

## •Article•

# Development of augmented reality serious games with a vibrotactile feedback jacket

Lingfei ZHU<sup>1</sup>, Qi CAO<sup>2</sup>, Yiyu CAI<sup>1\*</sup>

1. School of Mechanical & Aerospace Engineering, Nanyang Technological University, 639798, Singapore

2. School of Computing Science, University of Glasgow, 567739, Singapore

\* Corresponding author, [myycail@ntu.edu.sg](mailto:myycail@ntu.edu.sg)

Received: 18 March 2020 Accepted: 25 May 2020

**Citation:** Lingfei ZHU, Qi CAO, Yiyu CAI. Development of augmented reality serious games with a vibrotactile feedback jacket. *Virtual Reality & Intelligent Hardware*, 2020, 2(5): 454—470  
DOI: 10.1016/j.vrih.2020.05.005

**Abstract Background** In the past few years, augmented reality (AR) has rapidly advanced and has been applied in different fields. One of the successful AR applications is the immersive and interactive serious games, which can be used for education and learning purposes. **Methods** In this project, a prototype of an AR serious game is developed and demonstrated. Gamers utilize a head-mounted device and a vibrotactile feedback jacket to explore and interact with the AR serious game. Fourteen vibration actuators are embedded in the vibrotactile feedback jacket to generate immersive AR experience. These vibration actuators are triggered in accordance with the designed game scripts. Various vibration patterns and intensity levels are synthesized in different game scenes. This article presents the details of the entire software development of the AR serious game, including game scripts, game scenes with AR effects design, signal processing flow, behavior design, and communication configuration. Graphics computations are processed using the graphics processing unit in the system. **Results /Conclusions** The performance of the AR serious game prototype is evaluated and analyzed. The computation loads and resource utilization of normal game scenes and heavy computation scenes are compared. With 14 vibration actuators placed at different body positions, various vibration patterns and intensity levels can be generated by the vibrotactile feedback jacket, providing different real-world feedback. The prototype of this AR serious game can be valuable in building large-scale AR or virtual reality educational and entertainment games. Possible future improvements of the proposed prototype are also discussed in this article.

**Keywords** Augmented reality; AR serious games; Vibrotactile feedback jacket; Game scenes

## 1 Introduction

At present, immersive and interactive learning is increasingly recognized as a ubiquitous form of learning<sup>[1]</sup> and it can enhance students' learning motivation and efficiency<sup>[2]</sup>. In case of simulations of real-world events, serious games are used for purposes beyond entertainment<sup>[3]</sup>. Computer-based immersive serious games have been developed for various applications, such as education, learning, training, healthcare, rehabilitation, and politics<sup>[3-6]</sup>. Immersive serious games have been used in training in educational, medical,

and military sectors<sup>[7]</sup>. Educational serious games should integrate pedagogical theories and motivational principles to engage learners in a manner that traditional education does not<sup>[8]</sup>. They are effective tools to help learners acquire new knowledge and skills<sup>[9]</sup>.

Augmented reality (AR) is a technology used to expand the physical world and environment by overlaying digital information, such as images, videos, instruction texts, and virtual objects<sup>[10,11]</sup>. Computer-generated sensory inputs, such as virtual objects, animations, and sounds, in the AR systems can enhance user experiences by supplementing real-world information<sup>[12]</sup>. In addition to virtual reality (VR), AR can provide an immersive learning environment to improve educational processes. It enables certain learning exercises and teaching activities that are sometimes difficult to execute in traditional classroom education<sup>[13]</sup>. It also provides a safe and risk-free teaching and learning environment for certain applications<sup>[14]</sup>. As the costs of head-mounted devices (HMDs) and computational platforms are significantly decreasing, immersive AR serious games become more accessible and cost-effective<sup>[14]</sup>.

Various feedback methods in serious games can provide information regarding the correctness of corresponding responses to learners. Such feedback methods can improve their performance and learning outcomes<sup>[15]</sup>. Visual, audio, and motion interactions are common engagement elements for learners in immersive AR serious games. The integration of vibrotactile feedback can provide a complementary approach to exploit the full capabilities of human beings<sup>[16]</sup>. The vibrotactile feedback can be generated by wearable, graspable, or stationary devices<sup>[17]</sup>. AR systems can work with vibrotactile feedback devices to monitor, direct, or influence users' movements<sup>[10]</sup>. Vibrotactile feedback has become a cost-effective and popular approach to improve user experiences while interacting with virtual objects<sup>[18]</sup>. Vibrotactile feedback stimuli are widely adopted in the tasks of AR serious games, including navigations, redirecting attentions, or communications. It can improve the realism, immersion, or sense of presence in virtual worlds<sup>[19]</sup>. Moreover, the enhancement of the similarities between real scenarios and virtual environments can benefit the objectives of serious games<sup>[20]</sup>.

Vibration is the most commonly used vibrotactile stimulus type, and is better known than other alternatives in current wearable devices<sup>[21]</sup>. A high degree of variation in AR serious games can produce richer and more realistic user experiences<sup>[3]</sup>. The use of vibration actuators is a popular, practical, and inexpensive approach to provide vibrotactile feedback<sup>[18]</sup> as vibration actuators are easy to control and typically smaller than the size of a coin<sup>[21]</sup>. Vibrotactile vibration patterns and sequences can be key factors to improve realism and immersion sense in vibrotactile interactions in virtual environment<sup>[20]</sup>.

The vibrotactile vibration feedback can be a part of a multimodal virtual environment integrated with video and audio for sense of presence<sup>[20]</sup>. It can be built into wearable devices to be worn in different positions of the human body, such as the finger, hand, wrist, head, or upper body, to generate vibrotactile feedback in the AR scenes. Some related works have been reported in the literature. Maisto et al. presented wearable haptic devices for fingers, which provided the interaction forces for the weight of virtual objects in AR scenarios<sup>[12]</sup>. A smartwatch-type wearable device can provide tactile sensations on the wrist of users for two types of stimuli: wind and vibration<sup>[22]</sup>. Kaul and Rohs introduced a wearable device HapticHead, utilizing 20 vibration motors distributed in three concentric ellipses around the head to increase immersion for VR applications<sup>[23]</sup>. A wristband was capable of providing feedback of local vibration and pressure using eight actuators<sup>[24]</sup>. A vibrotactile HMD was built with seven electromechanical tactors placed around the user's head<sup>[19]</sup>. The vibrotactile HMD points toward the direction of a virtual object in the azimuthal plane using these tactors. A haptic feedback for VR HMDs was presented by combining vibrotactile feedback with thermal sources<sup>[25]</sup>. Louison et al. presented a vibrotactile device positioned on the user's right upper limb with 10 actuators<sup>[26]</sup>. Garciavalle et al. described an integration of a haptic vest with HMDs to interact with a virtual environment<sup>[20]</sup>. A total of 54 vibrotactile actuators were placed on a large vest and

38 vibrotactile actuators on a medium-size vest for vibration generations. The vest covered the torso of users. The vest also used thermoelectric actuators to create hot and cold sensations. A force jacket was created using an array of 26 actuated airbags and force sensors to provide directed force and vibrations to the torso of users in VR games<sup>[27]</sup>. Kishishita et al. presented a wearable force-feedback suit for the upper extremity<sup>[28]</sup>. There are seven possible positions for the attachment of the pneumatic actuators in the prototype. Rognon et al. described a haptic feedback device embedded in a jacket used in a flight simulation<sup>[29]</sup>. Ouyang et al. also introduced a vibrotactile vest for a flight simulation<sup>[30]</sup>. A haptic jacket with two types of actuators was demonstrated for pneumatically actuating kinesthetic movements of arm joints in a VR environment<sup>[31]</sup>. Some commercial feedback vests for VR games were also reported, such as KOR-FX gaming vest<sup>[32]</sup>, TACTOT haptic vest<sup>[33]</sup>, Vest PRO<sup>[34]</sup>, and TESLASUIT<sup>[35]</sup>. These vibrotactile feedback devices work with their specific games.

To make the vibrotactile feedback jackets more flexible and thus support varying AR serious games, the standardized and general design cycle, including the software, game engines, and hardware, should be discussed. Each part in the complete design cycle has standard and flexible interfaces to be updated by software commands. The software interfaces can be configured and controlled for different game scenes in various AR/VR serious games. These are important baselines for enabling the features of the multimodal feedback of visual, audio, and vibrotactile senses in AR/VR serious games. However, it is challenging to technically achieve multimodal feedback features. The entire design flow and architecture must be well planned, including the hardware configurations, game scripts, game scenes, game engine design, embedded software control commands, and AR/VR visualizations.

This article presents a complete design cycle for an AR serious game using visual, audio, and vibrotactile feedback. The vibrotactile feedback jacket is battery powered and specifically designed to provide real-world feedback and user experiences in different AR game scenes. An AR serious game is developed as a standalone game using the Unity engine. The complete system includes three major parts: hardware configurations of the HMD with a vibrotactile feedback jacket, software development of the AR game, and communication among different parts with gamers. These are presented in detail in the following sections.

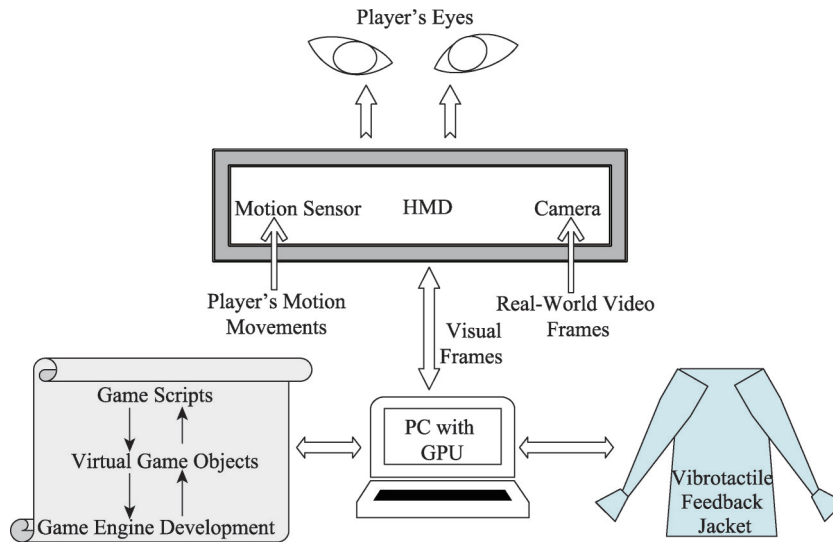
The remainder of this paper is organized as follows. Section 2 presents the configurations of the hardware system of the developed AR serious game, with the designs of the HMD system and vibrotactile feedback jacket. Section 3 describes the software development of the AR serious game, including the game scenes, interaction functions, serial communication format, and control methods. Section 4 details the performance evaluation and future development of the prototype. Section 5 concludes this paper.

## 2 Configurations of the HMD system and vibrotactile feedback jacket

The aim of this study is to develop computerized virtual models for an AR game and a vibrotactile feedback jacket that can be used in multimodal interactive and immersive learning. The overall block diagram of the entire system with the design of the AR game, including the hardware system and software development, is shown in [Figure 1](#). In this project, the hardware system consists of two parts: HMD and vibrotactile feedback jacket. These are introduced in Sections 2.1 and 2.2, respectively.

### 2.1 Setup of the HMD

In this project, an Oculus HMD is used to create the AR environment in the eye view of game players, providing a real-world background with virtual objects and game scenes. The HMD adopts the Oculus Rift development kit 2 (DK2) and a high-speed camera for AR constructions. The setup of the HMD includes



**Figure 1** Block diagram of the entire system.

one unit of the Oculus Rift DK2 VR device, one unit of the Leap Motion body sensor, and one unit of a high-speed camera, as shown in Figure 2.

Compared with the Microsoft HoloLens AR kit, the cost of the HMD in this project is cheaper, as shown in Table 1 and the integration of virtual objects into real-world scenes is easier.

The process of the HMD system is shown in Figure 3. In the setup of the AR game, the Oculus Rift DK2 functions as the visualization display and gyroscope. The head rotation of the users is captured and feedback to the control computer. The real-world frames of the images taken from the high-speed camera and the virtual objects generated during the AR game execution are mixed in real time. The mixed video streams are geometrically positioned into Oculus DK2. The Leap Motion body sensor simultaneously tracks users' hand movements. The game players can interact with virtual objects in the AR game scenes through their hand movements.



**Figure 2** Setup of the HMD in this project.

**Table 1** Price comparison between the HMD adopted in this project and Microsoft HoloLens

Component list				Total price
Oculus HMD	Oculus DK2 (\$350)	Leap Motion sensor (\$100)	High-speed camera (\$60)	\$510
HoloLens	HoloLens (\$3000)/HoloLens 2 (\$3500)			\$3000/\$3500

## 2.2 Setup of the vibrotactile feedback jacket

The vibrotactile feedback jacket is used to provide synthesized feelings when virtual objects interact with the users' actions in different game scenes. As shown in Figure 3, the Arduino Mega 2560 controller board receives the serial communication signals sent by the computer. The Arduino Mega 2560 controller then controls the vibration actuators through the DC motor drivers. These actions generate the vibration feedback to the users on the vibrotactile feedback jacket in accordance with different AR game scenes. In this project, the vibrotactile feedback jacket developed consists of the following components: one unit of Arduino Mega 2560 controller and communication device, 14 units of brushless vibration actuator, 14 units of H-Bridge DC motor driver board, and a pair of AA battery.

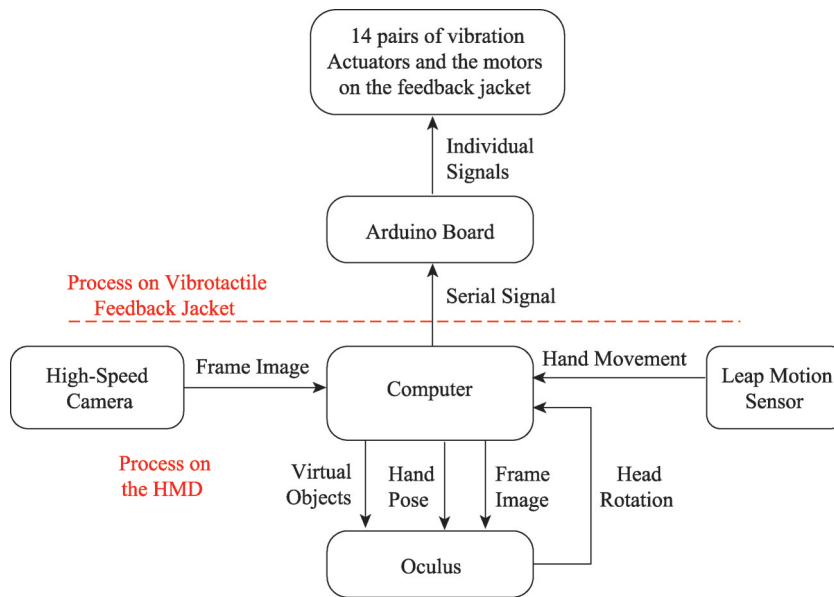


Figure 3 Working process of the HMD and vibrotactile feedback jacket.

The design diagram of the vibrotactile feedback jacket is shown in Figure 4. Powered by battery, the vibrotactile jacket has 14 vibration actuators with 14 corresponding DC motor drivers. The vibration actuators and DC motor drivers have small sizes. The vibrotactile feedback jacket uses pairs of one-actuator-one-driver modules to save space and simplify the wiring. The 14 vibration actuators with the motors cover the upper body of the users, which can be visualized in Figure 5. These vibration actuators are located at the following body position: four vibration actuators on the front side of the main body, four vibration actuators on the back side of the main body, two vibration actuators on the left arm, two vibration actuators on the right arm, one vibration actuator on the back of the left hand, and one vibration actuator on the back of the right hand.

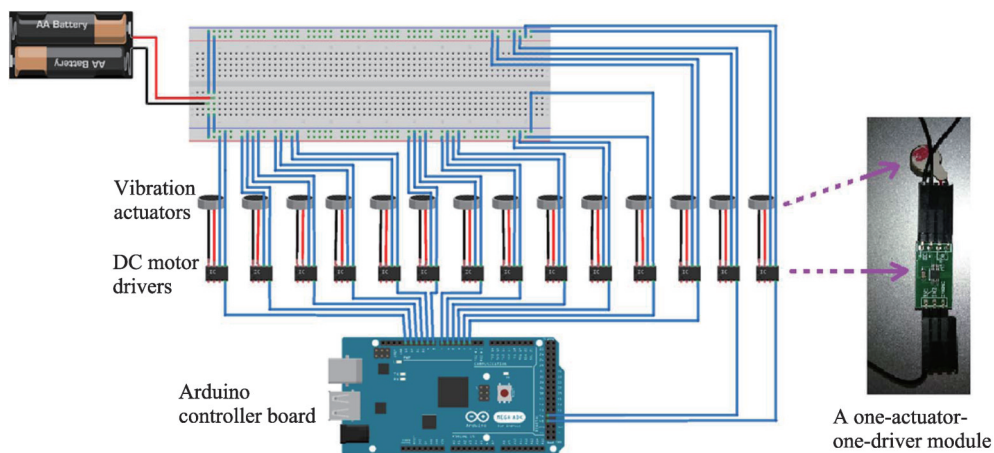


Figure 4 Design diagram of the vibrotactile feedback jacket.

### 3 Software development of AR game with vibrotactile feedback jacket

In this AR game, virtual objects can be preloaded into the game scenes to save processing time. The software development of the AR games with the vibrotactile feedback jacket consists of three parts: (1) game development, (2) controller behavior design, and (3) communication configurations.

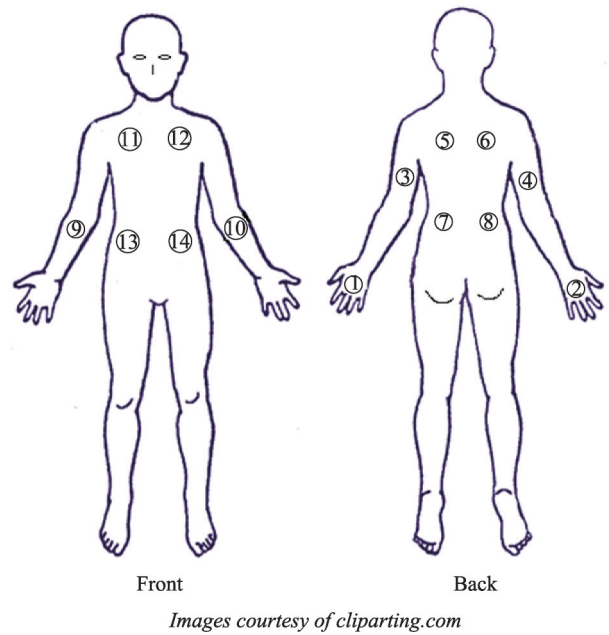
**Game development:** This part is mainly based on the Unity engine and C# language. The game scenes



are created based on a game script flowchart and then preprocessed for visual and performance optimization before it is streamed into the Oculus DK2 system. AR communication is realized with the users' hand poses in the game.

**Controller behavior design:** An Arduino board functions as a signal processor for the serious games. It receives, translates, error-corrects, and executes the signals from the game communication port that is linked to the hardware jacket presented in Section 2 (Figure 3). It provides the necessary vibration in line with the game progress.

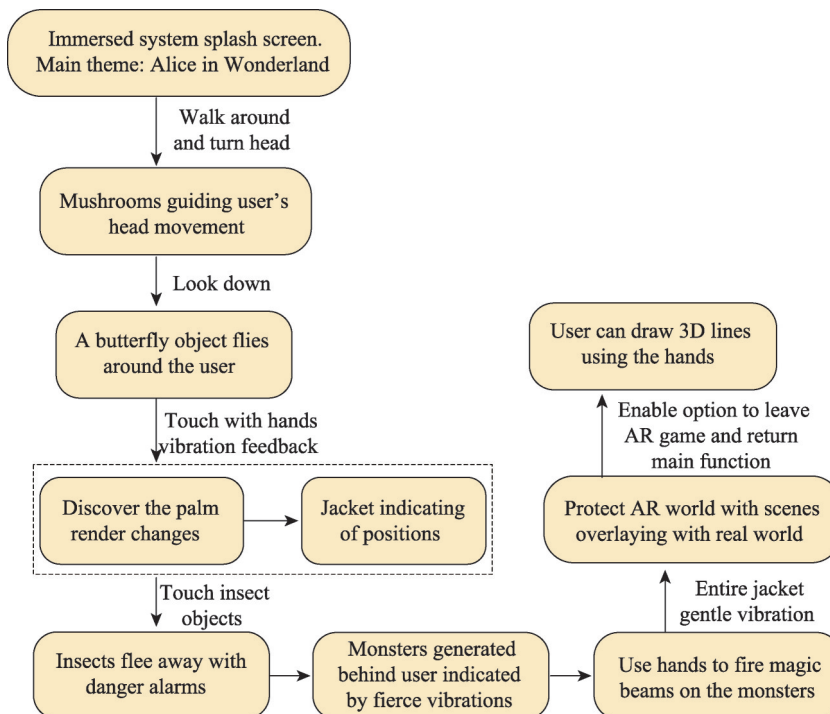
**Communication configurations:** It is designed with a high process speed and sufficient robustness, with a defined signal structure and transmission path.



**Figure 5** Positions of the vibration actuators on the vibrotactile feedback jacket.

### 3.1 AR game development

In this project, the scenes and virtual characters in the game include a fairy tale, with amazingly giant plants and insects. Some virtual characters are friendly, whereas others are aggressive to the gamers. The players can interact, touch, or even fight back the virtual characters. The flow chart of the script of the serious game with the vibrotactile feedback jacket is shown in Figure 6.



**Figure 6** Script of AR game with vibrotactile feedback jacket.

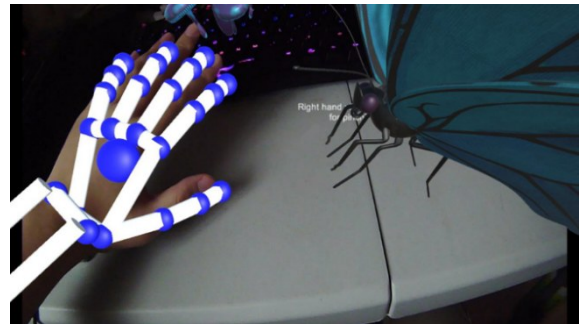
The scenes of the AR game comprise several virtual game objects and interaction functions. The functions are controlled through hand-gesture recognitions. These fundamental scene components and their descriptions are listed in Table 2. They translate the game script into a series of playing stages and create the AR world.

**Table 2 Virtual objects and interaction functions in game scenes**

Virtual Objects	Descriptions
Butterfly	Peaceful objects to the players/users. Only transfer position data and respond to users' touch events
Spider	Objects marked as enemies, with attack detection. Communication with hand magic effects, transfer data, and promote game progress
Mushroom	Environmental objects, for direction reference
Interaction Functions	
Double-click	On/off control for the magic particle beam trigger: The left hand shoots fire beam, whereas the right hand shoots ice beam
Touch	Touch function with butterfly objects
Pinch	Users can draw 3D virtual lines following their pinching fingers
Open palm	When the palm is open, together with the double-click "On" state, the hand pushes out a magic beam

In the AR game, vibration signals are triggered in some interaction scenes. Such a triggering is generated using the vibrotactile feedback jacket for players. There are several types of interactions involved in the AR game.

(1) Relative position of virtual butterfly objects: It is used to trigger the corresponding vibration actuators with variable strength on the vibrotactile feedback jacket indicating the butterfly's movements. The game scene of the wandering virtual butterfly is shown in [Figure 7](#).

**Figure 7 Virtual hand and wandering butterfly.**

(2) Relative position of virtual spider objects: Similarly, the corresponding vibration actuators on the vibrotactile feedback jacket are triggered to indicate the spider's movements. The game scene of the attacking virtual spider is shown in [Figure 8](#).

**Figure 8 Virtual spider appearing and attacking.**

(3) Attack action of virtual spider objects: It triggers the corresponding vibration actuators on the vibrotactile feedback jacket for the attack action of enemies, i.e., the virtual spider objects. Concurrently, fiercer vibration density and larger vibration area are generated by more actuators, compared with the vibrations caused by the relative position of virtual objects.

(4) Hand gesture control: There are two vibration actuators at each arm of a player and one vibration actuator at each hand on the vibrotactile feedback jacket. When shooting magic beams, all the vibration actuators on the detected arm and hand are triggered. For pinch drawing, only the vibration actuator on the hand vibrates, with a gentler vibration.

The game scenes in the computer are played as frames of images, which are two-dimensional in space. Preprocessing on these frames is required before passing them into Oculus for visualization. Oculus has lenses that magnify the images to provide a large field of view. However, these lenses also give a pincushion distortion effect to the images, which significantly affects the immersive feeling shown below in [Figure 9a](#). Hence, it is required to exactly apply the distortion correction effect to create an undistorted frame of images, as shown in [Figure 9b](#).

In this project, distortions on the frame of images are caused by two components: the high-speed camera module and Oculus DK2. The mathematical formulas to rectify the image distortions in this project are shown in Eqs. (1) and (2), based on a study conducted by Zhang<sup>[36]</sup>. The mathematical expressions are implemented into the Unity C# script, which can manually adjust distortion parameters to meet the project requirement.

$$x_d = x_u(1 + K_1r^2 + K_2r^4 + \dots) + (P_2(r^2 + 2x_u^2) + 2P_1x_ux_u)(1 + P_3r^2 + P_4r^4 + \dots) \quad (1)$$

$$y_d = y_u(1 + K_1r^2 + K_2r^4 + \dots) + (P_1(r^2 + 2y_u^2) + 2P_2x_ux_u)(1 + P_3r^2 + P_4r^4 + \dots) \quad (2)$$

where:

$(x_d, y_d)$  = distorted image point as projected on the image plane using specified lens;

$(x_u, y_u)$  = undistorted image point as projected using an idea pin-hole camera;

$(x_c, y_c)$  = distortion centre (assumed to be the principal point);

$K_n = n^{\text{th}}$  radial distortion coefficient;

$P_n = n^{\text{th}}$  tangential distortion coefficient;

$r = \sqrt{(x_u - x_c)^2 + (y_u - y_c)^2}$ ;

$\dots$  = infinite series.

There are high demands of computations on image processing and distortion correction. To realize real-time distortion correction, a mathematical computation is allocated and executed in the graphics processing unit (GPU). The Unity C# script for image distortion correction implemented in the project is shown in Figure 10. The parameter adjustment panel for the distortion correction is shown in Figure 11.

After the distortion correction, the frames of images and the game played in Oculus have an improved overall feeling. However, owing to the parameter deviations, perfect flat images cannot be achieved with the distortion correction.

The AR game scenes are integrated with the Leap Motion body sensor and high-speed camera.

The Leap Motion body sensor detects users' hands and provides the necessary information on the hand trajectories and poses. In this manner, the rendering of the virtual hand and arm can be achieved corresponding to the real hands' positions. Moreover, predefined hand poses can be memorized for special uses, as previously mentioned in the scene-interaction functions.

The high-speed camera is used to provide a real-world background. It is the key component to create AR effects because virtual objects in the game must visually match in the real world without any delay or with minimal delay. The camera is selected with a high-speed frame rate to run with Oculus at a refresh rate of 75Hz. The camera can reach 60 frames per second (FPS) at  $1280 \times 720$  resolution. It is selected as the

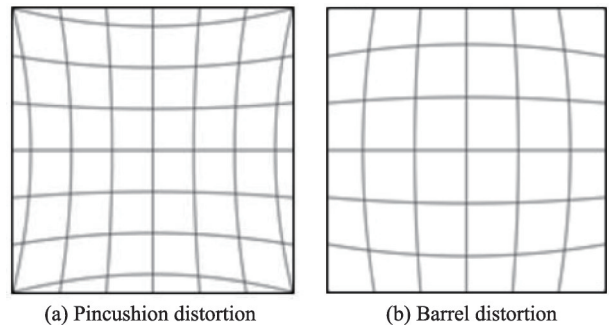


Figure 9 Image distortions and correction.

```
float2 DistortPixel(float2 points)
{
    float Inverse = Invert?-1:1;
    float cx = LensOffsetX;
    float cy = LensOffsetY;
    float k1 = RadialDistortionX * Inverse;
    float k2 = RadialDistortionY * Inverse;
    float p1 = TangentialDistortionX * Inverse;
    float p2 = TangentialDistortionY * Inverse;
    float k3 = K5Distortion * Inverse;

    float x = points.x - cx;
    float y = points.y - cy;
    float r2 = x*x + y*y;

    // Radial distortion
    float xDistort = x * (1 + k1 * r2 + k2 * r2 * r2 + k3 * r2 * r2 * r2);
    float yDistort = y * (1 + k1 * r2 + k2 * r2 * r2 + k3 * r2 * r2 * r2);

    // Tangential distortion
    xDistort = xDistort + (2 * p1 * x * y + p2 * (r2 + 2 * x * x));
    yDistort = yDistort + (p1 * (r2 + 2 * y * y) + 2 * p2 * x * y);

    // Back to absolute coordinates.
    xDistort = xDistort + cx;
    yDistort = yDistort + cy;

    return float2( xDistort, yDistort);
}
```

Figure 10 Unity C# script for distortion correction.



governing resolution for the AR game to minimize the time delay.

### 3.2 Controller behavior design of the developed game

Commercially available and featuring high speed and simple operations, an Arduino board is used as a controller that is configured for serial communication in this project. The controller behavior design includes a signal process flow in five stages: data receiving, data translation, error check and correction, execution, and buffer clear. The last stage is "buffer clear" for performance optimization.

(1) Data receiving: Vibration signals generated in the game scenes are regulated and transmitted out as universal serial bus (USB) communication signals in ASCII format. The signals are received by the controller Arduino board through its USB serial port. Every package of serial communication signals ending with "\n" is stored in the controller buffer for the next processing stage. The data package of signals is in the ASCII format. The vibration signals generated from the AR game process contain 14 ASCII characters, with the structure shown in Figure 12.

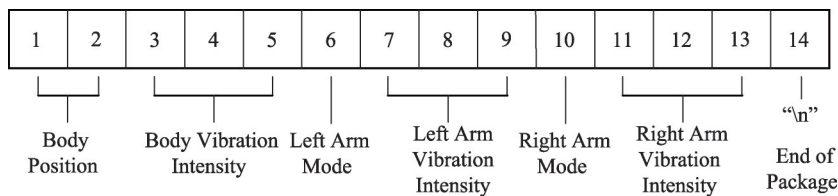


Figure 12 Data structure of signal packages.

(2) Data translation: The serial communication signals generated from the first stage must be translated from ASCII format into mathematical values in this stage. The translation is obtained using the code shown in Figure 13.

```
void infor(char info[20]){
    re.body_pos = 10 * ( info[0] - '0' ) + ( info[1] - '0' );
    re.body_str = 100 * ( info[2] - '0' ) + 10 * ( info[3] - '0' ) + ( info[4] - '0' ); //information of Body vibration

    re.Arm.arm_str_l = 100 * ( info[6] - '0' ) + 10 * ( info[7] - '0' ) + ( info[8] - '0' );
    re.Arm.arm_str_r = 100 * ( info[10] - '0' ) + 10 * ( info[11] - '0' ) + ( info[12] - '0' ); //information of both arms' vibration intensity

    re.Arm.mode_l = (bool) ( info[5] - '0' );
    re.Arm.mode_r = (bool) ( info[9] - '0' ); //information of arm vibration mode
}
```

Figure 13 Data translation function.

(3) Error check and correction: Sometimes, jitters occur in the USB serial communication resulting in faulty data, such as data "000" being transmitted as "€00." These types of corrupted non-numerical characters are treated as zeros. As such, these corrupted characters and their relevant data packages are wiped out and lost. The current execution loop is suspended for the security reason. The execution is released in the next loop.

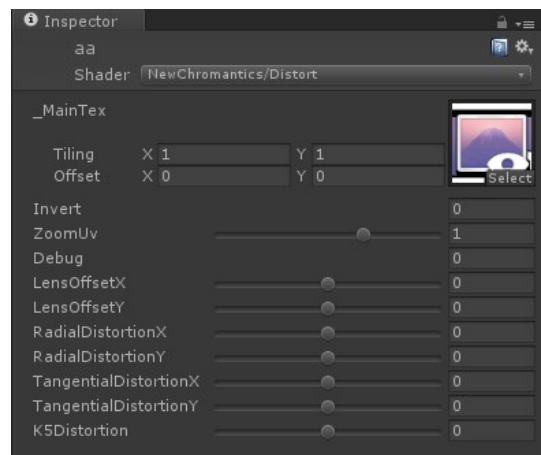


Figure 11 Distortion parameter panel.

(4) Execution: The translated signals are recognized by the system. The relevant functions of the AR game are activated.

(5) Buffer clearing: In this game design project, signals are sent at extremely high frequency, causing a data buffer overflow in the Arduino board. The Arduino board stores and processes data with a First-In-First-Out (FIFO) strategy. The received signal packages are stored in the FIFO data buffer queuing for further executions. This process causes a severe delay as the package of signals received at one moment is only executed sometime later after going through the FIFO data buffer. To solve this problem, a line of function code `while (Serial.read() != -1)` is inserted at the end of the main program; this clears the data buffer storage before receiving the next signals.

### 3.3 Communication configuration of the AR game

Packages of data signals start their generation once the game is launched. Three seconds after the game program starts execution, these packages of serial communication signals are transmitted out through a USB port into the hard disk. If the data stream of the generated signal packages is continuously received at a considerably high speed, then the hard disk is overloaded and too busy with the communication data stream as observed during the test runs in the project. The interval time of signals communication per step is 30ms, as a tradeoff for the interval time and hard disk load to improve the system's performance.

When receiving packages of signals related to the triggering of corresponding vibration actuators on the vibrotactile feedback jacket, the vibration actuators should immediately start the functioning of vibrations. However, in reality, the vibration actuators require the reaction time to start from the static status to fully function. The reaction time of the vibration actuators is shown in Eq. (3).

$$\text{Reaction Time} = \text{Time}_{\text{interval}} + \text{Time}_{\text{rising}} + \text{Time}_{\text{communication}} \quad (3)$$

The interval time  $\text{Time}_{\text{interval}}$  is 30ms, as mentioned earlier. The rising time  $\text{Time}_{\text{rising}}$  is a turn-on time of the vibration actuators before they fully function. The investigation and measurements are conducted for the vibration actuators in this project, as shown in Figure 14. The rising time of the vibration actuators is approximately 140ms. The notation  $\text{Time}_{\text{communication}}$  denotes the communication latency of the software driver, which is 16ms by default as assigned by the Microsoft driver. The reaction time of the vibration actuators is  $30\text{ms} + 140\text{ms} + 16\text{ms} = 186\text{ms}$ .

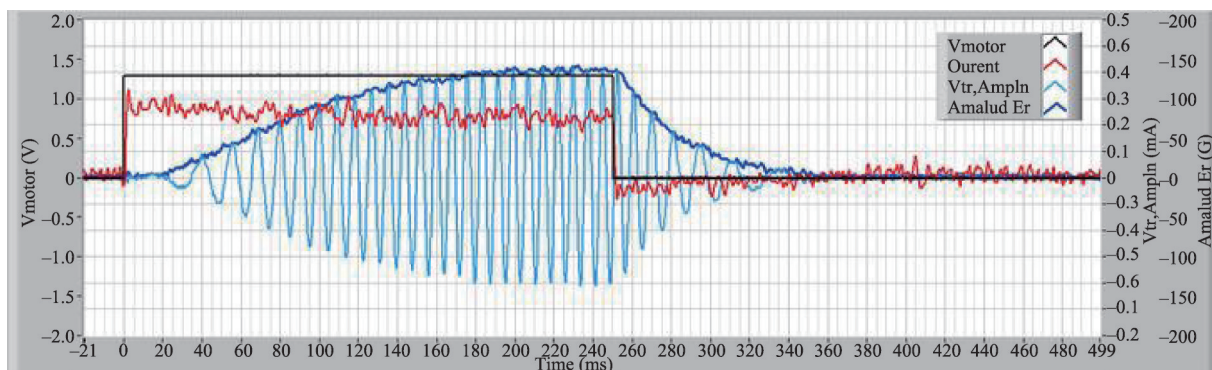


Figure 14 Turn-on time  $\text{Time}_{\text{rising}}$  of the vibration actuator.

Research on the human reaction speed suggests that the median reaction time for human is 273ms<sup>[37]</sup>. The reaction time of the vibration actuators in our implementation is less than the human reaction time. Hence, the vibrotactile feedback by the vibration actuators is sufficiently fast for game players, without feeling any delay.

## 4 Results and performance evaluation

The AR game is tested and evaluated. During the test runs and demonstrations, the game scenes flow through the multiple stages. Based on the performance evaluations, the heaviest computational load occurs in the game scenes with the hand object firing magic beams, as shown in Figure 15. Using Unity engine's profiler function, the game performance and utilized resources can be visualized in Figures 16 and 17, illustrating the differences in the usages of the central processing unit (CPU) (e.g., rendering, scripts, and physics) and GPU (e.g., opaque, transparent, and shadows/depth) between normal game scenes and game scenes with the hand object firing magic beams.

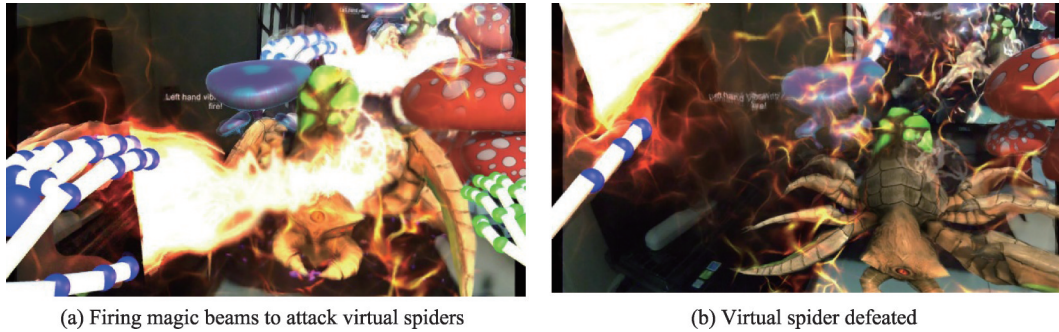


Figure 15 Game scenes with hand object firing magic beams to attack virtual spiders.

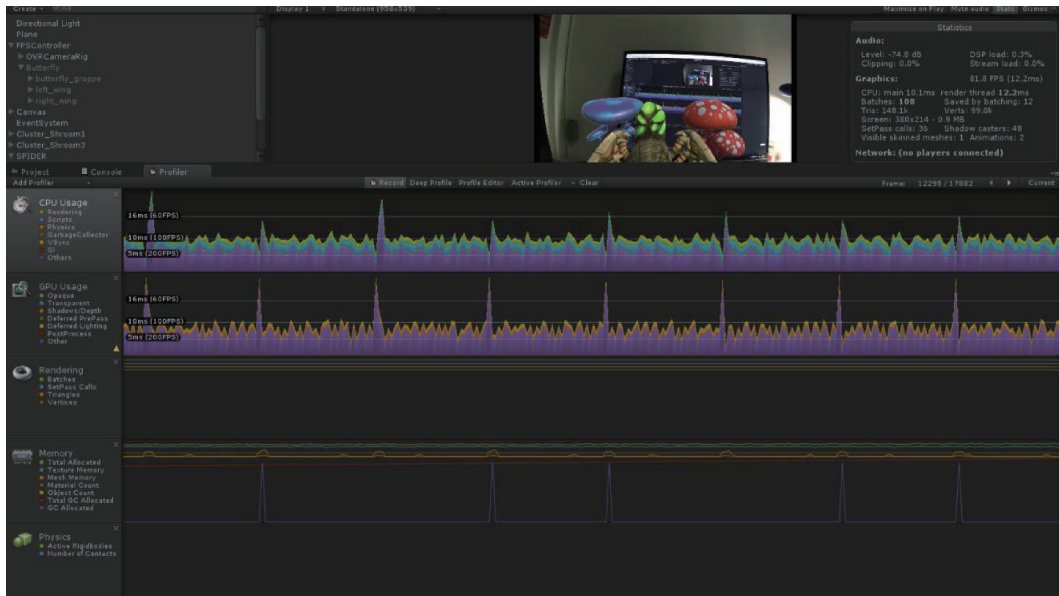


Figure 16 Unity engine's profiler for performance of normal game scenes.

The quantified computational load differences for normal game scenes and game scenes with the hand object firing magic beams are listed in Table 3. In the game scenes for the hand object firing magic beams, the FPS reduces by approximately 32%. The major reason for this FPS reduction is the particle system used with the magic beam, which requires a considerable amount of CPU resources. The increase in CPU resource consumptions caused the increase in the CPU time and render thread time, particularly the significant increase of the number of triangles (Tris), vertices (Verts), and shadow casters.

In addition to the game scenes of the hand object firing magic beams, the second heaviest computational load occurs at the rendering of the skeleton capsule hand object, as players use both skeleton hands to draw multiple 3D lines in real time in the AR game. These 3D lines are drawn to visualize and track the

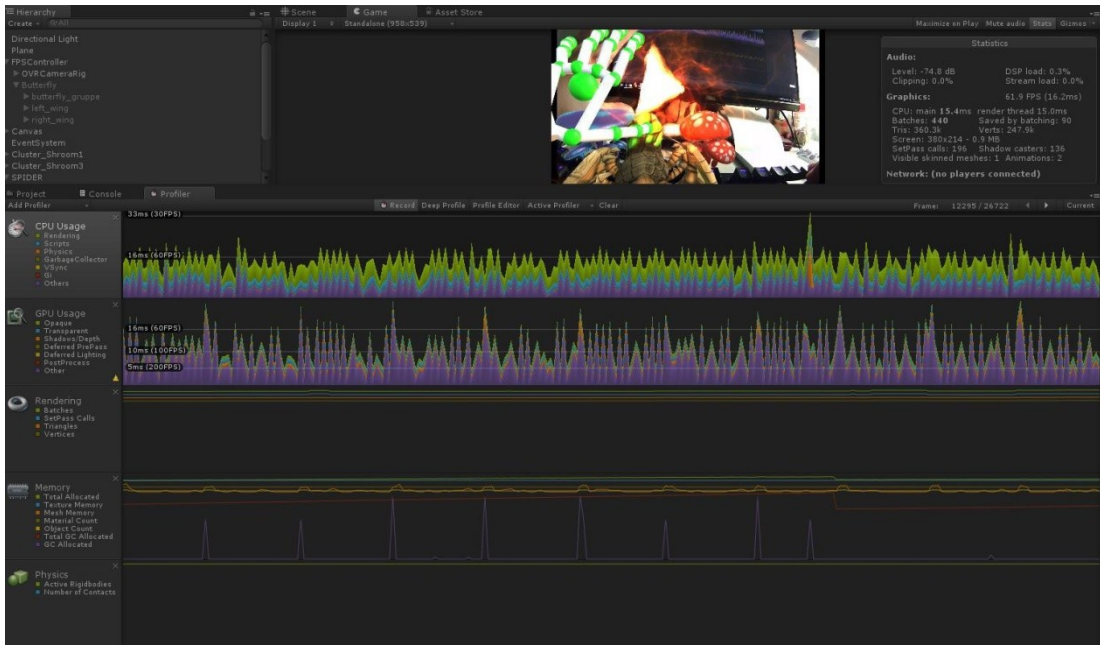


Figure 17 Unity engine's profiler for game scenes with hand object firing magic beams.

Table 3 Computational resources utilized at different game scenes

	Normal game scenes	Scenes for firing magic beams	Load increase (%)
FPS	81.8	61.9	32.1486
CPU time (ms)	10.1	15.4	52.48
Render thread (ms)	12.2	15.0	22.95
Tris ('000)	148.1	360.3	143.28
Verts ('000)	99.0	247.9	150.4
Shadow casters	48	136	183.33

movement of the skeleton hands in certain periods to interact with the game characters. The skeleton capsule hand objects are completely generated on calculations shown in Figure 18. Its rendering process is more complex than that in the normal game scenes.

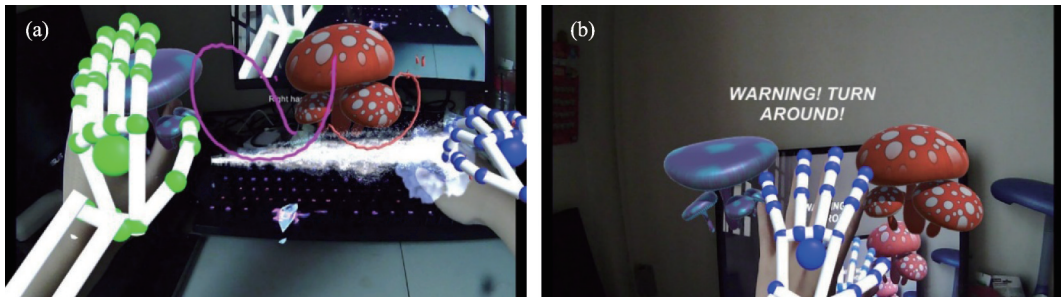


Figure 18 Rendering of the skeleton capsule hand object and drawing 3D lines: (a) draw 3D lines, (b) interact with virtual objects.

The vibration of the vibrotactile feedback jacket is fast responding. There are different vibration intensities at different positions that provide vibration feedback. However, the vibration makes noise on the vibrotactile feedback jacket when functioning, which causes difficulty to feel at considerably low vibration intensities. This vibration noise issue may be generated by the material inside the vibrotactile feedback



jacket. The lining of the vibrotactile feedback jacket is made of hard foam that can absorb energy from a collision. When the vibration actuators start working, the vibration may hit at the hard surface of the foam, creating noise. In this case, the surface vibration can be considered a kind of collision, which is then absorbed by the foam material.

The comparisons of the vibrotactile feedback jacket prototype with other vibrotactile vest or jacket are shown in Table 4. The vibration effects can be generated by different types of vibration devices with motors, actuators, transducers, or airbags. The dimensions of the devices can bring the differences. As shown in Table 4, the number of actuators varies from 2 to 54, depending on the body areas to be covered by the feedback jacket. The higher the number of actuators, the higher is the design costs and the higher is the vibration levels. The number of actuators causes more complicated wiring and system design. For the vibrotactile feedback jacket proposed in this project, various vibration patterns and intensity levels are generated in different game scenes. It can achieve more levels of vibration resolutions for the actuators, as compensation for the limited number of the actuators. Table 4 also shows that most of the feedback jackets cover the body torso. The proposed vibrotactile feedback jacket covers the game players' front side torso, back side torso, arms, and hands. Better user experiences can be achieved if the feedback jacket to cover more body areas. Moreover, as shown in Table 4, some feedback jackets are commercially available. However, their prices are higher than US\$100. The prices and quality should be improved to attract more players to adopt these kinds of feedback jackets for AR/VR games.

**Table 4 Comparisons of proposed vibrotactile feedback jacket with other vests or jackets**

	Number of Actuators	Body positions to cover	Actuation method	Price
Vibrotactile feedback jacket in this study	14 actuators	Torso, both arms, both hands	Vibration	N/A
Haptic vest <sup>[20]</sup>	54 for large size, 38 for medium size	Torso	Vibration, thermoelectric	N/A
Force jacket <sup>[27]</sup>	26 airbags and force	Torso	Vibration	N/A
Force-feedback suit <sup>[28]</sup>	7 actuators	Torso	Pneumatic	N/A
KOR-FX vest <sup>[32]</sup>	2 transducers	Around chest region	Vibration	US\$135
TACTOT vest <sup>[33]</sup>	40 actuators	Front torso, back torso	Vibration	US\$499
Vest Pro <sup>[34]</sup>	8 transducers	Front torso, back torso	Vibration	US\$599
TESLASUIT <sup>[35]</sup>	46 actuators	Torso, arms, legs	Vibration	N/A

## 5 Conclusions

### 5.1 Contributions

This article presents a design of an AR serious game enabling the interaction using a vibrotactile feedback jacket and an HMD. It also presents the hardware design of the vibrotactile jacket and software development of the AR serious game. The game scenes, game scripts, and signal processing flow are also described in detail. The graphics computations are processed by the GPU in the system. The vibrotactile feedback jacket provides real-world feedback in accordance with different AR game scenes. With 14 vibration actuators placed at different positions on the torso, arms, and hands, the vibrotactile feedback jacket can be triggered by a multimodal AR environment and game scenes. The vibrotactile jacket can generate various vibration patterns and intensity levels. The evaluation results of the developed AR serious game prototype are also discussed in this article. The prototype of the vibrotactile feedback jacket and AR serious game can be fundamental blocks to build larger-scale AR serious games.



## 5.2 Further possible improvements

The current prototype of the AR game is fairly complete and works smoothly for most general operations. However, it has limited game scripts, virtual objects, interaction functions, and vibration events with the vibrotactile feedback jacket. Hence, the prototype should be further improved to address these limitations and to make the system more comprehensive and attractive to gamers.

(1) The AR game script and game scene design can be enhanced by more complicated virtual objects. A few examples of virtual objects include virtual environmental decorations, butterfly-type neutral creatures, and spider-type enemy creatures. These virtual objects can be further defined with more behaviors under different scenarios. Artificial intelligence components can also be added to make the AR game more realistic and intelligent.

(2) More hand movements and gestures can be further developed to increase the AR interaction functions. These possible hand gestures include hand swiping, tapping, and on-hand virtual panels.

(3) The vibrotactile feedback jacket can be further improved using more compact digital circuits and chipsets for wireless communication modules, controllers, actuator pulse width modulation drivers, and power regulators. Such improvements can save a considerable amount of space and significantly reduce the wiring difficulties of the vibrotactile feedback jacket. More actuators can be added into the jacket to cover more positions on the user's body, arms, and hands. The connectivity can be enhanced into a Bluetooth wireless connection.

(4) In this project, the approach for image distortion correction requires expert knowledge, careful measurements, and time-consuming annotations to derive suitable camera parameters<sup>[38]</sup>. It is a tedious task. Automatic camera calibration algorithms for image distortion correction have been reported in several studies<sup>[39,40]</sup>. Such approaches with automatic image distortion correction are more practical in various circumstances<sup>[41]</sup>. In the future development of the project, off-the-shelf camera calibration algorithms for image distortion correction may be explored.

(5) The graphics rendering and computations of the AR game can be further optimized. One possible solution is to use multi-core parallel computing, which can nearly double the computation efficiency.

(6) With the developed prototype of the vibrotactile feedback jacket with AR serious games, it will be good to gain users' opinions and feedback through usability studies with a large number of learners or players to improve these AR serious games.

**Declaration of Competing Interest** The authors declare no competing financial interest.

## References

- 1 Cai Y. 3D Immersive & Interactive Learning. Singapore: Springer Verlag, 2013  
DOI:[10.1007/978-981-4021-90-6](https://doi.org/10.1007/978-981-4021-90-6)
- 2 Xie Y, Zhang Y Z, Cai Y Y. Virtual reality engine disassembly simulation with natural hand-based interaction. In: VR, Simulations and Serious Games for Education. Singapore: Springer Singapore, 2018, 121–128  
DOI:[10.1007/978-981-13-2844-2\\_11](https://doi.org/10.1007/978-981-13-2844-2_11)
- 3 Orozco M, Silva J, El A, Petriu E. The role of haptics in games. In: Haptics Rendering and Applications. InTech, 2012, 217–234  
DOI:[10.5772/32809](https://doi.org/10.5772/32809)
- 4 Barajas A O, Al Osman H, Shirmohammadi S. A Serious Game for children with Autism Spectrum Disorder as a tool for play therapy. In: 2017 IEEE 5th International Conference on Serious Games and Applications for Health (SeGAH). Perth, WA, Australia, IEEE, 2017, 1–7

- DOI:[10.1109/segah.2017.7939266](https://doi.org/10.1109/segah.2017.7939266)
- 5 DiPietro J, Kelemen A, Liang Y L, Sik-Lanyi C. Computer- and robot-assisted therapies to aid social and intellectual functioning of children with autism spectrum disorder. *Medicina*, 2019, 55(8): 440  
DOI:[10.3390/medicina55080440](https://doi.org/10.3390/medicina55080440)
  - 6 Tsikinas S, Xinogalos S. Studying the effects of computer serious games on people with intellectual disabilities or autism spectrum disorder: a systematic literature review. *Journal of Computer Assisted Learning*, 2019, 35(1): 61–73  
DOI:[10.1111/jcal.12311](https://doi.org/10.1111/jcal.12311)
  - 7 Lu A, Chan S, Cai Y Y, Huang L H, Nay Z T, Goei S L. Learning through VR gaming with virtual pink dolphins for children with ASD. *Interactive Learning Environments*, 2018, 26(6): 718–729  
DOI:[10.1080/10494820.2017.1399149](https://doi.org/10.1080/10494820.2017.1399149)
  - 8 Cai Y Y, Chiew R, Nay Z T, Indhumathi C, Huang L H. Design and development of VR learning environments for children with ASD. *Interactive Learning Environments*, 2017, 25(8): 1098–1109  
DOI:[10.1080/10494820.2017.1282877](https://doi.org/10.1080/10494820.2017.1282877)
  - 9 Tang J S Y, Falkmer M, Chen N T M, Bölte S, Girdler S. Designing a serious game for youth with ASD: perspectives from end-users and professionals. *Journal of Autism and Developmental Disorders*, 2019, 49(3): 978–995  
DOI:[10.1007/s10803-018-3801-9](https://doi.org/10.1007/s10803-018-3801-9)
  - 10 Borresen A, Wolfe C, Lin C, Tian Y, Raghuraman S, Nahrstedt K, Prabhakaran B, Annaswamy T M. Usability of an immersive augmented reality based telerehabilitation system with haptics (ARTESH) for synchronous remote musculoskeletal examination. *International Journal of Telerehabilitation*, 2019, 11(1): 23–32  
DOI:[10.5195/ijt.2019.6275](https://doi.org/10.5195/ijt.2019.6275)
  - 11 Culbertson H, Kuchenbecker K J. Ungrounded haptic augmented reality system for displaying roughness and friction. *IEEE/ASME Transactions on Mechatronics*, 2017, 22(4): 1839–1849  
DOI:[10.1109/tmech.2017.2700467](https://doi.org/10.1109/tmech.2017.2700467)
  - 12 Maisto M, Pacchierotti C, Chinello F, Salvietti G, de Luca A, Prattichizzo D. Evaluation of wearable haptic systems for the fingers in augmented reality applications. *IEEE Transactions on Haptics*, 2017, 10(4): 511–522  
DOI:[10.1109/toh.2017.2691328](https://doi.org/10.1109/toh.2017.2691328)
  - 13 Pridhvi Krishna M V, Mehta S, Verma S, Rane S. Mixed reality in smart computing education system. In: 2018 International Conference on Smart Systems and Inventive Technology (ICSSIT). Tirunelveli, India, IEEE, 2018, 72–75  
DOI:[10.1109/icssit.2018.8748813](https://doi.org/10.1109/icssit.2018.8748813)
  - 14 Bhargava A, Bertrand J W, Gramopadhye A K, Madathil K C, Babu S V. Evaluating multiple levels of an interaction fidelity continuum on performance and learning in near-field training simulations. *IEEE Transactions on Visualization and Computer Graphics*, 2018, 24(4): 1418–1427  
DOI:[10.1109/tvcg.2018.2794639](https://doi.org/10.1109/tvcg.2018.2794639)
  - 15 Johnson C I, Bailey S K T, van Buskirk W L. Designing effective feedback messages in serious games and simulations: A research review. In: *Instructional Techniques to Facilitate Learning and Motivation of Serious Games*. Cham: Springer International Publishing, 2016, 119–140  
DOI:[10.1007/978-3-319-39298-1\\_7](https://doi.org/10.1007/978-3-319-39298-1_7)
  - 16 Menelas B A, Benaoudia R. Use of haptics to promote learning outcomes in serious games. *Multimodal Technologies and Interaction*, 2017, 1(4): 31  
DOI:[10.3390/mti1040031](https://doi.org/10.3390/mti1040031)
  - 17 Vaquero-Melchor D, Bernardos A M. Enhancing interaction with augmented reality through mid-air haptic feedback: architecture design and user feedback. *Applied Sciences*, 2019, 9(23): 5123  
DOI:[10.3390/app9235123](https://doi.org/10.3390/app9235123)
  - 18 Lee J, Kim Y, Kim G J. Effects of visual feedback on out-of-body illusory tactile sensation when interacting with augmented virtual objects. *IEEE Transactions on Human-Machine Systems*, 2017, 47(1): 101–112  
DOI:[10.1109/thms.2016.2599492](https://doi.org/10.1109/thms.2016.2599492)
  - 19 de Jesus Oliveira V A, Brayda L, Nedel L, Maciel A. Designing a vibrotactile head-mounted display for spatial awareness in 3D spaces. *IEEE Transactions on Visualization and Computer Graphics*, 2017, 23(4): 1409–1417

- DOI:[10.1109/tvcg.2017.2657238](https://doi.org/10.1109/tvcg.2017.2657238)
- 20 Garcıavalle G, Ferre M, Brenosa J, Vargas D. Evaluation of presence in virtual environments: haptic vest and user's haptic skills. *IEEE Access*, 2018, 6: 7224–7233  
DOI:[10.1109/access.2017.2782254](https://doi.org/10.1109/access.2017.2782254)
  - 21 Rantala J, Majaranta P, Kangas J, Isokoski P, Akkil D, Špakov O, Raisamo R. Gaze interaction with vibrotactile feedback: review and design guidelines. *Human-Computer Interaction*, 2020, 35(1): 1–39  
DOI:[10.1080/07370024.2017.1306444](https://doi.org/10.1080/07370024.2017.1306444)
  - 22 Shim Y A, Lee J, Lee G. Exploring multimodal watch-back tactile display using wind and vibration. In: *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. Montreal QC, Canada, New York, ACM Press, 2018, 1–12  
DOI:[10.1145/3173574.3173706](https://doi.org/10.1145/3173574.3173706)
  - 23 Kaul O B, Rohs M. HapticHead: 3D guidance and target acquisition through a vibrotactile grid. In: *Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems*. California, USA, New York, ACM Press, 2016, 2533–2539  
DOI:[10.1145/2851581.2892355](https://doi.org/10.1145/2851581.2892355)
  - 24 Young E M, Memar A H, Agarwal P, Colonnese N. Bellowband: a pneumatic wristband for delivering local pressure and vibration. In: *2019 IEEE World Haptics Conference (WHC)*. Tokyo, Japan, IEEE, 2019, 55–60  
DOI:[10.1109/whc.2019.8816075](https://doi.org/10.1109/whc.2019.8816075)
  - 25 Wolf D, Rietzler M, Hnatek L, Rukzio E. Face/on: multi-modal haptic feedback for head-mounted displays in virtual reality. *IEEE Transactions on Visualization and Computer Graphics*, 2019, 25(11): 3169–3177  
DOI:[10.1109/tvcg.2019.2932215](https://doi.org/10.1109/tvcg.2019.2932215)
  - 26 Louison C, Ferlay F, Mestre D R. Spatialized vibrotactile feedback contributes to goal-directed movements in cluttered virtual environments. In: *2017 IEEE Symposium on 3D User Interfaces (3DUI)*. Los Angeles, CA, USA, IEEE, 2017, 99–102  
DOI:[10.1109/3dui.2017.7893324](https://doi.org/10.1109/3dui.2017.7893324)
  - 27 Delazio A, Nakagaki K, Klatzky R L, Hudson S E, Lehman J F, Sample A P. Force jacket: pneumatically-actuated jacket for embodied haptic experiences. In: *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. Montreal QC, Canada, New York, ACM Press, 2018  
DOI:[10.1145/3173574.3173894](https://doi.org/10.1145/3173574.3173894)
  - 28 Kishishita Y, Das S, Ramirez A V, Thakur C, Tadayon R, Kurita Y. Muscleblazer: force-feedback suit for immersive experience. In: *2019 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*. Osaka, Japan, IEEE, 2019, 1813–1818  
DOI:[10.1109/vr.2019.8797962](https://doi.org/10.1109/vr.2019.8797962)
  - 29 Rognon C, Koehler M, Duriez C, Floreano D, Okamura A M. Soft haptic device to render the sensation of flying like a drone. *IEEE Robotics and Automation Letters*, 2019, 4(3): 2524–2531  
DOI: [10.1109/LRA.2019.2907432](https://doi.org/10.1109/LRA.2019.2907432)
  - 30 Ouyang Q Q, Wu J, Wu M. Vibrotactile display of flight attitude with combination of multiple coding parameters. *Applied Sciences*, 2017, 7(12): 1291  
DOI:[10.3390/app7121291](https://doi.org/10.3390/app7121291)
  - 31 Günther S, Makhija M, Müller F, Schön D, Mühlhäuser M, Funk M. PneumAct: pneumatic kinesthetic actuation of body joints in virtual reality environments. In: *Proceedings of the 2019 on Designing Interactive Systems Conference*. San Diego, CA, USA, ACM, 2019, 227–240  
DOI:[10.1145/3322276.3322302](https://doi.org/10.1145/3322276.3322302)
  - 32 ImmerzInc, product of KOR-FX [2020-02-20]. <http://www.korfx.com/products>
  - 33 bHapticsInc, product of TACTOT, haptic vest for torso [2020-02-20]. <https://www.bhaptics.com/tactsuit/>
  - 34 Woojer USAInc, product of Vest Pro [2020-02-20]. <https://www.woojer.com/vest/>
  - 35 VRElectronics Ltd, product of TESLASUIT [2020-02-20]. <https://teslasuit.io/the-suit/>
  - 36 Zhang Z. A Flexible New Technique for Camera Calibration (Technical report). Microsoft Research. MSR-TR-98-71,

1998

- 37 Reaction Time Statistics (2020) [2020-02-20]. <http://www.humanbenchmark.com/tests/reactiontime/statistics>
- 38 Tang Z, Lin Y, Lee K, Hwang J, Chuang J. ESTHER: joint camera self-calibration and automatic radial distortion correction from tracking of walking humans. *IEEE Access*, 2019, 7: 10754–10766  
DOI:[10.1109/access.2019.2891224](https://doi.org/10.1109/access.2019.2891224)
- 39 Zhang Z D, Matsushita Y, Ma Y. Camera calibration with lens distortion from low-rank textures. In: *CVPR 2011*. Providence, RI, USA, IEEE, 2011, 2321–2328  
DOI:[10.1109/cvpr.2011.5995548](https://doi.org/10.1109/cvpr.2011.5995548)
- 40 Tehrani M A, Beeler T, Grundhöfer A. A practical method for fully automatic intrinsic camera calibration using directionally encoded light. In: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Honolulu, HI, USA, IEEE, 2017, 125–133  
DOI:[10.1109/cvpr.2017.21](https://doi.org/10.1109/cvpr.2017.21)
- 41 Kakani V, Kim H, Lee J, Ryu C, Kumbham M. Automatic distortion rectification of wide-angle images using outlier refinement for streamlining vision tasks. *Sensors*, 2020, 20(3): 894  
DOI:[10.3390/s20030894](https://doi.org/10.3390/s20030894)