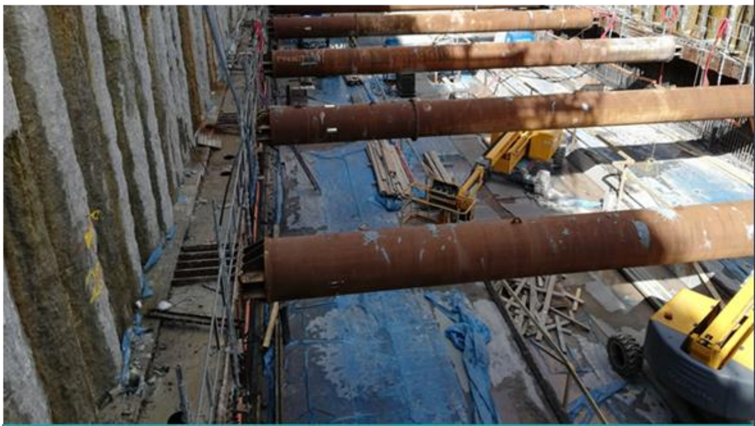

COGITO

CONSTRUCTION PHASE
DIGITAL TWIN MODEL

cogito-project.eu

D4.7 –

Interactive Visual Material for Workforce Training v1

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 958310

D4.7 – Interactive Visual Material for Workforce Training v1

Dissemination Level:	Public
Deliverable Type:	Demonstrator
Lead Partner:	AU
Contributing Partners:	CERTH, RSRG
Due date:	31-05-2022
Actual submission date:	22-06-2022

Authors

Name	Beneficiary	Email
Jochen Teizer	AU	teizer@cae.au.dk
Aparna Harichandran	AU	aparnaharichandran@cae.au.dk

Reviewers

Name	Beneficiary	Email
Tobias Hanel	FER	thanel@ferrovial.com
Elias Bruno Meusburger	RSRG	elias.meusburger@rsrg.com

Version History

Version	Editors	Date	Comment
0.1	AU	20.05.2022	Table of Contents
0.3	AU	24.05.2022	Sections 1, 2
0.4	AU	31.05.2022	Sections 3, 4
0.5	AU	31.05.2022	Draft Version Ready for Internal Review
0.6	FER	01.06.2022	FER's Comments in v0.5
0.7	RSRG	07.06.2022	RSRG's Comments in v0.5
0.8	AU	13.06.2022	Internal Review Comments Addressed
0.9	Hypertech	21.06.2022	Quality Checks
1.0	Hypertech, AU	22.06.2022	Deliverable Submission to the EU portal

Disclaimer

Executive Summary

The COGITO Deliverable D4.7 "*Interactive Visual Material for Workforce Training v1*" documents the work done towards developing a VirtualSafety application that enables personalized safety education and learning for the construction workforce. The overall objective of T4.4 is the prototype implementation of visual and interactive training materials~~,~~ using Virtual Reality. The VirtualSafety application dynamically updates VR training games through information streaming from the digital twins. The training scenarios in VR games are automatically created from the project intent information, project status knowledge, safety regulations, and historical knowledge provided by the digital twins. Therefore, construction workers can be trained in realistic training environments with relevant tasks they soon afterward pursue. Dynamic VR training is expected to be mutually beneficial for the Digital Twin Platform (DTP) and support periodic construction safety management. This deliverable describes work conducted between M10 and M19 for developing the VirtualSafety application and the conceptual components behind the development. A summary of the work done between M10 to M19 for developing VirtualSafety and a detailed example of creating one training scenario involving hand gesture recognition and analysis in VR are included.

Table of contents

List of Figures

List of Tables

List of Acronyms

Term	Description
COGITO	Construction Phase diGItal Twin mOdel
DOF	Degree of Freedom
DIP	Distal interphalangeal
DT	Digital Twin
DTCS	Digital Twin for Construction Safety
DTP	Digital Twin Platform
HMD	Head-mounted Display
IP	Interphalangeal
MCP	Metacarpophalangeal
PIP	Proximal interphalangeal
PPE	Personal Protective Equipment
VR	Virtual Reality

1 Introduction

The construction industry has remained one of the most dangerous workplaces for the past several years, and the latest report shows the highest increase in fatalities since 2007 (Bureau of Labor Statistics, 2020). Several training sessions are conducted for construction workers to create safety awareness. However, the injury rates in the industry remain high (Hou et al., 2021), and there is a compelling need to improve safety training methods. Even though the effectiveness of traditional training methods is supported by statistical evidence, the computer-aided training methods are superior in many aspects (Gao et al., 2019). Better engagement of trainees, provision of text-free interfaces and representation of actual workplaces are notable advantages of computer-aided training methods such as Virtual Reality (VR) serious games. Besides, recent studies in VR training demonstrate the possibility of automated performance assessment (Golovina et al., 2019) through run-time data collection and collaborative training using multiplayer games (Jacobsen et al., 2021).

The Digital Twin (DT) is an up-to-date digital counterpart of the physical entities in a system (Tao et al., 2019). The physical entities in construction include objects and processes in a construction project. The vast potential of digital twins for various applications such as construction safety, progress monitoring, resource allocation, and decision making is yet to be explored. Seamless data collection and transfer through DT is possible with state-of-the-art sensing technologies. The information from digital twins can significantly improve VR training in many aspects. Compared to conventional safety training methods, VR games offer participants a more immersive and interactive learning experience. However, training scenarios in most existing VR games lack complex tasks and the realistic environment required for actual construction.

1.1 Scope and Objectives of the Deliverable

The overall objective of T4.4 is the prototype implementation of visual and interactive training materials, using VR. The corresponding deliverables, D4.7 and D4.8, aim to develop the Health and Safety educational and training application called VirtualSafety. This application dynamically updates VR training games through information streaming from the digital twins. The training scenarios in VR games are automatically created from the project intent information, project status knowledge, safety regulations, and historical knowledge provided by the digital twins. Therefore, construction workers can be trained in realistic training environments with relevant tasks they soon afterwards pursue. Dynamic VR training is expected to be mutually beneficial for the Digital Twin Platform (DTP) and support periodic construction safety management. This deliverable describes work conducted between M10 to M 19 for developing the VirtualSafety application and the conceptual components behind the development.

1.2 Relation to other Tasks and Deliverables

This deliverable builds upon D2.5 “COGITO System Architecture v2” and mainly draws as-built 3D BIM from the Digital Twin Platform (DTP) (documented in D7.9 and D7.10). The 3D BIM enhanced with safety information (output of SafeConAI) is used to create training scenarios involving safe working practices. Therefore, this deliverable is also dependent on the Preventive Health and Safety Application, SafeConAI (D4.1 and D4.2).

1.3 Structure of the Deliverable

The rest of the document is organized as follows:

- Section 2 provides an overview of the Digital Twin for Construction Safety (DTCS) and integration of VirtualSafety;
- Section 3 presents the VirtualSafety and its components in detail; and
- Section 4 summarizes and concludes the work done for this deliverable.

2 Overview of Digital Twin for Construction Safety

The overall architecture of the Digital Twin for Construction Safety (DTCS) developed for the COGITO ecosystem is illustrated in Figure 1 (Harichandran et al., 2021; Teizer et al., 2022). The DTCS has four principal components: prevention through design and planning, conformance checking, right-time analysis and mitigation, and dynamic VR training. Figure 1 illustrates the information flow between various components and how active VR training using VirtualSafety can be integrated with the DT. By basing the VR environment on project status knowledge (as-built and as-performed), it is possible to generate learning scenarios based on real-life events for the given project. Simulating the safety status enables lessons learned to be transferred back to the digital twin and optimize current safety plans. Each component of the DTCS is briefly described in the following subsections.

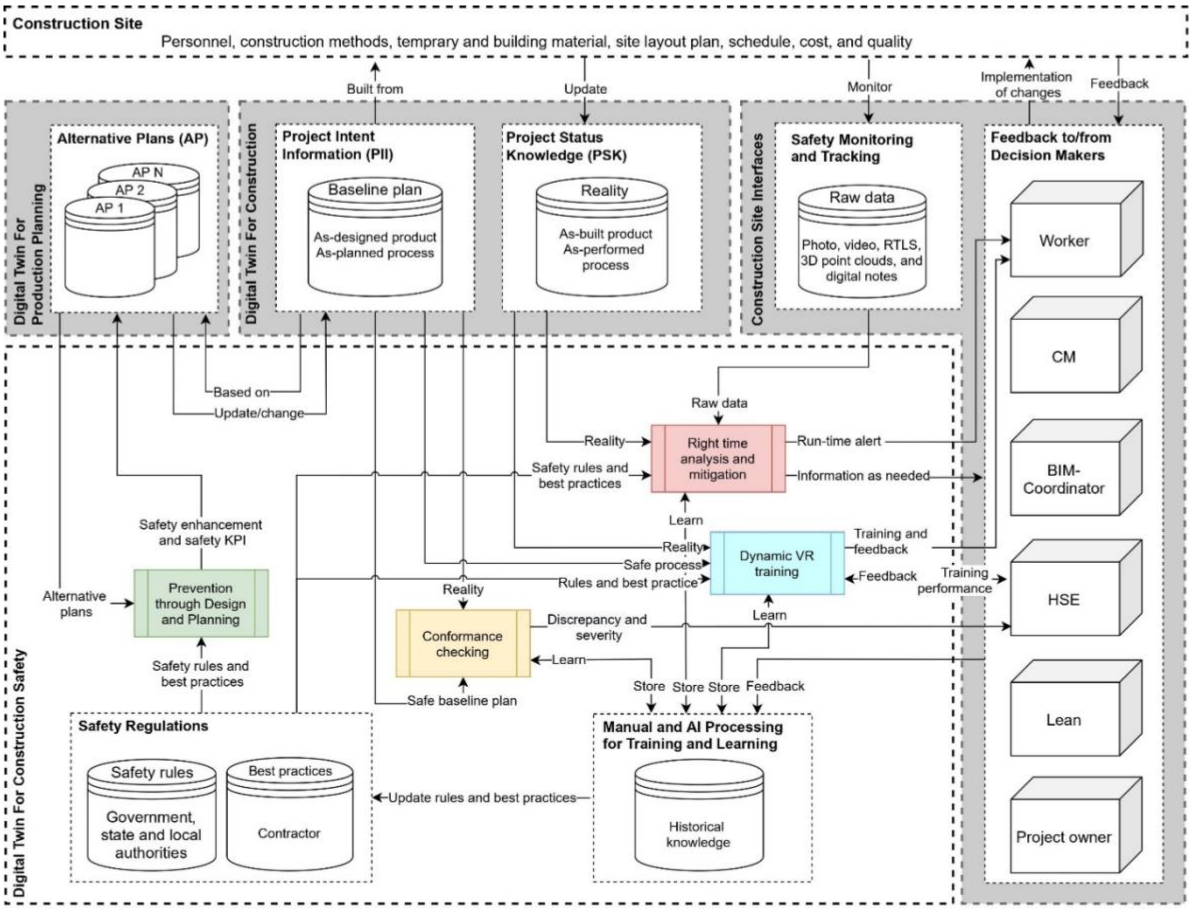

Figure 1 – The architecture of the Digital Twin for Construction Safety (DTCS) and interaction between various components; the dynamically updated VR training module is presented in blue

2.1 Prevention through design and planning

This component is termed SafeConAI (D4.1 and D4.2) in COGITO System Architecture v2. Alternative construction plans are handed to the prevention through the design and planning module (shown in green) of the DTCS and enhanced with safety, based on the safety regulation of the construction site. The system analyses the hazard spaces identified in the design and those identified in the process (e.g., crews working simultaneously on different stories, creating hazard zones in terms of being struck by an object from above). The safe alternative plans are returned to the Digital Twin for Production Planning (DTPP) for decision-makers' (stakeholders of the project) selection, consequently updating the baseline plan from which the construction site is built.

2.2 Conformance checking

The conformance checking module (shown in yellow) should find and classify discrepancies, including severity level, between the plan and reality. This information should be stored, and when the HSE expert has visited the problem, they can provide new information on the correctness of the output (in terms of both incident classification and its severity). This information provided by the HSE should be used to improve the classification of future occurrences and to update the best practice. An example of an updated best practice could be to use a safety net in some situations to avoid the repeated removal of a guardrail.

2.3 Right-time analysis and mitigation

The right-time analysis and mitigation module (shown in red) perform complex event processing and classification based on the reality of the construction site, the raw safety monitoring data, historical knowledge, and safety regulation. Then the workers are alerted to prevent both accidents (i.e., fatalities, serious injury, and minor injury) and incidents (i.e., close calls and unsafe acts) before they occur. The module subsequently performs an accident investigation, where the root cause of the incident or accident can be determined and prevented in the future. Besides, the feedback to and from the decision-makers is stored and used in processing/classification- and-investigation-mechanisms in this module. These are updated and used in the prevention through the design and planning module (i.e., SafeConAI), conceptually closing the loop of the digital twin for construction safety.

2.4 Dynamic Virtual Reality training

The dynamic VR training module (represented in blue) includes updating VR games and periodic training of workers using the latest information from the DTCS. This module is termed VirtualSafety (D4.7 and D4.8) in COGITO System Architecture v2. By feeding the model from SafeConAI enriched with the accident type and the area, the smart VR environment can produce relevant scenarios for each incident fed into the system. This will be done based on the location and type of hazard. Both colliders and data collection capabilities will be automatically created from this scenario before a training session is conducted. The workforce and the digital twin mutually benefit from this. The worker receives training, and the digital twin receives knowledge such as the safety performance of an individual and a group of workers that can be utilized when other prevention approaches are used.

3 Dynamic Virtual Reality training with VirtualSafety

3.1.1 Overview of VirtualSafety

A construction site is inherently complex, and the participating agents are presented with frequent unprecedented challenges. This is one of the main reasons for high accident rates, which involves even experienced workers. The accidents can be reduced greatly if the workforce is trained in a work environment similar to what they face in real life. The VirtualSafety application trains the workers with VR serious games consisting of dynamic content representing changes in the construction site. These dynamic VR games present the workforce with training environments created from the digital twin of their construction site. The training scenarios are periodically updated with the progress of the work. Therefore, the labours can perform the tasks, make mistakes, and learn without fearing the possibility of injury.

The framework for developing and implementing of the dynamically updated VR training games in the VirtualSafety application is illustrated in Figure 2. The first set of training scenarios is created with information from various digital twin databases such as historical knowledge, safety regulations, and a predefined object library. Then the training environments are periodically updated based on the information streams from the digital twin.

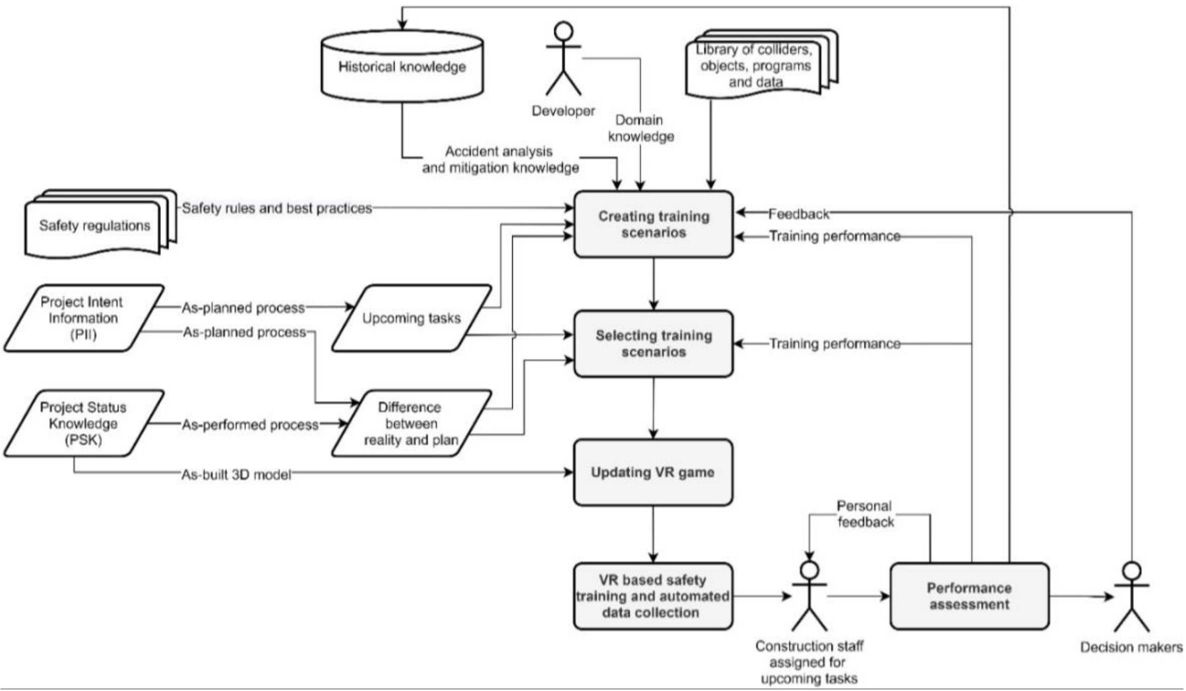

Figure 2 – The framework for dynamic VR training using VirtualSafety

3.1.2 Creating and selecting initial training scenarios

Creating safety training scenarios requires geometrical information, hazardous zones, and objects for interaction. The initial geometry of the training environment is created based on as-designed 3D models and training tasks based on as planned process from the Project Intent Information (PII). The geometrical information is later updated based on as-built products and training schemes based on as-performed processes from Project Status Knowledge (PSK). If there is any difference between reality and plan, the geometry and training tasks in the VR game are modified accordingly. Once the initial set of training scenarios is generated, all scenarios relevant to the upcoming tasks and work environment are selected from them for training. An example of creating and selecting training scenarios is shown in Figure 3. Here, the geometry of the building is created based on PII and PSK. The upcoming task is plastering as inferred from the project completion status. Therefore, the workers will be trained to perform plastering in the VR environment. All training scenarios relevant for plastering work are selected for updating the VR training game.

Figure 3 – An example of a training scenario for construction safety involves missing or inadequate guard rails at an elevated workstation

The potential hazardous zones are generated using the historical safety database and a library of predefined standard safety solutions. This library contains objects, colliders, programs, and data related to probable accident mitigation methods. The objects are created to interact with the training scenarios and rectify the hazards. For example, a guard rail is used at a leading edge or a cover over a hole. This interaction is enabled by creating object colliders that can be used to 'snatch' objects to a location. This information needs to be created automatically for updating the VR games.

Consider Figure 3 for determining hazardous zones. This is a leading-edge scenario where a guardrail is missing. The scenario task is to locate the hazard and choose the right rectification strategy. According to safety regulations and historical knowledge, the leading edges should be adequately secured to avoid falling from height. The current scenario task is rectifying the hazard by placing a guardrail. A specialized developer manually creates these scenarios. As new hazards can be introduced to the system using the information from the digital twin, the development is a continuous process that will be iterated throughout the project.

3.1.3 VR based safety training and automated data collection

Once the VR game is updated with all scenarios relevant to the upcoming tasks, the labours can be trained before their work. Each training task is designed according to safety regulations to inculcate awareness and teach best practices to the workforce. The serious games are updated with potential hazard zones, construction equipment, and relevant objects for interaction. Colliders are created at locations where possible interaction results in close calls or accidents. These colliders are programmed to collect safety violation data automatically. Therefore, the VR training can provide personalized feedback to the participants. Besides, the possible accident zones can be identified before starting the work.

In the example illustrated in Figure 3, the leading-edge and the virtual avatars of the workers are assigned with safety envelopes. The safety envelops are colliders of a specific size specified by safety regulations and historical knowledge. Whenever the workers move close to the leading edge, the safety envelopes interact. Then data related to these interactions such as time, duration, proximity, and the number of violations are automatically recorded. This data is used for performance assessment. The training scenarios in the game are updated based on the PSK. Therefore, the accident zones such as unguarded leading edges and cluttered workplaces can be identified during VR training and rectified before the commencement of actual work.

3.1.4 Performance assessment

The data relating to the performance of the players in terms of task execution, completion and safety violations are recorded during training. Automated assessment of the performance of individuals or groups is readily available after completing the training. The decision-makers and instructors can provide feedback to improve the performance. These trained workforce tend to have more situational awareness than those trained in an unrelated work environment without personalized feedback (Solberg et al., 2020). Examples of data collected from the training scenario in Figure 3: the number of close-call events, such as workers moving closer to the leading-edge, and time taken by the workers to detect the hazard, rectify the hazard and complete the task. Each worker can improve their performance in detecting the hazard or correcting it based on personalised feedback.

Another example of a training task is instructing the worker to detect unsafe conditions or practices around them and asking them to take the right course of action based on available options. Consider the scenario illustrated in Figure 4. The left part of the figure shows the unsafe scenario presented to the worker. The right part is when the work fixes those unsafe conditions by selecting the right choices. The data corresponds to the time and accuracy of corrective actions that can be used for estimating the safety performance of the workforce.

Figure 4 – Example scenario for detecting and rectifying unsafe conditions. a) Unsafe training scene involving a leading edge without barricades and a worker without PPE, b) Changes in the scene when the trainee selects correct actions

3.1.5 Creating and selecting new training scenarios and updating VR games

After the initial set of training, VR games are updated based on the performance of the labours and feedback from the decision-makers. If most players face difficulty performing certain tasks, the training scenarios need to be evaluated. Suppose there is an undetected hazard in the training scenario, and all participants interact with it. In that case, it needs to be included, even if it is not part of the upcoming task (for example, a cluttered workplace that may cause slips or trips). If the labours are unfamiliar with the available safety solutions or are underperforming, they should be further trained to achieve the required performance. New safety solutions should be introduced from the library whenever necessary (If guard rails in the leading-edge are being removed frequently, introduce safety nets). The new safety solutions can be generated based on the historical knowledge database. Besides, the newly created safety knowledge from the VR training is added to the knowledge database.

Alternate training scenarios should be introduced, or new training scenarios must be created if the existing scenarios are inadequate (The workers in Figure 3 should wear safety lanyards if the barricading does not provide enough protection). The decision-makers should be informed of any significant safety issues identified in the as-designed model during VR training (for example, issues in the site layout or logistics that may pose frequent and unintended interaction between workers and moving equipment). The VR environments should be updated after rectifying the issues on the construction site.

As the construction work progresses, the VR environment must be updated with the as-built 3D model from the digital twin. The training scenarios are either created or selected from the existing scenarios based on the difference between reality and plan. Therefore, the labours are always trained with the latest VR game before their work. Consider the example of updating in the VR environment with an as-built 3D model that contains unguarded leading edges. Here the program would look through the list of training scenarios and find the scenario linked to the potential incident. The encountered scenario is then placed on top of the as-built model as intractable objects, such as guardrails. The scenarios will also be created from the information obtained from the as-planned process, to ensure workers are trained in upcoming environments. The VR training games are dynamically updated until the completion of the project.

3.2 Technology Stack and Implementation Tools

The VirtualSafety application is developed from scratch in the game engine software Unity through C# programming language. The principal technologies such as hardware and software components used in this deliverable are given in Table 1. SteamVR software is essential for establishing a room-scale VR training environment. The SteamVR plugin to Unity supports data transfer and game development. The OpenVR XR plugin provides OpenVR rendering to Unity XR and enables rendering on all major VR devices through one interface. The Microsoft Visual Studio provides a platform for generating and editing scripts in C#.

The main hardware components for experiencing VR consist of a Head-Mounted Display (HMD), a pair of controllers, and a pair of base stations. The VirtualSafety application is currently implemented and tested through HTC Vive HMD and associated components (Figure 5). The user can view and hear in the virtual environment through the HMD and interact using the controllers. The base stations are mounted on the opposite corners of the training room to track the exact locations of the HMD and the controllers. Additional Vive trackers are used to track some specific body parts of the trainee or some other game objects in the virtual environment.

Table 1 – Libraries and Technologies used in Interactive Visual Material for Workforce Training v1

Library/Technology Name	Version	License
Unity	2020.3.6 f1	Software licensing
SteamVR	1.22.12	Software licensing
SteamVR Plugin	2.7.3	-
OpenVR XR Plugin	1.1.4	-
Microsoft Visual Studio 2017	15.9.27	Visual Studio Community
Manus Core	SDK 1.9.0	Proprietary
Manus Dashboard	1.0.0	-
Unity Plugin for Manus Polygon	1.9.0	-
HTC Vive Head Mounted Display (HMD)	Vive Pro	-
HTC Vive Controllers	SteamVR Tracking 2.0	-
SteamVR Base stations	2.0	-
Vive Trackers	3.0	-
Manus VR gloves	Prime X Haptic VR	-

Various training scenarios were developed to test the different components of VirtualSafety, such as runtime data collection, data analysis, and performance assessment. The training scenarios developed in this reporting period involves various additional components such as VR data glove and trackers for advanced data collection and analysis (Figure 6). The Prime X Haptic VR data glove (MANUS, n.d.) is used for finger tracking. This data glove contains a flex sensor skeleton to capture the bending of the fingers. It also includes an Inertial Measurement Unit (IMU) with nine DOF (Degree of Freedom) per finger to capture relative movements. For example, the data handling software Manus Core enables the collection and visualization of the data from the data glove. This software also helps to live stream the finger movement data to Unity. The VIVE Tracker (3.0) (VIVE United States, n.d.) is mounted on the data glove to track hand movements. The tracker enables seamless coordination between the real hand and its virtual counterpart in the VR environment. The tracking data is collected and streamed to Unity through the SteamVR application (Corporation, n.d.).

Figure 5 – HTC Vive Head Mounted Display (HMD), controllers and base stations

Figure 6 – A user wearing the data glove and tracker ready for gesture recognition

3.3 Summary of the Deliverable and An Application Example

This section summarizes of the work done between M10 to M19 for developing VirtualSafety and a detailed example of developing one training scenario involving hand gesture recognition and analysis in VR. The work in the early phase included creating VR training scenarios of everyday construction tasks in Unity. These scenarios have been further enhanced with automated data collection methods. The game objects of the participant (or trainee) have been assigned with C# scripts to collect the data corresponding to their movements and actions along with timestamps. The trainee's interaction with the surrounding objects, safety behavior, and task completion status are also recorded. The safety behaviour is assessed through various parameters such as the use of Personal Protective Equipment (PPE), the number of hazards identified, time taken for hazard identification and rectification. Once a participant completes training, the generated behavioural data is independently analysed to give personalized feedback. The automated data collection of trainees' performances is successfully tested in preliminary experiments. The next step is automatically analysing the behavioural data. Inbuilt programming capabilities of the game engine software are explored for this purpose.

The construction site is a collaborative work environment, and this can be simulated in VR through multiplayer serious games. A training scenario involving crane rigging is created to assess the safety challenges associated with communication between the crane operator and signaller. Hand gestures are widely used for communication on construction sites. They enable effective communication irrespective of the construction noises and language barrier between the workers. There are standard hand signals for operations such as crane rigging. However, misinterpretation of these predefined hand signals by the cranes operator or any other machines may cause severe accidents. Therefore, automatic recognition and interpretation of the hand gestures may potentially assist in effective communication between the signaller and the operator. It improves communication in VR training scenarios involving multiple workers, similar to the actual construction site. Automated gesture recognition in VR coupled with the collection of trainees' behavioural data can quantitatively estimate the effectiveness of VR-based training methods.

An automatic gesture recognition method has been developed to identify hand signals for crane rigging operations in VR training. The developed gesture recognition algorithm tracks information from a data glove and a tracker. Preliminary data on movements and orientation of hands and fingers were recorded. Mathematical models of hand gestures were created based on finger movement data. Gesture rules were created based on the rotation and orientation of the hand. The gesture models were combined with the gesture rules to develop the algorithm for automated gesture recognition.

3.3.1 Gesture Interactions and Recognition in Virtual Reality

Data-driven methods such as deep learning have been widely used for gesture recognition. However, the existing methods have several drawbacks. These methods often require large datasets for training that may not be available for newly created VR training scenarios. The performance of computer vision-based methods depends on environmental factors such as light, skin colour, and occlusion. The methods require several image processing techniques, which might affect the recognition accuracy. The computer vision-based method may not capture the signals correctly in crane rigging operations unless multiple cameras are installed at different levels. Autonomous cranes might depend on the ground-based human operator who wears intelligent gloves.

The interactions with VR controllers often lack the realistic experience during construction safety training. Besides, the nuances of finger movements and seamless coordination between real and virtual avatars are essential in collaborative training environments (Moelmen et al., 2021). Therefore, data gloves and trackers were introduced to enhance the interactive and realistic experience in VR training. Automatic gesture recognition during the training potentially improves the communication between the trainees. Thus, a gesture recognition method for VR training has been developed. A training scenario involving a signaller and a crane operator is envisioned where the communication has been enhanced by automatic gesture recognition. A gesture recognition algorithm has been developed based on the information streams from data gloves and trackers. The proposed method identifies and displays the gestures in real-time during the VR training.

The overall methodology for gesture recognition in VR consists of experiments and data analytics. First preliminary experiments were conducted to collect various data such as finger movement and hand orientation required for gesture recognition. Then, mathematical models of the hand gestures were created based on finger movement data. These gesture models were combined with the orientation data to develop the algorithm for automated gesture recognition. After that, experiments were designed to capture the efficacy of the proposed method in automatically recognizing crane rigging signals. Next, experiments were conducted to collect the data in run-time and implement the proposed method. The performance is evaluated by comparing the predicted hand gestures with independently created ground truth labels.

3.3.1.1 Development of Gesture Recognition Algorithm

The virtual reality experiments were conducted in two stages. The preliminary experiments in the first stage are for developing the gesture recognition algorithm. The VR experiments in the second stage are for validating the gesture recognition method. The user wears a data glove and tracker while making the specified gestures in both cases. The data corresponding to hand and finger movements were collected in real-time during the experiments. Figure 6 shows the user wearing the data glove and tracker, ready for the experiment.

A typical human hand consists of a wrist, a palm and five fingers. The hand is composed of 27 bones. The bones are categorized as 1) carpals in the wrist [short bones, 8 no.], 2) metacarpals in the palm [long bones, 5 no.], and 3) phalanges in fingers [long finger bones, 14 no.]. Each of the four fingers except the thumb is composed of a proximal phalange, intermediate phalange, and distal phalange. The thumb has only two phalanges, the proximal phalange, and the distal phalange. The placement of the metacarpal bone of the thumb enables its distal phalanges to oppose the distal phalanges of other fingers. This configuration of the thumb allows humans to grab objects in hand.

The joints in the hand facilitate various movements of fingers, as illustrated in Figure 7. Hinge joints prove 1 DOF (Degree of Freedom), i.e. flexion or extension. Saddle joints provide 2 DOF, i.e., flexion or extension and abduction or adduction. The metacarpophalangeal (MCP) is a saddle joint, whereas proximal interphalangeal (PIP), distal interphalangeal (DIP), and interphalangeal (IP) are hinge joints. In addition to carpometacarpal (CMC) joints, the thumb has two joints (MCP and IP) and all other fingers have three joints (MCP, PIP, and DIP).

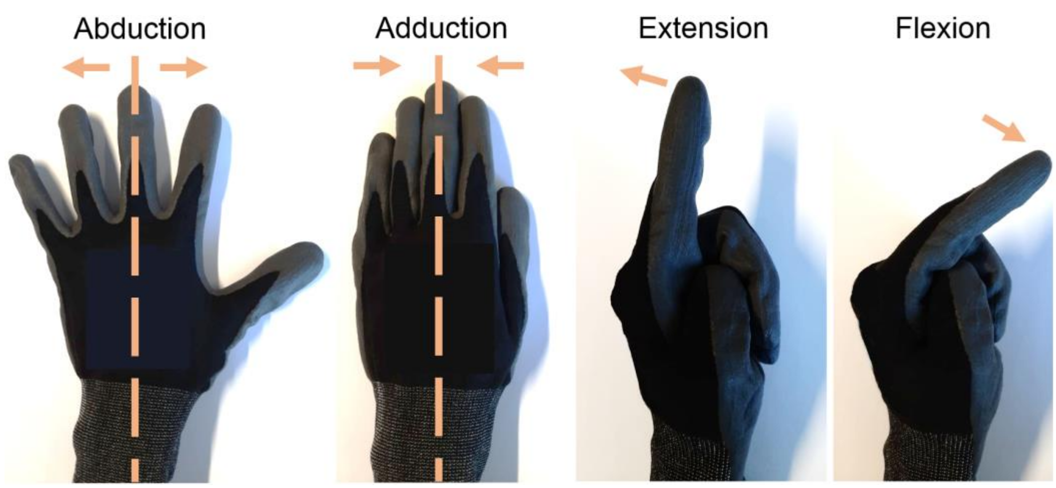

Figure 7 Motions of human fingers of a construction worker wearing protective gloves.

An overview of developing the gesture recognition algorithm is illustrated in Figure 8. It starts with preliminary experiments containing five selected hand signals for crane rigging. The experiments were conducted to understand the nature of hand gesture data for developing the recognition algorithm. The user makes the selected gestures in the experiment wearing a data glove and a tracker. Each set of the experiment contains one hand gesture. The current study has selected five crane rigging signals as shown in Figure 9: 1) stop, 2) raise boom, 3) lower boom, 4) hoist load, and 5) lower load. These hand signals are dynamic, i.e., they involve movements of the hand along with hand gestures. The proposed method is designed to recognize hand gestures in a static position (static hand signals). However, continuous real-time identification of gestures enables dynamic gesture recognition.

The finger data is live streamed to the Manus Core and visualized in the Manus dashboard. The data viewer displays values of flexion, extension, abduction, adduction, thumb rotation, and wrist rotation. The flexion and extension are estimated with respect to the finger joints, whereas abduction and adduction are estimated with respect to the midline of the hand and a finger joint. The finger data is simultaneously streamed from the Manus Core to Unity. Currently, the gesture models were created in Unity, as described in the next paragraph.

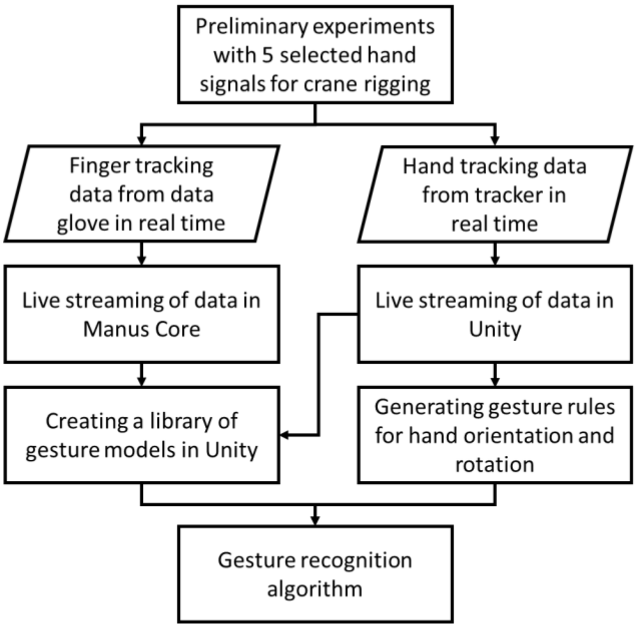

Figure 8 – Development of gesture recognition algorithm

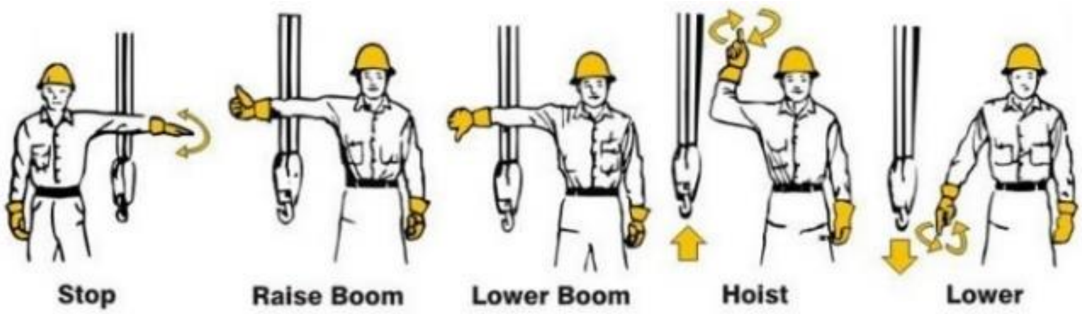

Figure 9 – Selected crane rigging signals for this study (29 CFR Part 1926 Cranes and Derricks in Construction; Final Rule, 2010)

Three hand gestures were selected to represent the finger positions in the five crane rigging signals. The selected hand gestures are: 1) 'thumbs up' (only thumb is straight, all other fingers are flexed), 2) 'pointing' (only index finger is straight, all other fingers are flexed), and 'high five' (all fingers are straight). Note that only finger movements can be tracked with the data glove. Additional information about the hand movement is required for identifying the crane rigging signals. Mathematical models were created for the three selected gestures in Unity. The mathematical model of a gesture defines the movement of each finger in a relative scale with respect to the finger joints and/or midline of the hand. Flexion and extension of a thumb are specified based on CMC, MCP, and IP; and that of other fingers based on MCP, PIP, and DIP. Abduction and adduction of a thumb are specified based on CMC, whereas that for other fingers is based on MCP. Thus, a library of predefined gesture models was created in Unity. Currently, the library contains three gesture models, each of representing the finger positions of the selected hand signals, as shown in Table 2.

Table 2 – Library on gesture models

Label	Hand signal	Gesture model	Need more information for recognition?
0	No recognition	-	Yes
1	Stop	HighFive	Yes
2	Raise boom	ThumbsUp	Yes
3	Lower boom	ThumbsUp	Yes
4	Hoist load	Pointing	Yes
5	Lower load	Pointing	Yes

After creating the gesture library, gesture rules were generated from the hand tracking data. The movements of the hand are tracked in real-time through the VIVE tracker. The real hand will appear as a game object in Unity and the same object will be seen by the users in the virtual environment. The game object of the hand is hereafter referred to as the hand object. The hand object has an attribute called 'transform' that contains the object's position, rotation, and scale in the virtual environment. The current study uses the transformation of the hand object to create rules for recognizing hand gestures.

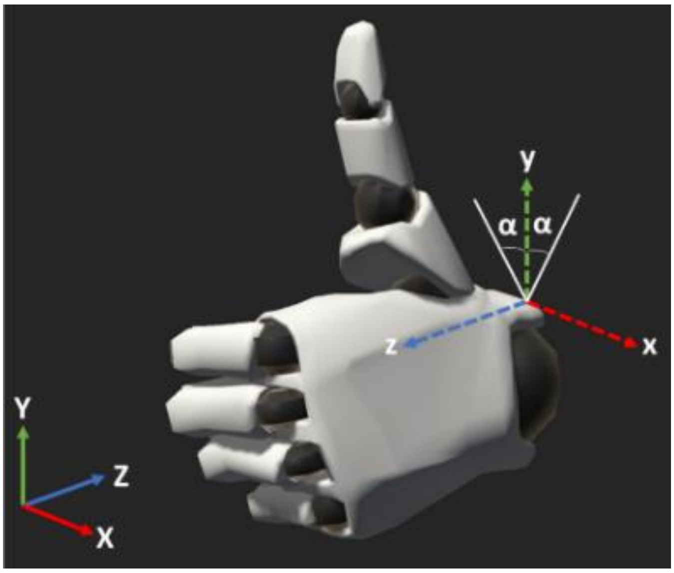

Figure 10 – Schematic of the Gesture rule for raise boom hand signal

Consider the example of the hand signals for 'raise boom' and 'lower boom'. Both of these hand signals have the same gesture model (ThumbsUp) to represent their finger positions. Therefore, rotation or orientation of the hand in the virtual environment is essential to distinguish between these hand signals. Thus, the gesture rules for recognizing these hand signals involve a specific range of values for these parameters from the viewer's perspective. Figure 10 shows a schematic representation of one of the gesture rules for the raise boom hand signal. Here, the solid arrow marks represent the global coordinate system (X, Y, Z) for the virtual environment and dotted arrow marks represent the location coordinate system (x, y, z) for the hand object. The tolerance of rotation of the hand object about the y axis is denoted by α. The gesture rule in this scenario is: if $\alpha <= 30^0$ for the gesture model 'ThumbsUp', the hand gesture is 'raise boom'. Note that this is a simplified illustration of a gesture rule. Orientation and rotation of the hand object with respect to all other axes will be specified in the actual gesture rule. The gesture recognition algorithm is developed by combining the gesture rules and gesture models. A detailed description of the gesture recognition method is given in the next section.

3.3.1.2 Gesture Recognition Method

The automated gesture recognition method developed for VirtualSafety is shown in Figure 11. First, the algorithm is implemented in actual virtual reality experiments containing various hand gestures. The raw hand movement data collected by the tracker and the finger movement data from the data glove are live streamed into Unity. The gesture recognition algorithm is attached as a script to the hand object in Unity. The recognition algorithm runs in a fixed interval to accurately capture the hand object's physics movements. The gesture data from the user is evaluated in each run. First, the gesture library is searched to see if the current finger movement data matches any predefined gesture models. If none of the gesture models matches the current gesture data, display 'No recognition' and proceed to the next frame. If any of the gesture models match with the current finger movement data, check the associated gesture rules. The orientation and rotation of the hand object are estimated, and the gesture rules are evaluated. If none of the rules is satisfied, display 'No recognition' and proceed to the next frame. Otherwise, determine the hand signal based on the gesture rules satisfied. Then display the identified hand signal and check whether the experiment is completed. If the experiment is complete, stop the iteration. Otherwise, proceed to the next frame.

3.3.1.3 Validating Gesture Recognition Method

The proposed gesture recognition method (Figure 11) is validated by virtual reality experiments where the user acts as a signaller for a crane operator. In the experiments, the user wearing the data glove and the tracker make various hand gestures for a specific interval. The gestures involve hand signals for crane rigging operations and some random gestures. The gesture data are collected from the data glove and trackers in real-time during the experiments. Simultaneously the collected data is analysed using the proposed method and results were also displayed in real-time. The display contains the predicted hand gesture, an associated colour and time stamps as shown in Figure 12. The predictions are also logged in a text file (.txt) as entries in the format "time (in seconds), predicted gesture, label"; e.g., "72.82 RaiseBoom 2". The entire experiment is recorded in the game view using the Unity recorder. The recorded videos of the experiments with time stamps were used to create ground truth labels for the gestures. The experiment was repeated five times. The accuracy of the recognition method is estimated as an average of all the repetitions.

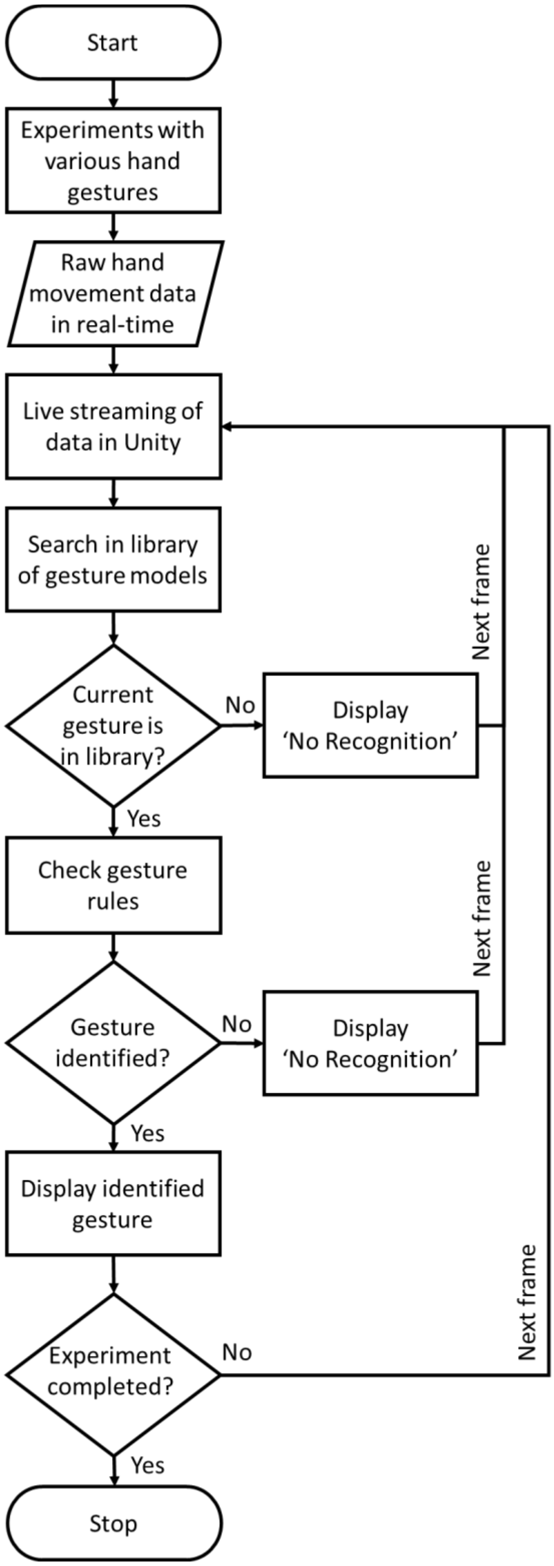

Figure 11 – Gesture recognition method

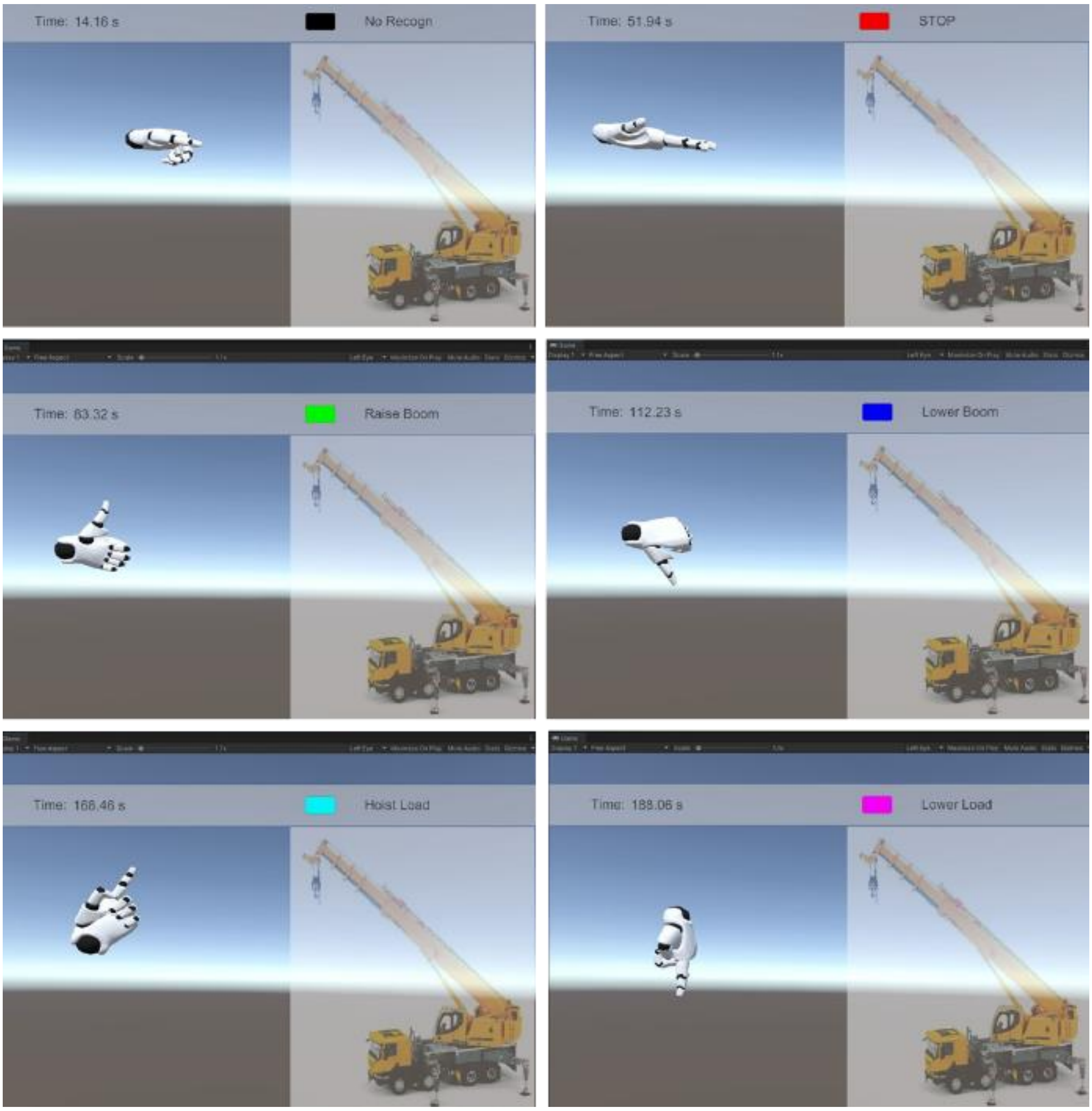

Figure 12 – Conveying hand signals to the crane operator in virtual reality. The real-time predictions by the recognition method for each hand signal are displayed with the time stamp

3.4 Licensing

The VirtualSafety application is offered in the form of a closed-source software component.

3.5 Installation Instructions

The VisualSafety application is available as a web-based application, thus, no installation is required.

3.6 Development and integration status

The current training scenarios in the serious games for the VirtualSafety application are developed for a single participant. The existing scenarios are being enhanced with more complex tasks involving multiple participants.

The application components for runtime data collection, analysis, and personalized feedback were independently developed and tested. These components are being integrated into a single training scene. The next step is updating the summary of the training performance and personalized feedback to the DTP for the decision-makers.

3.7 Assumptions and Restrictions

The VirtualSafety application is being developed for desktop computers, and the scenarios in the serious games are created with the intention of training in a room-scale training facility. The communication with the DTP and the rest of the COGITO components needs to be tested and the application must be modified based on the requirements.

4 Conclusions

This deliverable presents the functional components of the VirtualSafety application and their use for workforce training. The information related to tools and technology for the development of this application~~,~~ and an overview of data exchange and integration with the DTP are described.

The VirtualSafety application trains the workforce with VR games comprised of dynamic content representing changes in the construction site. These dynamic VR games present the workforce with training environments that are created from the digital twin of their construction site. The training scenarios are periodically updated with the progress of the work. Therefore, the labours can perform the tasks, make mistakes, and learn without fearing the possibility of injury. By feeding the model from SafeConAI enriched with the accident type and the area, the smart VR environment can produce relevant scenarios for each incident fed into the system. This will be done based on the location and type of hazard. Both colliders and data collection capabilities will be automatically created from this scenario before a training session is conducted. The workforce and the digital twin mutually benefit from this. The worker receives training, and the digital twin receives knowledge such as the safety performance of an individual and a group of workers that can be utilised when other prevention approaches are used.

References

Bureau of Labor Statistics. (2020). *National Census of Fatal Occupational Injuries in 2019*. www.bls.gov/iif

Corporation, V. (n.d.). *SteamVR*. Retrieved February 25, 2022, from https://www.steamvr.com/en/

Gao, Y., Gonzalez, V. A., & Yiu, T. W. (2019). The effectiveness of traditional tools and computer-aided technologies for health and safety training in the construction sector: A systematic review. *Computers & Education*, *138*, 101–115. https://doi.org/10.1016/J.COMPEDU.2019.05.003

Golovina, O., Kazanci, C., Teizer, J., & König, M. (2019). Using serious games in virtual reality for automated close call and contact collision analysis in construction safety. *36th International Symposium on Automation and Robotics in Construction (ISARC 2019)*, 967–974. https://doi.org/10.22260/isarc2019/0129

Harichandran, A., Johansen, K. W., Jacobsen, E. L., & Teizer, J. (2021). A Conceptual Framework for Construction Safety Training using Dynamic Virtual Reality Games and Digital Twins. *38th International Symposium on Automation and Robotics in Construction (ISARC 2021)*, 621–628. https://doi.org/https://doi.org/10.22260/ISARC2021/0084

Hou, L., Wu, S., Zhang, G. K., Tan, Y., & Wang, X. (2021). Literature review of digital twins applications in construction workforce safety. *Applied Sciences (Switzerland)*, *11*(1), 1–21. https://doi.org/10.3390/APP11010339

Jacobsen, E. L., Strange, N. S., & Teizer, J. (2021). Lean Construction in a Serious Game Using a Multiplayer Virtual Reality Environment. *29th Annual Conference of the International Group for Lean Construction (IGLC)*, 55–64.

MANUS. (n.d.). *Prime X Haptic VR*. Retrieved February 23, 2022, from https://www.manus-meta.com/haptic-gloves

Moelmen, I., Grim, H. L., Jacobsen, E. L., & Teizer, J. (2021). Asymmetrical Multiplayer Serious Game and Vibrotactile Haptic Feedback for Safety in Virtual Reality to Demonstrate Construction Worker Exposure to Overhead Crane Loads. *38th International Symposium on Automation and Robotics in Construction (ISARC 2021)*, 613–620. https://doi.org/https://doi.org/10.22260/ISARC2021/0083

29 CFR Part 1926 Cranes and Derricks in Construction; Final Rule, (2010). www.osha.gov

Solberg, A., Hognestad, J., Golovina, O., & Teizer, J. (2020). Active Personalized Training of Construction Safety using Run Time Data Collection in Virtual Reality. *20th International Conference on Construction Applications of Virtual Reality (CONVR)*, 19–30.

Tao, F., Zhang, M., & Nee, A. Y. C. (2019). Digital Twin and Virtual Reality and Augmented Reality/Mixed Reality. In *Digital Twin Driven Smart Manufacturing* (pp. 219–241). Academic Press. https://doi.org/10.1016/B978-0-12-817630-6.00011-4

Teizer, J., Johansen, K. W., & Schultz, C. (2022). The Concept of Digital Twin for Construction Safety. *Construction Research Congress 2022*, 1156–1165. https://doi.org/10.1061/9780784483961.121

VIVE United States. (n.d.). *VIVE Tracker (3.0)*. Retrieved February 23, 2022, from https://www.vive.com/us/accessory/tracker3/

COGITO

CONSTRUCTION PHASE
DIGITAL TWIN MODEL

cogito-project.eu

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 958310