Real-time Simulation of Construction Workers using Combined Human Body and Hand Tracking for Robotic Construction Worker System

Manoj Kurien [a], Min-Koo Kim [b,*] Marianna Kopsida [a], and Ioannis Brilakis [a]

[a] *Department of Engineering, University of Cambridge, Trumpington Street, Cambridge, United Kingdom*

[b] *Department of Building and Real Estate, The Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong*

Abstract

Construction is an inherently less safe sector than other sectors because it exposes workers to harsh and dangerous working environments. The nature of the construction industry results in a comparatively high incidence of serious injuries and death caused by falls from a height, musculoskeletal disorders and being struck by objects. This paper presents a new concept that can tackle this problem in the future. The central hypothesis of this study is that it is possible to eliminate injuries if we move the human construction worker off-site and remotely link his/her motions to a Robotic Construction Worker (RCW) on-site. As a first steppingstone towards this ultimate goal, two systems essential for the RCW were developed in this study. First, a novel system that combines 3D body and hand position tracking was developed to capture the movements of human construction worker. This combination of tracking enables the capture of changes in the orientations and articulations of the entire human body. Second, a real-time simulation system that connects a human construction worker off-site to a virtual RCW was developed to demonstrate the proposed concept in a variety of construction scenarios. The simulation results demonstrate the future viability of the RCW concept and indicate the

---

[*] Corresponding author. Tel.: +852 2766 5819 Fax: +852 2764 5131
E-mail address: minkoo.kim@polyu.edu.hk

promise of this system for eliminating the health and safety risks faced by human construction workers.

Key words: Construction site safety, Construction worker, human body tracking, hand tracking, construction simulation, Robotic Construction Worker (RCW), Robotics

1. Introduction

The construction industry is one of the largest industries in both developed and developing countries. Employing two million people in the UK, it is the country's biggest employing industry. Unfortunately, it is also well known that construction is an inherently less safe sector than other sectors because it exposes workers to harsh and dangerous working environments. This nature of the construction industry results in a comparatively high incidence of serious injuries and death. This safety handicap is also one of key reasons behind the lack of construction workforce. According to the Health and Safety Executive (HSE) UK [1], deaths and serious injuries amongst construction workers are unacceptably high and more frequent than in any other sectors of the UK economy. In 2014-15 [2], 35 construction workers were fatally injured and a further 65,000 suffered a major injury at work in the UK, and the fatal injury and work-related illnesses rates are over 3.5 times and 20% than the average rate across all industries. One immediate impact of this high rate of work place injury and illness is cost to business. The total economic cost of workplace injury and ill health in the construction sector in 2013-14 was reported as £0.9 billion [2]. In a similar manner to the UK construction industry, more than 26,000 U.S. construction workers have died at work over the past two decades [3]. As these statistics indicate, safety in construction remains a major problem which needs to be fundamentally resolved.

The causes of the safety problems of construction workers arising from construction activities are varied. For non-fatal injuries occurred in the UK in 2015 [2], about 80% were due to falls from a height, trip falls, lifting/handling or being struck by an object. For fatal injury cases, falls from a height accounted for nearly 50% of cases. Considering workplace illnesses, about 65% of cases were due to musculoskeletal disorders (MSDs). To act on this issue, the Health and Safety at Work etc. Act 1974 [4] imposed a duty on employers to ensure the safety of workers. The HSE in the UK also has strict safety criteria and use the deterrent effect of prosecution to enforce the criteria, primarily focusing on aiding all sectors to improve compliance with the law through inspections and investigating accidents and complaints. In addition, activities such as awareness days, issuing guidance, and providing advice ensure that the regulatory measure encourages the industry to focus on long-term health and safety. However, this action by law leads to a significant expense (the HSE spent $111 million in 2002-03) and the effectiveness of this approach is unclear due to a difficulty in assessing the improvement.

Alongside the regulative efforts, there are other approaches aiming to increase the safety of construction workers. Technological advances in areas such as personal protective equipment (PPE), Building Information Modeling (BIM) and safety training have improved worker safety. For example, the protective gear worn by construction workers including helmets, steel-toed boots etc., helps reduce the impact of falls, trips and being struck by objects on the body [5]even though the PPE increases worker discomfort and is ineffective against MSDs. Some recent studies shows the potential that BIM can enable the automatic identification of construction safety issues [6-8]. In addition, as a means of safety training, involvement of the workers in the decision-making processes of evaluating workplace risks also helps identify and manage risks effectively since the workforce has direct experience of site conditions and they are most aware of potential hazards [9]. For this approach, the awareness and willingness of the workers and the managers should be required to address the risks in construction site.

Although the aforementioned regulative and technological efforts have positive impact on construction workers's safety, it is reported that the improvement rate has plateaued in recent years according to [10], indicating the pressing need to tackle the problem. This paper presents a robotics-based novel approach to not only minimise the health and safety hazards of human construction workers at construction but also increase the productivity in construction. The concept of 'Robotic Construction Worker' (RCW) which moves the human construction worker off-site and remotely links their motions to a RCW on-site is proposed. This approach aims to not only minimize the risk of MSDs and the risks associated with humans being present in a hazardous environment but also increase productivity. Two essential systems for the RCW were developed in this study as a first steppingstone towards this ultimate goal, which are (1) combined body and hand tracking system for the efficient and natural control of the humanoid robot and (2) simulation environment system to test and demonstrate the RCW system. First, a novel framework of combining vision-based hand tracking with body tracking was developed. This framework is integral for the RCW system in order to control both the hand and body of the robot naturally and simultaneously in a real-time and to implement detailed construction tasks which often require hand-based elaborate skill and cannot be achieved without accurate hand tracking of a construction worker. In this study, this framework was realised with coordinate mapping and development of a software pipeline to enable the tracking systems to run independently and simultaneously. Second, a simulation game engine was used to develop virtual construction sites and test the proposed RCW system. It is assumed in this study that a realistic and real-time simulation is vital to enable training, testing, planning and model-based control of the robotics. The rationale and details of the two systems are described in Sections 3 and 4.

The paper is organized as follows. In Section 2, a review of the state of research aiming to address the current problem of construction workers safety is presented. Section 3 describes the long-term solution of the RCW and the framework proposed in this study, followed by the research methodology and the derived results in Section 4. Finally, Section 5 summarizes this study with future research directions.

2. Related work

Research efforts in improving construction worker safety have mainly lied in three areas: (1) wearable sensing techniques, (2) computer vision techniques and (3) robotic techniques.

2.1.Wearable sensing techniques

The use of on-body wearable sensors is widespread in several academic and industrial domains. Accelerometers and IMUs are one of the most popularly used sensors used to track the motions of construction workers. Such sensors can measure velocity, acceleration, orientation, and gravitational forces, and the acceleration data can be used to monitor the physiological condition of a human body [11]. Fang and Dzeng [12] developed an accelerometer-based fall portent detection system that used a hierarchical threshold-based algorithm. Bakhshi et al. [13] proposed an approach for measuring and monitoring human body joint angles using inertial measurement unit (IMU) sensors. Jebelli et al. [14] proposed to use IMU sensors attached to the ankle to characterise the fall risk of workers. Valero et al. [15] also presented a wearable system that can measure the postures and body motions of workers using scalable IMUs with a low level of intrusiveness and real-time processing. Cheng et al. [16] proposed an approach for monitoring safe and unsafe behaviour of construction workers using data fusion of Ultra wideband and electrocardiography sensors. In addition, researchers have also successfully employed motion sensors to evaluate heart rate, respiratory rate and energy expenditure [17-18]. However, there is a limitation in the wearable sensing techniques that attaching accelerometers and IMUs to the human operator can affect negatively the accuracy of measuring the targeted signals since sensing accuracy is heavily dependent on whether the sensors are attached correctly in the correct positions.

2.2. Computer vision techniques

Recently, computer vision has gain attention because it can be used for automated and continuous monitoring of construction workers at construction sites. Seo et al. [19] identified that continuous monitoring of conditions and actions at the construction site is essential to eliminate potential hazards in a timely manner. Computer vision techniques enable an automated means of monitoring the site to overcome the current limitations of slow and unreliable manual inspection by safety managers. These techniques involving object recognition and object tracking can reduce the risks of being struck by an object or vehicle and of falling from heights. However, there are very challenging issues to this approach, including occlusion and the identification of good camera positions, due to continuously changing environments and diverse machinery and objects on-site. Furthermore continuous monitoring may also impact privacy negatively and reduce the motivation of construction workers.

Monitoring the individual construction worker's actions might prevent unsafe actions that lead to work place accidents. Han et al. [20] investigated the use of Microsoft Kinect to collect prior models of unsafe actions and then identify similar actions in site videos. They address safety at heights by extracting 3D skeletal models using the Kinect from videos of workers climbing ladders and evaluating their behaviour. The main drawback of this approach is that it is very difficult to form representative priors of unsafe actions due to the large motion ranges of human movement and the varied nature of construction site activities.

Ray et al. [21] focused on real-time construction worker posture analysis to improve ergonomics. Such techniques employ training and monitoring to reduce the risk of MSDs. They utilized the Microsoft Kinect to extract the worker's pose (body joint angles and spatial locations). Then, using a set of rules formulated using body posture information, the worker's activities were classified into two classes; ergonomic or non-ergonomic. The automated method can reduce the risk of MSDs and address other key issues such as risks involved in lifting/moving objects. The shortcomings of this approach are that they do not account for the factors of time, repetition, and forceful exertion. A particular pose can be safe for short periods of time, or with minimal force whereas the same pose can be unsafe with larger forces or with repetition.

Another limitation in existing computer vision techniques for tracking the motion of a construction worker is that accurate tracking of the worker's hands has not been settled yet. According to [22], factors such as the high dimensionality of the hand pose and the chromatically uniform appearance of the hand and self-occlusions during hand movements make tracking the hand a challenge. Specifically, tracking of the human hand using RGB-D data is a difficult problem due to the extensive degrees of freedom of the hand and fingers, their relatively small size and the obstruction and occlusion of the fingers during motions from a single viewpoint. Although some recent studies [23-24] employed Leap Motion sensors [25] as means of hand tracking, using the sensors prevents the movements of the arms, elbows and rotation of the upper body to enable accurate hand tracking, which is not suitable for tracking construction workers. In addition, even though Sharp et al. [26] developed techniques through a new pipeline for per-frame pose estimation, followed by a generative model-fitting stage to track the hand, their software is still under development and at present unavailable. Recently, the research group led by Prof. Argyros treats hand tracking as an optimization task of seeking hand model parameters that minimize the discrepancy between the 3D structure of a hypothesized hand model and the observed hand structure [22]. Their recent work has promising results in tracking the full hand articulations in real-time and seems that it can fill the gap in knowledge of combined hand and body tracking to monitor a construction workers pose, finally providing a complete solution to posture and worker's action analysis to prevent unsafe actions.

2.3. Robotic techniques

Automation with robotics is an approach that can increase productivity, reduce the risk of MSDs and minimize the risks associated with humans being present in a hazardous environment. One such robot is SAM [26] which is a semi-automated mason robotic bricklayer. A human mason can lay about 300 to 500 bricks a day, while SAM can lay about 800 to 1,200 bricks a day. Furthermore, it does not need breaks, sleep etc. giving it another advantage over manual labourers. Fundamentally however, an automated robot does not have the fine skills, adaptability and flexibility that human construction workers have and human workers would still be required on-site. For instance, SAM requires a human construction worker to tidy up the mortar and place bricks in difficult areas such as corners. The simplest construction task of automating bricklaying itself poses an immense challenge and leaves much to be desired.

One method to overcome the challenges of automated robots at a construction site is remote/ teleoperation of construction robots. The remote controlled trench compactor [28] reduces the need for repeated strenuous actions by workers in trenches, reducing the risk of MSDs, and in addition reducing the risk of injuries due to trench collapses. However, the machinery increases the risk of other hazards such as being hit by them.

Hironao et al. [29] investigated the teleoperation of a robotic system with the use of Virtual Reality (VR) technology as a possibility of performing remote operation with greater safety. They recreated the scene in VR and investigated the teleoperation of a robotic crane system. The limitation of this approach is that other construction workers would still be required on site to perform activities that the construction machinery is unable to complete, potentially increasing the risk to construction workers due to the hazards of teleoperation of heavy machinery. Moreover, the existing robotics only perform a small range of the tasks that comprise the activity at a construction site.

2.4. Gaps in knowledge

Although the current state of research aforementioned in Sections 2.1-2.3 can reduce the risks of fall from heights, MSDs, and being struck by objects, the previous approaches do not fundamentally eliminate the risks to the construction worker. There are two significant research gaps in knowledge with these aforementioned techniques: (1) a framework that tracks both the hand and body of a construction worker simultaneously is yet to be developed. The independent tracking of the body and of the hand has been developed but the methods to address worker safety are limited due to their incapacity to fully track human motion [29]; and (2) a robotic solution to address all construction operations, ranging from using fine tools to heavy machinery, has not been developed. Existing robotics (e.g. bricklayer robot, tele-operated crane and remote controlled trench compactor) can only carry out a very small portion of the tasks necessary at a construction site.

3. Proposed solution

3.1. Long term solution roadmap

In the long run, the ultimate goal of the proposed solution is to develop a humanoid robot that can mimic the precise motions of a human construction worker, addressing the problems and limitations highlighted in Section 2. Here we assumed that a humanoid type would be the most suitable for implementing the RCW system based on the following reasons. First, the construction site, equipment and tools on site are all optimized for humans to work on site. The activation energy of bringing a humanoid robot into an environment designed for humans is very minimal compared to a different robot which would require numerous changes to the construction site and construction process. For example, the aforementioned robot SAM, robotic bricklayer, is very useful to the certain task of laying bricks but can only do a single task. Second, since construction sites can also be very different and vary during the different stages and for different types of construction, a humanoid robot is much more adaptable to the changing requirements of a construction site. For these reasons, this study focuses on a humanoid robot.

A RCW would copy the motions of a human construction worker to carry out construction tasks. The human construction worker (off-site) can control the RCW (on-site), removing the workforce from the hazards of the construction and reducing the risks of MSDs, falls from a height and being struck by objects. Furthermore, the increased capabilities of robots would reduce the need for lifting tools and heavy equipment as human limitations of strength and stamina would be overcome. This can also contribute to increased speed and efficiency in the construction industry. To effectively control the RCW remotely, the robot can be fitted with sensors that can provide visual, auditory and haptic feedback to the human controller. This solution incorporates research in the fields of computer vision, robotics and construction safety.

The RCW requires the research and development of the following systems as shown in Figure 1:

Insert Figure 1 here

3D hand and body tracking - Vision based tracking techniques enable efficient and relatively inexpensive methods that could use fluid human motion to control a high degree of freedom RCW. They provide a non-intrusive and natural method to map the movements of the human to the robot. Furthermore, this ensures minimal retraining for human construction workers as they would move and perform actions, as they previously would have on-site. A novel framework to recover and track the 3D position, orientation and full articulation of the human hand combined with the human body is integral in controlling the robot naturally and efficiently. In this study, A RGB-D sensor, Microsoft Kinect, was selected and used to implement both the hand and body tracking based on two reasons. First, existing research in the field of motion tracking with the Kinect has shown very promising results and for this reason the Kinect becomes a standard in motion tracking for research and development. Second, it turned out that other possible solutions such as attaching accelerometers and IMUs to the human operator are less suitable as their accuracy is heavily dependent on the sensors being attached correctly in the correct positions. In addition, another possible approach using Leap Motion [30] device which provides a decent accuracy in hand and finger motions tracking has a drawback that it relies on the hand remaining at a stationary position at a precise position above the device. This fixes the hand in 3D space in x, y and z and hence prevents the movements of the arms, elbows and rotation of the upper body. This feature of the device significantly limits movement and would not accurately model the motions needed to carry out construction tasks.

Real-time simulation **-** Given that construction robots are humanoid, method of controlling such a robot must be considered and chosen. Two possible methods to do this are an operator using a remote/joystick device or mapping the movements of the operator to the robot. Due to the large number of degrees of freedom of a humanoid robot and as construction frequently requires the use of two hands in 3D motion, a remote control using remote/joystick device would not provide the same ease of control as mapping the movements of an operator to the robot. Moreover, mapping the operator’s movements requires minimal training for construction workers as they would largely perform the same actions they previously did on site. For these reasons, this study uses the assumption that the humanoid robot is remotely controlled by mapping the operator’s movements to the robot, which can be realised with real-time simulation. To this end, a realistic simulation of the construction site, equipment, construction tasks and 3D human pose tracking is vital because it enables training, testing, planning and model-based control of the robotics. Performing simulations enables quicker, cheaper and safer methods to model and develop the capacity of the system in numerous scenarios and conditions. It also provides the framework for further research into the robotic control system, mapping the real to the virtual environment in real-time and for visualising the 3D hand and body tracking. There are numerous existing methods to perform simulations (e.g. [28]), but the novelty of this solution requires developing a new real-time simulation to demonstrate tracking and control of a robotic construction worker to perform construction tasks.

Robotic construction worker – Based on the above assumptions for the RCW system, the robot should be humanoid to fulfil the role of a human construction worker and perform all construction tasks. This ensures the proposed solution integrates seamlessly with the existing infrastructure, tools, equipment and set-up of a construction site. It is required that the humanoid robot incorporates at least the 3D movement of 20 key body joints (head, chest, shoulders, elbows, wrists, hands, spine, hip, thighs, knees and feet) and of the fingers and hands (26 DoF (3D position and orientation of the palm, 2 angles for the base of each finger and 2 for the remaining finger joints)). The best humanoid robots still need significant research and development to be able to fully mimic a human construction worker and replace them.

Robotic control and feedback system - A robust, stable and fast control system must be developed to map human motions to the actuators on the robot. This ensures fluid and natural control of the robot without significant delays. In addition, it must maintain stability on two feet under scenarios such as uneven terrain, walking, climbing, carrying heavy loads etc. The latest DARPA (Defense Advanced Research Projects Agency) [31] challenges reveal that this is still a significant problem that needs to be solved in the near future. In addition, a framework of relaying feedback information such as haptics, visual and auditory systems is necessary to enable the fast, reliable and efficient teleoperation of the RCW, resulting in a closed-loop system.

3.2. Proposed framework and scope of this paper

Figure 2 shows the proposed framework for the RCW system. This study develops two essential systems which would fill the gap in knowledge identified in Section 2.4: (1) A novel system to combine vision based 3D hand and body tracking; and (2) A real-time simulation to demonstrate combined tracking and to simulate a construction site and virtual construction.

Body tracking pipeline - The system begins with the Microsoft Kinect sensor to perform vision based body tracking. It enables a relatively inexpensive and flexible approach to acquiring and processing RGB-D data. The Kinect Software Development Kit (SDK) [32] is capable of tracking 20 human joints and the skeletal tracking pipeline available in the Natural User Interface (NUI) library calculates the 3D joint position and bone orientation.

Hand tracking pipeline - Vision based 3D hand tracking uses the same Kinect sensor, processing RGB-D data using the FORTH Hand Tracker library developed by Prof. Argyros's group [22]. This can produce a monocular solution to hand and body tracking. The FORTH Hand Tracker calculates a 27 DoF parametrized representation of the 3D hand configuration and can be decomposed into joint coordinates in 3D homogeneous coordinates. The details of the hand tracking technique is shown in [22]. In this study, the coordinate system of the hand tracking is mapped to the coordinate system of the body tracking, resulting in combined hand and body tracking.

Insert Figure 2 here

Client/Server software pipeline - Combining the two systems of hand tracking and body tracking requires the development of a client-server software pipeline. A software pipeline was developed in this study to enable the real-time combination of hand tracking with body tracking. This pipeline consists of a chain of processing segments arranged such that the output of each segment is the input of the next, and enables a real-time communication channel between the two systems. It also allows the two independent body and hand tracking pipelines to run separately, simultaneously and seamlessly, enabling data to be transferred to a simulation platform.

Full body tracking simulation - The real-time simulation to illustrate combined body and hand tracking, the construction environment and virtual construction was developed in the Unity3D game engine [33]. Unity3D has substantial ready-made components necessary for virtual reality simulations with the Kinect and it includes graphics rendering, physics and Kinect SDK support. Furthermore, existing Unity3D toolkits have ready-made human characters that can be controlled to demonstrate full body tracking. To establish the development of the RCW, the following simulations were proposed in this study: (1) Combined body and hand tracking by a virtual RCW, (2) Two-handed lifting, moving and placing of a virtual object, (3) Single-handed grasping, moving and placing of a virtual object, (4) The use of a virtual hammer and shovel tools, (5) Building a virtual wall, and finally (6) Building a virtual house.

Construction environment - The 3D virtual environment was developed to simulate a construction site, tools, objects and a construction worker. The virtual construction simulation was generated by preparing various scenes in the Unity3D game engine. In each scene, the objects, algorithms, 3D models, camera and lighting were designed and built in a 3D virtual space.

The task of developing the 3D full body pose tracking, a simulation of a construction site, construction tasks, and controlling a virtual RCW are composed of numerous sub-tasks. Due to the limited resource available, a full treatment of all the above aspects above is beyond the scope of this paper. The simplifications made in this study are as follows: (1) The combined hand and body tracking is developed only for a single hand, which is sufficient for the demonstration of combining body and hand tracking; (2) The virtual RCW is based on the 20 joints of the human body that are sufficient to demonstrate body tracking for the control of the virtual RCW. This enables one-to-one mapping of the 3D human body joints to the corresponding joints of the virtual RCW; and (3) The control of a virtual RCW does not use actuators, but rather a virtual rendering as the hardware equivalent of the humanoid robot is yet to be developed and it lies beyond the scope of this study.

4. Research methodology and results

A summary of the developed systems and results are shown in Figure 3. This section goes through the details of how each sub-system was developed.

Insert Figure 3 here

4.1. Body tracking

The Kinect performs body tracking by calculating real-time 3D joint coordinates and orientations. This is presented in a hierarchical (parent-child) structure as shown in Figure 4(a). The Kinect SDK classes '*Joint*' and '*Skeleton*' are the containers for the body tracking data and provide a structured manner to utilize this information within the Unity3D simulation.

To demonstrate tracking and control of a virtual RCW, a standard 3D model of a construction worker was utilized [34]. This is composed of a graphically rendered mesh, character joints, and colliders placed on the body (Figure 4(b)). The character joints are managed in the same hierarchical system used by the Kinect SDK, enabling one-to-one joint mapping. The transform component defining 3D position and orientation is then updated with the skeletal tracking data to move the character model, tracking the user's movements.

Insert Figure 4 here

4.2.Combining hand and body tracking

4.2.1. Software pipeline

Figure 5 illustrates the flowchart of the software pipeline system developed to enable the real-time combination of hand tracking with body tracking. The pipe server programmed in C# maintains a real-time communication channel to the FORTH Hand Tracker (Python pipe client) and to Unity3D (C# pipe client). The pipeline designed and developed enables the FORTH Hand Tracker to send hand coordinates to the pipe server, which then sends the coordinates to Unity3D, in real-time. The software pipeline utilises .NET library's *Named pipes* which provide inter-process communication between a pipe server and one or more pipe clients. The pipeline initialises by setting up a local server. The server instantiates two pipes - one designated to connect to the Hand Tracker and the other to connect to the Unity3D. The Hand Tracker software was modified to connect to the local server as a client. The Unity3D simulation also connects to the server as a client. The modified Hand Tracker sends the calculated hand coordinates to the server after coordinate scaling and data format conversion. The server encodes the hand coordinates into a Byte array for communication via pipes to the Unity3D. The Unity3D client decodes the coordinates from a Byte array into direction and position vectors for the simulation. The coordinates are mapped to the coordinates used in the simulation to update the simulated hand.

Insert Figure 5 here

4.2.2. Coordinate transformation of hand tracking

The FORTH Hand Tracker calculates hand coordinates in a different coordinate system (3D homogeneous coordinates) to the one used in the Unity3D simulation (3D scene coordinates). A coordinate transformation was performed to convert the output from the Hand Tracker to the

Unity3D. In homogenous coordinates, 3D transformation matrix can be represented by 4×4 matrix. The linear transformation is described below with the transformation matrices:

$$\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b} \tag{1}$$

Equation 1 shows the linear transformation where $\boldsymbol{A}$ is the 4×4 transformation matrix, $\boldsymbol{x}$ is the 4×1 matrix of hand joint coordinates in the FORTH Hand Tracker, and $\boldsymbol{b}$ is the 4×1 matrix of hand joint coordinates in the Unity3D scene and. The matrix $\boldsymbol{A}$ has twelve unknowns, and each known correspondence between the two systems provides three equations. Hence, four correspondences are required to calculate the transformation matrix. It is also essential to exercise all degrees of freedom when choosing correspondences to ensure a unique solution.

In addition to the 3D joint coordinates, the orientations of hand and finger segments are essential in developing the simulation. The hand tracking libraries adapted, however, do not explicitly calculate the orientation of each modelled segment and incorrectly defining this can lead to spurious simulations. This problem can be visualised in Figure 6 where the lack of orientation information can lead to incorrect representation of the tracking. A 3D object such as a simulated finger-tip game object has 6 degrees of freedom made of 3D rotations and 3D positions. Figure 6(a) illustrates the correct 3D position and rotation of a finger segment. Note that the arrows indicate x, y and z vectors from the centre of the finger-tip game object. Figure 6(b) illustrates the same finger segment game object with only the 3D position constrained which is the same position as in Figure 6(a). The unconstrained rotation degrees of freedom lead to the incorrect game object orientation, resulting in the inaccurate simulation.

Insert Figure 6 here

4.2.3. Orientation vectors

Figure 7 shows the illustration of orientation vectors used for the hand tracking. The orientation of finger segments can be defined by two orthogonal vectors - one to indicate the forward direction and the other to indicate the up direction. Defining the orientation of finger segments can constrain the 3D position of the segment if their positions are restricted by hierarchical joint position updates using Object Oriented Programming of the Wrist joint (see Figure 7(c)).

At each time step of the simulation, the developed system pipe calculates the current orientation vectors of segments, the new orientation vectors and then updates the simulated hand and fingers with rotations. The current orientation vectors are calculated from the simulated hand in Unity3D. The new orientation vectors are calculated from the FORTH hand tracking coordinates by vector subtraction of joint coordinates and by vector cross products as shown in Figure 7(b).

<u>Insert Figure 7 here</u>

4.3.Virtual Construction Environment

4.3.1. Overview

The Unity3D simulation is developed by preparing various scenes. In each scene, the objects, algorithms, 3D models, camera and lighting were designed and built in a 3D virtual space. As a viewer observes a 2D screen image of the 3D world, a virtual camera was generated to capture a view for display. The camera component also defines the size and shape of the region that falls within the view. The 3D virtual environment developed to simulate a construction site, tools, objects and a construction worker is shown in Figure 8.

<u>Insert Figure 8 here</u>

Since the Unity3D platform is built on object-oriented prgramming, every entity within the scene is a '*GameObject*' which is the base class. It contains a variety of parameters and functions and acts as a container class. This enables other classes to be parented to the base class with the use of child classes '*Components*'. Parenting and creating child classes with this technique enables the grouping of objects in the scene and the inheritance of any transformations or algorithms that control the objects. This method is used to move objects in simulation that are held by the construction worker by parenting the held *GameObject* to the hand *GameObject*.

4.3.2. Simulated physics

Physics is enabled with *Rigidbody*, *Collider*, *Trigger* and *Joint Components*. First, *Rigidbody* enables mass to be added to an object and for it to respond to gravity. The physics game engine typically calculates the motion of objects with *Rigidbody* and a *Rigidbody* enables the object to be moved by incoming collisions with the addition of *Collider Components*. In cases where the user defines the motion of a *Rigidbody*, the motion is non-physical and is hence known as kinematic. This is performed with the *Rigidbody* property called *IsKinematic* to remove its motion control from the physics engine. This is the technique used to move objects once the virtual RCW grasps the objects, as it tracks the motion of the human controller's arms and hands.

Collider components define the shape of an object for physical collisions. *Colliders*, which are invisible, need to conform to the shape of the graphical rendering, with rough approximations enabling more efficient calculations. The least processor-intensive *colliders*, the *Box Collider*, *Sphere Collider* and *Capsule Collider,* were used to bring physical characteristics to the virtual RCW and to the virtual construction site as shown in Figure 9.

Insert Figure 9 here

A *Trigger* enables the physics engine to detect when one *collider* enters the space of another, without creating the resulting `collision'. A *Trigger* does not behave as a solid by enabling other colliders to pass through it. This technique was utilized to enable easy single-handed grasping of objects in the simulation.

Joints enable the attachment of one *Rigidbody* to another or to a fixed point in space. *Character Joints* are used in this study to create the virtual RCW to demonstrate body and hand tracking. They are a ball-socket joint, which allows the limitation of the joint movement on each axis.

4.4. Simulated Construction Scenarios

4.4.1. Picking up, moving and dropping

Picking up, moving and dropping an object are basic tasks to demonstrate construction. The system of picking up virtual objects with a two-handed grasp was developed with the use of *Colliders* placed on the left and right hands of the virtual RCW and on the virtual box that was to be lifted and moved. It uses *Rigidbody* physics such that if both hands are touching the box, it sets *IsKinematic* to true and the box *GameObject* becomes a child component of the Right Hand *GameObject*. Thus, movement of the box is enabled as the box *GameObject* inherits the position updates of the Right Hand *GameObject*. If both hands are not colliding with the box, then it is dropped. The code developed for this scene is outlined in Figure 10.

Insert Figure 10 here

Insert Figure 11 here

The class *BoxPickUp*, as shown in Figure 10, is attached to a box *GameObject* in the simulation, enabling a user to interact with it. The method *Start()* initialises private booleans indicating the current state of the box object. It also acquires the *GameObjects* that define the right and left hands. The simulated colliders attached to the box *GameObject* enable it to run the method *OnTriggerEnter()* whenever a game object collides with the box. If the collided object was either the right or left hand game objects, it updates the state of the box. The *OnTriggerExit()*

method is called when objects stop colliding with the box, updating the state accordingly. During the continuous method *Update()*, the private booleans are checked to see if both hands are colliding with the box. Depending on this, it updates the state of the box game object and calls the method *pickupObject()* or *dropObject()*. These methods convert the box *GameObject* into kinematic or rigidbody respectively, enabling simulated grasping/dropping of the box. The *pickupObject()* method then stores the parent class of the box game object and makes the box game object a child object of the hand game object. This leads to position updates of the hand game object (the parent object) by user movement to update the position of the box game object (the child object), enabling the user to move the grasped box in simulation. The *dropObject()* method reverses this process in the same manner. The simulation results were able to demonstrate the use of natural human motion to move a crate as shown in Figure 11.

The combination of hand and body tracking enables the development of single hand grasping of virtual objects (e.g. tools). Single hand grasping was developed with the same principles as two-handed grasps - using *colliders* and the grasping algorithm as shown in Figure 10. Due to the large noise in hand tracking, larger triggers were designed over the palm of the hand and over the finger tips to enable more robust grasping. If triggers of both the palm and finger tips collide with a virtual object, it indicates closing of the fingers and positioning of the hand for grasping an object. The simulation result of single-handed grasping is shown in Figure 12.

Insert Figure 12 here

4.4.2. Building a wall

In addition to grasping, moving and placing objects, this scenario enables the demonstration of further scenarios, building a wall. Uneven mortar laying atop bricks was designed and the bricks that are misaligned atop this mortar was reformed using a hammer to correctly align misaligned bricks and repeating the procedure to increase the wall height as shown Figure 13.

Insert Figure 13 here

Mortar was modelled as a prefab cuboid along with the class *CementManager* for simulation of laying mortar on bricks. The simulation was designed such that the shovel tool enabled the user to spread mortar atop bricks with *colliders*. To demonstrate uneven mortar laying, an algorithm implemented a random generator to randomly change the mortar laid, making it a more realistic simulation. The *CementManager* class is added as a component to the brick *GameObject* and inherits the brick's public properties.

Bricks that were placed atop uneven mortar were programmed to be the child *GameObjects* of the mortar *GameObjects* so that the mortar's orientation and position properties are inherited. This enables the bricks to retain uneven positioning that is dependent on the randomly generated mortar that was placed. A new class *HammerHitAlignVertical* was developed to demonstrate hammering the uneven bricks such that they rotate to the correct orientation over the mortar. This was implemented by calculating the *GameObject*'s up vectors and rotating their orientation to align with the Global up vector upon colliding with the hammer. Finally, the simulation was further extended to demonstrate the construction of four walls and a roof to create a small virtual house as shown in Figure 14.

Quantitatively, validation of the proposed solution was carried out by comparing the time taken to implement the tasks in the simulation with the time taken in the real world. The task of building a small model house as shown in Figure 14 took on average 60 minutes in simulation where the task normally takes on average 20 minutes in the real world. The task in simulation took about three times longer than in the real world. The limiting factor for simulation speed is the slow movement speed of the operator, which was needed for the accurate tracking of the hand and fingers. It is expected that when model based hand tracking solutions are improved and optimized, it would significantly reduce the simulation time.

Qualitative measures of validation for the developed solution include exploring the complexity of tasks that can be handled. This was explored for grasping objects, moving them in 3D space, releasing and placing an object precisely and using tools. This simulation result demonstrates that the RCW system based on real-time simulation is capable of replicating real-life actions such as tools and building walls. With further development which is warranted for future work, it should be capable of simulating more complex construction tasks.

The limitations of the FORTH Hand Tracker lead to constraints on the speed of hand and finger movements for accurate tracking. To prevent loss of hand tracking, the movements must be relatively slow, smooth and minimize finger occlusions. Furthermore, the Hand Tracker operates accurately only within a one to two meter depth range from the Kinect sensor. These slow down the process of virtual construction and reduce the preciseness of moving and placing virtual objects.

Insert Figure 14 here

5. Conclusions and future work

In considering the larger goal of improving the health and safety of construction workers at a construction site, this study focused on tackling three major risk factors – (1) fall from heights, (2) musculoskeletal disorders and (3) being struck by objects. The authors proposed a novel solution called Robotic Construction Worker (RCW) system that effectively eliminates the risks faced by human construction workers. As a first step in establishing this solution, the authors developed two essential systems of the RCW – (1) combined body and hand tracking for the efficient and natural control of the humanoid robot and (2) a simulation environment to demonstrate, test and develop a virtual RCW.

Using a single Microsoft Kinect sensor, a novel framework of combining vision based hand tracking (FORTH Hand Tracker) with body tracking (Microsoft Kinect SDK) was developed. This was realised with coordinate mapping and a software pipeline to enable the tracking systems to run independently and simultaneously. The framework demonstrated accurate and successful combined tracking in real-time.

The Unity3D game engine was employed to develop a virtual construction site. This was used to illustrate the use of the combined tracking to carry out virtual construction - moving crates with two hands, picking and placing bricks with a single hand and the use of construction tools. This successfully demonstrated the building of walls, with mortar spreading and hammering bricks, to complete a virtual house. These results display, as a proof-of-concept, the promising capabilities of the RCW.

This research contributes to the building, civil and information engineering community by providing a novel approach to eliminating the risks faced by construction workers on site. Technical contributions of this research are twofold: (1) The development of a novel framework of combining vision based hand tracking and body tracking as a first ever. The full body vision based tracking system uses 20 body joints and 26 degrees of freedom hand; and (2) The development of a simulation of a realistic and physics based construction environment.

To further develop the RCW, suggestions for future work include: (1) develop a haptic feedback system for the user using the developed simulation. This demonstrates the feedback system of the long-term proposed solution, (2) demonstrate a wider range of construction tasks to develop a more detailed virtual construction environment. This enables the testing and development of the key features and capabilities of a RCW, and (3) develop the use of multiple Kinect sensors can enable 360 degrees of tracking the user, as currently the system is restricted by range of view of a single Kinect sensor .

ACKNOWLEDGEMENTS

Prof. Argyros is gratefully acknowledged for his insightful comments and suggestions during his visit to the Cambridge University Engineering Department and for supporting this project with his research and software. Dr. Kyriazis is gratefully thanked for his support with the FORTH Hand Tracker. Finally, the kind support and guidance from Mr. Huethwohl in the Construction Information Technology Laboratory at the University of Cambridge is gratefully acknowledged.

REFERENCES

[1] Health and Safety Executive (HSE), Improving health and safety in the construction industry, (2004) Retrieved from https://www.nao.org.uk/report/health-and-safety-executive-improving-health-and-safety-in-the-construction-industry, Accessed date: 30 September 2017.

[2] Health and Safety Executive (HSE), Statistics on fatal injuries in the workplace in Great Britain 2016, (2016) Retrieved from http://www.hse.gov.uk/statistics/pdf/fatalinjuries.pdf, Accessed date: 30 September 2017.

[3] S. Zhang, J. Teizer, J. K. Lee, C. M. Eastman, M., Venugopal, Building Information Modeling (BIM) and safety: Automatic safety checking models and schedules, Autom. Constr. 29 (2013) 183-195, https://doi.org/10.1016/j.autcon.2012.05.006

[4] Great Britain (1974), Safety and Health at Work etc. Act, HMSO, London.

[5] O. Rezenfeld, R. Sacks, Y. Rosenfeld, H. Baum, Construction Job Safety Analysis, Saf. Sci. 48(4), 2010, 491-498, https://doi.org/10.1016/j.ssci.2009.12.017

[6] H. Malekitabar, A. Ardeshir, M. H. Sebt, R. Stouffs, Construction safety risk drivers: A BIM approach, Saf. Sci. 82 (2016) 445-455, https://doi.org/10.1016/j.ssci.2015.11.002

[7] E. M. Wetzel and W. Y. Thabet, The use of a bim-based framework to support safe facility management processes, Aut. Constr. 60 (2015) 12-24, https://doi.org/10.1016/j.autcon.2015.09.004

[8] Y. Zou, A. Kiviniemi, S. W. Jones, A review of risk management through BIM and bim-related technologies, Saf. Sci. (2016), http://dx.doi.org/10.1016/j.ssci.2015.12.027

[9] Strategic Forum for Construction, Short guide to improving health and safety on construction sites through effective worker involvement, (2007) Retrieved from http://www.unitetheunion.org/uploaded/documents/Improving%20health%20%26%20safety%20on%20construction%20sites%20through%20effective%20worker%20involvement%20(SFfC%20guide)11-4961.pdf, Accessed date: 30 September 2017.

[10] Health and Safety Executive (HSE), Health and safety in construction sector in Great Britain, (2015) Retrieved from http://www.hse.gov.uk/statistics/industry/construction/construction.pdf, Accessed date: 30 September 2017.

[11] Y. Fang, Y.K. Cho, J. Chen, A framework for real-time pro-active safety assistance for mobile crane lifting operations, Autom. Constr. 72 (Part 3) (2016) 367–379, http://dx.doi.org/10.1016/j.autcon.2016.08.025

[12] Y.-C Fang, R.-J Dzeng, Accelerometer-based fall-portent detection algorithm for construction tiling operation, Autom. Constr. 84 (2017) 214-230, https://doi.org/10.1016/j.autcon.2017.09.015

[13] S. Bakhshi, M.H. Mahoor, B.S. Davidson, Development of a body joint angle

measurement system using IMU sensors, Engineering in Medicine and Biology Society, EMBC, 2011 Annual International Conference of the IEEE (2011) 6923-6926, http://dx.doi.org/10.1109/IEMBS.2011.6091743

[14] H. Jebelli, C.R. Ahn, T.L. Stentz, Fall risk analysis of construction workers using inertial measurement units: validating the usefulness of the postural stability metrics in construction, Saf. Sci. 84 (2016) 161–170, http://dx.doi.org/10.1016/j.ssci. 2015.12.012

[15] E. Valero, A. Sivanathan, F. Bosche, M. Abdel-Wahab, Musculoskeletal disorders in construction: A review and a novel system for activity tracking with body area network, Applied Ergonomics 54 (2016) 120-130, https://doi.org/10.1016/j.apergo.2015.11.020

[16] T. Cheng, G.C. Migliaccio, J. Teizer, U.C. Gatti, Data Fusion of Real-Time Location Sensing and Physiological Status Monitoring for Ergonomics Analysis of Construction Workers, J. Comp. Civil Eng., 27(3) (2017) 320-335 https://doi.org/10.1061/(ASCE)CP.1943-5487.0000222

[17] M.J. Tadi, T. Koivisto, M. Pänkäälä, A. Paasio, Accelerometer-based method for extracting respiratory and cardiac gating information for dual gating during nuclear medicine imaging, J. Biomed. Imaging 2014 (6) (2014) 1–11, http://dx.doi.org/10.1155/2014/690124

[18] W. Lee, K.-Y Lin, E. Seto, G.C. Migliaccio, Wearable sensors for monitoring on-duty and off-duty worker physiological status and activities in construction, Autom. Constr. 83 (2017) 341-353, https://doi.org/10.1016/j.autcon.2017.06.012

[19] J. O. Seo, S. U. Han, S. H. Lee, H. Kim, Computer vision techniques for construction safety and health monitoring, Adv. Eng. Inf. 29(2) (2015) 239-251, https://doi.org/10.1016/j.aei.2015.02.001

[20] S. U. Han, S. H. Lee, F. Pena-Mora, Vision-based detection of unsafe actions of a construction worker: Case study of ladder climbing, J. Comp. Civil Eng. (ASCE) 27(6) (2013) 635-644, https://doi.org/10.1061/(ASCE)CP.1943-5487.0000279

[21] S. J. Ray, J. Teizer, Real-time construction worker posture analysis for ergonomics training, Adv. Eng. Inf. 26(2) (2012) 439-455, https://doi.org/10.1016/j.aei.2012.02.011

[22] I. Oikonomidis, N. Kyriazis, A. A. Argyros, Efficient model-based 3d tracking of hand articulations using Kinect, In *Proceedings of the British Machine Vision Conference*, University of Dundee, United Kingdom. (2011) http://dx.doi.org/10.5244/C.25.101

[23] A. Vitali, C. Rizzi, A virtual environment to emulate tailor's work, Computer-Aided Design and Applications 14(5) (2017) 671-679, http://dx.doi.org/10.1080/16864360.2016.1273584

[24] G. Colombo, G. Facoetti, C. Rizzi, A. Vitali, Mixed reality to design lower limb prosthesis, Computer-Aided Design and Applications 13(6) (2016) 799-807, http://dx.doi.org/10.1080/16864360.2016.1168223

[25] Leap Motion Inc., Retrieved from https://www.leapmotion.com/, Accessed date: 30 September 2017.

[26] T. Sharp, C. Keskin, D. Robertson, J. Taylor, J. Shotton, D. Kim, C. Rhemann, I. Leichter, A. Vinnikov, Y. Wei, D. Freedman, P. Kohli, E. Krupka, A. Fitzgibbon, S. Izadi, Accurate, robust, and flexible real-time hand tracking, The 33rd Annual ACM Conference on Human Factors in Computing Systems, Seoul, Republic of Korea. (2015), http://dx.doi.org/10.1145/2702123.2702179

[27] J. Sklar, Robots lay three times as many bricks as construction workers, MIT Technology Review, (2015) Retrieved from

https://www.technologyreview.com/s/540916/robots-lay-three-times-as-many-bricks-as-construction-workers/ Accessed date: 30 September 2017.

[28] BOMAG, Trench compactor, (2015) Retrieved from http://www.bomag.com/world/en/construction-equipment-dictionary-trench-compactor.html Accessed date: 30 September 2017.

[29] Y. Hironao, X. Tang, T. Ni, D. Zhao, A. Y. Ahmad. Tele-operation construction robot control system with virtual reality. IFAC Proceedings Volumes, 42(16) (2009) 639-644, https://doi.org/10.3182/20090909-4-JP-2010.00108

[30] Y. Song, D. Demirdjian, R. Davis, Tracking body and hands for gesture recognition: Natops aircraft handling signals database, IEEE Automatic Face & Gesture Recognition and Workshops, (2011) 500-506, https://doi.org/10.1109/FG.2011.5771448

[31] Defence Advanced Research Projects Agency (DARPA), Retrieved from https://www.darpa.mil/ Accessed date: 30 September 2017.

[32] Microsoft Kinect, Microsoft Kinect for Windows Software Development Kit (SDK) 2.0, Retrieved from https://www.microsoft.com/en-us/download/details.aspx?id=44561 Accessed date: 30 September 2017.

[33] Unity3D, Unity documentation, Retrieved from http://docs.unity3d.com/Manual/ Accessed date: 30 September 2017.

[34] T. Takala and R. Pugliese, Reality-based User Interface System, (2015) Retrieved from http://blog.ruisystem.net/about/ Accessed date: 30 September 2017.

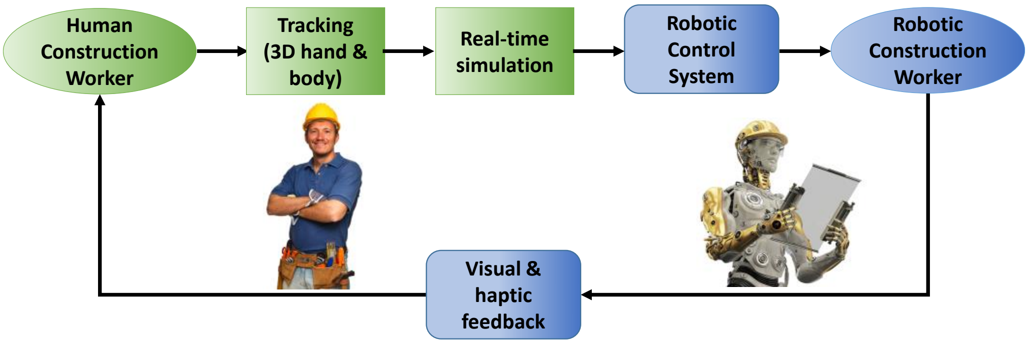


Figure 1. The feedback cycle from the human construction worker to the controlled Robotic Construction Worker. (The boxes in green indicate the research objectives of this study.)

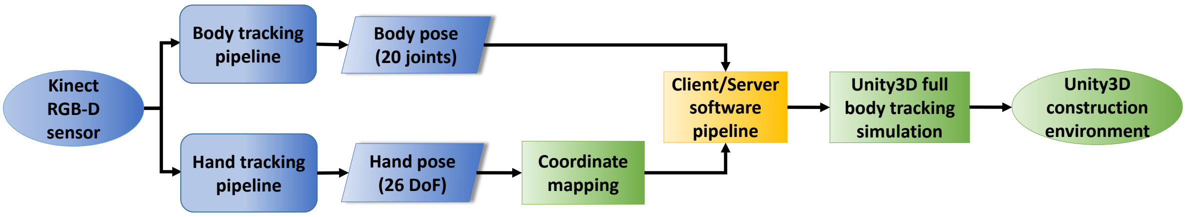


Figure 2. The proposed framework (colour coded to illustrate the authors' contribution. Blue - systems used as is. Green - newly developed systems. Yellow - existing systems extensively modified.)

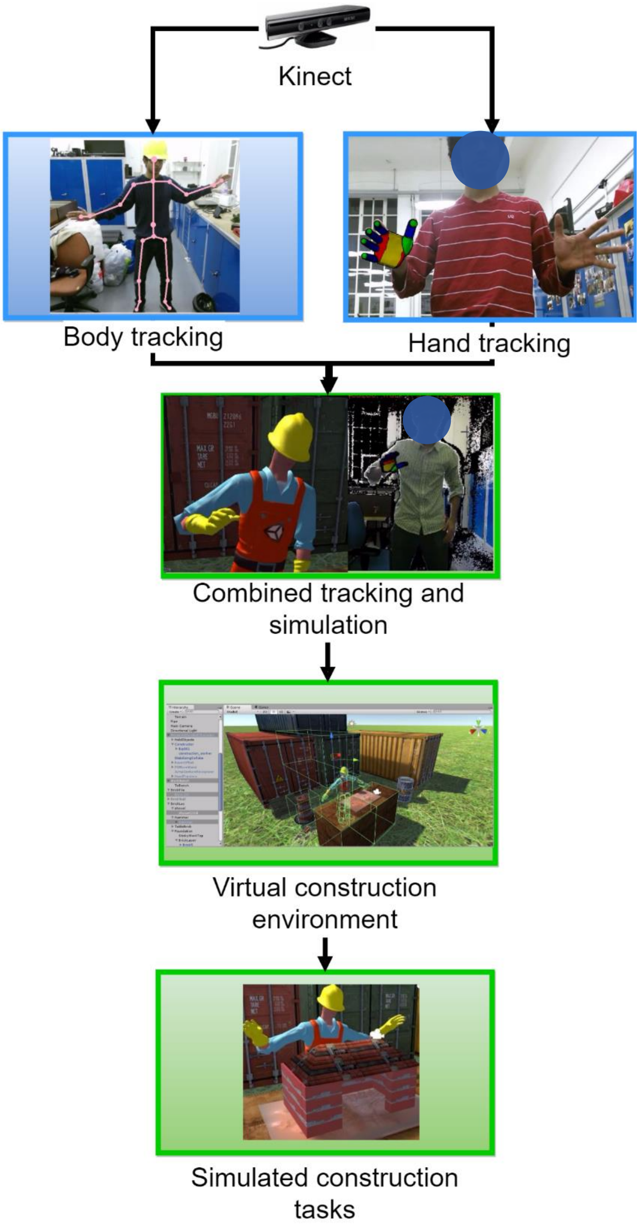
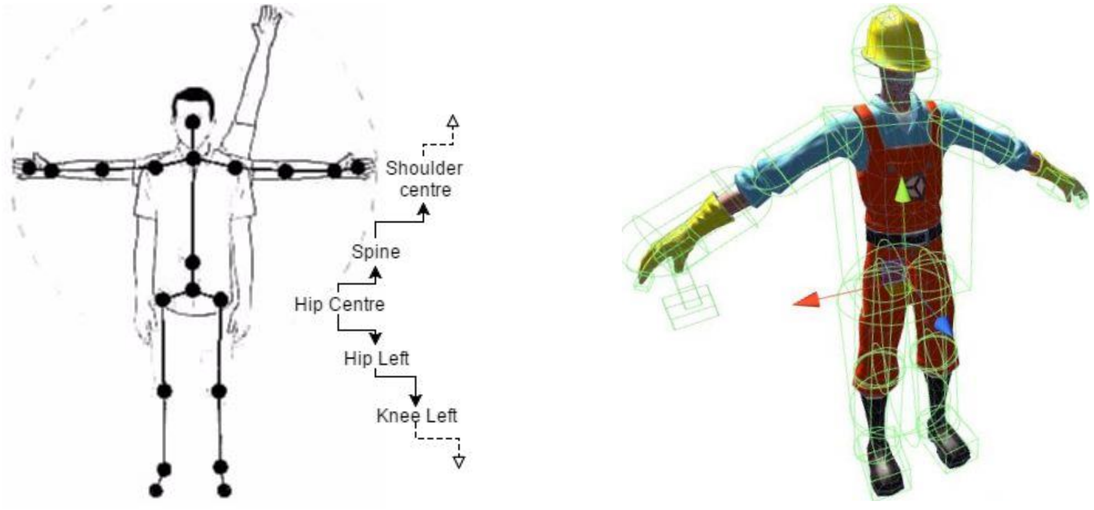


Figure 3. A summary of the developed system and results. (Blue indicates systems used as-is and green indicates novel systems.)



(a) (b)

Figure 4. Body tracking setup: (a) The hierarchical relationship of the different joints tracked and their orientation (image modified from [19]). The Hip Centre joint is set as the root and the hierarchy then extends to the feet, head, and hands. (b) The Unity3D character model with character joints and colliders shown as a wire frame.

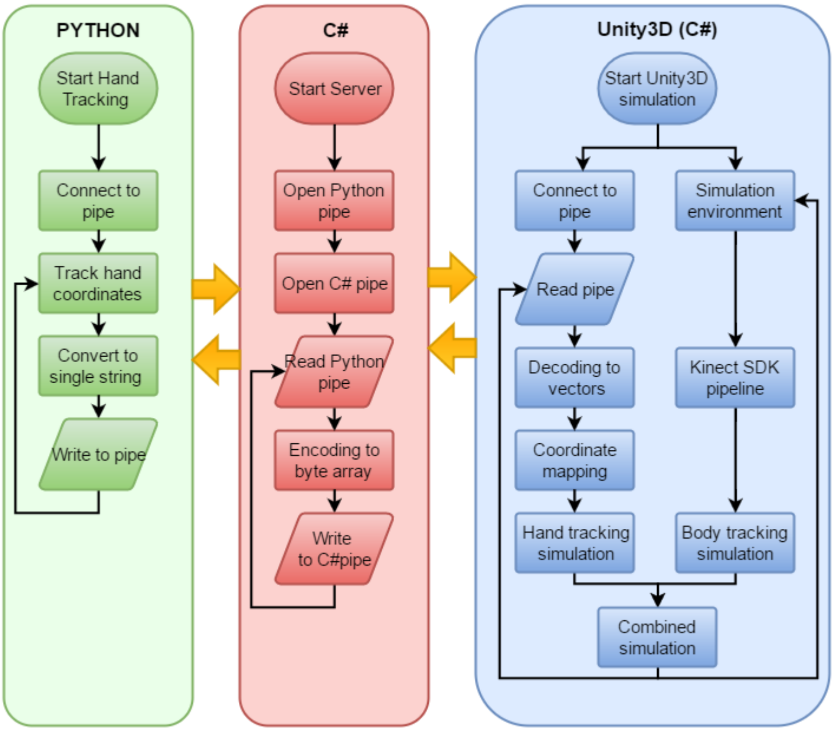
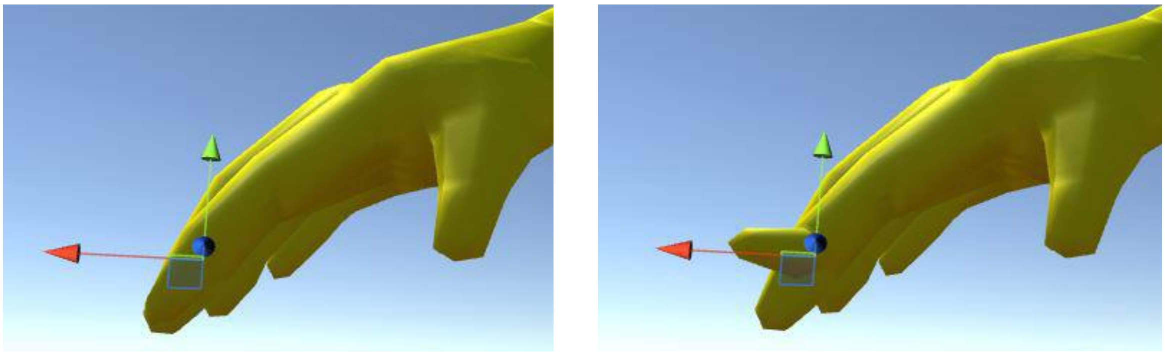


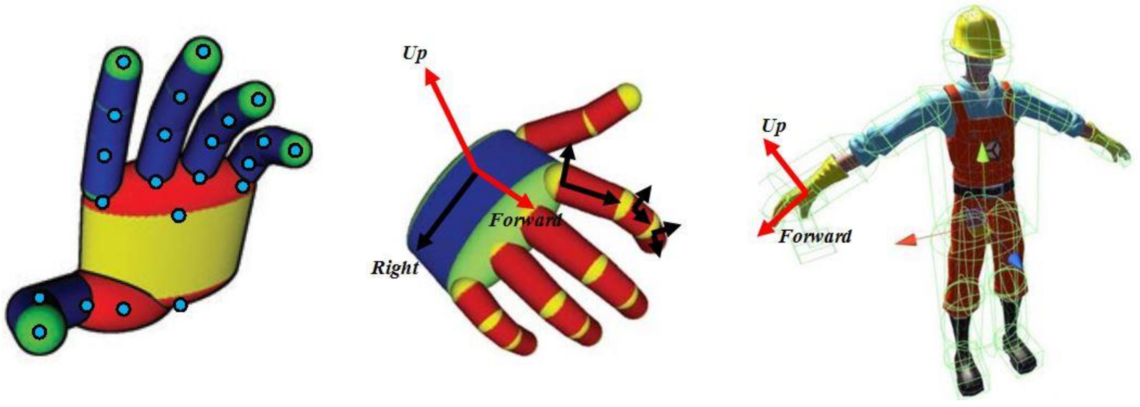
Figure 5. The flowchart of the software pipeline system developed to enable the real-time combination of hand tracking with body tracking. The Python client communicates to the C# server when it initially connects to the pipe and during the simulation, where at each time-step the hand coordinates are tracked and updated. The Unity3D simulation performs body tracking and hand tracking independently. After it connects to the C# server, it reads the hand coordinates at each time-step throughout the simulation.



(a) Correct coordinate transformation to fix the 3D position of a finger segment in 3D space and with the correct orientation of the segment.

(b) Correct coordinate transformation to fix the 3D position of a finger segment but with incorrect orientation.

Figure 6. An illustration of how fixing only the 3D position does not constrain the orientation of the segment in 3D space, leading to an inaccurate simulation of hand tracking.



(a) (b) (c)

Figure 7. Illustration of orientation vectors used for the hand tracking: (a) Blue dots indicate the 3D coordinates inferred from the hand tracking software pipeline; (b) Black arrows, using the index finger as an example, illustrate how orthogonal vectors that indicate the forward and up vectors of each hand segment are calculated. The cross product of the right and forward vectors calculates the orthogonal up vector; and (c) The Wrist orientation (both red vectors) can be mapped to the Unity3D Wrist orientation as the anchor point for coordinate mapping, which is updated with the Kinect SDK Pipeline.

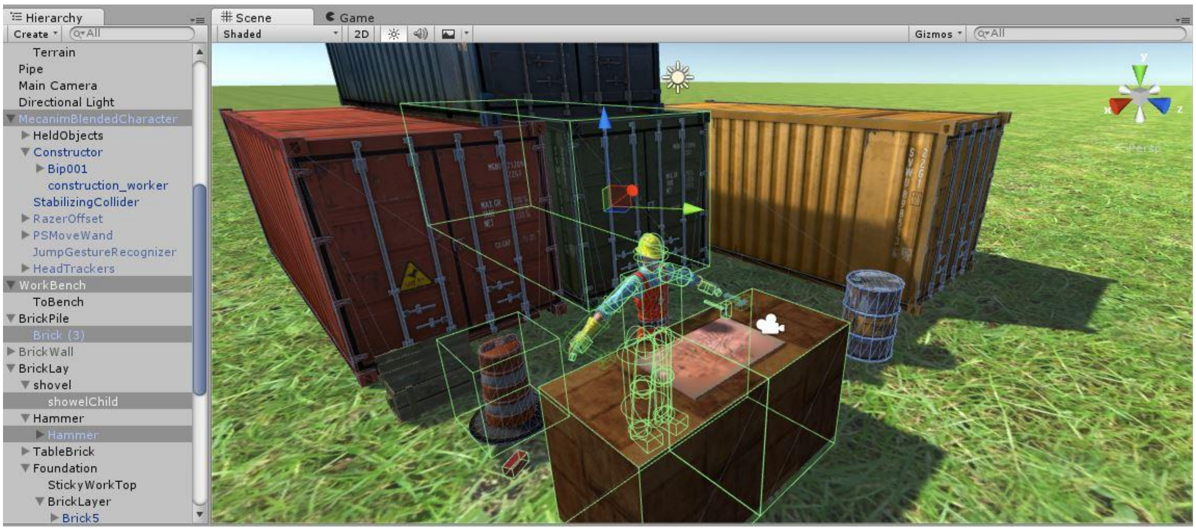


Figure 8. The virtual construction environment developed with a human character model, tools, bricks, crates, terrain, camera and lighting.

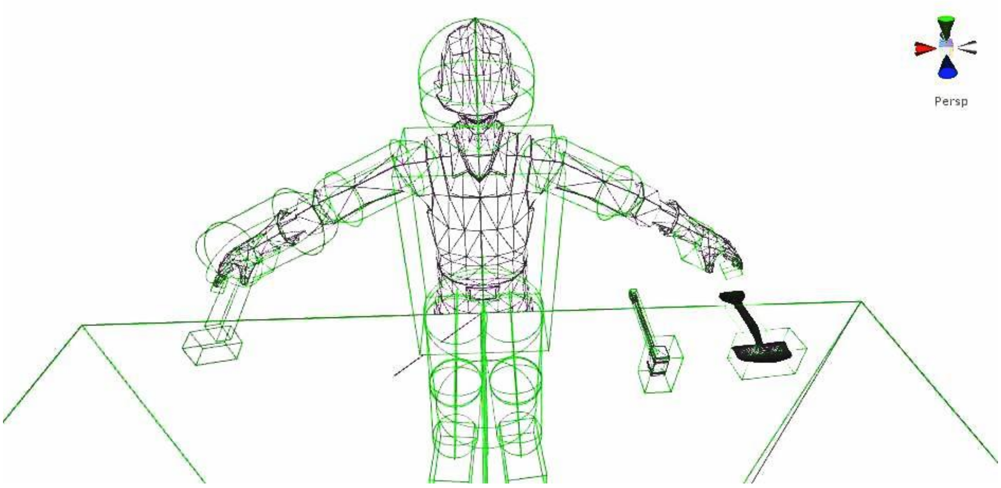


Figure 9. An illustration of the *collider* components developed to manage interactions of the construction worker's body, hands and objects in simulation (Green lines indicate the boundaries).

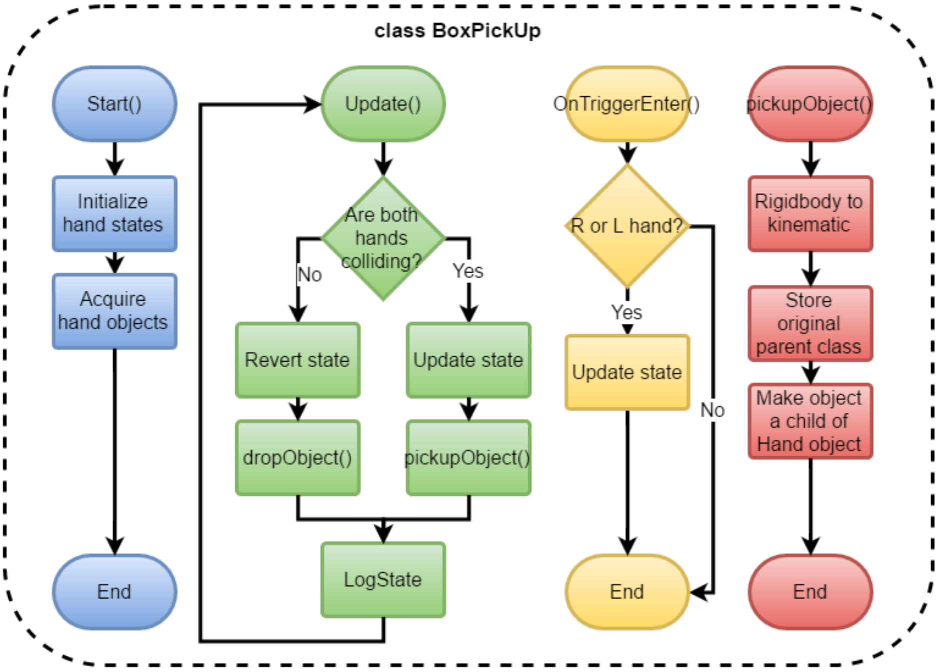


Figure 10. The *Start()*, *Update()*, *OnTriggerEnter()* and *pickUpObject()* functions in the class BoxPickup for two-handed grasping of virtual objects.

Figure 11. An example of picking up, moving and dropping a large crate in simulation using the two hand interaction. (Top left shows the user's live motions in front of the Kinect.)

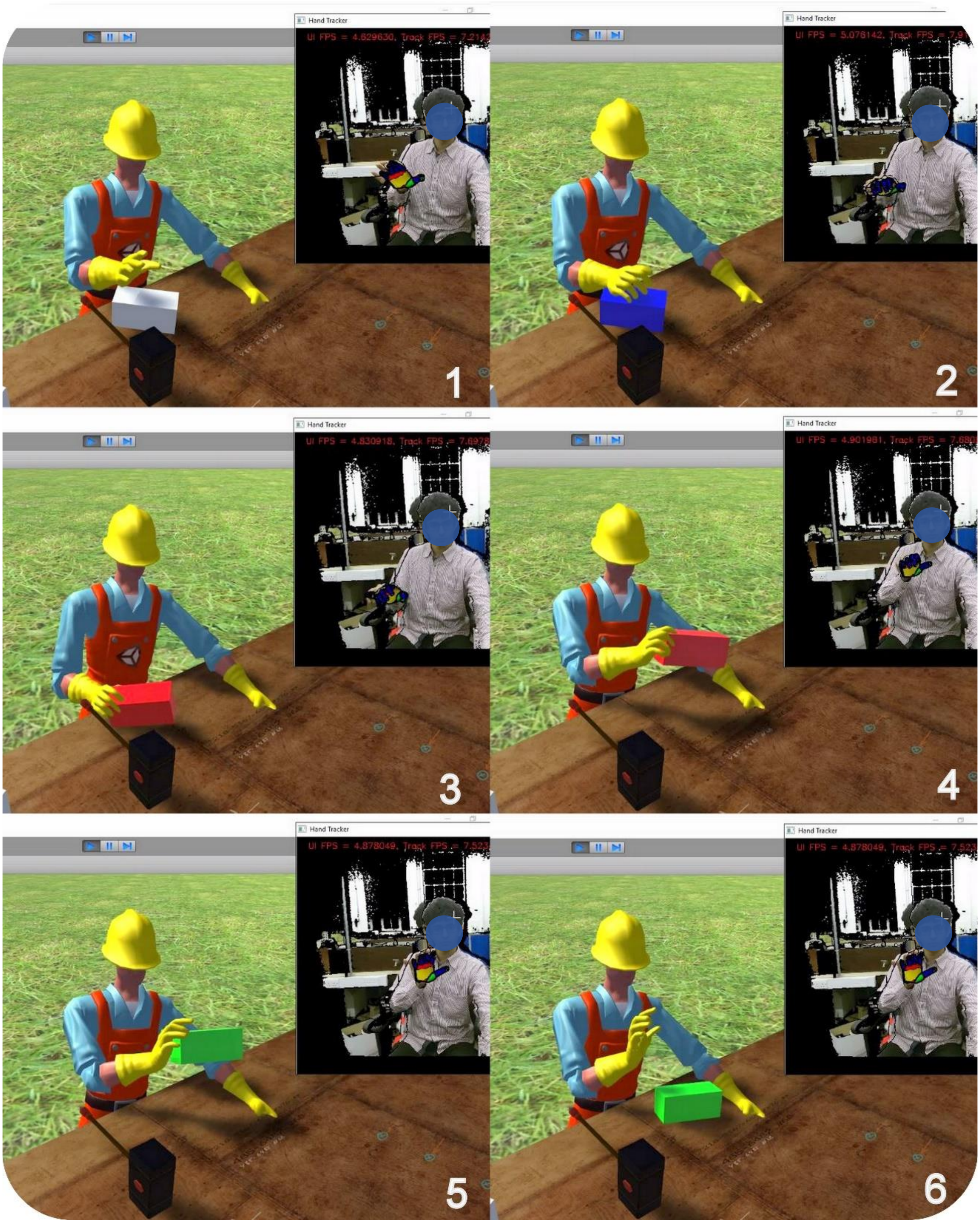


Figure 12. An example of single-handed grasping. (Combining body with hand tracking enables the grasping of objects with a single hand in realistic grips. The blue colour indicates the recognition of the hand trigger collision. The red colour indicates the recognition of both hand and finger triggers colliding and hence enables picking up the object. The green colour indicates relaxing of the grip i.e. dropping the object.)

Figure 13. Snapshots of the simulation of building a wall

Figure 14. A virtual house built to demonstrate virtual construction