ingegneria sismica
International Journal of Earthquake Engineering
Anno XLIII - Num. 3 - 2026

# Analysis of the Promoting Effect of Immersive Game Design Based on AR Technology on Players' Cognitive Experience

*Jing Fu*[1,*], *Chan Tang*[1] *and Jinhe Du*[1]

[1] School of Computer Science and Engineering Guangzhou Institute of Science and Technology, Guangzhou, Guangdong, 510000, China

**SUMMARY:** *Augmented reality (AR) technology is a technique that overlays virtual digital information onto the real world, enhancing users' perception of real-world games. In the field of augmented reality, improving players' cognitive experiences is a critical task. This paper extracts feature spaces from game images, selects the optimal model to fit their changes, and matches the game image data. Using the SIFT algorithm and ORB algorithm, the overall contrast of the images is enhanced. The G-AR algorithm is employed to enhance players' immersive experience in object grasping within the game. Additionally, the object grasping algorithm is optimized using Reinforcement Learning with Deep Deterministic Q-Network (RGRL) based on DDQN. A Unity scene is constructed, and through a series of scripts and UI interface designs, the overall game scene is designed. Through a questionnaire survey, the correlation between cognitive challenges and immersive experience is analyzed. AR technology and immersive experience. The significance coefficient for cognitive challenge and immersion is 0.4855, and the Pearson correlation coefficient for AR challenge and immersion is 0.7188, with a significance coefficient of 0.0248. The average scores for the metrics in scenes 1–4 are 3.8501, 2.0899, 3.1095, and 3.5531, respectively. Scene 2's score is below 3, indicating room for improvement in Scene 2.*

***KEYWORDS:*** *feature space; SIFT algorithm; G-AR algorithm; immersive experience; game scene design*

## 1 Introduction

Driven by the wave of digitalization and informatization, China's gaming industry is developing at an unprecedented pace. Augmented reality (AR) technology, with its unique immersive experience, has brought gamers a brand-new interactive experience [1-4]. AR technology is a new technology that seamlessly integrates real-world information with virtual-world information [5]. The three key features of AR technology are precisely what is needed for user experience in game design. First, the integration of virtual and real elements. Games bring real-world players into virtual gaming environments, which are distilled from real life. Games blend virtual and real elements, making AR and gaming a natural fit [6-8]. Second, real-time interaction. Game controls require players to respond in real time, with gameplay dictated by their immediate actions. If player behavior can be reflected in real time within the game environment, it significantly enhances immersion [9-11]. Finally, AR technology can add player behavior and real-world scenes to three-dimensional virtual games, enhancing the game's playability [12, 13].

*17702071890@163.com
https://doi.org/10.65102/is20261166
Published: 10 June, 2026

Augmented reality interaction technology has always been an important research topic both domestically and internationally. Zhang, B. developed a complete mobile augmented reality game development system based on AR technology for real-time strategy (RTS) games, optimizing existing technology through image recognition strategies using the SIFT feature matching algorithm, aiming to meet users' needs for information expansion [14]. Zuo, T. et al. investigated the relationship between students' autonomy and sense of presence in two learning scenarios—fantasy and everyday—to fully understand the implications of fantasy learning experience design. This provides guidance for designing more immersive games [15]. An, S. et al. introduced a program that does not require additional hardware to obtain 3D environmental information, namely the AR CORE SDK, and combined it with the Unity 3D game engine, significantly enhancing the development efficiency of augmented reality games while also improving user accessibility and usability [16]. Shafana, A. R. F. and Silpasuwanchai, C. studied the impact of gesture interaction and screen size on immersive AR game experiences, finding that touch and tilt gestures each have their advantages, while screen size has a relatively minor impact on player experiences, providing important insights for AR game designers [17]. However, a game that relies solely on technological uniqueness cannot sustainably attract and satisfy users' growing emotional needs. Therefore, how to deepen users' cognitive experiences in augmented reality games is a pressing issue in the field of immersive game design.

This paper first enhances game scene images using two algorithms, SIFT and ORB, to extract feature spaces for matching, match image information, and improve overall image contrast. The G-AR algorithm is proposed to enhance the sense of object grasping in game scenes, further optimizing the immersive experience. Based on this, the robot grasping algorithm RGRL, optimized by the DDQN algorithm, is incorporated to improve grasping accuracy. AR-related API image processing interfaces are introduced to build a Unity scene. Through script and UI interface design, game interface management and processing logic are implemented to complete an immersive game design based on AR technology. Relevant experiments are designed to test players' emotional values and cognitive engagement metrics after the game experience, analyzing the relationship between AR technology, immersion, and cognition.

# 2 Immersive game design based on AR technology

## 2.1 Implementation of the image matching module in mobile games

### 2.1.1 Analysis of Image Matching Methods

Figure 1 shows the flowchart of the image matching algorithm. The image matching process varies widely depending on the application domain, with hundreds or even thousands of different methods available. However, regardless of the method used, the process consists of the following three steps [18].

First, information used for matching must be extracted from the image, also known as the feature space. This can include overall or local grayscale information, image edges, contours, or point features. The extracted features must meet three requirements: they must be common to both images, they must be evenly distributed, and they must be easy to match. The suitability of the feature space selection has a significant impact on the efficiency of image matching. Next, a metric to determine the similarity between discriminative features, also known as a similarity criterion, must be established. The determination of this criterion varies depending on the selected feature space. Finally, based on the geometric changes between the two images to be matched, the optimal model to fit these changes, also known as the search strategy, must be selected. The choice of search strategy will largely determine the speed of the matching process.

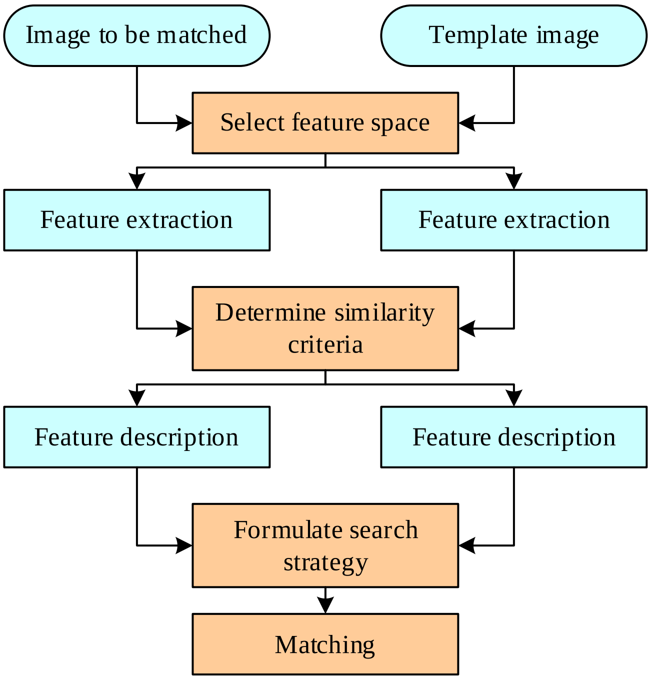


*Figure 1: The general process of image matching*

In various application domains, constrained by the characteristics of image data and the focus of matching, a wide variety of image matching methods have emerged. Based on the approach used during matching, these methods can be broadly categorized into three types: grayscale-based matching algorithms, feature-based matching algorithms, and deep learning-based matching algorithms. This paper will focus on the first two methods.

(1) Grayscale-based matching algorithms

The fundamental principle of grayscale-based matching algorithms is to move the template image across the target image and identify the most similar regions as the matching result. Whether two images match depends on whether they share similar grayscale information or grayscale statistical information. Commonly used methods include the Mean Absolute Difference (MAD) algorithm, the Sum of Absolute Differences (SAD) algorithm, the Sum of Squared Differences (SSD) algorithm, the Sequential Similarity Detection Algorithm (SSDA), and image hashing algorithms.

The Mean Absolute Difference (MAD) algorithm is a matching algorithm proposed in 1971 and is a commonly used method in pattern recognition. Its basic principle involves using the template image to traverse the target image, measuring the gray-scale differences at corresponding positions, and identifying the most similar portion as the matching result.

Assume that $S(x, y)$ is an image of size $A \times B$ to be matched, and $T$ is a template image of size $P \times Q$. In the image to be matched, take the image of size $M \times N$ with the upper-left corner at $(i, j)$ and denote it as $S^{i,j}$. Let $S^{i,j}(m,n)$ and $T(m,n)$ represent the grayscale values of the subgraph and template graph at position $(m,n)$, respectively. The MAD algorithm formula is expressed as:

$$D(i,j)=\frac{1}{P\times Q}\sum_{m=1}^{P}\sum_{n=1}^{Q}\left|S^{i,j}(m,n)-T(m,n)\right| \tag{1}$$

In the above equation, $D(i,j)$ represents the similarity measure of the corresponding point at $(i,j)$ in the template graph $T$ in the graph to be matched. When $D(i,j)$ reaches its minimum value, the corresponding $(i,j)$ is the best matching position.

The SAD algorithm is basically the same as MAD, and is expressed by the following formula:

$$D(i,j)=\sum_{m=1}^{P}\sum_{n=1}^{Q}\left|S^{i,j}(m,n)-T(m,n)\right| \tag{2}$$

The SSD algorithm is basically the same as SAD, and is expressed by the following formula:

$$D(i,j)=\sum_{m=1}^{P}\sum_{n=1}^{Q}\left[S^{i,j}(m,n)-T(m,n)\right]^2 \tag{3}$$

$$\varphi(i,j,m,n)=\left|S^{i,j}(m,n)-\bar{S}(m,n)-T(m,n)+\bar{T}(m,n)\right| \tag{4}$$

In this context, $\bar{S}(m,n)$ denotes the average gray value of all pixels in the subgraph region, while $\bar{T}(m,n)$ denotes the average gray value of all pixels in the template image.

Next, in the region covered by the template image, the gray-scale error $\varphi$ between the corresponding pixel points of the subgraph and the template image is calculated. The absolute error values of randomly selected pixel points are linearly accumulated, and when their accumulated value exceeds the threshold $T_h$, the subgraph is determined to be mismatched with the template image. The detection function can be defined as:

$$D(i,j)=\left\{H\left|\min_{1\le H\le P\times Q}\left[\sum_{n=1}^{H}\varphi(i,j,m,n)\ge T_h\right]\right.\right\} \tag{5}$$

Here, $H$ denotes the cumulative count at this point. The position $(i,j)$ with the maximum value of $D(i,j)$ is the optimal matching point, as this position has more pixels similar to the template image under conditions of lower sum of errors.

(2) Feature-based matching algorithms

Feature-based matching algorithms primarily extract features from images, use similarity metrics and constraints to determine geometric transformations between images, and apply these transformations to the images to be matched. Two typical feature-based image matching algorithms are SIFT and ORB.

### 2.1.2 Optimization of feature-based image matching methods

In Section 2.1.1, two commonly used feature-based matching methods—SIFT and ORB—were mentioned. The SIFT algorithm is a classic image matching algorithm. Each feature point extracted generates a 128-dimensional feature vector for description. The Euclidean distance is then calculated to determine the optimal matching position. However, this process involves a significant computational load, resulting in low real-time performance in general application scenarios. The ORB algorithm uses binary descriptors to describe feature points. only requiring the calculation of the Hamming distance to determine the optimal matching position. This method has a low computational load and is fast in most cases, but the binary descriptors also result in a higher false matching rate. Figure 2 shows the algorithm flow. To improve matching efficiency, this paper first enhances the image and then uses the above image matching methods for matching to increase the number of correct matching point pairs. The image enhancement method involves smoothing and sharpening the two images to be matched, then overlaying them on the original image at a certain ratio to enhance the overall contrast of the image.

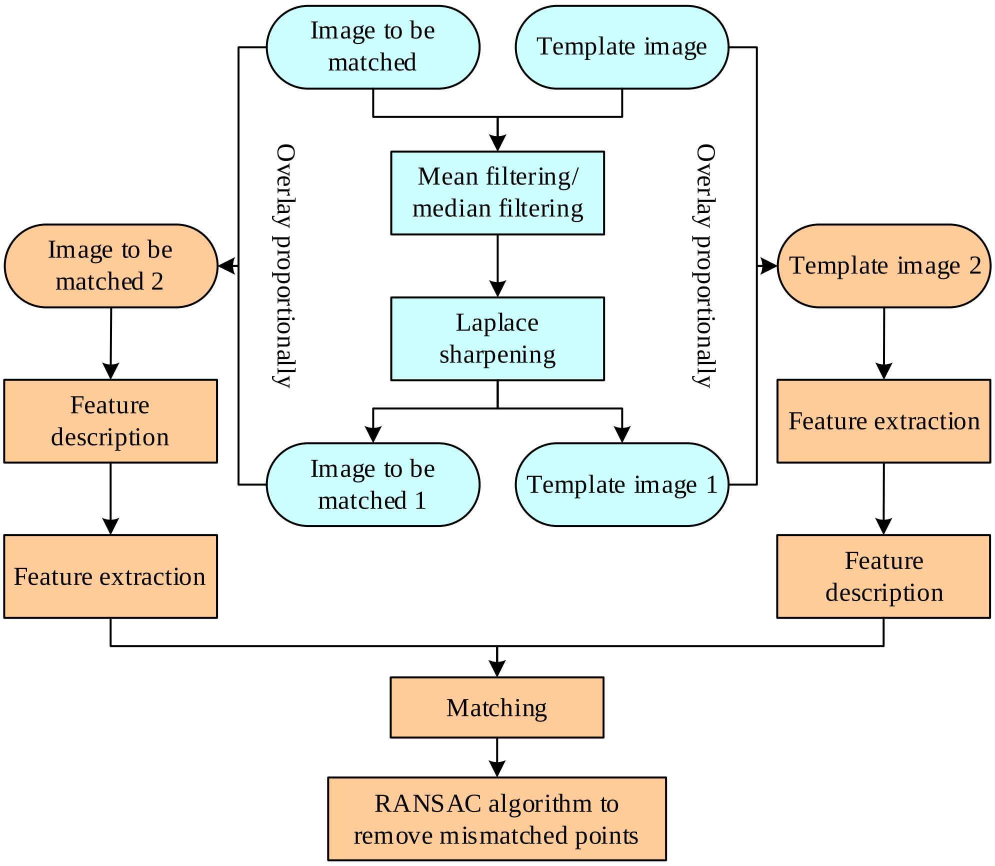


*Figure 2: The algorithm process adopted in this paper*

Step 1: Smooth the image using mean filtering or median filtering.

To prevent unnecessary noise information from being enhanced when enhancing image features in the next step, it is necessary to first smooth the image to remove noise. The most commonly used method for smoothing images is image filtering, which can suppress image noise while retaining image detail features. Common image smoothing methods include mean filtering and median filtering. Both belong to smoothing filters, but the former is a smoothing linear filter, while the latter is a smoothing nonlinear filter.

Mean filtering is the simplest type of filter, primarily using the neighborhood averaging method. It replaces the pixel value of a point in a digital image with the average value of all pixel values in its neighborhood. After processing, the grayscale at point $(x, y)$ in the image can be expressed as:

$$g(x,y)=\sum f(x,y)/m \tag{6}$$

Among them, $m$ represents the total number of pixels contained in the template.

Median filtering is based on sorting statistics theory, the principle of which is to replace the pixel value of a point in a digital image with the median of all pixel values in its $k*k$ neighborhood window. After processing, the grayscale at image point $(x,y)$ can be expressed as:

$$g(x,y)=med\left\{f(x-k,y-k),(k\in W)\right\} \tag{7}$$

In this context, $W$ represents the size of the two-dimensional template, typically chosen as $3*3$ or $5*5$.

Step 2: Sharpen the image using the Laplacian operator

After reducing image noise using the above method, further sharpening of the image can be performed to emphasize feature information and enhance image details. Image sharpening, also known as edge enhancement processing, enhances edge information and gray-scale transition areas in an image, thereby improving overall contrast and making the image clearer. Unlike the image smoothing process mentioned earlier, image sharpening achieves this by increasing the gray-scale difference between a point in the image and its neighboring pixels.

Among the methods of image sharpening, the simplest and most commonly used method is the Laplacian operator, which is a linear quadratic differential operator [19]. Its basic idea is: calculate the average gray value of other pixels in the neighborhood centered on a certain point. If the gray value of that point is low, further reduce its gray value; if the gray value of that point is high, further increase its gray value. The Laplacian operator is defined as:

$$\nabla^2 f(x,y)=\frac{\partial^2 f(x,y)}{\partial x^2}+\frac{\partial^2 f(x,y)}{\partial y^2} \tag{8}$$

Its expression can be written as:

$$\begin{aligned}\nabla^2 f(x,y)= f(x+1,y)+f(x-1,y)+f(x,y-1)\\+f(x,y+1)-4f(x,y)\end{aligned} \tag{9}$$

The enhancement operator is represented as:

$$\begin{aligned}g(x,y)=f(x,y)-\nabla^2 f(x,y)=5f(x,y)-f(x+1,y)\\-f(x-1,y)-f(x,y-1)-f(x,y+1)\end{aligned} \tag{10}$$

The commonly used Laplace operator four-neighborhood template was formed:

$$\begin{bmatrix}0 & -1 & 0\\ -1 & 5 & -1\\ 0 & -1 & 0\end{bmatrix} \tag{11}$$

In addition, the eight-neighbor template matrix is also commonly used, as shown below:

$$\begin{bmatrix} -1 & -1 & -1 \\ -1 & 9 & -1 \\ -1 & -1 & -1 \end{bmatrix} \tag{12}$$

## 2.2 Game Design Based on Augmented Reality

### 2.2.1 Immersive interaction technology based on augmented reality

There are many reasons for discrepancies between the operator and the intelligent robot's decisions. For example, the neural network may fail to accurately scan the entire surface of the target during target detection, thereby affecting the determination of the grasping position. Large-area obstructions to the operator's view of the target's grasping angle can also affect the intelligent robot's extraction of position points. Therefore, the grasping success rate calculation defined in the study is shown in Equation (13):

The purpose of the immersive game designed in this paper is to enhance players' cognitive skills, including thinking ability, memory, reaction speed, and logical reasoning. The development of mobile internet technology has given rise to AR technology, which has elevated the playability and richness of games to new heights. AR technology can real-time calculate the angle and position of the images captured by the camera and add attributes and materials to the acquired points to present new images. Through AR technology, games can provide a more immersive experience, allowing players to feel more real and engage in deeper interactions [20]. In AR technology, the operator does not feel a sense of physical contact when grabbing objects displayed on the screen, which weakens the sense of manual grasping and the immersive experience. To address this issue, the study proposed the G-AR algorithm, whose core idea is to identify potential grasping positions through image detection, then present these positions via augmented reality, allowing the operator to gaze at all potential grasping positions and subjectively select the optimal grasping position. The G-AR algorithm comprises three essential modules: the intelligent robot subjective interaction module, the G-AR grasping point decision module, and the potential grasping point detection module. The specific description of the G-AR algorithm is as follows: Input: gaze position, RGB depth image, and color image. Output: robot grasping position. Intermediate steps: use the RGB image to determine candidate grasping positions via the YOLOv5 network. Assuming $N$ candidate grasping positions, where $N$ is greater than or equal to 1, the candidate grasping positions can be output to the imaging software. The G-AR algorithm combination is illustrated in Figure 3.

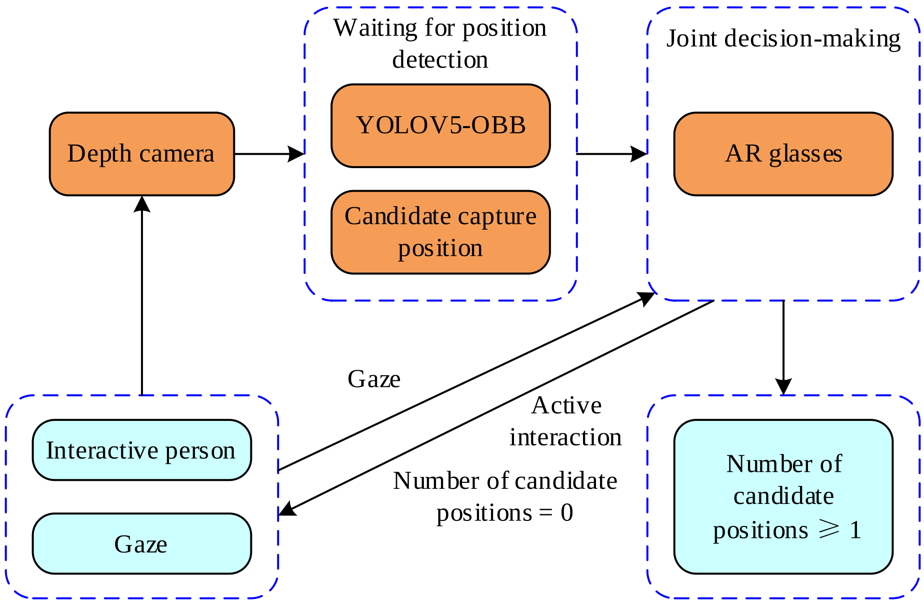


*Figure 3: G-AR algorithm module combination*

$$Acc = \frac{N_{success}}{N_{num}} \times 100 \tag{13}$$

In Equation (13), $N_{success}$ denotes the total number of successful and failed attempts during the grasping process, and $N_{num}$ denotes the total number of tests conducted during the grasping process. After resolving the grasping decision problem, the final grasping operation can be performed once the operator and the intelligent robot reach a consistent decision.

### 2.2.2 Improved Immersive Game Interaction Design

Since the intelligent robot system in the game needs to move directly above the target grasping point when grasping the target, it must not cause damage to the grasping target or accidentally injure the operator during this process. Therefore, this study proposes an improved reinforcement learning-based intelligent robot grasping optimization algorithm, RGRL, based on DDQN. This grasping algorithm requires data on the operator's hand position and the size of the grasping target, as well as the positional and angular relationships between the grasping target and the hand, enabling it to effectively avoid the hands and accurately grasp the target object. The DDQN algorithm structure is shown in Figure 4.

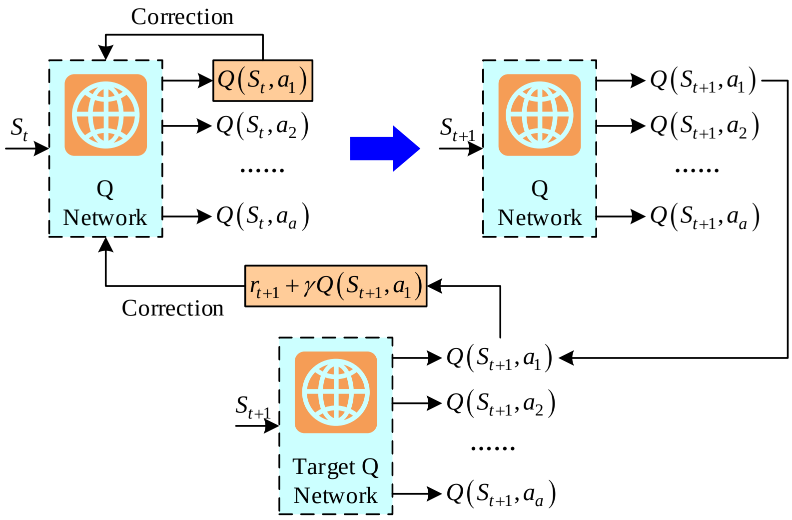


*Figure 4: Data update structure of DDQN algorithm*

In the figure, $Q$ represents the input state of the network, and the output is the $Q$ value corresponding to various actions. The more accurate the $Q$ value, the better the training effect of the $Q$ network. When selecting $Q(S_{t+1}, a_{t+1})$, DDNQ first finds the action corresponding to the maximum output value in the $Q$ network, and then finds the output value of the TargetQ network corresponding to this action. The expressions within the brackets represent known items, and all known items—state, action, reward, and updated state—during the execution of the RGRL algorithm are stored in the memory bank [M]. The RGRL algorithm primarily consists of two structurally consistent networks, $q-eval$ and $q-t\arg et$. $q-eval$ is responsible for outputting the grasping action to be adopted in the given state, while $q-t\arg et$ is responsible for evaluating the value generated by that action. The network operation process of DDQN is calculated as shown in Equation (14):

$$Q(s,a;\theta) = r + \gamma^* Q\left(s', \arg\max_a Q\left(s', a; \theta^-\right)\theta\right) \tag{14}$$

In equation (14), $\theta$ represents the current state of the network $q-eval$, $\gamma$ represents the reward decay parameter, $\theta^-$ represents the state of the entire RGRL network, $\theta^-$ represents the detailed numerical value of $q-t\arg et$, $a$ represents the grasping action under $\theta^-$, $r$ represents the reward obtained by the grasping action $a$ under $\theta^-$, and s' denotes the updated state. Thus, the optimal action calculation for the $q-eval$ network in real-time is shown in Equation (15):

$$a = \arg\max_\alpha Q(s, A; \theta) \tag{15}$$

In equation (15), $A$ represents the action interval in the current state, and $\alpha$ represents the maximum action. The RGRL algorithm treats the relative distance between the intelligent robot and the operator's hand as the state criterion, as shown in equation (16):

$$\begin{gathered} pos_{hand}, m = pos_{hand}, m + random\left(-\Delta s_m, \Delta s_m\right) \\ \left(\min_m < pos_{hand}, m < \max_m, m \in \{x, y, z\}\right) \end{gathered} \tag{16}$$

In Eq. (16), $pos_{hand}$, $m$ represents the angle and position of the $m$ axis of the manipulator's hand in the simulation space, and the $random$ function needs to be randomly selected from the input values $(-\Delta s_m, \Delta s_m)$, $\Delta s_m$ represents the random moving distance of the manipulator's hand in the simulation scene, $\min_m$ represents the minimum moving distance, and $\max_m$ Indicates the maximum distance traveled. A crucial step in the RGRL algorithm is to trigger the reward mechanism, and the appropriate reward function has a great effect on improving the efficiency of the entire grasping algorithm. The reward function is calculated as shown in Eq. (17):

$$r = \begin{cases} 80, & 1 < \Delta x < 5 \ and \ 5 < \Delta y < 8 \ and \ 8 - 1 < \Delta z < 2 \\ -50, & Collision \ occurred \\ -5, & l_{now} \geq l_{last} \\ 5, & l_{now} < l_{last} \\ -10, & Out \ of \ range \end{cases} \tag{17}$$

In Equation (17), $l_{now}$ represents the current shortest distance between the intelligent robot and the target, while $l_{last}$ represents the shortest distance between the intelligent robot and the target in the previous step. $\Delta x$ represents the distance between the operator's hand and the intelligent robot's gripper on the $x$ axis, $\Delta y$ represents the distance between the operator's hand and the intelligent robot's gripper on the $y$ axis, and $\Delta z$ represents the distance between the operator's hand and the intelligent robot's gripper on the $z$ axis. Combining the above equations, the steps of the proposed RGRL algorithm can be preliminarily presented. First, the input includes the operator's hand angle and position, robot position, $\gamma$, maximum number of interactions, interaction time, and number of targets. Second, the output is the optimal grasping action $\alpha$. Finally, the operator makes a subjective judgment based on the output of the optimal grasping action.

# 3 Immersive Game Design and Implementation

## 3.1 Game Development

Since this game is an AR game, we first introduced AR-related APIs and implemented the corresponding image processing interfaces for these APIs. Next, since the game's algorithms require image processing technology, we had to reference the corresponding DLL package for image conversion. Finally, we set up the Unity scene and added physics engines and animation processing components to the scene objects.

## 3.2 Game Script Design

The game is primarily divided into three modules: the AR module, the image processing module, and the game control module. The game control module includes the UI management module, the character control module, and the game management module. The design is scripted using the MVC pattern, and the scripts are associated with objects in the relevant scenes to achieve object control and game logic processing.

## 3.3 UI Interface Design

The UI interface is primarily designed to display game-related data for users. To ensure user-friendly operation and easy-to-understand data, this game has adopted UGUI. The UI module primarily includes interfaces such as character selection, game start, main interface, character death, and game restart. A series of event listeners are implemented for the UI to enable real-time updates of game data. Additionally, UI management objects are used to control the hiding and displaying of UI elements.

## 3.4 Game Implementation

Since the image information obtained in Unity3D is not in standard matrix form, the System.Drawing library must be introduced to convert image-related information into matrix representation. After matrix conversion, the image information is in RGB format. To facilitate edge information calculation, the RGB image information is converted to grayscale image information during implementation. The Laplacian operator is then used to extract edge information from the image data. Using a two-dimensional edge information matrix, the difference in image information between consecutive frames is calculated based on the characteristics of multi-frame video to obtain the difference in information of moving objects. According to the rules of neck movement, the displacement difference of the neck moving up, down, left, and right is obtained. Figure 5 shows the basic implementation process of the game.

The implementation of the game primarily includes game interface management and game processing logic management. The game interface includes the login interface, start interface, score interface, and exit game interface. When a player enters the game, they first access the game through the login interface. If the login username and password are correct, they enter the start interface, where players can select different avatars as their characters. Once the player enters the game scene interface, the game logic module takes control, which includes image processing information to control the character's movement (the up, down, left, and right movements of the neck control the character's up, down, left, and right movements). If the character encounters obstacles or hunters, their health will decrease, and their position will change. Through character movement and item collection, the character's score is altered. If the character's health drops to zero, the game returns to the restart interface, where the player can choose to exit the game or return to the login interface.

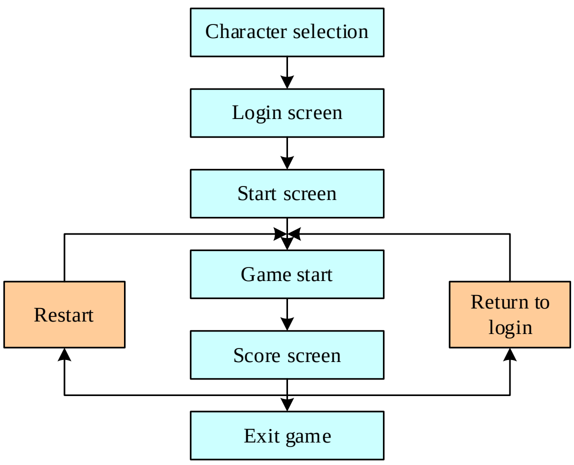


*Figure 5: The basic process of the game*

# 4 Analysis of players' cognitive experiences in immersive game design using AR technology

## 4.1 The Impact of Immersive Game Interaction on Players' Cognitive and Emotional Experiences

### 4.1.1 Research Methods

(1) Experimental Design

The experiment employed a 2 (different game types: space shooting and ground shooting) × 2 (different input methods: keyboard/mouse and controller) × 2 (different target tracking methods: head-mounted display positioning and hand positioning) within-subjects experimental design.

(2) Participants

The experiment included 10 participants aged 18–30 years (M = 22.36, SD = 3.48), including 5 females and 5 males. All participants were right-handed, had normal hearing, and had normal or corrected vision. Participants with prior AR game experience were prioritized for the game task.

(3) Experimental Instruments

1) EEG Acquisition Equipment

The experiment utilized the SynAmps2 amplifier developed by Neuro Scan. The SynAmps2 features exceptional noise-reduction capabilities, ensuring precise sampling. The SynAmps2 consists of a 70-channel amplifier, comprising 63 unipolar, 4 bipolar, and 2 high-potential input channels, with each channel's AD conversion rate precise to 24 bits.

2) HTC VIVE

The HTC VIVE consists of a head-mounted display, a positioning system that simultaneously tracks the display and controllers within a room, and two handheld controllers. The headset features an OLED screen with a combined resolution of 2160×1200 for both eyes, a 2K resolution, and a latency rate of 60ms or less.

3) Xbox Game Controller

The Xbox game controller can be connected wirelessly via Bluetooth to Windows 10 PCs, tablets, and Android devices. The anti-slip grip and surface material of the controller help players maintain a comfortable grip while easily controlling the game.

4) Lenovo Desktop Computers

Two Lenovo desktop computers are used: one for AR game display control and one for EEG signal acquisition. To facilitate participants' use of the keyboard/mouse as input devices during the game experience, raised labels are affixed to the W key (control upper direction), A key (control left direction), S key (control lower direction), and D key (control right direction) on the keyboard, enabling users to control the game more easily.

### 4.1.2 Descriptive statistics of player emotion values

The descriptive statistics of emotional values for different game types, input methods, and target tracking methods are shown in Figure 6. The immersive game designed using AR technology is a space shooting game, while the ground shooting game is a traditional game type. In the immersive games designed in this study, the emotional values are ranked as follows: interstellar keyboard manual AR games > interstellar keyboard headset-based AR games > interstellar controller headset-based AR games > interstellar controller manual AR games > ground keyboard manual games > ground controller headset-based games > ground controller manual headset-based games > ground keyboard headset-based games. The average emotional values are 0.09659, 0.06599, 0.02503, 0.02319, 0.000543, -0.00821, -0.00995, and -0.01148, respectively.

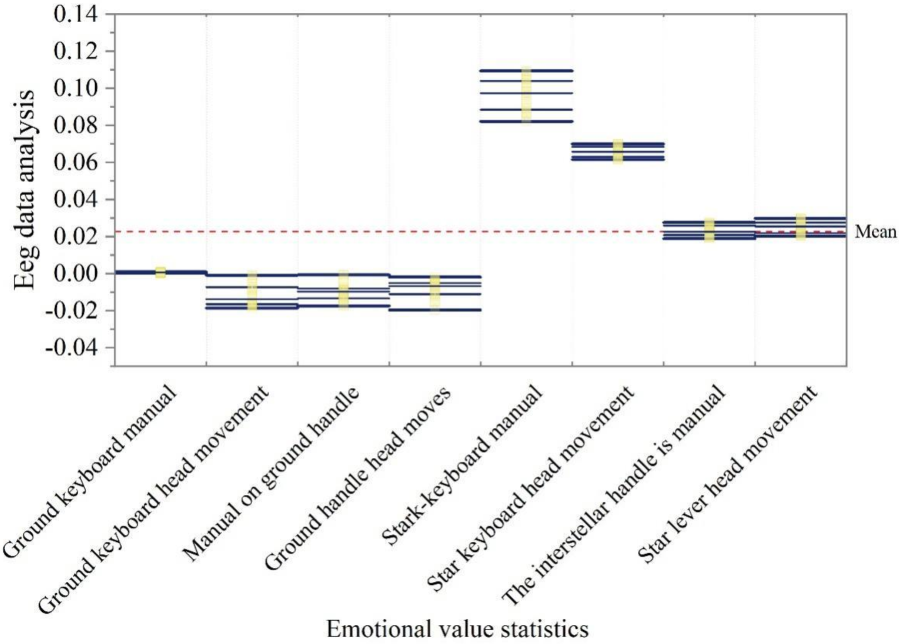


*Figure 6: Emotion values for different game types, input methods, and target tracking methods*

A 2x2x2 repeated measures analysis of variance (ANOVA) was conducted on emotional values for game type, input method, and target tracking method. Table 1 presents the results of the emotional value ANOVA. The main effect of AR shooting game type was significant, with $F=37.4895$ and $p=0.0000<0.001$, indicating that different game types influence participants' emotional experiences during gameplay. Interstellar shooting games elicited higher emotional responses than ground-based shooting games. The main effect of input method was not significant, $F=2.6485$, $p>0.05$. The main effect of tracking method was not significant, $F=0.1369$, $p>0.05$. The interaction between shooting game type and input method was not significant, $F = 2.0485$, $p > 0.05$. The interaction effect between shooting game type and target tracking method was not significant, $F = 0.2645$, $p > 0.05$. The interaction effect between target tracking method and input method was not significant, $F = 0.1966$, $p > 0.05$. The triple interaction effect between shooting game type, input method, and target tracking method was not significant, $p > 0.05$.

*Table 1: Analysis of emotional value variance*

| / | DF | MS | F | P |
|---|---|---|---|---|
| Shooting game type | 1 | 0.1248 | 37.4895 | 0.0000 |
| Input method | 1 | 0.0296 | 2.6485 | 0.1248 |
| Target tracking method | 1 | 0.0016 | 0.1369 | 0.7485 |
| Shooting game type * Input method | 1 | 0.0269 | 2.0485 | 0.1633 |
| Shooting game type * Target tracking method | 1 | 0.0026 | 0.2645 | 0.6548 |
| Input method * Target tracking method | 1 | 0.0028 | 0.1966 | 0.6688 |
| Shooting game type * Input method * Target tracking method | 1 | / | 0 | 0.9948 |

## 4.2 Cognitive Engagement and AR Technology

### 4.2.1 Cognitive Engagement Score

According to the multi-resource theory, VACP evaluates resource utilization during task execution through four channels: visual, auditory, analytical judgment, and control. This is used to measure the cognitive engagement level of participants. Figure 7 shows the cognitive engagement scores. Among the four scenarios, the auditory channel score was highest in the hit state scenario (Scenario 3) at 8.3554. In the defense state scenario (Scenario 2) and the visual channel, as well as the "healing" state scenario (Scenario 4), the auditory channel scores were highest at 8.012 and 8.0771, respectively. In the "healing" state scenario, the lowest analytical judgment score was 3.7158, followed by the control score of 4.6168 in the same scenario. Overall, the visual and auditory channels had higher resource usage, while control movement and analytical judgment had lower scores than visual and auditory channels, indicating that analytical judgment and control had lower resource usage.

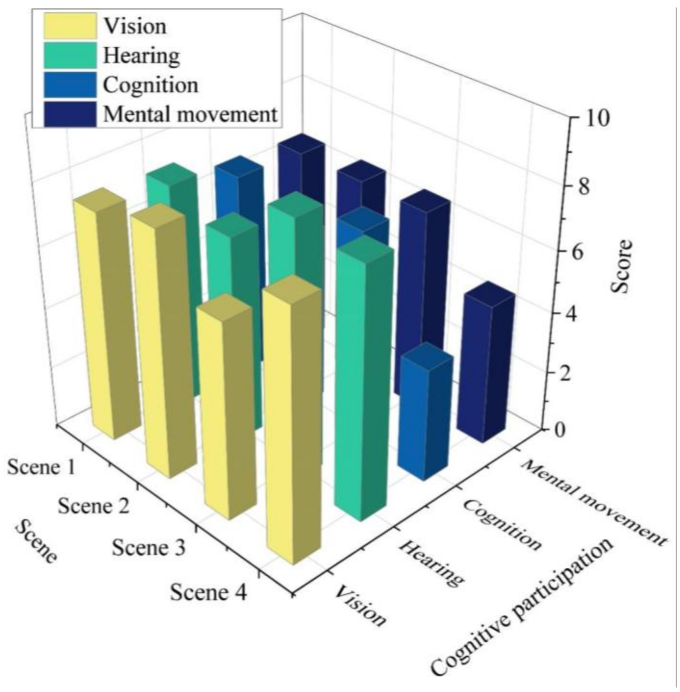


*Figure 7: Cognitive participation score*

Figure 8 shows the average scores for four different state scenarios. Different game scenarios have distinct emotional states and levels of engagement. In the attack state scenario (Scenario 1) and the "healing" state scenario (Scenario 4), the three AR technologies have comparable proportions. In the attack state scenario, all three emotional scores are above 6, with Scenario 1's positive, negative, neutral emotional scores in Scenario 1 are 7.4856, 7.7685, and 7.0488, respectively, while in Scenario 4, the scores for these three emotions are 6.0485, 7.8486, and 8.0854, respectively. In Scenario 4, the neutral emotional score is the highest. In the defensive state scenario (Scene 2) and the hit state scenario (Scene 3), there is no neutral emotion involvement, and participants exhibit a higher proportion and score of negative emotions. In all four scenarios, negative emotions account for over 30% of the total, particularly in the defensive and hit state scenarios, where they exceed 50% and have higher scores, making them the primary emotional expressions. The reasons for the emergence of negative emotions may include: the thresholds for negative emotions such as fear and anxiety are relatively low, and the cost of triggering such emotions is relatively low. The high difficulty of the game causes psychological stress, leading to the emergence of negative emotions such as anxiety and fear. Although participants experienced negative emotions, this did not affect their ability to complete the experiment; on the contrary, it enabled them to participate more actively in the game.

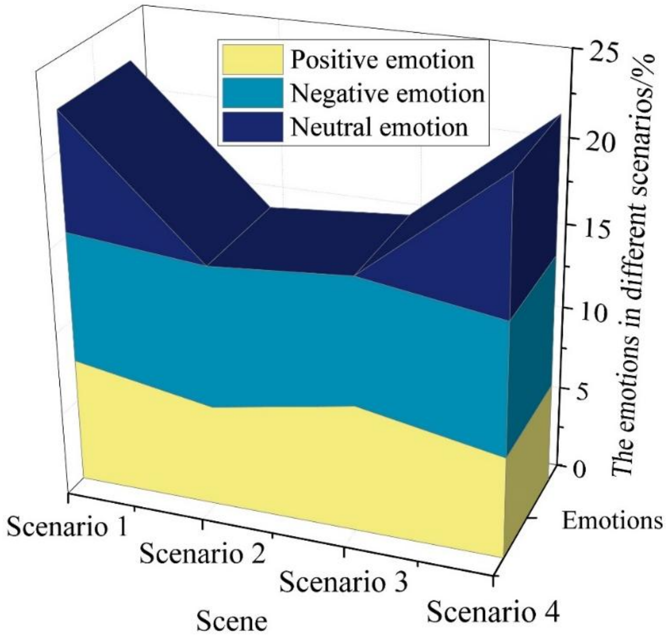


*Figure 8: The average score of the four status scenarios*

### 4.2.2 Immersion Analysis

Cognitive challenges refer to challenges to a user's memory, observation skills, and problem-solving abilities. Typically, to win the game, users need to possess strong spatial and logical reasoning abilities, decision-making skills, and planning capabilities to meet the requirements of cognitive challenges. Emotional challenges refer to the feelings users experience when facing realistic game scenarios, playing distinct game characters, and engaging in compelling stories. Emotional challenges in games reflect the interest and appeal of the game's plot or story. In this game, cognitive challenges primarily involve interpreting and predicting opponents' behavior and making decisions in an instant, while emotional challenges pertain to participants' emotional experiences of the story and scenarios throughout the game. Figure 9 shows the scores for cognitive and emotional challenges, with average scores of 7.761 and 8.0441, respectively. For the 10 participants, this game presents significant cognitive and emotional challenges. Additionally, while the participants' perceptions of cognitive and emotional challenges varied, most participants reported higher emotional challenge scores than cognitive challenge scores.

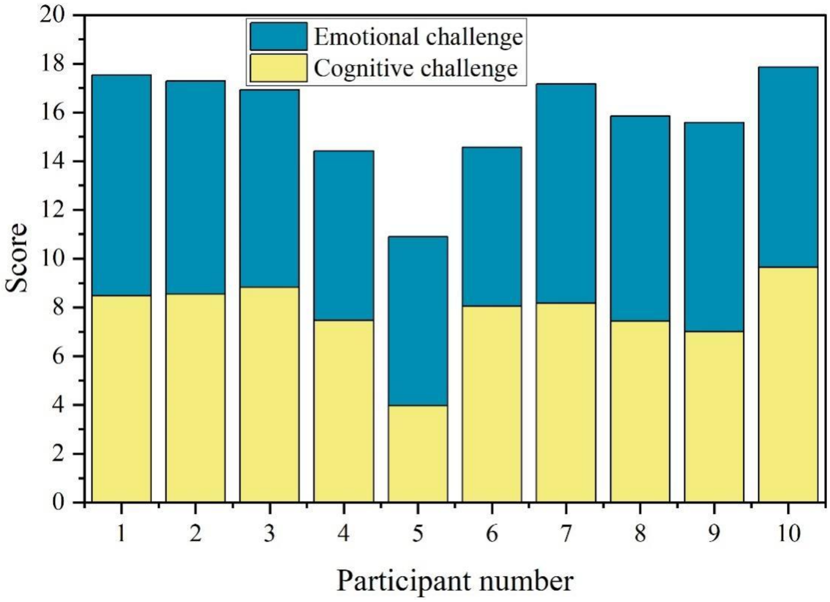


*Figure 9: Cognitive challenges and emotional challenges*

In an AR environment, physical presence primarily refers to the degree to which users experience an intuitive sense of presence in the virtual environment, as well as their ability to control, actively search for, and manipulate objects. In this dimension, users' attention is focused on the environment of control and navigation. Figure 10 shows the scores for the three dimensions of immersion. 60% of participants scored above 6.5 for physical presence, indicating that in this game, users' intuitive sensations and active behaviors in the virtual environment are relatively positive. Most participants were able to perform control operations and focus their attention on the game environment. Social presence refers to users' sense of the general presence of people in the virtual environment, the perceived credibility of their virtual avatars, and the degree of interaction with people in the virtual environment. 30% of participants scored 6 or higher on social presence, indicating that in the current AR game environment, participants' sense of social presence was the strongest. This demonstrates that the virtual “opponents” have a tangible presence for participants and can engage in active, interactive experiences with them. Self-presence refers to the degree to which users experience a certain emotional response to their intuition, real body, and virtual avatar as a unified entity, as well as the events occurring on their virtual avatar. For 80% of participants, self-presence scores exceeded 5, indicating a relatively good self-presence experience. This suggests that most participants have a positive emotional response to their intuition and virtual avatar within the virtual environment.

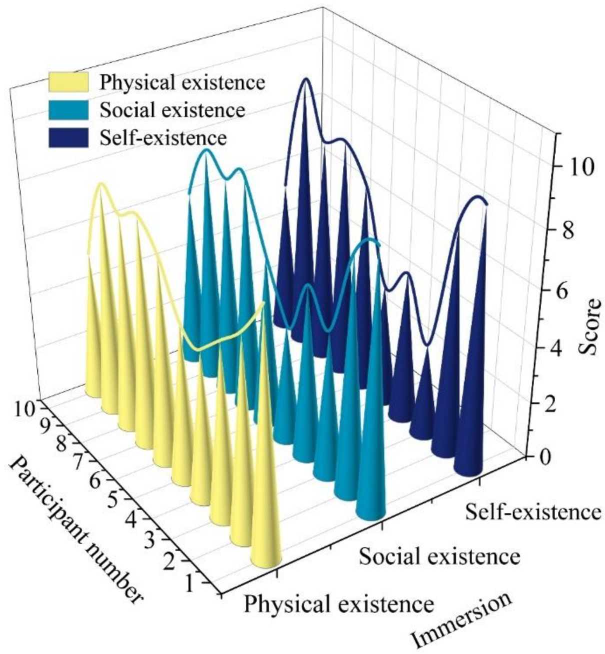


*Figure 10: Immersion score*

### 4.2.3 Correlation Analysis between AR Technology, Immersion, and Cognition

Studying immersion is an important way to test AR devices, so researching the relationship between immersion and cognition and emotion is of great significance for improving user experience and designing AR game products. Using IBM SPSS 24 to perform correlation analysis on the questionnaire data, Table 2 shows the correlation between cognitive engagement, AR technology, and immersion, while Table 3 shows the correlation between cognitive challenge, AR challenge, and immersion.

The Pearson correlation coefficient between immersion and AR technology is 0.7485, with a significance coefficient of 0.0085, which is less than 0.01, indicating a significant strong correlation. However, the significance coefficient between cognitive engagement and immersion is 0.1645, and the significance coefficient between cognitive engagement and AR technology is 0.3548. Therefore, there is no significant correlation between cognitive engagement, AR technology, and immersion. The data indicates that immersion and AR technology influence each other; higher immersion levels increase the degree of AR technology, and improvements in AR technology also enhance immersion.

The significance coefficient between cognitive challenge and immersion is 0.4855, and the Pearson correlation coefficient between AR challenge and immersion is 0.7188, with a significance coefficient of 0.0248. Therefore, the correlation between cognitive challenge and immersion is not significant. While the significance coefficient between AR challenges and immersion is less than 0.05, indicating a significant positive correlation. That is, fluctuations in immersion and changes in AR challenges influence each other; the stronger the immersion, the higher the AR challenges. AR challenges are at the core of self-presence; the better the sense of self-presence, the higher the user's experience of the game, meaning that the higher the immersion score, the stronger the user's perception of the game's challenge. This is also why users have higher scores for AR technology and AR challenges.

*Table 2: Cognitive participation in the analysis of the relationship in the technology and immersion*

| / | Project | Cognitive participation | AR technology | Immersion |
|---|---|---|---|---|
| Cognitive participation | Pearson correlation coefficient | 1 | 0.3486 | 0.4858 |
| | Sig.(double side) | - | 0.3548 | 0.1645 |
| | Number | 10 | 10 | 10 |
| AR technology | Pearson correlation coefficient | 0.3486 | 1 | 0.7485** |
| | Sig.(double side) | 0.3548 | - | 0.0085 |
| | Number | 10 | 10 | 10 |
| Immersion | Pearson correlation coefficient | 0.4858 | 0.7485** | 1 |
| | Sig.(double side) | 0.1645 | 0.0085 | - |
| | Number | 10 | 10 | 10 |

*Table 3: Cognitive challenges are associated with a sense of immersion*

| / | Project | Cognitive challenge | AR Challenge | Immersion |
|---|---|---|---|---|
| Cognitive challenge | Pearson correlation coefficient | 1 | 0.4855 | 0.2568 |
| | Sig.(double side) | - | 0.1645 | 0.4855 |
| | Number | 10 | 10 | 10 |
| AR Challenge | Pearson correlation coefficient | 0.4855 | 1 | 0.7188* |
| | Sig.(double side) | 0.1645 | - | 0.0248 |
| | Number | 10 | 10 | 10 |
| Immersion | Pearson correlation coefficient | 0.2568 | 0.7188* | 1 |
| | Sig.(double side) | 0.4855 | 0.0248 | - |
| | Number | 10 | 10 | 10 |

## 4.3 Player Experience

Table 4 shows the comparison of average player experience metrics. by calculating the average of the test data from 10 testers, with scores rounded to four decimal places, the following average comparisons were obtained: the average scores for the 11 evaluation metrics in Scenarios 1–4 were 3.8501, 2.0899, 3.1095, and 3.5531 points, respectively. Only the player experience score for Scenario 2 was below 3 points, indicating that Scenario 2 requires improvement.

*Table 4: Comparison of the user experience index mean*

| / | Indicator | Scene 1 | Scene 2 | Scene 3 | Scene 4 |
|---|---|---|---|---|---|
| Experience | Interactivity | 4.1485 | 1.1485 | 3.4152 | 4.1945 |
| | Ease Of Use | 3.5185 | 3.2485 | 2.9669 | 3.6425 |
| | Timeliness | 3.3485 | 2.9485 | 3.1485 | 3.3425 |
| | Fun | 4.4185 | 1.0485 | 3.0424 | 3.5269 |
| Observation | Easy Access | 3.8596 | 3.4258 | 3.1235 | 3.2485 |
| | Sound Literacy | 3.1485 | 1.3488 | 2.6485 | 2.8588 |
| | Aesthetics | 4.4585 | 1.4858 | 3.9636 | 4.1858 |
| | Richness | 3.3458 | 2.3458 | 2.6425 | 3.7485 |
| | Novelty | 4.6898 | 1.2485 | 3.6364 | 3.9557 |
| Abstract | Cognitive Effect | 4.0648 | 2.4699 | 3.1885 | 3.3048 |
| | Essential Cognitive Effect | 3.3499 | 2.2699 | 2.4286 | 3.0755 |

# 5 Conclusion

This paper employs two feature matching methods, SIFT and ORB algorithms, to achieve image matching within mobile games. Leveraging augmented reality technology, it provides players with an immersive gaming experience and introduces an improved DDQN algorithm to enhance the interactivity of immersive games. Based on the proposed algorithm, the game script and UI interface were designed sequentially to complete the development of the immersive game.

Among the four game scenes, Scene 3 achieved the highest score in the auditory channel, with a score of 8.3554. In Scene 4, the scores for judgment and control were the lowest, at 3.7158 and 4.6168 respectively, indicating that the analysis of judgment and control resources is relatively low.

Analyzing the game's impact on players' cognitive and emotional challenges, among the 10 participants, the average scores for cognitive challenges and emotional challenges were 7.761 and 8.0441, respectively. This indicates that the game presents significant challenges for players, and for most participants, emotional challenges outweighed cognitive challenges.

A correlation analysis was conducted between cognition, AR technology, and immersion. The Pearson correlation coefficient between immersion and AR technology was 0.7485, with a significance coefficient of 0.0085, which is less than 0.01, indicating a strong correlation between the two. Immersion and AR technology are mutually influential.

# About the Author

Jing Fu was born in Nanyang, Henan. China, in 1995. She holds a Master of New Technologies Arts from the Accademia di Belle Arti di Venezia (Venice Academy of Fine Arts), Italy. Her primary research focuses on Virtual Reality (VR)/ Augmented Reality

# References

[1] Abdullah, F., & Jamil, A. A. (2021). Reviewing Appropriate Game Design Elements for Mobile Augmented Reality Games. Sciences, 11(2), 882-891.

[2] Haahr, M., Rudenko, S., & Jakubowski, K. (2025). Alice Dalí augmented reality: evaluating a cultural outdoors game for intergenerational play. Entertainment Computing, 52, 100865.

[3] Sudarmilah, E., Ustia, N., & Bakhtiar, D. N. (2019). Learning Media based on Augmented Reality Game. Int. J. Eng. Technol, 8(1.1), 154-157.

[4] Laine, T. H. (2018). Mobile educational augmented reality games: A systematic literature review and two case studies. Computers, 7(1), 19.

[5] Marto, A., & Gonçalves, A. (2022). Augmented reality games and presence: a systematic review. Journal of Imaging, 8(4), 91.

[6] Liu, H. X. (2022). Building the “Complete Game”: an overview study of a development strategy for Geo AR mobile games. In International Conference on Human-Computer Interaction (pp. 604-622). Springer, Cham.

[7] Paavilainen, J., Korhonen, H., Alha, K., Stenros, J., Koskinen, E., & Mayra, F. (2017, May). The Pokémon GO experience: A location-based augmented reality mobile game goes mainstream. In Proceedings of the 2017 CHI conference on human factors in computing systems (pp. 2493-2498).

[8] Phipps, L., Alvarez, V., de Freitas, S., Wong, K., Baker, M., & Pettit, J. (2016). Conserv-AR: A virtual and augmented reality mobile game to enhance students' awareness of wildlife conservation in Western Australia. Mobile Learning Futures–Sustaining Quality Research and Practice in Mobile Learning, 214.

[9] Paraschivoiu, I., Buchner, J., Praxmarer, R., & Layer-Wagner, T. (2021, October). Escape the fake: Development and evaluation of an augmented reality escape room game for fighting fake news. In Extended abstracts of the 2021 annual symposium on computer-human interaction in play (pp. 320-325).

[10] Lin, Y. C., & Hou, H. T. (2024). The evaluation of a scaffolding-based augmented reality educational board game with competition-oriented and collaboration-oriented mechanisms: Differences analysis of learning effectiveness, motivation, flow, and anxiety. Interactive Learning Environments, 32(2), 502-521.

[11] Li, J., Van der Spek, E. D., Yu, X., Hu, J., & Feijs, L. (2020, June). Exploring an augmented reality social learning game for elementary school students. In Proceedings of the interaction design and children conference (pp. 508-518).

[12] Hou, H. T., Fang, Y. S., & Tang, J. T. (2023). Designing an alternate reality board game with augmented reality and multi-dimensional scaffolding for promoting spatial and logical ability. Interactive Learning Environments, 31(7), 4346-4366.

[13] Stalheim, O. R., & Somby, H. M. (2024). An embodied perspective on an augmented reality game in school: pupil's bodily experience toward learning. Smart Learning Environments, 11(1), 24.

[14] Zhang, B. (2017). Design of mobile augmented reality game based on image recognition. EURASIP Journal on Image and Video Processing, 2017(1), 90.

[15] Zuo, T., Birk, M. V., Van der Spek, E. D., & Hu, J. (2022). The mediating effect of fantasy on engagement in an AR game for learning. Entertainment Computing, 42, 100480.

[16] An, S., Bae, J., & Kim, S. K. (2018). Game Design of Augmented Reality RPG using Artificial Intelligence. Journal of The Korea Society of Computer and Information, 23(9), 15-20.

[17] Shafana, A. R. F., & Silpasuwanchai, C. (2024). Investigating the role of gesture modalities and screen size in an AR 3D game. Multimedia Tools and Applications, 83(6), 18169-18184.

[18] Adam L. Kaczmarek. (2025). A Method for Adapting Stereo Matching Algorithms to Real Environments. Applied Sciences,15(7),4070-4070.

[19] Ping Ma,Hao Yuan,Yiyang Chen,Hongtian Chen,Guirong Weng & Yuan Liu. (2024). A

Laplace operator-based active contour model with improved image edge detection performance. Digital Signal Processing,151,104550-.

[20] Jia Liu,Renjie Zhang,Isidro Butaslac,Taishi Sawabe,Yuichiro Fujimoto,Masayuki Kanbara & Hirokazu Kato. (2025). EverywhereAR: A Visual Authoring System for Creating Adaptive AR Game Scenes. IEEE transactions on visualization and computer graphics,PP,