

FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

Design and Evaluation of a VR-Based Framework for Assembly Training

Lourenço Alexandre Correia Gonçalves



Master in Informatics and Computing Engineering

Supervisor: Rui Pedro Ferreira Pinto

July 28, 2025

Design and Evaluation of a VR-Based Framework for Assembly Training

Lourenço Alexandre Correia Gonçalves

Master in Informatics and Computing Engineering

July 28, 2025

Resumo

Esta dissertação apresenta o desenho, a implementação e a avaliação de um sistema de treino em Realidade Virtual (RV) para orientar utilizadores na execução de tarefas de montagem manual. O trabalho foca-se em treino procedural generalizável, utilizando interações imersivas baseadas em movimentos das mãos. Foi desenvolvido um *framework* modular e extensível em Unity, incorporando funcionalidades como dependências lógicas entre tarefas, geração contextual de pistas e registo de desempenho.

Para avaliar a eficácia do sistema, foi realizado um estudo experimental controlado que comparou o treino baseado em RV com métodos de instrução tradicionais, no contexto de uma tarefa de montagem de uma secretária e cadeira baseada no modelo HRC (*Human-Robot Collaboration*). Os participantes foram divididos em dois grupos, tendo cada um realizado a tarefa após métodos de treino distintos. Foram recolhidas e analisadas métricas quantitativas, como o tempo de conclusão da tarefa e a taxa de erro, bem como feedback qualitativo sobre usabilidade.

Embora o sistema de treino em RV não tenha superado significativamente os métodos tradicionais, demonstrou uma eficácia comparável no suporte à aprendizagem procedural, e os participantes responderam positivamente à natureza imersiva e interativa do sistema. Estes resultados sugerem que a RV pode constituir uma alternativa viável às abordagens convencionais de formação, com potencial para melhorias futuras na experiência do utilizador e no design instrucional.

Abstract

This dissertation presents the design, implementation, and evaluation of a Virtual Reality (VR) training system for guiding users through manual assembly tasks. The work focuses on generalizable procedural training using immersive, hand-based interactions. A modular and extensible VR framework was developed in Unity, incorporating features such as logical task dependencies, contextual hint generation and performance logging.

To assess the system's effectiveness, a controlled experimental study compared VR-based training against traditional instructional methods in the context of a desk and chair assembly task based on the human-robot collaboration (HRC) model set. Participants were divided into two groups, each completing the task after undergoing different training methods. Quantitative metrics such as task completion time and error rate, alongside qualitative feedback on usability and engagement, were collected and analyzed.

While the VR-based training system did not significantly outperform traditional methods, it demonstrated comparable effectiveness in supporting procedural learning, and participants responded positively to the immersive and interactive nature of the system. These results suggest that VR can serve as a viable alternative to conventional training approaches, with potential for further improvements in user experience and instructional design.

Acknowledgements

I'd like to begin by thanking my parents for their continued support throughout my academic path. Your support allowed me to focus fully on my studies, knowing that I always had a solid foundation behind me.

A sincere thank you to my supervisor, Rui Pinto, for the guidance, feedback, and perspective you provided during the development of this dissertation.

Thanks as well to my colleagues, who stuck with me as we worked on various curricular projects, even when things didn't go as smoothly as planned.

Finally, to my friends: thank you for keeping me sane throughout my academic journey. Thank you for the laughs, trips, and late night chats. You made the tough parts easier and the good parts better.

Lourenço Gonçalves

“Any sufficiently advanced technology is indistinguishable from magic”

Arthur C. Clarke

Contents

1	Introduction	1
1.1	Context	1
1.2	Motivation	1
1.3	Problem	2
1.4	Objectives	3
1.5	Document Structure	3
2	Introduction to Extended Reality	5
2.1	Extended Reality Hardware	5
2.1.1	Technologies and Interaction Methods in Extended Reality	6
2.2	Virtual Reality	7
2.3	Augmented Reality	8
2.3.1	Technologies in Augmented Reality	8
2.4	Conclusion	9
3	State of the Art	10
3.1	Literature Review	10
3.1.1	Methodology Overview	10
3.2	Current Advances and Challenges	11
3.2.1	Positive Takeaways	11
3.2.2	Negative Takeaways	13
3.2.3	Critical Discussion	14
4	Framework Design	16
4.1	Technology Stack	16
4.2	Architecture Overview	17
4.2.1	Components	17
4.3	Design Principles	24
4.3.1	Modularity, Scalability, Extensibility and Maintainability	24
4.3.2	Interoperability and Standardization	24
4.3.3	Security and Data Integrity	25
4.3.4	Human-Centered Design	25
4.4	System Behavior	25
4.4.1	Piece Placement and Removal	25
4.4.2	On Piece Placed/Removed	27
4.5	Current Limitations	28

5	Assembly Scenario Design and Implementation	30
5.1	HRC Model Set Overview	30
5.2	Case Study: Custom Desk and Chair Assembly	30
5.2.1	Physical Prototype Implementation	30
5.3	VR Training Framework Implementation	34
5.3.1	3D Model Preparation for Unity	34
5.3.2	Assembly Creation	34
6	Study Design	36
6.1	Participants & Training Groups	36
6.2	Experimental Procedure	36
6.3	Evaluation Metrics	38
6.3.1	Quantitative Metrics	38
6.3.2	Qualitative Metrics	38
6.4	Questionnaires	39
6.4.1	Consent Form	39
6.4.2	Basic Information Questionnaire	40
6.4.3	Virtual Reality Training Experience Questionnaire	40
6.4.4	Traditional Training Experience	41
6.4.5	Assembly Task Experience	41
7	Results and Discussion	42
7.1	Performance During Virtual Reality Training	42
7.1.1	Task Completion Time	42
7.1.2	Number of errors	43
7.2	Real-World Assembly Task Results	43
7.2.1	Statistical analysis	44
7.2.2	Descriptive Analysis	47
7.3	Subjective Feedback	48
7.3.1	Virtual Reality Training Feedback	49
7.3.2	Traditional Training Feedback	50
7.3.3	Real-world Assembly Feedback	51
7.4	Discussion	52
7.4.1	Learning Trends in Virtual Reality	52
7.4.2	Transfer of Knowledge to Real-World Performance	53
7.4.3	Error Behavior and Technology Limitations	53
7.4.4	User Feedback	53
7.4.5	Implications and Future Directions	54
8	Conclusions	55
8.1	Main Contributions	55
8.2	Future work	56
9	Evaluation of the Work in the Context of the United Nations Sustainable Development Goals	57
	References	59

A Results (Raw Data)	62
A.1 VR Training Data	62
A.2 Real-World Assembly Data	63

List of Figures

4.1	Component Diagram	18
4.2	Palm Menu	22
4.3	Assembly Progress Interface	24
4.4	Piece placement and removal sequence diagram	26
4.5	onPiecePlaced sequence diagram	28
5.1	Completed physical desk and chair assembly	31
5.2	3D Printed Foot	32
5.3	Top90 Connector	32
5.4	Shelf90 Bracket	32
5.5	Shelf180 Bracket	32
5.6	Chair Left Side	32
5.7	Chair Right Side	32
5.8	Chair Back Support	33
5.9	Completed chair and desk assembly (Virtual)	34
5.10	Chair Assembly Process	35
5.11	Desk Assembly Process	35
7.1	Evolution of the average time per task	42
7.2	Evolution of the average mistakes per task	43
7.3	VR Group Time Normality	44
7.4	Traditional Group Time Normality	45
7.5	Correlation between VR and real-world completion time	45
7.6	VR Group Errors Normality	46
7.7	Traditional Group Errors Normality	46
7.8	Correlation between VR and real-world errors	47
7.9	Participants with n errors for each group	48
7.10	VRSUQ Results	49
7.11	VR Training Feedback	50
7.12	Traditional Training Feedback	51
7.13	Real-world Assembly Feedback	52

List of Tables

5.1	Parts and materials for the assembly task	33
7.1	Descriptive statistics	47
9.1	Evaluation of the VR Training System Against Selected UN Sustainable Development Goals	58
A.1	VR Task Performance Data	62
A.2	Real-World Task Performance Data	63

Abbreviations

AR	Augmented Reality
CSV	Comma-Separated Values
HMD	Head-Mounted Display
HRC	Human-Robot Collaboration
PLA	Polylactic Acid
SDG	Sustainable Development Goals
SDK	Software Development Kit
SLAM	Simultaneous Localization and Mapping
STL	Standard Tessellation Language
UI	User Interface
URP	Universal Render Pipeline
VR	Virtual Reality
VRSUQ	Virtual Reality System Usability Questionnaire
XR	Extended Reality

Chapter 1

Introduction

1.1 Context

Virtual Reality (VR) and Augmented Reality (AR) technologies have seen rapid growth in recent years [1] and are increasingly applied in industrial domains, including manufacturing and assembly tasks [2, 3]. This growth is driven by advancements in hardware and software, decreasing costs, and growing public interest. Simultaneously, industries that involve high-risk or logistically complex hands-on training environments are exploring the integration of advanced technologies to improve training efficacy and operational efficiency.

Modern production environments often require workers to learn complex, multi-step procedures. Traditional training methods, which depend heavily on supervised, hands-on practice with real equipment, can be time-consuming, expensive, and sometimes even dangerous. In many cases, training on hazardous assembly machines isn't feasible or safe in a real-world setting. Classroom instruction alone also tends to fall short in preparing trainees for the hands-on challenges they'll face on the factory floor. In comparison, virtual and augmented reality technologies offer immersive, interactive simulations that can safely replicate real assembly environments and procedures [2, 4]. These simulations give trainees the opportunity to practice tasks repeatedly, receive instant feedback, and explore different approaches without risking injury or damaging equipment. Research has shown that VR-based simulators can match the effectiveness of physical training in terms of skill development and error reduction [5] without the dangers associated with real-world tools and components [4, 6]. As a result, VR and AR training methods have the potential to address many of the challenges posed by traditional approaches, providing a safer, more flexible, and scalable learning experience.

1.2 Motivation

Given the need for efficient and safe training in assembly industries, immersive extended reality (XR) technologies show strong potential. Virtual reality simulations can replicate real assembly

procedures and environments, allowing trainees to develop skills in a realistic yet controlled setting. For example, a field study conducted at an automotive plant found that workers trained in an immersive **VR** environment performed just as well on actual assembly tasks as those trained through traditional methods, with results showing continued improvement and positive skill transfer over time [6]. Similarly, training with **AR** can help improve both accuracy and confidence. Head-mounted **AR** systems have been shown to reduce assembly errors and the need for rework when compared to traditional, unassisted training methods, even though users may work more slowly at first as they adjust to the technology [7]. In general, mixed-reality training tends to be at least as effective as traditional instruction, and **AR** in particular often shows advantages in improving long-term retention and task performance [3, 7].

Beyond performance metrics, **VR** and **AR** training offer practical advantages that align well with industry needs. Since **VR** training is entirely virtual, it avoids the material costs and equipment downtime that come with physical training setups. Virtual exercises can be repeated as often as needed without causing wear on real machinery. For high-risk or specialized assembly tasks, like working with toxic materials or complex medical devices, **VR** provides a safe, controlled environment where operators can gain valuable experience that would be difficult or dangerous to acquire otherwise [4]. More broadly, **XR** training can increase trainee engagement and confidence, shorten learning curves, and allow distributed or remote learners to access the same high-quality instruction.

In summary, by offering immersive rehearsal of assembly processes with immediate feedback, **VR** and **AR** have the potential to accelerate skill acquisition, reduce errors, and lower training costs in manufacturing settings [6, 7].

1.3 Problem

Despite these advantages, there remain several challenges that limit the current effectiveness of **VR** and **AR** for assembly training. First, many existing **VR** training systems rely on hand-held controllers, resulting in unnatural or indirect interaction with virtual parts.

Usability and comfort are also concerns. Users of head-mounted displays (HMD) frequently report higher cognitive workload and lower subjective usability compared to traditional training. For example, one controlled study found that participants in a **VR**-based assembly training reported significantly more perceived task load and lower system usability than those in **AR** or video-based training [2]. Prolonged use of head-mounted displays can cause eyestrain, neck fatigue, or motion sickness, which may limit the duration of training sessions.

Finally, while the cost of **VR** and **AR** hardware has steadily decreased over recent years, making advanced headsets and tracking devices more accessible than before, challenges remain in the broader deployment of these technologies. Developing effective training software tailored to specific assembly tasks often involves significant time and resources, which can be a barrier for many organizations, especially smaller ones. Additionally, integrating **VR** and **AR** solutions into existing workflows requires careful planning and expertise, which further adds to implementation

complexity. Although the hardware is becoming more affordable and user-friendly, the software development and system integration demands continue to slow down widespread adoption in some sectors [8]. Thus, overcoming these practical challenges alongside issues such as sensory fidelity and user comfort is crucial for realizing the full potential of immersive training in assembly industries.

1.4 Objectives

This dissertation aims to develop a training framework that tackles the challenges identified in the context of assembly training, leveraging the latest advancements in virtual reality technology to improve interaction fidelity and user comfort. The framework will address issues such as unnatural input methods, cognitive workload, usability concerns, and implementation complexity. A study will then be conducted to evaluate its effectiveness by addressing the following research questions:

RQ1 Is virtual reality training effective in supporting skill acquisition and real-world knowledge transfer in assembly tasks?

RQ2 How does virtual reality training compare to traditional instructional methods in terms of learning effectiveness and user experience in assembly tasks?

These questions form the basis for investigating the potential of virtual reality as a viable training solution in manufacturing contexts. To guide this investigation, the dissertation proposes the following hypotheses:

H1 Virtual reality training supports effective skill acquisition and enables meaningful transfer of knowledge to real-world assembly tasks. (RQ1)

H2 Participants trained using virtual reality will demonstrate equal or superior learning effectiveness and report a more positive user experience compared to those trained through traditional instructional methods. (RQ2)

1.5 Document Structure

This document is organized into eight chapters. Chapter 2 provides an introduction to XR technologies. Chapter 3 presents a state-of-the-art review, examining current research, challenges, and trends in XR assembly training systems. Chapter 4 details the design and development of the virtual training framework, explaining its architecture, features and limitations. Chapter 5 focuses on the design and implementation of a specific assembly scenario used to evaluate the framework, including the creation of physical and digital prototypes, and their corresponding assembly process. Chapter 6 outlines the experimental study design, describing participant selection, procedures, and evaluation metrics. Chapter 7 presents the collected data, followed by an analysis and discussion

of the results. Chapter 8 summarizes the key findings, implications, and potential future work. Finally, Chapter 9 highlights the contributions of this research in accordance with the United Nations Sustainable Development Goals.

Chapter 2

Introduction to Extended Reality

Extended Reality (**XR**) refers to the entire spectrum of immersive environments, ranging from fully virtual worlds (Virtual Reality, **VR**) to augmented overlays on the real world (Augmented Reality, **AR**). **XR** encompasses **AR**, **VR**, and Mixed Reality (**MR**), using advanced sensors and computing to blend physical and digital experiences.

One review defines **XR** as providing “virtual and augmented ways to interact with information in the real and virtual worlds” [9]. Another describes **XR** as an umbrella term offering immersive and interactive alternatives to traditional methods, preserving core educational values while introducing innovative digital experiences [10].

XR is already used in various domains, including industrial training, entertainment, and spatial navigation, enabling users to experience or annotate reality in novel ways [9, 10].

2.1 Extended Reality Hardware

XR experiences depend on various hardware types:

Head-Mounted Displays

Head-mounted displays (**HMDs**) are wearable headsets provide either full immersion (**VR**) or see-through overlays (**AR**). Examples include Meta Quest, HTC Vive, Valve Index (**VR**), and Microsoft HoloLens or Magic Leap (**AR**).

Smartphones and Tablets

Common mobile devices with cameras and sensors (gyroscope, accelerometer) can run **AR** applications. They provide a moderate immersion at low cost and wide accessibility, making **AR** experiences easy to deploy (for navigation, gaming, etc.) without special hardware [9, 10].

Projection-Based Systems

Room-scale systems like CAVEs (Cave Automatic Virtual Environments) and augmented projection setups use multiple projectors on walls/floors. These environments can display virtual or augmented content on real surfaces, allowing collaborative XR experiences. Projection-based XR was common in early VR research (“PowerWalls, CAVEs”) and later adapted for AR. For example, “projection-based augmented reality” systems can overlay design models onto physical prototypes, offering a collaborative workspace [11]. Although highly immersive, projection systems are typically non-portable and costly compared to HMDs [11].

2.1.1 Technologies and Interaction Methods in Extended Reality

XR systems integrate a variety of interaction methods and tracking technologies to make experiences intuitive and immersive:

Tracking Systems

- **Inside-Out Tracking:** Cameras or sensors on the device track environmental features to determine user/device position. Many standalone HMDs (Meta Quest, HoloLens) use inside-out tracking for six degrees of freedom (3 translational, 3 rotational) motion sensing [12].
- **Outside-In Tracking:** External cameras or beacons (e.g. Infrared base stations in older VR setups) monitor markers on the user/controllers. This can offer very precise tracking in a fixed volume [13].
- **Body Tracking:** Advanced setups use motion-capture or depth sensors to track full-body movement, enabling the user’s entire body or limbs to appear in the virtual space.

Haptic Feedback

Haptic devices (vibration motors, force-feedback controllers or gloves) deliver tactile sensations that simulate touch or impact. These feedback cues (“vibrations and pressure”) enhance realism and user presence. Research shows that adding tactile/haptic stimulation makes virtual objects “feel more realistic by enhancing the user’s sense of presence and physical engagement” [14]. For example, haptic gloves or vests can simulate feeling a virtual surface or the recoil of firing a weapon, thereby deepening immersion.

Eye-Tracking and Foveated Rendering

Eye-tracking sensors in an HMD monitor the user’s gaze direction. This allows gaze-based interaction (e.g. looking at a menu item to select it) [15] and also enables foveated rendering, where the system renders high-resolution graphics only where the eye is looking, while reducing detail in the periphery. Foveated rendering has been shown to “decrease computation” significantly without perceptible quality loss, because it exploits the eye’s natural focus [16]. In practice, eye-tracking

and foveated rendering greatly improves performance on limited hardware while still maintaining visual fidelity in the user's focal region [16]. Foveated rendering is still used in HMDs that don't support eye-tracking, by rendering the center of the image, where the user's gaze will be focused most of the time, with higher quality [17].

Hand Tracking and Gesture Recognition

Camera-based or sensor gloves can track hands and fingers, allowing users to interact naturally with virtual objects without controllers, providing a natural and intuitive interface. As one review notes, hand gestures "provide a natural means of interaction that can be used across a variety of applications" [18]. In AR, recognizing hand poses or gestures (e.g. pinch, grab) lets users manipulate holograms as if touching them. Many AR/VR platforms now include built-in hand tracking for hand interactions.

Voice Commands

Speech interfaces allow hands-free control of XR applications. Users can issue commands or input text via voice, which is useful when manual interaction is impractical. Modern XR systems often support local or cloud-based speech recognition (e.g. Cortana on HoloLens) [19].

2.2 Virtual Reality

Virtual Reality surrounds users in a completely digital environment, replacing the physical world with a virtual computer-generated one [9, 20]. VR offers an immersive and controlled experience, which is ideal for simulation, training, and entertainment.

Hardware for Virtual Reality

Virtual Reality primarily relies on head-mounted displays along with projection-based systems. HMDs, like the Meta Quest, are capable of very immersive experiences by incorporating complex features such as motion tracking and high-resolution displays. These devices are portable and accessible and, therefore, are the most widely used for virtual reality applications [21]. Projection-based systems, on the other hand, are not portable and are much more expensive than HMDs [21].

Technologies in Virtual Reality

Motion controllers are widely used to allow users to pick up, manipulate, and operate virtual tools or objects. Hand-based controls, reliant on hand tracking and hand gesture recognition, are another alternative for interacting with the virtual world. Full-body tracking is commonly employed in social VR and fitness applications, where the user's real body movements are reflected in their avatars, enhancing immersion and presence. Haptic devices provide tactile feedback during virtual experiences, such as simulating the sensation of touching surfaces, feeling resistance when

handling objects, or sensing impacts in games and training scenarios [14]. Eye-tracking supports gaze-based interaction methods, for example allowing users to select interface elements simply by looking at them, and it also enables foveated rendering to improve graphical performance [15, 16, 17]. Spatial audio is used to create realistic sound environments where audio sources move relative to the user's head position, such as footsteps approaching from behind or ambient sounds changing as users explore virtual spaces, thereby reinforcing spatial awareness and presence [22].

2.3 Augmented Reality

Augmented Reality (AR) overlays digital information onto the real world, enhancing rather than replacing the user's perception. AR can be used for navigation aids (overlying directions on the road), training (displaying assembly instructions on machinery), and interactive guidance (e.g. step-by-step repair tutorials overlaid on equipment). By blending the virtual and real worlds, AR supports tasks where overlaying contextual information is important. [23]

Hardware for Augmented Reality

AR Headsets or Smart Glasses, such as see-through head-mounted displays like the Microsoft HoloLens and Magic Leap, display digital content directly within the user's field of view. These devices are equipped with sensors for spatial mapping and hand tracking. For example, the HoloLens 2 provides real-time environment scanning and fully articulated two-hand tracking [24]. Such headsets can pin holograms to physical locations or allow users to manipulate AR objects using gestures or voice commands.

Mobile devices, including almost all modern smartphones and tablets, support augmented reality through their built-in cameras and sensors. Utilizing frameworks such as ARKit or ARCore, these devices perform basic simultaneous localization and mapping (SLAM) and can overlay 3D models on live video feeds. Although less immersive than dedicated AR glasses, mobile AR offers massive accessibility and cost-effectiveness, enabling quick AR experiences such as virtually placing furniture within a room.

Projection systems for AR operate similarly to VR CAVES by casting images onto real-world surfaces, like a physical workbench, to provide shared AR experiences without the need for wearable devices. This approach is particularly useful for collaborative design or exhibition spaces; however, projection-based AR systems generally lack portability.

2.3.1 Technologies in Augmented Reality

Key AR technologies rely on advanced tracking and recognition to align virtual content with the real world. Simultaneous Localization And Mapping (SLAM) is fundamental in AR, using camera and sometimes depth sensor data to construct a 3D map of the environment and track the devices position in real time [25]. This capability ensures virtual objects stay anchored as users move, making them appear fixed within the physical space [25].

Gesture recognition enables natural manipulation of **AR** content by interpreting hand poses and motions. Users can perform actions such as pinching or pointing to grab and move holograms. Studies indicate that hand gestures, often combined with other input methods, are key to intuitive **AR** control [18].

Voice commands provide a hands-free interaction mode, allowing users to summon menus, issue commands, or annotate scenes when manual input is impractical.

Beyond **SLAM**, environment and object mapping allow **AR** systems to recognize surfaces or specific items. Advanced **AR** headsets maintain a real-time 3D mesh of surroundings (spatial mapping) to accurately occlude virtual objects and enable complex interactions [24].

2.4 Conclusion

XR technologies bring together a large array of technologies to create immersive experiences. Virtual Reality shines when it comes to fully simulated environments. Augmented Reality, on the other hand, adds useful digital information right on top of the real world, enhancing how we interact with our surroundings. Both rely on smart tracking methods, feedback such as haptics and spatial audio, and natural ways to interact (like gestures, eye movement, and voice commands) to enhance the experience.

Chapter 3

State of the Art

3.1 Literature Review

This section details the approach taken to identify, collect, screen, and select the most relevant academic contributions to the field of operator guidance methodologies in **VR/AR**, particularly in the context of assembly processes. This review aims to ensure that the selected papers represent a comprehensive overview of the current state of research, as well as the whole context surrounding the application of **VR/AR** technologies in the industry.

3.1.1 Methodology Overview

The process was divided into four distinct phases: defining eligibility criteria, collecting the relevant literature, screening papers to refine the selection, and choosing the final set of studies that best address the topic at hand.

3.1.1.1 Eligibility Criteria

Given the review's focus on **VR/AR**-enhanced operator guidance within assembly, specific eligibility criteria were defined to ensure relevance and academic rigor. Only papers published after 2016 were included, as this cutoff date helps to capture the latest advancements in **VR/AR** applications for industrial environments. In addition, only studies published in English were considered.

This review focused on research articles examining **VR** and/or **AR** training within the context of Industry 5.0. Although the primary focus of this dissertation is on assembly training, broader research in closely related areas was also deemed relevant. Studies discussing only general **VR/AR** applications, without a clear link to operator guidance and training, were excluded.

3.1.1.2 Search Strategy

The search strategy employed IEEE Xplore, ACM Digital Library, ScienceDirect, and Google Scholar, using the following query to yield broad and comprehensive results across all databases:

```
("virtual reality" OR "augmented reality" OR "VR" OR "AR") AND  
("operator guidance" OR "assembly training")
```

3.1.1.3 Screening Process

Preliminary Screening

The first screening involved reviewing the titles, abstracts, keywords and conclusions of all collected papers. Papers that clearly did not address **VR/AR** in operator guidance or those that focused on unrelated fields were excluded. In papers that seemed only tangentially related, a more in-depth review was conducted to verify their relevance.

Full-Text Screening This stage ensured that the papers met all eligibility criteria and offered relevant insights into **VR/AR** applications for operator guidance. Priority was given to studies that directly addressed **VR/AR**'s role in guiding operators within assembly tasks, although papers with broader industrial applications were also included if they provided valuable insights. Each study was expected to clearly explain its methodology, detailing how **VR/AR** systems were implemented and evaluated.

Both positive and negative research results were deemed relevant, as positive findings can showcase effective practices and improvements, while negative results can reveal challenges and limitations that must be considered during the development of the solution.

3.1.1.4 Selection of Final Papers

Several criteria were applied to ensure the final selection of studies upheld both academic rigor and relevance. Papers with a substantial citation count were favored, as this reflects their influence within academic and industrial communities. Nonetheless, emerging research with fewer citations was also included if it demonstrated noteworthy innovation. Studies that documented **VR/AR** applications in real-world assembly were particularly valued, as they offer practical insights relevant to researchers and industry practitioners alike. In the end, 43 papers were selected.

3.2 Current Advances and Challenges

This section discusses the current state of the art in training and assembly tasks, highlighting effective methods and their limitations.

3.2.1 Positive Takeaways

3.2.1.1 Virtual Reality

VR environments provide immersive experiences, substantially improving the transfer of learning and retention. Such systems prove particularly advantageous for sequence learning and complex tasks, as they enable users to engage in practice within controlled virtual settings. Constructive feedback from trainees significantly highlights **VR**'s ability to engage users effectively [6].

Semi-immersive **VR** systems maintain user engagement without causing fatigue when session durations are optimized (1–2 hours). Gamification elements in **VR** can further enhance user motivation, task understanding, and performance [26, 27].

A comprehensive study by Werrlich *et al.* [28] showed that **HMD**-based **VR** training offers better user satisfaction with lower cognitive workload compared to traditional paper-based methods. Participants praised the multimodal interaction, including voice commands and improved visual guidance, which simplified complex assembly tasks.

Results from Wolfartsberger *et al.* [29] showed that reduced levels of assistance in **VR** training applications increased cognitive engagement, improving learning outcomes. For instance, participants in a non-assisted **VR** training environment had results similar to those receiving tutoring, achieving comparable assembly times and error rates while experiencing a lower mental load.

Schwarz *et al.* [6] presented an evaluation of immersive **VR** assembly training, which indicated that **VR** performed similarly to conventional training regarding skill transfer and retention. However, **VR** had distinct advantages in learning task sequences and familiarity. User-centered experiences were highly engaging and comfortable during training.

3.2.1.2 Augmented Reality

The performance of **AR** systems was prominent in enhancing task precision and minimizing errors, especially in occluded assembly tasks. **AR**'s visual guidance helps locate components, reducing cognitive load and task completion times [5].

Chu *et al.* [5] explored the effectiveness of **AR** in guiding occluded component assemblies, demonstrating that visual cues, such as tracking hand movement, significantly reduced assembly times for complex tasks like locating mortise holes in hidden regions. However, they noted that positional deviations between virtual and physical pieces could disrupt hand-eye coordination, emphasizing the need for precise calibration.

One of **AR**'s notable strengths lies in its ability to enhance long-term knowledge retention. By integrating contextual and interactive cues, **AR** supports users in retaining complex task sequences and procedures over extended periods [30, 31].

Gamified **AR** training methods further enhance effectiveness, increasing user motivation and engagement, which leads to faster and more accurate task completion [27]. Well-designed **AR** systems with intuitive interfaces and low cognitive overhead receive positive usability feedback [28].

3.2.1.3 Shared Benefits of **AR** and **VR**

Both **AR** and **VR** bring common advantages that prove to be efficient in assembly and training tasks. They provide immersive and interactive environments, offering a more engaging and effective learning experience [5, 6]. By providing real-time guidance and feedback, they reduce errors during the execution of tasks [6, 32]. Their adaptability allows them to adjust to various skill

levels, from beginners to experts, enabling wide applications across different settings [2]. Furthermore, **AR** and **VR** improve long-term retention by enabling hands-on practice and repeated interaction, consolidating task comprehension and procedural learning [6, 30]. All these common benefits make **AR** and **VR** a must-have in contemporary industrial training methods.

3.2.1.4 Key Takeaways

- **AR** and **VR** increase task accuracy and long-term knowledge retention, especially during complex or repetitive tasks [30, 32].
- **VR** is particularly effective for immersive experiences and sequence learning, with high engagement and skill retention comparable to traditional methods [6].
- Minimally guided and gamified approaches increase user motivation and cognitive engagement for both **AR** and **VR** [27, 29].
- Immersive **VR** environments increase task familiarity, particularly for intuitive learning of complex assembly tasks [6].

3.2.2 Negative Takeaways

3.2.2.1 Virtual Reality Challenges

Despite its advantages, **VR** training suffers from its dependency on visual assistance, which can lead to dependency and sluggish performance in real-world tasks [33]. A lack of realism in **VR**, particularly for tasks based on fine motor skills, limits its effectiveness in physical assemblies [6]. Usability issues, such as nausea and discomfort during long sessions, also detract from user satisfaction [26].

Wolfartsberger *et al.* [29] noted that overly guided **VR** applications decrease learning effects due to restricted cognitive engagement. Higher error rates and frequent hints in assisted **VR** conditions compared to unassisted and traditional methods indicate the need for balanced guidance.

Schwarz *et al.* [6] reported that **VR** systems often fail to provide realistic haptic feedback for tasks that involve fine motor skills, such as cable connections and precise placements. Features like auto-snap in **VR** environments can lead users to underestimate the physical effort required in real-world tasks.

3.2.2.2 Augmented Reality Limitations

Tracking and calibration errors, common in **AR** systems, result in positional misalignments that compromise their precision and accuracy [5].

Chu *et al.* [5] noted that in occluded setups, although **AR** visual cues aid in the identification of approximate locations, users often revert to tactile feedback to make precise adjustments caused by the perceived discrepancies in the virtual overlays. This reliance on haptic feedback over visual guidance highlights current limitations in **AR**'s calibration and tracking technologies.

Büttner *et al.* [31] emphasized the limited efficiency of projection-based AR in training applications. Their study showed that although projection-based AR efficiently avoids systematic mislearning, it does not match the speed and accuracy provided by personal training. Moreover, the use of projection-based AR did not lead to better long-term recall compared to traditional paper-based textbooks. They also reported an increase in cognitive load, specifically because of information clutter on surfaces, and errors related to the occlusion of critical task details. Users encountered difficulties in differentiating expected signals from actual objects in complex scenarios, leading to slower task completion and frequent errors in identifying assembly components.

3.2.2.3 Shared Constraints of AR and VR

Both AR and VR share certain limitations, including poor retention of highly complex tasks and prolonged training times. High development costs and long preparation times for AR/VR guidelines also hinder wider application.

Both AR and VR face shared challenges, including poor retention rates for highly complex tasks and extended training periods [2]. The high cost and extensive preparation time required to develop AR/VR instructions pose significant barriers to broader adoption [34].

The study by Werrlich *et al.* [28] also highlighted physical discomfort associated with prolonged use of HMDs. While participants liked the interface, the HMD's weight caused fatigue, emphasizing the need for lighter, ergonomic designs.

3.2.2.4 Key Takeaways

- VR often fails to provide realistic haptic feedback and may foster dependency on virtual assistance, reducing real-world task efficiency. [6, 29].
- Calibration and tracking accuracy issues in AR systems can impair performance in fine motor tasks. [5].
- Projection-based AR leads to heightened cognitive load and slower task execution in complex scenarios. [31].
- Usability issues, such as hardware discomfort, limited generalizability, and high development costs, affect both AR and VR. [34].
- Auto-snap features and reduced physical effort in VR may misalign virtual and real-world task requirements [6].

3.2.3 Critical Discussion

The findings in this chapter identify important directions for the development of operator guidance methodologies in virtual and augmented reality for assembly tasks. Although AR and VR have shown their potential in increasing productivity, precision, and worker engagement, a few critical challenges have to be overcome in this development.

Head-mounted displays are one of the most prominent choices for extended reality (XR) systems due to their ability to immerse users in experiences while also reducing the cognitive load [28]. HMDs are well-suited for activities requiring focus and sequential learning, placing them at a strategic position for upcoming XR instruction models.

AR, on the other hand, provides real-time, contextual guidance—a quality of extreme value in operator assistance in complex tasks with occluded components [5]. However, it suffers from calibration and tracking issues, which lower its precision. Those technical barriers have to be overcome in order for AR to ensure reliability during high-stakes assembly tasks.

Minimally guided and gamified approaches in AR and VR enhance user motivation and cognitive engagement [27, 29]. These strategies should be incorporated thoughtfully to balance user autonomy and system support, ensuring skill development without fostering dependency on assistive features.

Gaps, such as the lack of realistic haptic feedback in VR and reliance on tactile cues in AR, hinder the seamless transfer of the training outcome to real-world scenarios. Future systems will need to focus on improved haptic feedback for VR and precise overlay stability for AR, to underpin fine motor operations [31].

The high development costs and long times of preparation are just two of the economic and practical constraints that are huge barriers to the adoption of AR/VR technologies [34]. Developing modular, adaptable solutions—using cost-reduction approaches such as reusable components—will be important, possibly through open-source platforms for wide implementation.

In conclusion, advancing AR and VR guidance methodologies requires a focus on user-centric design, technical reliability, and scalable implementation. By prioritizing HMD-based systems, resolving AR's tracking limitations, and integrating motivational elements like gamification, the assembly industry can achieve meaningful improvements in operator guidance and assembly processes.

Chapter 4

Framework Design

To address the challenges commonly encountered in **VR/AR** assembly training, this dissertation proposes a practical and scalable framework aimed at improving realism and knowledge retention. **VR** was selected as the training medium due to its risk-free environment and lower implementation overhead. Although **AR** enables guided interaction with physical components, it requires either access to a real assembly line, or a physical mock-up of one, which would significantly increase cost and complexity. Despite offering less realism, **VR** provides a safer and more flexible platform for training purposes.

This chapter presents the methodological and technical foundation for the Virtual Reality training system developed. While a specific implementation will be described in the next chapter, this section outlines the general framework, designed for modularity and extensibility across various assembly scenarios. It encompasses system architecture, core technologies, design principles, interaction patterns, and data handling, with a particular emphasis on usability and Industry 5.0-aligned human-centric design.

4.1 Technology Stack

The **VR** training system was developed using the Unity game engine (version 6000.0.24f1), selected for its robust **VR** development capabilities, extensive asset ecosystem, and flexibility in scripting and rendering. The project uses Unity's Universal Render Pipeline (**URP**), which offers a good balance between graphical quality and performance, and is recommended by Meta for **VR** projects [35].

To streamline development and ensure compatibility with Meta Quest headsets, the system integrated the Meta All-in-One **SDK** [36]. This **SDK** automates the Unity project setup for **VR**, handling essential configurations with the click of a button.

A key feature of the **SDK** is its modular *Building Blocks*, which provide prebuilt components for common **VR** tasks including hand tracking, gesture detection, movement, and object manipulation [37]. The system uses *Building Blocks* for **XR** camera configuration, hand tracking, and

gesture-based control, features that are essential in training scenarios where natural, responsive input enhances the user experience.

In addition, XR Plugin Management is used to handle multiple XR plugins in a unified way, with the OpenXR plugin serving as the primary runtime interface. OpenXR ensures cross-platform support, allowing the application to be adapted to other XR devices with minimal code changes, while still maintaining full functionality on Meta headsets.

4.2 Architecture Overview

The VR training system is structured around a set of modular, loosely coupled components that communicate through an event-driven architecture. Each component is designed to fulfill a specific function within the training workflow, allowing for a clear separation of responsibilities and easier maintenance and future expansion.

4.2.1 Components

The system is composed of nine main components: the *EventManager*, which facilitates communication between components; the *TaskManager*, which creates assembly tasks; the *AssemblyManager*, which provides configuration and control over the assembly process; the *AssemblySlot*, which defines placement rules and prerequisites; the *AssemblyPiece*, which represents interactive objects; the *AssemblyHint*, which provides visual guidance for slots and pieces; the *TestController*, which records performance data; the *AssemblyProgressInterface*, which shows real-time task progress; and the *PalmMenu*, which offers user controls.

Figure 4.1 shows a component diagram representing the VR training system.

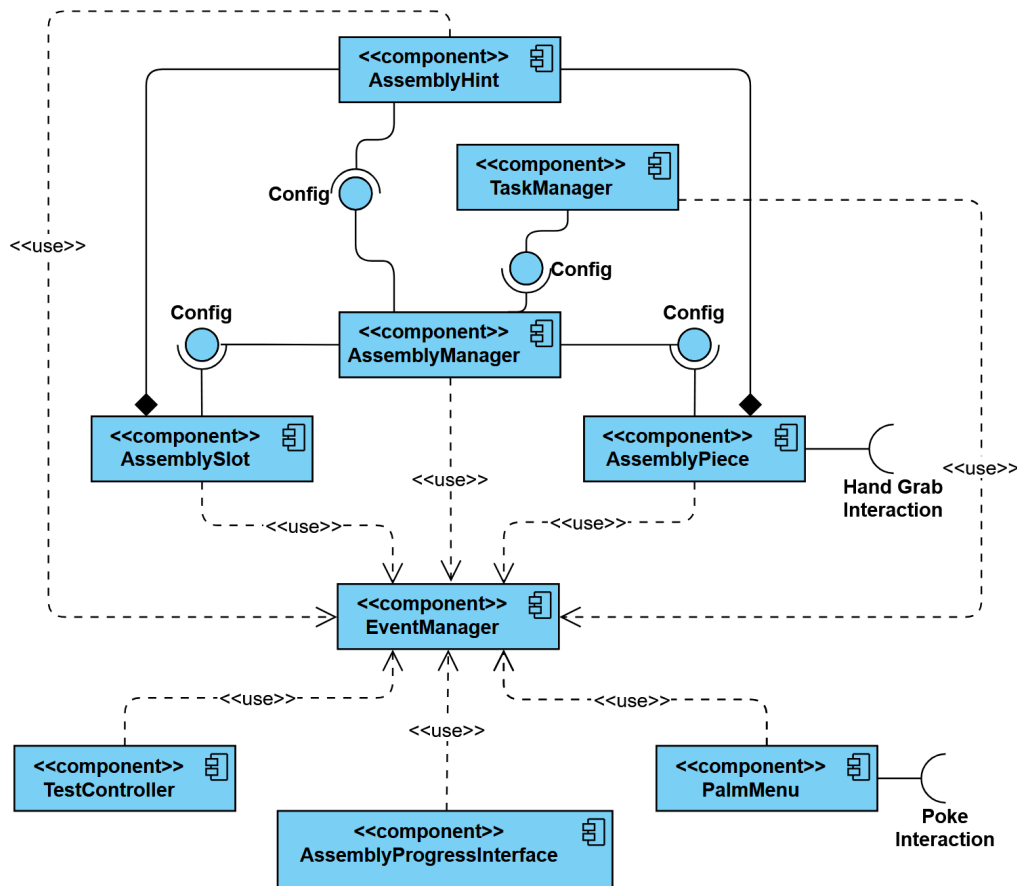


Figure 4.1: Component Diagram

The image shows that all components rely on the *EventManager*, which acts as the central communication hub. The *TaskManager* configures the *AssemblyManager* according to the requirements of each unique task. The *AssemblyManager*, in turn, supplies necessary parameters to the *AssemblySlot*, *AssemblyPiece* and *AssemblyHint* components. Interaction with the *AssemblyPiece* and *PalmMenu* is driven by user input, utilizing Meta's hand gesture system, specifically, the *Hand Grab Interaction* for selecting pieces and the *Poke Interaction* for navigating the menu.

4.2.1.1 Event Manager

The *Event Manager* implements a centralized system used for decoupled communication between components. It follows a publish-subscribe pattern, where components can emit events without needing to reference listeners directly, thus minimizing interdependencies and enhancing modularity. It follows a singleton design pattern, ensuring that only one instance exists throughout the entire application lifecycle.

The framework makes use of the following events:

- **onTaskStarted:** Triggered when a new assembly task begins.
- **onTaskEnded:** Triggered when the current assembly task ends.

- **onPiecePlaced(*AssemblyPiece*, *bool*)**: Fired when a piece is placed. The boolean indicates whether the placement was correct.
- **onPieceRemoved(*AssemblyPiece*, *bool*)**: Fired when a piece is removed. The boolean indicates whether it had been correctly placed.
- **onSlotFilled(*AssemblySlot*)**: Triggered when a slot becomes filled with the correct piece.
- **onProgressChanged(*float*)**: Broadcasts the current progress as a float between 0 and 1.
- **onMistakeMade**: Fired when the user places a piece incorrectly or removes a correct one.
- **onHintGiven**: Triggered whenever a visual hint is displayed to the user.
- **onAssemblyCompleted**: Fired when the entire assembly is successfully completed.

Each event has an associated trigger method that shares the same parameters as the event itself, as the trigger needs to provide the required data when invoking the event. The naming convention used distinguishes the two clearly: events are named with an *on* prefix (e.g., *onTaskStarted*), while their trigger methods use the event name without the prefix (e.g., *TaskStarted*). This pattern helps separate the event declaration from the action that fires it, improving their readability.

4.2.1.2 TaskManager

The *TaskManager* orchestrates the sequential execution of the assembly tasks, each represented by a *TaskConfig* that specifies the task name and whether automatic hints should be enabled. It also controls the transition between tasks based on assembly completion events. It starts the first task on initialization and listens for the assembly completion events to detect task completion.

When a task starts, it sends the assembly configuration to the *AssemblyManager* and triggers a *TaskStarted* event. Upon assembly completion, it triggers a *TaskEnded* event, waits for a short delay, resets the assembly, and then begins the next task if available.

4.2.1.3 Assembly Manager

The *AssemblyManager* oversees the entire assembly process. It tracks all assembly slots and pieces, manages hints, and monitors progress. It initializes slot and piece data at startup, listens for events such as piece placement and piece removal, and calculates the assembly progress. When all required slots are correctly filled, it triggers the *AssemblyCompleted* event. It also manages the hint system, automatically giving hints if configured to do so, and determines the next eligible hint based on the assembly order and dependencies.

Additionally, it handles resetting the assembly state between tasks and ensures dependency logic is respected during hint generation.

4.2.1.4 Assembly Piece

Assembly pieces are the primary interactable objects in the system. Users can pick them up using virtual hands, tracked through hand tracking technology. Each piece tracks whether it is currently placed, grabbed, or idle. Its physics behavior is dynamically adjusted: disabled during manipulation or placement, and re-enabled when released or removed. When the piece is released, it checks for the nearest slot within a threshold distance defined in the *AssemblyManager*. If a suitable slot is detected and can accept the piece, it is placed and snapped into position. When a placed piece is grabbed and moved, when released, the distance to the slot will be checked and the piece will be removed if it's farther than the threshold. Both piece placements and removals trigger their respective events.

4.2.1.5 Assembly Slot

Each training task is made up of multiple assembly slots that are tracked by the *AssemblyManager*. Each slot must be assigned a required *AssemblyPiece*. To create an assembly task, the slots must be placed manually, putting together the whole assembly. For this purpose, slots are visible in the editor, but are not rendered during training.

In addition to the spatial placement of slots, activation rules can also be defined through the slot prerequisites. A piece cannot be placed until the slot is activated. Prerequisites are essential in the assembly system to replicate the logical dependencies found in real-world tasks. They ensure that certain components cannot be placed before others, enforcing a certain task sequence. When a slot has no prerequisites, it is activated by default. Each slot may depend on one or more other slots, and its activation is governed by distinct activation modes:

- **AND:** All prerequisite slots must be filled.
- **OR:** At least one prerequisite slot must be filled.
- **ONE:** Exactly one prerequisite slot must be filled.
- **NOTALL:** Some but not all prerequisite slots must be filled.

These rules allow for sophisticated assembly workflows where parts are interdependent, enforcing a loose order, where the prerequisites must be met, but their order of activation isn't relevant.

4.2.1.6 Assembly Hint

The hint system provides dynamic visual hints to guide users in locating the next slot and corresponding piece. This includes:

- Enabling the render of a target slot.
- Highlighting compatible, unplaced pieces.

- A pulsating effect to attract attention.

Hints are displayed only when enabled and only for currently valid placements. Once a piece is correctly placed, hints reset and new ones are generated as necessary. Automatic hints can be enabled and disabled in the *AssemblyManager*.

Hint Order:

Hints must be provided in a specific sequence to ensure smooth progression through the assembly process. Because some slots rely on complex prerequisite conditions, choosing which available slot to fill at random can lead to a deadlock, where several slots remain unfilled, yet none are active due to their conditions not being met.

For example, consider a slot with a NOTALL activation mode. This slot becomes active only if not all of its prerequisite slots are filled. If hints lead the user to fill all the prerequisite slots first, the NOTALL slot will no longer be valid for activation—it will remain inactive. To prevent this, the hint system must account for the logic of each activation condition and prioritize hinting slots in an order that preserves the possibility of activating all remaining slots. This ensures that users are always able to proceed without getting stuck due to prematurely filled dependencies.

To ensure a valid sequence of slot activations, the assembly manager generates an initial list of all slots and organizes them in an order that respects both hierarchical and prerequisite constraints. To achieve this, each *AssemblySlot* keeps track of its own dependents. Before attempting to hint at a slot, a check is conducted to ensure that filling this slot will not lock any of its dependents, if it does, another one is tried. This ensures that the hinted sequence allows every slot to eventually become active and filled.

Even with the validity of the sequence ensured, the hints should follow an order that is easy to memorize and logical from a human perspective. For example, consider an assembly with three separate slot groups, each of them with elements that are spatially close. The easier and most natural way to fill them would be to complete each group before moving on to the next one. To achieve this, the default order is based on the scene object hierarchy, from top to bottom, depth-first.

To illustrate the hint ordering logic, consider the following example hierarchy used in a generic example:

```
RootObject
  GroupA
    SlotA1
    SlotA2
  GroupB
    SubgroupB1
      SlotB1a
      SlotB1b
    SlotB2
  GroupC
```

SlotC1

In this example, groups and subgroups are empty containers used to organize related slots. The hint system uses this hierarchy to determine a default depth-first traversal order: *SlotA1*, *SlotA2*, *SlotB1a*, *SlotB1b*, *SlotB2*, and finally *SlotC1*.

This ordering reflects a natural progression, prioritizing localized groups of components before moving on. Even if it's technically possible to fill *SlotC1* early, the system encourages completing entire subgroups before advancing, making the process more intuitive for users.

If dependencies exist (e.g., *SlotB1a* depends on *SlotB1b*), then the system adjusts the hint sequence dynamically. It will only suggest *SlotB1a* after *SlotB1b* is validly filled. This ensures that hints always guide the user in a valid order, even when the hierarchy alone doesn't define it.

4.2.1.7 Palm Menu

The palm menu is a simple user interface attached to the trainee's non-dominant hand (as configured in the headset settings). It can be toggled by pinching the thumb and index fingers together, and it automatically orients itself perpendicularly to the user's palm to maintain readability. The menu is paginated, with each page containing a single button, and allows the user to scroll horizontally by swiping their finger. To press a button, the user must poke it with their dominant index finger.

The menu includes two buttons: a **Hint Button**, which requests a single manual hint, and a **Reset Button**, which resets the current assembly task to its initial state, clearing all placed components. The menu is illustrated in Figure 4.2.

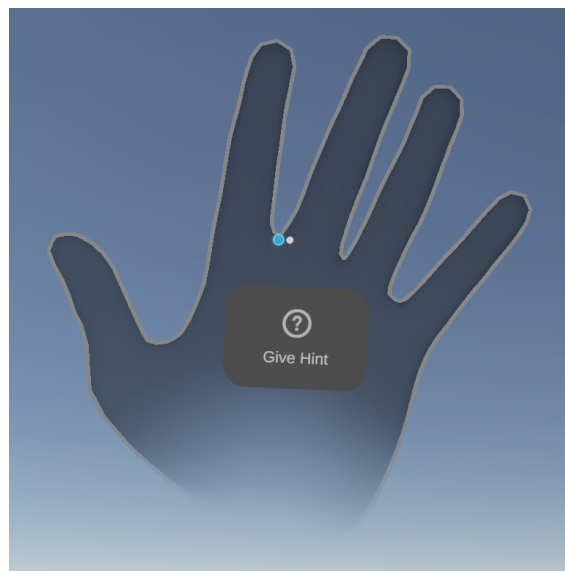


Figure 4.2: Palm Menu

4.2.1.8 Test Controller

The *TestController* is responsible for monitoring user performance throughout each training task and recording relevant metrics for later analysis. It works independently of the assembly logic and is activated automatically when the application starts. This system collects only quantitative data.

Each session begins by assigning a **User ID**, which is used to create a uniquely named folder on disk. It does not allow data from a previous test to be overwritten: if results exist for that id, it will stop the program.

When the test begins, the system listens to various assembly-related events such as piece placement, removal, mistakes, and hint usage. These values are tracked and changed as the user interacts with the assembly.

Upon task completion, all task results are serialized into a **CSV** file within the user-specific folder.

Each entry logs the following data:

- **userID**: Unique identifier for the participant.
- **taskID**: Index of the task within the session.
- **time**: Duration of the task in seconds.
- **pieceCorrectPlacements**: Number of times a piece was correctly placed in the right location.
- **pieceIncorrectPlacements**: Number of times a piece was placed in the wrong location.
- **pieceCorrectRemovals**: Number of times an incorrectly placed piece was removed.
- **pieceIncorrectRemovals**: Number of times a correctly placed piece was mistakenly removed.
- **nHints**: Number of manual hints requested during the task.

4.2.1.9 Assembly Progress Interface

The interface, shown in Figure 4.3, contains a bar indicating the current progress. As the user moves, the interface rotates to face the them regardless of their position. It listens to the *onProgressChanged* event to update its progress.

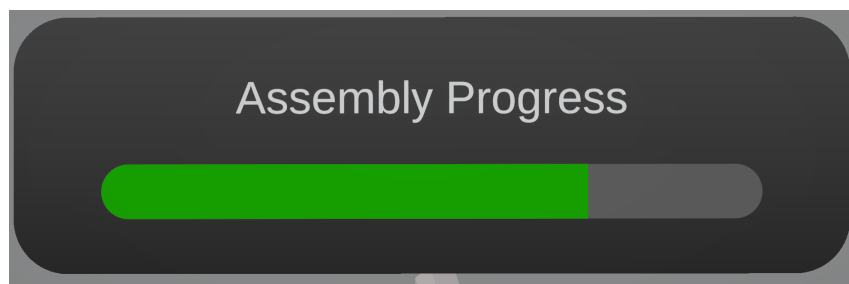


Figure 4.3: Assembly Progress Interface

4.3 Design Principles

The design of the **VR** training framework adheres to a set of core principles that guided the architecture and interaction design choices, enabling the system to scale beyond the prototype implementation and remain adaptable to various assembly training scenarios.

4.3.1 Modularity, Scalability, Extensibility and Maintainability

The system is structured around loosely coupled, self-contained modules with well-defined responsibilities. While the goal is to keep components as decoupled and independent as possible, some—such as assembly slots, pieces, and hints—require coordination through a central component like the *AssemblyManager*. The architecture tried to balance modular autonomy and centralized managers to coordinate other modules when necessary.

This simplifies scalability by allowing new assembly tasks to be created as a combination of existing modules, scaling from simple object placements to complex multi-step procedures with conditional dependencies.

Extensibility also benefits from the clear boundaries between components, making it straightforward to introduce new ones without impacting unrelated parts. This encourages iterative development and innovation.

Finally, maintainability is also enhanced by the modular design, as debugging and testing can focus on individual components in isolation. Clear responsibilities reduce unintended side effects and simplify future code revisions.

4.3.2 Interoperability and Standardization

To maximize hardware compatibility, the system uses the OpenXR runtime, which ensures broad **XR** platform support instead of working only on Meta devices. This choice aligns with modern software development practices that favor open standards over proprietary solutions. The event system is designed to be decoupled and generic, supporting future interfacing with external tools and systems.

4.3.3 Security and Data Integrity

The system's file-based data collection mechanism enforces write protections to avoid overwriting user data, with each session being uniquely identified and stored separately. However, with the prototype being offline-only, security measures related to network communication or external system interactions are not currently applicable, but would become essential in future versions that involve connectivity with external tools and systems, beyond the local environment.

4.3.4 Human-Centered Design

Central to the framework is a human-centered approach, consistent with Industry 5.0 values, which emphasize the synergy between advanced technology and human workers. Usability is a core principle of the system's design, and is prioritized throughout all aspects of user interaction.

The interface elements, including the palm menu and visual hint system, are designed to be discoverable and non-intrusive. Interactions are natural, relying on intuitive hand gestures like poking, grabbing, pinching and swiping. This decision to rely on hand tracking instead of hand-held controllers was made to allow users to interact directly with virtual objects in a way that closely mimics real-world manipulation.

The hint sequencing algorithm guides users through assembly steps in a natural, intuitive progression that reflects common human problem-solving strategies. It prioritizes hints in an order that feels logical and easy to follow, helping users build understanding incrementally as they would in a real-world assembly. This design supports a smooth learning experience by aligning with how people naturally break down and approach tasks in the real world.

4.4 System Behavior

This section illustrates how the system functions in practice by presenting key interaction flows using sequence diagrams. These diagrams capture the dynamic behavior of the system's components during user interactions, clarifying how different modules interact with each other. The goal is to provide a clear understanding of runtime operations and the relationships between core components.

4.4.1 Piece Placement and Removal

Figure 4.4 illustrates the runtime interactions involved in placing and removing an assembly piece. The sequence diagram models the behavior across four participants: the *User*, *AssemblyPiece*, *AssemblySlot*, and the *EventManager*.

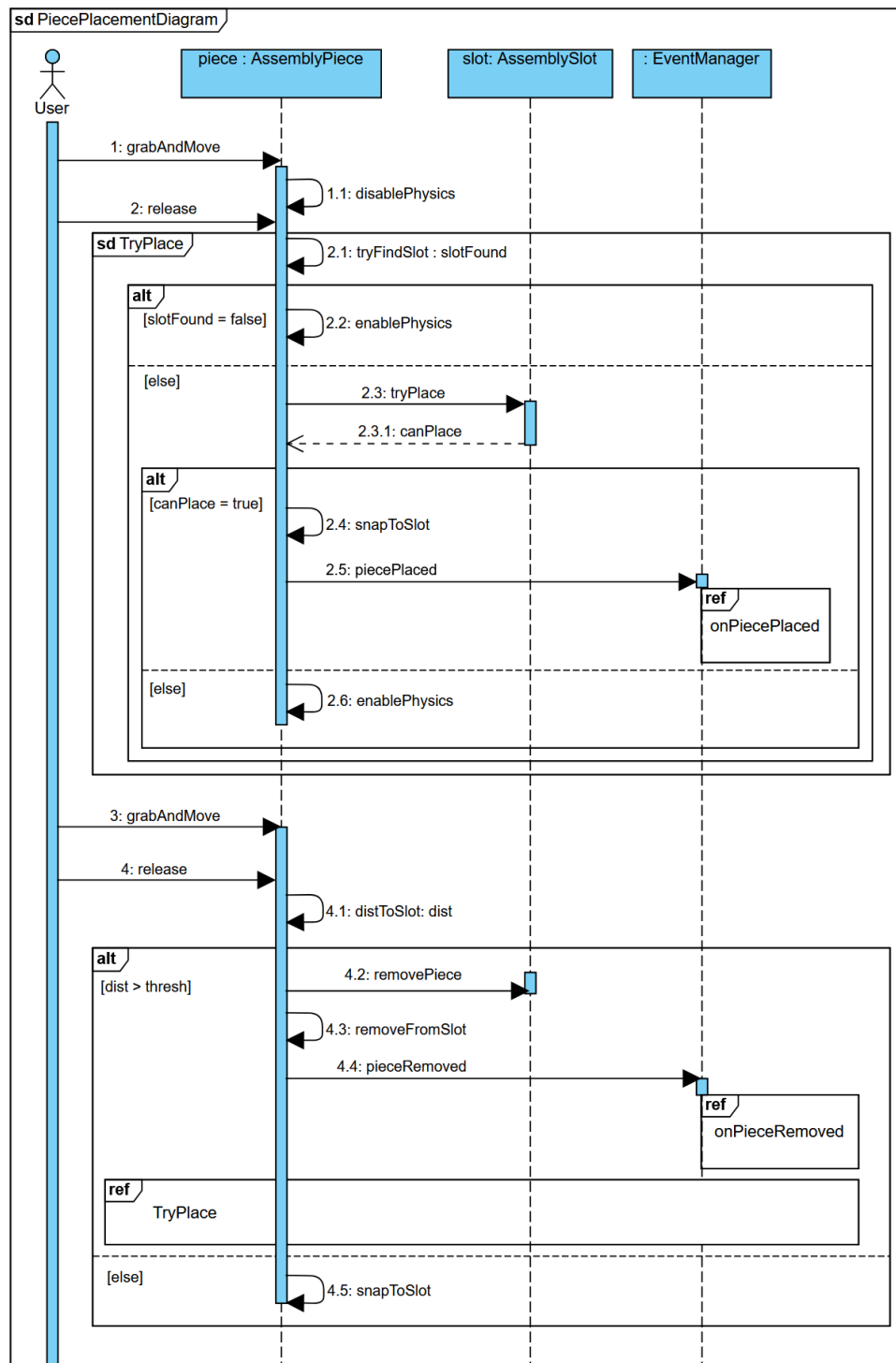


Figure 4.4: Piece placement and removal sequence diagram

1. Piece Placement

The sequence begins with the user performing a *grabAndMove* action on an assembly piece. Upon releasing the piece, the following events occur:

1. The piece disables its physics to avoid collisions with other objects while being moved.
2. It attempts to find a valid slot via *tryFindSlot*.
3. If no valid slot is found, physics are re-enabled, and the interaction ends.
4. If a valid slot is found:
 - The piece calls *tryPlace* on the slot.
 - The slot evaluates whether placement is valid and returns a response via *canPlace*.
 - If placement is allowed, the piece snaps into the slot using *snapToSlot*, and the event *piecePlaced* is triggered and sent to the *EventManager* (see **onPiecePlaced**).
 - Otherwise, if placement is denied, the piece re-enables physics to resume its natural behavior.

2. Piece Removal

Later, the user grabs and releases the piece again. This triggers a distance check:

1. The piece calls *distToSlot* to evaluate how far it is from the slot.
2. If the distance exceeds a defined threshold:
 - The piece initiates *removePiece* on the slot, and *removeFromSlot* on itself.
 - Once removed, a *pieceRemoved* event is sent to the *EventManager* (see **onPieceRemoved**).
 - The system then reuses the earlier logic in *TryPlace* to attempt a new placement at the current location.
3. If the piece remains within the valid range, it simply re-snaps into the slot using *snapToSlot*.

4.4.2 On Piece Placed/Removed

Figure 4.5 illustrates the runtime interactions involved when the *onPiecePlaced* event is triggered. The sequence diagram models the behavior across five participants: the *AssemblyPiece*, *AssemblyManager*, *AssemblyProgressUI*, *TestController*, and the *EventManager*. The process for the *onPieceRemoved* event follows the same process.

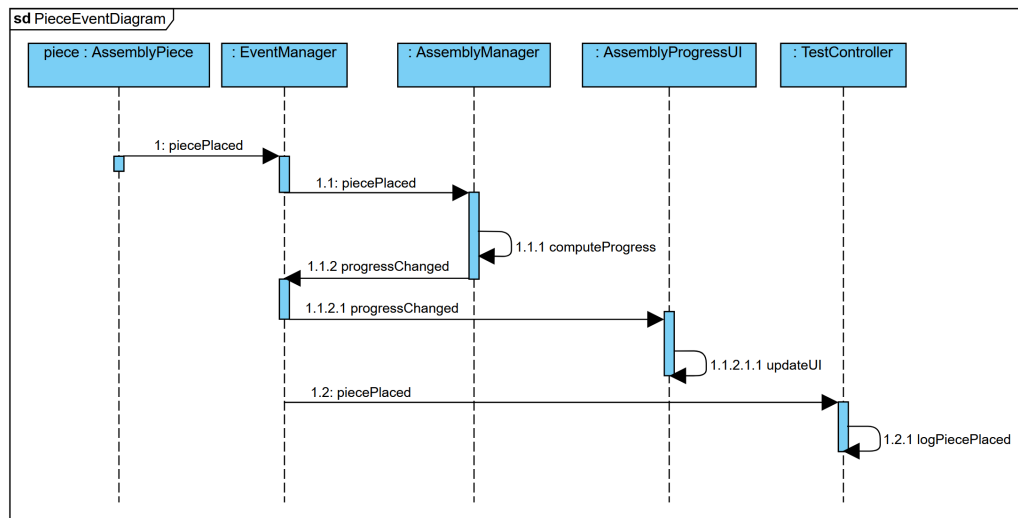


Figure 4.5: onPiecePlaced sequence diagram

The sequence begins when a piece is placed, initiating the *piecePlaced* event. Upon this action, the following events occur:

1. The *piecePlaced* event is sent to the *EventManager*.
2. The *EventManager* propagates the *piecePlaced* event to the *AssemblyManager*.
3. The *AssemblyManager* then performs the following actions:
 - It computes the progress of the assembly via *computeProgress*.
 - It triggers a *progressChanged* event and sends it back to the *EventManager*.
4. The *EventManager* receives the *progressChanged* event and forwards it to the *AssemblyProgress UI*.
5. The *AssemblyProgressUI* updates its user interface by calling *updateUI*, showing the new progress.
6. Concurrently with the *progressChanged* event propagation, the *EventManager* also sends the initial *piecePlaced* event to the *TestController*.
7. The *TestController* logs the piece placement.

4.5 Current Limitations

While the proposed **VR** training framework offers a solid starting point for assembly training, it remains a prototype with several important limitations that need to be addressed.

Limited Physical Realism

Despite VR's ability to simulate complex environments and interactions, the current framework lacks physical haptic feedback, which is crucial for realistic assembly training. Users can see and manipulate virtual objects but do not experience the tactile sensations, resistance, or weight differences of real components.

Visual and Cognitive Load Considerations

Although the hint system is designed to help users progress smoothly, it currently lacks the ability to adapt. Without adaptive guidance, beginners might receive too many hints, which can discourage active problem solving, while more experienced users may feel held back by the fixed pace.

Lack of Integration with Industrial Systems

Unlike solutions that connect directly with external industrial systems and workflows, the VR framework functions entirely as a standalone virtual environment. It cannot integrate with real-time data, machinery, or production processes.

Assembly Creation Overhead

While the modular design supports extensibility, creating new assembly tasks requires manual setup of slots, pieces, and prerequisite logic, which can be time-consuming and demands technical expertise. The framework does not support automated generation that could streamline assembly creation.

Chapter 5

Assembly Scenario Design and Implementation

This chapter presents the implementation of the virtual reality (VR) assembly training solution for the specific case study based on the Human-Robot Collaboration (HRC) model set. The focus is on the assembly of a custom furniture set composed of a desk and a chair, illustrating the practical realization of the framework introduced previously.

5.1 HRC Model Set Overview

The HRC model set [38] provides modular components designed to replicate typical flat-pack furniture assembly tasks. Its well-defined parts and connectors facilitate creating realistic and versatile scenarios suitable for training and research purposes. For this dissertation, the model set was selected due to its compatibility with both physical prototyping and VR simulation, enabling an integrated training approach.

5.2 Case Study: Custom Desk and Chair Assembly

The specific assembly task implemented in this study involves a custom-designed desk and chair constructed by combining various HRC model set pieces. This setup represents a moderately complex task, requiring users to manipulate a mix of 3D printed parts, wooden panels, and dowels. The design reflects common real-world assembly challenges such as multi-material handling and sequential part fitting, making it an appropriate scenario to validate the VR training solution.

5.2.1 Physical Prototype Implementation

The physical prototype was built using a combination of 3D printed components and manually cut wooden elements (dowels and panels). Figure 5.1 depicts the completed physical assembly.



Figure 5.1: Completed physical desk and chair assembly

5.2.1.1 3D Printing

All 3D printed parts were produced with a Creality Ender3 S1 Pro printer, using the following settings:

- **Material:** PLA
- **Nozzle Temperature:** 215°C (first layer), 200°C (subsequent layers)
- **Bed Temperature:** 70°C (first layer), 60°C (subsequent layers)
- **Infill:** 20%
- **Supports:** Tree type, enabled for complex parts
- **Adhesion:** Adhesion layer used

The printing process was carried out in batches of varying sizes. The more complex pieces were first printed individually to ensure that the chosen print settings and supports didn't cause any issues. This approach helped prevent failed prints that could ruin an entire batch.

The 3D models for printing were obtained from the public **HRC** Model Set repository by ScazLab [38].

Below are images of every 3D-printed component:

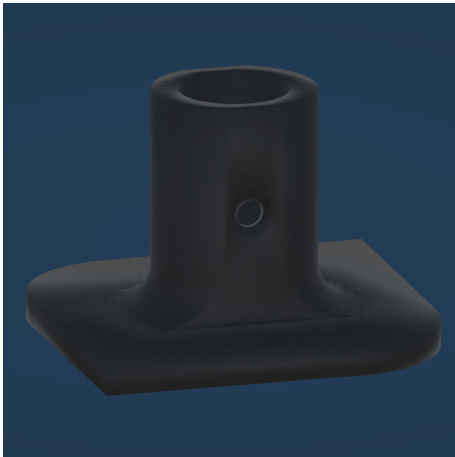


Figure 5.2: 3D Printed Foot



Figure 5.3: Top90 Connector



Figure 5.4: Shelf90 Bracket



Figure 5.5: Shelf180 Bracket

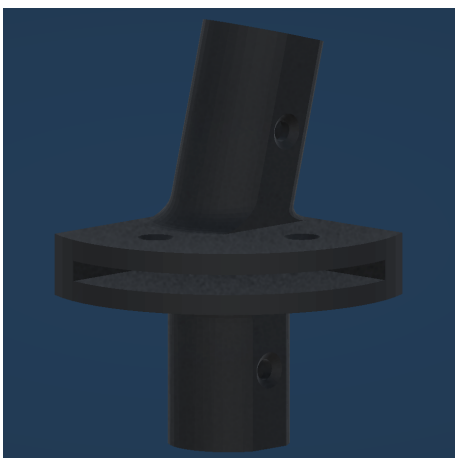


Figure 5.6: Chair Left Side



Figure 5.7: Chair Right Side



Figure 5.8: Chair Back Support

5.2.1.2 Wooden Components

The wooden panels were manually cut from 3 mm thick plywood sheets, while the dowels, 1 cm in diameter, were cut into 15 cm long segments. All cutting was performed using a handsaw; as a result, the dimensions were not perfectly precise, but they were sufficiently accurate to allow for a functional assembly.

5.2.1.3 List of components

The assembly consisted of a total of 48 pieces: 26 were 3D-printed components, 5 were wooden panels, and 17 were dowels. Table 5.1 shows the detailed list of all components.

Material	Part Name	Quantity	Dimensions
PLA (3D Printed)	Foot	10	-
	Top90	8	-
	Shelf90	2	-
	Shelf180	2	-
	ChairLeft	1	-
	ChairRight	1	-
	ChairBack	2	-
Plywood (3 mm thickness)	Small Panel	3	$\approx 15.25 \text{ cm} \times 20.30 \text{ cm}$
	Long Panel	1	$\approx 15.25 \text{ cm} \times 40.65 \text{ cm}$
	Chair Back Panel	1	$\approx 6.35 \text{ cm} \times 17.15 \text{ cm}$
Wooden Dowels	Dowels	17	15 cm length, 1 cm diameter

Table 5.1: Parts and materials for the assembly task

5.3 VR Training Framework Implementation

The implementation of the VR training system involved preparing 3D models for Unity, assigning the appropriate components to each part, and configuring the assembly slots into the final structure. In the end, the result was the assembly as pictured in Figure 5.9

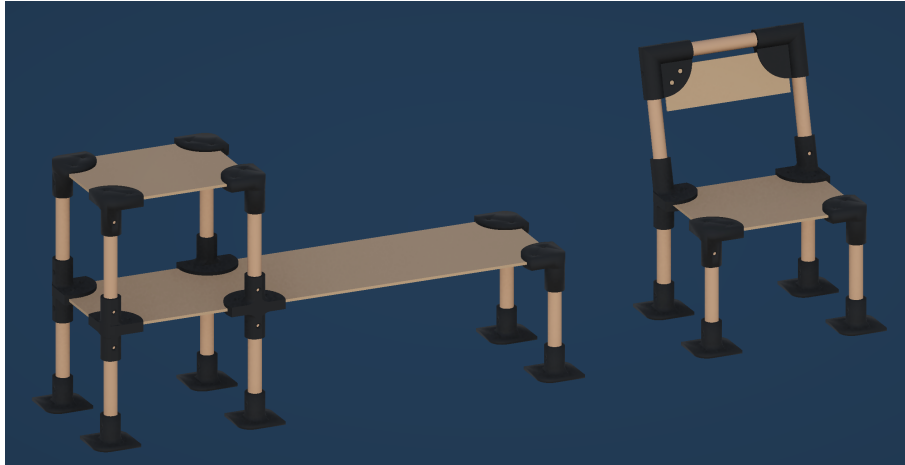


Figure 5.9: Completed chair and desk assembly (Virtual)

5.3.1 3D Model Preparation for Unity

The 3D models for the HRC Model Set pieces from ScazLab’s repository [38] were in the Standard Tessellation Language (STL) format, which Unity does not natively support. Additionally, the raw models often contained improper scaling, orientation, or anchor positions. Therefore, a preprocessing step was conducted. Each model was first imported into Blender, where its dimensions were adjusted to match real-world measurements. Orientations were corrected to align with Unity’s coordinate system, and anchor points were set to the model’s geometric center. The models were then exported in the Filmbox (FBX) format, which Unity uses.

Once imported into Unity, custom materials were created and assigned to the different parts to reflect their real-world appearance. A dark plastic material was used for PLA parts, and wooden materials were created to reflect the different colors of the panels and dowels.

In Unity, each piece and slot used in the assembly was turned into a prefab. For assembly pieces, the prefab consisted of a parent *GameObject* with the *AssemblyPiece* component, while the 3D model was added as a child. The same structure was applied to assembly slots: the parent held the *AssemblySlot* component, and the visual representation was placed underneath as a child object. This setup kept the interaction logic and visual elements separate, making the system more organized and easier to manage.

5.3.2 Assembly Creation

The assembly was structured as two independent units: the chair and the desk. Although participants were not required to follow a strict order during the task, the assembly was designed to

provide hints for the chair first before moving on to the desk.

On the other hand, a top-down assembly order was enforced using slot dependencies. For the chair, this required starting with the legs, then attaching the seat base, and finally the backrest. Figure 5.10 illustrates the assembly process in steps, from left to right.

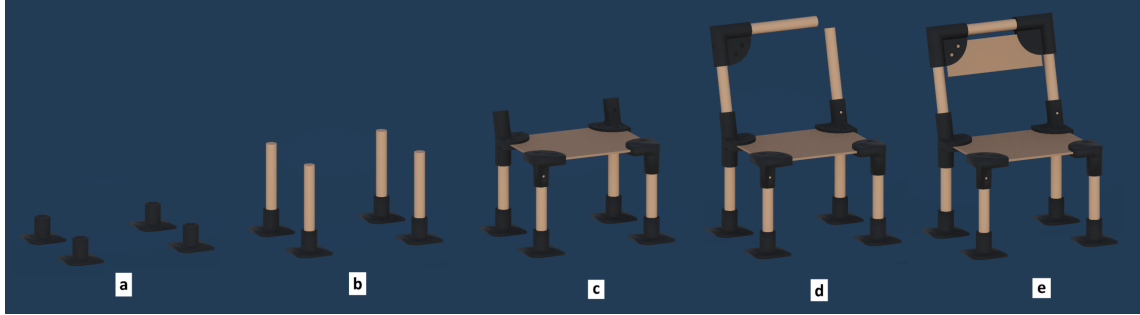


Figure 5.10: Chair Assembly Process

In step *c*, the seat panel uses a *OR* activation condition, depending on the four corners, with only one having to be placed beforehand. In step *d*, the topmost dowel must be inserted before both corner pieces are placed (*NOTALL* condition), otherwise, it would not physically fit.

The desk assembly, illustrated step by step in Figure 5.11 began with the legs, followed by the lower platform, the vertical supports for the upper section, and finally the top shelf.

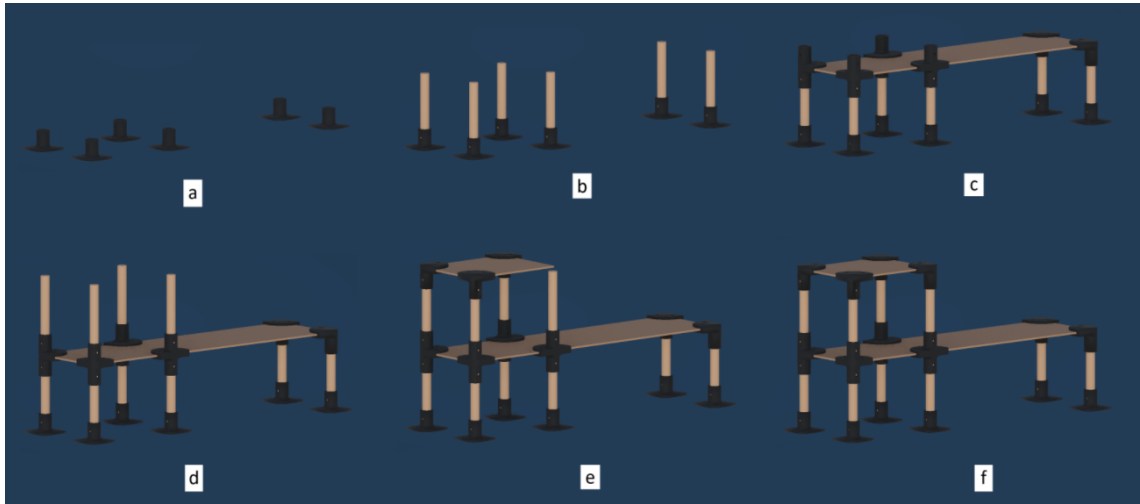


Figure 5.11: Desk Assembly Process

In step *c*, a support piece must be placed before attaching the panels (*OR* condition). In step *e*, at least one of the corner slots must remain unoccupied to allow the corresponding panel to fit properly, enforcing another *NOTALL* condition.

Chapter 6

Study Design

This chapter presents the experimental study designed to evaluate the effectiveness of virtual reality (VR) training versus traditional instruction for a physical assembly task. The study compares VR-based training to conventional training.

6.1 Participants & Training Groups

Eighteen participants took part in the study, with nine assigned to each training group. Most participants were aged 22 or 23. Participants were randomly assigned to one of two training conditions:

- **VR Training Group:** The participants of this group underwent VR Training through the system described in the previous chapter.
- **Traditional Training Group:** The traditional training group was taught in person, with the supervisor performing the assembly and explaining each step. This served as an analogy for an assembly line where a learn-by-doing approach is not viable, either due to safety risks or the high cost of mistakes.

6.2 Experimental Procedure

Step 1: Pre-training Briefing

Upon arrival, all participants were welcomed and provided with an explanation of the purpose, design and procedure of the study. After signing an informed consent form (6.4.1), participants were introduced to the assembly task and shown the physical components they would be using, but they were not allowed to touch them. After the introduction, participants were asked to fill the Basic Profile questionnaire to gather demographic data (such as age, gender, education) and prior experience with VR.

Step 2: Training Phase

VR Group

The VR group trained using the virtual assembly system described in the previous section. The procedure was divided into two phases, with each phase having one task which would be repeated twice.

- The first phase had hints enabled, and its purpose was to familiarize the participant with the controls and the assembly process. The supervisor was allowed to give help, but only with respect to the controls.
- In the final phase, hints were disabled. The first task aimed to help trainees become familiar with assembling without the assistance of the hints, while the second encouraged them to complete the task as quickly as possible, but to put more emphasis on correctness rather than speed.

The performance metrics for this group were collected automatically by the testing system.

Traditional Training Group

Members of this group were shown and explained the assembly process once and were allowed to ask questions, before being asked to replicate it. They were not allowed to physically intervene or touch the components.

Step 3: Real-world Assembly Task

After completing their respective training sessions, participants were asked to perform the real-world assembly task. They were given a single chance to complete the assembly, as the goal was to compare the performance between both training groups. Giving the participant more chances would also include the knowledge they gained from the previous real-world attempts, which is outside the scope of the study.

Participants were instructed to complete the assembly as fast and efficiently as possible without sacrificing correctness. During the process, a supervisor collected the performance metrics manually. The supervisor was not allowed to help or give any hints, and the participants were not interrupted unless they performed actions that could damage the components.

Step 4: Post-task Feedback

Upon completing the assembly, participants were asked to complete a questionnaire evaluating their experience. Participants were also encouraged to provide open-ended comments about challenges faced, what they would have changed in the training, or suggestions for improvement.

6.3 Evaluation Metrics

6.3.1 Quantitative Metrics

As previously explained, the performance data during the VR training was gathered automatically by the testing system, and manually collected for the real-world assembly. The performance indicators used are the following:

- **Completion Time:** The total time (in seconds) each participant took to complete the task. Measured from the moment the task started until the final correct piece was placed and confirmed.
- **Correct Piece Placements:** The total number of correct piece placements.
- **Incorrect Piece Placements:** The number of incorrect piece placements.
- **Correct Piece Removals:** The number of correctly placed pieces that were later intentionally removed. This behavior may indicate uncertainty.
- **Incorrect Piece Removals:** The number of pieces that were removed after being placed incorrectly. In this specific study, the value is always equal to the *Incorrect Piece Placements*, since every piece placed incorrectly must be removed to place the correct one. The value was included in the VR training system since it could be useful in other assembly scenarios.
- **Number of Hints Used:** The number of hints given to the participant during the task. Only applies in the VR Group training, as there are no hints during the real-world task.

These metrics enable a comparison not only in terms of task completion speed but also interaction efficiency, mistake rates, and dependency on external support (hints).

6.3.2 Qualitative Metrics

Qualitative feedback was collected using post-task questionnaires, providing insights into the participants' subjective experience, including ease of use, confidence, clarity of the instructions, and overall usability.

For the VR Group, the Virtual Reality System Usability Questionnaire (VRSUQ) [39] was used. The questionnaire (6.4.3) evaluates usability based on three ISO-aligned dimensions: Effectiveness, Efficiency, and Satisfaction. Effectiveness includes compatibility, feedback, and error prevention. Efficiency addresses understandability, learnability, and error recovery. Satisfaction captures user enjoyment, as well as physical and emotional discomfort. All items are rated on a 5-point Likert scale (1 = Strongly Disagree to 5 = Strongly Agree, except for items 3, 8, and 9 that are reverse coded). A high VRSUQ score, which scales from 0 to 100, according to Equation 6.1, indicates high usability and positive user perception.

$$\text{VRSUQ Score} = \left(\frac{\sum_{i=1}^9 \text{Item}_i - 9}{36} \right) \times 100 \quad (6.1)$$

Participants in the traditional training group answered a shorter set of questions (6.4.4) focused on clarity of explanation, preparedness, and confidence. The participants of both groups also completed a final questionnaire (6.4.5) about the real-world task experience to evaluate perceived task difficulty and training effectiveness.

6.4 Questionnaires

6.4.1 Consent Form

Before participating in the study, all participants were given an informed consent form explaining the purpose, procedure, risks, and data confidentiality details of the study. The form emphasized voluntary participation and the right to withdraw at any time without penalty.

Assembly Training Form — Informed Consent

You are invited to participate in a research study exploring the effectiveness of virtual reality (VR) training compared to traditional slide-based instruction in a furniture assembly task. Your participation is entirely voluntary.

If you choose to participate, you will be assigned to one of two training conditions: a VR simulation or a traditional instructor. After this training, you will attempt to physically assemble a simple piece of furniture. During this process, we will record non-personal performance metrics such as task duration, number of mistakes, and any assistance used. You will also be asked to complete short questionnaires about your experience.

There are no significant risks associated with this study. However, participants using VR may experience mild discomfort such as dizziness or motion sickness. You may stop the session at any time for any reason without any negative consequences.

All information collected during the study will be anonymized and stored securely. Your personal identity will not be linked to your results, and the data will be used solely for academic research purposes. You have the right to withdraw your participation at any point without providing a reason.

By proceeding, you acknowledge that you have read and understood the information above and freely agree to participate in this study.

6.4.2 Basic Information Questionnaire

Question	Response Options
User ID	Numeric input
Age	Numeric input
Education Level	Secondary Education (12th grade) Professional Course Bachelor's Degree Master's Degree PhD Other (text input)
Current Field of Study/Profession	Open text
Experience with Virtual Reality	Likert scale: 1 (No experience) to 5 (Very experienced)
Experience with Furniture Assembly	Likert scale: 1 (No experience) to 5 (Very experienced)

6.4.3 Virtual Reality Training Experience Questionnaire

Likert-Scale Questions (1 = Strongly Disagree, 5 = Strongly Agree):

- 1 The system responded well to my manipulations.
- 2 I think the virtual reality system provides clear feedback on my manipulations.
- 3 I kept making errors/mistakes while using the virtual reality system.
- 4 I could clearly understand the information presented within the virtual environment.
- 5 I think this system is user-friendly, straightforward to learn, and designed in such a way that most people will find it easy to adapt to.
- 6 I think it is easy to correct errors made during virtual reality experiences.
- 7 I enjoyed the virtual reality experience.
- 8 I felt dizzy, motion sickness, or a headache while experiencing virtual reality.
- 9 While experiencing virtual reality, I felt mental burdens such as tension, frustration, and time pressure.
- 10 The instructions were clear and easy to follow.
- 11 The instructions helped me understand the assembly process.
- 12 I feel more confident doing the real task after training in VR.

Questions 1 to 9 are taken from the Virtual Reality System Usability Questionnaire (VR-SUQ) [39]. The last three are oriented towards the assembly experience in VR.

Open-Ended Questions:

Did you encounter any technical issues during VR training? If so, please describe them.
Other comments and improvement suggestions.

6.4.4 Traditional Training Experience

Likert-Scale Questions (1 = Strongly Disagree, 5 = Strongly Agree):

The instructions were clear and easy to follow.
The instructions helped me understand the assembly process.
I feel more confident doing the real task after the explanation.

Open-Ended Questions:

Was something missing or unclear in the instructions?
Any suggestions to improve the training experience?

6.4.5 Assembly Task Experience

Likert-Scale Questions (1 = Strongly Disagree, 5 = Strongly Agree):

The task was easy to complete.
I made fewer mistakes than I expected.
The training experience was useful to complete the assembly.

Open-Ended Question:

Did you encounter any difficulties/issues during the assembly task? If so, please explain them.

Chapter 7

Results and Discussion

This chapter presents the results from the experimental evaluation of VR-based and traditional training. Both quantitative and qualitative data are reported and critically analyzed. The results are organized into three main sections: performance during VR training, participant feedback on training experiences, and the outcomes of the real-world assembly task.

7.1 Performance During Virtual Reality Training

This section presents the quantitative results collected during the VR training sessions, focusing on task completion time and errors. The raw data for the VR training performance can be found in Appendix A.1.

7.1.1 Task Completion Time

Figure 7.1 shows the evolution of the average task completion time across the four VR training tasks, numbered from 0 to 3.

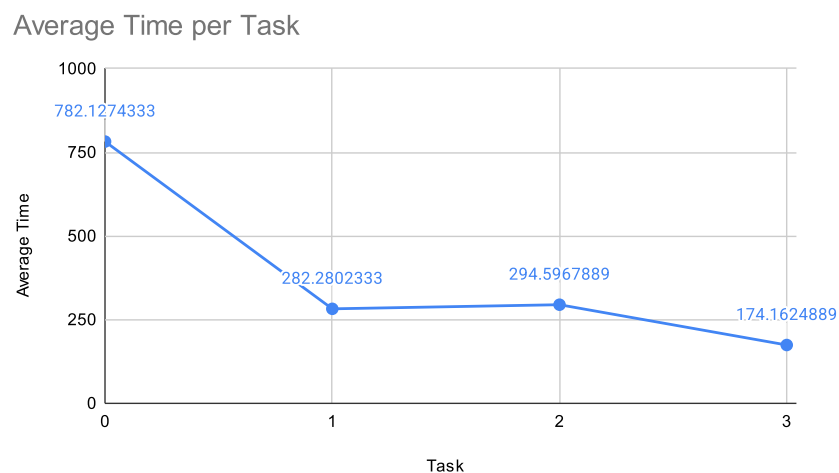


Figure 7.1: Evolution of the average time per task

Task 0 exhibited the longest average time. Task 1 showed a substantial decrease in average time compared to Task 0. Task 2 showed a slight increase in average time relative to Task 1. Task 3 had the lowest average time among all tasks.

7.1.2 Number of errors

Figure 7.2 shows the number of mistakes made per task, and follows a similar trend to the completion time.

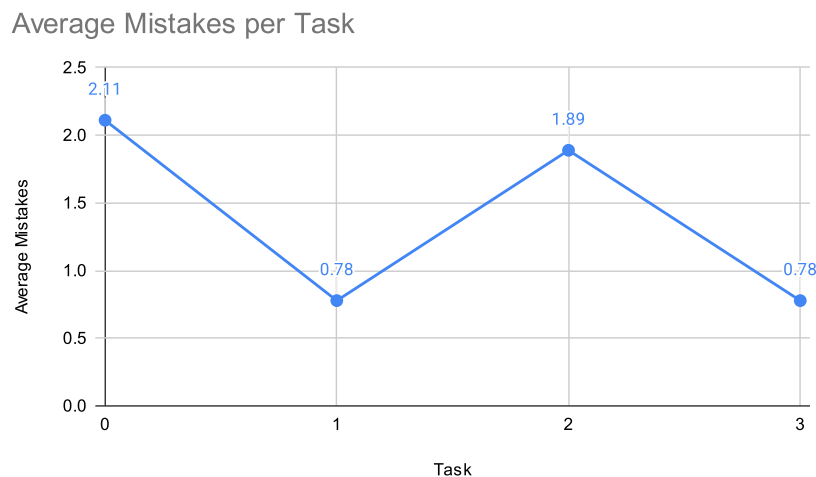


Figure 7.2: Evolution of the average mistakes per task

Task 0 had the highest number of errors. Task 1 showed a decrease in errors compared to Task 0. Task 2 exhibited a significant increase in errors relative to Task 1. Task 3 showed a decrease in errors compared to Task 2.

Incorrect removals were the most frequent error type, occurring 38 times across all participants, compared to 12 instances of **incorrect placements**. These errors primarily happened when participants did not fully release their grip after placing a piece, leaving their hand in a near-grabbing position. When the hand moved near a placed part, the tracking system sometimes registered an unintended grab, causing the part to be removed. Despite the higher number of incorrect removals, 89% of participants made no more than 5 such errors across all tasks. Placement accuracy averaged 99.18%, with individual user accuracy ranging from 95.12% to 100%, and the most common accuracy value being 100%.

No participant requested hints manually on tasks 2 and 3, where automatic hints were disabled.

7.2 Real-World Assembly Task Results

This section details the results of the real-world assembly task, analyzing descriptive statistics and performing a statistical analysis comparing the performance of both training groups. The raw data for the real-world assembly task performance can be found in Appendix A.2.

7.2.1 Statistical analysis

The statistical analysis compared the completion time and number of errors between the two groups in the real-world task. First, normality tests were conducted to determine the appropriate statistical method. Since there are two independent groups, a t-test was used if the data followed a normal distribution; otherwise, the Mann-Whitney U test was applied. To assess normality, histograms with kernel density estimation (KDE) curves, quantile–quantile (Q-Q) plots, and the Shapiro-Wilk test were used. The histograms provide an overview of the distribution shape, while the Q-Q plots allow for visual inspection of deviations from a normal distribution. The Shapiro-Wilk test served to further confirm the conclusions drawn from the visual representations. A p-value threshold of 0.05 was used to evaluate normality in the Shapiro-Wilk test, as well as for determining statistical significance in the comparative tests.

Furthermore, to investigate whether performance in the VR training could predict real-world outcomes, the Pearson correlation coefficient was used to assess the relationship between VR performance in Task 3 and the real-world assembly task. The analysis used the same significance threshold of 0.05 to determine the strength and relevance of the correlations. Scatter plots with fitted regression lines were generated to visualize these relationships.

7.2.1.1 Completion Time

Figures 7.3 and 7.4 show the corresponding visualizations for the completion time of the VR and traditional groups, respectively.

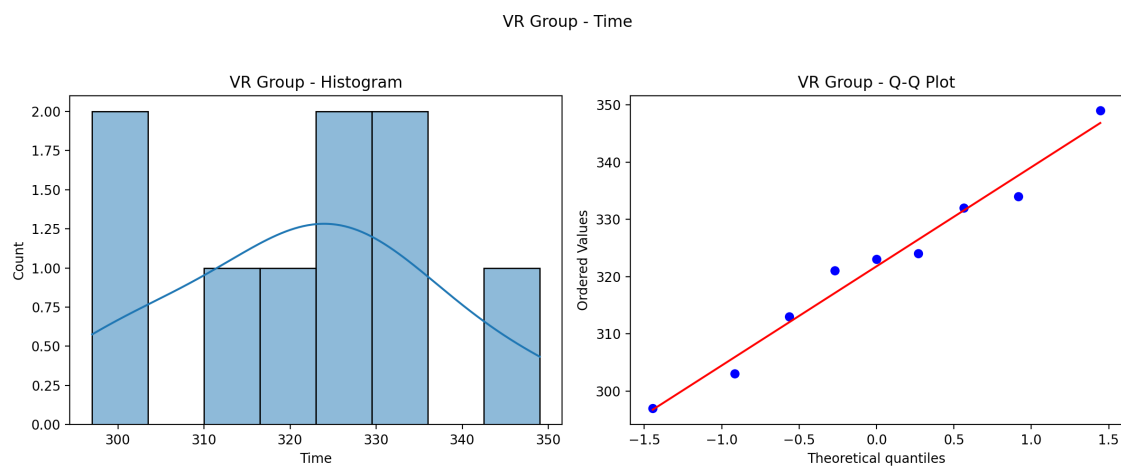


Figure 7.3: VR Group Time Normality

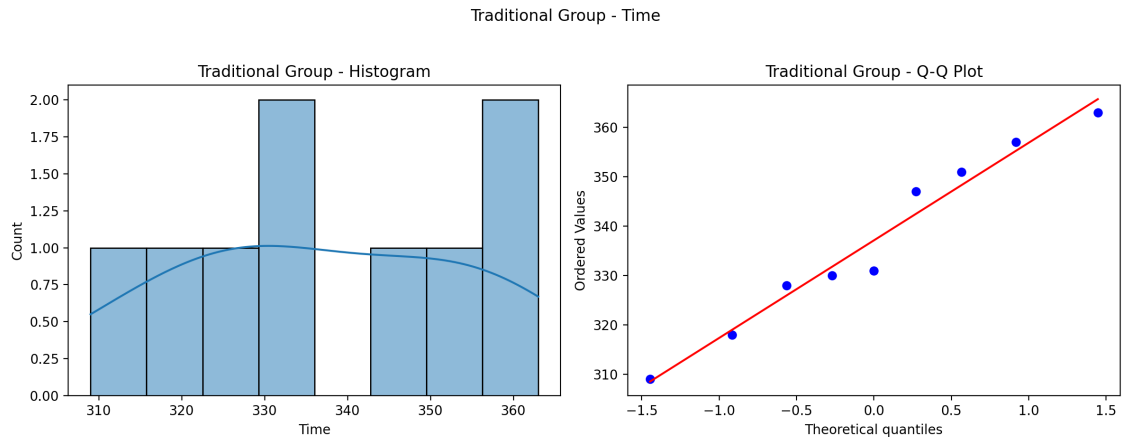


Figure 7.4: Traditional Group Time Normality

For the **VR** group, the KDE curve displayed a bell-shaped distribution similar to the normal distribution, and the points in the Q–Q plot closely followed the diagonal line, suggesting normality. For the traditional group, the KDE curve did not resemble a bell curve and appeared more flattened. However, the Q–Q plot still showed the data points lying approximately along the diagonal line. These visual results align with the outcomes of the Shapiro-Wilk test, which yielded $p = 0.9496$ and $p = 0.7415$ for the **VR** and traditional groups, respectively, which are far above the threshold, indicating that both groups did not significantly deviate from a normal distribution.

With the normality confirmed, an independent samples t-test was conducted to compare the completion times between both training groups. The test yielded $t = -1.892$ and $p = 0.07705$, which is above the threshold, indicating that the difference was not statistically significant. This means that there is no strong evidence of a performance difference in task completion time based on training method.

Figure 7.5 shows the correlation between **VR** Task 3 completion time and real-world task completion time with a scatter plot with a fitted regression line.

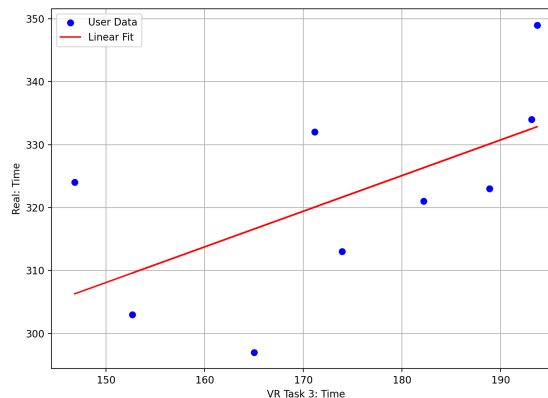


Figure 7.5: Correlation between VR and real-world completion time

The correlation coefficient for completion time was $r = 0.604$ with a p -value of 0.085, indicating a moderate positive correlation that is not statistically significant, as the p -value exceeds

the threshold. This suggests that while a faster time in VR tended to align with faster real-world time, the relation was not strong enough to validate it.

7.2.1.2 Number of Errors

Figures 7.6 and 7.7 show the corresponding visualizations for the number of errors of the VR and traditional groups, respectively.

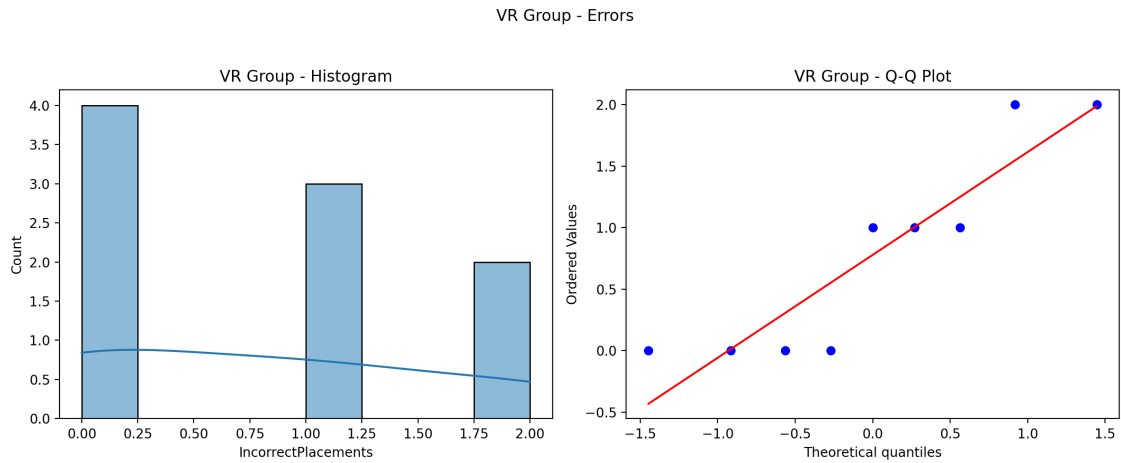


Figure 7.6: VR Group Errors Normality

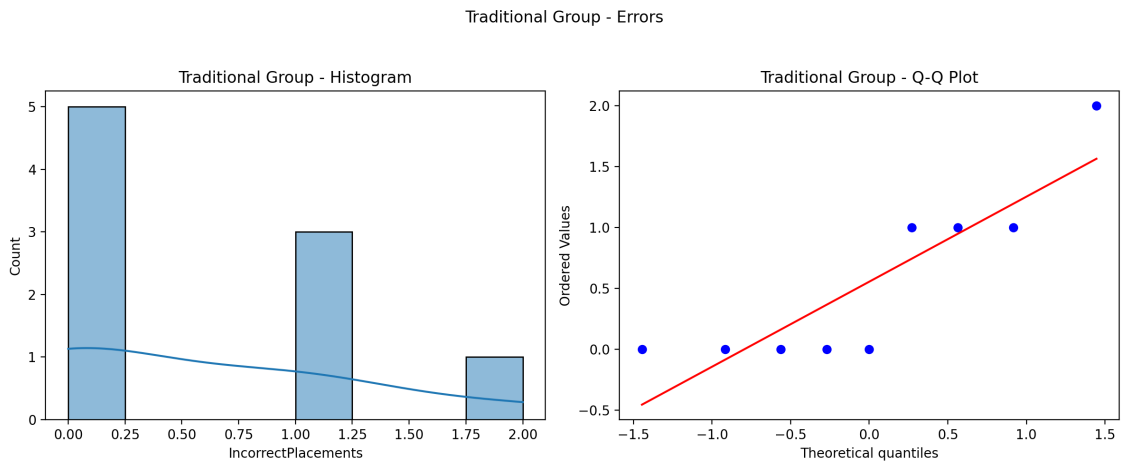


Figure 7.7: Traditional Group Errors Normality

The KDE curves for both groups did not exhibit the bell-shaped pattern characteristic of a normal distribution, and the points in the Q-Q plots deviated noticeably from the diagonal line, indicating a considerable deviation from a normal distribution. These visual observations were supported by the Shapiro-Wilk test results, which yielded $p = 0.02543$ for the VR group and $p = 0.00767$ for the traditional group, both below the 0.05 threshold. This confirmed that the error data for both groups did not follow a normal distribution.

Given the non-normal distribution in both groups, the Mann-Whitney U test was used to compare the number of errors between both groups. The test yielded $U = 46.500$ and $p = 0.59528$, indicating no statistically significant difference in the number of errors between the two training methods. This means that there is no strong evidence of a difference in the number of errors based on the training method.

Figure 7.8 shows the correlation between VR Task 3 and real-world task error counts.

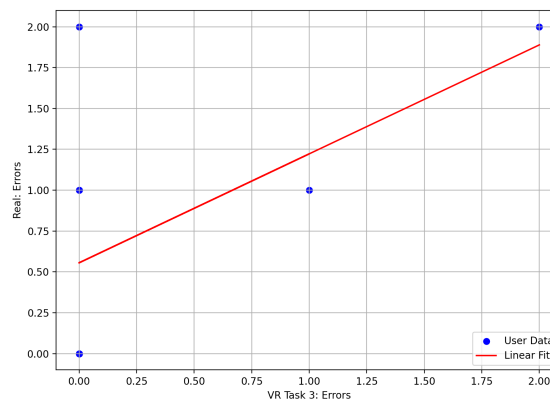


Figure 7.8: Correlation between VR and real-world errors

The correlation coefficient for the number of errors was $r = 0.566$ with a p -value of 0.112, indicating, similarly to the completion time, a moderate positive correlation that is not statistically significant, as the p -value exceeds the threshold. This suggests that while higher error counts in VR tended to align with high error counts in the real world, the relation was not strong enough to validate it.

7.2.2 Descriptive Analysis

Descriptive statistics for completion time and number of errors are summarized in Table 7.1.

Group	Mean Time (s)	Std. Dev. Time (s)	Mean Errors	Std. Dev. Errors
Overall	329.44	18.45	0.67	0.77
VR	321.78	15.99	0.78	0.83
Traditional	337.11	18.31	0.56	0.73

Table 7.1: Descriptive statistics

Overall, the average completion time for all participants was approximately 329.4 seconds, with a mean error count of 0.67. The standard deviations indicate variability, with completion times showing greater spread than error counts.

Comparing the two groups, participants in the VR group completed the task in an average of 321.8 seconds, while those in the traditional training group averaged 337.1 seconds. The VR group had a mean error count of 0.78, compared to 0.56 for the traditional group. The standard

deviation for completion time was slightly higher in the traditional training group, while the VR group showed a slightly higher variability in error count.

The fastest completion time was 4 minutes and 57 seconds (297 seconds), recorded by a participant in the VR group. The slowest completion time was 6 minutes and 3 seconds (363 seconds), from the traditional training group. This corresponds to a range of 1 minute and 6 seconds (66 seconds) between the fastest and slowest completion times.

The number of errors made by participants ranged from 0 to 2. Half of the participants completed the assembly without errors. Among participants who made errors, two-thirds made one error, while one-third made two errors. Figure 7.9 shows the number of participants (y-axis) with n errors (x-axis) for each training group.

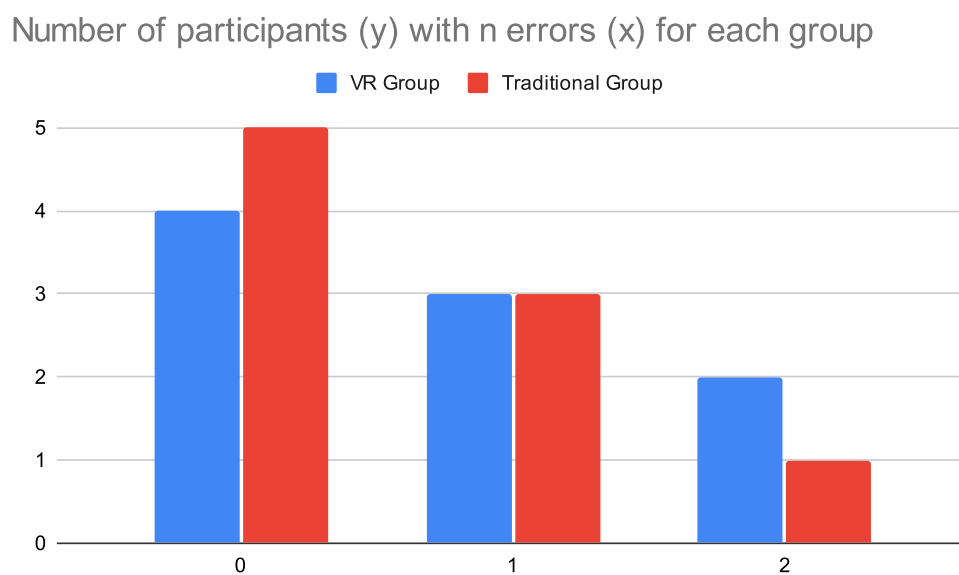


Figure 7.9: Participants with n errors for each group

The traditional group had more participants with no errors and fewer participants with two errors. All participant errors involved the pieces **chairLeft** and **chairRight**, which have similar appearances leading participants to occasionally mistake these pieces or place them in an incorrect orientation.

7.3 Subjective Feedback

This section presents the qualitative results from questionnaires and participant comments, relative to all stages and groups of the experimental procedure: VR training, traditional training, and the real-world assembly.

7.3.1 Virtual Reality Training Feedback

Participants in the VR group answered questions designed to assess effectiveness, efficiency and satisfaction with the VR system (from VRSUQ), as well as additional questions evaluating the perceived effectiveness of the assembly training. The average VRSUQ score was 95.12 out of 100, ranging from 86.11 to 97.22. Figure 7.10 shows a stacked column chart showing the frequency of each score for each VRSUQ question.

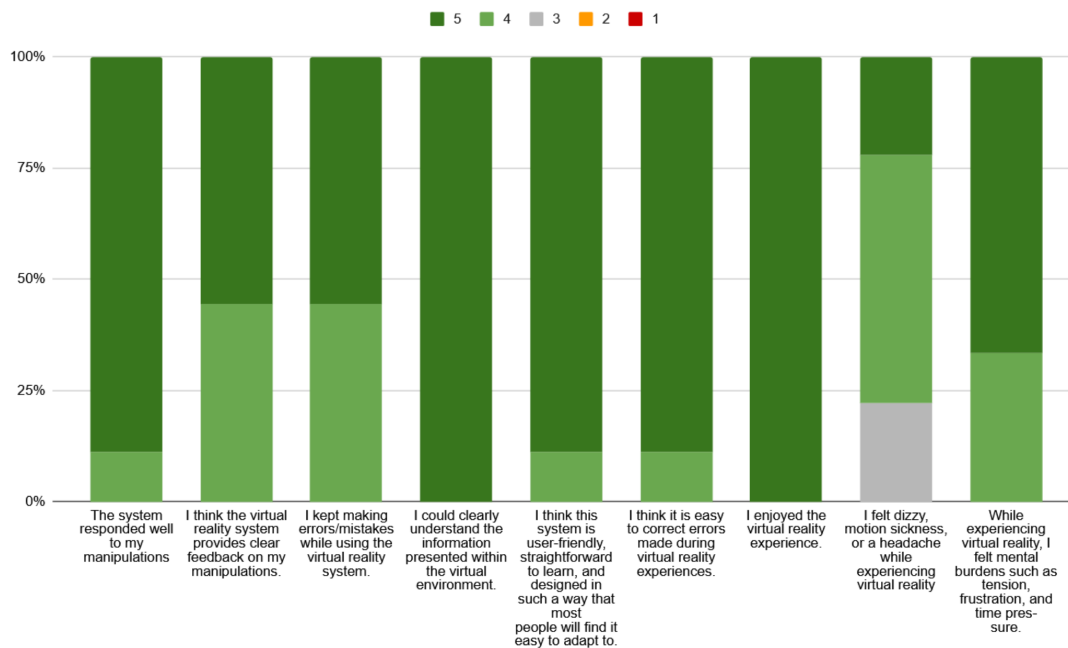


Figure 7.10: VRSUQ Results

The results cover three evaluated dimensions: effectiveness (questions 1–3), efficiency (questions 4–6), and satisfaction (questions 7–9). Among these, efficiency shows the highest scores, with question 4 receiving a perfect score from every participant. Question 8 received the lowest scores, with over 75% of participants reporting some degree of motion sickness, dizziness, or headache while in virtual reality. No participant gave this question a score below 3. Question 7 received a perfect score from all participants.

The issue of accidentally removing a correctly placed piece is not strongly reflected in the questionnaire results, as participants did not report frequent mistakes (question 3) nor frustration (question 9). This issue occurred occasionally and was resolved easily.

The questions evaluating VR assembly training also had positive results, as shown by Figure 7.11

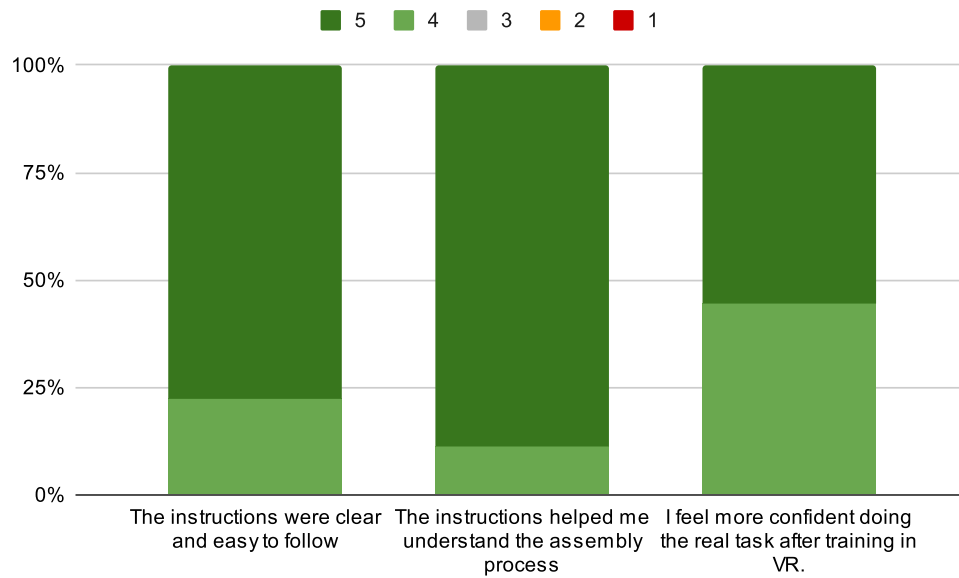


Figure 7.11: VR Training Feedback

The results show that participants found the instructions easy to follow and helpful for understanding the assembly. Participants also reported confidence in their ability to complete the real assembly after training in VR.

When asked about problems encountered during the VR training session, participants mentioned accidentally removing pieces. Some also reported difficulties with toggling the hand menu, where the pinching gesture required to open and close sometimes took multiple attempts to register correctly.

7.3.2 Traditional Training Feedback

The participants in the traditional training group answered a similar set of three questions, as shown in Figure 7.12



Figure 7.12: Traditional Training Feedback

The results were similar to those of the **VR** training group, with participants being able to easily follow the instructions and understand the assembly process, feeling more confident in doing the real assembly after the explanation.

7.3.3 Real-world Assembly Feedback

The participants from both groups answered three questions about their experience with the real-world assembly after their respective training. The results are shown in Figure 7.13

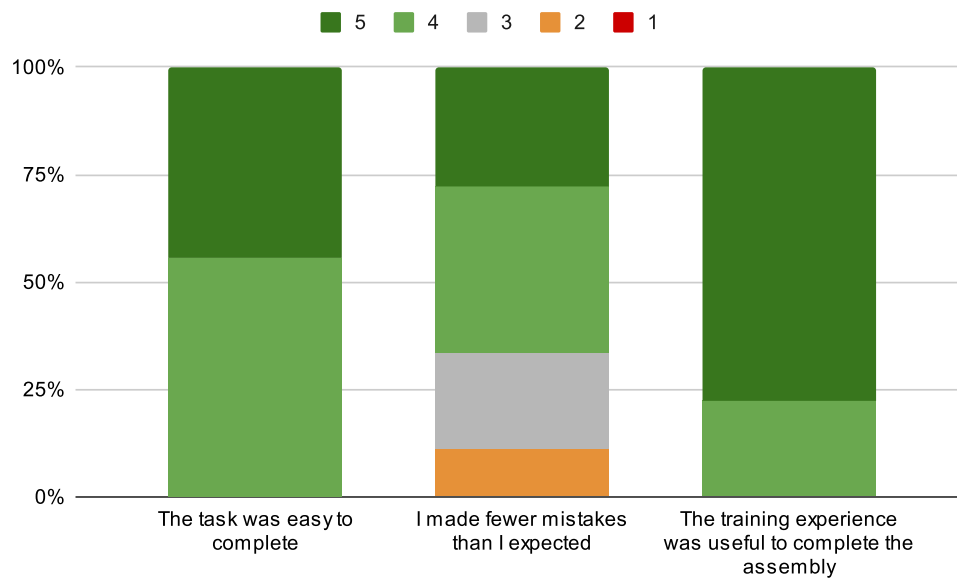


Figure 7.13: Real-world Assembly Feedback

Most participants believed that the task was easy to complete, and their prior training helped them during the real assembly. The results are more varied, with most participants making more mistakes than they expected.

The only problem the participants reported was some difficulty in fitting some of the pieces, with some of the connections between the 3D-printed and wooden parts being too tight.

7.4 Discussion

The results of this experiment aimed to assess the effectiveness of **VR** training compared to traditional means. While the data revealed slight performance advantages for the **VR** group, statistical analysis did not confirm any significant differences between training groups. These findings form the foundation for addressing **RQ1** and **RQ2**.

7.4.1 Learning Trends in Virtual Reality

The performance across the four **VR** tasks showed clear evidence of learning. Task completion time and error count decreased significantly after Task 0, in which participants familiarized themselves with the system and the assembly. The sharp drop in both metrics in Task 1 indicates rapid adaptation. Task 2 saw an increase in time and errors following the removal of automatic hints, with performance improving again in Task 3, which had the shortest average time and fewer mistakes. The rebound suggests that participants effectively learned the assembly steps and interaction methods, at least within the **VR** setting.

The fact that no participant requested hints manually in the latter two tasks reinforces the notion that users had developed sufficient confidence during the initial practice. Moreover, the

high placement accuracy and rarity of mistakes, even without guidance, support the idea that participants internalized the process effectively. These results highlight the potential of VR for skill acquisition, a key concern of RQ1.

7.4.2 Transfer of Knowledge to Real-World Performance

The results of the real-world assembly revealed only marginal differences between the VR and traditional training groups. Although the VR group had a slightly faster average completion time and slightly higher error rates, neither metric reached statistical significance.

This outcome might have been affected by task simplicity. The overall number of errors was low, with half of the participants making no mistakes, and most of those who did making only one. This suggests that the task might have been too simple to reveal any performance gaps between both training methods. In low-difficulty contexts, the advantages in training effectiveness could be overshadowed by individual differences between participants. The small sample size may have also masked any differences in performance, preventing the results from having statistical significance.

Furthermore, the correlation between the performance of the VR group in virtual and physical settings also failed to achieve significance. This may again be attributed to a small sample size, but could also reflect a mismatch between both assembly mediums. While VR training does offer immersive spatial interaction, the transition to real-world physical manipulation introduced new stimuli. For instance, grabbing a virtual object lacks the tactile feedback of real components, and fitting two pieces together in VR does not provide the physical resistance experienced in the real world. These points underscore a key aspect of RQ1: the extent to which VR training transfers effectively to real-world performance.

7.4.3 Error Behavior and Technology Limitations

Incorrect piece removals were exclusive to the VR training environment, with no participants removing a correctly placed piece in the real-world assembly. These errors stemmed from issues with the hand-tracking system rather than user mistakes. This issue could be mitigated by adjusting some of the hand tracking sensitivity and pinch gesture thresholds, but the root of the issue is users unintentionally moving their hand through the placed pieces in a near-grabbing pose, which doesn't occur in the real world due to physical collisions. Additionally, the difficulty in toggling the hand menu reported by some users is also caused by a failure to properly register the pinch gesture.

While these limitations did not drastically impact learning, they likely influenced user experience, an important factor in comparing training approaches, as addressed in RQ2.

7.4.4 User Feedback

The subjective feedback from questionnaires and participant comments provided useful insight into the strengths and weaknesses of both training methods.

Participants in the **VR** group reported high satisfaction with the system, reflected by the high **VRSUQ** score. They found the system easy to use and understand, effective and responsive. However, most users reported mild degrees of motion sickness, dizziness or headaches. Neither this nor the occasional accidental removal of pieces appeared to impact their overall enjoyment, as participants reported high levels of satisfaction and minimal frustration. Most participants also found the system effective for teaching the assembly process and expressed confidence in their ability to complete the real assembly, suggesting that the system successfully conveyed the training objectives and provided instructions clearly and effectively. Overall, these results reinforce the potential of **VR** as an engaging and efficient training tool, in line with **RQ2**, despite some drawbacks related to motion sickness and interaction challenges.

Participants in the traditional training group also reported a positive experience, finding the instructions clear and easy to follow, and being confident in performing the real assembly.

Feedback after the real-world assembly highlighted that most participants found the task manageable and that their prior training was helpful. The difficulty participants reported in fitting some pieces is likely due to 3D printing imperfections, probably resulting from printing the parts in a batch.

7.4.5 Implications and Future Directions

The findings of this study highlight the viability of **VR**-based training as an effective alternative to traditional methods for assembly training. Although no statistically significant differences were found between groups in real-world performance, the comparable results suggest that **VR** training can prepare users with similar proficiency. These findings support the general premise of both **RQ1** and **RQ2**, showing that **VR** can deliver training outcomes that are at least on par with conventional methods, while offering additional benefits such as a risk-free training environment and remote accessibility.

An important implication concerns the role of automated guidance during training. The temporary removal of hints in the last **VR** tasks led to an initial decrease in performance, reflecting the challenge of transitioning to independent problem-solving. However, performance improved again in the next task, suggesting that users were able to adapt and internalize the assembly process. This indicates the potential value of a **VR** system where guidance is phased out progressively to promote autonomy. Future research should explore adaptive hint mechanisms that respond dynamically to individual performance, gradually reducing assistance based on user readiness.

Enhancing hand tracking precision and providing clearer feedback could help minimize the interaction issues observed in this study, especially for tasks that require fine motor skills. Similarly, though motion sickness was generally mild, its consistent presence highlights the need for further refinement, especially if **VR** is to be used for longer sessions.

Looking forward, there is potential to apply **VR** training to more complex and varied assembly tasks, including those that involve teamwork or real-time collaboration. Doing so would offer deeper insight into how **VR** performs across different assembly scenarios and needs.

Chapter 8

Conclusions

8.1 Main Contributions

This dissertation has presented a comprehensive framework for VR-based assembly training assembly, addressing several key challenges identified in existing VR training approaches. The main contributions can be summarized as follows:

- Developed a detailed and implementable VR training system that incorporates interactive assembly components and a guided hint system that respects assembly logic and dependencies.
- Introduced a structured assembly sequence framework with enforced slot activation conditions, enabling a controlled yet flexible training experience.
- Demonstrated evidence of effective learning within a VR environment.
- Provided empirical insights on skill transfer from VR to real-world assembly.
- Identified key limitations affecting VR training performance and user experience.

The creation of an interactive VR training system that incorporates assembly logic and dependency-aware hints represents a meaningful advancement toward more realistic and effective virtual training environments. By embedding assembly sequence constraints directly into the system, learners are guided to engage with the task in a logical order, which helps build a deeper understanding while avoiding cognitive overload that can occur in less structured VR experiences. This approach supports controlled progression through the assembly while still allowing flexibility in how learners approach the task.

The evidence of effective learning within the VR environment confirms that thoughtfully designed interactive systems can promote skill acquisition. This supports the view that VR can serve as a credible alternative to traditional training, particularly in situations where physical resources are limited or risks are involved.

By exploring how skills transfer from VR to real-world assembly, this work offers valuable data to help refine VR training design. Although the transfer results were broadly comparable to traditional methods, the findings highlight specific areas where VR systems can improve to better prepare users for real-world tasks.

Thus, this dissertation lays a foundation for future developments in VR-based assembly training, demonstrating its potential as an effective and flexible learning tool.

8.2 Future work

There are several promising directions to build upon the work of this dissertation. One clear next step is to test VR training with more complex and varied assembly tasks. Introducing multi-step processes, different types of components, and scenarios more prone to errors could help reveal more clearly how VR compares to traditional methods and where it offers unique benefits.

Exploring collaborative training in VR is another interesting area. Adding real-time teamwork and communication challenges would better reflect real industrial settings and show how VR supports group learning and coordination.

Improving the interaction experience remains important. Enhancing hand tracking accuracy and incorporating more realistic haptic feedback could reduce unintended mistakes and increase users' sense of presence, which may lead to better transfer of skills to real-world assembly.

Developing smarter, adaptive guidance systems that adjust hints based on how learners progress could also make training more effective. Gradually reducing assistance as users become more confident encourages independent problem-solving and deeper understanding.

User comfort is another critical aspect. Finding ways to reduce motion sickness and other discomforts will be vital for longer or repeated training sessions, helping VR become a practical option for everyday use.

Beyond these points, further refinement of the VR training framework itself is needed. Integrating VR training with external management systems, such as Manufacturing Execution Systems (MES) or Learning Management Systems (LMS), would also open the door for seamless tracking, scheduling, and reporting in real-world industrial environments.

Chapter 9

Evaluation of the Work in the Context of the United Nations Sustainable Development Goals

This chapter evaluates the contributions of the **VR** assembly training system presented in this dissertation against relevant United Nations Sustainable Development Goals (**SDGs**). The **SDGs** provide a global framework for addressing major social, economic, and environmental challenges. By situating this work within the **SDG** framework, we assess how it advances sustainable development through improved training methods, skill acquisition, and workforce development.

SDG	Target	Contribution	Performance Indicators and Metrics
4: Quality Education	4.4: Increase the number of youth and adults with relevant skills for employment.	The VR training system supports effective and scalable skill development in assembly tasks using immersive, interactive modules. It enhances access to technical education while reducing barriers to workforce entry. By allowing flexible deployment with minimal supervision, it supports continuous, self-paced learning applicable to real-world industrial settings.	Performance is evaluated through VR and real-world task accuracy, completion times, user confidence, and training completion rates.
8: Decent Work and Economic Growth	8.8: Protect labour rights and promote safe and secure working environments for all workers.	The VR system creates a risk-free, simulated environment where learners can practice industrial tasks without exposure to physical hazards. It promotes safety-first education, especially in high-risk assembly and manufacturing scenarios.	Effectiveness is reflected in the absence of training-related injuries, increased adoption for hazardous tasks, and high participant-reported safety and confidence.
9: Industry, Innovation and Infrastructure	9.5: Enhance scientific research and upgrade technological capabilities of industrial sectors.	This research contributes to innovation in training methodologies through adaptive, dependency-aware hinting systems and modular assembly logic. It paves the way for integration with industrial platforms, supporting smarter workforce development pipelines.	Success is measured through system implementation milestones, preliminary integrations with industrial software, and industry feedback from pilot deployments.

Table 9.1: Evaluation of the VR Training System Against Selected UN Sustainable Development Goals

In summary, the **VR** training system contributes positively to several **SDGs** by enhancing education quality, supporting workforce readiness, promoting technological innovation, encouraging sustainable training practices, and maintaining user well-being. These impacts highlight the potential of immersive technologies to align with broader sustainable development priorities in industry and education.

References

- [1] Statista Market Insights. Ar vr - worldwide, 2024.
- [2] Lea M. Daling, Marisa Tenbrock, Ingrid Isenhardt, and Sabine J. Schlittmeier. Assemble it like this! – Is AR- or VR-based training an effective alternative to video-based training in manual assembly? *Applied Ergonomics*, 110:104021, July 2023.
- [3] Lea M. Daling and Sabine J. Schlittmeier. Effects of augmented reality-, virtual reality-, and mixed reality-based training on objective performance measures and subjective evaluations in manual assembly tasks: A scoping review. *Human Factors*, 66(2):589–626, 2024.
- [4] Jared H. Hornsby, Austin Arnwine, Ashley Stowe, Mary Lin, Chris Mason, Mike Eakins, Devon Veller, Christian Mendoza, David Metcalf, Victor Oriafio, Andrew Dunar, and Bob Crockett. Vr haptics for glovebox operations training. *Frontiers in Virtual Reality*, Volume 6 - 2025, 2025.
- [5] Chih-Hsing Chu and Ching-Hung Ko. An experimental study on augmented reality assisted manual assembly with occluded components. *Journal of Manufacturing Systems*, 61:685–695, October 2021.
- [6] Stephanie Schwarz, Georg Regal, Marina Kempf, and Raimund Schatz. Learning Success in Immersive Virtual Reality Training Environments: Practical Evidence from Automotive Assembly. In *Proceedings of the 11th Nordic Conference on Human-Computer Interaction: Shaping Experiences, Shaping Society*, NordiCHI '20, pages 1–11, New York, NY, USA, October 2020. Association for Computing Machinery.
- [7] Stefan Werrlich, Carolin Lorber, Phuc-Anh Nguyen, Carlos Emilio Franco Yanez, and Gunther Notni. Assembly Training: Comparing the Effects of Head-Mounted Displays and Face-to-Face Training. In Jessie Y.C. Chen and Gino Fragomeni, editors, *Virtual, Augmented and Mixed Reality: Interaction, Navigation, Visualization, Embodiment, and Simulation*, pages 462–476, Cham, 2018. Springer International Publishing.
- [8] Teravision Technologies. The business and technical challenges of ar and vr adoption. <https://www.teravisiontech.com/blog/the-business-and-technical-challenges-of-ar-and-vr-adoption>, 2023. Accessed: 2025-06-29.
- [9] Oleksii Shatokhin, Andrius Dzedzickis, Milda Pečiulienė, and Vytautas Bučinskas. Extended reality: Types and applications. *Applied Sciences*, 15(6), 2025.
- [10] Xiaonan Zhu, Po-Lin Lai, Xinjie Li, Yaoyan Wang, and Xi Pei. A systematic review of literature on the use of extended reality in logistics and supply chain management education: Evolution of research themes and system-level trends. *Systems*, 13(7), 2025.

- [11] André Stork, Oliver Bimber, and Raffaele Amicis. Projection-based augmented reality in engineering applications. 03 2002.
- [12] VR & AR Wiki contributors. Inside-out tracking. https://vrrarwiki.com/wiki/Inside-out_tracking, 2025. Accessed: 2025-06-29.
- [13] VR & AR Wiki contributors. Outside-in tracking. https://vrrarwiki.com/wiki/Outside-in_tracking, 2025. Accessed: 2025-06-29.
- [14] Yang Gao and Charles Spence. Enhancing presence, immersion, and interaction in multisensory experiences through touch and haptic feedback. *Virtual Worlds*, 4(1), 2025.
- [15] VR & AR Wiki contributors. Eye tracking. https://vrrarwiki.com/wiki/Eye_tracking, 2025. Accessed: 2025-06-29.
- [16] Rachel Albert, Anjul Patney, David Luebke, and Joohwan Kim. Latency requirements for foveated rendering in virtual reality. *ACM Transactions on Applied Perception*, 14(4):25:1–25:13, 2017.
- [17] VR & AR Wiki contributors. Foveated rendering. https://vrrarwiki.com/wiki/Foveated_rendering, 2025. Accessed: 2025-06-29.
- [18] Munir Oudah, Ali Al-Naji, and Javaan Chahl. Hand gesture recognition based on computer vision: A review of techniques. *Journal of Imaging*, 6(8):73, 2020.
- [19] Microsoft Learn. Voice input on hololens. <https://learn.microsoft.com/en-us/windows/mixed-reality/design/voice-input>, March 2022. Accessed: 2025-06-29.
- [20] VR & AR Wiki contributors. Virtual reality. https://vrrarwiki.com/wiki/Virtual_Reality, 2025. Accessed: 2025-06-29.
- [21] Théo Combe, Jean-Rémy Chardonnet, Frédéric Merienne, and Jivka Ovtcharova. Cave and HMD: Distance perception comparative study. *Virtual Reality*, 2023. Published online March 29, 2023; Accessed: 2025-06-29.
- [22] Thomas Potter, Zoran Cvetković, and Enzo De Sena. On the relative importance of visual and spatial audio rendering on vr immersion. *Frontiers in Signal Processing*, Volume 2 - 2022, 2022.
- [23] VR/AR Wiki contributors. Augmented reality. https://vrrarwiki.com/wiki/Augmented_Reality, n.d. Accessed: 2025-06-29.
- [24] Microsoft. Hololens 2 hardware, 2023. Accessed: 2025-06-29.
- [25] Interaction Design Foundation - IxDF. 2023. What is simultaneous localization and mapping (slam)?, 2023. Accessed: 2025-06-29.
- [26] Sharon Macias-Velasquez, Hugo I. Medellin-Castillo, and Abel Garcia-Barrientos. New-user experience evaluation in a semi-immersive and haptic-enabled virtual reality system for assembly operations. *International Journal of Human-Computer Studies*, 190:103312, October 2024.

- [27] Diep Nguyen and Gerrit Meixner. Gamified Augmented Reality Training for An Assembly Task: A Study About User Engagement. In *2019 Federated Conference on Computer Science and Information Systems (FedCSIS)*, pages 901–904, September 2019. ISSN: 2300-5963.
- [28] Stefan Werrlich, Austino Daniel, Alexandra Ginger, Phuc-Anh Nguyen, and Gunther Notni. Comparing HMD-Based and Paper-Based Training. In *2018 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 134–142, October 2018. ISSN: 1554-7868.
- [29] Josef Wolfartsberger, Robert Zimmermann, Gabriele Obermeier, and Daniel Niedermayr. Analyzing the potential of virtual reality-supported training for industrial assembly tasks. *Computers in Industry*, 147:103838, May 2023.
- [30] Xiaonan Yang, Wanting Mao, Yaoguang Hu, Jingfei Wang, Xuanzhu Wan, and Haonan Fang. Does augmented reality help in industrial training? A comprehensive evaluation based on natural human behavior and knowledge retention. *International Journal of Industrial Ergonomics*, 98:103516, November 2023.
- [31] Sebastian Büttner, Michael Prilla, and Carsten Röcker. Augmented Reality Training for Industrial Assembly Work - Are Projection-based AR Assistive Systems an Appropriate Tool for Assembly Training? In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, CHI '20, pages 1–12, New York, NY, USA, April 2020. Association for Computing Machinery.
- [32] Emanuele Marino, Loris Barbieri, Fabio Bruno, and Maurizio Muzzupappa. Assessing user performance in augmented reality assembly guidance for industry 4.0 operators. *Computers in Industry*, 157-158:104085, May 2024.
- [33] Daniel Niedermayr, Josef Wolfartsberger, and Martina Maurer. Virtual Reality for Industrial Assembly Training: The Impact of Tool Interaction Realism on Learning Outcomes. In *2023 IEEE International Symposium on Mixed and Augmented Reality Adjunct (ISMAR-Adjunct)*, pages 183–190, October 2023. ISSN: 2771-1110.
- [34] Tim Bosch, Gu van Rhijn, Frank Krause, Reinier Könemann, Ellen S. Wilschut, and Michiel de Looze. Spatial augmented reality: a tool for operator guidance and training evaluated in five industrial case studies. In *Proceedings of the 13th ACM International Conference on Pervasive Technologies Related to Assistive Environments*, PETRA '20, pages 1–7, New York, NY, USA, June 2020. Association for Computing Machinery.
- [35] Meta Platforms, Inc. Set up unity for xr development, 2025. Accessed: 2025-06-27.
- [36] Meta Platforms, Inc. Meta xr all-in-one sdk (upm), 2025. Accessed: 2025-06-27.
- [37] Meta. Explore meta quest features with building blocks, 2025. Accessed: 2025-06-27.
- [38] ScazLab. Hrc model set, 2018. Accessed: 2025-06-14.
- [39] Yong Min Kim and Ilsun Rhiu. Development of a virtual reality system usability questionnaire (vrсуq). *Applied Ergonomics*, 119:104319, 2024.

Appendix A

Results (Raw Data)

A.1 VR Training Data

Table A.1: VR Task Performance Data

UserID	TaskID	Time (s)	Correct Placements	Incorrect Placements	Correct Removals	Incorrect Removals	Hints
1	0	717.98	41	0	0	3	45
1	1	301.82	38	1	1	0	40
1	2	302.85	44	0	0	6	0
1	3	188.88	40	2	2	2	0
4	0	850.16	38	0	0	0	42
4	1	243.37	38	0	0	0	38
4	2	250.22	41	2	2	3	0
4	3	152.67	38	0	0	0	0
5	0	741.74	40	0	0	2	42
5	1	287.87	38	0	0	0	38
5	2	298.15	39	0	0	1	0
5	3	182.18	38	0	0	0	0
7	0	878.44	41	0	0	3	46
7	1	319.86	39	0	0	1	40
7	2	313.54	39	2	2	1	0
7	3	193.13	38	0	0	0	0
9	0	762.93	40	1	1	2	45
9	1	307.67	39	0	0	1	40
9	2	321.18	38	1	1	0	0
9	3	173.91	38	0	0	0	0
11	0	692.82	40	0	0	2	42
11	1	283.90	39	0	0	1	40
11	2	319.38	38	0	0	0	0
11	3	171.14	38	0	0	0	0
14	0	762.67	40	1	1	2	46
14	1	281.57	39	0	0	1	40
14	2	297.35	38	1	1	0	0
14	3	193.72	39	0	0	1	0
15	0	803.49	39	0	0	1	41
15	1	258.89	39	0	0	1	40
15	2	280.67	38	0	0	0	0
15	3	165.01	38	1	1	0	0
17	0	828.91	40	0	0	2	42
17	1	255.55	39	0	0	1	40
17	2	268.04	38	0	0	0	0
17	3	146.82	39	0	0	1	0

A.2 Real-World Assembly Data

Table A.2: Real-World Task Performance Data

Group	UserID	Time (s)	Correct Placements	Incorrect Placements	Correct Removals	Incorrect Removals
vr	1	323	38	2	2	0
trad	2	363	38	0	0	0
trad	3	347	38	1	1	0
vr	4	303	38	1	1	0
vr	5	321	38	0	0	0
trad	6	330	38	0	0	0
vr	7	334	38	2	2	0
trad	8	351	38	0	0	0
vr	9	313	38	0	0	0
trad	10	328	38	1	1	0
vr	11	332	38	1	1	0
trad	12	357	38	0	0	0
trad	13	318	38	1	1	0
vr	14	349	38	0	0	0
vr	15	297	38	1	1	0
trad	16	309	38	2	2	0
vr	17	324	38	0	0	0
trad	18	331	38	0	0	0