Research on a Sign Language Assisted Learning System Based on Unity Platform and Augmented Reality Technology

Hongwei Wang[1,2,*], Xin Lin[1], Qiankun Peng[1], Jinlong Yang[3], Huoquan Chen[4]
[1]*Faculty of Mathematics and Computer Science, Quanzhou Normal University, Quanzhou, China;*
[2]*Fujian Provincial Key Laboratory of Data-Intensive Computing, Quanzhou, China*
[3]*School of Education, Quanzhou Normal University, Quanzhou, China*
[4]*School of Innovation and Entrepreneurship of Quanzhou Normal University, Quanzhou, China*
**Corresponding Author*

Abstract: With the development of technology, augmented reality technology, as an effective auxiliary tool, is gradually being applied in the field of education. This study develops an interactive sign language learning system based on the Unity3D platform, aimed at ordinary people who want to learn sign language, exploring how to use modern technology to improve the efficiency and fun of sign language learning. The system adopts augmented reality technology and uses the Vuforia plugin to recognize sign language images and dynamically display corresponding sign language actions, enhancing the intuitiveness and interactivity of the learning process. Combining Blender modeling and animation functions, the system presents high-precision 3D sign language action models. Users can rotate and zoom the model through the touch screen to observe the details of the actions. The augmented reality sign language assisted learning system developed in this study provides ordinary users with a simple and convenient way of learning sign language, enhances the fun of learning, and provides new technical support for the promotion and popularization of sign language. It also makes a beneficial exploration for the application of augmented reality technology in sign language education.

Keywords: Augmented Reality; Sign Language Simulation; Vuforia; Unity3D; Educational Technology

1. Introduction

With the continuous advancement of modern science and technology, the education sector is increasingly integrating emerging technologies to facilitate rapid development [1]. The application of virtual simulation technology, particularly experimental simulation in education, is gradually becoming an important pedagogical approach. This transformation not only addresses the urgent societal demand for high-quality education but also provides new impetus for the digitalization of education. Currently, approximately 5% of the global population suffers from hearing loss. In China alone, there are 27.9 million individuals with hearing impairments [2]. Traditional sign language instruction methods exhibit numerous deficiencies; despite the availability of substantial sign language literature, learning efficiency through reading alone remains considerably low [3], failing to effectively meet the needs of sign language learners.

From a technical perspective, the rapid development of Augmented Reality (AR) technology in recent years has rendered its educational applications feasible. In chemistry education, AR technology can visualize molecular structures [4]; in biology, microscopic cellular structures can be demonstrated [5]; in physics, magnetic field distributions and electron motions can be illustrated [6]; in geography, earth models can be presented [7]. AR technology not only enriches pedagogical methods and student experiences but also enhances learning interest and engagement. The maturity of these technologies has established a solid foundation for the development of AR-based assistive learning systems for sign language.

The application of AR technology in education has gradually garnered widespread attention. Internationally, AR education research commenced earlier, with prestigious institutions such as MIT and Stanford University having initiated a series of related projects. AR and VR technologies can provide more accessible learning experiences for students with special needs. By allowing students to explore virtual environments at their own pace and according to their own conditions, these technologies can facilitate learning for individuals with physical or cognitive disabilities [8].

Aditama et al. developed an AR-based SIBI sign language learning system, which enables learners to comprehend sign language details more intuitively by overlaying virtual gesture models onto real-world environments [9]. Currently, international AR sign language learning systems have achieved significant progress in technical application; however, they still demonstrate deficiencies when processing complex gesture motions, and the maturity of relevant technologies awaits further improvement. With continuous breakthroughs in related technologies, AR sign language learning systems are anticipated to find broader applications across diverse domains, thereby further enhancing the quality and coverage of sign language education and providing more efficient and convenient learning modalities for sign language learners.

As domestic scholars have increasingly focused on sign language learning, AR technology has been gradually introduced into the design of sign language learning systems. Cheng Hongfang et al. explored the application of mobile augmented reality technology in sign language learning, proposing design concepts that utilize mobile devices such as smartphones and tablets to provide more interactive learning environments for sign language learners [10]. These studies have furnished valuable theoretical foundations and practical experience for innovations in domestic sign language education.

2. Requirements Analysis

2.1 System Content Requirements

The target users of this system are individuals interested in sign language learning without prior foundation. Consequently, the system design requirements prioritize intuitive, accessible, and engaging learning environments. These users exhibit dispersed learning schedules and locations, necessitating high portability and flexibility to accommodate mobile device usage and offline operation. Furthermore, ordinary users demonstrate elevated demands for personalized learning content, expecting tailored learning pathways and feedback based on individual progress and competency levels.

Beyond essential sign language learning functionalities, the system incorporates supplementary modules to facilitate knowledge consolidation and reference. Specifically, the sign language resource repository module displays information for gestures previously scanned through recognition markers, enabling repeated viewing of sign language motions. The sign language practice module presents multiple-choice questions regarding gesture identification, subsequently recording user accuracy rates and response durations upon completion.

2.2 Functional Requirements Analysis

To address the core requirement of enabling users to acquire fundamental sign language communication skills, this system is designed to provide an intuitive, interactive, and accessible learning experience through advanced technologies. The functional requirements are systematically specified as follows:

(1) AR Functionality. Upon scanning designated recognition markers (e.g., cards, specific identifiers) using mobile device cameras, the system leverages Augmented Reality (AR) technology to overlay 3D sign language gesture models onto real-world environments displayed on the device screen. Users can freely manipulate these 3D models through touchscreen gestures, including single-finger drag for rotation and two-finger pinch for scaling. Rotation enables observation of the gesture from multiple perspectives (frontal, lateral, and posterior views), while scaling facilitates detailed examination of critical elements such as finger joint flexion and palm orientation.

(2) Sign Language Learning Functionality. Users can activate the playback function to observe complete gesture sequences. Concurrently, the system provides clear audio narration and textual descriptions, elaborating on essential components including initial hand configuration, motion trajectory, and gesture transitions, thereby facilitating comprehension of movement essentials.

(3) Sign Language Resource Repository. The system automatically archives sign language content (3D models/animations, nomenclature, detailed descriptions) corresponding to previously scanned recognition markers into the user's personal repository. Users can rapidly locate target content through categorical browsing or keyword search (e.g., sign language

nomenclature, descriptive vocabulary). Each repository entry distinctly presents the sign language designation, descriptive text, and provides direct access to associated animated demonstrations and audio descriptions.

(4) Sign Language Practice Functionality. Upon completion of scanning all designated recognition markers, users unlock the practice module. The system dynamically generates multiple-choice questions (in the format of image-to-meaning matching) based on acquired knowledge points (e.g., specific vocabulary, phrases, or sign language categories). Upon completion of question selection, the system provides immediate feedback, explicitly indicating response correctness.

3. System Design

3.1 Functional Module Design

Based on the requirements analysis, the system is decomposed into the following functional modules, as illustrated in Figure 1. The overall system architecture comprises four distinct modules: the AR Module, the Sign Language Learning Module, the Sign Language Resource Repository Module, and the Sign Language Practice Module.

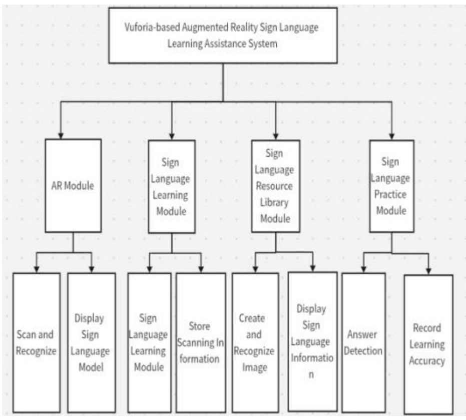

Figure 1. System Functional Module Decomposition

3.1.1 AR module design

The core design of the AR module focuses on implementing image recognition-based augmented reality interactive learning. A dedicated set of sign language recognition markers is defined and generated using Doubao AI, which can be integrated into accompanying physical picture books or learning cards. Upon scanning these markers through mobile device cameras, the system performs real-time target recognition. Upon successful identification, the system leverages the selected AR engine (Vuforia) to precisely anchor and render corresponding 3D sign language gesture models onto the real-time video stream overlaid on the device screen. The models exhibit standard, seamless motion loops. Regarding user interaction design, the system supports real-time manipulation of models through multi-touch gestures, including single-finger translation for rotation and two-finger pinch for scaling, enabling users to observe hand details (e.g., finger joint flexion, palm orientation) and motion trajectories from arbitrary perspectives.

3.1.2 Sign language learning module design

The sign language learning module collects and organizes extensive commonly used sign language vocabulary. Through 3D modeling technology, refined hand models are constructed, and precise, seamless standard motion animation sequences are generated for each sign language term. Each animation is synchronized with clear audio narration (or detailed textual descriptions) elaborating on movement essentials, semantic meaning, and usage contexts. The user interface design provides intuitive playback controls (play/pause/progress bar) for viewing animations and listening to explanations. The design core lies in providing multi-sensory input through audio-visual integration, deepening users' comprehension of sign language gesture morphology, meaning, and application.

3.1.3 Sign language resource repository module design

The core of the resource repository module is the design of an efficient, scalable data storage and management system. A lightweight database is constructed using JSON format, with structural design encompassing key attributes for each sign language entry: vocabulary designation, detailed descriptive text, corresponding 3D animation resource paths, and potential categorical tags. Based on the aforementioned data structure, a user-friendly browsing interface is designed. The interface clearly presents entry listings and keyword search functionality. Upon user selection of a specific entry, the interface design seamlessly loads and displays the complete information for that entry (designation, description, associated animation/video player). Navigation controls (e.g., "Previous Entry," "Next Entry") are designed for convenient traversal of the resource repository.

3.1.4 Sign language practice module design

The core design of the practice module is to provide an intelligent assessment and feedback mechanism for consolidating learning outcomes. The primary format employs multiple-choice questions, with question types designed as image-to-meaning matching. Question content is precisely aligned with core vocabulary and knowledge points from the learning module and resource repository. Upon user response submission, the system provides immediate feedback, explicitly indicating answer correctness. Upon completion of all questions, the system records response accuracy rates and completion duration.

4. System Implementation

4.1 Sign Language Modeling

In this system, Blender is employed for the production of 3D models and animations of sign language gestures. Realistic hand models are created through Blender's modeling tools, and skeletal structures are added to the models using rigging functionality. The completed models and animations are exported in formats supported by Unity (e.g., FBX) for loading and presentation within the Unity environment.

4.1.1 Hand model construction

Hand modeling is performed with the addition of sleeves, background elements, and lighting configurations to ensure optimal visual effects. The hand modeling process and its rendered output are illustrated in Figure 2. Subsequently, bones are added to the hand to facilitate gesture animation production. The effect of adding skeletal structures to the hand model is shown in Figure 3.

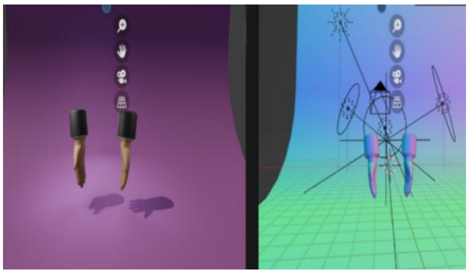

Figure 2. Hand Model and Its Rendered Output

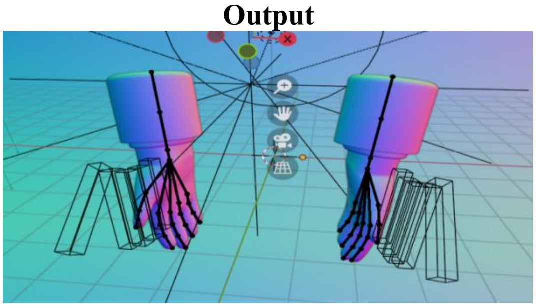

Figure 3. Hand Model with Skeletal Structure

4.1.2 Gesture animation production

All bones are selected and pose mode is activated. Through right-click context menu, "Insert Keyframe" is selected with "Location + Rotation" parameters to record the initial static pose. The timeline cursor is then moved to the subsequent frame, hand configuration is adjusted, and keyframe insertion is repeated. Pressing the spacebar enables preview of hand motion transitions. This operation is iteratively executed to record each incremental hand pose change. At the final frame, the initial static keyframe is duplicated to return to the original pose, as illustrated in Figure 4.

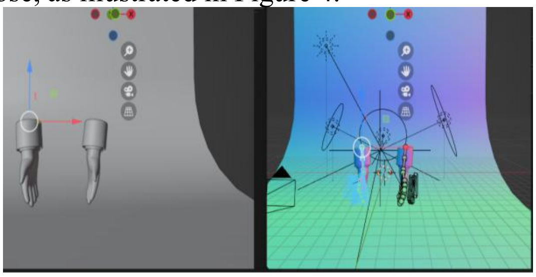

Figure 4. Gesture Animation Production Process

4.2 AR Scanning Module

The AR scanning module is implemented through Vuforia technology, enabling users to scan specific images using mobile device cameras to trigger corresponding sign language gestures. The module performs real-time image recognition and presents sign language gestures within the augmented reality environment, providing interactive experiences.

4.2.1 Scanning database acquisition

To enable image recognition through Vuforia, target images must first be uploaded to the Vuforia official website. Upon logging into the Vuforia platform, the Target Manager is accessed, and Generate Database is selected to create a new database. Images are subsequently uploaded to this database, which is then downloaded from the website as a Unity package. The acquisition of the Unity package for target images is illustrated in Figure 5.

Figure 5.Get Unity Asset Package

4.2.2 Scanning module construction

A new scene is created, with the default camera removed and replaced by an "ARCamera," which utilizes the rear-facing camera by default (front-facing camera when running on desktop environments). An ImageTarget is added to the hierarchy, with the desired scanning image selected (options become available only after importing the Unity package). The FBX model exported from Blender is assigned to the ImageTarget, with gesture positioning adjusted accordingly. As each ImageTarget corresponds to a single image, sufficient ImageTargets are created and individually named according to the quantity of sign language gestures, completing the scanning configuration. Selection of corresponding scanning images is illustrated in Figure 6.

Figure 6. Selection of Scanning Images and Gesture Assignment

4.2.3 Scanning action recording

The system requires tracking whether gesture images have been scanned. This is achieved through the observer behaviour component monitoring AR target image recognition status. When a target image is recognized (Status. TRACKED), the function Unlock Sign on Target Detected is triggered. This function reads sign language vocabulary data from the JSON file, locates the vocabulary entry matching the recognized image name, and sets the is Unlocked field of this entry to 1, indicating that the sign language gesture has been unlocked. Subsequently, the updated data is written back to the JSON file to ensure persistent storage.

The partial JSON structure is as follows:

```
{
"signName": "BYE",
"imageName": "Bye",
"description": "",
"videoFile": "Bye.mp4",
"isUnlocked": 0
}
```

The core implementation code with annotations is presented below:

```
UnlockSignOnTargetDetected(string imageName)
 {
string jsonData = File.ReadAllText(jsonPath);
SignDatadata=JsonUtility.FromJson<SignData>(jsonData);
var entryToUnlock = data.entries.FirstOrDefault(e => e.imageName == imageName);
    if (entryToUnlock != null)
    {
      entryToUnlock.isUnlocked = 1;
          File.WriteAllText(jsonPath, JsonUtility.ToJson(data, true));
      Debug.Log("Unlocked sign: " + entryToUnlock.signName);
    }
    else
    {
      Debug.LogWarning("No matching sign found for image: " + imageName);
    }
}
```

4.3 Sign Language Resource Repository Module

The sign language resource repository module presents detailed information for sign language gestures, including gesture nomenclature, animated videos, and descriptive text. Users can browse different sign language gestures through "Previous" and "Next" navigation controls, and view gesture animations through playback control buttons. This module provides an intuitive interface to facilitate users' comprehension and retention of individual sign language gestures, constituting a critical component of sign language learning, as illustrated in Figure 7.

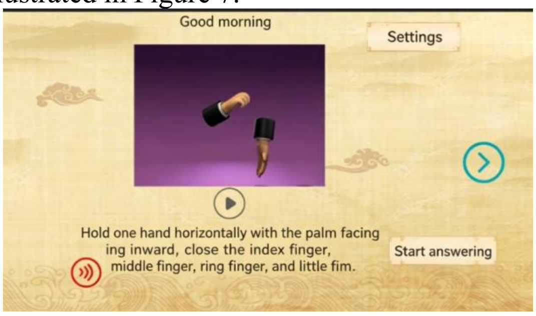

Figure 7. Sign Language Resource Repository Module Interface

The system initializes video playback through the VideoPlayer component. The VideoPlayer component requires configuration of the video URL path, typically set via the videoPlayer.url property. Upon video preparation completion, playback is initiated through the videoPlayer. Play () method. User control over video playback state is enabled through PlayVideo and PauseVideo methods, as illustrated in Figure 8

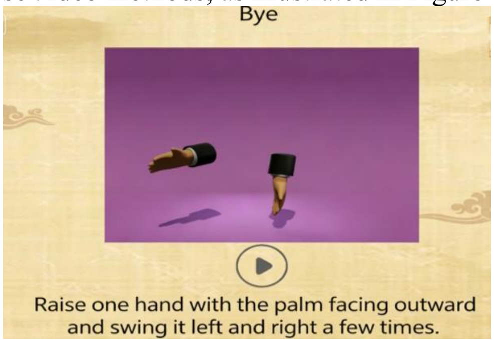

Figure 8. Video Playback and Display

4.4 Practice Module

The practice module facilitates consolidation of sign language knowledge through multiple-choice questions. The system records user response outcomes, including accuracy rates and completion duration, to assess learning progress. Through iterative question answering, the module reinforces users' retention and application capabilities of sign language, logging learning data to promote improvement in learning effectiveness.

4.4.1 Question Generation

The system employs a randomized strategy, extracting non-repetitive questions from a predefined answer Datas list through the Get Random Numbers method, ensuring diversity and fairness in practice. Concurrently, the Get Random Int method randomly determines answer positioning (left or right), enhancing randomness and preventing users from exploiting pattern recognition. The SetTiMu method is responsible for configuring the current question, including video playback and answer option presentation, as illustrated in Figure 9.

Figure 9. Practice Module Interface

The core implementation code with annotations for this functionality is presented below:

4.4.2 User interaction

User interaction constitutes the core of the question-answering system. The On Click Answer method processes user responses, evaluates answer correctness, and provides immediate feedback through the result Text UI component. This immediate feedback mechanism not only enhances user engagement but also enables real-time assessment of learning effectiveness. Additionally, the implementation incorporates delayed processing of user operations (via the Invoke method), ensuring sufficient duration for users to review feedback results.

4.4.3 Historical record management

The system reads and updates user practice history through the Load Historical Data and Update Historical Data methods. These data are stored in JSON format files, ensuring data persistence and cross-platform compatibility. The Save Leader board method persists updated historical data to the file, as illustrated in Figure 10.

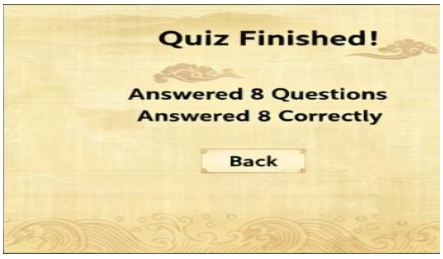

Figure 10. The Quiz Completion Page

The partial core code and comments for implementing this functionality are as follows:

```
if (curAnswerIndex >= extractNum)
{
overRightNumText.text =
$"{rightAnswerNum}";
overUiObj.SetActive(true);
historicalData = LoadHistoricalData();
for (int i = historicalData.resultDatas.Count-1;
i >= 0; i--)
{
GameObject item=
GameObject.Instantiate(historyItemPrefab.game
Object, contentTran);
item.transform.localScale = Vector3.one;
item.transform.localEulerAngles = Vector3.zero;
private void SetTiMu()
{
curAnswerIndex += 1;
aRawImage.GetComponent<Button>().interactable =
true;
```



```
bRawImage.GetComponent<Button>().interactable =
true;
resultText.gameObject.SetActive(false);
if (curAnswerIndex < extractNum)
{
String
videoPath=Path.Combine(Application.streamingAsse
tsPath,
answerDatas[randomNumbers[curAnswerIndex]].vid
eoName);
videoPlayer.url = videoPath;
videoPlayer.prepareCompleted += (source) =>
{
Debug.Log("Video prepared successfully.");
videoPlayer.Play();
};
videoPlayer.Prepare();

curTimuNumText.text = $"{(curAnswerIndex +
1)},{extractNum}";
rightNumText.text = $"{rightAnswerNum}";
}
else
{
}
}
item.transform.GetComponent<HistoryItem>().
SetData(historicalData.resultDatas[i].timer,
historicalData.resultDatas[i].rithtNum);
creatHistoryItems.Add(item.transform.GetComp
onent<HistoryItem>());
item.SetActive(true);
}
UpdateHistoricalData(GetCurDataAndTime(),
rightAnswerNum);
}
```

4.5 System Publishing

In Unity Hub, locate the project version (2022.3.34f1c1), click Add Module, and select the Android module

Open the project, click File → Build Settings, drag in the scenes to be configured, select the Android platform, click Switch Platform, and then click Build. Transfer the exported APK file to an Android device via WeChat. The exported APK file is shown in Figure 11.

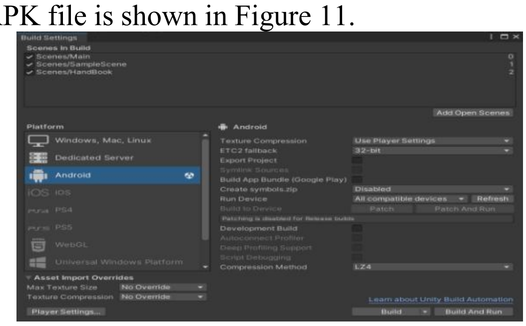

Figure 11. Exporting the APK File

5. Conclusion

This study developed an interactive sign language learning system based on Vuforia. By leveraging AR technology, it creates an intuitive, vivid, and highly interactive learning platform for sign language learners. Based on user requirements, the system implements core functionalities such as real-time sign language demonstration, personalized learning paths, and multi-sensory learning experiences. It also enhances learning engagement and efficiency through mechanisms including randomized question generation and history management. The application of modern technologies enriches the ways of learning sign language, effectively accommodates diverse user needs, and provides new insights for the popularization of sign language education.

Acknowledgments

This work was supported in partly by the Natural Science Foundation of Fujian Provincial Science and Technology Department under Grant No.2022J011108; partly by Fujian Provincial Science and Technology Program Guidance Project under Grant No.2024H0019; partly by the Fuzhou- Xiamen- Quanzhou National Independent Innovation Demonstration Project under Grant No.2024QZFX04; partly by the Fujian Social Science Project of China under Grant No.FJ2022B067.

References

[1] Yu T. Y., Xiong S. P., & Zhang M. Z. (2023). Research on the dissemination effect of AR technology in preschool education. Education and Teaching Forum, (35), 45-48.

[2] Li X. M. (2017). Research on Chinese sign language recognition for small-to-medium vocabulary based on neural networks (pp. 1-10). University of Science and Technology of China, Hefei. (Doctoral dissertation)

[3] Zhang, Yanxiao, Yuecong Min, and Xilin Chen. "Teaching chinese sign language with a smartphone." Virtual Reality & Intelligent Hardware 3.3 (2021): 248-260.

[4] Zhu P. F. (2019). Research on augmented reality (AR) technology promoting high school students' learning of chemical microstructure. Chemistry Teaching, (9), 34-38.

[5] Cai S., Li Y. T., Zhao Q., et al. (2018).

Exploring microscopic cells with augmented reality technology — The 11th empirical case of AR in K-12 education. Information Technology in Primary and Secondary Education, (12), 66-68.
[6] Hou D., & Song H. Z. (2018). Application research of 3D virtual augmented reality technology in middle school physics teaching. Journal of Tianjin Normal University (Elementary Education Edition), 19(3).
[7] Tong Q., & Hong Y. (2023). Design and development of junior high school geography teaching software based on augmented reality technology. Journal of Hubei Normal University (Natural Science Edition), 43(4), 62-67.
[8] Al-Ansi, Abdullah M., et al. "Analyzing augmented reality (AR) and virtual reality (VR) recent development in education." Social Sciences & Humanities Open 8.1 (2023): 100532.
[9] Aditama P W, Putra P S U, Yusa I M M, et al. Designing augmented reality sibi sign language as a learning media//Journal of Physics: Conference Series. IOP Publishing, 2021, 1810(1): 012038.
[10] Cheng H. F. (2018). Application research of mobile augmented reality technology in sign language learning teaching. Journal of Wuhu Vocational and Technical College, 20(2), 81-83.

