real environment, facilitating accurate occlusion representation. This approach comes at the cost of computational intensity, making real-time processing a challenge.

Ohta et al. [121] introduce a novel "client-server" depth sensing approach, where clients acquire real-time depth data of the real environment from a server. While this method provides accurate depth data, discrepancies may arise if the viewpoints of both clients and servers do not align, affecting the quality of depth sensing from the client's perspective.

Furthermore, **Yokoya et al.** [122] utilize stereo matching for acquiring depth information of real objects. However, the utility of stereo matching is limited to specific regions, where virtual objects are rendered to optimize computational resources. The main challenge lies in its accuracy, as the boundary between the virtual and real environments may not be adequately modeled.

A distinctive method proposed by **Fuhrmann et al.** [123] leverages 3D models of real objects, referred to as "phantoms," to obtain models with properties similar to real objects. This approach excels when dealing with rigid objects, effectively mitigating occlusion-related issues.

Lastly, **Lu and Smith** [124] focuses on segmenting objects and calculating depths within areas covered by virtual objects. Their approach utilizes GPU-based methods to compute occlusion between real and virtual objects in real-time. This method becomes particularly useful when virtual and real objects exhibit independent motion.

Dong and Kamat [123] present a comprehensive framework that operates in two stages. Firstly, the framework employs a Time of Flight (TOF) camera for rendering, which yields depth and color buffers of real objects. Subsequently, the virtual object is redrawn with depth buffer testing enabled, allowing for the handling of occlusion, including parts hidden by other virtual objects or real-world entities.

3.2.3 3D Reconstruction-Based Approaches

This category presents a distinctive approach to resolving occlusion that revolves around the creation of three-dimensional models of real-world objects, enabling a detailed comparison of depth information between these objects and their virtual counterparts.

Fuhraman et al. [123] introduce a method tailored for static scenes, where the user is modelled as a kinematic chain of articulated solids. This unique approach simulates the occlusion effects of virtual objects within the static environment.

Ong et al. [123] propose a user-friendly framework for deriving a 3D model of the real scene. This is accomplished by leveraging recovered geometry information and user-segmented object silhouettes.

On a different note, **Lepetit et al.** [125] developed a method that empowers users to outline the boundaries of occluding objects in key views. The 3D occluding boundary is subsequently obtained based on re-projection and refinement in intermediate frames, ultimately achieving correct occlusion. This method faces challenges when handling viewpoint changes, which can result in inaccurate occlusion handling.

Yuan Tian [123] introduces a two-stage approach, comprising an offline and an online stage. In the offline stage, depth maps of the real scene are obtained using an RGB-D camera, with noise reduction applied to depth values. The online stage involves comparing the Z coordinates of real and virtual objects, enhancing real-time performance and yielding synthetic images with accurate occlusion.

Portales et al. [126] employ photogrammetry and AR to create high-precision 3D models integrated into the real world. This approach leverages close-range photogrammetry algorithms to construct 3D models, which are then seamlessly incorporated into the user's augmented environment through a see-through video head-mounted display (HMD).

Furthermore, **Carrion-Ruiz et al.** [127] present an architecture centred around reconstructing the 3D model of the Queen Victoria sculpture using photogrammetry. This model is incorporated into an AR application that allows users to observe the augmented scene through a window.

The basic concept of depth-based methods consists of the real-time computation of depth information for the actual scene. However, they require multiple cameras, which increases the cost and is best adapted for static scenes and specific perspectives. Contour-based methods, are cost-effective but may grapple with occlusion. The 3D reconstruction-based method offers efficient performance and effectively manages mutual occlusion, making it ideal for wide-angle viewing scenarios.

3.3 Research Questions

The objective of the present study is to collect and analyze all robust and sophisticated research that has investigated hand pose estimation methods. The extraction of salient features and the most important approaches will be proposed, and their properties will be discussed.

To accomplish the intended goals and find out the way chosen by researchers for their research and evaluation methods, research papers for which alternative techniques are introduced and benchmark datasets are discussed. Most researchers have examined QoS characteristics, proposed objective functions, and areas of focus that are crucial for designing these methods. These research questions (RQs) are listed as follows:

RQ1. What are the primary objectives of the researchers in the hand pose estimation field?

RQ2. What is the proposed strategy, and what methods are employed? How did the researchers conduct their research?

RQ3. What are the benchmarks or datasets that are used during training, and which case studies are taken into account?

RQ4. What evaluation metrics were designed to evaluate the results in each paper?

RQ5. What other study has been viewed to compare the performance of each paper?

RQ6. What are the most used challenges that are resolved in each paper by researchers?

RQ7. What is the research study that used the real-time concept in their proposed architecture?

3.4 Hand Recognition techniques

The human hand is an incredible tool capable of performing infinite actions. It is no wonder that numerous scientists have been intrigued by the modeling and simulation of human hands for various purposes. Despite the high interest and necessity, generating and perceiving hand motions have always been challenging due to two main problems:

- **Deformation of the Skin:** As joints move, the skin around joints and on the palm persists in deformation. Realistic animation requires special care to depict these deformations in simulations accurately.

- **Constrained Articulation of Joints:** The articulation of the human hand involves over 27 bones, resulting in roughly 27 degrees of freedom (DOF) even when excluding the DOF of the palm. Moreover, the movements of finger joints are intricately coordinated through constraints. These constraints ensure that only feasible hand configurations are achievable within the hand's configuration space.

Overcoming these challenges is crucial for creating accurate and lifelike simulations of hand movements. Researchers and developers in various fields, including robotics, animation, and virtual reality, have devoted significant effort to address these issues and capture the complexity and versatility of human hand motions. By finding innovative solutions to these problems, they aim to enhance the capabilities of technology to mimic and interact with human hand movements more realistically and effectively.

The bones within the skeletal structure combine to create a rigid body system, with joints that possess one or more degrees of freedom(DoF) for rotational movement [17]. These joints are named in the following manner, progressing from the wrist towards the fingertips:

- **Carpometacarpal (CMC) Joints:** joints that connect the metacarpal bones to the wrist.

- **Metacarpophalangeal (MCP) joints:** These joints are located between the fingers and the palm, enabling movement at the base of the fingers.

- **Interphalangeal (IP) joints:** joints are located between the segments of the fingers. They can be further categorized into:
  - **Distal Interphalangeal (DIP) joints:** These are the joints at the fingertips, connecting the middle and distal phalanges.

- **Proximal Interphalangeal (PIP) joints:** These joints are located between the proximal and middle phalanges of the fingers [17].

Gaining a comprehensive understanding of hand anatomy assists us in accurately representing the hand's spatial configuration. A kinematic model of the hand can be constructed based on its anatomical structure to capture its kinematic characteristics. The interphalangeal(IP) joints possess a single degree of freedom (1 DoF) for flexion and extension. In contrast, the carpometacarpal(CMC) joints are mostly treated as stationary, though those of the little and ring fingers exhibit some limited motion associated with palm folding. However, modeling the thumb presents challenges due to various considerations concerning the metacarpophalangeal(MCP) joint of the thumb, also known as the trapeziometacarpal(TM) joint. It can be depicted as a 2 DoF saddle joint, akin to the other MCP joints supporting abduction/adduction and flexion/extension, or possess solely flexion–extension capability (Figure 3.1).

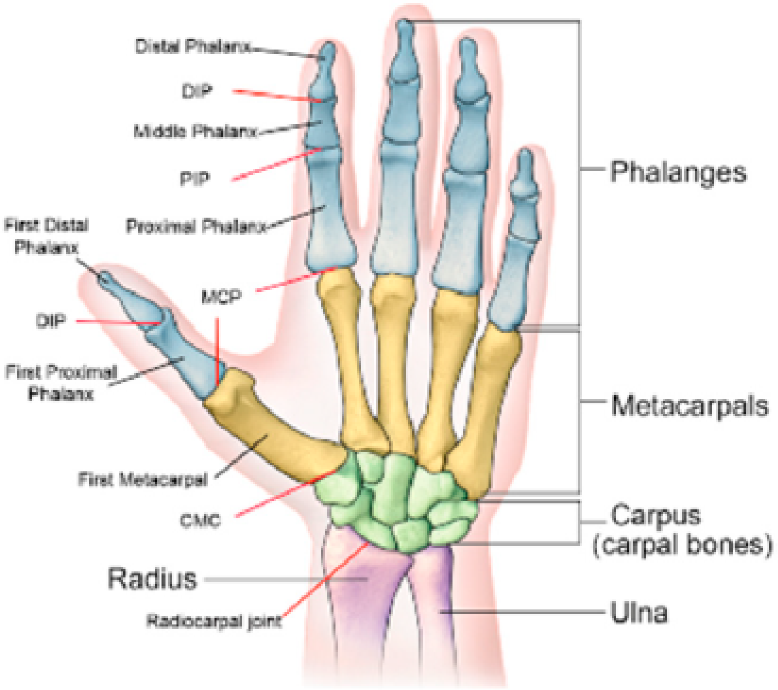


Figure 3.1: The skeletal structure of the hand observed from the palm-facing perspective [17].

By employing an analysis of degrees of freedom and the kinematic model, we can construct a feature vector that represents the configuration of a hand. In particular, the six degrees of freedom (DoF) frame associated with the wrist-hand joint is commonly called the "global configuration." In addition, the angular DoF for each finger is called the "local configuration," aggregated to form a feature vector, providing DoF hand pose estimation on a global scale. Constructing a high-resolution anatomical model can prove excessively intricate for numerous applications. As a result, various simplifications are proposed to maintain models at the necessary level of complexity [18].

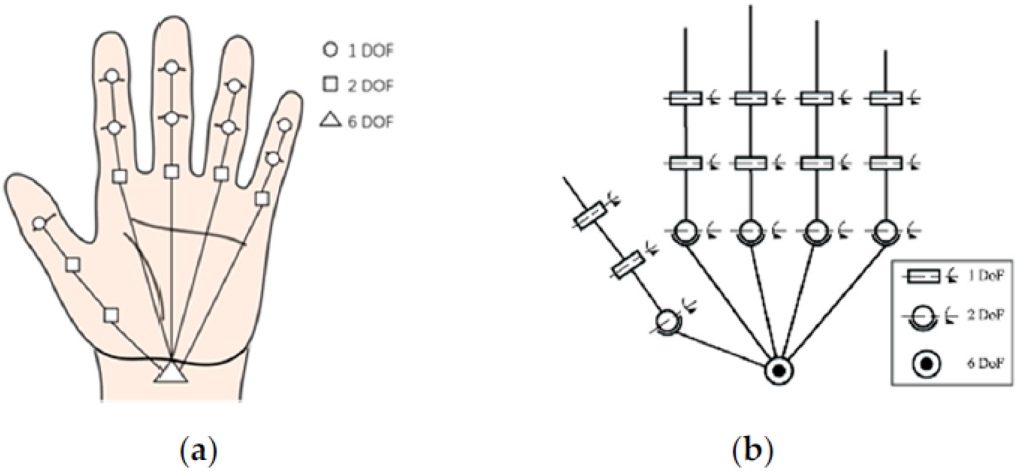


Figure 3.2: Common kinematic model employed for hand pose estimation. (a) A kinematic model with 27 DoF. (b) A kinematic model with 26 DoF [18].

for that, numerous research efforts have been dedicated to exploring the field of gesture recognition, aiming to harness the potential of human body movements as a means of interaction with technology. Among these investigations, various classification systems have been put forward, differentiating approaches based on the specific technology for capturing gesture data. The proposed classification framework, presented by La Viola in 1999 [128], provides valuable insights by categorizing gesture recognition into two distinct methodologies: sensor-based and vision-based approaches.

3.4.1 Sensor-Based Approaches

This category centers around utilizing a range of sensors to capture and interpret gestures. It encompasses diverse technologies, including accelerometers, gyroscopes, depth sensors, and wearable devices such as gloves or motion-tracking suits. These sensors detect and measure aspects such as motion, orientation, and acceleration, generating data that can be analyzed to identify specific gestures. These approaches include magnetic, acoustic, inertial, haptic(mechanical) sensors and touch surfaces.

3.4.1.1 Magnetic Sensors

Magnetic sensors utilize the low-frequency magnetic field emitted between a transmitter and a receiver to determine the position and orientation relative to the magnetic source.

The main drawback is the distortion of the magnetic field caused by metals.

3.4.1.2 Acoustic Sensors

Acoustic sensors transform sound into electrical signals. These sensors are relatively affordable, lightweight, and metal-compatible. However, their sensitivity to disturbance is a significant issue.

3.4.1.3 Inertial Sensors

This technology enables the calculation of rotation along all three axes by utilizing Earth's gravity and the user's movements. This type of sensor has recently been integrated into most smartphones, tablets, and other gaming consoles and devices. Katzakis et al. [19] used a smartphone with an inertial sensor to manipulate 3D objects within a virtual reality platform. As shown in Figure 3.3

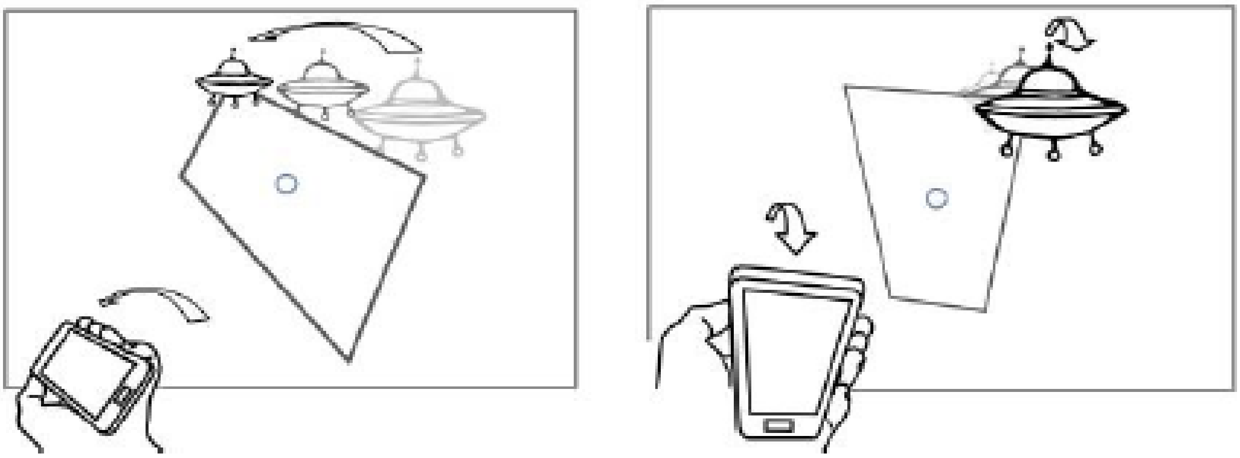


Figure 3.3: Manipulation of a 3D object on a screen using the inertial sensor of a smartphone [19].

3.4.1.4 Haptic devices

Haptic devices enable users to engage in tactile interactions with virtual objects, adding a sense of touch and physical feedback to digital experiences. This system provides the user with the perception of touch, like a touchscreen, and the sensation of 3D movement in space using a force feedback device.

Recently, Apple introduced a touchscreen that detects the degree of finger pressure. This technology, called "3DTouch," available on the "iPhone 6S," allows for 3D interaction on a 2D screen (Figure 3.4). Rabbi et al. [20] outlined the characteristics of various sensors, as presented in Table 3.1.

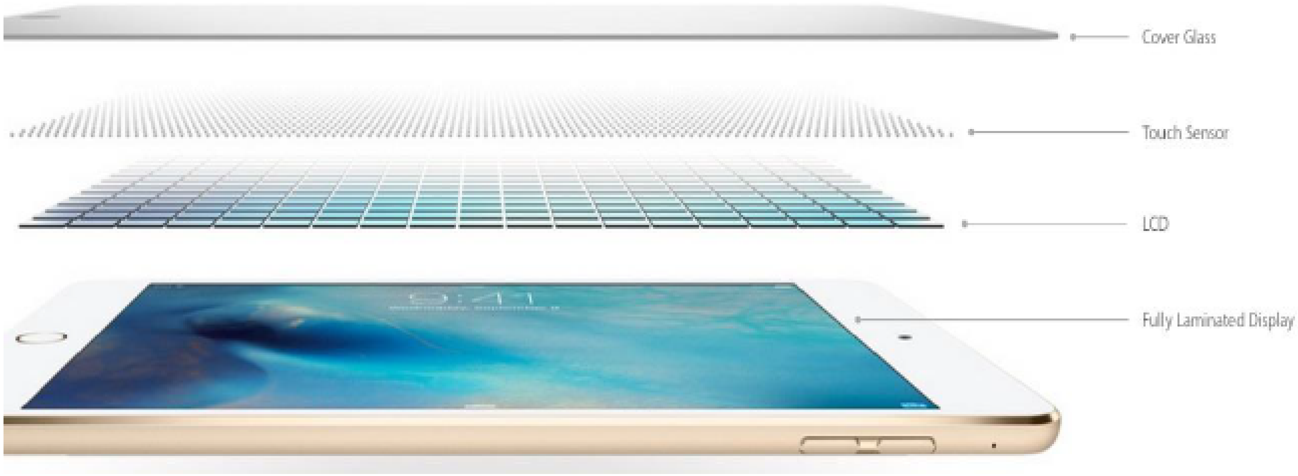


Figure 3.4: The principle of Apple 3DTouch technology [20]

Sensor-based tracking	**Accuracy**	**DOF**	**Cost**	**Advantage**	**Drawbacks**
Optical Sensor	Accurate	3/6 DOF	Cheaper	High Precision, Fast Response	Complex Data Processing, occlusion
Magnetic Sensor	Less accurate	6 DOF	Cheaper	Non-Contact, Cost-Effectiveness	Limited Sensitivity, Complexity
Acoustic Sensor	Less accurate	3/6 DOF	Cheaper	Low Power, no distortion, small	Occlusion, noisy environments
Inertial Sensor	Accurate	1/3 DOF	Cheaper	Portability, Highly Responsive	complex calibration
Hybrid techniques	Accurate	6 DOF	Costly	High accuracy, stable, Quality and Calibration	higher power consumption, expensive, high complexity.

Table 3.1: Summary of sensor-based-tracking characteristics [20]

3.4.2 Vision-based approaches

This type of approach involves determining the camera position using optical sensor data. These optical sensors can be categorized into three types: infrared sensors, visible-light sensors, and 3D-structure sensors. It is based on constructing a gesture's possible appearances from different viewpoints and conditions. Several representations have been proposed to recognize and model hand gestures. According to [21], two significant hand categories gesture representation exist: 3D models and appearance methods. As shown in Figure 3.5.

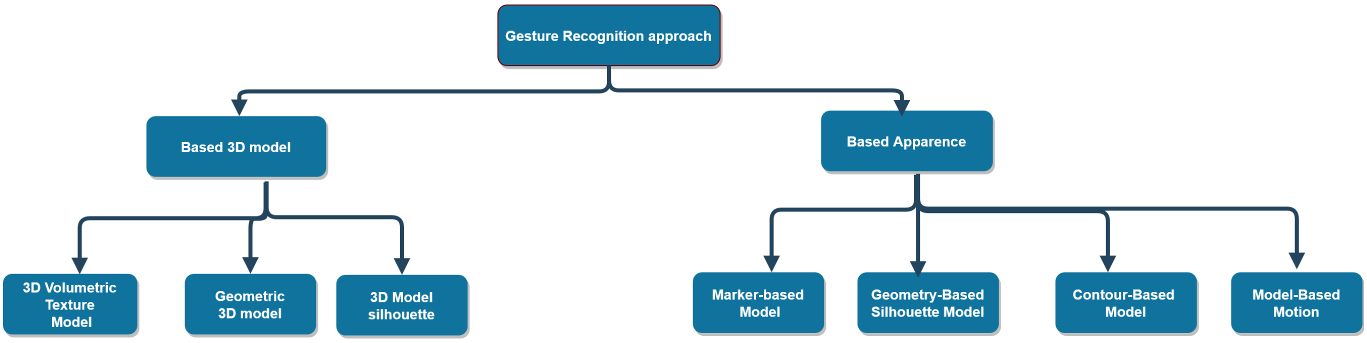


Figure 3.5: Hand Gesture Recognition Architecture according to [21]

3.4.2.1 Appearance-based method

The principal idea of this approach is to identify the marked hands with readily identifiable markers(such as gloves, colors, or coded targets) as proposed by Benbelkacem et al. [129] to interact with the system. The colored gloves are designed to improve the recognition of dynamic hand gestures by an RGB camera, allowing for natural interaction between the user and the virtual environment(Figure 3.6).

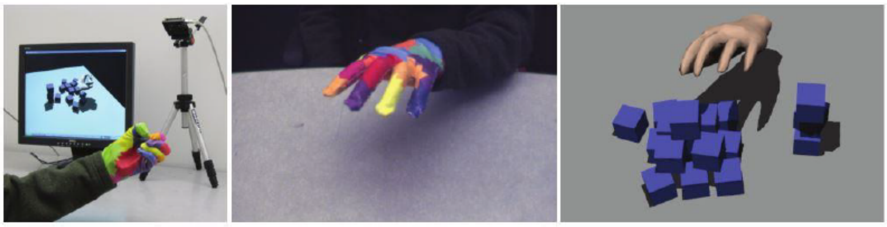


Figure 3.6: The principle of Apple 3DTouch technology [20]

Bellarbi et al. [22] introduced wearing small colored markers on the fingers of the user's hands (Figure 3.7) to manipulate digital documents projected onto a table. The markers are made from colored paper to avoid being cumbersome for the user. However, this technique enables the recognition of a limited set of static gestures for 2D interaction and exhibits certain anomalies when the colors worn are confused with those in the scene.

On the other hand, numerous approaches have been proposed to interact naturally with the machine using the hand without the need for the user to wear a marker or sensor on their hand. This is achieved by processing images captured by the camera. Some approaches have tackled the gesture recognition problem as a classification issue, employing classification techniques. In this context, Neural networks, inspired by the

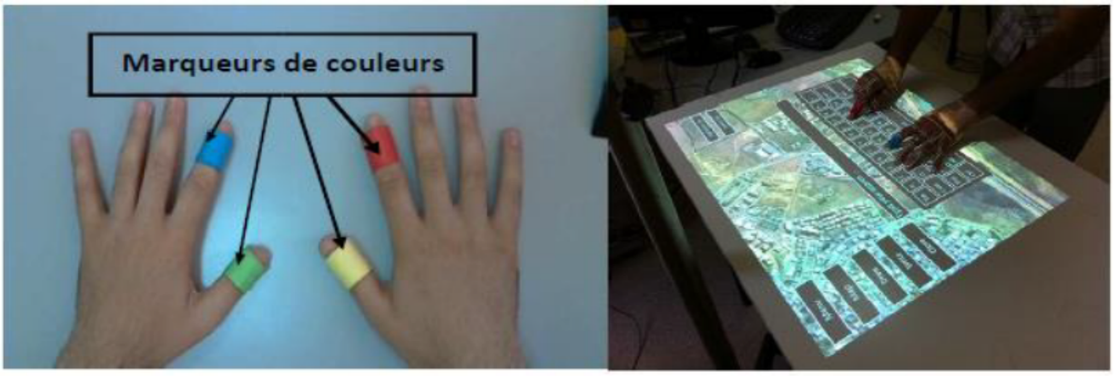


Figure 3.7: The proposed work of Bellarbi et al. [22] based on color marker interaction technique

human brain's architecture, have been widely adopted for gesture recognition. Most studies such as [130–132] have leveraged neural networks to classify and recognize gestures effectively. Hidden Markov Models(HMMs) are known as well for being able to capture temporal relationships. Nevertheless, it is important to note that some of these techniques may not be suitable for real-time scenarios due to computational complexity or extensive training requirements. Chen et al. [133] proposed a Support Vector Machine algorithm from machine learning that finds application in gesture recognition for crafting decision boundaries that effectively separate different gesture categories in feature space.

The Representative feature descriptors have been employed to recognize hand gestures, such as SIFT, SURF, LBP, BRIEF, and FERNs. The matching of inter-frame features is accomplished through tree-based searching, hashing, or the full-search method. Similarly, in the case of feature-based methods, the initial step involves identifying the matching relationship between 2D-image features, followed by the estimation of the features' 3D world-frame coordinates.

3.4.2.2 3D Model-based Techniques

Several approaches have been proposed based on depth-sensing cameras, such as Microsoft Kinect and Asus' Xtion, allowing a more accurate understanding of the spatial relationships between objects and gestures(Figure 3.8).

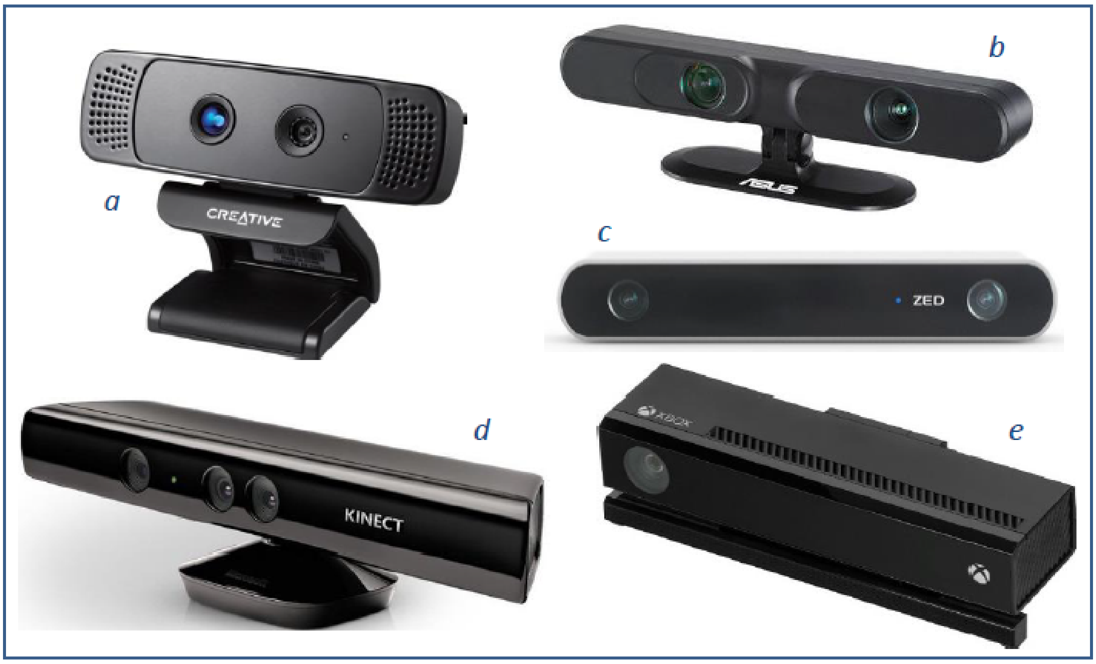


Figure 3.8: Some examples of depth-sensing Cameras: a) Creative Caméra1. b) Xtion Pro. c) Zed caméra. d) Kinect. e) Kinect V2

Wang et al. [134] introduced a 3Gear technique for gesture recognition approach that capitalizes on the capabilities of the Kinect sensor. The core idea behind 3Gear is to enable the recognition of user gestures by comparing them to a pre-defined database of known gestures.

Further, Messaci et al. [23] employed a technique to interact within a virtual environment for training technicians in the maintenance field.

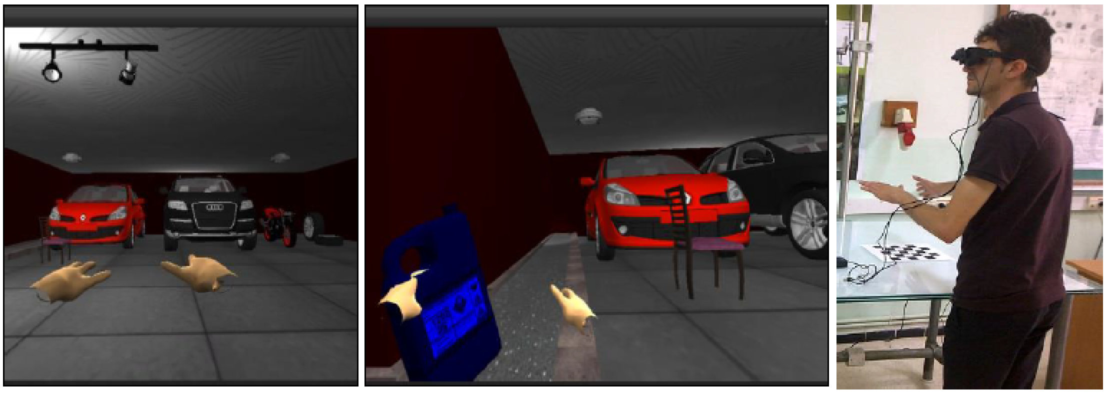


Figure 3.9: Example of a user navigating within a virtual environment proposed in. [23]

Recently, most research has experienced a new level of robustness and accuracy due to the advancement of deep learning techniques as the work of [135, 136]. These techniques

demonstrate effectiveness and robustness for 3D hand gesture recognition. However, the captured devices require a fixed user position and do not offer complete mobility.
In October 2012, Holz et al. [24] developed the "Leap Motion," a new device for hand gesture recognition and tracking. The device consists of software components and a hardware controller. The controller, measuring 13 mm x 30 mm x 76 mm, is equipped with two infrared cameras (Figure 3.10) and can detect and recognize hands within a range of up to 50 cm.

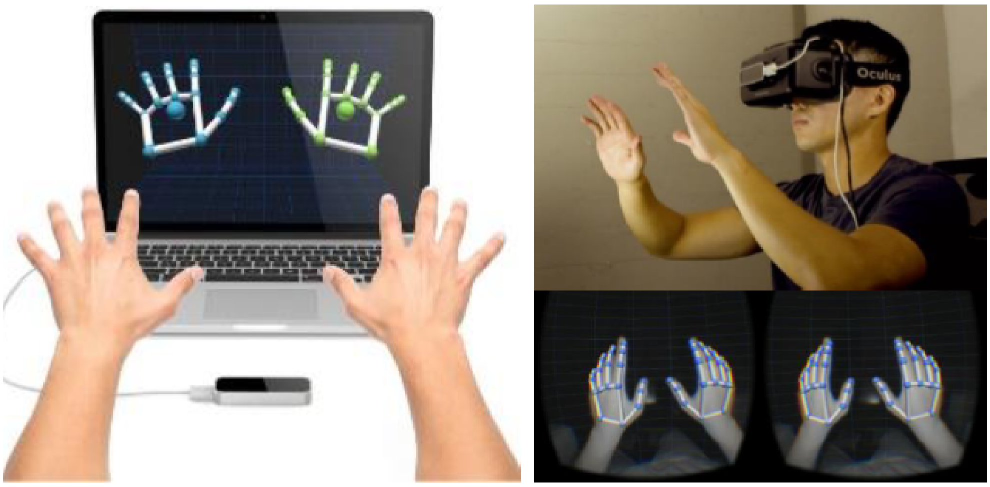
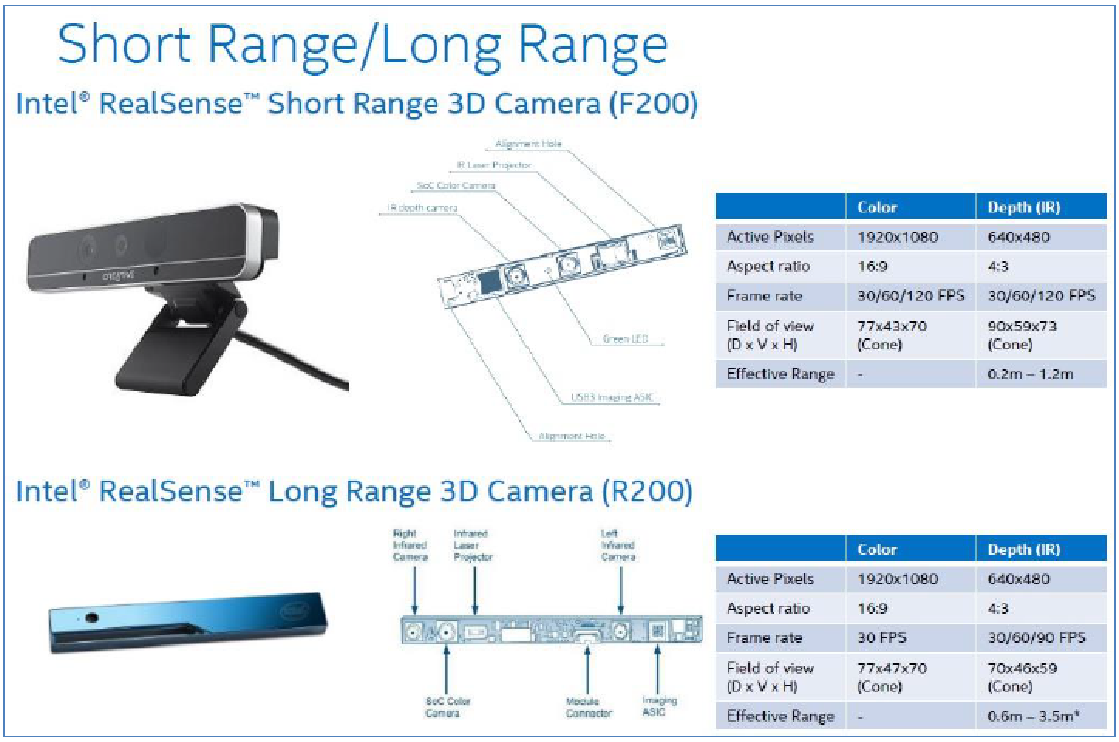


Figure 3.10: Leap Motion devices [24].

Intel's RealSense technology, introduced in July 2016, is a cutting-edge sensing solution designed to enable depth perception, motion tracking, and gesture recognition. The RealSense device is compact and incorporates three distinct types of sensors(RGB camera with resolution 1080p; Infrared camera; Infrared Laser Projector) as shown in Figure 3.11. Combining these three sensors allows the RealSense camera to capture both the visual appearance of objects and their spatial position. This makes it particularly well-suited for applications requiring gesture recognition, object tracking, hand tracking, facial recognition [24].



Short Range/Long Range

Intel® RealSense™ Short Range 3D Camera (F200)

	Color	Depth (IR)
Active Pixels	1920x1080	640x480
Aspect ratio	16:9	4:3
Frame rate	30/60/120 FPS	30/60/120 FPS
Field of view (D x V x H)	77x43x70 (Cone)	90x59x73 (Cone)
Effective Range	-	0.2m – 1.2m

Intel® RealSense™ Long Range 3D Camera (R200)

	Color	Depth (IR)
Active Pixels	1920x1080	640x480
Aspect ratio	16:9	4:3
Frame rate	30 FPS	30/60/90 FPS
Field of view (D x V x H)	77x47x70 (Cone)	70x46x59 (Cone)
Effective Range	-	0.6m – 3.5m*

Figure 3.11: Intel RealSense cameras [24].

3.5 Deep learning hand pose estimation methods:

Hand pose estimation is an intriguing research topic in numerous applications, which has progressed rapidly using deep learning. Many researchers have started to apply it to computer vision, and hand pose estimation has been one of them.

Based on previously available statistics (around 60 surveys in the period 2015–2019 [137]) that prove the exciting topic of hand pose estimation throughout the progress of deep learning techniques. So, we conducted a new taxonomy based on deep learning, which is grouped into three categories depending on the input data: (a) Depth-based, (b) Image-based, and (c) RGBD-based methods.
According to this taxonomy, each category contains two sub-categories; the third doesn't have a sub-category. The visualization of the proposed classification can be observed in Figure 3.12.

3.5.1 Depth-based methods

As mentioned previously, this section has been divided into two sub-sections. The first one uses 2D depth maps, and the second uses 3D representations to predict the 3D location

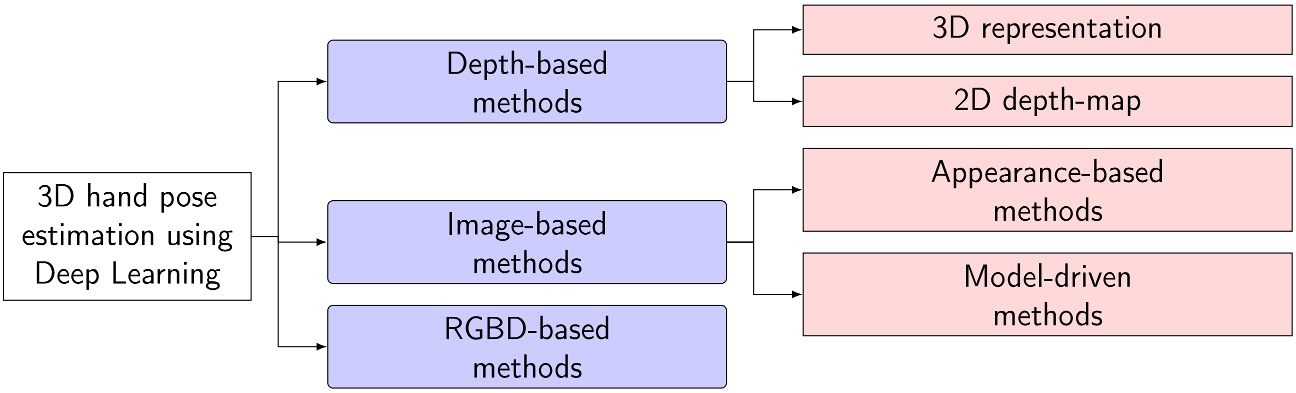


Figure 3.12: The proposed taxonomy for 3D hand pose estimation

of each keypoint on the hand.

3.5.1.1 2D depth maps

They employed the depth information of the image as an input modality for predicting the 3D hand joint locations. In one of the oldest published works, **Tompson et al.** [138] created accurate datasets of labeled ground-truth of hand gestures in an automatic process depth image using a Linear-Blend-Skinning (LBS) hand model. Then, it used two stages of a CNN architecture to extract dense features. This network employs heatmaps of each joint and uses inverse kinematics (IK) to improve joint predictions.

Oberweger et al. [135] used different convolutional network structures to determine the 3D joint location of the hand. The information is used as a prior model of hand pose to add it to the network, which is a bottlenecking linear layer with fewer neurons that can predict the position of the joint through depth imaging but doesn't need an IK stage. The same authors, **in** [139], improve the previous architecture by adding residual networks (ResNet) into the network. They introduce data enhancement during the training and hand localization by using a CNN detection refinement network to get better performance.

Zhou et al. [140] designed a novel network called the Hand Branch Ensemble Network (HBE). The three branches of the network correspond to three parts of a hand: the thumb, the index finger, and the rest of the fingers. The features of each branch are grouped and projected to a low-dimensional layer by a similar bottleneck layer to ensure the 3D coordinates of the hand joints.
Further, **Du et al.** [141] developed a pose regression network called (CrossInfoNet). The network decomposes the hand into two main parts: palm pose estimation and finger pose estimation. The network used heatmaps as constraints when getting structure from the

2D depth maps to extract features. **Sinha et al.** [142] presented 18 joints and 21 DOFs of the hand using a regression method that took a cropped depth map image as input and produced a low-dimensional activation feature that represents the global or local joint parameters of a hand pose and utilizes a matrix completion to output an activation feature that is synchronized with the other features in a population database.
At the end, **Wan et al.** [143] proposed the "Crossing Nets" architecture aiming to learn statistical relationships between 3D hand pose and corresponding depth images using two generative deep learning GANs (Generative Adversarial Networks) and VAEs (Variationnal AutoEncoders) to design the latent space of depth images to the latent space of 3D pose. The first section (VAEs) generates the latent space, resulting in the distribution of Zy. In the second part, GANs are sampled from the distribution of the latent space of depth images Zx. By adding a fully connected layer to map the two latent spaces. The network matches the results of GANs' ability to generalize and the parts of VAEs that are in charge of learning the pose and improving the pose estimation.

3.5.1.2 3D Representation

This sub-category groups all recent approaches used as input: 3D point clouds or 3D voxels, as in the work of **Ge et al.** [144] used an intrinsic camera to transform the 2D depth images into 3D point clouds and project them onto three orthogonal planes to obtain several views of each hand pose. They regressed 2D heatmaps from each projected image using three identical CNNs to estimate the joint of each plane. All are combined to get the final 3D hand pose.
On the other hand, **Qe et al.** [145] designed hierarchical PointNet that takes the point cloud as input and extracts the hand's main features hierarchically, allowing robust generalization. Besides, they normalized the sampled 3D points in an oriented bounding box(OBB) by applying a Principle Component Analysis (PCA) on the voxel's coordinates and selecting the best three components. They rendered 3D shapes on those planes to make the network more consistent and to learn 3D hand articulation easier. Ultimately, they fed the depth maps into the network and obtained the probability for each joint in each plane: xy, xz, and yz. Thus, they fused the probabilities by multiplying them together.
In later work, **Ge el al.** [146] designed a 3D CNN architecture to estimate 3D hand pose in real-time, which takes a 3D volumetric as input to regress directly to 3D joint locations. Specifically, they defined the three 3D shapes with the Truncated Signed Distance Function (TSDF). In TSDF, the distance of the voxel to the closest surface will be stored in each voxel. Then, it fed into 3D CNN and three fully connected layers.

Similarly, **Moon et al.** [147] developed a detection-based, voxel-to-voxel network(V2V) that accepts as input a voxelized grid (3D voxelized depth map). Since the input and output are in 3D, the proposed model is based on a 3D CNN encoder-decoder network that performs all convolution and deconvolution operations in the 3D domain. The idea is taken as input depth maps from different angles of a monocular hand with the same 3D pose. To approximate the 3D hand pose, the model must be trained from different depth map inputs to produce the same pose. Therefore, a 3D point cloud has precisely one feed into the network that contains three 3D CNN layers and three fully connected layers. Also, training the network on the 3D point cloud of the hand directly to generate the 3D pose via a 3D encoder-decoder is more accessible than massive datasets containing all hand shapes.

After the accomplishment of residual blocks of ResNet network [148] in object classification, **Moon et al.** utilize ResNet blocks with a deeper network to optimize their network performance. It outperformed previous techniques on well-known depth-based hand datasets and was first placed in the HANDS 2017 frame-based 3D hand pose estimation competition.

Likewise, **Huang et al.** [149] defined a structure-aware 3D hourglass network for hand pose estimation. It takes a normalized binary voxel from a depth image as input and produces regressively 3D heatmaps for each joint. By using the hourglass as a fundamental building block and adding additional hand skeleton constraints, the task of estimating the hand is more accurate, and they approximate heatmaps for each bone, which serves as an intermediary supervisor for numerous datasets.

		Architecture	**Advantage**	**Drawback**
2D depth maps	Tompson et al. [138]	Deep network, (LBS) model	The advantage is used IK as an intermediate heatmap for a more robust hand estimation	Need some improvement: Optimizing IK stage to minimize jitter and use Kalman filter to clean up ConvNet feature output.
	Oberweger et al. [135]	Deep Prior	Easier to implement and enhances prediction quality	Less accurate results on ICVL dataset due to noisy annotation data.

	Zhou et al. [140]	HBE network	Less complexity, faster, efficient for real-time applications, better for severe interactions with objects	Learning the entire hand with three separate branches may lead to inaccuracies and missing data.
	Du et al. [141]	CrossInfoNet	Accurate results, won first place compared to state-of-the-art methods	Outperforms using depth maps compared to 3D point-based methods.
	Sinha et al. [142]	DeepHand	Handles occlusion and missing stages, preventing reset of settings	Activation feature estimation only accurate in joint angle domain.
3D representations	Ge et al. [150]	Multi-view CNN	Uses 3D data for real-time robust 3D coordinates, better generalization	Some failures due to the lack of a calibrated hand model, ground truth, and temporal data.
	Ge et al. [145]	PointNet hierarchical	Uses a normalized point cloud for efficient 3D hand articulation learning	Failure cases due to the absence of depth data near the hand joint.
	Ge et al. [146]	3D CNN	Achieves better info using a 3D point cloud, robust estimation with less error than state-of-the-art	Accuracy at 64x64x64 resolution takes time, requires more memory.
	Moon et al. [147]	V2v-PoseNet	Uses a 3D voxelized grid for accurate joint prediction	Inaccuracies due to the change from voxel-to-voxel to pixel-to-voxel.

Table 3.2: Summary of Depth-based hand pose estimation methods according to our proposed taxonomy.

3.5.2 Image based method

The image-based strategy has two sub-categories: appearance-based and model-driven. The first aims to predict 3D key points location or heatmaps, while the second seeks to predict hand joints such as pose and shape.

3.5.2.1 Appearance-based methods

It is a widely used approach for estimating the position of joints through RGB images. **Zimmermann et al.** [151] created a network for hand pose estimation employing four different deep learning networks. The first used a CNN for hand segmentation to localize the hand in the image using a lighter version of **Wei et al.' s** [152] human body detector trained on hand datasets. A mask image, including the hand pixels, is predicted as the network output. The hand is clipped and scaled to feed into "PoseNet," a detection-based network that generates score maps for each joint. To estimate 3D coordinates, **Zimmermann et al.** constructed a network called PosePrior that used a regression technique to predict 3D coordinates by transforming the canonical space using a transformation matrix into a three-dimensional hand pose to estimate 3D coordinates.
Simon et al. [153] provided a method for estimating the hand pose. It is prone to occlusion difficulties because it uses a multi-camera system called Multiview bootstrapping. They used the Panoptic Studio at Carnegie Mellon University, which has over 500 cameras in a spherical configuration (480 VGA and 30+ HD cameras). This approach used a keypoint estimator to produce noisy labels in multiple hand views. The noisy detection is triangulated in 3D using multiview geometry. Thus, the projected triangulation is employed in the training process to improve the detector.
Furthermore, **Spurr et al.** [154] defined the cyclic concept(GAN) for generating a relationship between RGB image and 3D hand pose, which took as input 2D keypoint data, RGB images, and 2D depth images to evaluate the generality of the proposal method. It plans to develop a cross-modal latent space by combining two VAE networks, one for estimating the pose latent space and the other for generating the input data. Each VAE encodes RGB images and 3D keypoints, respectively. Thus, the architecture contains two decoders for RGB images and 3D joint configurations. The latent space was achieved by training the decoder of the 3D→3D and the trained encoder of the RGB→RGB. The obtained latent space estimates 3D hand poses from RGB images considered superior to the most advanced level under different settings.
Additionally, **Theodoridis et al.** [155] introduced a new cross-modal variational alignment method comprising two variational autoencoder networks that aligned two latent spaces. Two VAEs were trained independently during the training step to generate the RGB→3D and the single 3D→3D latent space. The method used variational alignment to project the first network latent space onto the single-modal VAE through an intermediary distribution.

Most recently, **Iqbal et al.** [30] presented a 2.5D pose representation from RGB images, subsequently generating a 3D pose estimation. The proposed architecture consists of an hourglass network that performs 2D latent space and depth maps to obtain a 2.5D representation using a differentiable loss function. Notably, the training process was designed to teach the network how to generate depth maps, eliminating the need for actual depth data. The experimental results achieve a state-of-the-art 2D and 3D hand pose estimate in severe occlusion on numerous challenging datasets.
Spurr et al. [156] based on the network mentioned above, which is deemed a robust network due to its better quality in estimating 2.5D representation using the set of loss terms that have been constructed. Those phrases implied biomechanical constraints (BMC) on 3D joint predictions, which might be incorporated through training to produce anatomically realistic 3D hand poses only using 2D supervision. Hence, the proposed approach results better when using the biomechanical constraint than the state-of-the-art methods.

A closely related work in spirit to our contribution is the research conducted by **Li et al.** [29], which introduced a novel and compressed latent distribution representation that directly links and constrains the 2D and depth feature maps of each joint. This representation offers several advantages, including addressing the channel correspondence issue and enhancing cross-dataset accuracy. The global architecture design consists of an encoder-decoder network with two key components. The first component estimates the 2D hand pose using a 2D latent representation. The second component employs a Latent Distribution Representation(LDR) to jointly predict the 3D hand pose. The model is crafted to preserve feature map resolution while retaining critical regional details. This approach aligns closely with our research objectives and demonstrates the potential of novel representations and architectural designs in improving the accuracy and robustness of 3D hand pose estimation.

3.5.2.2 Model-driven methods

According to this category, which included all methods that exploited the hand model parameters, the work of **Mueller et al.** [157] proposed an architecture for estimating 3D hand tracking for real-time using a combination of Convolutional Neural Networks (CNN) and a kinematic 3D hand model. They built novel synthetic datasets called GeoConGan(Geometrically Consistent CycleGAN) to convert computer-generated images to real images. The network consists of two stages: CNN hand join regressor (RegNet) to predict a 2D hand heatmap and 3D joint locations. RegNet output is fed into the next stage, "kinematic skeleton," to get both 2D and 3D predictions while minimizing fitting energy to verify the anatomic accuracy of the obtained hand.
Guan et al. [158] defined a real-time method for estimating the 3D hand shape and pose from a single RGB image. The MobileNetV3 network is generated to extract the main features from an input image. It is considered one of the latest generations of efficient and lightweight CNN

targeted for mobile devices. The output of the previous network was fed into a 3D regression module to produce a camera, hand shape, and joint angle parameters employed by the 3D hand model MANO to reconstruct the hand, which is projected to a 2D plane image using the camera parameters.

In the same context, **Ge et al.** [159] built a Graph Convolutional Neural Network to conduct linear regression on a full 3D mesh of the hand surface, which provides more information about the 3D hand shape and pose. Using a two-stacked hourglass network to extract global characteristics from RGB images and 2D heatmaps. After that, a residual network was merged and encoded as a latent feature vector. The output is fed into (GCNN), which comprises two fully connected layers. The feature vector is translated into 80 vertices in a rough graph with 64 weak features by applying the operations of convolving, upsampling, and assigning to a finer graph followed by two other upsampling layers and four graphic convolutional layers. The network outputs 3D coordinates of 1280 mesh vertices. So, the 3D hand pose is linearly regressed from the 3D hand mesh. Similarly, **Taheri et al.** [160] generated a large dataset containing complex 3D objects. While "grasping" is the most common motion that is used in the collection of data. The (GrabNet) model aims to understand how humans grasp and manipulate objects. It is comprised of two modules: coarse prediction and refinement. The first conditional Variational Autoencoder (cVAE) used a condition on the object shape encoded using the Basis Point Set (BPS) to generate a 3D grasping hand and the coarse net estimate of a plausible grasp. A refinement network improves contacts, which inputs an initial grasp and the distance D from MANO vertices to the object mesh. By giving unknown 3D objects to the CoarseNet, it gets an initial grasp and passes this to the refineNet to acquire the final grasp estimate.
Boukhayma et al. [161] presented ResNet50 as a deep convolutional encoder that takes as the input RGB image and optionally a 2D joint keypoint to estimate the pose, shape, and view parameters. It consists of the MANO hand model and a re-projection module to compute the articulated mesh deformation hand model that predicts hand parameters. Then, they project the estimated hand pose into the image domain and leverage 2D weak annotations as supervision. At training time, the network used a made-up dataset and four "losses" on the encoder's network to reduce their size and only let the reconstruction of physically plausible hands happen.

		Architecture	Advantage	Drawback
Appearance-based	Zimmermann et al. [151]	CNN: HandSegNet, A mask, PoseNet, PosePrior	Learning 3D pose prior permits to predict realistic and accurate pose.	Unavailability of annotated huge scale dataset with realistic images.
	Spurr et al. [154]	Variational Autoencoder Networks	Learns a usable hand model throughout pairs of hand configurations with many modalities	Inaccurate results on depth data.
	Simon et al. [162]	Multiview CNN	Accurate results compared to the depth sensor, offers complex object interactions	Works in most views, but not so well in views where the hand is obscured.
	Theodoridis et al. [155]	Cross-modal Variational alignment	More generality and reproducibility testing two datasets.	Not specifically developed to estimate 3D hand pose.
	Spurr et al. [156]	Cross-modal	Reduce the depth ambiguity, enabling the network to utilize 2D images effectively.	Biomedical constraints are more difficult to define.
Model-driven	Mueller et al. [157]	CNN: ResNet50, kinematic hand model	Robust to self-occlusion and occlusion by objects, particularly in moving egocentric perspectives	Some failure cases occurs during interaction.

	Guan et al. [158]	CNN: MobileNetV3, MANO model	More efficient and accurate results for 3D hand shape and pose.	The lack of prediction on a big dataset and the need to optimize a network to design a hand model.
	Ge et al. [159]	Graph CNN	Tested on a new dataset and two others that built efficient and appropriate 3D hand mesh.	Requires Mocap data to generate a large dataset.
	Taheri et al. [160]	GrabNet: CoarseNet, RefineNet, MANO	Create a Self-built dataset (GRAB) that contains enough data and variability to learn human grasping of objects	Doesn't have synchronized image data because they're focusing on accurate MoCap.
	Boukhayma et al. [161]	ResNet50, MANO model	Using concatenation of encoder and decoder, allows estimating pose with good generability on images in the wild	Adding the MANO model as the corrective blend shapes parameters for fine-tuning to get higher performance.

Table 3.3: Summary of Image-based methods for 3D hand pose estimation according to our proposed taxonomy categorized by input modality.

3.5.3 RGBD-based methods

It is one of the most widely used approaches in 3D hand pose estimation due to the accurate and robust results compared with the depth-based and image-based approaches that use RGB images and depth values in input with different cases. As mentioned in the work of **Dibra et al.** [163] defined an unsupervised system to prevent the need for the real world during training.

Using both depth maps and RGB images to estimate the 3D hand pose. Initially, the network extracts the background from a monocular RGB image, with the output being fed into SynthNet. A CNN model trained explicitly on synthetic data that contains paired depth maps and RGB images to infer joint rotation angles. Adding a depth loss component to SynthNet by combining ground-truth paired depth maps and expected angles with a Point Cloud Sampled (PCL) from the hand model. This helps them fine-tune the network to real monocular data.

Meuller et al. [164] designed an approach for estimating the 3D articulated hand pose from the RGB-D sensor, separated into two sub-networks. The Hand localization network (HALNet) is achieved using a CNN to provide the 2D image location of the center of the hand (image-level heatmap) of the root. A 3D joint regression network (JORNet) was employed to regress root-relative 3D joint locations from the cropped hand image. They estimated the joint angles of the kinematic pose tracking framework based on the projected 3D joint positions and 2D joint heatmaps to generate a spatially smooth and optimized pose.

In addition, **Kazakos et al.** [165] developed a framework called FuseNet that uses two distinct deep learning streams for RGB images and depth disparity maps. The main idea is to fuse at various levels: (a) input level, which can be generated by putting the fusion block after the input layer; (b) feature level, which employs the fusion block after any convolution operation, pooling, or fully-connected hidden layer; and (c) score level, which refers to the combination of each stream's prediction. The authors provide a learnable function termed the locally connected fusion function. As a result, the performance of the two-stream architecture barely beats that of the depth-based single-stream approach.

In the same context, **Mofarreh et al.** [166] designed an architecture based on hybrid networks aiming to estimate 3D hand poses using RGBD images. The network consists of three convolution layers and four fully connected layers to build the coordinates of joints. The output of the previous network is fed into the SHNet to reduce the dimensionality of the estimated joints using Principal Component Analysis (PCA). The second network includes popular networks such as VGG16, DenseNet169, and MobileNet. The third one used the residual concept and referred to RCNet.

Choi et al. [167] created a real-time technique based on local shape descriptors to retrieve nearest neighbors from the annotated dataset to exploit RGB-D data. This data evaluates the unknown pose parameters using a joint matrix factorization and completion (JMFC) method on a hand pose library.

Table 3.4: Summary of RGBD-based methods for 3D hand pose estimation according to our proposed taxonomy categorized by input modality.

		Architecture	Advantage	Drawback
RGBD-based method	Dibra et al. [163]	CNN: SegNet, SynthNet, PoseNet	Self-build synthetic dataset with realistically rendered hand; Strongly and accurate results	Owing to differences between the ground truth skeleton in StereoDS and the proposed hand model skeleton.
	Meuller et al. [164]	CNN:HALNet, JORNet	Perform well in egocentric viewpoints, noticeable occlusion, and scene clutter	Some failure cases during hand-object-interaction require deep domain adaptation.
	Kazakos et al. [165]	FuseNet	Double stream architecture perform well during training with depth images.	FuseNet doesn't improve their results compared to other approaches.
	Mofarreh et al. [166]	Hybrid deep learning	More robust and accurate to handle occlusion problem by incorporating residual layers.	NYU datasets are overfitted in the face of more complicated networks, causing the hand poses to perform poorly.
	Choi et al. [167]	Collaborative filtering	Accurate results in numerous cases of hand configurations under occlusion.	Utilizing nuclear norm regularization in JMFC algorithm to achieve lower rank factors.

3.6 Datasets and Evaluation Measurement

In this section, we present the most available dataset in the hand pose estimation field and the evaluation measure used to evaluate the effectiveness and robustness of the proposed method:

3.6.1 Benchmark Datasets

Many datasets have been created in hand pose estimation that played an essential role during training. However, early datasets contain only depth data, and with the advent of accurate methods, they need an extensive dataset. For that, most research focuses on building synthetic and real data. For that, we present some significant datasets and discuss their characteristic in detail:

- **Stereo hand pose tracking benchmark(STB) [168]:** is an RGBD dataset created using a stereo camera. It comprises 12 video sequences and 18,000 stereo image pairs (15,000 for training and 3,000 for evaluation), with 21 3D keypoint annotations and corresponding depth maps.

- **EgoDexter [169]:** is RGB-D datasets obtained from egocentric viewpoints of hand interaction with objects in a realistic chaotic area using an imaging device mounted to the human body. It is made up of 4 sequences totaling 3190 frames. Because no training set is given, EgoDexter is used for evaluation.

- **Dexter+Object [170]:** Like EgoDxter, this dataset concludes six test sequences with two subjects. A static camera captures the images and contains the interaction between the hand and a cuboid object and is primarily used for evaluation reasons.

- **Rendered Hand pose Datasets(RHD) [151]:** contains 43,986 hand images in total(41,258 for training and 2728 for evaluation). The RHD dataset comprises 20 participants executing 39 actions, as well as depth maps, 2D and 3D location annotation information, and segmentation masks. Because of the varying views, hand shapes, obstructed fingers, large visual variety, and noise. The dataset is ranked as one of the most challenging difficulties.

- **Freihand dataset [170]:** is an RGB multiview dataset comprising 134K frames, 130K for training, and 4K for evaluation. It is performed in ground-truth 3D locations of the hand, 3D hand shape, hand mask, and the intrinsic camera matrix.

- **ICVL Hand Pose Dataset [171]:** The Imperial College Vision Lab dataset is one of the most well-known in hand pose estimation. It is mainly composed of 332,5K depth images, 331K from training, and 1.5K from evaluation, and incorporates sequences from 10 different participants to take 26 distinct positions. The depth images are of excellent quality, with clear outlines and minimal noise.

- **NYU Hand Pose Dataset [165]:** The dataset from New York University (NYU) has 82K images. The training sets contain 72K samples for one subject's training, whereas the test sets have 8.2K frames. It used ground truth to obtain the annotated images.
- **BigHand 2.2M Benchmark Hand Dataset [172]:** The largest hand pose dataset, as its name suggests, contains 2.2 million depth images with annotated joint locations generated from 10 different subjects employing kinematics and six 6D electromagnetic sensors.
- **MSRA15 [173]:** This dataset contains around 76K RGBD images collected from 17 hand gesture movements by Intel's Creative Interactive Gesture Camera across 9 individuals. The training is done on eight different subjects, and the rest is used for evaluation. The small size of this dataset is a drawback, and the annotations have a high mistake rate.
- **HandNet Dataset [173]:** One of the most comprehensive datasets in the depth category. They used kinematic sensors with ten different individuals to produce diverse hand sizes, half male and half female and contained 202K training frames, 10K testing frames, and 2.7K validation samples.
- **GANerated Hand Dataset [169]:** One of the newest RGB-based datasets with interaction objects is built to estimate the hand pose while it is obscured. It comprises around 330k frames of synthesized hand shapes annotated in 3D using a model. The hand pose samples were captured using kinematic electromagnetic sensors. In addition, CycleGAN was used to make these computer-generated images look real. They used several backgrounds behind the hand to create a realistic scene. Artificial objects were also placed on the hand to provide occlusion.
- **HANDS 2017 [174]:** The dataset is generated by selecting images and sequences from the BigHand2.2M and First-Person Hand Action (FHAD)datasets. It comprises depth images, 957K images for the training set with 5 subjects, and 295K frames for the testing set with 5 unseen subjects. It combines various hand configurations, hand shapes, viewpoints, and occlusions.
- **SynthHand5M [169]:** This dataset employs Manuel Bastion LAB to create a synthetic hand model with 5 million depth images (4.5 million for training and 500 thousand for

testing). It has joint angle and segmentation masks, as well as annotations of 3D joint locations with 26 DOFs and 1193 3D hand meshes. As shown in table 5 demonstrates the above datasets with common properties.

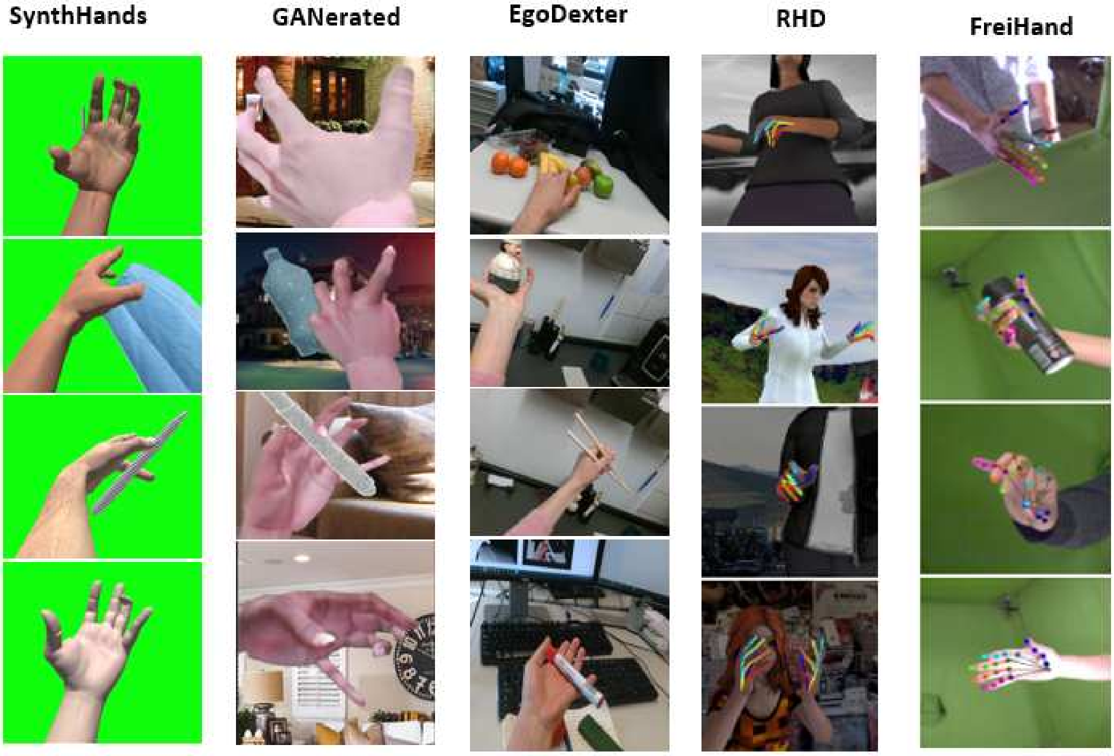


Figure 3.13: Example of different datasets for estimation hand pose.

Datasets	Year	Modality	Type	Annotated	Subjects	Frames	Resolution
STB [168]	2017	RGB+depth	Real	21	1	18K	640480
EgoDexter [169]	2012	RGB+depth	Real	5	4	3190	640480
Dexter+Object	2016	RGB+depth	Real	5	2	3K	640320
FreiHand [170]	2019	RGB	Real	21	32	134K	224224
ICVL [171]	2014	Depth	Real	16	10	332.5K	320240
NYU [165]	2014	Depth	Real	36	82K	2	640x480
BigHand2.2M [172]	2017	Depth	Real	21	10	2.2M	640480
MSRA15 [173]	2015	RGB+depth	Real	21	9	76K	320240
HandNet [175]	2015	Depth	Real	6	10	212K	320x240
GAN [157]	2018	Depth	Synthetic	21	-	330K	640480
Hands2017 [174]	2017	Depth	Real	-	10	1.1M	640480
SynHand5M [169]	2018	Depth	Synthetic	22	-	5M	320240
Rendered Hand Pose [151]	2017	RGB+Depth	Synthetic	21	-	41K	320320

Table 3.5: Commonly used public datasets for 3D hand pose with their properties.

3.6.2 Evaluation Metrics

To evaluate 3D hand pose estimation methods, some metrics are viral. There are three weadly measurements:

- The first one is **End-Point-Error(EPE)**: It is one of the most used metrics for estimating hand pose. It is based on the average Euclidean distance from the predicted to the ground truth over all joints in one image. The measured unit is (in mm). The formula is mentioned in (1).

  $$EPE = \frac{1}{J}\sum_{j=1}^{J}\left\|j^{pred} - j^{gt}\right\|^2 \tag{3.1}$$

  Where J is the number of estimated keypoints, and j^{pred} are the 3D coordinates of the estimated and actual j^{gt} keypoints, respectively.

- The second is employed for assessing the performance of 2D and 3D hand pose, which is **Pourcentage of Correct Keypoints(PCK)** [176] as in equation (2): is measures the accuracy of localization of distinct keypoints within a matching threshold. Each test image head segment length is outperformed as the threshold, denoted as PCK@0.5. If the mean percentage of predicted joint positions and ground truth is less than 0.2 times the torso diameter, PCK is called PCK@0.2. The higher the PCK value, the better model performance is regarded.

  $$PCK_\sigma = \frac{1}{J}\sum_{j=1}^{J}\delta\left(\left\|w_j^{pre} - w_j^{gt}\right\| < \sigma\right) \tag{3.2}$$

  Where w_j^{pre} represents the predicted 3D coordinates of the j^{th} keypoint,w_j^{gt} is its actual 3D keypoints, and σ is an indicate function.

- The last one **AUC (Area Under the Curve)** [177]on the pourcentage of correct keypoints(PCK) under different error thresholds. As demonstrated in equation (3)

  $$AUC_J = \int PCK_\sigma^J \tag{3.3}$$

3.7 Bibliometric analysis of related works using Vos Viewer analysis

3.7.1 VosViewer Overview

The investigation into methodologies for enhancing hand pose estimation encompassed an exploration of shifts in the interconnection among essential terms within research article titles, abstracts, and keywords. To establish a reference point, particular emphasis was placed on the

year 2010, a critical juncture marked by the widespread integration of technological advancements in computer vision and machine learning techniques for hand pose estimation. This temporal milestone played a pivotal role in comprehending the trajectory of these methodologies as they evolved in the realm of hand estimation. Initially, a collection of keywords was gathered, including those provided by the authors of each research paper within the specified time frame. However, considering the potential variations in keyword formulation across authors, a new set of terms was introduced to categorize these keywords. Each term in this set represents a distinct category of keywords and is referred to as an "item" in VOSviewer terminology. The subsequent analysis focused on assessing the strength of correlation among these items. To illustrate the relationships among these items, the VOSviewer tool (version 1.6.19) was utilized, involving the application of a similarity measure to quantify the level of correlation between two items.
VOSviewer offers a variety of visualization tools, such as co-occurrence maps, term maps, and network maps, which provide researchers with valuable insights into the structure and dynamics of scientific knowledge domains [178]. These visual representations help researchers identify key research areas, visualize author collaboration networks, and track the evolution of research themes over time. By presenting complex bibliometric data in an easily understandable manner, VOSviewer makes it easier for researchers to gain a deep understanding of the research landscape. Its effectiveness has led to its widespread adoption in bibliometric and scientometric studies. It provides valuable assistance in conducting data-driven analyses, equipping researchers with vital tools to effectively explore and make sense of extensive scholarly literature. By leveraging the insights derived from VOSviewer analyses, researchers can actively contribute to advancing knowledge and fostering innovation within their specific domains [178].

3.7.2 The Most Cited Documents ranking

In the context of individual research papers, the extensively cited publications cover a wide range of keywords and subject areas that are pertinent to the topic of research papers related to the assessment and improvement of hand pose estimation. These well-referenced papers seem to touch upon various aspects and subfields within this research domain, suggesting that they are influential and provide valuable insights across different areas of hand pose estimation research. Table 3.6 presents the top thirteen documents with the highest citation counts. The documents authored by Pisharady et al.in 2015 has received the highest number of citation. The most frequently cited article is titled "Recent methods and databases in vision-based hand gesture recognition : A review". This article has been cited 248 times.

In the second position, the research paper by Meuller et al. in 2019, titled "Real-time pose and shape reconstruction of two interacting hands with a single depth camera", delves into the substantial aspects of estimation of both the pose and shape of the hand from a single depth camera. This method stands out for its ability to simultaneously track and represent the 3D

positions and orientations of each hand's joints and fingers, capturing the intricate details of hand movements and interactions. Moreover, the approach goes beyond pose reconstruction by also focusing on accurately reconstructing the 3D shapes of the hands, including their fingers and palms. This article leads to handling the pose estimation challenges. By emphasizing the simultaneous tracking of interacting hands contributes significantly to improving the realism and functionality of various interactive and immersive technologies.
In the same context, the documents published in 2019 by Li et al. titled" A survey on 3D hand pose estimation : Cameras, methods, and datasets", surveys on 3D hand pose estimation represent a thorough and insightful examination of this dynamic field. Their review is structured around three essential pillars: cameras, methods, and datasets. In terms of camera technologies, the authors meticulously explore the advantages and limitations of various camera types, including depth cameras and RGB cameras, shedding light on their roles in achieving precise hand pose estimations. From traditional computer vision techniques to cutting-edge deep learning approaches, Li et al. offer a comprehensive overview, highlighting the strengths and weaknesses of each method and their adaptability to different real-world scenarios. The significance of this paper summarizing the existing research is that the authors identify challenges and propose future directions. They underscore the relevance of 3D hand pose estimation in practical applications, highlighting its potential to reshape user experiences and enable innovative technologies.

The bibliometric investigation aimed to delve into the scholarly influence and impact of research articles about the domain of hand pose estimation. The quantification of citations, a pivotal gauge of research's importance and acknowledgment, assumed the role of the primary metric for assessing each publication's significance. Through an examination of citation dispersion across these articles, valuable discernments emerged regarding the prominence of specific works. Articles with a substantial citation count reflect a robust resonance within the research community, indicative of groundbreaking contributions or pioneering insights. On the flip side, articles with a lower number of citations might indicate areas with limited impact or research that is still in its nascent stages. Furthermore, our analysis went beyond just counting citations to include temporal citation patterns. This material aspect provided a more profound understanding of the long-term relevance and sustainability of the articles.

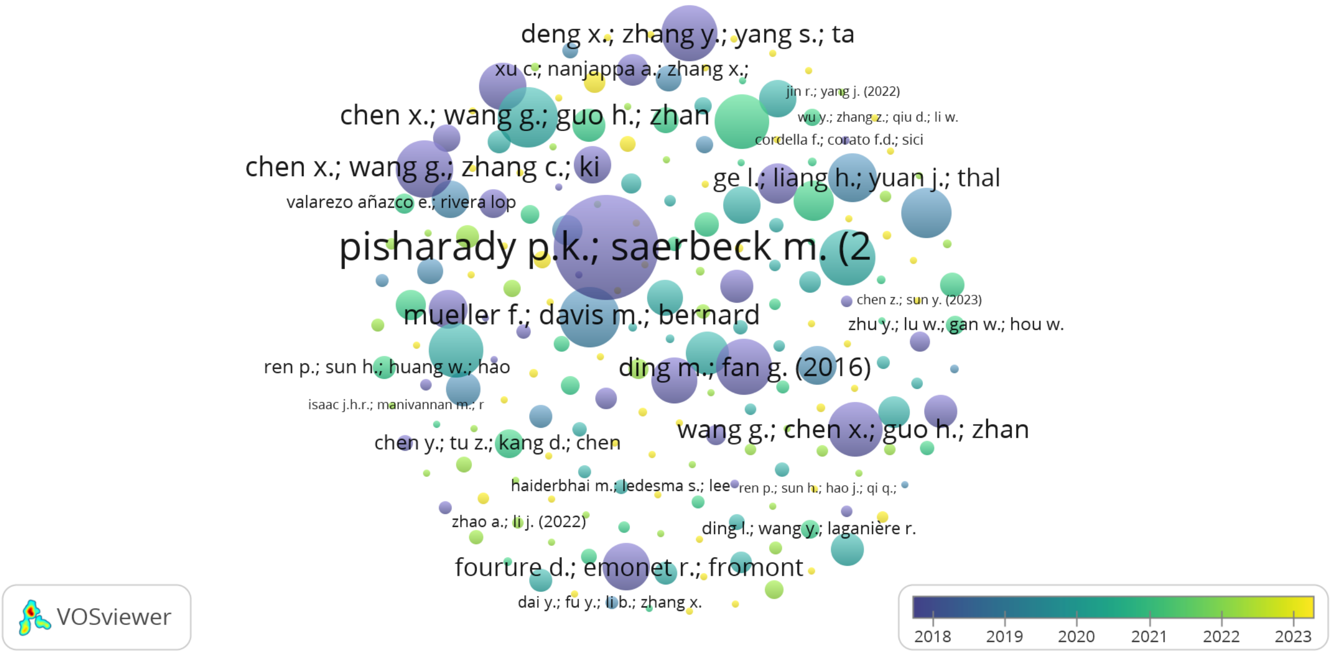


Figure 3.14: A bibliometric analysis of publications related to the topic of Hand Pose estimation. The size of each node (represented as circles) corresponds to the number of documents associated with a particular author for six years(From 2018 to 2023).

3.7.3 The Most Cited Authors bibliographic analysis

Table 3.7 provides a visual representation of the top 15 authors who have exhibited exceptional productivity in document output and the top 15 authors who have accumulated substantial citation counts. This ranking assessment was carried out using a comprehensive counting method within VOSViewer, where each document or citation was considered equally significant, regardless of the total number of authors involved in a particular document. The citation analysis was conducted without imposing a minimum threshold for the number of documents. Notably, Pisharady et al. emerged as the most highly cited author in hand estimation from 2015 to 2023, as evidenced by the impressive number of citations attributed to their work.
Using VOSViewer, we performed a co-authorship analysis and created clusters using the association strength technique. To be included, authors had to have a minimum of one document, leading to 199 authors meeting this criterion. In addition, each author from the list is prominently featured within the interconnected central clusters, using blue, red, green, yellow, and pink colors. These clusters represent groups of authors with strong collaborative relationships, and authors who frequently collaborate tend to be visually positioned close to each other in the visualization.

Table 3.6: Examples of the most cited journal paper on "hand pose estimation" based on total citation parameter.

Document Title	Authors	Year	Total Citation	Total link Strength
Recent methods and databases in vision-based hand gesture recognition: A review	Pisharady, P.K, Saerbeck, M.	2015	248	1176
Real-time pose and shape reconstruction of two interacting hands with a single depth camera	Mueller & al.	2019	83	21771
Pose guided structured region ensemble network for cascaded hand pose estimation	Chen et al.	2020	80	10383
SHPR-Net: Deep Semantic Hand Pose Regression From Point Clouds	Chen & al.	2018	73	3586
Joint Hand Detection and Rotation Estimation Using CNN	Deng & al.	2018	70	4352
Hand sign language recognition using multi-view hand skeleton	Rastgoo & al.	2020	69	1087
Feature Boosting Network for 3D Pose Estimation	Liu & al.	2020	69	2899
Articulated and generalized Gaussian kernel correlation for human pose estimation	Ding & al.	2016	69	4352
A survey on 3D hand pose estimation: Cameras, methods, and datasets	Li & al.	2019	56	2206
Real-Time 3D Hand Pose Estimation with 3D Convolutional Neural Networks	Ge & al.	2019	55	48345
Depth-Based Hand Pose Estimation: Methods, Data, and Challenges	Supančič & al.	2018	51	8427
RGB2Hands: Real-Time Tracking of 3D Hand Interactions from Monocular RGB Video	Wang & al.	2020	43	13408
Latent regression forest: Structured estimation of 3D hand poses	Tang et al.	2017	37	17497

Table 3.7: Top 15 most cited authors on Hand pose estimation topic from (2018-2023)

Ranking	Authors	Total Citations	Total link Strength
01	Pisharady, P.K, saerbeck, M.	247	12
02	Rastgo r.	136	46
02	Meuller f.	83	67
3	Chen x.	80	131
4	Chen x.	73	104
5	Deng x.	2018	70
6	Rastgoo & al.	70	5
7	Liu t& al.	69	25
8	Wang g.& al.	69	2
9	Gomez-denoso f.	68	1
10	Li r.	66	37
11	Ge l.	2018	51
12	Supančič j.s	61	29
13	Fourure d.	56	51
14	grzejszczak.t	55	77
15	Wang j.	50	119

3.7.4 Keywords Occurrence network for "Hand Pose estimation Topic"

By utilizing Author Keyword analysis in VOSViewer, a total of 1678 keywords were identified. The top 109 keywords, presented in Figure 3.16 and sorted by their overall frequency, encompass various terms from various fields, including Engineering, Computer science, and Medical Science, among others. Through VOSViewer co-occurrence analysis, these author keywords were visually represented and interconnected. To simplify the visualization and focus on a more manageable set of keywords, only terms that occurred 17 times or more were retained, resulting in a total of 22 terms organized into four clusters (indicated by green, purple, yellow, and blue colors in Figure 3.17). Clusters represent groups of closely connected nodes, and terms that frequently appear together tend to be positioned closer to each other in the visualization. These clusters were formed using the association strength method. These clusters identify thematic areas within your dataset.

For that, co-occurrence analysis using VOSviewer is a valuable method for identifying research themes, trends, and patterns within a dataset of scholarly publications. It can provide insights into the relationships between keywords and help researchers discover meaningful associations and research directions.

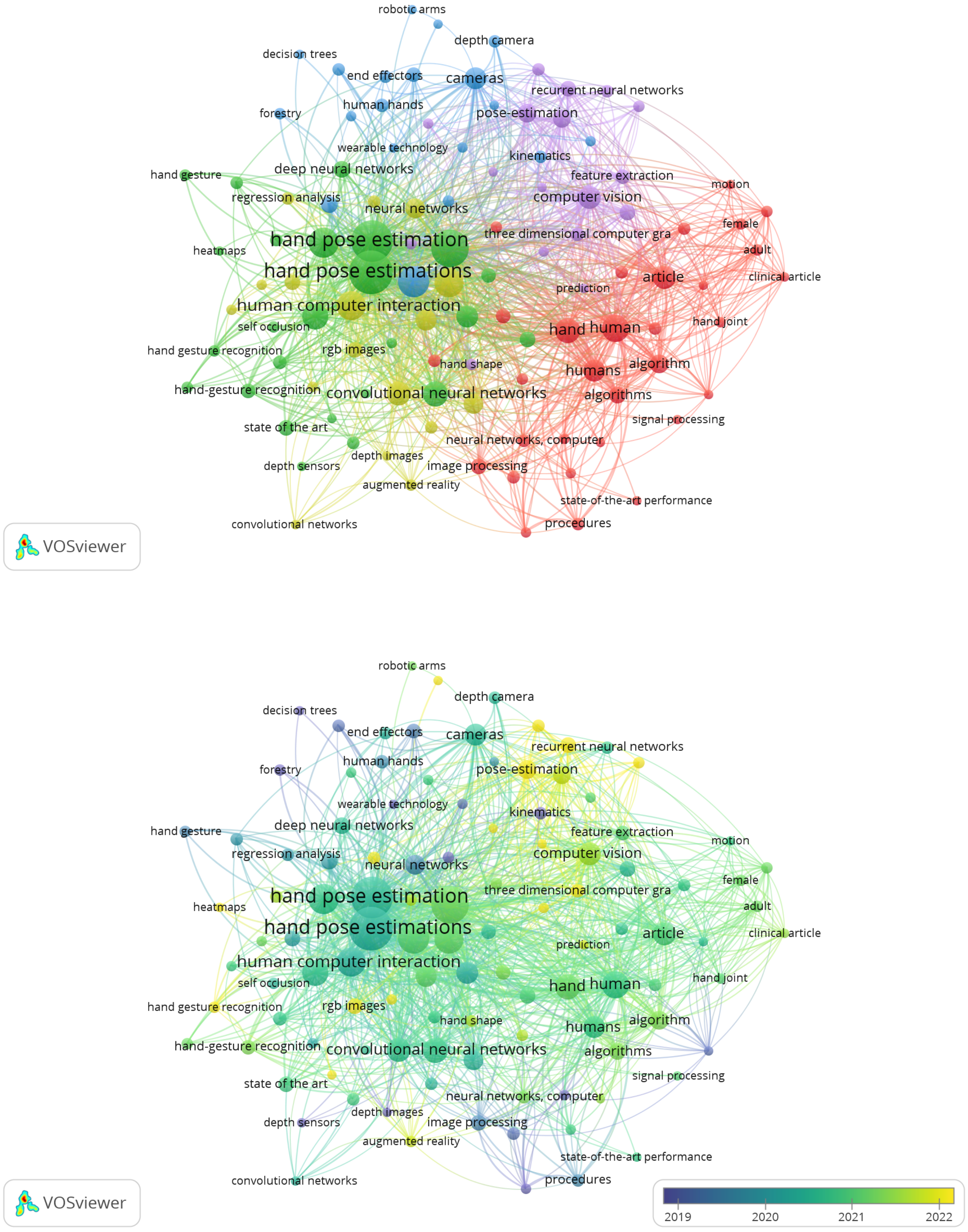


Figure 3.15: Author keyword co-occurrence network for "hand pose estimation" The size of each node (circle) indicates the number of documents associated with an author. Lines represent co-occurrence between two authors and appear when authors co-occur at least once

3.8 Discussion

In this section, we discuss all the obtained results; Firstly, we respond to the first research question(RQ1) and acquire a thorough comprehension of the research, it is crucial to determine the research objective according to our classification. Table 3.4 mentions that the largest focus of researchers has been on RQ1, RQ2, RQ3, RQ4, and RQ5. There is a significant difference between existing research in terms of accuracy and time speed as provided in the literature. The following are descriptions of the research objective categories corresponding to the research questions:

1. Recently, because of the interesting topic of 3D hand pose estimation and its application in the computer vision field. So, the main objective of most researchers is to develop methods aimed at resolving the abovementioned challenges by learning human hands during interaction with virtual objects using deep learning algorithms.

2. Depending on Q2. Most researchers in the hand pose estimation focus on building architectures based on different convolutional neural networks or popular architectures such as ResNet50, MobileNet, Graph CNN, or Cross-modal.

3. For Q3, in order to increase the efficiency and reliability of the proposed method, which is dependent on the quality and quantity of training data. Datasets have a significant role in DNN-based hand pose estimation. The majority of researchers evaluated their approaches on accessible synthetic datasets, which are less expensive to develop than real-world datasets that use advanced hardware (such as depth cameras).

4. To respond to the Q4, All defined methods used three commonly popular metrics for evaluation quantitatively: Mean End-Point-Error(mEPE), Percentage of Correct Keypoints (PCK) which is considered an accuracy measure and sub-metric "AUC" is the Area Under the Curve on PCK for different thresholds, and End-Point-Error(EPE)is a common error metric. More details are explained in section 3.6.

5. In RQ5. The other case studies that are considered to be compared with each proposed approach are dependent on self-reliability, accuracy, effectiveness, and most challenges to be solved in each paper.

6. To respond to RQ6, most authors aimed to resolve one of the available challenges faced by the issue of hand estimation such as high articulation, data labeling, low resolution, hand gesture, or occlusion. In our research question, we focus on the occlusion challenge.

7. According to RQ7, the concept of real-time requirements is necessary for hand estimation, especially on hand-object interaction, and our research objective, we focus on this aspect due to the lack of research to predict the human pose in real-time.

As seen in Table 3.8 and the proposed QOS parameters, most researchers are focused on 4 or 5 primary questions while the rest have overlooked them. From the above Qos requirements and their probability of occurrence in the literature, it is essential to identify the relevant and valuable Qos factors and their importance percentage.

Our proposed taxonomy above highlights extant research in the field of hand estimation, which is a topic of great interest in recent years. Hand pose systems may accurately track the movements of the hand in a relatively controlled environment in real time. On the other hand, many obstacles have been faced during the estimation of hand pose, especially in open and complicated settings where the number of computing resources required and these issues have not yet been resolved.

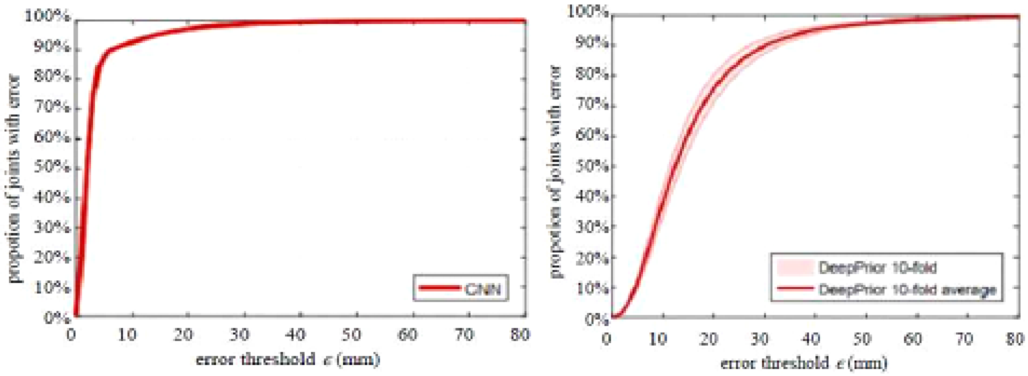


Figure 3.16: The dispersion of End-point error using (a) CNN architecture on BigHand dataset (above graph) and (b) Egocentric dataset (bellow graph) [25]

From the analysis above of our classification, the key to depth-based methods is to use depth information (2D depth map or 3D representation) to estimate the position of each joint in hand. It is generally considered a robust approach that offers simple concepts, implementation, and high interpretability. However, the hardware requirements are too expensive, and actual use cases are severely limited. Thus, due to the unavailability of depth data, hand pose methods from single RGB images were developed rapidly.

RGB-based approaches are easier to implement, and a suitable generalization of the model is used everywhere and considered a standard method using deep learning. It has the advantage of reducing the dimensionality of input from 2.5D to 2D to produce methods dramatically simple. Further, it does not require expensive hardware equipment and fewer instructions on implementation scenarios. The data required to train an image-based is significantly greater than that required to train an equivalent network utilizing depth maps.

RGBD-based methods aim to associate two last categories. It is taken as input, an RGB image and depth value is used through estimating. As a result, it achieves better results and accurate approximation to tackle most challenges, such as occlusion, compared to others and alleviates

	Approach	RQ1	RQ2	RQ3	RQ4	RQ5	RQ6	RQ7
Zhou et al [140]	HBE network	✓	✓	✓	✓	✓		
Du et al [141]	CrossInfoNet	✓	✓	✓	✓	✓		
Oberweger et al [139]	DeepPrior++	✓	✓	✓	✓	✓		
Ge et al [150]	MultiView CNN	✓	✓	✓		✓		
Ge et al [146]	3D CNN	✓	✓	✓	✓	✓		✓
Moon et al [147]	V2V-PoseNet	✓	✓	✓	✓	✓		✓
Spurr et al [154]	Cross-modal	✓	✓	✓	✓			
Iqbal et al [162]	CNN	✓	✓	✓	✓	✓		
Meuller et al [157]	CNN	✓	✓	✓	✓	✓	✓	✓
Guan et al [158]	MobileNetV3-Small	✓	✓	✓	✓	✓	✓	✓
Ge et al [159]	Graph CNN	✓	✓	✓	✓	✓		
Zimmermann et al [151]	CNN: HandSegNet, PoseNet	✓	✓	✓	✓	✓		
Dibra et al [163]	CNNs	✓	✓	✓	✓	✓		
Mofarreh et al [166]	Hybrid Deep Learning	✓	✓	✓	✓	✓	✓	✓
Wan et al [143]	CrossingNets	✓	✓	✓	✓	✓		
Theodoridis et al [155]	Cross Modal	✓	✓	✓	✓	✓		

Table 3.8: Desired objectives in the investigated researchers.

Depth ambiguity.

The second issue examined in our research is the results collected in Table 9 when we compare the methods that used either 2D CNN or 3D CNN for estimating the location of each joint. The input data to 3D CNN is a 3D representation(Point clouds or 3D voxels). The obtained precision in 3D CNN is more robust than in 2D CNN. Therefore, the error calculated in 3D CNN is less than in 2D CNN. Besides, The efficiency of 3D CNN methods is superior to that of 2D CNN due to the 3D data that has more dimensions than the 2D data. So, the calculation time is higher.

The third issue in this study concerned the estimation of 3D pose methods quantitatively using prevalent challenges datasets: MSRA, NYU, ICVL, etc. We classify each 3D hand pose method using a popular database during training. We obtained excellent accuracy and fewer errors as mentioned in the work of **Oberweger et al.** tested their network called "PosePrior" in the ICVL dataset(AUC is 90% and EPE is 10mm). The work of **Iqbal et al.** gets higher precision (99%). In [156], the authors are training on two datasets(RHD is 19.73 and STB is 8.56), and the average 3D error is high compared to other datasets. On the other hand, the accuracy is very high (respectively on STB is 85% and NYU is 98%). As shown in Table 3.9. The results in accuracy using the evaluation metric Area Under the Curve(AUC) and End Point End(EPE) to compute the mean error. Further, figure 3.17 shows the main results in two benchmark datasets (BigHands and egocentric datasets)using CNN architecture. Figure 3.17(top) is tested on the BigHand dataset, which has released more than 90% accuracy and less than 10mm(Threshold error) for 3D distance error. In contrast, the results in Figure (bottom) The egocentric dataset showed more than 30% of 3D distance errors being less than 10mm.

Table 3.9: Results of 3D hand pose estimation approaches according to our proposed taxonomy regarding accuracy and benchmark datasets.

Approaches	Literature	Evaluation Metrics		CNN types		Test Speed	Datasets
		mEPE	AUC	2D	3D		
2D depth maps	Zhou et al [140]	6.24	-	✓		673	ICVL, HANDS2017
	Du et al [141]	7.20	-	✓		124.5	
	Sinha et al [141]	16.35	-	✓		32	
	Oberweger et al [139]	10.4 19.73	0.920 0.998	✓		500	
	Wan et al [143]	10.2 15.5 12.2	-	✓		90.9	
	Tompson et al. [138]	4.10	-	✓		24.9	Self-built dataset
3D representations	Ge et al [144]	6.30 7.70	-	-	✓	82	ICVL MSRA15
	Ge et al [146]	14.1 9.58	-	-	✓	215	MSRA15 NYU
	Moon et al [147]	-	0.580 0.994 0.710	-	✓	3.5	EgoDexter STB Dexter+Object
	Moon et al [147]	9.1 6.3 7.7	-	-	✓	41.8	NYU ICVL MSRA
Model-driven	Spurr et al [156]	19.73 8.56	0.849 0.983	✓	-	-	RHD STB

Continued on the next page

	Theodoridis et al. [154]	15.61	0.907	✓	-	-	RHD
		6.93	0.997				STB
	Iqbal et al [162]	13.14	0.943	✓	-	150	RHD
		-	0.994				STB
		-	0.580				EgoDexter
		-	0.710				Dexter+Object
	Zimmermann et al [151]	5.00	-	✓	-	-	RHP
		-	0.57				Dexter+Object
Appearance-based	Meuller et al [157]	-	0.965	✓	-	-	Stereo
		-	0.80				Ganerated
		-	0.54				Dexter+Object
	Guan et al [158]	-	0.994	✓	-	110	STB
		-	0.741				FreiHand
	Ge et al [159]	6.37	0.920	✓	-	50	STB
		-	0.998				RHD
	Boukhayma et al [161]	45.33	0.674	✓	-	-	EgoDexter
		25.53	0.763				Dexter+Object
RGBD-based	Dibra et al [163]	-	0.994	✓	-	-	Stereo
		-	0.967				Senze3D
	Sridhar et al [179]	13.24	-	✓	-	10	fingerwave
	Deng et al [180]	17.4	0.74	-	✓	30	NYU
		12.1	0.96				ICVL

3.9 Conclusion

In this chapter, we address the occlusion challenges by employing classical methods that perform numerous techniques to handle this issue. Since the invention of deep learning algorithms, breakthrough results have been achieved mainly in hand pose estimation. Earlier methods have some limitations regarding hardware and smaller data used during training. Therefore, recent approaches based on DNNs are robust, efficient, accurate, and have better generalization. Thus, we explained the significant challenges and classified the most approaches using our proposal taxonomy, noticing the advantages and drawbacks of each work. Then, we presented benchmark datasets with their essential characteristics and evaluation metrics necessary to evaluate the robustness and efficiency of methods. Finally, we listed some works and discussed the results regarding speed(fps), metrics, datasets, and type of CNN architecture.

Chapter 4

Real time handling occlusion in augmented reality based on photogrammetry

4.1 Introduction

Handling occlusion in augmented reality(AR) is a critical challenge to create realistic and immersive AR experiences. Occlusion refers to the ability of virtual objects to appear behind real-world objects, creating a sense of depth and realism in the AR scene. The traditional static vision of AR consists of interaction and navigation through human beings using computer screens. In this context, this chapter focuses on new technology based on photogrammetry and AR strengthening the possibilities in 3D data visualization, navigation, and interaction that address occlusion challenges. The proposed approach falls under the category of a model-based approach, which involves creating an accurate 3D model of real-world objects, typically performed offline.

AR primarily emphasizes real-time processing, high-speed performance, and the seamless integration of virtual elements with rough models of the real world. On the other hand, photogrammetry contributes to the visual realism of the environment in the augmented 3D world. It serves as a powerful tool for extracting precise geometric information from digital imagery and facilitating the rapid, efficient, and accurate alignment of 3D models with imagery data. This synergy between AR and photogrammetry holds promise for enhancing the quality and realism of augmented reality experiences.

Our first contribution was published in a chapter in the book of the Mediterranean International Conference on Pattern Recognition and Artificial Intelligence (MedPRAI), 2021. The published paper is entitled *Real-Time Handling of Occlusion in Augmented Reality Based on*

Photogrammetry [1]. Please visit this link for more information.

4.1.1 Challenges

When striving to achieve a seamless augmented reality experience, three primary categories of challenges emerge:

- **Illumination Problems:** These issues pertain to ensuring consistent lighting conditions between the virtual and real environments. It's essential to render both the virtual and real objects with matching lighting conditions to maintain realism.
- **Camera Parameter Estimation:** When aligning the actual camera with virtual objects, challenges arise in accurately estimating camera parameters. This involves addressing registration problems to correctly position virtual objects within the real-world scene.
- **Occlusion Management:** To provide users with a natural viewing experience, occlusion problems need to be effectively managed. Researchers have proposed various approaches to tackle this issue, including contour-based methods, depth-based techniques, and 3D reconstruction-based methods [181].

Addressing these challenges is crucial for creating compelling augmented reality applications that seamlessly blend virtual and real elements.

Recently, establishing accurate relationships between virtual and real objects has gained significant importance. The issue of occlusion, particularly in AR research, has garnered considerable attention. The ability to correctly depict occlusion between virtual and real objects is of paramount importance for users' comprehension and their perception that virtual objects truly coexist within the physical world. This aspect holds immense relevance for achieving precise and effective augmented reality systems [182]. Consequently, an AR application must possess the capability to autonomously assess and display occlusion between virtual and real objects to deliver a convincing and immersive experience.

4.2 System Overview

This paper introduces an innovative approach aimed at addressing real-time occlusion challenges by harnessing the synergy of two cutting-edge technologies: photogrammetry and Augmented Reality(AR). The primary objective is to tackle issues related to the placement of virtual objects within the context of the real environment in AR systems. Our proposed methodology is structured into two distinct stages:

[1] https://link.springer.com/chapter/10.1007/978-3-031-04112-9_4

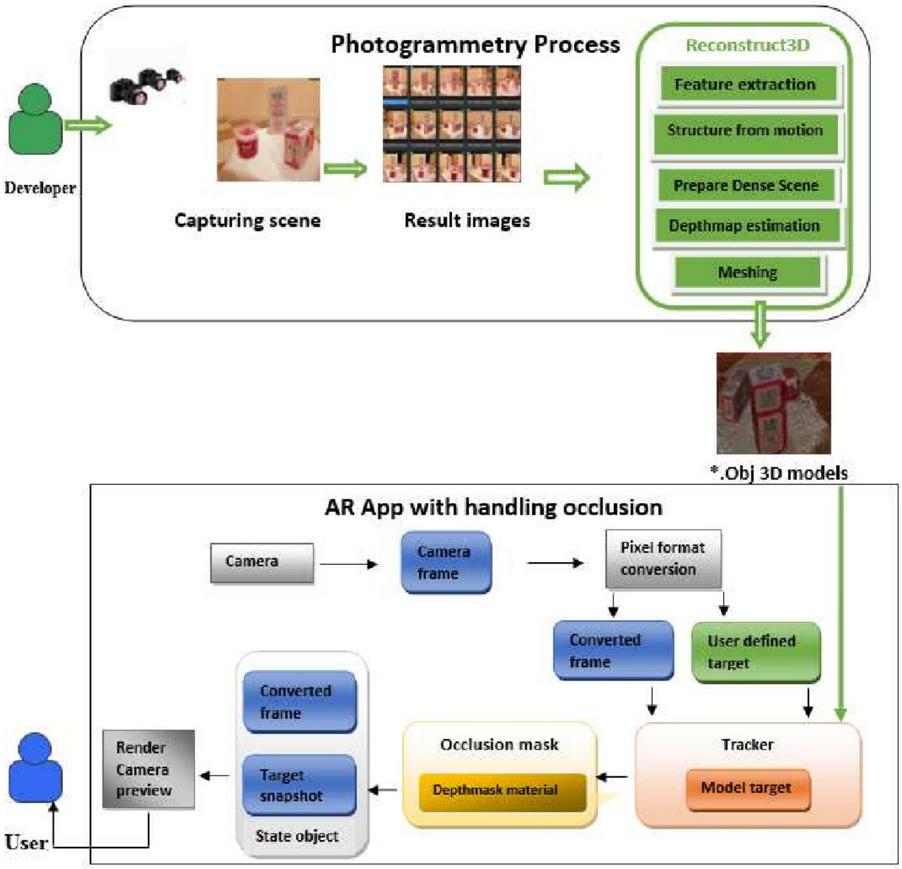


Figure 4.1: System overview of the proposed approach.

- **Photogrammetric 3D Modeling:** Our initial phase revolves around employing close-range photogrammetry techniques to construct a comprehensive 3D model of the real-world scene. This process meticulously follows the pipeline outlined by Alicevision's Meshroom, comprising a structured sequence of 12 nodes. This meticulous modeling approach forms the foundational framework upon which our occlusion-handling solution is built.

- **AR Application Development with Occlusion Handling:** The core of our approach relies on a marker-less tracking system that enables the extraction of key feature points from the real-time, real-world environment, all meticulously registered within the physical space. To facilitate this, we leverage the capabilities of the Vuforia Software Development Kit(SDK), which is instrumental in tracking our model target, obtained through the photogrammetry process for ensuring accurate disparity and accommodating high complexity.

4.2.1 Photogrammertic 3D modeling

Photogrammetry, often hailed as the art and science of deriving precise measurements from photographs, serves as the cornerstone of our approach. It operates on the fundamental principle of extracting the spatial geometry of a scene from an assortment of unordered photographs or videos. While traditional photography reduces the complexity of a three-dimensional scene into a two-dimensional plane, the photogrammetry technique reconstructs three-dimensional models from a collection of images. Our focus lies on the meticulous application of Close-Range Photogrammetry, a specialized branch predominantly employed in industrial contexts where non-contact measurements are required with a high degree of precision, typically ranging from (1 cm^2 to 1 mm^2).

Photogrammetry is the science of making measurements from photographs. It infers the geometry of a scene from a set of unordered photographs or videos. Photography is the projection of a 3D scene into a 2D plane, losing depth information. It aims to create three-dimensional models from a set of images using a known pipeline of photogrammetry. It is classified into three categories: far-range, close-range, and very close-range. As shown in Figure 4.2.

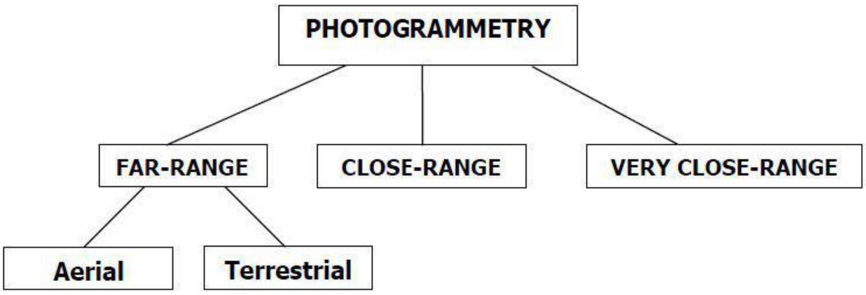


Figure 4.2: Types of photogrammetry pipeline

In this paper, we are interested in Close-range photogrammetry generally applied for industrial applications where non-contact measurements have to be done with a range varying between (1 cm^2 to 1 mm^2). The essence of Close-Range Photogrammetry lies in capturing photographs using digital cameras or even cell phones from relative proximity, typically within a range of 300 meters from the subject. Within this framework, we involve a well-established photogrammetric pipeline, a structured sequence of default nodes that collectively contribute to the creation of precise 3D models.

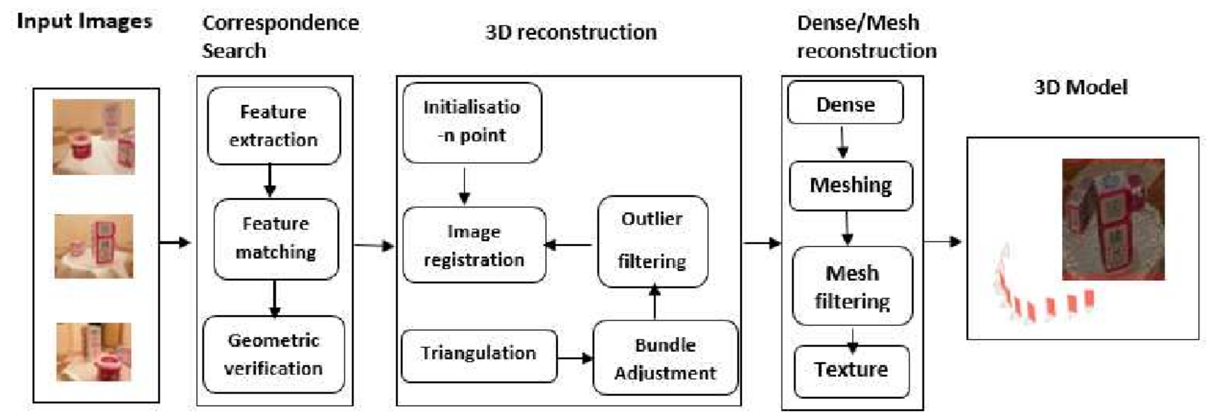


Figure 4.3: The flow diagram of photogrammetric.

CameraInit

represents the essential starting point within the default photogrammetric pipeline. At this initial stage, the process ingests a series of photographs, each captured from distinct viewpoints, serving as the raw data source for subsequent analysis. The outcome of this phase is the generation of (.sfm) files, denoting "Structure from Motion," which encapsulate critical information vital for the photogrammetric reconstruction process.

One of the key tasks undertaken during Camera Initialization is the meticulous characterization of each camera associated with the input photographs. These cameras are defined by a comprehensive set of parameters, including image size (specifying image dimensions in pixels), focal length (determining optical characteristics), intrinsic and extrinsic parameters (encompassing internal properties and external positioning), bundle point coordinates (optimized for accuracy), distortion coefficients (used for distortion correction), and camera/sensor type (indicating the specific camera model) [183].

Feature extraction

Feature Extraction plays a pivotal role in the photogrammetric pipeline, with its primary objective being the extraction of feature sets from pixels that remain relatively invariant despite variations in camera viewpoints during image capture. This ensures that the image scene's characteristics contain consistent features across all images, a crucial aspect of the photogrammetric process. To achieve this, the SIFT (Scale-Invariant Feature Transform) system emerges as a highly effective and widely adopted method for feature detection [183].

The SIFT algorithm can be distilled into four key steps, each contributing to the extraction of robust and scale-invariant features:

1. **Scale Space Generation:** This step involves producing a scale space for an image by convolving it with a Gaussian kernel at different scales, a process known as blurring. The application of the Difference of Gaussian (DoG) technique further refines this process, generating a set of images from the initially blurred ones.

2. **Accurate Keypoint Localization:** Once the scale space is established, the algorithm proceeds to pinpoint the precise locations of keypoints within the images. These keypoints represent distinctive and robust features within the scene.

3. **Orientation Assignment:** After locating keypoints, an orientation is assigned to each of them. This step enriches the feature description, ensuring that keypoints possess not only location and scale information but also orientation information.

4. **Descriptor Computation:** The final critical step involves computing a descriptor for the local image region surrounding each keypoint. These descriptors are meticulously designed to be highly distinctive and invariant, making them resistant to variations such as changes in viewpoint and illumination. Notably, these descriptors are stored in a compact 128-bit format.

Image matching

This node represents a crucial stage within the photogrammetric process, with the primary objective of identifying and pairing images that share similar portions of the scene. This task doesn't involve a detailed feature-by-feature resolution but rather relies on image retrieval methods to find pairs of images that contain comparable information. The overarching aim is to construct a dense image descriptor, facilitating the calculation of distances between all such descriptors.

To achieve this, the approach employs a tree-based vocabulary method, widely recognized for its efficacy in image matching. The process unfolds as follows:

1. **Dense Image Descriptor Generation:** Initially, the focus is on generating dense image descriptors. These descriptors encapsulate the salient characteristics of the images and are crucial for facilitating efficient image matching.

2. **Vocabulary of Tree Approach:** The process proceeds by employing a vocabulary of tree structures. These trees serve as a structured representation of the descriptors. Each node within these trees corresponds to a specific feature descriptor, and the leaves of the trees are associated with individual feature descriptors.

3. **Descriptor Decomposition:** After feature descriptors are extracted and compared, their descriptors are decomposed to the nodes of the vocabulary trees. This step is instrumental in identifying and categorizing descriptors based on their structural attributes within the trees [183].

Feature Matching

This stage offers a pivotal stage in the photogrammetric process, focused on establishing robust correspondences between features in pairs of images that are both accurate and well-suited for this purpose. This critical phase unfolds in a sequence of steps to ensure the reliability of feature matches:

1. **Photometric Matching:** Initially, Feature Matching sets out to achieve photometric matches between sets of feature descriptors within two input images. For each feature detected in the first image, a list of potential matching candidates in the second image is generated. This step aims to identify feature correspondences purely based on visual appearance.

2. **Refinement through Candidate Filtering:** To enhance the accuracy of feature matching, a rigorous filtering process is applied to the candidate matches. Each feature in the first image is cross-referenced with a list of rival features in the second image. Poor matching candidates are systematically eliminated, ensuring that only strong and reliable feature matches persist. This step is underpinned by the assumption that there is one suitable feature match in the second image for each feature in the first image.

3. **Brute Force Descriptor Identification:** Feature Matching leverages the brute force method to identify the two closest descriptors in the second image for every feature in the first image. This meticulous matching approach contributes to the precision of feature correspondences.

4. **Geometric Filtering with Epipolar Geometry:** Beyond photometric criteria, Feature Matching introduces geometric filtering into the process. Leveraging epipolar geometry, this stage further refines feature matches by considering the spatial positions of features within the images. An outlier detection framework called RANSAC (RANdom SAmple Consensus) is employed to robustly identify and discard erroneous matches.

Structure from Motion (SfM) This node is a pivotal stage in the photogrammetric pipeline, tasked with unraveling the intricate relationships between various observations obtained from input images. This multifaceted process culminates in the extraction of the structural scene's essential elements: 3D points, their precise positions in space, the orientation of these points, and the internal parameters of each camera involved.

The SfM pipeline is orchestrated in a deliberate manner to ensure an efficient and accurate reconstruction of objects within the scene. Here's a breakdown of its core steps:

1. **Initial Image Pair Selection:** The SfM process starts by selecting the most suitable initial image pair. This choice is critical as it sets the foundation for the entire reconstruction endeavor. The selected pair should offer robust matches of initial features, forming a solid basis for subsequent reconstruction steps.

2. **Iterative Process:** Following the selection of the initial image pair, an iterative approach is adopted to progressively introduce additional views. This incremental strategy allows for the gradual expansion of the reconstruction and the incorporation of more images into the scene.

3. **Track Creation:** The process of creating tracks consolidates all feature matches between pairs of images. Each track represents a point in space observed from multiple camera perspectives, often containing some outliers. Throughout the SfM process, disjoined tracks are systematically eliminated to maintain data quality.

4. **Essential Matrix Estimation:** The estimation of the essential matrix between two images is a pivotal step. Typically, the first image serves as the origin of the coordinate system. The process involves registering the position and orientation of the second image while triangulating corresponding 2D features into 3D points.

5. **Camera Pose Estimation:** Resectioning comes into play for each new camera introduced into the scene. The Perspective-n-Point (PnP) algorithm is employed within a RANSAC framework to accurately determine the position and orientation of each camera device. This stage significantly validates feature relationships.

6. **Group Adjustment:** After these individual steps, a group adjustment phase is initiated to refine the extrinsic and intrinsic parameters of all cameras, as well as the positions of all points within the scene. This global optimization step ensures that all elements align cohesively.

7. **Triangulation and Candidate Point Selection:** Triangulation processes are conducted to establish new 3D points. This results in an expanded pool of candidate points to be considered for further selection, enriching the reconstruction.

Prepare dense scene

The primary objective of the Prepare Dense Scene node within the photogrammetric pipeline is to achieve several key outcomes essential for subsequent stages of scene reconstruction. It is critical for obtaining accurate and high-quality results in the photogrammetry process.

1. **Undistorted Images:** One of the primary goals of this node is to produce undistorted images. Camera lenses often introduce distortions, such as radial and tangential distortions, to captured images. These distortions can affect the accuracy of subsequent processing steps. To mitigate these distortions, the node applies correction techniques to generate undistorted images. These corrected images serve as a foundation for precise depth calculations and feature matching in the subsequent stages.

2. **Elimination of Re-projection Errors:** In the pursuit of accurate 3D reconstruction, it is crucial to eliminate re-projection errors. Re-projection errors occur when points in the 3D scene are projected back into the 2D images. These errors can be caused by various factors, including lens distortions, calibration inaccuracies, or imprecise camera poses. It aims to minimize or eliminate these errors, ensuring that the 3D points align accurately with their corresponding 2D projections in the images.

3. **Depth Computation:** Another key task of this node is to compute depth information from distortion functions. Depth information is essential for creating a complete 3D representation of the scene. By analyzing the distortions in the images and the known camera parameters, the node calculates depth values for different points in the scene. This depth information is crucial for generating accurate 3D models and understanding the spatial relationships between objects in the scene.

Depth map estimation The Depth Map Estimation node represents a critical phase within the photogrammetry pipeline, tasked with the computation of depth values for each pixel in the images. This depth information is essential for creating accurate 3D representations of the scene. Several methods are employed for depth map estimation, including Block Matching, Semi-Global Matching, and ADCensus. In the context of the Alice Vision Meshroom pipeline, Semi-Global Matching is utilized.

The following are the steps involved in Depth map estimation using Semi-Global Matching:

1. **Camera Selection:** The node begins by selecting the N-best or closest neighboring cameras for each image. This selection is based on front-parallel planes obtained by intersecting the optical axis with pixel selections from neighboring cameras. These neighboring cameras are crucial for computing depth data.

2. **Depth Volume Generation:** A volume (W, H, Z) is generated, representing the depths of each pixel in the image. This volume is populated with depth contenders for each pixel. To estimate the similarity between contenders, a Zero Mean Normalized Cross-Correlation (ZNCC) is applied to each pixel pair. This similarity assessment helps identify the most depth values.

3. **Cost Volume Filtering:** The depth estimation process often introduces noise. To mitigate this, the node applies filtering along the X and Y axes, which groups local cost values. This filtering step reduces the impact of outlier values, enhancing the overall accuracy of the depth map.

4. **Depth Refinement:** After filtering, the node identifies local minima within the cost volume. These minima correspond to potential depth values. The plane index is replaced with the actual depth value from the depth map volume. A refining step is applied to achieve sub-pixel accuracy in-depth calculations.

5. **Parallel Processing:** Depth maps are computed independently for each image, and parallel processing is employed. This approach allows for more efficient computation, although it may require more time overall.

6. **Uniformity Enhancement:** Filtering is applied to the depth maps to improve uniformity across multiple cameras. This step helps ensure that depth values are consistent and accurate throughout the scene [184].

Depth map filter

This node plays a crucial role in refining the depth maps obtained from the previous processing stages, addressing issues related to inconsistencies and occlusions. Its primary purpose is to enhance the overall quality and accuracy of the depth information, particularly in areas that may be affected by occlusions.

The following are the main functions and objectives of the Depth Map Filter node:

1. **Inconsistency Mitigation:** Depth maps generated from different views or images may exhibit inconsistencies or discrepancies in certain regions. These inconsistencies can result from various factors, including variations in lighting conditions, occlusions, or errors in the depth estimation process. This Filter node identifies and mitigates these inconsistencies to create a more coherent and uniform depth map.

2. **Occlusion Handling:** Occlusions occur when certain objects or surfaces in the scene block or obscure others from the perspective of the camera. As a result, depth maps may contain erroneous depth values in occluded areas. This Filter node employs techniques to identify and correct depth values in occluded regions, ensuring that the depth information accurately reflects the scene's geometry, even in occluded areas.

3. **Depth Map Fusion:** In multi-view or multi-camera photogrammetry setups, the Depth Map Filter may involve fusing or merging depth maps from different viewpoints. This fusion process aims to create a unified and consistent depth map that incorporates information from multiple sources. This can help improve the overall accuracy of the 3D reconstruction.

4. **Noise Reduction:** Depth maps may also contain noise or outliers that can adversely affect the quality of the 3D model. This node applies noise reduction techniques to minimize the impact of noisy depth values, resulting in a smoother and more accurate representation of the scene.

Meshing

The Meshing node serves as a pivotal stage within the photogrammetry pipeline, aiming to create a detailed and dense geometric representation of the image scene in three dimensions. This process involves several intricate steps designed to achieve an accurate and comprehensive 3D mesh. Here's an overview of the core objectives and methodologies employed by the Meshing node:

1. **Octree Fusion:** At the outset, this node combines each depth map value into an octree structure. Octrees are hierarchical data structures that divide space into progressively smaller regions. By incorporating depth information into this structure, the node prepares the data for subsequent processing.

2. **3D Delaunay Tetrahedralization:** The Meshing node undertakes a 3D Delaunay tetrahedralization process. This mathematical technique divides 3D space into non-overlapping tetrahedral cells, forming a mesh that adheres to the Delaunay criterion. This criterion ensures that no tetrahedron's circumsphere contains another point from the dataset, contributing to mesh quality and precision.

3. **Weight Calculation:** Within the tetrahedralized volume, the node calculates weights for cells and facets. This involves employing a complex voting method to assign weights that reflect the significance of each element within the mesh. Accurate weight assignment is essential for subsequent steps in the pipeline.

4. **Graph Cut Max-Flow:** To optimize the mesh construction, the Meshing node employs a Graph Cut Max-Flow algorithm to determine the most efficient way to cut the volume, ensuring that the resulting mesh faithfully captures the scene's intricate geometry. Graph cut techniques play a critical role in achieving a precise and detailed mesh.

5. **Mesh Simplification:** Following the initial mesh construction, this node may apply filtering to eliminate sub-optimal or problematic cells on the mesh's surface. This simplification process reduces redundancy among vertices and streamlines the final mesh, resulting in a more efficient representation of the scene. It serves as a pivotal stage within the photogrammetry pipeline, aiming to create a detailed and dense geometric representation of the image scene in three dimensions. This process involves several intricate steps designed to achieve an accurate and comprehensive 3D mesh [184].

Mesh filtering

This node represents a crucial post-processing step following the meshing stage in the photogrammetry pipeline. Its primary purpose is to refine and enhance the quality of the generated 3D mesh by applying various filtering operations. These operations serve to optimize the mesh, improve its visual appeal, and streamline its structure for further analysis and visualization. As following are the key filtering operations typically performed by the Mesh Filtering node:

1. **Smoothing the Mesh:** One of the primary filtering operations involves smoothing the mesh's surface. Smoothing algorithms are applied to reduce irregularities and noise in the mesh, resulting in a more aesthetically pleasing and visually coherent representation. Smoothing helps create a polished and refined mesh that closely resembles the true scene geometry.

2. **Eliminating Large Triangles:** In some cases, the initial mesh may contain overly large or irregular triangles that can affect the mesh's quality and computational efficiency. The Mesh Filtering node identifies and eliminates such large triangles, leading to a more balanced and evenly distributed mesh topology.

3. **Maintaining the Largest Mesh:** While filtering out unwanted elements, it's essential to preserve the core structure of the mesh. The node ensures that the largest and most significant portions of the mesh are retained, allowing for the preservation of critical details and overall scene fidelity.

4. **Removing Small Mesh Components:** Conversely, small and insignificant mesh components that do not contribute substantially to the scene's representation are removed. This step helps simplify the mesh and reduce computational complexity while maintaining the essential features of the scene [184].

4.2.2 AR application development with handling occlusion

Computer vision is responsible for generating 3D virtual objects that align with the same perspective as the real-world images captured by tracking cameras. When it comes to augmented reality image registration, a different computer vision approach, mainly related to video tracking, is employed. Typically, these methods involve two distinct stages: tracking and reconstructing/recognizing. This paper focuses on Vuforia, an augmented reality software development kit(SDK) designed for mobile devices, which plays a pivotal role in facilitating augmented reality applications. Vuforia specializes in the real-time recognition and tracking of planar images and 3D objects. With this capability, developers can accurately position and align virtual objects, including 3D models and multimedia content, concerning real-world objects when viewed through the camera of a mobile device. The virtual object continually tracks the image's real-time position and orientation, ensuring that the viewer perceives the virtual element as an integral part of the actual physical environment. This synchronization creates the illusion that the virtual object seamlessly blends into the real-world scene.

In this context, the Vuforia SDK is utilized to create a model with distinctive features that enable us to determine the precise location of a similar shape within the real environment. Our approach involves crafting a virtual environment that faithfully replicates the real scene down to its minutest details. To explore the concept of occlusion, we introduce virtual objects that seamlessly blend with this environment. Within this virtual environment, we incorporate a virtual camera capable of traversing in multiple directions to comprehensively capture all facets of the scene.

Consequently, we obtain real-time video recordings captured by the physical camera within the actual scene. For each frame of this video feed, we employ coordinate transformation calculations facilitated by Vuforia software to project our 3D model onto the 2D frame. This process allows us to seamlessly integrate the virtual elements into the real-world view.

Additionally, we determine the precise pose of the physical camera by aligning its real-world coordinates with those of the virtual camera. This alignment ensures that the virtual objects are perceived as though they exist at the forefront of the physical environment.

In theory, achieving the correct occlusion relationship between real and virtual objects involves comparing their respective Z coordinates. This comparison allows us to determine which objects should appear in front and which should be hidden behind others, creating a realistic visual effect. One essential component in creating this illusion is the utilization of an occlusion mask. This mask plays a pivotal role in producing the perception of looking through a screen or surface, similar to how our eyes naturally perceive depth and occlusion in the real world.

In our paper, a Depth Mask material provided by Unity3D was employed to effectively manage the occlusion between objects. This material aids in defining which parts of the virtual scene should be obscured by other objects, ensuring that the virtual elements interact convincingly with the real-world environment. Ultimately, the use of such occlusion techniques enhances the overall realism of the augmented reality experience, making it more immersive and engaging for users.

4.3 Results and Discussion

we provide an in-depth exploration of the implementation of our proposed architecture and showcase some of the achieved results.

4.3.1 Experimental results

Data acquisition

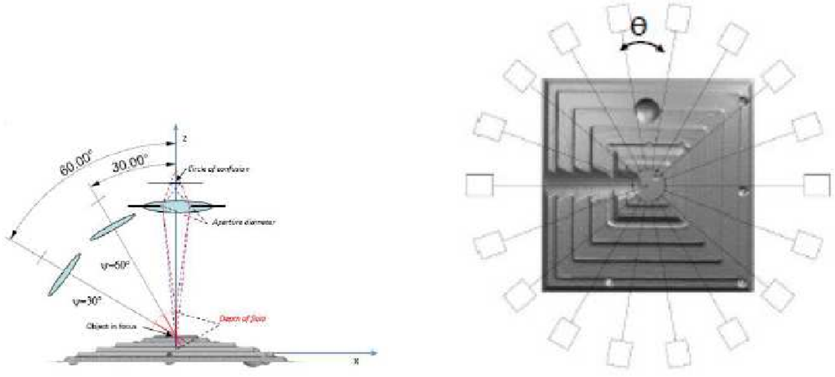
We utilized a Canon camera to capture multiple images of the study scene. The scene we have designed for our study comprises three small objects: a refrigerator, an electric oven, and a cooking pot. It is important to emphasize that in close-range photogrammetry, a key aspect of success lies in precisely defining what needs to be measured and reconstructed when acquiring images. To achieve accurate results, it is recommended that the triangulation of surface points should fall within a specific range, typically between 60 to 110°. In practical terms, this means that consecutive photos should ideally have an angular difference of 20 to 30°between them [185]. In the process of capturing images around an object, it is crucial to maintain a systematic approach to ensure the completeness and accuracy of the data. This typically involves adhering to a clockwise orientation when positioning the camera around the object and capturing images from different heights and viewing angles at each image acquisition position.

To define the location of a camera in this context, two key parameters are utilized:

1. **Tilt Angle (Ψ):** This parameter represents the tilt of the camera relative to the XY plane of the object. It essentially defines the angle at which the camera is inclined or tilted concerning the surface of the object.

2. **Step Angle (Θ):** The step angle refers to the rotating phase of the turning table or platform on which the object is placed. It determines the angular increment or step by which the camera or the object itself is rotated during the image capture process.

These parameters are essential for achieving accurate and comprehensive photogrammetric data for subsequent reconstruction and analysis [183].



(a) Tilt angle (Ψ)

(b) step angle (Θ)

Figure 4.4: Camera Positions Strategy.

Photogrammetry Algorithm

We employ a photogrammetry algorithm, specifically close-range photogrammetry, utilizing the Meshroom software program, as discussed in Section 3.1. This methodology enables us to create a 3D model of the real scene, as depicted in the figure below:

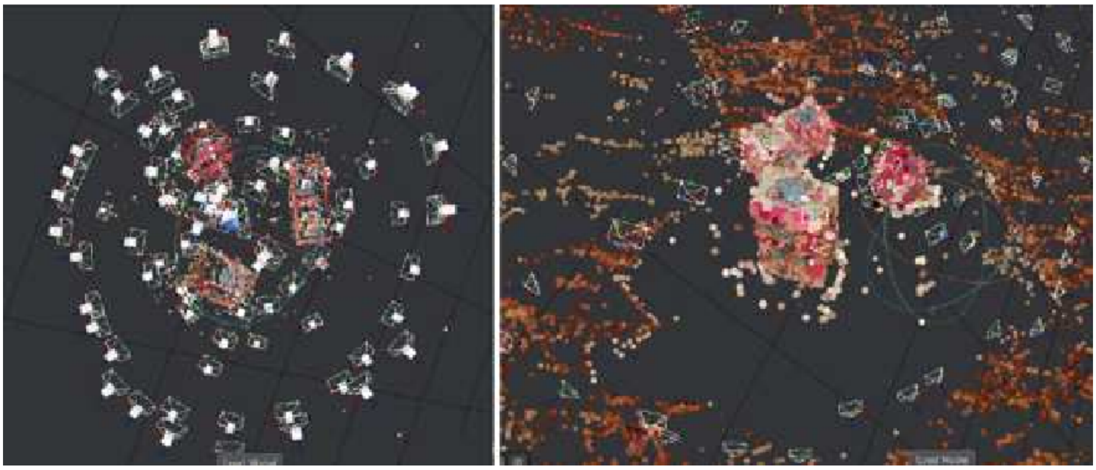


Figure 4.5: The results of photogrammetry pipeline on Meshroom.

AR application

After successfully reconstructing the 3D model of our real scene, we proceed with a crucial post-processing step by utilizing Meshlab. Following this, we employ the Model Target Generator (MTG), a valuable tool that enables us to convert the existing 3D model into a Vuforia Engine database. This database is then utilized by the Vuforia Engine for Model Target tracking.

The Model Target Generator serves several essential functions in this context. It ensures that the spatial features of our model are well-prepared for tracking, establishes the initial snapping positions by defining angles covering 360°from all sides of the objects, and finally, exports the refined model target.

We have developed our AR application using Unity3D software. This application augments the real scene with virtual elements that closely resemble the actual environment. The primary focus of this AR application is to handle mutual occlusion in real-time, ensuring that virtual objects interact convincingly with real-world objects.

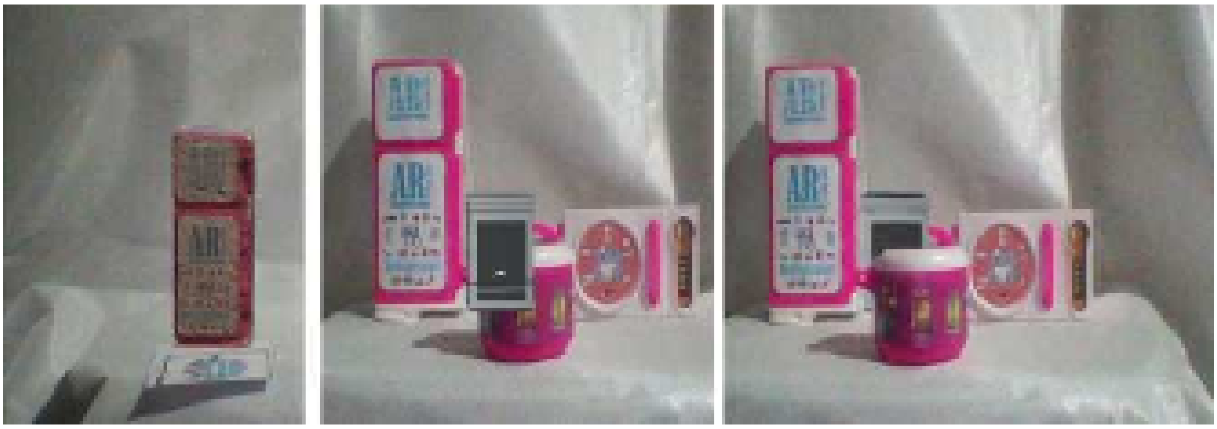


Figure 4.6: (a) AR image. (b) AR occlusion (c) AR handling occlusion

The issue of occlusion in augmented reality is a critical research area, which occurs when objects in the real world block the view of objects located behind them from a specific viewpoint. This phenomenon can lead to user confusion, as real objects may be positioned differently from virtual objects in the augmented reality scene.

From Figure 4.6, we demonstrate the outcomes of our augmented reality application for image tracking as shown in Figure 4.6(a), which is the successful result of our application. However, in Figure 4.6(b), we illustrate incorrect relative relationships between real and virtual objects, highlighting the absence of occlusion handling. Here, it appears that the virtual object is erroneously positioned in front of the real-world objects. After applying our method (Figure 4.6(c)), the occlusion problem is resolved, and the virtual objects are correctly positioned about the real-world objects. This demonstrates that using an occlusion mask is essential, particularly when parts of our model target are obscured by virtual object components. In such cases, occlusion masks effectively solve the occlusion problem in real-time, leading to a more realistic and coherent augmented reality experience.

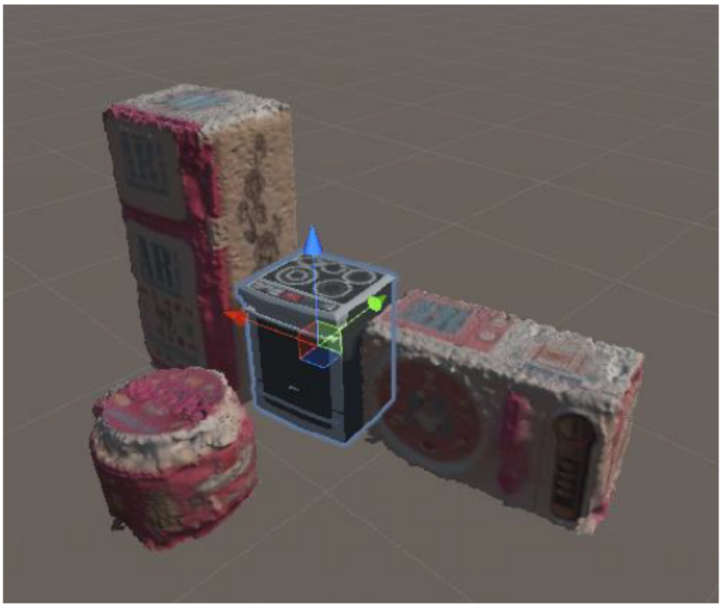


Figure 4.7: Resolving occlusion using Shader dapthMap material provided in Unity3D

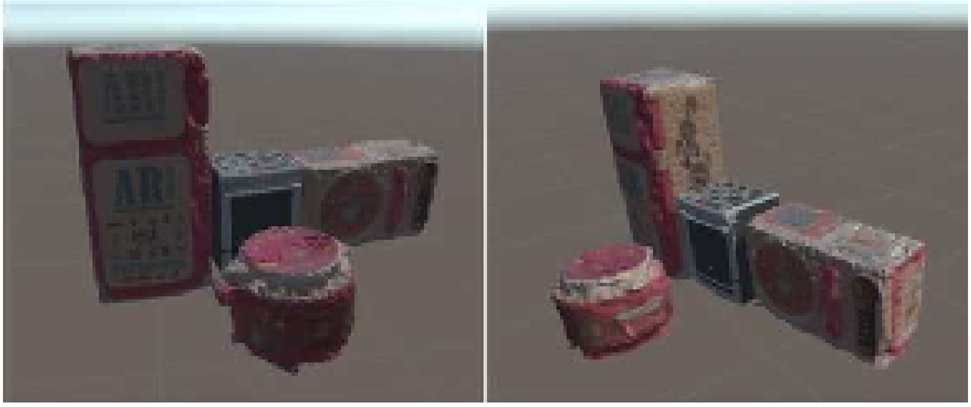


Figure 4.8: Handling occlusion in Augmented reality app with different point views in real-time using Unity3D

4.3.2 Discussion

In this section, we will validate the results obtained through our approach, which combines two primary objectives: photogrammetry and augmented reality, with a specific focus on managing augmented reality occlusion in real time. To achieve this, we employed the Meshroom software to implement a photogrammetry pipeline composed of 12 nodes. This comprehensive algorithm was designed to ensure high-performance output for the 3D model. Through our validation, we aim to showcase the success of our photogrammetric approach in creating a detailed and visually accurate representation of the scene.

Numerous interconnected parameters play a pivotal role in enhancing the quality of the photogrammetric process. Initially, we achieved remarkable measurement precision through the strategic placement of a single camera, carefully selected to ensure a comprehensive view, resulting in accuracy as fine as 5um. This positioning not only bolstered the model's efficiency but also expedited the extraction of detailed feature data from data points in a notably brief time frame. Furthermore, during testing across diverse scenarios, we consistently attained reliable performance.

The camera's resolution emerged as a critical factor, profoundly influencing the calibration of images utilized within our software and, consequently, yielding superior outcomes. Additionally, the dimensions of objects within the scene were tailored to generate a highly refined reconstructed model. The quality of the photogrammetric mesh hinged on an array of parameters, including the number of images, the default settings inherent to the photogrammetry methodology, camera calibration, and an image acquisition protocol.

Notably, one pivotal parameter involved capturing sequential photographs at angles no less than 60 °, with an ideal intersection angle of 90°, thereby heightening recognition accuracy.

As previously discussed, we employ Vuforia for tracking real-world objects and determining the camera position, facilitating the development of an augmented reality application aimed at mitigating occlusion issues. The visualization of the 3D model through augmented reality yields highly accurate and visually appealing results. Our primary goal is to advance the notion of image-based 3D modeling by leveraging close-range photogrammetry to effectively address occlusion challenges, enabling the creation of realistic real-time visual experiences.

To evaluate the outcomes of our approach, we conducted a comparative analysis based on five crucial criteria to demonstrate the superior efficiency of our system compared to previous methods. Firstly, we consider the applicability of the scene. In practical scenarios, the methods proposed by **Sanches et al.** [116] and **Lu and Smith** [124] involve dynamic objects, while **Lepetit et al.** [125] approach cannot accommodate such scenarios. Secondly, we evaluate the criteria related to the versatility of handling occlusion across a wide range of viewing angles and volumes. **Lepetit et al.** [162] method imposes restrictions on the viewpoint, significantly limiting its practicality. In contrast, our system, along with **Sanches et al.** [116] and **Lu and Smith** [124] approaches, offers flexibility in choosing viewpoints. Thirdly, we consider the equipment necessary for implementing the application. **Sanches et al.** [116] employ Fiducial markers, efficiently addressing the registration problem in AR using optical tracking and overlaying virtual objects onto markers to align them with the real environment. In our work, we utilize a DSLR camera to capture sets of photographs and the Vuforia SDK for real-time recognition, tracking of planar images and 3D objects, and camera pose estimation. Lastly, real-time performance is a critical requirement for all these methods. Additional details are provided in Table 4.1

Method	Scene	Viewpoint	Equipment	Handling occlusion	Real-time
Sanches et al. [116]	Dynamic	Arbitrary	Fiducial markers	Yes	Yes
Lu and Smith [124]	Dynamic	Arbitrary	No	No	Yes
Lepetit et al. [125]	Static	Restricted	No	No	Yes
Our method	**Static**	**Arbitrary**	**Vuforia SDK**	**Yes**	**Yes**

Table 4.1: A comparison between the proposed method and the previous methods.

4.4 Conclusion

Through the exploration of outcomes, this chapter has effectively showcased the potential of employing a photogrammetry algorithm for an efficient occlusion handling approach centered around 3D reconstruction for augmented reality applications. The fundamental idea revolves around addressing mutual occlusion using two distinct technologies: augmented reality and photogrammetry. Additionally, we have developed an AR application using the popular SDK software, Vuforia, enabling us to extract key scene features and estimate camera poses for precise alignment within our AR systems. Consequently, we have effectively managed occlusion by real-time comparison of the third coordinate between real and virtual objects through the use of masks. In the experiment, we created a small dataset composed of different viewpoints of the real scene involving small objects The results demonstrate its capability to meet the demands of accuracy, efficiency, and real-time performance in augmented reality applications with resolving occlusion in the major scenario.

Chapter 5

Hand pose estimation based on regression method from monocular RGB cameras for handling occlusion

5.1 Introduction

The human hand plays a fundamental and instinctive role in enabling our interaction with the physical world. In this chapter, we propose our original contribution for hand pose regression architecture, specifically designed to learn how hands interact with objects and overcome occlusion challenges. Our overall pipeline comprises three primary stages: In the first stage, we employ a ResNet34 network as the backbone to extract feature maps from the cropped RGB image. This initial step is crucial for capturing essential information from the input image. In the second stage, the network is trained to produce non-normalized 2D heatmaps representing the locations of hand joints. To ensure that these heatmaps reflect the probability map of each joint accurately. Additionally, we introduce a novel approach called Latent Heatmap Representation(LHR) to regress the 2D coordinates of hand joints by converting feature maps output into 2D coordinates. In the third stage, we incorporate convolutional layers into the network to process the generated heatmaps further. By concatenating the intermediate feature maps obtained in this stage, we can estimate the global 3D pose of the hand. Finally, we regress a hierarchical tree structure representing the hand as a tree to predict the configuration of hand bones within the cropped hand. Our proposed architecture is designed to effectively handle occlusion challenges and improve the accuracy of hand pose estimation, particularly in scenarios involving hand-object interactions.

We conducted an extensive evaluation of our proposed method using three benchmark datasets [26, 27, 37]. Our analysis included both quantitative and qualitative evaluation, and the results improved the competitive performance of our approach when compared to numerous state-of-the-art methods. Importantly, our method achieves this high level of performance while maintaining optimal execution times. It demonstrates its ability to accurately predict the 2D locations of hand joints and the 3D hand bones, even in challenging scenarios characterized by self-occlusion and object occlusion. These findings underscore the robustness and efficiency of our proposal in handling complex real-world situations.

This contribution was published in a Multimedia Tools and Applications Journal 2023. The published article is entitled *Hand pose estimation based on regression method from monocular RGB cameras for handling occlusion* [1]. Please visit this link for more information.

5.2 Methodology

Our main objective is to propose a novel deep-learning architecture to estimate 2D and 3D poses from a single RGB image destined to resolve the occlusion problem. The 3D hand pose is represented by a sequence of 3D joint coordinates,$\Phi^{3D} = \{\phi\}_{k=1}^{K} \in T_{3D}$ where T_{3D} is 3-dimensional hand joint location, with K= 21. The 2D hand pose estimation is depicted by a two-dimensional array joint coordinated, where $\Phi^{2D} = \{\phi\}_{k=1}^{K} \in S_{2D}$ is the K-dimensional hand joint space with K= 21.

The proposed "ResUnet" framework combines ResNet-34 layers with Unet as its foundational network. This combination is particularly effective for tasks where the input and output have similar dimensions. The Unet network consists of two primary pathways. The first path utilizes a pre-trained ResNet-34, which is a 34-layer ResNet network, to extract key features from an RGB cropped hand image denoted as $I \in +\mathbb{R}^{128\times128\times3}$, as indicated in Table 4.1. The second path of Unet, known as the Expansive path, includes four consecutive multi-feature blocks combined with upsample Blocks referred to as Unet-Blocks, as illustrated in Figure 5.1. The initial Unet-Block takes two features, denoted as $\mathbb{F}_g = \mathbb{F}_4, \mathbb{F}3$, fuses them as input, and produces grouped features denoted as $\mathbb{F}out$. We employ bilinear upsampling to enhance the quality of the input images and obtain multi-scale features. Subsequently, these features are used for estimating 2D heatmaps before being passed on to the next Unet-Block.

The following Unet-Blocks follow a similar structure but use distinct input features. For each block, the upsampling layer is concatenated with the corresponding feature vector, denoted as $\mathbb{F}_{skip}$. These concatenated features are later fed into the convolutional layers for further processing.

[1] https://link.springer.com/article/10.1007/s11042-023-16384-9

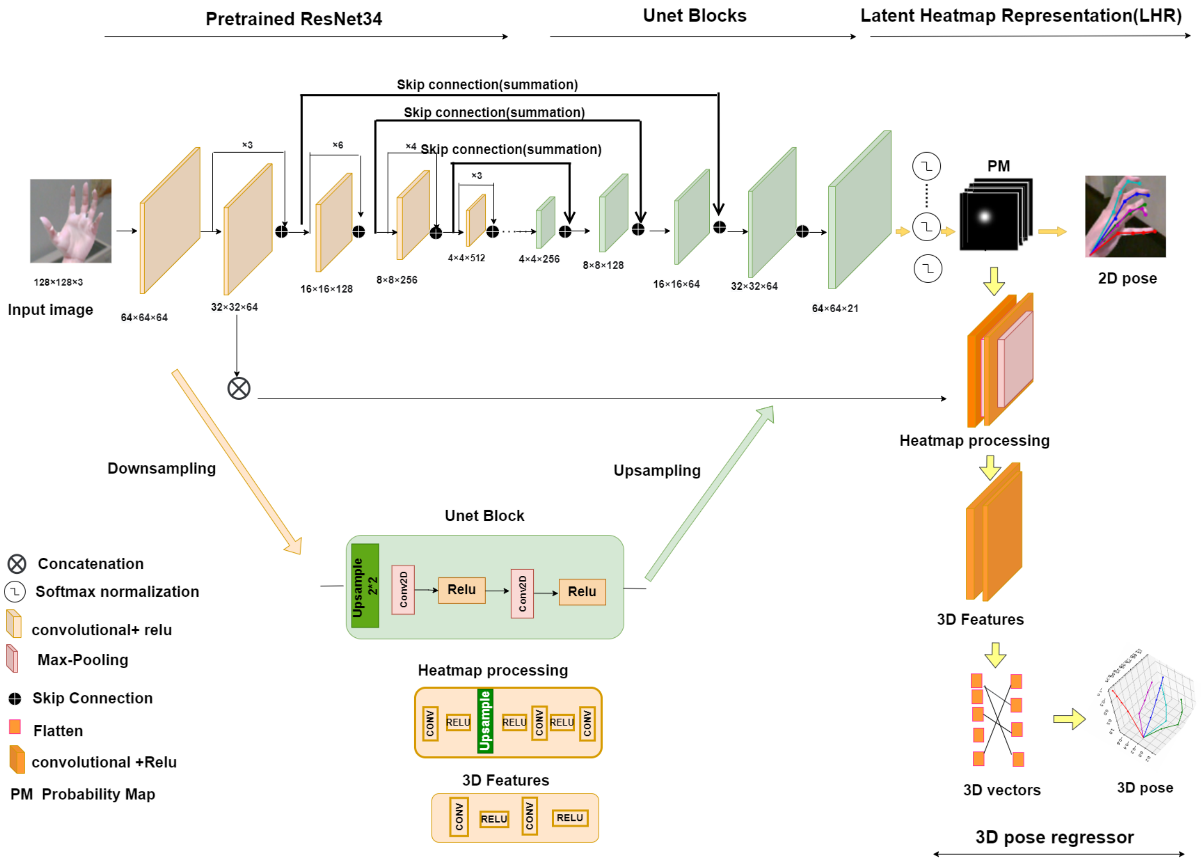


Figure 5.1: The overall pipeline of our architecture for estimating 2D and 3D pose regression.

The output of the last block has a size of 64. Furthermore, we add an upsample layer with bilinear mode and two convolution layers separate with RELU as an activation function to obtain nonlinear transformation and enhance our architecture ability to fit data. The output is a tensor of size(21, 64). We get the 2D feature heatmaps from the Unet blocks, including pose data from intermediate outputs.

During the training process, we generate 2D heatmaps at each stage produced by each of the Unet-Blocks. However, during the inference phase, we exclusively utilize the final output results and normalize them using the Softmax function. This normalization step converts the feature maps into normalized heatmaps, which is essential for predicting 2D hand joint locations, and this prediction is achieved using a Latent Heatmaps Representation (LHR).

Specifically, we employ two convolutional layers followed by the Rectified Linear Unit (ReLU) activation function and Max pooling. This processing is aimed at further refining the feature heatmaps and preparing them for concatenation with intermediate features, ultimately resulting in the generation of 3D features. As mentioned in Equation 1. This combination of processing steps enhances the accuracy and robustness of predicting the 2D hand joint locations using the Latent Heatmaps Representation.

$$output = concat([features_{local}, UnetBlock3], dim = 1) \tag{5.1}$$

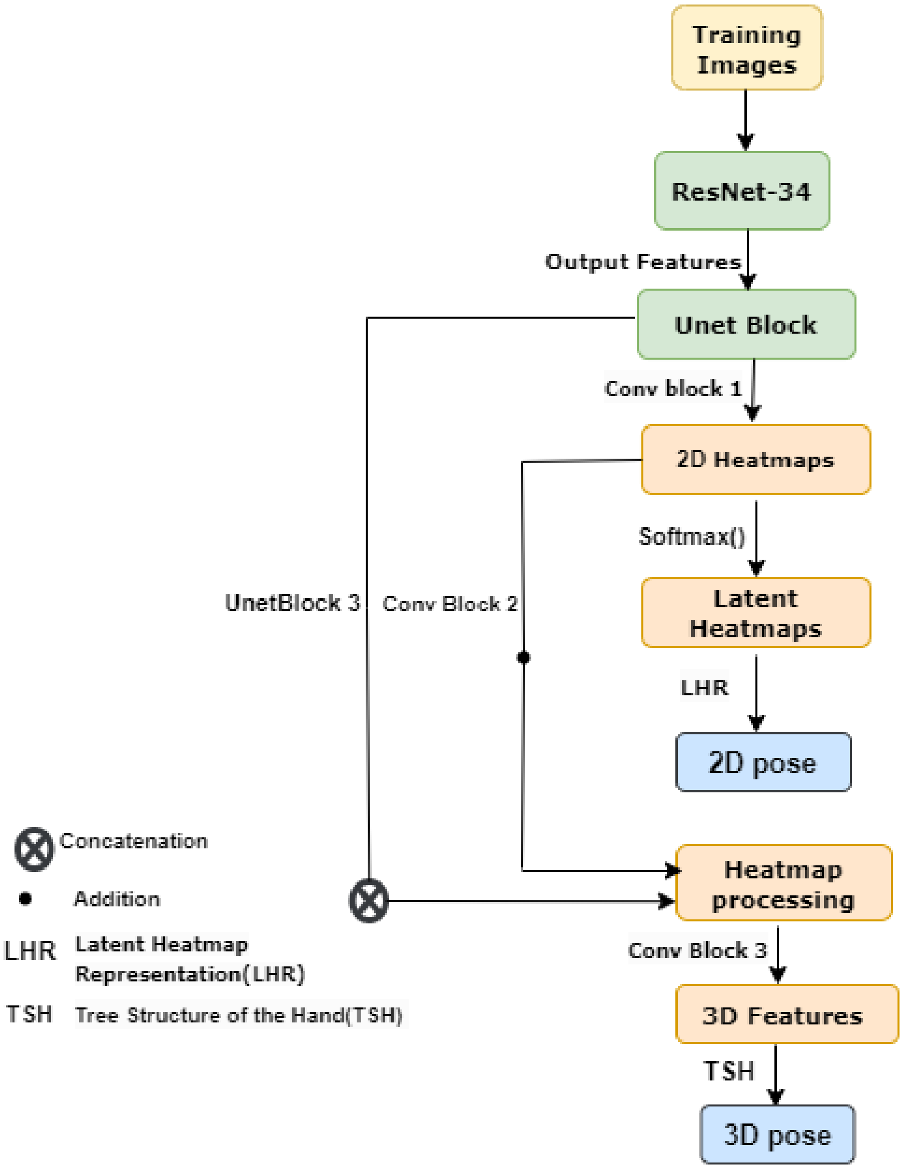


Figure 5.2: The sequential flow of our architecture.

To make our proposal more comprehensible, we have devised a sequential flow that outlines the key stages from the input image to the desired outputs. It is essential to mention that Figure 2 offers further insights and detailed explanations of these steps, providing a visual aid to enhance understanding.

To further estimate 3D hand pose coordinates, we introduce two additional convolutional layers with dimensions (128, 64) and (64, 32), respectively. A flattening operation follows these layers, and two fully connected layers process the resulting tensor. This process ultimately produces a tensor with a size of (3, 21), which represents the estimated 3D hand pose coordinates.

To compute the bone lengths of the hand, we employ a tree structure that defines the relationships between different hand joints. We calculate the length of each bone as the difference between two adjacent joints. This method allows us to predict the lengths of 20 bones in the hand, providing valuable information about 3D hand pose.

5.2.0.1 Image Feature Extractor

We utilize the ResNet-34 layer network, as introduced in the research by He et al. [148], to extract key features from cropped images. This choice is motivated by several factors. Firstly, ResNet-34 represents a more recent iteration within the family of Deep Residual networks and is recognized for its lightweight nature in convolutional neural networks. Moreover, the 34-layer ResNet architecture notably reduces training errors and performs well on validation data. It offers faster training compared to ResNet-50 and consumes less memory. This underscores the effective handling of the degradation issue in this context and highlights the potential for enhanced precision through increased network depth. As depicted in Table 4.1, we can find a comprehensive overview of the backbone architecture of ResNet-34.

Layer	Feature size	Output	kernel	Stride	Padding
Conv2D	128	64	3x3	2	1
BN	64	64	-	-	-
RELU	64	64	-	-	-
Max-pool	64	64	3	2	1
ResNet-Block1	64	64	3x3	1	1
ResNet-Block2	64	128	3x3	2	1
ResNet-Block3	128	256	3x3	2	1
ResNet-Block4	256	512	3x3	2	1

Table 5.1: The network architecture of ResNet-34

5.2.0.2 2D Pose Regression Network

The most effective method identified for estimating 2D hand poses involves a novel concept known as Latent Heatmap Representation(LHR). This approach surpasses others in performance and is considered the most dependable means of predicting 2D pose coordinates by generating 2D heatmaps. LHR offers several notable advantages over alternative methods:

- **Addressing Occlusions:** LHR excels in handling occluded hand poses, a challenge that can be problematic for alternative techniques. It maintains accuracy even when parts of the hand are obscured.

- **Continuous Location Estimation:** This representation enables the continuous estimation of joint locations, providing more precise and fine-grained pose predictions than discrete methods.

- **Utilizing multiple training data sources:** LHR allows for learning from various training data sources, enhancing the model ability to generalize and adapt to different hand poses and variations.

In this approach, the hand pose is depicted as a 2D latent heatmap, where the value assigned to each pixel indicates the likelihood of the corresponding joint presence at that particular location.

To tackle this challenge effectively, we developed a deep-learning architecture that takes a single RGB image as input and generates a set of K-low-resolution 2D heatmaps as output. Typically, this network is trained using three benchmark datasets along with their corresponding hand pose annotations, allowing it to learn how to construct a 2D latent heatmap accurately.

According to the foundation laid by Iqbal et al. [30], introduced a heatmap representation involving the generation of 2.5D heatmaps, which include 2D heatmaps for 2D keypoint localization and depth maps for each joint to predict depth values, our innovation lies in the development of a novel architecture that combines ResNet-34 with UnetBlocks to generate 2D heatmaps directly from single images, eliminating the need for depth values for each hand image. However, the approach presented in [30] was based on a CNN architecture that learned 2.5D heatmaps in a latent manner using the softmax function. This function converts the heatmaps into 2.5D coordinates in a distinct way.

Subsequently, a constrained normalization approach was employed to reconstruct the scale-normalized absolute 3D pose directly. The global structure of the Latent Heatmap Representation(LHR) is presented in Figure 5.3. In this context, a single image is the input, as previously described. Once the network processes the input image, it employs a technique known as the UnetBlock with a skip connection to generate a 2D heatmap. The essential advantage of using LHR is that it achieves a higher output resolution, significantly enhancing the precision of locating each hand joint.

The network output saves the learned model features as latent variables denoted as F_k^{2D}, which are used to approximate 2D latent heatmaps.

After the 2D heatmap is generated, the values in each channel of the heatmap, corresponding to different keypoints on the hand, are normalized using a spatial softmax function. This function transforms the tensor values into a probability distribution, ensuring that the sum of values in each channel equals one. This normalization guarantees that the probabilities assigned to each hand keypoint are consistent and can be interpreted as a 2D probability map. The model enables 2D coordinates of the hand joints with a high degree of accuracy, following the formula:

$$PM_i(p) = \frac{exp(\beta_i F_i^{2D(p)})}{\sum_{p' \in \Omega} exp(\beta_i F_i^{2D(p')})} \tag{5.2}$$

Where p represents a point on the probability map, Ω is the sequence of all pixels on the 2D feature map F_i^{2D} of the i^{th} joint and β_i is a learnable factor that controls the spread of the probability map.

The 2D coordinates of the k^{th} keypoint are then computed as the weighted mean for the x and y coordinates, where the weights come from the normalized heatmaps, and the generated x, y coordinates fall within the range [0, image width], as shown in the formula:

$$p_k = \sum_{p \in \Omega} PM(p).p \tag{5.3}$$

In essence, this process allows for precise estimation of the 2D joint coordinates of the hand keypoints using the information encoded in the heatmaps.

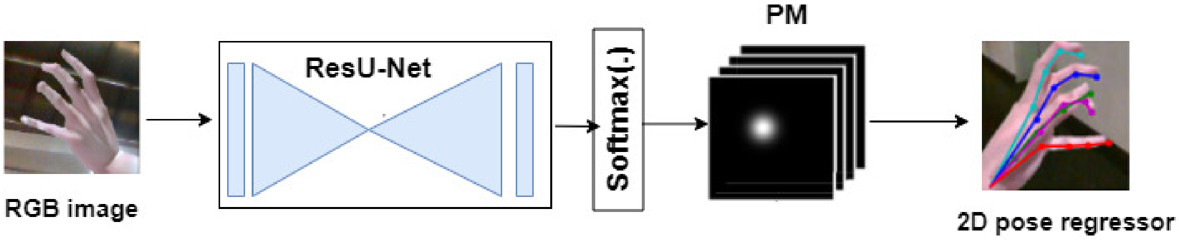


Figure 5.3: The ResU-Net architecture used to estimate the 2D pose regression via the Latent Heatmaps Representation(LHR). PM represents a probability map after applying a Softmax() function.

5.2.0.3 3D pose Estimation Network

The third branch within our comprehensive framework is focused on regressing 3D hand pose. This task follows the previous stage, where we obtain 2D joint locations as latent heatmaps. In this phase, we adopt a representation based on a tree structure of the hand, illustrated in Figure 5.4. This representation aims to predict the bones of the hand rather than individual joints, as it offers greater accuracy and stability.

To maintain consistency in notation, we define each bone as $\beta_k = \{\beta_k | k = 1, ...k\}$. This notation helps us establish a clear and consistent framework for describing and predicting the 3D pose by focusing on the relationships and lengths of the bones, which can provide a more robust and precise characterization of the hand.

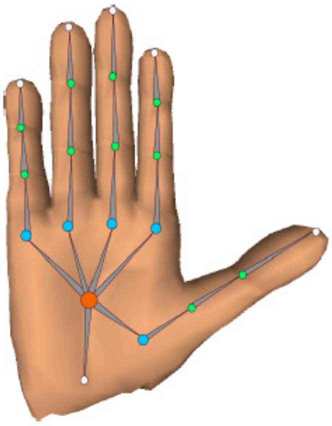


Figure 5.4: The proposed hand bones representation.

For each joint denoted as J^{th}, we define its associated bone as a direct vector pointing from that bone to its origin. This relationship can be expressed through the following equation:

$$B_i = J_i - J_{\text{parent}(i)} \tag{5.4}$$

Here, J_i represents the position of the current joint J_i, and $J_{\text{parent}(i)}$ is determined by a predefined function that returns the index of the parent joint for the current joint J_i. In Figure 4.5, we can observe the representation of the hand skeleton, consisting of 21 joints.

The bones, denoted as β_k, are calculated as the difference between the positions of the keypoints that make up each bone. This computation is based on the geometric structure of the hand, allowing us to derive the bone vectors from the joint positions, thereby providing a detailed skeletal hand structure.

The backbone structure of this branch consists of two convolutional layers, with each layer, followed by a Rectified Linear Unit (ReLU) activation function to generate 3D feature maps. Max-pooling layers are placed after every pair of convolutional layers. Then, we concatenate the resulting 3D features with intermediate features. The final step involves flattening the data using two fully connected layers. The output is a tensor with dimensions (20, 3) representing the 3D coordinates of K=20 bones.

To compute the global coordinate system of a specific joint, we sum the local coordinates of all bones along the path that connects to that joint.

In the learning process, we supervise the bones. We introduce the bones loss L_β, which is determined by the following equation:

$$L_\beta = \frac{1}{2}\sum_{j=1}^{J}\left\|\beta^{pred} - \beta^{gt}\right\|^2 \tag{5.5}$$

Where L_β represents the bone loss, β^{pred} is the predicted bone, β^{gt} is the ground truth bone, and J represents the number of joints. This loss function quantifies the difference between the predicted and ground truth bones, serving as a crucial supervision signal during training to refine the 3D pose estimation of the hand.

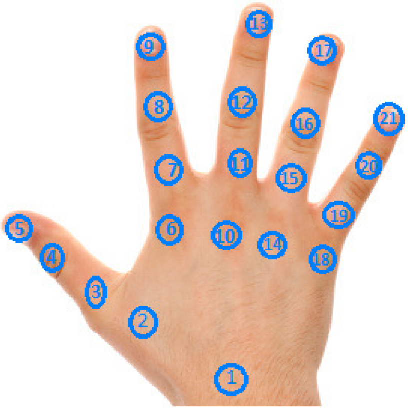


Figure 5.5: Illustration of hand Skeleton and joints index. 1 indicates the wrist joint; 2-4 indicates the thumb finger joints; 7-9 indicates the joints of the Index finger; 11-13 indicates the joints of the middle finger; 15-17 indicates the joints of the ring finger; 19-21 indicates the little finger.

5.3 Implementation Details

In our method, we conducted experiments on station computers using various software and hardware components. Specifically, we implemented our approach using the following:

- **Deep Learning Library:** we utilized PyTorch version 1.8 as our deep learning framework [186]. PyTorch is well-known for its flexibility and ease of use in developing neural network models.

- **GPU Acceleration:** To accelerate training and inference processes, we used CUDA version 10.1 and cuDNN version 7.6.4. These technologies are designed to leverage the computational power745 of NVIDIA GPUs.

- **Graphics Card:** We employed the NVIDIA GeForce GTX 1070 graphics card, a 64-bit GPU, to enhance the speed and efficiency of our model training.

- **System Configuration:** Our overall training process was conducted on a system equipped with an AMD Ryzen 5 3600 6-Core CPU and 16GB of system memory(RAM).

This combination of hardware and software components allowed us to conduct experiments successfully and train our model effectively, leveraging the power of GPUs to expedite the training process.

5.3.1 Data pre-processing

An essential step in the data preparation involves splitting the entire dataset into training and validation sets. The validation set is crucial in monitoring validation loss and deciding when to stop the model training process. On the other hand, the test set is reserved for the final evaluation of the trained model performance. Before feeding the images into the network, we applied some preprocessing steps. We employed a squared crop technique, calculating the pixel distances between the leftmost and rightmost keypoints and the distances between the lowest and highest keypoints in each image. We then selected the maximum of these distances. This approach ensures that we obtain a square region of interest that encompasses the entire hand. After cropping, we used bilinear interpolation to resize the input images to a consistent size that helps maintain the quality and information content of the images while resizing them to a standardized format.

These preprocessing steps prepare the data for training and ensure that the model receives input images in a consistent and suitable format for accurate learning and inference.

We incorporated data augmentation into an available dataset, a common strategy in deep learning and computer vision tasks. This technique is precious when the labeled data available for training is limited or insufficient to create a robust and accurate model. Additionally, data augmentation helps mitigate overfitting, a phenomenon where a model becomes overly specialized in memorizing the training data instead of generalizing to new and unseen data.

By artificially expanding the size and diversity of the training data through data augmentation, we can prevent the model from memorizing specific examples and, instead, encourage it to learn more robust and generalizable features. This results in a model better equipped to handle a variety of data inputs and perform well on unseen examples, ultimately improving its overall performance and reliability. We implement various augmentation techniques for each image in the GANerated and SynthHands datasets as described below:

- We extensively utilize ColorJittern, which introduces random adjustments to an image's brightness, contrast, and saturation, falling within a range of -25% to +25%.

- Gaussian Blur, a commonly employed image processing technique, induces blurriness in the images. This is achieved by convolving the image with a Gaussian filter, where pixel weights are determined based on their proximity to the filter center. We apply a range of Gaussian filter radius, from 0 to 0.8, to control the degree of blurriness. A radius of 0.8 corresponds to a substantial blur effect of 800 pixels.

- We standardize the pixel values, ensuring they have a mean of zero and a standard deviation of one. This standardization facilitates the model data processing. We specify the mean as [0.485, 0.456, 0.406] and the standard deviation as [0.229, 0.224, 0.225], scaling them to the range [0.0, 1.0].

Furthermore, we incorporate specific transformations into the SynthHands dataset, including random scaling that alters the image and 2D coordinates. To ensure consistency in the order of hand keypoints across datasets, it is essential to estimate 2D joint locations as 2D vectors. In our work, we adopt the same keypoint order as presented in the SynthHand dataset by Müeller et al. [26]. After applying the scaling operation, the 2D keypoint coordinates (x and y) change based on the following formulas (Equations 6 and 7), with x ranging from 0 to the image width and y ranging from 0 to the image height:

$$x_{new} = (x_{old} * scale\_factor) - (crop\_width/2) \tag{5.6}$$

$$y_{new} = (y_{old} * scale\_factor) - (crop\_height/2) \tag{5.7}$$

In these equations, x_{old} and y_{old} represent the original pixel coordinates of the keypoint, while $scale\_factor$ denotes the randomly applied scaling percentage to the image. Additionally, $crop\_width$ and $crop\_height$ represent the dimensions of the center-cropped image.

The initial part of the formula scales the pixel coordinates based on the scaling factor, while the subsequent part adjusts the pixel coordinates to align them with the center of the cropped image. This dual operation ensures that the keypoint locations are accurately updated to account for both the scaling and cropping of the image.

Moreover, we introduce rotations for color images and incorporate 3D coordinates into the SynthHands dataset. The 2D coordinates are generated by projecting the 3D keypoints onto the image plane using an intrinsic matrix.

5.3.2 Dataset

We evaluate quantitatively our proposed framework using three available datasets:

1. **GANerated dataset** [27] stands as a prominent RGB-based dataset that captures hand poses during interactions with objects, even when the hand is partially obscured. This dataset comprises approximately 330,000 synthetic images depicting hand poses, each meticulously annotated with three-dimensional information, with 21 distinct joints.

   The creation of these synthetic images was facilitated by employing a CycleGAN network, which not only generated the images but also provided corresponding annotations. These annotations were generated using sophisticated artificial intelligence techniques, ensuring the accuracy and utility for hand pose estimation research.

2. **SynthHands** [26] represents a robust RGBD dataset for hand pose estimation. This extensive dataset boasts an impressive collection of 63,500 images, each comprising color and depth information and boasting a high resolution of 640x480 pixels. These images were captured using an Intel RealSense scene camera from five egocentric viewpoints. This dataset encompasses various variations, including skin color, hand shape, background complexity, and challenging scenarios involving wrist and arm rotations. SynthHands introduced intricate hand-object interactions, featuring seven distinct object shapes and 145 textures.

   One of the key strengths of this dataset lies in its provision of precise and reliable ground truth data. It annotates each image with accurate 3D position data for 21 distinct key-points, essential for training and evaluating hand pose estimation algorithms. This dataset is a valuable resource for researchers, enabling the exploration and advancement of hand pose estimation, gesture recognition, and related computer vision applications in various real-world contexts.

3. **Stereo Hand Pose Tracking Benchmark(STB)** [37] is a widely utilized resource for training and validating RGB-based 3D hand pose estimation methods. This dataset comprises a substantial collection of 18,000 images, split into 15,000 for training and 3,000 for testing. It includes stereo and depth images with a resolution of 640x480 pixels.

   The stereo images were captured using a Point Grey Bumblebee2 stereo camera, which allows for capturing images from two slightly different viewpoints, enabling depth perception. Further, depth images were obtained using an Intel RealSense F200 depth camera, providing further depth information for accurate hand pose estimation. It is composed of 12 sequences with six distinct background settings.

   For accuracy and precision, the STB dataset provides annotations for the 3D positions of 21 key hand joints. These annotations serve as ground truth data, facilitating the training and evaluation of hand pose estimation algorithms. As illustrated in Figure 5.6, we represent some images from the three available datasets.

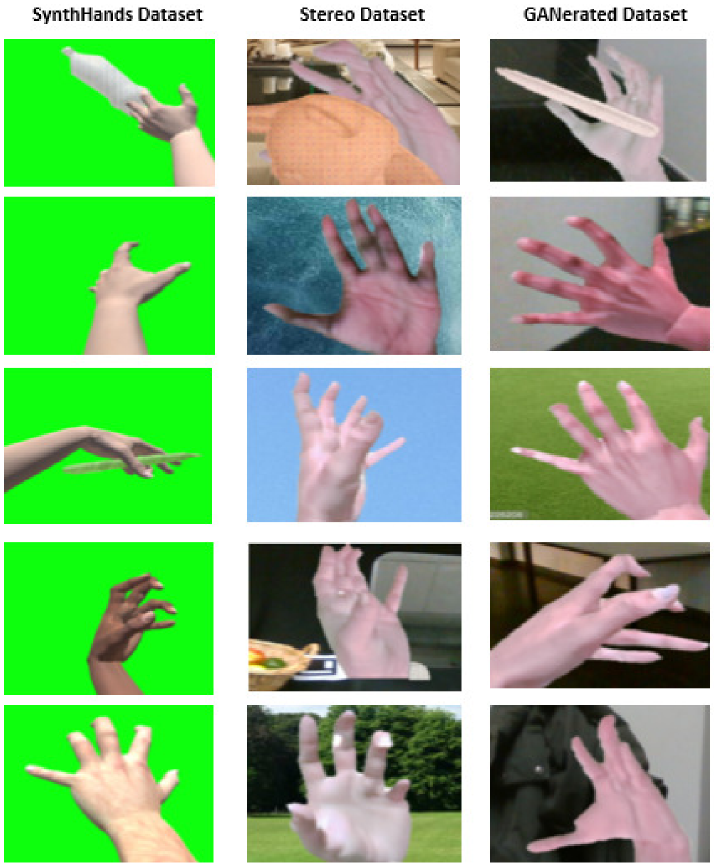


Figure 5.6: Some sample images from the GANerated, Stereo Hand Pose Tracking Benchmark(STB), and SynthHands datasets.

5.3.3 Metrics Evaluation

To evaluate the precision of our proposal and to make a meaningful comparison with state-of-the-art methods, we employ the three most commonly used metrics in the realm of hand pose estimation:

- **End-Point-Error(EPE):** The EPE metric quantifies the mean 3D Euclidean distance error between the calculated joint positions and the ground truth. In 3D hand pose estimation, these distances are typically expressed in millimeters(mm), while in 2D, they are measured in pixels(px). Mathematically, EPE is defined as:

$$EPE = \frac{1}{J}\sum_{j=1}^{J}\left\|j^{pred} - j^{gt}\right\|^{2} \tag{5.8}$$

  Where J represents the number of estimated keypoints, and j^{pred} and j^{gt} denote the 3D coordinates of the estimated and actual keypoints, respectively.

- **Percentage of Correct Keypoints (PCK):** PCK is a widely used error metric for 3D hand pose estimation. It assesses the accuracy of localizing individual keypoints within a specified matching threshold. The threshold is typically determined as a fraction of the head segment length for each test image and is denoted as PCK@0.5. Higher PCK values indicate better model performance. Mathematically, PCK is computed as:

$$PCK_{\sigma} = \frac{1}{J} \sum_{j=1}^{J} \delta \left(\left\| w_j^{pre} - w_j^{gt} \right\| < \sigma \right) \tag{5.9}$$

where w_j^{pre} represents the estimated 3D coordinates of the j^{th} joints,w_j^{gt} corresponds to the actual 3D keypoint, and σ is an indicate function.

- **Area Under the Curve (AUC):** AUC is considered a comprehensive criterion for evaluating the performance and correctness of the model. It estimates the proportion of true keypoints(PCK) under various error thresholds, providing a holistic assessment of model performance. Mathematically, AUC for a specific joint is calculated as:

$$AUC_J = \int PCK_{\sigma}^{J} \tag{5.10}$$

5.4 Experimental and Results

We have conducted quantitative and qualitative evaluations of our technique to address the challenge of learning 2D and 3D hand pose regression, explicitly targeting the issue of occlusion. Additionally, we have demonstrated the generalization capabilities of our approach by producing accurate predictions even on single-hand datasets. Our results underscore the efficiency in predicting hand poses, whether for single-hand scenarios or interactions involving multiple fingers.

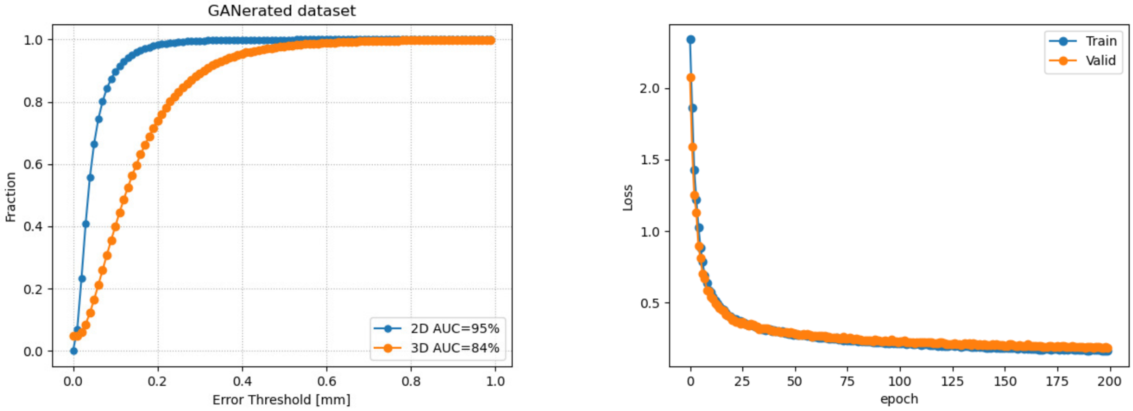
Furthermore, we have performed a comparative study with recent methods closely related to ours, using three benchmark datasets as the testing grounds. These comparisons highlight the strengths and advantages of our approach in the context of hand pose estimation, shedding light on its potential contributions to the field.

5.4.1 Quantitative Evaluation

To quantitatively validate our results and enhance the robustness of our strategy, we have employed various metrics for quality assessment. Our efforts have led to significant performance improvements, surpassing previous methods by a considerable margin across three datasets: the GANerated dataset (as depicted in Figure 5.7), the Stereo dataset (shown in Figures 5.13,5.14,5.15), and the SynthHands dataset (illustrated in Figure 5.11, 5.12).

In particular, we have utilized the Mean Square Error(MSE) as a pivotal metric to assess errors during the training and validation stages on the GANerated dataset. This metric has proven highly effective in producing superior outcomes when combined with the correct hyperparameters and data augmentation techniques, as detailed in Section 4.1.

Our comprehensive training approach involved 200 epochs, utilizing 100 batches with 64 batches per epoch. We employed a stochastic gradient descent optimizer with a momentum value of 0.9 and a learning rate of 0.005. Additionally, the weights in the ResNet network were kept frozen throughout the training process. These experimental settings yielded notable improvements in loss, as depicted in Figure 5.7(b).



(a) 2D and 3D PCK metrics under the perspective threshold of our architecture.

(b) Mean Square Error(MSE) applied during training and validation test data.

Figure 5.7: Quantitative Evaluation of our proposed approach on GANerated dataset.

During the data augmentation process, we provide the same hyperparameters as those employed during the training. Our investigations have revealed intriguing disparities in the impact of augmentation on 2D and 3D training errors when applied to the GANerated and SynthHands datasets, which we have noted in Table 2.

In the case of the GANerated dataset, we observed that before augmentation, both 2D and 3D errors were comparatively higher. However, after augmentation, these errors decreased by approximately 0.058 and 0.063, respectively. Furthermore, when assessed using the same hyperparameters, the 2D and 3D AUC percentages exhibited remarkable increases of approximately 94% and 84%, respectively. These results underscore the beneficial impact of incorporating augmentation techniques in the GANerated dataset, leading to improved performance compared to models trained without augmentation.

However, the SynthHands dataset consists of hand images with a green screen background, allowing for straightforward separation of the hand from the background. When applied to these images, the green background can affect the data augmentation procedure. In particular, while rotating or translating the image, the green backdrop may become apparent in regions where it was previously obscured, resulting in an unnatural augmentation of the hand region.

Despite this potential concern, our investigations on the SynthHands dataset revealed that data augmentation improved metric-based results. This suggests that the advantages of data augmentation, such as increased data variability and improved generalization to new data, outweigh the limitations posed by the green background.

		Metrics			
		2D mEPE	3D mEPE	2D AUC	3D AUC
Before Augmentation	GANerated dataset	0.063	0.1718	0.93	0.823
	SynthHands datasets	0.034	0.134	0.960	0.860
After Augmentation	GANerated dataset	0.058	0.165	0.94	0.842
	SynthHands datasets	0.041	0.1842	0.953	0.811

Table 5.2: The effect of data augmentation on the GANerated and SynthHands datasets using evaluation metrics: 2D and 3D mean End Point Error(mEPE) and 2D and 3D AUC.

Our work stands out for its close alignment with the research conducted by Mueller et al. [27], particularly in its utilization of the GANerated dataset. Figure 5.8 showcases the performance curves, with distinctive approaches highlighted: synthetic images (represented in blue) with color augmentation (depicted in orange), a combination of synthetic and GANerated data (indicated in green), and the integration of Projlayer (highlighted in red).

Our approach has achieved a remarkable AUC value of 0.945, proving a significant advancement compared to the work of Mueller et al. This superior performance underscores the efficacy and innovation embedded in our method compared to prior research.

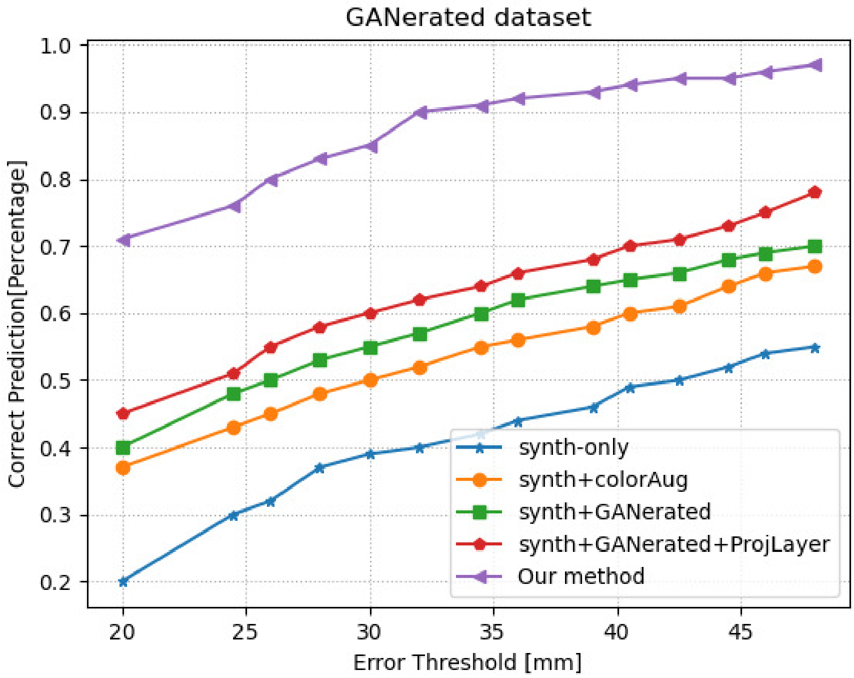


Figure 5.8: 3D PCK on GANerated datasets. Comparison with the work of Meuller et al. [26] in a different manner.

In Figure 5.9, we can observe the 2D Mean End-Point-Error(mEPE) for the complete set of hand joints. The total average mEPE of approximately 4.94 mm demonstrates the successful distribution of errors for each joint during the training process. This outcome signifies the robust learning capability of our model in accurately estimating hand poses.

A low mEPE indicates that our approach has effectively learned to predict the positions of hand joints with precision, showcasing its ability to handle the complexities of hand pose estimation. This distribution of errors across all joints further emphasizes the reliability and accuracy of predictions, which is crucial for various computer vision applications and tasks.

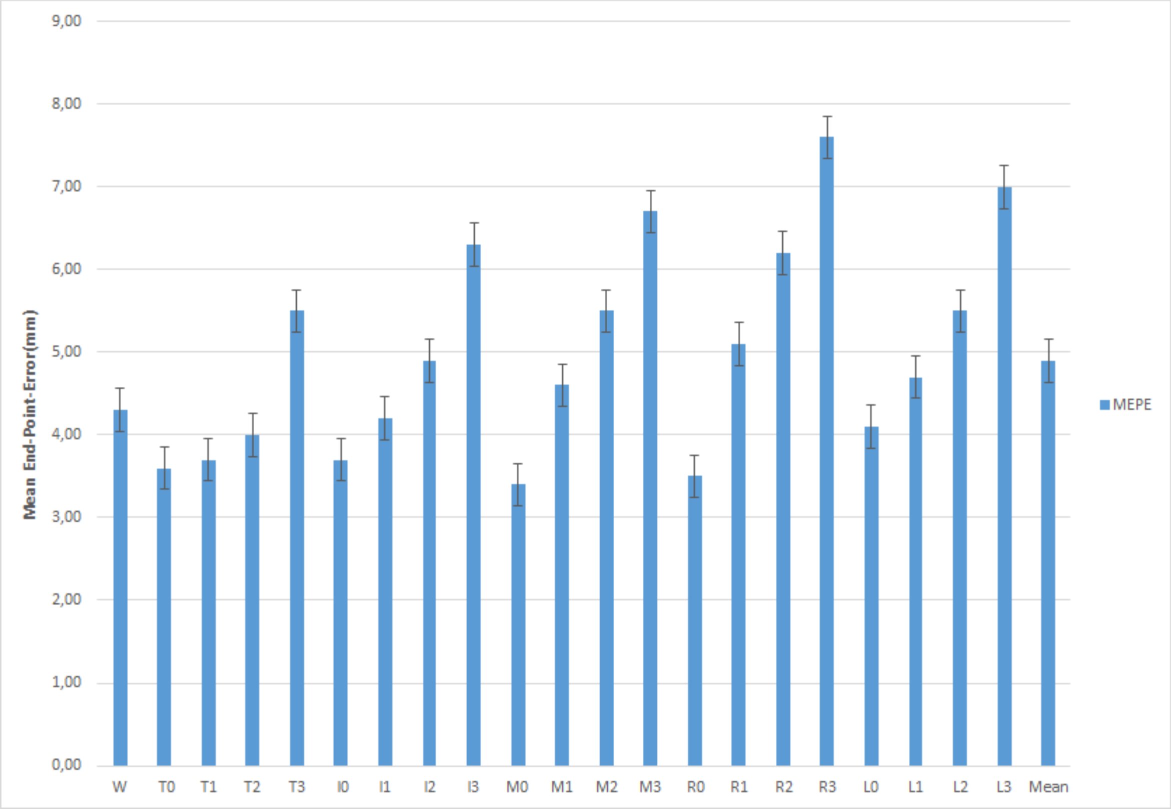


Figure 5.9: The Mean End-Point-Error of each joint of a full hand on GANerated dataset [27].

The comparison of the 2D and 3D Percentage of Correct Keypoints(PCK) curves, as depicted in Figure 5.10, clearly highlights the superiority of our predictions. These curves demonstrate our efficiency in estimating 2D and 3D hand poses, particularly when evaluated on the extensive SynthHands dataset.

Remarkably, it is essential to note that very few previous works have compared their performance specifically on the SynthHands dataset. Our comparison only finds a counterpart in the work of Li et al. [28]. This unique evaluation setting further underscores its exceptional nature, showcasing its dominance in hand pose estimation on the SynthHands dataset.

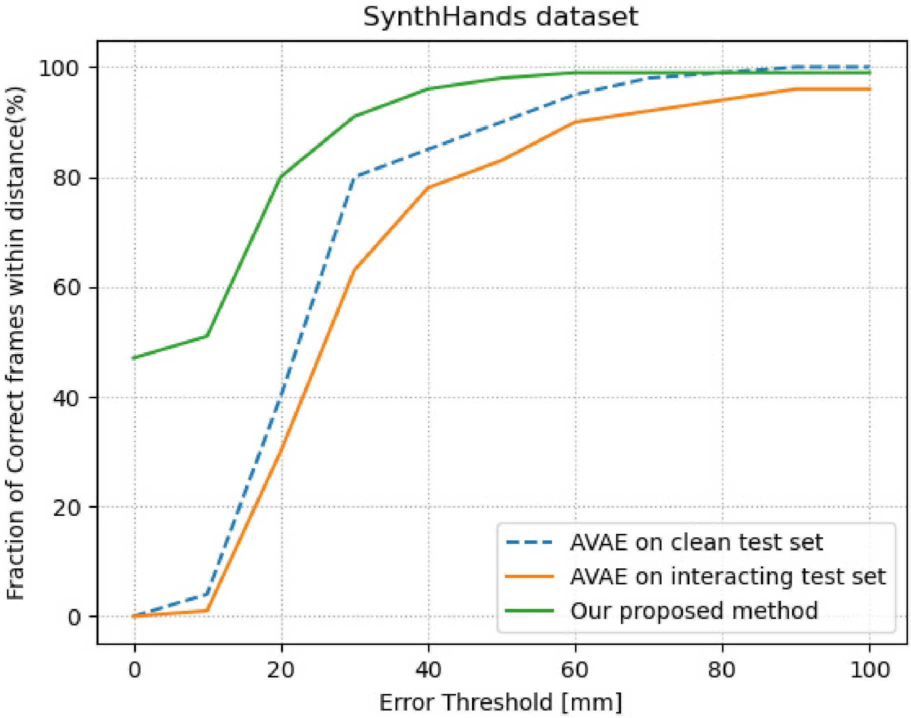


Figure 5.10: 3D PCK on SynthHands datasets compared with Li et al. [28] and their differents training options and architectures.

Table 3 provides a comprehensive comparison of four distinct training datasets, each comprising various proportions of hand-object interaction instances using the SynthHands training samples. We explain each splitting dataset using interacting hands as follows:

- A. In the first scenario, represented as 100% clean data, it is observed that the error for interacting hand images increases during the training and test phases, with an error of approximately 19.13 mm. This suggests that training solely on clean data may not be sufficient to handle the complexities of hand-object interactions effectively.

- B. The second scenario involves using 75% clean data and 25% interacting hand data. The interacting samples were used without augmentation. This approach led to a decrease in joint error, down to approximately 14.16 mm compared to scenario A. This improvement is attributed to combining clean data with a smaller proportion of accurate interacting data, which aids in capturing the nuances of hand-object interactions.

- C. In the third scenario, a balanced mix of 50% clean and 50% interacting data was employed. The results indicate significantly lower errors during training. This outcome could be attributed to the more balanced dataset, which likely requires less data augmentation and, in turn, results in improved training outcomes.

- D. Scenario D involves a dataset with 25% clean data and 75% interacting data. The results show that this approach yields less favorable outcomes, suggesting that the clean data augmentation employed by Li et al. [28] may have limitations in generalizing to unseen images during the training process.

- E. Finally, our experiment(E) focuses solely on utilizing interacting hands from the SynthHands dataset. This approach leads to a notably lower error of approximately 6.33 mm. This result highlights the robustness of our approach, which outperforms other scenarios in handling occlusion cases associated with hand-object interactions.

Mean End-Point-Error on Test Dataset(mm)	
Training Dataset	interacting hand
A	19.13
B	14.16
C	14.35
D	15.99
E	**6.33 Ours**

Table 5.3: Comparison with the work of Li et al. [28] under different option training on SynthHands.

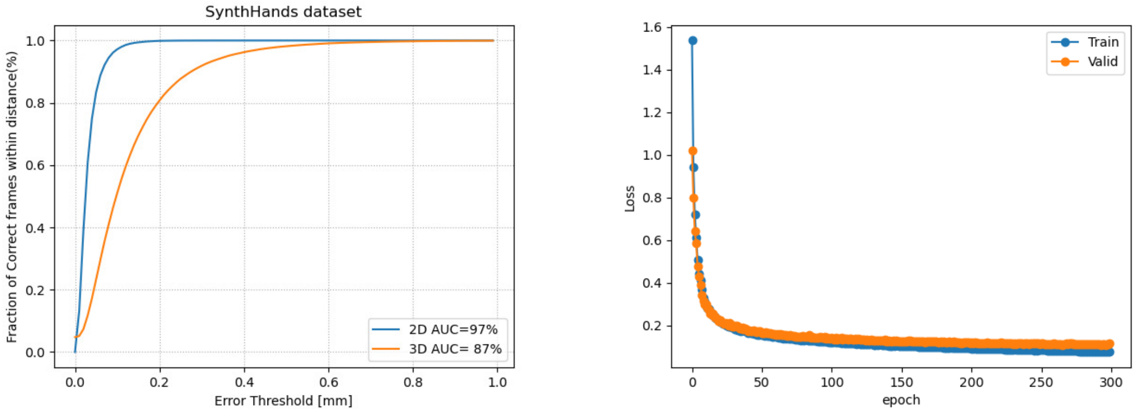
Our results demonstrate that the best performance is achieved when utilizing interacting hand data with augmentation, as outlined in Section 4.1. This approach leads to robust performance across various metrics and scenarios.

We introduced a novel metric, 3D Percentage of Correct Keypoints (PCK), calculated over different thresholds ranging from 0 to 100 mm, and compared it with the work of Li et al. [28]. The evaluation encompassed two training options: one focused on clean data and the other on interacting hand data. The results, as depicted in Figure 5.11, showcase the robustness of our model, with a 3D AUC of approximately 87% and a 2D AUC of around 97%. These metrics highlight the ability of a model to accurately estimate hand poses across various thresholds, underscoring its efficacy.

Furthermore, regarding loss, Figure 5.11(b) illustrates the Mean Square Error(MSE) throughout the training and validation phases. The plot indicates an optimized value as the number of training epochs increases, reflecting the learning process's effectiveness.

In addition to these metrics, we used the 3D Mean End-Point Error to compare our model with the work of Li et al. [28] across various testing scenarios. As shown in Figure 5.12, our model consistently outperforms the competition, with a mean Euclidean distance error of 6.33

mm. This result underscores the model's proficiency in estimating hand poses across various joints, further validating its effectiveness.



(a) The successful pose estimation frames in 2D and 3D under different error thresholds.

(b) Mean square Error(MSE) on training and validation dataset with appropriate hyperparameters.

Figure 5.11: Quantitative evaluation of our proposal on SynthHands datasets.

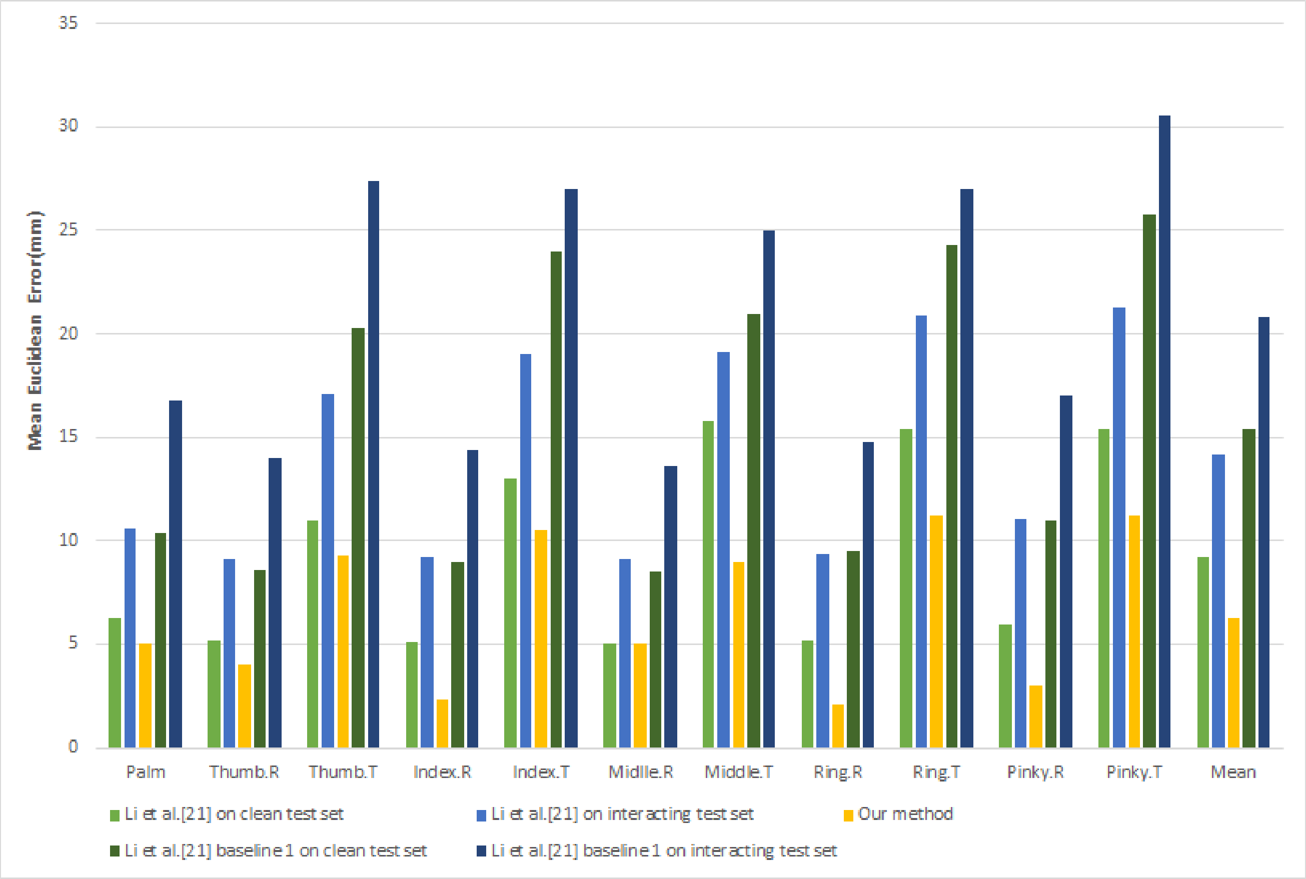


Figure 5.12: Mean End-point-Error of different joints of full hand on SynthHands Dataset.

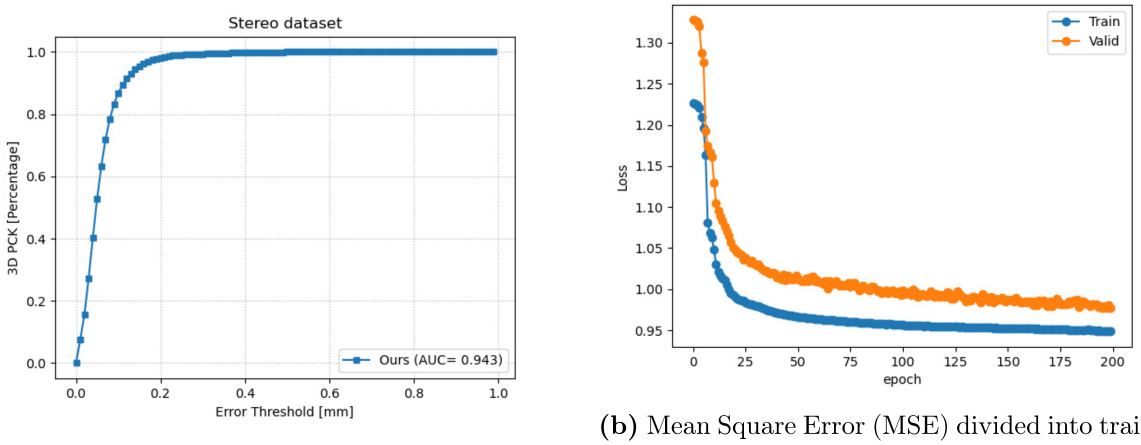
To validate the improvement achieved by our method, we conducted tests on a third benchmark dataset, Stereo DS, using the same metric evaluations employed in previous experiments. Here are the key findings:

In Figure 5.13(a), we present the 2D Percentage of Correct Keypoints (PCK) curve, which spans thresholds ranging from 0 to 100 mm. The curve illustrates excellent results, with an Area Under the Curve(AUC) of approximately 0.943. This high AUC value underscores the efficiency model in accurately estimating 2D hand poses on the Stereo DS.

Figure 5.13(b) outlines the training configuration used for this dataset. We employed appropriate hyperparameters, including 100 batches per epoch, a batch size 64, and training for 200 epochs without augmentation. Stochastic gradient descent optimization was performed with a learning rate of 0.005. The weights were defined in the range of [10,1], and the output model produced 2D and 3D vectors with latent heatmaps. Importantly, the ResNet34 model used to extract features remained unfrozen throughout training.

The results indicate lower error rates during the training process, with an error of approximately 0.948, and during validation, with an error of around 0.977. These outcomes highlight the effectiveness of our model in accurately estimating hand poses on the Stereo DS, further validating its performance across different benchmark datasets.

In conclusion, our model consistently achieves impressive results across various datasets, including Stereo DS, as demonstrated by high AUC values and low error rates, reaffirming its capability to handle different hand pose estimation challenges.



(a) 2D PCK benchmark over a different threshold.

(b) Mean Square Error (MSE) divided into training and validation tests.

Figure 5.13: Quantitative evaluation on the STB dataset.

However, we conducted a comparative analysis with recently proposed methods, utilizing the 3D Percentage of Correct Keypoints (PCK), as depicted in Figure 5.14. These methods, including [26, 27, 33, 35, 36], have garnered significant attention for their promising results on the STB dataset. Our analysis revealed that our model achieved a remarkable AUC within the range of 20-50 mm, approximately 0.999, surpassing the performance of all recent approaches.

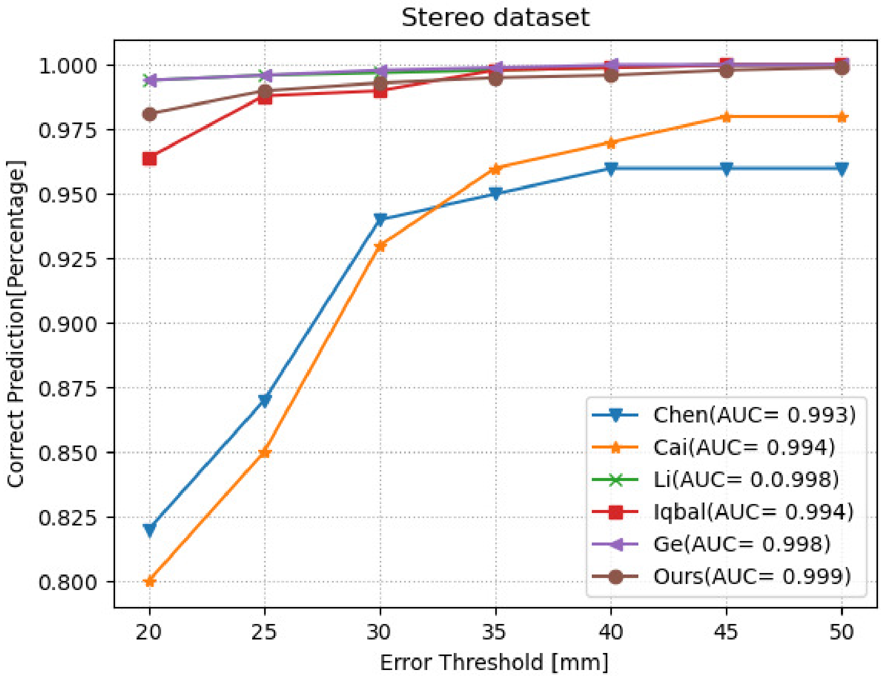


Figure 5.14: Comparison with the SOTA methods [28–32] on the STB dataset using 3D PCK. The X-axis is the threshold values(i.e., the maximum permitted distance between estimated and ground truth), and the Y-axis is the 3D PCK over the perspective threshold. The "AUC" shown in this curve is between 20 and 50[mm].

Additionally, the experimental results presented in Table 5.4 demonstrate superior performance, achieving an AUC metric of 98% within the threshold range of [0-50] mm when compared to the works of [33, 187]. Notably, our model reports the lowest error rate, approximately 2.96, in comparison to [28, 188], even though some methods, such as [28, 189], do not provide AUC metrics. These results underscore the generalization capability of our models, particularly on the STB dataset, which features greater challenges regarding occluded fingers and complex backgrounds.

To illustrate the effectiveness of our approach, we conducted a comparison with state-of-the-art works [28–32] on the STB dataset, as depicted in Figure 5.15. This comparison demonstrates that our model is capable of robustly estimating hand poses across a range of hand interactions and intricate pose articulations. Further, our results indicate a high level of performance, particularly in terms of the AUC metric within the range of 20-50 mm, achieving approximately 0.942. Furthermore, by the works of Iqbal et al. [30] and Li et al. [28], which are closely related to our approach, we conducted a comprehensive comparison, as presented in Table 5.5.

Li et al. [28] employed a latent distributed heatmap (LDH) to predict 3D hand poses from RGBD images and achieved an AUC of 0.94 on the STB dataset. Iqbal et al. [30] provided a 2.5D repre-

Method	2D mEPE(mm)(in pixel) ↓	AUC of 2D PCK_{0-50}
Li et al. [29]	13.58	-
Zimmermann et al. [33]	5.522	0.817
kourbane et al. [187]	6.21	0.796
Gao et al. [188]	5.059	0.831
Yuan et al. [189]	5.801	-
Ours	**2.96**	**0.942**

Table 5.4: 2D comparison with the existing methods on STB dataset utilizing Mean EPE[mm](pixels) and 2D AUC. "↓" indicates that lower is good, "↑": higher is better

sentation heatmap from RGBD images for hand pose estimation and obtained an AUC of about 0.94, evaluated in 2D. In contrast, our approach surpasses these results with an AUC of approximately 0.986, signifying superior performance than existing methods that reported lower results. Thus, we evaluated the Mean End-Point Error(mEPE) metric, which measures the average Euclidean distance error for keypoint predictions. Li et al. and Iqbal et al. achieved mEPE values of about 3.57 mm and 3.54 mm, respectively, while our model achieved a lower mEPE of 2.96 mm, highlighting its enhanced accuracy in predicting hand keypoints.

Furthermore, we introduced a 3D mEPE metric to assess the performance of our approach and conducted a comparison with the works of Li et al. [28] and Iqbal et al. [30]. Our model achieved a 3D mEPE of approximately 6.86 mm, outperforming the competition, with Li et al. reporting a 3D mEPE of 15.77 mm and Iqbal et al. achieving 9.43 mm. These results emphasize superior accuracy and precision in our estimating 3D hand poses approach.

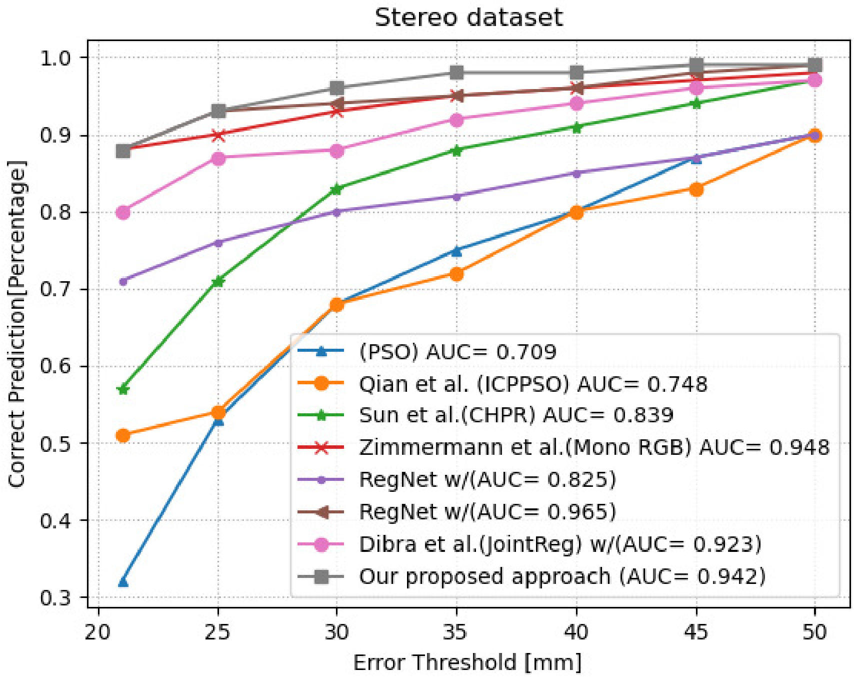


Figure 5.15: Comparison with the state-of-the-art work [31,33–36]on the STB dataset utilizing 3D PCK. The X-axis is the threshold values, and Y-axis is the 3D PCK over the perspective threshold. The Area Under the Curve in this curve is AUC(1-100)[mm].

Method	Iqbal et al. [30]	Li et al. [29]	Ours
2D Mean EPE ↓	3.57	3.40	**2.96**
3D Mean EPE ↓	15.77	9.43	**6.86**
AUC(0-30) ↑	0.89	0.94	**0.942**

Table 5.5: Comparison with the current work in the same context as ours [29, 30] on STB dataset utilizing 2D and 3D Mean EPE[mm] and 2D AUC. "↓" indicates that lower is good, "↑": higher is better Mean EPE[mm](pixels) and 2D AUC. "↓" indicates that lower is good, "↑": higher is better

5.4.2 Qualitative Evaluation

In addition to the quantitative assessments, we conducted a qualitative evaluation on three available datasets: GANerated, SynthHands, and STB dataset. This qualitative analysis serves to visually demonstrate the effectiveness of our approach in accurately predicting hand poses from diverse viewpoints, encompassing 2D and 3D keypoints, as well as 2D skeletons.

To further emphasize the viability and robustness of our approach, we extended our evaluation to include several unseen images, particularly in challenging scenarios marked by severe occlusion. The results of this evaluation are showcased in Figures 5.16, 5.17, and 5.18, highlighting the capability to deliver accurate predictions even in situations of significant occlusion.

This qualitative assessment underscores the practical utility and reliability of our proposal across various real-world scenarios, reaffirming its effectiveness in hand pose estimation.

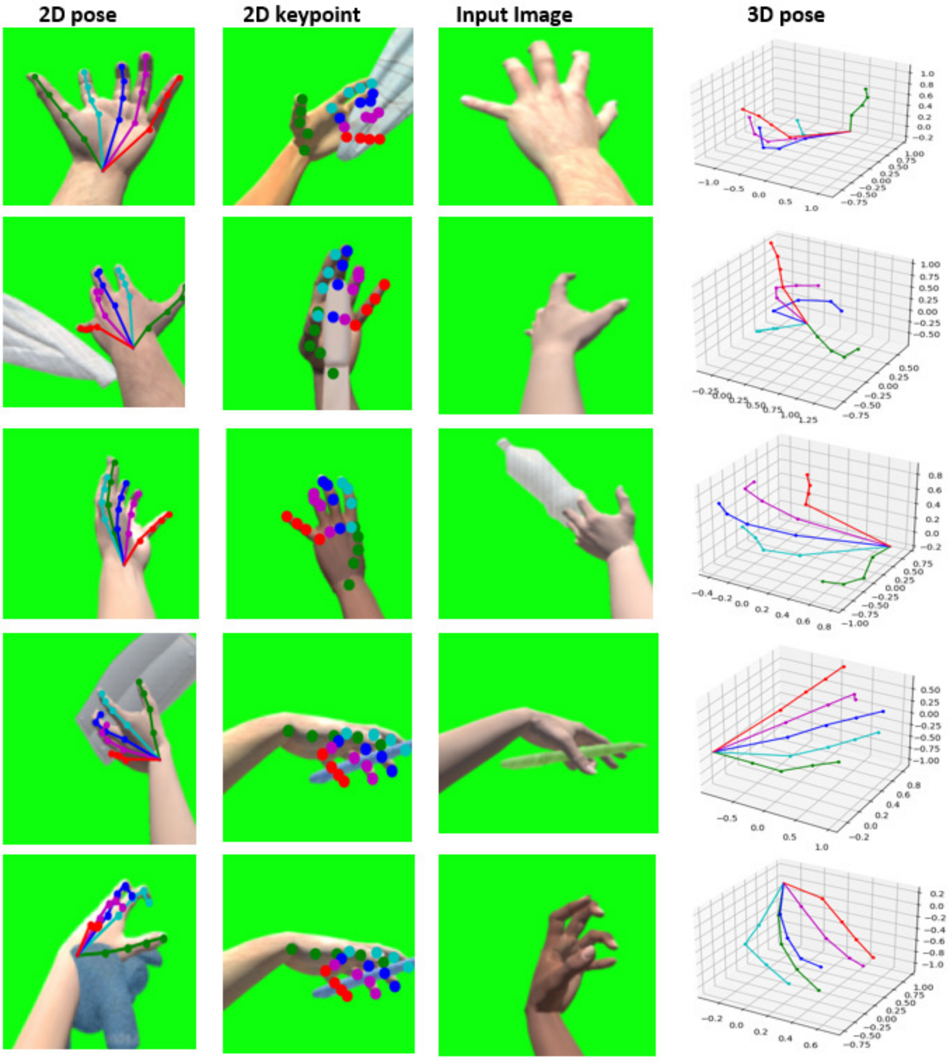


Figure 5.16: Qualitative results on SynthHANDS dataset [26]. The first and the second row are images estimated from the proposed approach with 2D keypoints and a 2D skeleton. The third one represents some images from a dataset, and the following row shows the estimated 3D joints of the hand.

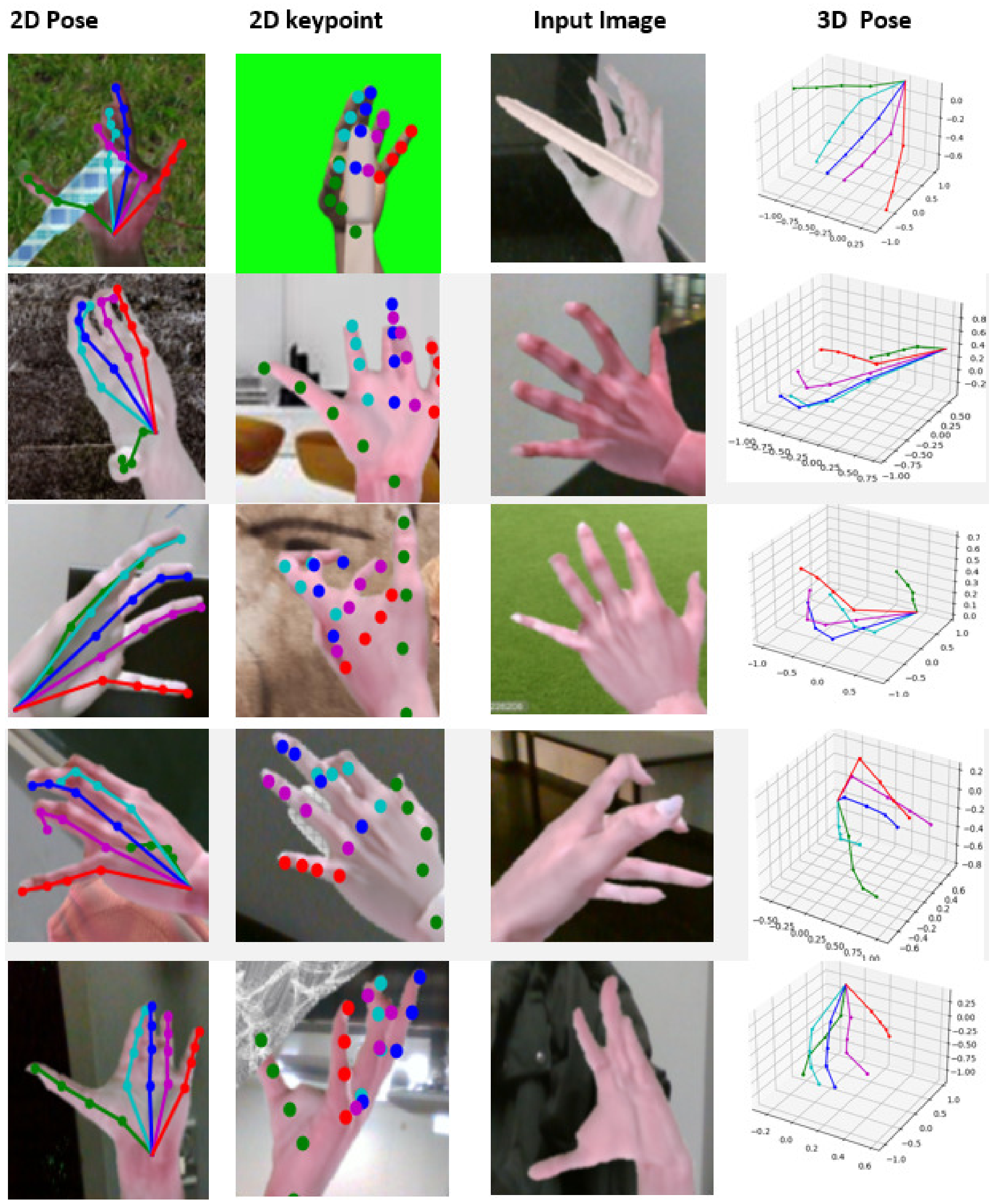


Figure 5.17: Qualitative results on GANerated dataset [27]. The initial and the second columns are images estimated from the proposed approach with 2D keypoints and 2D skeleton. The third one represents some hand images, and the following column shows the estimated 3D joints of the hand.

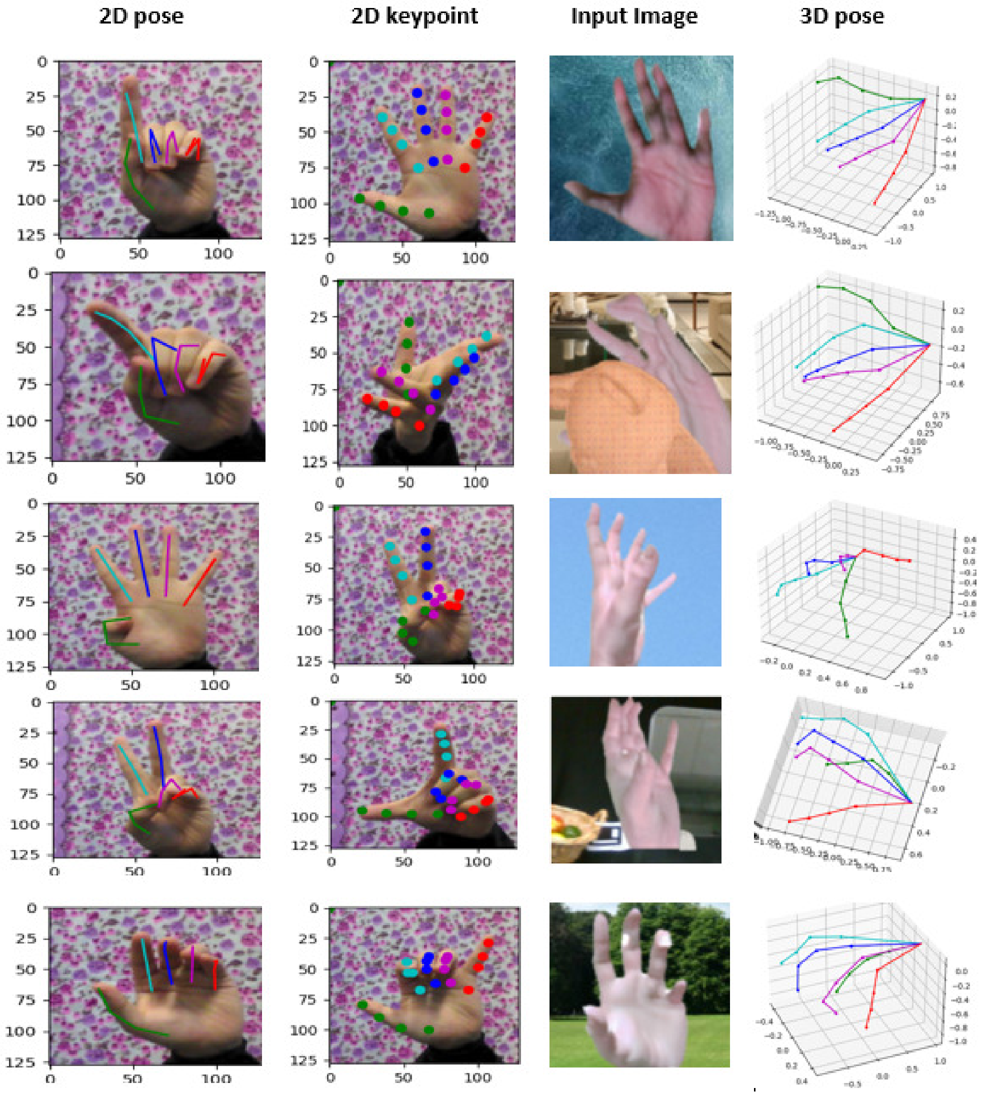


Figure 5.18: Qualitative results on STB dataset [37]. The initial and the second columns are some images estimated from our proposed approach with 2D keypoints and a 2D skeleton. The third one represents the hand image, and the following column shows the estimated 3D joints of the hand.

5.5 Conclusion

In this chapter, we have addressed the challenge of accurately estimating both 2D and 3D hand poses from a single RGB image while dealing with the complexities of hand occlusion during interactions.

To achieve precise and robust hand pose estimation under these conditions, we developed a deep learning model called "ResUnet," which combines the ResNet and Unet architectures as its foundation. We introduced a 2D regression pose using a Latent Heatmap Representation(LHR) generated from RGB inputs to estimate 2D hand poses. To regress the 3D hand pose, adding layers to concatenate with intermediate features of the ResUnet architecture. To enhance the stability and accuracy of our predictions, we employed a tree structure of the Hand to estimate bones, which proved to be a more robust approach than estimating joints alone. Quantitative and qualitative results demonstrate that our proposed framework significantly outperforms existing methods in assessing hand poses from different viewpoints and challenging occlusion scenarios. Further, our model can accurately predict joint angles, highlighting its effectiveness.

Chapter 6

General conclusion

Augmented reality(AR) has witnessed a remarkable evolution and growing popularity in recent years, as indicated by the latest statistics. AR has transcended its initial novelty phase and has firmly established itself as one of the most sought-after forms of immersive technology. As AR technology continues to advance, it promises to redefine the way we interact with the digital and physical worlds, making it more intuitive and seamless. With companies committing to this technological frontier, we can anticipate a future where AR not only enhances our daily lives but also drives innovation and unlocks new opportunities for businesses and consumers alike. The latest findings from industry experts, such as Market Research Future, reveal that the augmented reality sector is thriving and expanding more rapidly than almost any other innovative technology. Currently, the AR industry is on track to achieve an impressive compound annual growth rate(CAGR) of approximately 41.5%, with a projected value of $461.25 billion by the year 2030 [190]. In Chapter 1, we introduce the Augmented Reality foundations, discussing their definitions and advocating both conceptual and technological approaches. It covers AR's purpose, widely used applications, and key devices, concluding with popular software and hardware used in its implementation, serving as a comprehensive review of essential AR concepts.

In this dissertation, we investigate the occlusion issues, a long-standing challenge in AR that has been an active research topic in the literature for almost 30 years. We have presented two contributions to resolving occlusion in real-time by employing two solutions: handcraft methods and deep learning.

Handling occlusion in Augmented Reality based on photogrammetry: One of the earliest and most challenging in Augmented Reality is solving occlusion because it is crucial to establishing precise relationships between the virtual and real world. This facet holds great importance in creating accurate and efficient augmented reality systems. Occlusion refers to the ability to realistically depict how virtual objects interact with and are obscured by real-world objects within the user's field of view. To deal with these issues, in Chapter 3, we present model-based methods based on close-range photogrammetry algorithms using software to construct a 3D model of the real environment. In addition, we perform marker-less tracking that aims to determine the location and movement of objects in the scene based on their visual features using Vuforia tools. Further, we create a dataset including captured images from different viewpoints incorporating virtual objects to explore the occlusion. To realize the precise occlusion after the alignment between virtual and physical scenes, we employ an occlusion mask that compares their respective depth in the scene. This depth mask is provided to achieve a more realistic and visually appealing scene by controlling which parts of objects are visible and which are not, depending on their position in 3D space relative to other objects.

Hand pose estimation based on regression method from monocular RGB cameras for handling occlusion:

The second contribution of this dissertation is how to resolve the occlusion that occurs during the interaction with external objects. In Chapter 2, in the theoretical parts, we specify the use case of our work, and we focus on the hand pose estimation field, which is considered almost recent topic due to the challenges faced like hand parsing, data labeling techniques, hand motion, fingertip detection, hand localization, and self-occlusion. Additionally, we perform a novel taxonomy rooted in deep learning approaches encompassing the recent research depending on the input modality into depth-based, image-based, and RGBD-based categories. These diverse approaches reflect the dynamic landscape of research aimed at enhancing our ability to accurately and robustly estimate hand poses in various contexts. Thus, we discuss the highlighting method by its advantages and limitations. We provide benchmark datasets for estimating hand pose, their characteristics, and the commonly employed evaluation metrics for assessing these methodologies. Finally, we deliberate on potential avenues for future research, including considerations of speed, precision, and the selection of CNN architectures within this swiftly evolving field.

Recently, deep neural networks have demonstrated remarkable effectiveness across various research domains. The surge in interest in deep learning, which began around 2010, can be largely attributed to the increasing digitization of society, leading to exponential growth in the volume of available data. Deep learning algorithms thrive on substantial datasets, and as people continue to upload vast amounts of data, such as texts, images, and videos each year, the Computer Vision community can leverage this wealth of information to enhance the efficiency of their systems. Nevertheless, certain types of naturally digitized data, such as sequences of hand gestures or depth images, remain challenging to capture. To harness the capabilities of deep learning algorithms with a smaller dataset, a prudent approach involves employing transfer learning strategies. In a second time, we aimed to go further in handling occlusion in hand pose estimation. First, we proposed to perform online learning, which allows the system to detect the human hand from cropped images and interact with the different objects in real-time, an essential capacity for real applications. Second, we have taken over the whole pipeline process to estimate the 2D and 3D pose under the occlusion issue and used the power of deep learning models to increase our system's efficiency and robustness.

Our framework is mainly composed of three steps. First, we used a deep learning model that can take RGB images as input called a ResNet34 layer network to extract both hand posture and shape descriptors. The network is trained using a Three benchmark hand pose estimation dataset, and we transferred its knowledge to extract relevant features. Thus, as the second pathway, we used four multi-feature blocks called "Unet-Blocks" to generate non-normalized 2D heatmaps that represent the presence of the location of each joint. Thus, we apply a Latent Heatmap representation (LHR) method to regress the 2D pose coordinates. As the third step, we add convolutional blocks to process heatmaps and concatenate the first part's intermediate features to obtain 3D features. Finally, to perform a 3D hand pose, we defined the bones of the hand instead of joints by presenting the hand as a tree structure. While implementing our proposal, we employ data augmentation techniques to provide more reliable and highest results compared with the state-of-the-art method. We learn the human hand to grasp objects while tackling occlusion problems accurately.
The experimental outcomes validated the effectiveness of our method in not only estimating hand pose but also surpassing current best practices. Furthermore, the tests indicated that our system can identify a gesture well before completion, rendering it highly suitable for real-time applications. Employing the transfer learning strategy enabled us to outperform existing deep learning models with fewer parameters.

As a future perspective, our experiments demonstrated that the proposed approach ensures successful dynamic hand pose estimation, although they do not surpass human performance. Initially, it is essential to note that hand pose estimation remains an ongoing area of research, and the model created for extracting features in our framework has room for improvement. In 2017, Yuan et al. [191] introduced the extensive Big-Hand2.2M hand pose dataset, which is expected to facilitate researchers in enhancing the performance of their hand pose estimation algorithms. For that, we plan to see more accurate, diverse, and much bigger datasets in this field with well-annotated data, which helps to relieve the assumptions about hand pose. However, with the impressive progress of hand pose estimation methods related to the depth advanced cameras and using VR/AR devices as one of the crucial techniques in human-computer interaction technology. Hand pose estimation will be quietly developed and gain more interest with available applications for human beings.

Furthermore, we intend to improve our proposal by implementing kinematic fitting techniques to ensure physically plausible hand poses, capturing absolute hand motion, and enhancing our model's robustness for real-time fingertip refinement. Thus, we aim to explore advanced applications such as Augmented Reality, Virtual Reality, and human-computer interaction to enhance the practicality of our proposal. This will pave the way for natural interactions between human hands and the virtual world, ultimately delivering an exceptional user experience.

Finally, tracking both hands for hand pose estimation introduces many complex challenges. One of the primary issues is handling occlusion, as the hands are in constant motion and can frequently obstruct each other, resulting in data loss and potential estimation errors. The obtained findings in this dissertation tackled the occlusion from a single image. This task also significantly burdens computational resources due to the increased load, particularly in real-time applications where low latency is critical. Moreover, collecting and annotating datasets for two-hand tracking is more labor-intensive, necessitating precise labeling of hand poses for both hands. The dynamic interactions between the two hands add another layer of complexity, and accurately tracking two hands in 3D environments involves addressing depth information and occlusions in 3D space. Edge cases, user variability, and real-time performance are additional factors that require careful consideration. These challenges collectively underline the need for proposing robust algorithms and sensors to realize the potential of hand pose estimation in various interactive scenarios.

Bibliography

[1] Louis B Rosenberg. Virtual fixtures: Perceptual tools for telerobotic manipulation. In *Proceedings of IEEE virtual reality annual international symposium*, pages 76–82. Ieee, 1993.

[2] Paul Milgram and Fumio Kishino. A taxonomy of mixed reality visual displays. *IEICE TRANSACTIONS on Information and Systems*, 77(12):1321–1329, 1994.

[3] Hannes Kaufmann and Dieter Schmalstieg. Mathematics and geometry education with collaborative augmented reality. In *ACM SIGGRAPH 2002 conference abstracts and applications*, pages 37–41, 2002.

[4] Christopher Lindinger, Roland Haring, Horst Hörtner, Daniela Kuka, and Hirokazu Kato. Multi-user mixed reality system 'gulliver's world': a case study on collaborative edutainment at the intersection of material and virtual worlds. *Virtual Reality*, 10:109–118, 2006.

[5] Ye Paschenko. Ar–stvorennia novoi avtoservisnoi realnosti (ar–the creation of a new car service reality). suchasna avtomaisternia 1-2(129), 2020.

[6] Rick Cavallaro. The foxtrax hockey puck tracking system. *IEEE Computer Graphics and Applications*, 17(2):6–12, 1997.

[7] Jung Eun Lee, Nan Zeng, Yoonsin Oh, Daehyoung Lee, and Zan Gao. Effects of pokémon go on physical activity and psychological and social outcomes: a systematic review. *Journal of clinical medicine*, 10(9):1860, 2021.

[8] Tobias Hollerer, Steven Feiner, and John Pavlik. Situated documentaries: Embedding multimedia presentations in the real world. In *Digest of Papers. Third International Symposium on Wearable Computers*, pages 79–86. IEEE, 1999.

[9] Bill and Fox. Howard behar: Former president, starbucks. *The Future of the Workplace: Insights and Advice from 31 Pioneering Business and Thought Leaders*, pages 1–5, 2020.

[10] Nassir Navab, A Bani-Kashemi, and Matthias Mitschke. Merging visible and invisible: Two camera-augmented mobile c-arm (camc) applications. In *Proceedings 2nd IEEE and*

ACM International Workshop on Augmented Reality (IWAR'99), pages 134–141. IEEE, 1999.

[11] Natalie Bursztyn, Brett Shelton, Andy Walker, and Joel Pederson. Increasing undergraduate interest to learn geoscience with gps-based augmented reality field trips on students' own smartphones. *GSA Today*, 27(5):4–11, 2017.

[12] Fidel Díez-Díaz, Martín González-Rodríguez, and Agueda Vidau. An accesible and collaborative tourist guide based on augmented reality and mobile devices. In *Universal Access in Human-Computer Interaction. Ambient Interaction: 4th International Conference on Universal Access in Human-Computer Interaction, UAHCI 2007 Held as Part of HCI International 2007 Beijing, China, July 22-27, 2007 Proceedings, Part II 4*, pages 353–362. Springer, 2007.

[13] Michael Bajura, Henry Fuchs, and Ryutarou Ohbuchi. Merging virtual objects with the real world: Seeing ultrasound imagery within the patient. *ACM SIGGRAPH Computer Graphics*, 26(2):203–210, 1992.

[14] Soo Kyun Kim, Shin-Jin Kang, Yoo-Joo Choi, Min-Hyung Choi, and Min Hong. Augmented-reality survey: from concept to application. *KSII Transactions on Internet & Information Systems*, 11(2), 2017.

[15] Mark Weiser. The computer for the 21 st century. *Scientific american*, 265(3):94–105, 1991.

[16] Daniel Wagner and Dieter Schmalstieg. Handheld augmented reality displays. In *IEEE Virtual Reality Conference (VR 2006)*, pages 321–321. IEEE, 2006.

[17] Jonathan Maw, Kai Yuen Wong, and Patrick Gillespie. Hand anatomy. *British Journal of Hospital Medicine*, 77(3):C34–C40, 2016.

[18] Bryan Buchholz and Thomas J Armstrong. A kinematic model of the human hand to evaluate its prehensile capabilities. *Journal of biomechanics*, 25(2):149–162, 1992.

[19] Nicholas Katzakis, Robert J Teather, Kiyoshi Kiyokawa, and Haruo Takemura. Inspect: extending plane-casting for 6-dof control. *Human-centric Computing and Information Sciences*, 5(1):1–22, 2015.

[20] Ihsan Rabbi and Sehat Ullah. A survey on augmented reality challenges and tracking. *Acta graphica: znanstveni časopis za tiskarstvo i grafičke komunikacije*, 24(1-2):29–46, 2013.

[21] Siddharth S Rautaray and Anupam Agrawal. Vision based hand gesture recognition for human computer interaction: a survey. *Artificial intelligence review*, 43:1–54, 2015.

[22] Abdelkader Bellarbi, Samir Benbelkacem, Nadia Zenati-Henda, and Mahmoud Belhocine. Hand gesture interaction using color-based method for tabletop interfaces. In *2011 IEEE 7th International Symposium on Intelligent Signal Processing*, pages 1–6. IEEE, 2011.

[23] A Messaci, N Zenati, A Bellarbi, and M Belhocine. 3d interaction techniques using gestures recognition in virtual environment. In *2015 4th International Conference on Electrical Engineering (ICEE)*, pages 1–5. IEEE, 2015.

[24] Martin Ebner. Game-based learning with the leap motion controller. In *Handbook of research on gaming trends in P-12 education*, pages 555–565. IGI Global, 2016.

[25] Shanxin Yuan, Qi Ye, Bjorn Stenger, Siddhant Jain, and Tae-Kyun Kim. Bighand2. 2m benchmark: Hand pose dataset and state of the art analysis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4866–4874, 2017.

[26] Franziska Mueller, Dushyant Mehta, Oleksandr Sotnychenko, Srinath Sridhar, Dan Casas, and Christian Theobalt. Real-time hand tracking under occlusion from an egocentric rgb-d sensor. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1154–1163, 2017.

[27] Franziska Mueller, Florian Bernard, Oleksandr Sotnychenko, Dushyant Mehta, Srinath Sridhar, Dan Casas, and Christian Theobalt. Ganerated hands for real-time 3d hand tracking from monocular rgb. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 49–59, 2018.

[28] Shile Li, Haojie Wang, and Dongheui Lee. Hand pose estimation for hand-object interaction cases using augmented autoencoder. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 993–999. IEEE, 2020.

[29] Moran Li, Jialong Wang, and Nong Sang. Latent distribution-based 3d hand pose estimation from monocular rgb images. *IEEE Transactions on Circuits and Systems for Video Technology*, 31(12):4883–4894, 2021.

[30] Umar Iqbal, Pavlo Molchanov, Thomas Breuel Juergen Gall, and Jan Kautz. Hand pose estimation via latent 2.5 d heatmap regression. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 118–134, 2018.

[31] Liangjian Chen, Shih-Yao Lin, Yusheng Xie, Hui Tang, Yufan Xue, Xiaohui Xie, Yen-Yu Lin, and Wei Fan. Generating realistic training images based on tonality-alignment generative adversarial networks for hand pose estimation. *arXiv preprint arXiv:1811.09916*, 2018.

[32] Yujun Cai, Liuhao Ge, Jianfei Cai, and Junsong Yuan. Weakly-supervised 3d hand pose estimation from monocular rgb images. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 666–682, 2018.

[33] Christian Zimmermann and Thomas Brox. Learning to estimate 3d hand pose from single rgb images. In *Proceedings of the IEEE international conference on computer vision*, pages 4903–4911, 2017.

[34] Franziska Mueller, Florian Bernard, Oleksandr Sotnychenko, Dushyant Mehta, Srinath Sridhar, Dan Casas, and Christian Theobalt. Ganerated hands for real-time 3d hand tracking from monocular rgb. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 49–59, 2018.

[35] Chen Qian, Xiao Sun, Yichen Wei, Xiaoou Tang, and Jian Sun. Realtime and robust hand tracking from depth. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1106–1113, 2014.

[36] Endri Dibra, Silvan Melchior, Ali Balkis, Thomas Wolf, Cengiz Oztireli, and Markus Gross. Monocular rgb hand pose inference from unsupervised refinable nets. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 1075–1085, 2018.

[37] Yu Zhang, Chi Xu, and Li Cheng. Learning to search on manifolds for 3d pose estimation of articulated objects. *arXiv preprint arXiv:1612.00596*, 2016.

[38] Mark Billinghurst. Grand challenges for augmented reality. *Frontiers in Virtual Reality*, 2:12, 2021.

[39] Iulian Radu. Why should my students use ar? a comparative review of the educational impacts of augmented-reality. In *2012 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 313–314. IEEE, 2012.

[40] Alexander Ptukhin, Konstantin Serkov, Artem Khrushkov, and Ekaterina Bozhko. Prospects and modern technologies in the development of vr/ar. In *2018 Ural Symposium on Biomedical Engineering, Radioelectronics and Information Technology (USBEREIT)*, pages 169–173. IEEE, 2018.

[41] Pietro Cipresso, Irene Alice Chicchi Giglioli, Mariano Alcañiz Raya, and Giuseppe Riva. The past, present, and future of virtual and augmented reality research: a network and cluster analysis of the literature. *Frontiers in psychology*, page 2086, 2018.

[42] Danhang Tang, Hyung Jin Chang, Alykhan Tejani, and Tae-Kyun Kim. Latent regression forest: Structured estimation of 3d articulated hand posture. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3786–3793, 2014.

[43] Jonathan Tompson, Murphy Stein, Yann Lecun, and Ken Perlin. Real-time continuous pose recovery of human hands using convolutional networks. *ACM Transactions on Graphics (ToG)*, 33(5):1–10, 2014.

[44] Emad Barsoum. Articulated hand pose estimation review. *arXiv preprint arXiv:1604.06195*, 2016.

[45] Iason Oikonomidis, Nikolaos Kyriazis, Antonis A Argyros, et al. Efficient model-based 3d tracking of hand articulations using kinect. In *BmVC*, volume 1, page 3, 2011.

[46] Anna Bosch, Andrew Zisserman, and Xavier Munoz. Scene classification using a hybrid generative/discriminative approach. *IEEE transactions on pattern analysis and machine intelligence*, 30(4):712–727, 2008.

[47] Seungryul Baek, Kwang In Kim, and Tae-Kyun Kim. Pushing the envelope for rgb-based dense 3d hand pose estimation via neural rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1067–1076, 2019.

[48] Ronald T Azuma. A survey of augmented reality. *Presence: teleoperators & virtual environments*, 6(4):355–385, 1997.

[49] Jean-Yves Didier, David Roussel, Malik Mallem, Samir Otmane, Sylvie Naudet, Quoc-Cuong Pham, Steve Bourgeois, Christine Mégard, Christophe Leroux, and Arnaud Hocquard. Amra: augmented reality assistance for train maintenance tasks. In *Workshop Industrial Augmented Reality, 4th ACM/IEEE International Symposium on Mixed and Augmented Reality (ISMAR 2005)*, pages Elect–Proc, 2005.

[50] Olivier Hugues. *Réalité Augmentée pour l'Aide à la Navigation*. PhD thesis, Bordeaux 1, 2011.

[51] Philippe Fuchs. *Le traité de la réalité virtuelle*, volume 2. Presses des MINES, 2006.

[52] Emmanuel Dubois. Conception, implémentation et evaluation de systèmes interactifs mixtes: une approche basée modèles et centrée sur l'interaction. *Habilitation à Diriger les Recherches, Université de Toulouse, Mémoire*, 2009.

[53] Samir Otmane. *Modèles et techniques logicielles pour l'assistance à l'interaction et à la collaboration en réalité mixte*. PhD thesis, Université d'Evry-Val d'Essonne, 2010.

[54] Michael A Gigante. Virtual reality: definitions, history and applications. In *Virtual reality systems*, pages 3–14. Elsevier, 1993.

[55] Ivan E Sutherland et al. The ultimate display. In *Proceedings of the IFIP Congress*, volume 2, pages 506–508. New York, 1965.

[56] Michael Heim. *The metaphysics of virtual reality*. Oxford University Press, USA, 1993.

[57] Jacques Cohen. Special issue on computer augmented environments: back to the real world. *ACM Communications*, 36(1), 1993.

[58] Ralf Dörner, Wolfgang Broll, Paul Grimm, and Bernhard Jung. Virtual und augmented reality. *Grundlagen und Methoden der Virtuellen und Augmentierten Realität. Berlin und Heidelberg: Springer*, 2013.

[59] Michael Zyda. From visual simulation to virtual reality to games. *Computer*, 38(9):25–32, 2005.

[60] Marcus Täuber. Exposition auf knopfdruck: Virtual reality exposure therapy (vret) bei phobien und süchten. *Psychotherapie-Wissenschaft*, 13(1), 2023.

[61] Hao Song, Fangyuan Chen, Qingjin Peng, Jian Zhang, and Peihua Gu. Improvement of user experience using virtual reality in open-architecture product design. *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, 232(13):2264–2275, 2018.

[62] Aixiang Yuan and Jinhwan Hong. Impacts of virtual reality on tourism experience and behavioral intentions: Moderating role of novelty seeking. *Journal of Hospitality & Tourism Research*, page 10963480231171301, 2023.

[63] Mel Slater. Place illusion and plausibility can lead to realistic behaviour in immersive virtual environments. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 364(1535):3549–3557, 2009.

[64] Daniel Freeman, Sarah Reeve, Abi Robinson, Anke Ehlers, David Clark, Bernhard Spanlang, and Mel Slater. Virtual reality in the assessment, understanding, and treatment of mental health disorders. *Psychological medicine*, 47(14):2393–2400, 2017.

[65] Ruo Wei Chen and Kan Kan Chan. Using augmented reality flashcards to learn vocabulary in early childhood education. *Journal of Educational Computing Research*, 57(7):1812–1831, 2019.

[66] Chao Yuan and Liang Chen. Mitigating urban heat island effects in high-density cities based on sky view factor and urban morphological understanding: a study of hong kong. *Architectural Science Review*, 54(4):305–315, 2011.

[67] Jennifer Herron. Augmented reality in medical education and training. *Journal of Electronic Resources in Medical Libraries*, 13(2):51–55, 2016.

[68] Camelia Macariu, Adrian Iftene, and Daniela Gîfu. Learn chemistry with augmented reality. *Procedia Computer Science*, 176:2133–2142, 2020.

[69] Ronald Azuma, Yohan Baillot, Reinhold Behringer, Steven Feiner, Simon Julier, and Blair MacIntyre. Recent advances in augmented reality. *IEEE computer graphics and applications*, 21(6):34–47, 2001.

[70] Laurent Vaissie and Jannick P Rolland. Accuracy of rendered depth in head-mounted displays: role of eyepoint location. In *Helmet-and Head-Mounted Displays V*, volume 4021, pages 343–353. SPIE, 2000.

[71] Marcus Tonnis, Christian Sandor, Gudrun Klinker, Christian Lange, and Heiner Bubb. Experimental evaluation of an augmented reality visualization for directing a car driver's attention. In *Fourth IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR'05)*, pages 56–59. IEEE, 2005.

[72] Carsten Matysczok, Rafael Radkowski, and Jan Berssenbruegge. Ar-bowling: immersive and realistic game play in real environments using augmented reality. In *Proceedings of the 2004 ACM SIGCHI International Conference on Advances in computer entertainment technology*, pages 269–276, 2004.

[73] DWF Van Krevelen and Ronald Poelman. A survey of augmented reality technologies, applications and limitations. *International journal of virtual reality*, 9(2):1–20, 2010.

[74] Mark Harris. Machine for shopping-[engineering future]. *Engineering & Technology*, 4(11):18–20, 2009.

[75] Bruce Clark. Teaching case: Lululemon mirror. *Available at SSRN*, 2023.

[76] Thad Starner. *Wearable computing and contextual awareness.* PhD thesis, Massachusetts Institute of Technology, 1999.

[77] Marcus Tonnis and Gudrun Klinker. Effective control of a car driver's attention for visual and acoustic guidance towards the direction of imminent dangers. In *2006 IEEE/ACM International Symposium on Mixed and Augmented Reality*, pages 13–22. IEEE, 2006.

[78] Neda Shabani and Azizul Hassan. Augmented reality for tourism service promotion in iran as an emerging market. In *Virtual and Augmented Reality: Concepts, methodologies, tools, and applications*, pages 1808–1818. IGI Global, 2018.

[79] Vassilios Vlahakis, M Ioannidis, John Karigiannis, Manolis Tsotros, Michael Gounaris, Didier Stricker, Tim Gleue, Patrick Daehne, and Luis Almeida. Archeoguide: an augmented reality guide for archaeological sites. *IEEE Computer Graphics and Applications*, 22(5):52–60, 2002.

[80] David Ingram. Trust-based filtering for augmented reality. In *International Conference on Trust Management*, pages 108–122. Springer, 2003.

[81] Ivan E Sutherland. A head-mounted three dimensional display. In *Proceedings of the December 9-11, 1968, fall joint computer conference, part I*, pages 757–764, 1968.

[82] Fang Hu, Dan Xie, and Shaowu Shen. On the application of the internet of things in the field of medical and health care. In *2013 IEEE international conference on green computing and communications and IEEE Internet of Things and IEEE cyber, physical and social computing*, pages 2053–2058. IEEE, 2013.

[83] Dominik Willers. Augmented reality at airbus. In *International symposium on mixed & augmented reality*, 2006.

[84] Huberta Kritzenberger, Thomas Winkler, and Michael Herczeg. Collaborative and constructive learning of elementary school children in experiental learning spaces along the virtuality continuum. *Mensch & Computer 2002: Vom interaktiven Werkzeug zu kooperativen Arbeits-und Lernwelten*, pages 115–124, 2002.

[85] Yan Guo, Qingyun Du, Yi Luo, Weiwei Zhang, and Lu Xu. Application of augmented reality gis in architecture. *The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 37:331–336, 2008.

[86] Gun Lee and Mark Billinghurst. Cityviewar outdoor ar visualization. In *Proceedings of the 13th International Conference of the NZ Chapter of the ACM's Special Interest Group on Human-Computer Interaction*, pages 97–97, 2012.

[87] Robert Aish. Bentley systems. *Migration from an individual to an enterprise computing model and its implications for AEC Research.*

[88] Hiroshi Ishii, John Underkoffler, Dan Chak, Ben Piper, Eran Ben-Joseph, Luke Yeung, and Zahra Kanji. Augmented urban planning workbench: overlaying drawings, physical models and digital simulation. In *Proceedings. International Symposium on Mixed and Augmented Reality*, pages 203–211. IEEE, 2002.

[89] Marco Sacco, Stefano Mottura, Luca Greci, and Giampaolo Vigan. Insitute of industrial technologies and automation. *National Research Council, Italy*, 2011.

[90] Julie Carmigniani, Borko Furht, Marco Anisetti, Paolo Ceravolo, Ernesto Damiani, and Misa Ivkovic. Augmented reality technologies, systems and applications. *Multimedia tools and applications*, 51:341–377, 2011.

[91] Brett Ridel, Patrick Reuter, Jeremy Laviole, Nicolas Mellado, Nadine Couture, and Xavier Granier. The revealing flashlight: Interactive spatial augmented reality for detail exploration of cultural heritage artifacts. *Journal on Computing and Cultural Heritage (JOCCH)*, 7(2):1–18, 2014.

[92] Anastassia Angelopoulou, Daphne Economou, Vassiliki Bouki, Alexandra Psarrou, Li Jin, Chris Pritchard, and Frantzeska Kolyda. Mobile augmented reality for cultural heritage. In *Mobile Wireless Middleware, Operating Systems, and Applications: 4th International ICST*

Conference, Mobilware 2011, London, UK, June 22-24, 2011, Revised Selected Papers 4, pages 15–22. Springer, 2012.

[93] Jiyoung Kang. Ar teleport: Digital reconstruction of historical and cultural-heritage sites for mobile phones via movement-based interactions. *Wireless personal communications*, 70:1443–1462, 2013.

[94] Li Yi-bo, Kang Shao-peng, Qiao Zhi-hua, and Zhu Qiong. Development actuality and application of registration technology in augmented reality. In *2008 international symposium on computational intelligence and design*, volume 2, pages 69–74. IEEE, 2008.

[95] Gerhard Reitmayr and Dieter Schmalstieg. Location based applications for mobile augmented reality. In *Proceedings of the Fourth Australasian user interface conference on User interfaces 2003-Volume 18*, pages 65–73, 2003.

[96] Assaf Feldman, Emmanuel Munguia Tapia, Sajid Sadi, Pattie Maes, and Chris Schmandt. Reachmedia: On-the-move interaction with everyday objects. In *Ninth IEEE International Symposium on Wearable Computers (ISWC'05)*, pages 52–59. IEEE, 2005.

[97] Fabio D'Agnano, Catherina Balletti, Francesco Guerra, and Paolo Vernier. Tooteko: A case study of augmented reality for an accessible cultural heritage. digitization, 3d printing and sensors for an audio-tactile experience. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 40:207–213, 2015.

[98] Ana Javornik, Yvonne Rogers, Ana Maria Moutinho, and Russell Freeman. Revealing the shopper experience of using a" magic mirror" augmented reality make-up application. In *Conference on designing interactive systems*, volume 2016, pages 871–882. Association for Computing Machinery (ACM), 2016.

[99] Sarah E Reed, Oliver Kreylos, Sherry Hsi, Louise H Kellogg, Geoffrey Schladow, M Burak Yikilmaz, Heather Segale, Julie Silverman, Steve Yalowitz, and Elissa Sato. Shaping watersheds exhibit: An interactive, augmented reality sandbox for advancing earth science education. In *AGU Fall Meeting Abstracts*, volume 2014, pages ED34A–01, 2014.

[100] Mauro Figueiredo, Maria-Ángeles Cifredo-Chacón, and Vítor Gonçalves. Learning programming and electronics with augmented reality. In *Universal Access in Human-Computer Interaction. Users and Context Diversity: 10th International Conference, UAHCI 2016, Held as Part of HCI International 2016, Toronto, ON, Canada, July 17-22, 2016, Proceedings, Part III 10*, pages 57–64. Springer, 2016.

[101] Xiaoyun Duan, Shin-Jin Kang, Jong In Choi, and Soo Kyun Kim. Mixed reality system for virtual chemistry lab. *KSII Transactions on Internet and Information Systems (TIIS)*, 14(4):1673–1688, 2020.

[102] Dorin Aiteanu, Bernd Hillers, and Axel Graser. A step forward in manual welding: demonstration of augmented reality helmet. In *The Second IEEE and ACM International Symposium on Mixed and Augmented Reality, 2003. Proceedings.*, pages 309–310. IEEE, 2003.

[103] Sebeom Park, Shokhrukh Bokijonov, and Yosoon Choi. Review of microsoft hololens applications over the past five years. *Applied sciences*, 11(16):7259, 2021.

[104] Zhihan Lv, Liangbing Feng, Haibo Li, and Shengzhong Feng. Hand-free motion interaction on google glass. In *SIGGRAPH Asia 2014 Mobile Graphics and Interactive Applications*, pages 1–1. 2014.

[105] Volker Paelke. Augmented reality in the smart factory: Supporting workers in an industry 4.0. environment. In *Proceedings of the 2014 IEEE emerging technology and factory automation (ETFA)*, pages 1–4. IEEE, 2014.

[106] Hui-Woog Choe, Yong Ju Kim, Jung Hee Park, Takefumi Morizumi, Emil F Pai, Norbert Krauss, Klaus Peter Hofmann, Patrick Scheerer, and Oliver P Ernst. Crystal structure of metarhodopsin ii. *Nature*, 471(7340):651–655, 2011.

[107] Jon Peddie and Jon Peddie. Types of augmented reality. *Augmented Reality: Where We Will All Live*, pages 29–46, 2017.

[108] Chryssi Birliraki, George Margetis, Nikolaos Patsiouras, Giannis Drossis, and Constantine Stephanidis. Enhancing the customers' experience using an augmented reality mirror. In *HCI International 2016–Posters' Extended Abstracts: 18th International Conference, HCI International 2016 Toronto, Canada, July 17–22, 2016 Proceedings, Part II 18*, pages 479–484. Springer, 2016.

[109] Abdulmotaleb El Saddik, Mauricio Orozco, Mohamad Eid, and Jongeun Cha. *Haptics technologies: Bringing touch to multimedia.* Springer Science & Business Media, 2011.

[110] Mafkereseb Kassahun Bekele, Roberto Pierdicca, Emanuele Frontoni, Eva Savina Malinverni, and James Gain. A survey of augmented, virtual, and mixed reality for cultural heritage. *Journal on Computing and Cultural Heritage (JOCCH)*, 11(2):1–36, 2018.

[111] Sylvain Cardin, Howard Ogden, Daniel Perez-Marcos, John Williams, Tomo Ohno, and Tej Tadi. Neurogoggles for multimodal augmented reality. In *Proceedings of the 7th Augmented Human International Conference 2016*, pages 1–2, 2016.

[112] Dhiraj Amin and Sharvari Govilkar. Comparative study of augmented reality sdks. *International Journal on Computational Science & Applications*, 5(1):11–26, 2015.

[113] Yuan Tian, Tao Guan, and Cheng Wang. Real-time occlusion handling in augmented reality based on an object tracking approach. *Sensors*, 10(4):2885–2900, 2010.

[114] Yuan Tian, Tao Guan, and Cheng Wang. An automatic occlusion handling method in augmented reality. *Sensor Review*, 2010.

[115] Taiki Fukiage, Takeshi Oishi, and Katsushi Ikeuchi. Reduction of contradictory partial occlusion in mixed reality by using characteristics of transparency perception. In *2012 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 129–139. IEEE, 2012.

[116] Silvio RR Sanches, Daniel M Tokunaga, Valdinei F Silva, Antonio C Sementille, and Romero Tori. Mutual occlusion between real and virtual elements in augmented reality based on fiducial markers. In *2012 IEEE Workshop on the Applications of Computer Vision (WACV)*, pages 49–54. IEEE, 2012.

[117] Yuzhu Lu and Shana Smith. Gpu-based real-time occlusion in an immersive augmented reality environment. *Journal of Computing and Information Science in Engineering*, 9(2), 2009.

[118] Kenichi Hayashi, Hirokazu Kato, and Shogo Nishida. Occlusion detection of real objects using contour based stereo matching. In *Proceedings of the 2005 international conference on Augmented tele-existence*, pages 180–186, 2005.

[119] H Setohara, H Kato, K Kawamoto, and K Tachibana. A simple solution of occlusion problem in augmented reality and its application for interaction. *Transactions of the Virtual Reality Society of Japan*, 9(4):387–395, 2004.

[120] Hansung Kim, Seung-jun Yang, and Kwanghoon Sohn. 3d reconstruction of stereo images for interaction between real and virtual worlds. In *The Second IEEE and ACM International Symposium on Mixed and Augmented Reality, 2003. Proceedings.*, pages 169–176. IEEE, 2003.

[121] Yuichi Ohta, Yasuyuki Sugaya, Hiroki Igarashi, Toshikazu Ohtsuki, and Kaito Taguchi. Share-z: Client/server depth sensing for see-through head-mounted displays. *Presence: Teleoperators & Virtual Environments*, 11(2):176–188, 2002.

[122] Naokazu Yokoya. Stereo vision based video see-through mixed reality. In *Proc. 1ˆ< st> Int. Symp. On Mixed Reality, Yokohama (March 1999)*, 1999.

[123] Yuan Tian, Yan Long, Dan Xia, Huang Yao, and Jincheng Zhang. Handling occlusions in augmented reality based on 3d reconstruction method. *Neurocomputing*, 156:96–104, 2015.

[124] Yuzhu Lu and Shana Smith. Gpu-based real-time occlusion in an immersive augmented reality environment. *Journal of Computing and Information Science in Engineering*, 9(2), 2009.

[125] Vincent Lepetit and M-O Berger. A semi-automatic method for resolving occlusion in augmented reality. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition. CVPR 2000 (Cat. No. PR00662)*, volume 2, pages 225–230. IEEE, 2000.

[126] Cristina Portalés, José Luis Lerma, and Santiago Navarro. Augmented reality and photogrammetry: A synergy to visualize physical and virtual city environments. *ISPRS Journal of Photogrammetry and Remote Sensing*, 65(1):134–142, January 2010.

[127] B Carrión-Ruiz, S Blanco-Pons, A Weigert, Stephen Fai, and JL Lerma. Merging photogrammetry and augmented reality: The canadian library of parliament. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 42(2/W11):367–371, 2019.

[128] Joseph J LaViola Jr. A survey of hand posture and gesture recognition techniques and technology. 1999.

[129] Samir Benbelkacem, Nadia Zenati-Henda, Fayçal Zerarga, Abdelkader Bellarbi, Mahmoud Belhocine, Salim Malek, and Mohamed Tadjine. Augmented reality platform for collaborative e-maintenance systems. *Augmented Reality-Some Emerging Application Areas, InTech*, pages 211–226, 2011.

[130] Kouichi Murakami and Hitomi Taguchi. Gesture recognition using recurrent neural networks. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 237–242, 1991.

[131] Ekaterini Stergiopoulou and Nikos Papamarkos. Hand gesture recognition using a neural network shape fitting technique. *Engineering Applications of Artificial Intelligence*, 22(8):1141–1158, 2009.

[132] Haitham Hasan and S Abdul-Kareem. Retracted article: Static hand gesture recognition using neural networks. *Artificial Intelligence Review*, 41:147–181, 2014.

[133] Yen-Ting Chen and Kuo-Tsung Tseng. Multiple-angle hand gesture recognition by fusing svm classifiers. In *2007 IEEE International Conference on Automation Science and Engineering*, pages 527–530. IEEE, 2007.

[134] Robert Wang, Sylvain Paris, and Jovan Popović. 6d hands: markerless hand-tracking for computer aided design. In *Proceedings of the 24th annual ACM symposium on User interface software and technology*, pages 549–558, 2011.

[135] Markus Oberweger, Paul Wohlhart, and Vincent Lepetit. Hands deep in deep learning for hand pose estimation. *arXiv preprint arXiv:1502.06807*, 2015.

[136] Ao Tang, Ke Lu, Yufei Wang, Jie Huang, and Houqiang Li. A real-time hand posture recognition system using deep neural networks. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 6(2):1–23, 2015.

[137] Van-Hung Le and Hung-Cuong Nguyen. A survey on 3d hand skeleton and pose estimation by convolutional neu-ral network. *Adv Sci Technol Eng Syst J*, 5:144–159, 2020.

[138] Jonathan Tompson, Murphy Stein, Yann Lecun, and Ken Perlin. Real-time continuous pose recovery of human hands using convolutional networks. *ACM Transactions on Graphics (ToG)*, 33(5):1–10, 2014.

[139] Markus Oberweger and Vincent Lepetit. Deepprior++: Improving fast and accurate 3d hand pose estimation. In *Proceedings of the IEEE international conference on computer vision Workshops*, pages 585–594, 2017.

[140] Yidan Zhou, Jian Lu, Kuo Du, Xiangbo Lin, Yi Sun, and Xiaohong Ma. Hbe: Hand branch ensemble network for real-time 3d hand pose estimation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 501–516, 2018.

[141] Kuo Du, Xiangbo Lin, Yi Sun, and Xiaohong Ma. Crossinfonet: Multi-task information sharing based hand pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9896–9905, 2019.

[142] Ayan Sinha, Chiho Choi, and Karthik Ramani. Deephand: Robust hand pose estimation by completing a matrix imputed with deep features. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4150–4158, 2016.

[143] Chengde Wan, Thomas Probst, Luc Van Gool, and Angela Yao. Crossing nets: Combining gans and vaes with a shared latent space for hand pose estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 680–689, 2017.

[144] Liuhao Ge, Zhou Ren, and Junsong Yuan. Point-to-point regression pointnet for 3d hand pose estimation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 475–491, 2018.

[145] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in neural information processing systems*, 30, 2017.

[146] Liuhao Ge, Hui Liang, Junsong Yuan, and Daniel Thalmann. 3d convolutional neural networks for efficient and robust hand pose estimation from single depth images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1991–2000, 2017.

[147] Gyeongsik Moon, Ju Yong Chang, and Kyoung Mu Lee. V2v-posenet: Voxel-to-voxel prediction network for accurate 3d hand and human pose estimation from a single depth map. In *Proceedings of the IEEE conference on computer vision and pattern Recognition*, pages 5079–5088, 2018.

[148] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[149] Fuyang Huang, Ailing Zeng, Minhao Liu, Jing Qin, and Qiang Xu. Structure-aware 3d hourglass network for hand pose estimation from single depth image. *arXiv preprint arXiv:1812.10320*, 2018.

[150] Liuhao Ge, Hui Liang, Junsong Yuan, and Daniel Thalmann. Robust 3d hand pose estimation in single depth images: from single-view cnn to multi-view cnns. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3593–3601, 2016.

[151] Christian Zimmermann and Thomas Brox. Learning to estimate 3d hand pose from single rgb images. In *Proceedings of the IEEE international conference on computer vision*, pages 4903–4911, 2017.

[152] Shih-En Wei, Varun Ramakrishna, Takeo Kanade, and Yaser Sheikh. Convolutional pose machines. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 4724–4732, 2016.

[153] Zhe Cao, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Realtime multi-person 2d pose estimation using part affinity fields. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7291–7299, 2017.

[154] Adrian Spurr, Jie Song, Seonwook Park, and Otmar Hilliges. Cross-modal deep variational hand pose estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 89–98, 2018.

[155] Thomas Theodoridis, Theocharis Chatzis, Vassilios Solachidis, Kosmas Dimitropoulos, and Petros Daras. Cross-modal variational alignment of latent spaces. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 960–961, 2020.

[156] Adrian Spurr, Umar Iqbal, Pavlo Molchanov, Otmar Hilliges, and Jan Kautz. Weakly supervised 3d hand pose estimation via biomechanical constraints. In *European Conference on Computer Vision*, pages 211–228. Springer, 2020.

[157] Franziska Mueller, Florian Bernard, Oleksandr Sotnychenko, Dushyant Mehta, Srinath Sridhar, Dan Casas, and Christian Theobalt. Ganerated hands for real-time 3d hand

tracking from monocular rgb. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 49–59, 2018.

[158] Guan Ming Lim, Prayook Jatesiktat, and Wei Tech Ang. Mobilehand: Real-time 3d hand shape and pose estimation from color image. In *International Conference on Neural Information Processing*, pages 450–459. Springer, 2020.

[159] Liuhao Ge, Zhou Ren, Yuncheng Li, Zehao Xue, Yingying Wang, Jianfei Cai, and Junsong Yuan. 3d hand shape and pose estimation from a single rgb image. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10833–10842, 2019.

[160] Omid Taheri, Nima Ghorbani, Michael J Black, and Dimitrios Tzionas. Grab: A dataset of whole-body human grasping of objects. In *European conference on computer vision*, pages 581–600. Springer, 2020.

[161] Adnane Boukhayma, Rodrigo de Bem, and Philip HS Torr. 3d hand shape and pose from images in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10843–10852, 2019.

[162] Umar Iqbal, Pavlo Molchanov, Thomas Breuel Juergen Gall, and Jan Kautz. Hand pose estimation via latent 2.5 d heatmap regression. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 118–134, 2018.

[163] Endri Dibra, Silvan Melchior, Ali Balkis, Thomas Wolf, Cengiz Oztireli, and Markus Gross. Monocular rgb hand pose inference from unsupervised refinable nets. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 1075–1085, 2018.

[164] Franziska Mueller, Dushyant Mehta, Oleksandr Sotnychenko, Srinath Sridhar, Dan Casas, and Christian Theobalt. Real-time hand tracking under occlusion from an egocentric rgb-d sensor. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1154–1163, 2017.

[165] Evangelos Kazakos, Christophoros Nikou, and Ioannis A Kakadiaris. On the fusion of rgb and depth information for hand pose estimation. In *2018 25th IEEE International Conference on Image Processing (ICIP)*, pages 868–872. IEEE, 2018.

[166] Mohammad Mofarreh-Bonab, Hadi Seyedarabi, Behzad Mozaffari Tazehkand, and Shohreh Kasaei. 3d hand pose estimation using rgbd images and hybrid deep learning networks. *The Visual Computer*, pages 1–10, 2021.

[167] Chiho Choi, Ayan Sinha, Joon Hee Choi, Sujin Jang, and Karthik Ramani. A collaborative filtering approach to real-time hand pose estimation. In *Proceedings of the IEEE international conference on computer vision*, pages 2336–2344, 2015.

[168] Jiawei Zhang, Jianbo Jiao, Mingliang Chen, Liangqiong Qu, Xiaobin Xu, and Qingxiong Yang. A hand pose tracking benchmark from stereo matching. In *2017 IEEE International Conference on Image Processing (ICIP)*, pages 982–986. IEEE, 2017.

[169] Franziska Mueller, Florian Bernard, Oleksandr Sotnychenko, Dushyant Mehta, Srinath Sridhar, Dan Casas, and Christian Theobalt. Ganerated hands for real-time 3d hand tracking from monocular rgb. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 49–59, 2018.

[170] Guan Ming Lim, Prayook Jatesiktat, and Wei Tech Ang. Mobilehand: Real-time 3d hand shape and pose estimation from color image. In *International Conference on Neural Information Processing*, pages 450–459. Springer, 2020.

[171] Danhang Tang, Hyung Jin Chang, Alykhan Tejani, and Tae-Kyun Kim. Latent regression forest: Structured estimation of 3d articulated hand posture. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3786–3793, 2014.

[172] Shanxin Yuan, Qi Ye, Bjorn Stenger, Siddhant Jain, and Tae-Kyun Kim. Bighand2. 2m benchmark: Hand pose dataset and state of the art analysis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4866–4874, 2017.

[173] Xiao Sun, Yichen Wei, Shuang Liang, Xiaoou Tang, and Jian Sun. Cascaded hand pose regression. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 824–832, 2015.

[174] Shanxin Yuan, Qi Ye, Guillermo Garcia-Hernando, and Tae-Kyun Kim. The 2017 hands in the million challenge on 3d hand pose estimation. *arXiv preprint arXiv:1707.02237*, 2017.

[175] Aaron Wetzler, Ron Slossberg, and Ron Kimmel. Rule of thumb: Deep derotation for improved fingertip detection. *arXiv preprint arXiv:1507.05726*, 2015.

[176] Yi Yang and Deva Ramanan. Articulated human detection with flexible mixtures of parts. *IEEE transactions on pattern analysis and machine intelligence*, 35(12):2878–2890, 2012.

[177] Karan Ahuja, Paul Streli, and Christian Holz. Touchpose: Hand pose prediction, depth estimation, and touch classification from capacitive images. In *The 34th Annual ACM Symposium on User Interface Software and Technology*, pages 997–1009, 2021.

[178] Louis Moustakas. A bibliometric analysis of research on social cohesion from 1994–2020. *Publications*, 10(1):5, 2022.

[179] Srinath Sridhar, Antti Oulasvirta, and Christian Theobalt. Interactive markerless articulated hand motion tracking using rgb and depth data. In *Proceedings of the IEEE international conference on computer vision*, pages 2456–2463, 2013.

[180] Xiaoming Deng, Shuo Yang, Yinda Zhang, Ping Tan, Liang Chang, and Hongan Wang. Hand3d: Hand pose estimation using 3d neural network. *arXiv preprint arXiv:1704.02224*, 2017.

[181] Ihsan Rabbi and Sehat Ullah. A survey on augmented reality challenges and tracking. *Acta graphica: znanstveni časopis za tiskarstvo i grafičke komunikacije*, 24(1-2):29–46, 2013.

[182] Ya Zhou, Jin-Tao Ma, Qun Hao, Hong Wang, and Xian-Peng Liu. A novel optical see-through head-mounted display with occlusion and intensity matching support. In *International Conference on Technologies for E-Learning and Digital Entertainment*, pages 56–62. Springer, 2007.

[183] Nikhil S Potabatti. *Photogrammetry for 3D Reconstruction in SOLIDWORKS and its Applications in Industry.* PhD thesis, Purdue University Indianapolis, Indiana, 2019.

[184] Sebastian Bullinger, Christoph Bodensteiner, and Michael Arens. A photogrammetry-based framework to facilitate image-based modeling and automatic camera tracking. *arXiv preprint arXiv:2012.01044*, 2020.

[185] Gorka Kortaberria, Unai Mutilba, Eneko Gomez-Acedo, Alberto Tellaeche, and Rikardo Minguez. Accuracy evaluation of dense matching techniques for casting part dimensional verification. *Sensors*, 18(9):3074, 2018.

[186] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. 2017.

[187] Ikram Kourbane and Yakup Genc. A graph-based approach for absolute 3d hand pose estimation using a single rgb image. *Applied Intelligence*, pages 1–16, 2022.

[188] Chengying Gao, Yujia Yang, and Wensheng Li. 3d interacting hand pose and shape estimation from a single rgb image. *Neurocomputing*, 474:25–36, 2022.

[189] Shanxin Yuan, Bjorn Stenger, and Tae-Kyun Kim. Rgb-based 3d hand pose estimation via privileged learning with depth images. *arXiv preprint arXiv:1811.07376*, 2018.

[190] Market Research Future. Augmented reality (ar) market size to hit usd 461.25 billion at a cagr of 41.50Published on September 27, 2022.

[191] Shanxin Yuan, Qi Ye, Bjorn Stenger, Siddhant Jain, and Tae-Kyun Kim. Bighand2. 2m benchmark: Hand pose dataset and state of the art analysis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4866–4874, 2017.