IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 7 December 2023, accepted 28 December 2023, date of publication 1 January 2024,
date of current version 10 January 2024.

Digital Object Identifier 10.1109/ACCESS.2023.3348846

RESEARCH ARTICLE

Interactive Design With Gesture and Voice Recognition in Virtual Teaching Environments

KE FANG[1] AND JING WANG[2]
[1]Network and Information Center, Chengdu Normal University, Chengdu 610000, China
[2]Office for the Advancement of Educational Informatization, Chengdu Normal University, Chengdu 610000, China

Corresponding author: Ke Fang (fk@cdnu.edu.cn)

This work was supported in part by the Sichuan Provincial Research Center for the Application and Development of Educational Informatization under Grant JYXX23-006.

This work involved human subjects or animals in its research. The authors confirm that all human/animal subject research procedures and protocols are exempt from review board approval.

ABSTRACT In virtual teaching scenarios, head-mounted display (HMD) interactions often employ traditional controller and UI interactions, which are not very conducive to teaching scenarios that require hand training. Existing improvements in this area have primarily focused on replacing controllers with gesture recognition. However, the exclusive use of gesture recognition may have limitations in certain scenarios, such as complex operations or multitasking environments. This study designed and tested an interaction method that combines simple gestures with voice assistance, aiming to offer a more intuitive user experience and enrich related research. A speech classification model was developed that can be activated via a fist-clenching gesture and is capable of recognising specific Chinese voice commands to initiate various UI interfaces, further controlled by pointing gestures. Virtual scenarios were constructed using Unity, with hand tracking achieved through the HTC OpenXR SDK. Within Unity, hand rendering and gesture recognition were facilitated, and interaction with the UI was made possible using the Unity XR Interaction Toolkit. The interaction method was detailed and exemplified using a teacher training simulation system, including sample code provision. Following this, an empirical test involving 20 participants was conducted, comparing the gesture-plus-voice operation to the traditional controller operation, both quantitatively and qualitatively. The data suggests that while there is no significant difference in task completion time between the two methods, the combined gesture and voice method received positive feedback in terms of user experience, indicating a promising direction for such interactive methods. Future work could involve adding more gestures and expanding the model training dataset to realize additional interactive functions, meeting diverse virtual teaching needs.

INDEX TERMS Game engines, hand tracking, human–computer interaction, recurrent neural networks, speech processing, virtual environments.

I. INTRODUCTION

The application of Virtual Reality (VR) technology in educational simulations and virtual laboratories has been increasingly recognized for its potential to enhance learning experiences [1], [2]. However, the prevalent reliance on traditional input devices like keyboards, mice, or even touchscreens in VR environments limits the naturalness and intuitiveness of user interactions [3], [4]. Existing improvements in this area have primarily focused on replacing controllers with gesture recognition [5], [6], [7], [8], [9]. However, the exclusive use of gesture recognition may have

The associate editor coordinating the review of this manuscript and approving it for publication was Andrea Bottino.

limitations in certain scenarios, such as complex operations or multitasking environments [10]. Recognizing the need for more immersive and user-friendly interfaces, this research focuses on the development of a VR interaction method based on gesture and voice recognition. Gesture recognition allows for a more natural and intuitive form of interaction, closely mimicking real-world actions, thus reducing the cognitive load and improving the learning curve for users in VR educational settings [11], [12]. Similarly, voice recognition offers a hands-free, efficient way to navigate and control virtual environments, making it particularly useful for educational purposes where multitasking is often required [13]. By integrating these interaction modalities, this research aims to create a more engaging and effective educational experience in VR, aligning with the evolving needs of modern learners.

TABLE 1. Differences Between Two Gesture Recognition Methods.

Comparison Criteria	RGB-Based Gesture Recognition	Depth-Based Gesture Recognition
Image Type	Colour Image (RGB channels)	Depth Image (distance information)
Data Processing	Computer vision, image processing	Point cloud processing, skeletal tracking
Accuracy	Moderate	High
Robustness	Highly influenced by lighting conditions, needs to handle background interference and occlusions	Highly robust against lighting and background occlusions
Cost	Lower (standard colour cameras)	Higher (dedicated depth cameras)
Computational Requirement	Relatively high	Relatively low

II. RESEARCH BACKGROUND AND OBJECTIVES

A. DEVELOPMENT AND APPLICATION NEEDS OF VR IN EDUCATION

In recent years, the evolution of VR technology has entered a phase of rapid development. The exploration of VR in the field of education is becoming increasingly widespread. Notably, research on VR-based education in higher education has achieved certain advancements. Peixoto et al. [14] explores the application of immersive virtual reality technologies in teaching English as a Foreign Language (EFL) from a cognitive science perspective. The study compares traditional listening teaching methods with VR-based methods, finding that VR technology significantly enhances students' language learning outcomes. Toti and Kunicina [15] explores how virtual reality technologies impact the modernization of curricula at Western Balkan universities in the field of renewable sources. The researchers found that VR technology enables students to understand and apply complex concepts related to renewable energy more effectively. Johnson-Glenberg et al. [16] delved into the effects of embodied learning and digital platforms on the retention of physics knowledge from a cognitive science perspective. By using a system based on VR technology to conduct experiments on students, they discovered that embodied learning could significantly boost knowledge retention. Radianti et al. [17] systematically reviewed immersive VR applications in higher education, summarising design elements, lessons learnt, and proposing future research agendas. Campos et al. [18] studied the impact of VR when teaching vector studies to first-year engineering students, concluding that students can interact directly with vectors in a VR environment, aiding a deeper understanding of vector concepts. Legi et al. [19] discussed the application of VR models in education as an adaptation for the era of 5.0. Using a descriptive qualitative method, data was collected from observations, interviews, and document reviews of 40 participants from a school. The findings demonstrated that VR models can enhance students' comprehension abilities, with the students' KKM scores rising from 25% to 90%.

Within China, Wang and Wang [20] designed a simulation teaching system tailored for PLC experimental instruction, Wang and Jiang [21] developed a VR live streaming multi-user online teaching system based on panoramic videos, Zhu [22] designed applications for big data and cloud computing's virtual reality experiment platform, and Guo et al. [23] embarked on a study of a lathe teaching training system using VR. Most of the above VR teaching systems are implemented on devices such as desktop or tablet computers, and this dependency limits the naturalness and intuitiveness of user interaction [3], [4]. Systems based on VR headsets, on the other hand, can deliver a more authentic virtual reality experience, hence they are gaining increasing attention from researchers.

Currently, head-mounted virtual reality educational systems predominantly utilise controllers or other devices for interaction. However, in virtual teaching scenarios such as skill training, experimental operations, gesture learning, sports and fitness training, and artistic creation, the use of controllers can reduce the effectiveness of the training [24], [25], [26], [27], [28]. Therefore, a design and implementation based on artificial intelligence for VR interaction has been chosen [5], aiming to achieve interaction in virtual scenes solely through the head-mounted display, enhancing the user experience and immersion.

B. APPLICATION OF ARTIFICIAL INTELLIGENCE TECHNOLOGY

In recent years, the application of artificial intelligence (AI) in VR scenarios has started to gain attention. For instance, Tseng [29] explored an intelligent augmented reality system based on voice recognition that can entirely depend on image and voice recognition for interaction. Khundam et al. [30]

and colleagues developed a VR intubation training system, discovering that hand tracking allows learners to intuitively recognise and correct their posture, promoting self-study and practice. However, the exclusive use of gesture recognition may have limitations in certain scenarios [10]. Consequently, this study proposes an integrated approach of gesture and voice interaction to enhance the flexibility and efficiency of user engagement [31].

In VR educational scenarios, the implementation of gesture recognition technology often uses RGB-based detection methods [32], [33] or depth-based detection methods [34], [35]. The differences between the two methods are shown in Table 1.

These two methods have been applied to many VR peripherals for hand tracking and gesture recognition, such as Leap Motion (Ultraleap), Microsoft Kinect (Microsoft) [36] and others. In the most recent studies, headsets using Inside-out technology [37], [38] seem to be trending, like Oculus Quest 1-2 (Meta) and HTC Vive Pro 2, HTC Vive XR (HTC). They utilise built-in optical cameras or depth cameras of the headsets, combined with the product's SDK under the OpenXR standard to achieve hand tracking without any other tracking peripherals.

Furthermore, integrating voice recognition APIs into virtual environments has become more feasible [39]. This study explored the potential of incorporating voice recognition APIs into virtual settings. While this method requires the transformation of speech into text on the servers of the API service provider, a process that involves inherent costs, and potentially demands enhanced network reliability, it nonetheless offers an interesting means to achieve local scene control through the application of transcribed texts. To assist interested readers, Supplementary Material 1 includes example code for using the iFlytek Voice Recognition API in a Unity environment.

Considering the cost, the ease of use of hand-tracking SDKs, and the demand for network-independent local deployment, this study has designed a system using the HTC Vive Pro2 headset for hand tracking. Hand rendering and strategy-based gesture recognition are implemented in Unity, along with the development of a locally-operated Chinese keyword recognition model. This model employs gesture activation for navigation and control interactions. The paper will specifically illustrate the use case scenario of the Teacher Training Simulation System, a multi-scenario platform aimed at enhancing the teaching skills and postures of student teachers in diverse environments.

C. RESEARCH OBJECTIVES AND CONTRIBUTIONS

The primary objective of this paper is to explore the implementation and design of gesture and voice-based interaction in virtual teaching scenarios. By applying gesture recognition and voice recognition technologies, we aim to develop a more humanized and efficient method for virtual teaching interaction, thereby enhancing teaching effectiveness and learning experiences and promoting the application prospects of virtual reality technology in the field of education.

The primary objectives and contributions encompass the following aspects:

- Innovative Interaction Design in VR: Research on the combined use of gestures and voice for interaction in VR remains scarce. This study explores the application of gesture and voice-based interaction design in virtual teaching scenarios, proposing a more humanized and efficient interaction method, thereby enriching the current body of research in this area.
- Gesture Recognition Implementation in Unity: This paper presents hand rendering in virtual scenes based on hand keypoint coordinates in Unity, alongside strategies for gesture recognition. We provide example codes for gesture recognition, which can be referenced by other HMD products using the OpenXR framework.
- Voice Command Classification Model in Unity: The study designed and implemented a Chinese voice command word classification model in Unity. This model assists gesture navigation and scene control, reducing the memory load of pure gesture interaction. The model achieved 95% accuracy in a customized validation set, demonstrating its application potential. The paper also provides the model's code and the processing flow code, facilitating adjustments by interested readers.
- Promotion of VR Technology in Education: The study highlights the value and role of virtual reality technology in education, advocating for its broader application.
- Enhancing Understanding of VR Technology: By increasing awareness and understanding of virtual reality technology, this research contributes to the promotion of its application and development in other fields.

III. SYSTEM DESIGN AND IMPLEMENTATION

The interaction functionality in virtual teaching scenarios is principally divided into three modules: gesture recognition, voice recognition, and UI interaction. The system employs the HTC OpenXR SDK for hand tracking. Taking a simulated teaching system for teacher trainees as the envisaged application scenario, a Chinese voice command classification model is developed to distinguish between commands such as 'teleport interface', 'teaching interface', and 'settings interface'. The Unity XR Interaction Toolkit, among other tools, is utilised to facilitate UI interaction. The architectural framework of this virtual teaching interaction feature is depicted in Figure 1.

The choice to utilize the fist gesture for voice activation, as opposed to other gestures, is based on its simplicity and ease of recognition [40]. The fist gesture is visually distinctive and less likely to be confused with other gestures, thereby reducing the likelihood of inadvertent activations. Additionally, employing voice commands like ''teleport interface'' instead of directly saying ''teleport to a specific location'' serves two purposes. Firstly, it reduces the complexity of

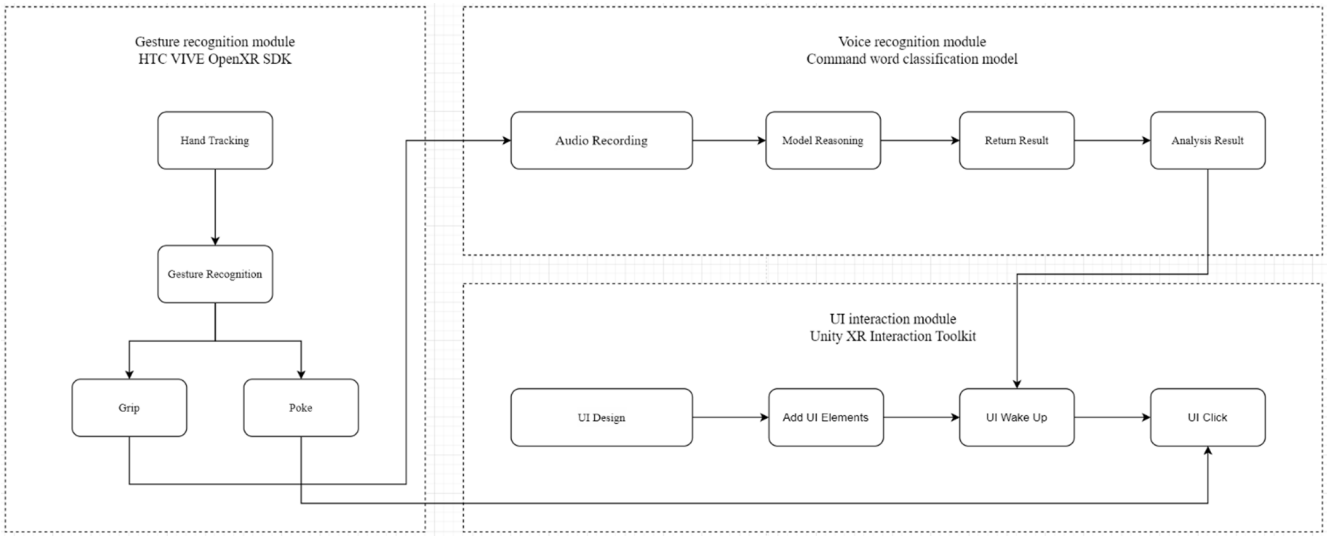


FIGURE 1. Interaction function architecture in virtual teaching environment.

voice recognition and enhances accuracy [41]. Short and consistent command phrases minimize errors during the voice recognition process and improve the system's response time. Secondly, this approach intuitively presents teleportation options on the UI interface, facilitating easier selection and interaction.

A. GESTURE RECOGNITION MODULE DESIGN AND IMPLEMENTATION

The design and implementation of the gesture recognition module involve capturing and tracking hand movements using the HTC VIVE OpenXR SDK in the Unity environment. The logic of implementation is based on hand tracking achieved through the built-in optical camera of the HTC Vive Pro2 HMD, rendering the hand model in Unity, and enabling fundamental gesture recognition functionalities.

Initially, the system acquires the coordinates of the hand joints. These data are updated during each frame rendering process in Unity, ensuring the dynamic accuracy of the hand model.

Subsequently, a hand model comprising 26 joint nodes is imported into Unity and associated with the HMD object. The Transform components of each joint in the model are dynamically updated to reflect changes in actual hand movements. The system is capable of determining and rendering the visibility status of the hand model during the update process.

Moreover, this study designs a strategy-based gesture recognition algorithm. The algorithm utilises static methods to determine the extended or curled status of the fingers. For instance, Let v_1 represent the vector from the fingertip joint to the wrist joint, with its norm squared denoted as $\|v_1\|^2$. Let v_2 represent the vector from the last joint of the finger to the wrist joint, with its norm squared as $\|v_2\|^2$. If $\|v_1\|^2 > \|v_2\|^2$, the finger is considered to be in an extended state. Conversely, if $\|v_1\|^2 < \|v_2\|^2$, the finger is considered to be in a flexed state, as depicted in Figure 2. A grip gesture is identified when the four fingers, excluding the thumb, are curled, and a poke gesture is determined when the index finger is extended while the other three fingers are curled.

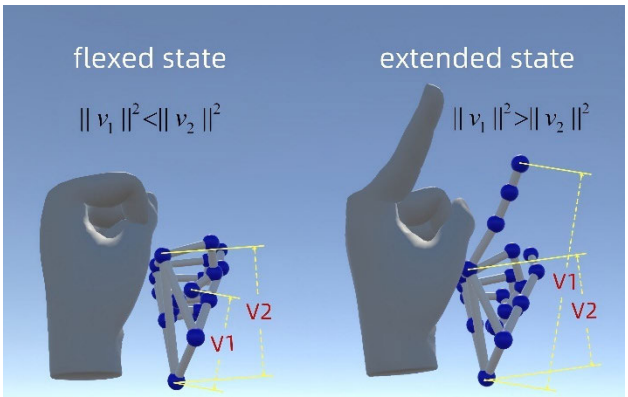


FIGURE 2. Finger state judgment example.

In practical applications, these gesture recognition functionalities are integrated into Unity's RenderModel script. The system monitors gestures in real-time by invoking the recognition methods within the Update() function and can apply the recognition results to various interactive scenarios, such as activating voice recognition or interacting with UI elements. For interested readers, Supplementary Material 2 provides the process of hand tracking in Unity and some example codes.

B. DESIGN AND IMPLEMENTATION OF THE VOICE RECOGNITION MODULE

The Chinese voice keyword classification model is constructed and deployed for voice classification:

1) DATA COLLECTION AND PREPROCESSING

a: DATA COLLECTION

127 valid audio recordings were obtained, capturing different intonations from 43 males and 41 females. Each audio sample includes Chinese voice commands: ''Transmission Interface'', ''Teaching Interface'', and ''Settings Interface''. In addition, 20 erroneous audio samples and 10 background audio recordings lasting 10 seconds each were also collected.

b: AUDIO SEGMENTATION

Key voice segments were extracted and silent parts removed by setting a minimum silence length of 1500 ms and a silence threshold of −50 dB. This process successfully segmented and categorized 381 effective audio clips.

c: AUDIO AUGMENTATION

To expand the dataset and enhance the model's generalization capability, audio enhancement techniques were applied to the correct audio segments. This included time stretching/compression (sampled from a uniform distribution) and pitch shifting (randomly selected within a range of [−5,5] steps), yielding 1143 enhanced audio samples.

d: AUDIO SYNTHESIS

Inspired by the work of Supriya et al. in keyword recognition [42], an audio synthesis strategy was employed. Through 4000 iterations, combining correct, incorrect, and background audio, a diverse and enriched set of 4000 synthetic audio samples was created

e: LABEL ANNOTATION

During the audio synthesis process, a time-step annotation strategy of 50 ms was used. Initially, one-hot encoding (C1, C2, C3) with categorical cross-entropy loss function was applied, but due to a high prevalence of incorrect labels, the training accuracy was only 80%. Subsequently, an improved annotation scheme inspired by the YOLO algorithm [43] was adopted, incorporating existence and category labels (Existence, C1, C2, C3), resulting in an output dimension of (1375,4).

f: DATASET DIVISION AND FEATURE EXTRACTION

Data partitioning was performed using K-Fold cross-validation (K=5) and a 75-25 training-validation split. Mel-frequency spectral analysis was used for feature extraction, setting the FFT window size to 200, the window interval to 80 samples, utilizing 101 Mel filters, and capping the maximum frequency at 8000 Hz. The transformed data dimension was (5513,101).

2) DEFINITION OF LOSS FUNCTION, ACCURACY, AND F1 SCORE

To effectively train the model, this research customised a loss function, accuracy function, and F1 score function that encompass both existence and classification considerations.

a: LOSS FUNCTION

The loss related to existence (1) is computed using the binary cross-entropy between the actual existence labels and the predicted values:

$$L_{presence} = -\frac{1}{N}\sum_{i=1}^{n}\left[y^{(i)}_{true,presence}\log(y^{(i)}_{pred,presence}) + (1-y^{(i)}_{true,presence})\log(1-y^{(i)}_{pred,presence})\right] \tag{1}$$

The classification loss (2) is computed using the categorical cross-entropy between the actual classification labels and predicted values. However, classification loss is only computed when the predicted existence value exceeds a threshold (0.7):

$$mask^{(i)} = \begin{cases} 1, & \text{if } y^{(i)}_{pred,presence} > 0.7 \\ 0, & \text{otherwise} \end{cases}$$
$$L_{classes} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{c=1}^{C} mask^{(i)} y^{(i,c)}_{true,classes}\log(y^{(i,c)}_{pred,classes}) \tag{2}$$

The final loss function (3) is the average of the existence and classification losses:

$$L = \frac{L_{presence} + L_{classes}}{2} \tag{3}$$

b: ACCURACY FUNCTION

Given that the model's output for existence is a probability between 0 and 1, these output values are first rounded to obtain binary existence predictions. Subsequently, these predictions are compared with the actual existence labels to compute the existence accuracy (4):

$$presence\_accuracy = \frac{1}{N}\sum_{i=1}^{N} 1\left(y^{(i)}_{true,presence} = round\left(y^{(i)}_{pred,presence}\right)\right) \tag{4}$$

Classification accuracy (5) is computed only when the existence label is 1 (indicating true existence):

$$classes\_accuracy = \frac{\sum_{i=1}^{N} 1\left(y^{(i)}_{true,classes} = \arg\max_c\left(y^{(i,c)}_{pred,calsses}\right)\right)\times y^{(i)}_{true,presence}}{\sum_{i=1}^{N} y^{(i)}_{true,presence} + \varepsilon} \tag{5}$$

The overall accuracy (6) is the average of the existence accuracy and classification accuracy:

$$total\_accuracy = \frac{presence\_accuracy + classes\_accuracy}{2} \tag{6}$$

c: F1 SCORE FUNCTION

For the existence F1 score calculation:

First, a threshold is used to determine the existence predictions:

$$y^{(i)}_{pred,presence} = \begin{cases} 1, & \text{if } y^{(i)}_{pred,presence} > 0.7 \\ 0, & \text{otherwise} \end{cases} \tag{7}$$

Formulas for True Positives of Presence Prediction, Actual Positives of Presence Prediction, and Predicted Positives of Presence Prediction (8) are:

$$\begin{aligned} TP_{pres} &= \sum_{i=1}^{N} y^{(i)}_{true,presence} \times y^{(i)}_{pred,presence} \\ PP_{pres} &= \sum_{i=1}^{N} y^{(i)}_{true,presence} \\ \text{Pred}P_{pres} &= \sum_{i=1}^{N} y^{(i)}_{pred,presence} \end{aligned} \tag{8}$$

The respective formulas for precision (9) and recall (10) are:

$$\text{Precision}_{pres} = \frac{TP_{pres}}{\text{Pred}P_{pres} + \varepsilon} \tag{9}$$

$$\text{Recall}_{pres} = \frac{TP_{pres}}{PP_{pres} + \varepsilon} \tag{10}$$

The F1 score for existence (11) is:

$$F1_{pres} = 2 \times \frac{\text{Precision}_{pres} \times \text{Recall}_{pres}}{\text{Precision}_{pres} + \text{Recall}_{pres} + \varepsilon} \tag{11}$$

For the classification F1 score calculation:

Predictions for classification (12):

$$y^{(i)}_{pred,classes} = \arg\max_{c} y^{(i,c)}_{pred,classes} \tag{12}$$

The formulas for True Positives of Classification Prediction, Actual Positives of Classification Prediction, and Predicted Positives of Classification Prediction (13) are as follows:

$$\begin{aligned} TP_{class} &= \sum_{i=1}^{N} y^{(i)}_{pred,presence} \times 1\left(y^{(i)}_{true,classes} = y^{(i)}_{pred,classes}\right) \\ PP_{class} &= \sum_{i=1}^{N} y^{(i)}_{true,peesence} \\ \text{Pred}P_{class} &= \sum_{i=1}^{N} y^{(i)}_{pred,presence} \end{aligned} \tag{13}$$

The classification accuracy, recall, and F1 score are calculated based on Equations (9), (10), and (11). The final F1 score is the average of the existence F1 score and the classification F1 score.

3) MODEL CONSTRUCTION, TRAINING, AND RESULTS ANALYSIS

a: MODEL ARCHITECTURE

The model initiates with an input layer, followed by a one-dimensional convolutional layer for preliminary feature extraction. To enhance robustness, batch normalization and ReLU activation functions are introduced, complemented by a Dropout layer for regularization to mitigate overfitting risks. The architecture also includes two GRU layers, each followed by Dropout and batch normalization layers, effectively capturing complex features in time-series data. Additionally, an attention mechanism layer is incorporated to focus on key parts of the input data.

b: OUTPUT LAYER DESIGN

At the tail of the model, two output layers are designed: one using a sigmoid activation function to predict the presence, and another using softmax activation for class discrimination. The outputs of these layers are combined to form the final output of the model, optimizing classification effectiveness and prediction accuracy.

c: TRAINING STRATEGY

In this study, the dataset is split into training and validation sets in a 75-25 ratio. The Adam optimizer [44] is employed, with a learning rate set at 0.001. The training is set to a maximum of 100 epochs, processing 64 samples per batch. To prevent overfitting and reduce unnecessary computational costs, training is terminated early if there is no decrease in validation loss over 10 consecutive epochs.

d: PERFORMANCE EVALUATION

In the final epoch, the model demonstrated the following performance metrics:

- Training Loss: 0.02026
- Validation Loss: 0.02463
- Training Accuracy: 94.20%
- Validation Accuracy: 94.47%
- Training F1 Score: 99.95%
- Validation F1 Score: 99.19%

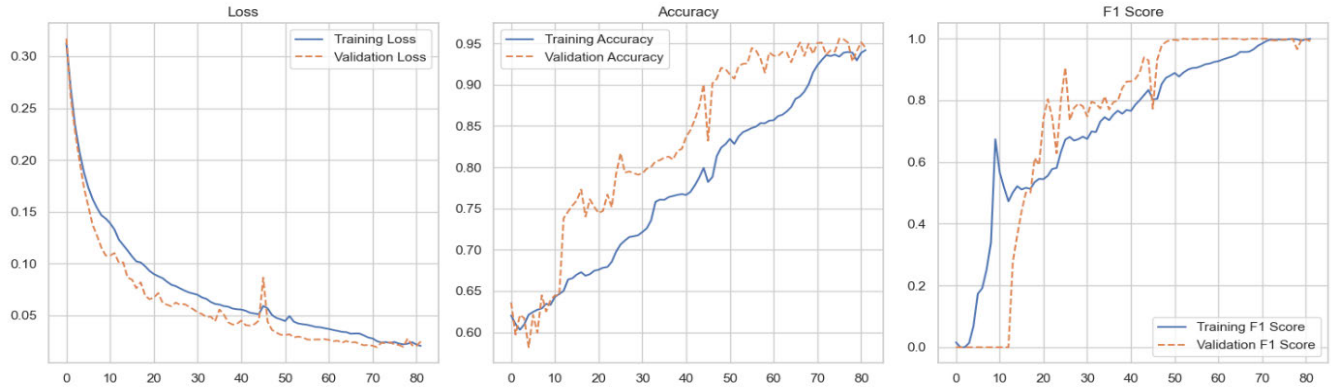
Throughout the training cycle, the model achieved its best validation loss of 0.01902 at the 72nd epoch, with a corresponding validation accuracy of 95.10%.

e: RESULT PRESENTATION

To visually present the evolution of the model's performance, graphs depicting the changes in loss, accuracy, and F1 score over the training process were plotted (as shown in Figure 3). These graphs clearly illustrate a steady improvement in model performance as training progresses, reaching an optimal state at a certain point, demonstrating its effectiveness and potential in audio signal processing and classification tasks.

4) MODEL DEPLOYMENT AND RESPONSE

Once the model was successfully trained, inference was conducted using a method to invoke Python scripts within Unity.




```
Training loss from the last epoch: 0.020260900259017944
Validation loss from the last epoch: 0.02463165856897831
Training accuracy from the last epoch: 0.9419872760772705
Validation accuracy from the last epoch: 0.9446504712104797
Training F1 score from the last epoch: 0.9995111227035522
Validation F1 score from the last epoch: 0.9919109344482422
Best validation loss: 0.019021114334464073 (at epoch 72)
Corresponding validation accuracy: 0.9509943127632141
Corresponding validation F1 score: 1.0
```

FIGURE 3. Model training results.

A threshold was set to determine the final recognition result based on the criteria: ''a presence greater than 0.99 and being classified as a particular category for 50 continuous steps''. Based on the analysis, the recognition results returned were either ''Transmission Interface'', ''Teaching Interface'', ''Settings Interface'', or ''Others''. An illustrative example of the implementation code for the voice recognition model can be found in Supplementary Material 3.

Two methods were designed:

- StartSpeechRecognition Method: Initiates the recording of the user's voice.
- GetSpeechRecognitionResult Method: Halts the recording, invokes the Python script for inference, and returns the recognition outcome.

C. DESIGN AND IMPLEMENTATION OF THE UI INTERACTION MODULE

The implementation of the UI interaction module primarily includes the design of the UI module and the realisation of gesture-to-UI interaction functionalities. To achieve efficient and stable interaction effects, the Unity XR Interaction Toolkit plugin was employed, customising its features through bespoke scripts and modifications to the plugin properties.

1) DESIGN OF THE UI MODULE

A Canvas instance was designed as a child object of the camera. It comprises several UI interfaces, such as the Voice Interface, Transmission Interface, and Teaching Interface, as well as various UI elements like buttons, sliders, and text.

2) IMPLEMENTATION OF GESTURE-TO-UI INTERACTION

a: REALISATION OF UI ACTIVATION FEATURE

The corresponding UI interface is activated by monitoring the fist-clenching gesture and recognition outcomes. When the fist-clenching gesture commences, the Voice Interface is activated within the callback function OnGripGestureStart and displays ''Voice Recognition Initiated''. Concurrently, the StartSpeechRecognition method is invoked to begin recording. Upon the conclusion of the fist-clenching gesture, the Voice UI exhibits ''Voice Recognition Halted'' and showcases the recognition result returned by the GetSpeechRecognitionResult method.

Within the callback function OnGripGestureEnd, based on the recognition result, the pertinent UI interface is activated. For instance, if the recognised command is ''Transmission Interface'', the Transmission UI is consequently activated. A visual representation of the Transmission Interface activation process (with the left side illustrating the initiation of voice recognition through fist-clenching and the right side depicting the conclusion of the gesture, returning the recognition outcome and opening the relevant UI) is depicted in Figure 4.

FIGURE 4. Teleport interface activation screen.

b: IMPLEMENTATION OF UI CLICK FUNCTIONALITY

Within the first joint of the index finger on the hand model, a child object, ''Poke'', equipped with the XRPokeInteractor component from the Unity XR Interaction Toolkit plugin is introduced. The XR Interaction Manager component,

in conjunction with the poke gesture and Unity EventSystem and XRUIInputModule components, facilitates the UI click functionality.

Upon recognition of the poke gesture, the Poke object is activated within the OnPokeGestureStart callback function. The XRPokeInteractor component on the Poke is capable of detecting collisions between the fingertip and UI elements, subsequently triggering the pertinent interactive events. Acting as the interaction overseer, the XR Interaction Manager component receives the event and relays it to the designated UI element for interaction initiation. Unity EventSystem is responsible for managing the flow of events within the scene and collaborates with the XRUIInputModule component to process input events related to UI elements.

For instance, within the teleportation interface UI, when the index finger touches a button, the button's OnValueChange event activates the Transport callback function, resulting in a change of position. The manifestation of the UI click culminating in the teleportation screen (left: teleport button click, right: teleport to designated position) is illustrated in Figure 5.

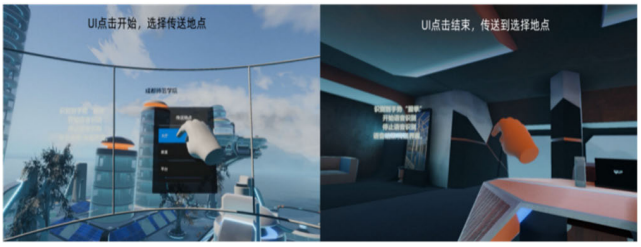


FIGURE 5. Teleport screen triggered by UI click.

IV. EXPERIMENTAL DESIGN AND RESULT ANALYSIS

A. EXPERIMENTAL DESIGN

1) PURPOSE OF THE EXPERIMENT

The purpose of this experiment was to compare the performance of gesture plus voice recognition with traditional controller operations in terms of operational efficiency and user experience, thereby validating the potential advantages of gesture plus voice recognition technology.

2) EXPERIMENTAL SUBJECTS

The experiment recruited 20 university undergraduates from diverse grade levels and academic backgrounds. Their ages ranged from 18 to 22 years, including 14 males and 6 females. All these students were members of a work-study program at our school.

Before the official experiment, all participants underwent comprehensive training relevant to the experiment, covering the necessary hardware knowledge and operational skills. The training included:

- Hardware Knowledge for the Experiment: Introduction to the VR equipment used in the experiment, including headsets, controllers, etc.
- Operational Skills in the Experiment: Instructions on how to perform the required operations in the virtual environment, such as gesture, voice, controller interactions, and environmental adaptation.

To ensure the effectiveness of the experiment, a rigorous assessment of the participants' operational skills was conducted. This included their proficiency in basic operations and their ability to handle errors, ensuring that they reached the required level of proficiency. "Reaching proficiency" was defined as being able to competently complete all experimental operations.

3) EXPERIMENTAL PROCEDURE

The experimental environment was constructed using Unity to design a virtual scene with gesture plus voice interaction capabilities, and traditional VR controllers were prepared for comparison.

Three experimental tasks were set:

- Task 1: Using gesture recognition to activate voice recognition and saying "Teleport Interface", then clicking the "Lobby" button to teleport versus using the controller side button to wake up the teleport interface and clicking the "Lobby" button to teleport.
- Task 2: Adjusting the brightness of the lobby lights using both control methods via a slider button.
- Task 3: Entering a 9-digit number (147258369) using a simulated keyboard with both control methods.

After each participant had mastered the task operations, they signaled the start of the task, and a stopwatch was used for timing. Upon completion of the three tasks, participants filled out a Likert scale [45] based on speed, fluency, naturalness, and user preference, as well as a user feedback survey on the mode of operation, to collect more comprehensive user feedback. Photos of the experimental process are shown in Figure 6.

FIGURE 6. Experiment process photos.

B. RESULTS AND ANALYSIS

1) QUANTITATIVE DATA ANALYSIS

Descriptive statistics, t-tests, calculation of Cohen's d effect size, and Analysis of Variance (ANOVA) were employed for the data analysis in this study.

a) Descriptive statistics indicate that for Task 1, the average completion time using gesture + voice was 8.39s, compared to 6.99s using the controller. For Task 2, gesture + voice averaged at 4.58s, whilst the controller took 3.43s. For

Task 3, gesture + voice averaged at 14.11s, in contrast to the controller's 16.84s.

b) Results from the t-test are as follows: Task 1: t=9.424248655359744, p=1.3560356306333986e-08, Task 2: t=18.85631973002477, p=9.263050654483422e-14, Task 3: t=-9.967566734727422, p=5.545634252659465e-09.

Through the t-test on completion times, gesture + voice for Tasks 1 and 2 took significantly longer than using the controller, showing statistical significance. However, for Task 3, the gesture + voice was notably faster. These p-values are far below 0.05, implying the null hypothesis can be rejected, indicating a significant difference between the two methods. This suggests that these time differences are not just due to random sample variation.

c) Cohen's d effect size results are: Task 1: Cohen's d=3.2323319670346726, Task 2: Cohen's d=5.610460766042297, Task 3: Cohen's d=-3.3879429989496868.

Cohen's d is a standard measure for effect size. With values exceeding 3 for Tasks 1 and 2, it indicates a strong effect in time efficiency with gesture + voice taking more time compared to the controller. Conversely, with Task 3's Cohen's d at −3.39, it suggests a significant time advantage for gesture + voice over the controller.

d) Results from the Analysis of Variance (ANOVA) are:

The data reveals that the type of task and the interaction between the task and method have a significant influence on the results, with F values of 7343.06 and 152.32 respectively, both with p-values well below 0.05. However, the method alone does not show a significant effect on the results, with an F value of 0.39 and a p-value of 0.54. This means that the null hypothesis cannot be rejected, suggesting there is no significant efficiency difference between gesture + voice recognition and the traditional controller across all tasks.

2) QUALITATIVE DATA ANALYSIS

The qualitative feedback from participants provided valuable insights into preferences for interaction methods. These responses primarily focused on the intuitiveness, naturalness, efficiency, and satisfaction aspects of gesture+voice operation versus traditional joystick control.

a: LIKERT SCALE ANALYSIS

An analysis of the Likert scale data was conducted across four dimensions: speed, fluency, naturalness and overall preference. Average scores for gesture+voice and joystick control were calculated in each dimension, observing the score differences between the two interaction methods. This provided an analysis of the overall user experience with both interaction types. The results are as follows:

- Speed: Joystick control scored higher on average than gesture+voice in speed, indicating its efficiency advantage.
- Fluency: Joystick control also received higher scores in fluency, demonstrating a more coherent and seamless user experience.

TABLE 2. Results of Variance Analysis.

	F Value	Num DF	Den DF	Pr > F
Task	7343.0626	2.0000	38.0000	0.0000
Method	0.3879	1.0000	19.0000	0.5408
Task:Method	152.3167	2.0000	38.0000	0.0000

- Naturalness: In contrast, gesture+voice scored significantly higher in naturalness, suggesting that users found this interaction more intuitive and natural.
- Preference: Gesture+voice interaction received higher scores in overall preference, indicating a general favorability towards this method of interaction.

Detailed results of the Likert scale analysis are presented in Table 3.

Despite joystick control's superior performance in speed and fluency, gesture+voice interaction demonstrated significant advantages in key aspects of user experience, such as naturalness and overall preference. This suggests that although gesture+voice interaction may slightly lag in certain performance metrics compared to traditional joystick control, its intuitive and natural interactive experience grants it an advantage in user preference.

b: PARTICIPANT FEEDBACK ANALYSIS

This feedback was visualised by creating a word cloud. A word cloud is a visual representation technique used to depict frequency information from text data. In a word cloud, words that appear more frequently are displayed in a larger font, offering an intuitive view of the most common words or phrases within the textual data.

The word cloud derived from the participants' feedback can be seen in Figure 7.

The word cloud presented reflects a collection of terms associated with feedback on ''gesture'', ''voice'', and ''gamepad'' interactions. The size of words within a word cloud typically signifies their importance or frequency of occurrence. In this word cloud, the most prominent terms are ''gesture'', ''voice'', and ''operation'', indicating the significance of these concepts in the related discourse or dataset.

Adjacent to ''gesture'', terms such as ''intuitive'', ''high'', ''tech'', and ''fun'' suggest that gesture-based controls are perceived as intuitive, advanced in technology, and enjoyable. Surrounding ''voice'', words like ''easy'', ''natural'', ''skills'', and ''allows'' imply that voice controls are user-friendly, natural to use, and enable users to demonstrate or apply specific skills.

Conversely, the section pertaining to ''gamepad'' includes terms like ''doubts'', ''unnatural'', ''stressful'', and ''memorizing'', which may infer that, in comparison, the use of gamepads could be associated with a negative experience. This word cloud could potentially originate from a study or survey aimed at comparing user experiences with gesture and voice controls versus traditional gamepad usage. Through this analysis, insights into user perceptions regarding these different modes of interaction are gleaned.

TABLE 3. Detailed results of the likert scale analysis.

Participant	Speed (Gesture+Voice)	Speed (Joystick)	Fluency (Gesture+Voice)	Fluency (Joystick)	Naturalness (Gesture+Voice)	Naturalness (Joystick)	Preference (Gesture+Voice)	Preference (Joystick)
1	4	4	3	4	5	3	4	3
2	3	4	3	4	4	3	4	2
3	2	4	3	4	5	3	5	2
4	3	4	3	4	4	3	4	2
5	4	4	2	4	5	3	5	2
6	3	4	4	4	4	3	4	3
7	4	4	3	4	5	3	4	2
8	3	3	4	3	4	3	5	3
9	3	4	3	4	4	3	5	2
10	3	4	3	4	4	3	4	2
11	3	4	4	3	5	3	5	2
12	3	4	3	4	5	3	4	2
13	4	4	2	4	4	3	4	2
14	3	4	3	4	4	3	5	2
15	2	4	3	4	5	3	4	3
16	3	4	3	4	4	3	5	2
17	3	4	4	4	5	3	5	2
18	3	4	3	4	4	3	5	2
19	4	4	2	4	5	3	4	2
20	3	4	3	4	5	3	5	3
Average	3.15	3.95	3.05	3.9	4.5	3	4.5	2.25

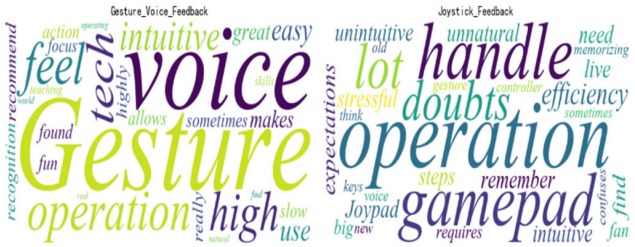


FIGURE 7. Word cloud generated from participant feedback.

C. EXPERIMENTAL DISCUSSION

This experiment, by comparing the user experience and efficiency of gesture plus voice recognition versus traditional controller operations, aimed to explore the effectiveness of this interactive method. Although the experimental tasks were primarily focused on UI interactions, rather than directly simulating specific actions in teaching, these tasks still provided valuable insights.

1) QUANTITATIVE EXPERIMENTAL RESULTS

Descriptive statistics revealed the average completion time for different tasks using both interaction methods, laying the groundwork for subsequent statistical analysis. The t-test results indicated significant differences in completion times between gesture+voice and joystick controls in certain tasks. The effect size calculated using Cohen's d further affirmed the practical significance of these differences. Analysis of Variance (ANOVA) showed that the type of task and the interaction between task and method of operation had a significant impact on the results, while the method of operation itself did not significantly affect the task completion time.

2) QUALITATIVE EXPERIMENTAL RESULTS

Most participants found the gesture+voice operation to be more intuitive and user-friendly. They reported that this mode of operation enabled a more natural interaction with the virtual reality environment, in contrast to the joystick control, which lacked intuitiveness and naturalness. Feedback highlighted the sometimes slow speed of gesture recognition, identifying an area for future improvement.

3) COMPARISON WITH PREVIOUS STUDIES

This study exhibits a certain degree of innovation in the field of VR interaction, particularly in exploring the combined use of gestures and voice for operational purposes. Current literature primarily focuses on gesture-based operations [5], [6], [7], [8], [9]. These studies provide a crucial background for understanding the application of gestures in VR, yet the exploration of combining gestures with voice interaction remains relatively limited.

This research evaluated the efficiency and user experience of the gesture-plus-voice operation method through specific experimental tasks. This approach not only offers a new perspective on how gestures and voice can work in tandem but also lays a foundation for future research to explore this combined mode of interaction. The experiments indicate that in some aspects, the gesture-plus-voice operation method is similar in efficiency to traditional controller operations, but it may offer greater intuitiveness and a more natural feel in terms of user experience.

V. CONCLUSION AND FUTURE RESEARCH DIRECTIONS

This study proposed an interaction method integrating gestures and voice, exemplified through a teacher training simulation system, particularly suitable for virtual teaching scenarios requiring hand skill training. The study

incorporated the HTC OpenXR SDK, Unity engine, and a custom voice classification model, effectively achieving accurate recognition of gestures and voice commands, thereby enhancing the interactive experience in virtual teaching. Experiments validated the effectiveness of the system in terms of control and operation. Although the research scope was limited, it demonstrated the potential of adding more gesture types and expanding the training dataset for the voice model, offering valuable preliminary insights into the application of interaction methods in educational environments.

Future work will focus on the following aspects:

- Long-Term Use and User Adaptability Studies: Investigate the adaptability and efficiency of users in long-term use of voice-plus-gesture systems compared to traditional controller systems. This will help understand the performance and user preferences of different interaction systems over extended use.
- Optimization of Hand Training Educational Systems: Building on the current research findings, further development and optimization of a teaching system suitable for hand training is proposed, particularly focusing on enhancing the intuitiveness and naturalness of interaction. Additionally, incorporating more gestures and expanding the model training dataset are recommended to realize additional interactive functions, thereby meeting the diverse needs of virtual teaching.
- Evaluation of Multitasking Capabilities: Assess the effectiveness of voice-plus-gesture systems in complex operations and multitasking scenarios, compared to traditional controller systems.
- In-depth Analysis of User Experience: Conduct a thorough analysis of user experience when using voice-plus-gesture systems, including learning curves, user satisfaction, and fluidity of interaction.
- Technological Improvements and Innovations: Explore new technologies and methodologies to further enhance the accuracy and response speed of voice-plus-gesture systems.

Through these studies, we aim to further enhance the interactive experience and teaching effectiveness in virtual teaching scenarios, providing support for future educational innovations.

REFERENCES

[1] D. Peng and J. Wang, "Analysis of research hotspots and status of virtual reality in education and teaching," in *Proc. 4th Int. Conf. Educ., Knowl. Inf. Manage. (ICEKIM)*, Nanjing, China, 2023, pp. 1–5.

[2] J. M. Sáez-López, J. A. González-Calero, R. Cózar-Gutierrez, and J. del Olmo-Muñoz, "Scratch and unity design in elementary education: A study in initial teacher training," *J. Comput. Assist. Learn.*, vol. 39, no. 5, pp. 1528–1538, Oct. 2023.

[3] S. Gupta, S. Bagga, and D. K. Sharma, "Hand gesture recognition for human computer interaction and its applications in virtual reality," in *Advanced Computational Intelligence Techniques for Virtual Reality in Healthcare*, vol. 875D. Gupta, A. Hassanien, and A. Khanna, Eds. Cham, Switzerland: Springer, 2020, doi: 10.1007/978-3-030-35252-3_5.

[4] Z. Wang, H. Wang, H. Yu, and F. Lu, "Interaction with gaze, gesture, and speech in a flexibly configurable augmented reality system," *IEEE Trans. Hum.-Mach. Syst.*, vol. 51, no. 5, pp. 524–534, Oct. 2021, doi: 10.1109/THMS.2021.3097973.

[5] Y. Li, J. Huang, F. Tian, H.-A. Wang, and G.-Z. Dai, "Gesture interaction in virtual reality," *Virtual Reality Intell. Hardw.*, vol. 1, no. 1, pp. 84–112, Feb. 2019, doi: 10.3724/SP.J.2096-5796.2018.0006.

[6] C. Khundam, V. Vorachart, P. Preeyawongsakul, W. Hosap, and F. Noël, "A comparative study of interaction time and usability of using controllers and hand tracking in virtual reality training," *Informatics*, vol. 8, no. 3, p. 60, Sep. 2021, doi: 10.3390/informatics8030060.

[7] Y. Feng, "Design and research of music teaching system based on virtual reality system in the context of education informatization," *PLoS ONE*, vol. 18, no. 10, Oct. 2023, Art. no. e0285331, doi: 10.1371/journal.pone.0285331.

[8] H. Zhao, M. Cheng, J. Huang, M. Li, H. Cheng, K. Tian, and H. Yu, "A virtual surgical prototype system based on gesture recognition for virtual surgical training in maxillofacial surgery," *Int. J. Comput. Assist. Radiol. Surg.*, vol. 18, no. 5, pp. 909–919, Nov. 2022, doi: 10.1007/s11548-022-02790-1.

[9] C. Kim, C. Kim, H. Kim, H. Kwak, W. Lee, and C.-H. Im, "Facial electromyogram-based facial gesture recognition for hands-free control of an AR/VR environment: Optimal gesture set selection and validation of feasibility as an assistive technology," *Biomed. Eng. Lett.*, vol. 13, no. 3, pp. 465–473, Aug. 2023, doi: 10.1007/s13534-023-00277-9.

[10] F.-R. Sheu and N.-S. Chen, "Taking a signal: A review of gesture-based computing research in education," *Comput. Educ.*, vol. 78, pp. 268–277, Sep. 2014, doi: 10.1016/j.compedu.2014.06.008.

[11] N. Nooruddin, R. Dembani, and N. Maitlo, "HGR: Hand-gesture-recognition based text input method for AR/VR wearable devices," in *Proc. IEEE Int. Conf. Syst., Man, Cybern. (SMC)*, Toronto, ON, Canada, Oct. 2020, pp. 744–751, doi: 10.1109/SMC42975.2020.9283348.

[12] G. Li, D. Rempel, Y. Liu, W. Song, and C. H. Adamson, "Design of 3D microgestures for commands in virtual reality or augmented reality," *Appl. Sci.*, vol. 11, p. 6375, Jan. 2021, doi: 10.3390/app11146375.

[13] J. Yang, M. Chan, A. Uribe-Quevedo, B. Kapralos, N. Jaimes, and A. Dubrowski, "Prototyping virtual reality interactions in medical simulation employing speech recognition," in *Proc. 22nd Symp. Virtual Augmented Reality (SVR)*, Porto de Galinhas, Brazil, 2020, pp. 351–355, doi: 10.1109/SVR51698.2020.00059.

[14] B. Peixoto, L. C. P. Bessa, G. Gonçalves, M. Bessa, and M. Melo, "Teaching EFL with immersive virtual reality technologies: A comparison with the conventional listening method," *IEEE Access*, vol. 11, pp. 21498–21507, 2023, doi: 10.1109/ACCESS.2023.3249578.

[15] L. Toti and N. Kunicina, "The impact of virtual reality technologies on the modernization of the curricula of Western Balkan universities in the field of renewable sources," in *Proc. 10th Int. Conf. Electr., Electron. Comput. Eng. (IcETRAN)*, Jun. 2023, pp. 1–6, doi: 10.1109/IcETRAN59631.2023.10192195.

[16] M. C. Johnson-Glenberg, C. Megowan-Romanowicz, D. A. Birchfield, and C. Savio-Ramos, "Effects of embodied learning and digital platform on the retention of physics content: Centripetal force," *Frontiers Psychol.*, vol. 7, p. 1819, Nov. 2016.

[17] J. Radianti, T. A. Majchrzak, J. Fromm, and I. Wohlgenannt, "A systematic review of immersive virtual reality applications for higher education: Design elements, lessons learned, and research agenda," *Comput. Educ.*, vol. 147, Apr. 2020, Art. no. 103778.

[18] E. Campos, I. Hidrogo, and G. Zavala, "Impact of virtual reality use on the teaching and learning of vectors," *Frontiers Educ.*, vol. 7, Sep. 2022, Art. no. 965640.

[19] H. Legi, Y. Giban, and P. Hermanugerah, "Virtual reality education in era 5.0," *J. Res. Social Sci., Econ., Manage.*, vol. 2, no. 4, pp. 504–510, Nov. 2022.

[20] H. Wang and Z. Wang, "Design of simulation teaching system based on virtual reality," *Comput. Simul.*, vol. 39, no. 4, pp. 205–209, 2022.

[21] S. Wang and Z. Jiang, "Multi-person online teaching system based on virtual reality live streaming," *Comput. Appl. Softw.*, vol. 39, no. 10, pp. 132–140, 2022.

[22] Y. Zhu, "Design and research on the application of virtual reality experimental platform based on big data and cloud computing," *Invention Innov., Vocational Educ.*, vol. 864, pp. 131–135, Jan. 2021.

[23] J. Guo, G. Wang, and Y. Liu, "Research on lathe teaching training system based on virtual reality technology," *Laser Mag.*, vol. 34, no. 1, pp. 59–60, 2013.

[24] P. Soltani and A. H. P. Morice, "Augmented reality tools for sports education and training," *Comput. Educ.*, vol. 155, Oct. 2020, Art. no. 103923, doi: 10.1016/j.compedu.2020.103923.

[25] B. Xie, H. Liu, R. Alghofaili, Y. Zhang, Y. Jiang, F. D. Lobo, and C. Li, "A review on virtual reality skill training applications," *Frontiers Virtual Real.*, vol. 2, Apr. 2021, Art. no. 645153, doi: 10.3389/frvir.2021.645153.
[26] H.-S. Hsiao and J.-C. Chen, "Using a gesture interactive game-based learning approach to improve preschool children's learning performance and motor skills," *Comput. Educ.*, vol. 95, pp. 151–162, Apr. 2016, doi: 10.1016/j.compedu.2016.01.005.
[27] I. M. Rezazadeh, X. Wang, M. Firoozabadi, and M. R. H. Golpayegani, "Using affective human–machine interface to increase the operation performance in virtual construction crane training system: A novel approach," *Autom. Construct.*, vol. 20, no. 3, pp. 289–298, May 2011, doi: 10.1016/j.autcon.2010.10.005.
[28] D. Reiners, M. R. Davahli, W. Karwowski, and C. Cruz-Neira, "The combination of artificial intelligence and extended reality: A systematic review," *Frontiers Virtual Reality*, vol. 2, Sep. 2021, Art. no. 103778.
[29] J.-L. Tseng, "Intelligent augmented reality system based on speech recognition," *Int. J. Circuits, Syst. Signal Process.*, vol. 15, pp. 178–186, Mar. 2021.
[30] C. Khundam, V. Vorachart, and P. Preeyawongsakul, "A comparative study of interaction time and usability of using controllers and hand tracking in virtual reality training," in *Proc. Informat. Conf.*, 2021, p. 60.
[31] L. M. Chun, H. Arshad, T. Piumsomboon, and M. Billinghurst, "A combination of static and stroke gesture with speech for multimodal interaction in a virtual environment," in *Proc. Int. Conf. Electr. Eng. Informat. (ICEEI)*, Denpasar, Indonesia, Aug. 2015, pp. 59–64, doi: 10.1109/ICEEI.2015.7352470.
[32] J. Wang, F. Mueller, F. Bernard, S. Sorli, O. Sotnychenko, N. Qian, M. A. Otaduy, D. Casas, and C. Theobalt, "RGB2Hands: Real-time tracking of 3D hand interactions from monocular RGB video," *ACM Trans. Graph.*, vol. 39, no. 6, pp. 1–16, Dec. 2020.
[33] D.-S. Tran, N.-H. Ho, H.-J. Yang, E.-T. Baek, S.-H. Kim, and G. Lee, "Real-time hand gesture spotting and recognition using RGB-D camera and 3D convolutional neural network," *Appl. Sci.*, vol. 10, no. 2, p. 722, Jan. 2020.
[34] S. Yuan, "Depth-based 3D hand pose estimation: From current achievements to future goals," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 2636–2645.
[35] A. Tagliasacchi, M. Schröder, A. Tkach, S. Bouaziz, M. Botsch, and M. Pauly, "Robust articulated-ICP for real-time hand tracking," *Comput. Graph. Forum*, vol. 34, no. 5, pp. 101–114, Aug. 2015.
[36] K. Aditya, P. Chacko, D. Kumari, D. Kumari, and S. Bilgaiyan, "Recent trends in HCI: A survey on data glove, LEAP motion and Microsoft Kinect," in *Proc. IEEE Int. Conf. Syst., Comput., Autom. Netw. (ICSCA)*, Jul. 2018, pp. 1–5.
[37] G. Buckingham, "Hand tracking for immersive virtual reality: Opportunities and challenges," *Frontiers Virtual Reality*, vol. 2, p. 140, Oct. 2021.
[38] V. Angelov, E. Petkov, and G. Shipkovenski, "Modern virtual reality headsets," in *Proc. Int. Congr. Hum.-Comput. Interact., Optim. Robot. Appl. (HORA)*, 2020, pp. 1–5.
[39] N. Numan, D. Giunchi, B. Congdon, and A. Steed, "Ubiq-Genie: Leveraging external frameworks for enhanced social VR experiences," in *Proc. IEEE Conf. Virtual Reality 3D User Interface Abstr. Workshops (VRW)*, Shanghai, China, Mar. 2023, pp. 497–501, doi: 10.1109/VRW58643.2023.00108.
[40] K. M. Sagayam and D. J. Hemanth, "Hand posture and gesture recognition techniques for virtual reality applications: A survey," *Virtual Reality*, vol. 21, no. 2, pp. 91–107, Jun. 2017, doi: 10.1007/s10055-016-0301-0.
[41] J. J. LaViola Jr., "Context aware 3D gesture recognition for games and virtual reality," in *Proc. ACM SIGGRAPH Courses*, 2015, pp. 1–61.
[42] K. Supriya, "Trigger word recognition using LSTM," *Int. J. Eng. Res.*, vol. 9, no. 6, Jun. 2020, Art. no. 103778.
[43] P. Jiang, D. Ergu, F. Liu, Y. Cai, and B. Ma, "A review of YOLO algorithm developments," *Proc. Comput. Sci.*, vol. 199, pp. 1066–1073, Jan. 2022.
[44] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.
[45] R. W. Emerson, "Likert scales," *J. Vis. Impairment Blindness*, vol. 111, no. 5, p. 488, Sep. 2017. Accessed: Dec. 5, 2023. [Online]. Available: http://gale.com/apps/doc/A689978208/HRCA?u=anon~f9ea813c&sid=googleScholar&xid=952ff466

KE FANG was born in Chengdu, Sichuan, China, in 1983. He received the bachelor's degree in mathematics and applied mathematics from the Chengdu University of Technology, and the master's degree in software engineering from the University of Electronic Science and Technology of China. In 2006, he joined Chengdu Normal University. Throughout his academic career, he has shown a profound interest in fields, such as artificial intelligence, software engineering, virtual reality, and educational informatization. He successfully led two city-department-level research projects and actively participated in several others, making significant contributions. Furthermore, he has published numerous research articles in various domestic journals.

JING WANG was born in Jianyang, Sichuan, China, in 1975. She has led and participated in various educational reform projects with Chengdu Normal University and provincial-level projects. Furthermore, she has published research articles in several domestic journals. Her research interest includes educational informatization.