

RESEARCH

Open Access



Design of mobile augmented reality game based on image recognition

Boping Zhang

Abstract

The work studied a complete set of development system for mobile augmented reality game based on combination of AR technology and RTS game. An effective image recognition strategy was proposed through SIFT feature matching algorithm. Integrated with cloud image recognition module, the response speed of image recognition module was improved by eliminating error matching point. Through optimization of the scheme and existing technologies, the improved image recognition algorithm was applied to the game system, realizing RTS game system on mobile intelligent terminal. The experimental results showed that the proposed method meets the requirement of augmented reality system in terms of efficiency and accuracy. In addition, the demand of users for information expansion can be satisfied by the method to some extent, which is more applicable than traditional augmented reality.

Keywords: Mobile augmented reality, Image recognition, SIFT feature recognition, RTS game

1 Introduction

Augmented reality (AR) is a three-dimensional scene where virtual objects are superimposed on real scene. In such a scene, virtual objects can be quickly generated, manipulated, and rotated to enhance users' understanding of the real environment [1, 2]. As an extension of simulated real technology, AR integrates emerging technologies of computer graphics, computer vision, image processing, sensor technology, human-computer interaction, and photoelectric display. The capabilities of intellectualization and information processing for mobile terminals are gradually enhanced with the development of wireless mobile network technology and increasing bandwidth. The users frequently access the Internet through mobile intelligent terminal. Multi-sensor equipment on mobile phone has provided hardware foundation for application of AR. In terms of software support, companies such as Microsoft and Qualcomm continually introduce various kinds of SDK with intellectual properties through innovative research and development. Such hardware and software development lays a guarantee for hierarchical and personalized service of mobile information <https://tieba.baidu.com/p/4143666840>.

Mobile augmented reality is becoming a hotspot in the field of game development. In 1997, Feiner et al. [3] developed MARS, the world's first mobile augmented reality system, which is mainly used in navigation technology. In 2000, Tomas et al. [4] released AR-Quake game, an extension of the Quake game on the PC. It is very popular at that time, allowing games to be played indoors and outdoors. Cheek et al. [5] released Human-Pacman in 2003, which is a mobile interactive entertainment system equipped with GPS and inertial sensors for positioning and visual sensing, as well as a touch-controlled human-computer interface. Schrier of the Massachusetts Institute of Technology designed a mobile augmented reality educational game called "relive the war of independence" [6]. Wherein, the playing scene is settled in Lexington, the starting place of American revolutionary war. Navigated by PAD with GPS, the participants explore Lexington battle-related public green area and other buildings. Besides, PDA overlays to display the virtual history characters and artifacts, as well as visual audio materials when they arrive at the target location.

Ingress is a mobile augmented reality game released by Google in 2012 [7]. In this game, a group of European

Correspondence: bopingzhang@yeah.net
School of Information Engineer, Xuchang University, Xuchang, Henan
461000, China

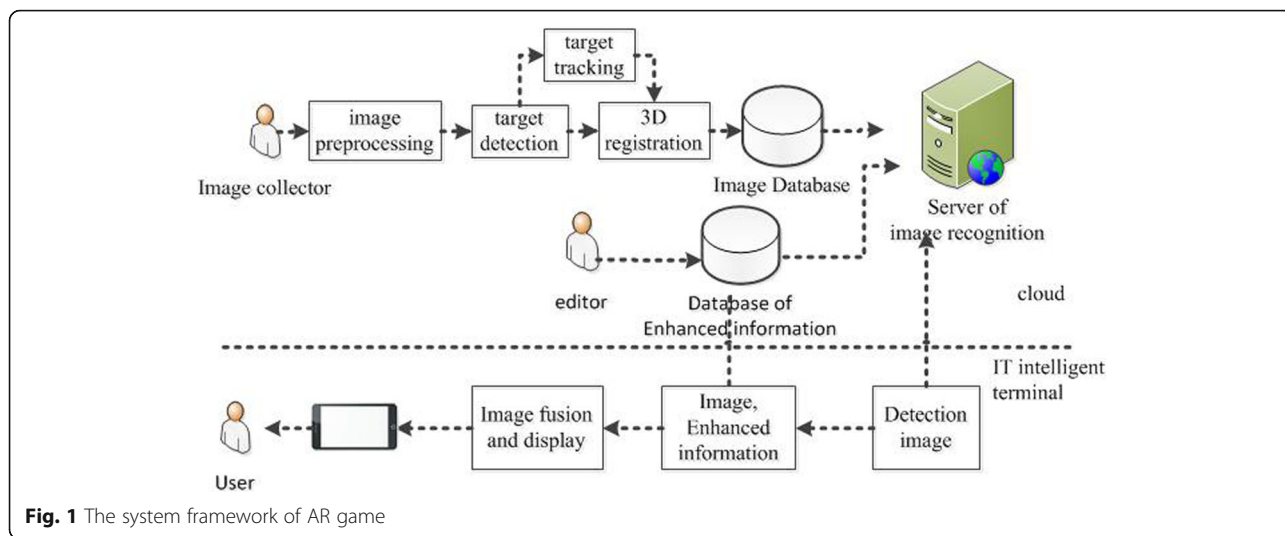


Fig. 1 The system framework of AR game

scientists finds some mysterious energy with unknown sources and usage, and some researchers think that the energy should be controlled or it will enslave the human beings after being affected mentally. The camp is divided into two groups named “Enlightened” and “Resistance”—the former trying to accept the energy as a gift to humanity; the latter striving to resist and protect the rest of our resources and wealth. The game players, called “Agent” in the game, fight each other to control the strongholds such as landmarks or sculptures in the real world.

Pokemon Go, an augmented reality (AR) RPG mobile game in pet nurturance <https://www.nintendo.com/>, jointly developed by Nintendo, Pokemon Company, and

Niantic Labs Company of Google, is welcomed globally. It has won five certificates by Guinness World Record in August 2016. However, Pokemon Go has not been released in mainland China. It indicates that AR game has not grown into a mature industry at present, where both opportunities and challenges exist.

In the work, AR technology and RTS game (Real-Time Strategy Game) were combined to study a complete development system of mobile augmented reality game. A mobile AR game named LastStandStan was developed to improve the speed of instant image recognition for clients. Proposing an improved SIFT feature matching algorithm integrated with cloud image recognition module, the response speed of image

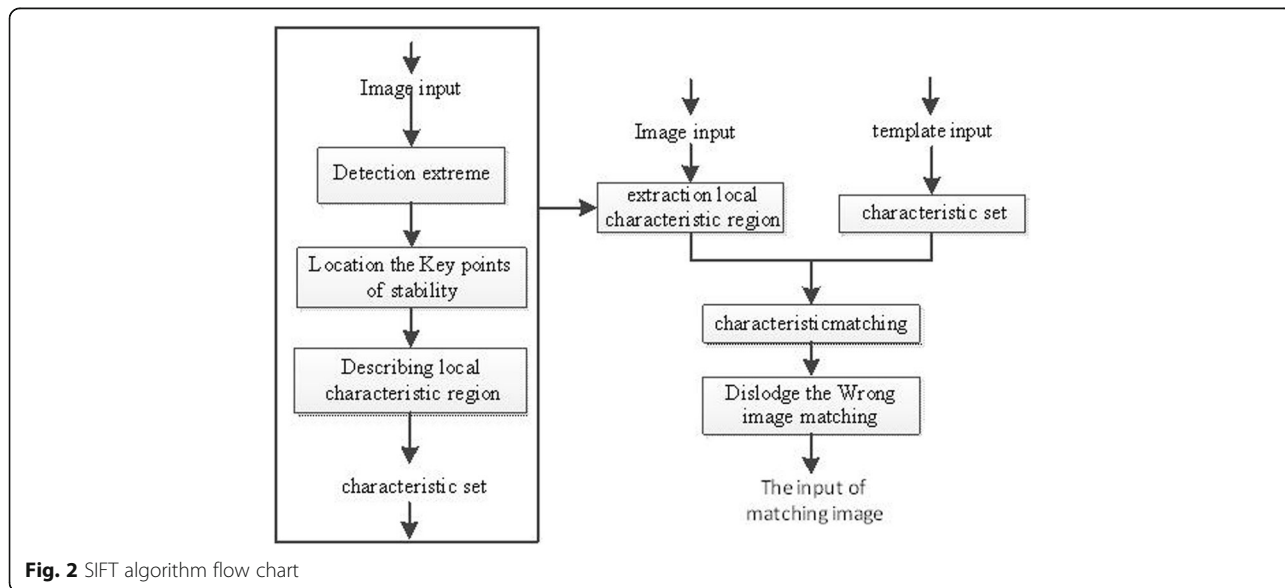
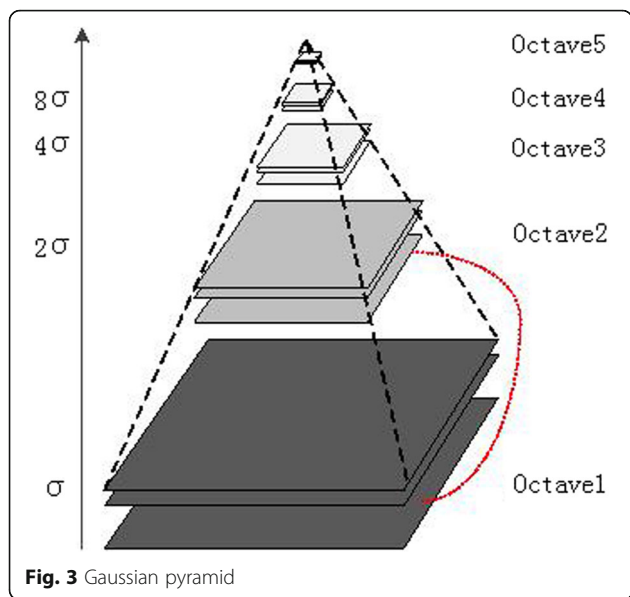


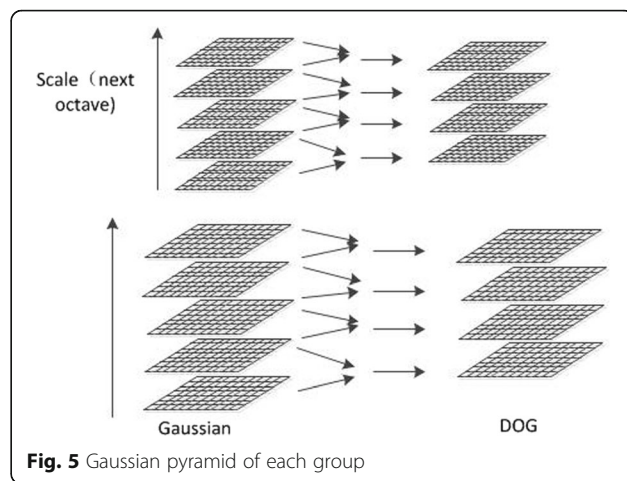
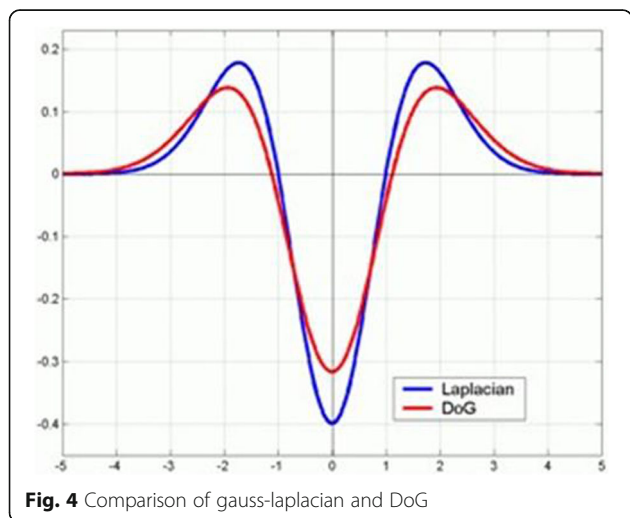
Fig. 2 SIFT algorithm flow chart



recognition module was improved by eliminating error matching point. Through optimization of the scheme and existing technologies, we applied the improved image recognition algorithm to the game system, realizing RTS game system on mobile intelligent terminal. The experimental results prove that the proposed method meets the requirement of augmented reality system in terms of efficiency and accuracy. In addition, the demand of users for information expansion can be satisfied by the method to some extent, which is more applicable than traditional augmented reality.

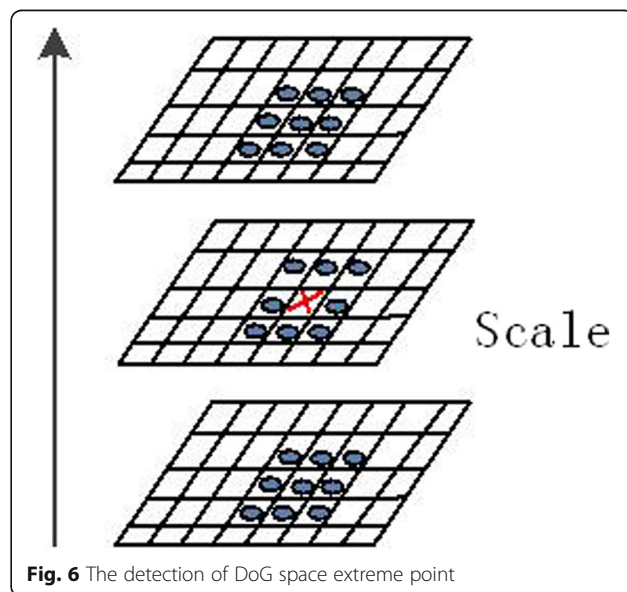
The work included the following three aspects:

1. To implement RTS game system on mobile intelligent terminal, the system architecture based



on C/S was designed, divided into the cloud subsystem and the mobile terminal subsystem.

2. In order to improve the matching speed of image, the optimal value of SIFT algorithm was found by searching on the traditional SIFT (scale-invariant feature transform) algorithm through lots of experiments, allowing user to set the optimal parameters manually. Under the environment of Easy AR, the improved SIFT algorithm is exported as a package running on the cloud server in case of designing games, with its main function of image recognition by SIFT algorithm to improve the matching speed of the random images.
3. In the work, a game named *LastStandStan* has been developed as an application of the improved SIFT algorithm, which can run on Android phones currently.



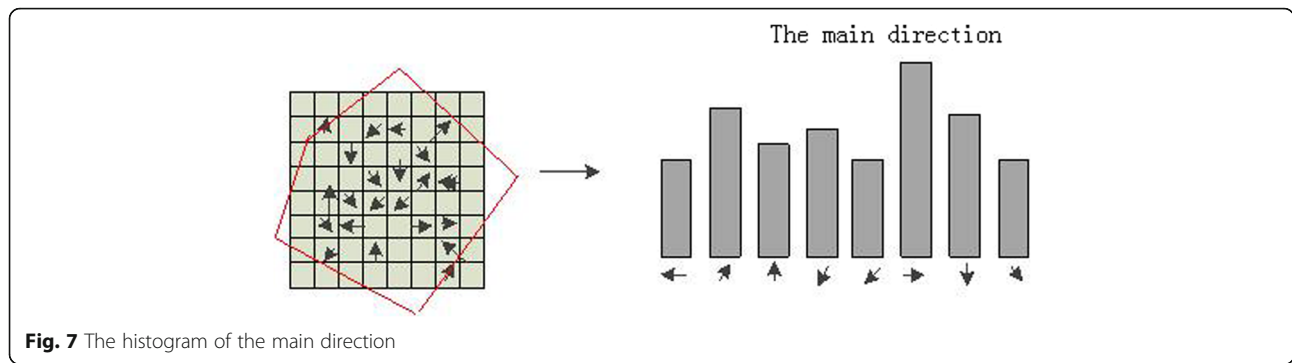


Fig. 7 The histogram of the main direction

2 Overall structure of game system

At present, the commonly used image recognition methods in the augmented reality games are mainly based on special objects and image natural feature points [8]. Method based on special objects has great limitation in applications for it requires additional special markers on target image, affecting user experience to some extent. Moreover, it is difficult to identify the target image in the presence of partial shade on special markers. The method based on image natural feature points does not need additional markers; thus, it is more often to be used in the field of augmented reality with its concise and flexible way of usage. There are mainly two ways, the method based on partial image characteristics/machine learning [9]. Traditional SIFT algorithm based on partial image characteristics has high matching precision and a better robustness for image reverse, illumination, and perspective change. Meanwhile, it produces large amount of calculation data stored in the mobile terminal, causing a larger burden due to the limitation of hardware equipment, such as mobile processor and memory. It fails to meet the requirement of real-time application in augmented reality system [10].

Table 1 Artificial intelligence

Name	Category	Props description
Robot	Robot	Providing hints of the next level of action when switching scenes
Steed knight	Robot	Searching for extraterrestrial life (starting wars), with service life limit
Defenses device	Device	Attacking surrounding enemies at regular interval
Energy collector	Device	Acquiring a certain amount of energy every fixed time
Alien life	Monster	Continually moving toward the base; blowing themselves when touching defense equipment. The attack of each enemy on the base will reduce a certain degree of durability.

In the work, a C/S architecture-adopted RTS system is built based on cloud image recognition. The STIF classifier data can be stored in the server by C/S architecture technology, thus effectively reducing the memory load in mobile terminal and improving the response speed in

Table 2 Game numerical table

Name	Basic values	Dynamic change
Energy	150	Energy collectors collect 100 points per minute
Steed knight	1	With 6 times of service life, it requires 1000 points of energy and 90 points of metal for regeneration after the service life is used up.
Artificial intelligence	10%	Increase per 1% consumes 100 points of energy and 20 points of metal.
Mineral	0	Each victory obtains 1–3 points.
Metal	50	Each victory obtains 30–50 points.
Durability	80%	Increase per 1% will consume 100 points of energy and 20 points of metal.
Life research	0%	Being less than or equal to 99% is equivalent to artificial intelligence, while above 99% will trigger the final story.
Defense device A	1	Maximum blood volume is 40, launching a detection of a range of 13 every 2 s
Defense device B	1	Maximum blood volume is 50, launching a detection of a range of 14 every 1.8 s
Defense device C	1	Maximum blood volume is 80, launching a detection of a range of 14 every 1.6 s
Defense device D	1	Maximum blood volume is 80, launching a detection of a range of 16 every 1.6 s
Large monster	0	It is automatically and randomly generated by the system every 3 s. Its self-explosion causes 15 damage points to the base.
Small monster	0	It is automatically and randomly generated by the system every 3 s. Its self-explosion causes 5 damage points to the base.

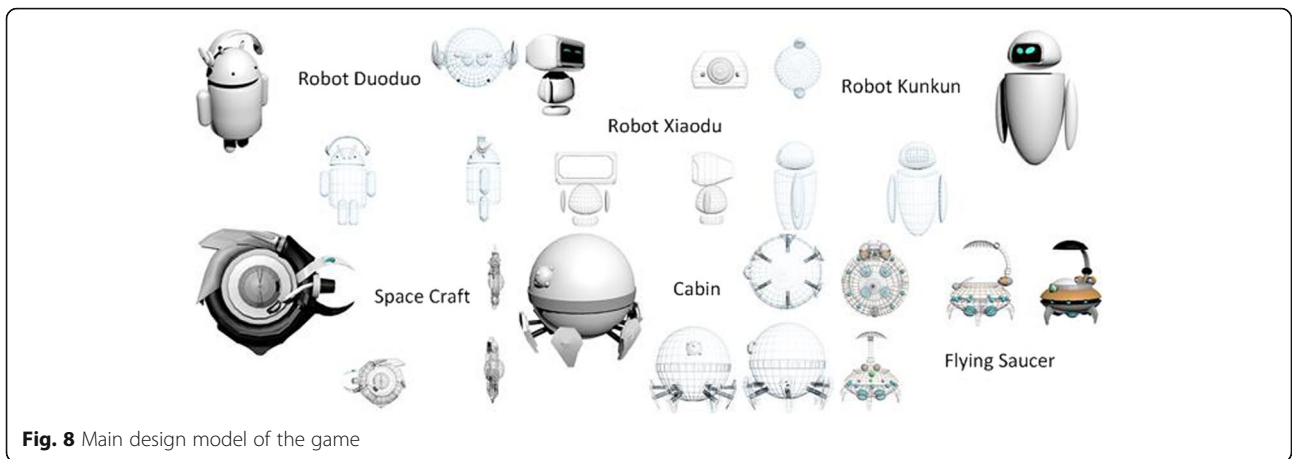


Fig. 8 Main design model of the game

system. Figure 1 shows the framework of augmented reality game system [11].

The RTS game system on the user's intelligent terminal establishes socket communication with the cloud server through wireless network such as 3G, 4G, and WIFI, realizing the data exchange between application and cloud server. After RTS game system uploads the currently identified image to the cloud, the recognition system retrieves and identifies the image from database. If the current image exists in the database, the RTS game system will download the enhanced information associated with the image to local storage via the cache mechanism. Finally, the interface

of the smart phone is designed to show the related enhancement information of the recognition scene, and the rendering engine of Unity3D is used to draw the 3D model in real time, which is divided into subsystems of cloud management and intelligent terminal management. In terms of cloud management subsystem, the first step is to acquire image through the input device. After pretreatment, the scene is checked whether a target that needs display augmentation exists. If the target exists, 3D registration is conducted. If not, the automatic tracking program is started. After tracking, 3D registration is performed and then stored in information database. Meanwhile, an augmentation

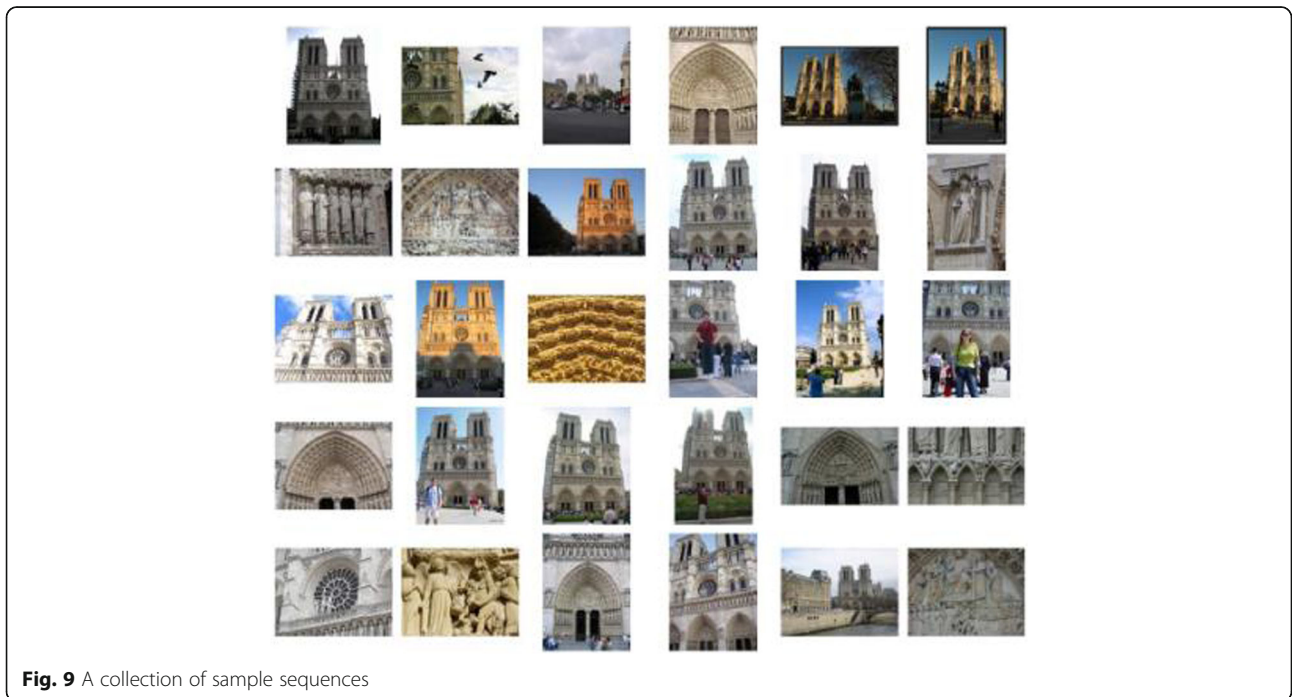


Fig. 9 A collection of sample sequences

Table 3 The matching time and quantity of Sigman

Parameter Sigman	The initial value						
Values	0.01	0.16	0.32	0.5	1	2	4
Matching quantity	45	45	45	46	45	45	45
Matching time	0.011	0.011	0.011	0.011	0.011	0.011	0.011

information database is established to store 3D model as well as text, video, and audio information. The information index is built to improve retrieval speed. As the basis of image recognition, such information can provide sample training and quick identification. The intelligent terminal management subsystem obtains augmented 3D model information of the cloud through image recognition. Virtual information is displayed on corresponding coordinate positions through image fusion.

3 Methods of SIFT image recognition

3.1 Basic principles of SIFT image recognition algorithm

SIFT algorithm has characteristics of similarity and unchanged rotation. When structuring features, multiple details need to be focused and processed to ensure faster operation and more intensive positioning [12]. Figure 2 shows the flow block diagram of SIFT algorithm.

Generation processes of local descriptive features include [13]:

1. Detecting extreme points: Gaussian differential function is used to search the image and identify potential fixed points at all scales.
2. Positioning key points: The scale of candidate position is determined through the model. The degree of stability determines the choice of key points.

3. Determining orientation of key points: Using gradient-orientation histogram, each key point is assigned with an orientation at the highest gradient, thus determining main orientations of key points.
4. Describing key points: Calculating local gradients of the image, a symbol is used to represent each gradient.

3.2 Key point detection

1. Scale-space theory

Scale space, proposed in the middle of the twentieth century, is defined after development as follows: Introducing a scale parameter into processing model, the continuously changing scale parameters are used to obtain and propose expression sequence of scale space. Then, principal contour of scale space is formulated as feature vector, extracting features such as edge [14].

Scale-space method means that the scale image becomes increasingly blurred when the scale becomes larger, thus simulating formation process of the target in the human eye retina.

2. Expression of scale space

Scale space of the image is expressed as Formula (1).

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y) \tag{1}$$

where $G(x, y, \sigma)$ is Gaussian function, $I(x, y)$ the original image $I(x, y)$, and $*$ the convolution operation.

$$Gx, y, \sigma = \frac{1}{2\pi\sigma^2} e^{-((x-d/2)^2+(y-b/2)^2)/2\sigma^2} \tag{2}$$

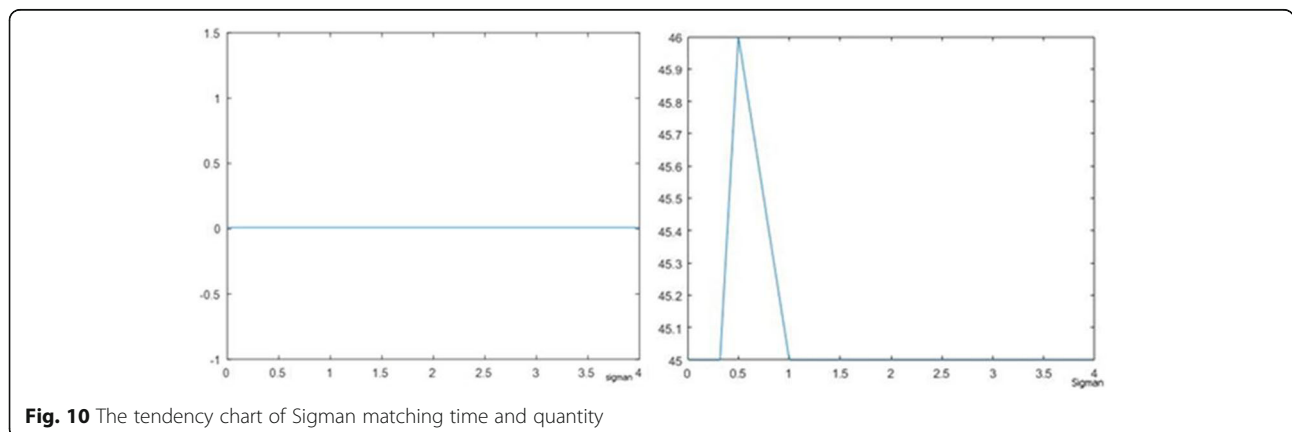


Fig. 10 The tendency chart of Sigman matching time and quantity

Table 4 The matching time and quantity of thresh

Parameter thresh	Initial value						
Values	0.01	0.02	0.03	0.04	0.1	0.3	0.5
Matching quantity	47	47	47	46	28	5	0
Matching time	0.02	0.014	0.012	0.011	0.005	0	0.001

where d and b are the dimensions of Gaussian template; (x, y) is the pixel position; σ is the scale space factor.

3. Constructing Gaussian pyramid

Operations of this process include Gaussian blur and down sampling (See Fig. 3).

The pyramid is characterized by different sizes and tower-like model with increasingly smaller sizes from bottom to the top.

The tower is realized as follows: Original image is used in the first layer, while new image obtained from down sampling in the next layer. There are n layers in each tower. The number of layers is calculated as follows.

$$n = \log_2\{\min(p, q)\} - d \quad d \in [0, \log_2\{\min(p, q)\}] \quad (3)$$

where p and q are the sizes of original image; d is the logarithmic value of minimum dimension of the image on top of the tower.

4. Gaussian difference pyramid

According to previous studies, after the scales of maximum and minimum values of Gaussian Laplacian function $\sigma^2 \nabla^2 G$ are normalized, their results and other feature extraction functions such as gradient can generate the most stable image feature. Then,

Gaussian difference function is developed, which is approximated by the scale-normalized Gaussian Laplacian function $\sigma^2 \nabla^2 G$. The relationship is described as follows.

$$\frac{\partial G}{\partial \sigma} = \sigma^2 \nabla^2 G \quad (4)$$

Difference approximately replaces differential:

$$\sigma^2 \nabla^2 G = \frac{\partial G}{\partial \sigma} \approx \frac{G(x, y, k\sigma) - G(x, y, \sigma)}{k\sigma - \sigma} \quad (5)$$

then

$$G(x, y, k\sigma) - G(x, y, \sigma) \approx (k-1)\sigma^2 \nabla^2 G \quad (6)$$

where $k-1$ is a constant.

As shown in Fig. 4, the red is DoG operator curve, while the blue is Gauss-Laplacian curve. Extreme value is detected by replacing Laplacian with DoG operator [15].

$$D(x, y, \sigma) = (G(x, y, k\sigma) - G(x, y, \sigma)) * I(x, y) = L(x, y, k\sigma) - L(x, y, \sigma) \quad (7)$$

In the calculation, Gaussian difference image is obtained by subtracting the upper and lower layers of Gaussian pyramid of each group (see Fig. 5).

5. Space extreme point detection

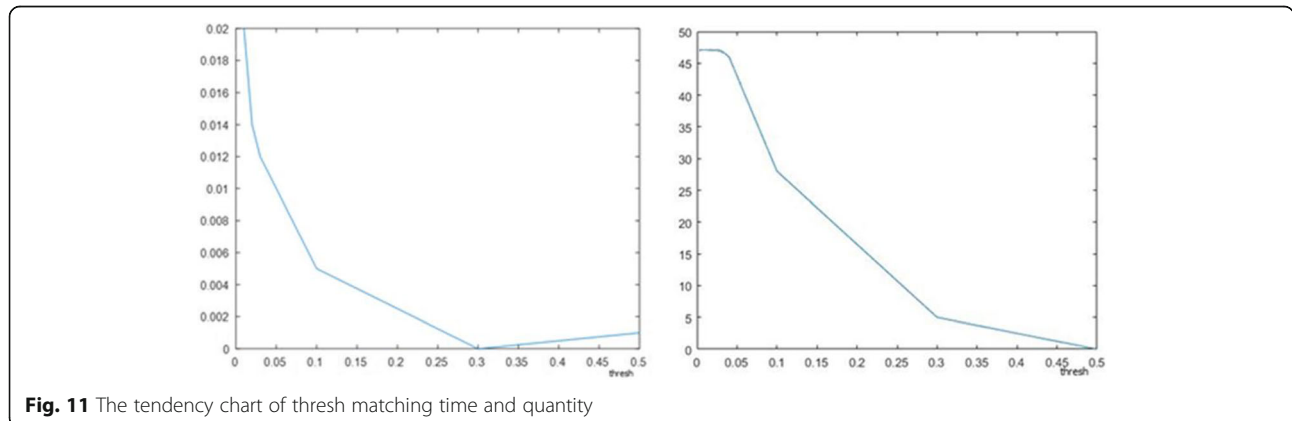


Table 5 The matching time and quantity of r

Parameter	r	Initial value					
		0.5	5	10	20	50	100
Values	0.01	0.5	5	10	20	50	100
Matching quantity	59	7	28	46	56	62	62
Matching time	0.017	0.01	0.05	0.011	0.016	0.018	0.018

Local extreme points constitute key points in the space of Gaussian difference. In search for key points, the images between two adjacent layers in the same group are compared. Then, each pixel point is compared with all the neighboring points around it to judge its size. Figure 6 shows that the red intermediate detection point is compared with corresponding 26 points in surrounding as well as up and down scale space, thus ensuring that the extreme points can be detected.

In the right image of Fig. 5, $N + 2$ layers of DoG pyramid and $N + 3$ layers of Gaussian pyramid are required if there are N extreme points in each group. Due to influence of factors such as edge response, extreme points generated in such a case are not all stable.

Key points include location, scale, and orientation. To maintain invariance of view angle and illumination, the key points need to be described by a set of vectors. The descriptors need to include key points and pixels that contribute to the key points. Meanwhile, independent characteristics of the descriptors should be ensured to improve the probability of correct matching of feature points.

Key point matching is divided as follows.

1. Gradient calculation

The modulus and orientation are determined by Formula (8).

$$m(x, y) = \sqrt{\begin{matrix} (N(x + 1, y) - N(x - 1, y))^2 \\ + (N(x, y + 1) - N(x, y - 1))^2 \\ \theta(x, y) = \text{atan2} \left(\frac{N(x, y + 1) - N(x, y - 1)}{N(x + 1, y) - N(x - 1, y)} \right) \end{matrix}} \tag{8}$$

where N represents scale space value of key points.

2. Gradient histogram statistics

The gradient and orientation of pixels in neighborhood are counted, shown in the form of a histogram. Figure 7 shows that the orientation ranges from 0 to 360°, with a bin per 10° and 36 bins in all. The peak in the field of feature points represents gradient orientation. The histogram of maximum values is the main orientation of key points [16]. Meanwhile, the histogram, with the peak 80% greater than main orientation, is selected as auxiliary orientation, thus improving match robustness.

The entire algorithm has not ended after key points are matched successfully, as large number of mismatch points occur in matching process. Generally, Ransac method is used to eliminate mismatch points in SIFT matching algorithm [17]. The core of the algorithm is continual iteration through repeated testing.

3.3 SIFT image recognition

SIFT feature looks for stable pole through scale space to generate local feature descriptors, where the stable ones have high representative and distinguishing features. However, they are generated in small number in one picture. Therefore, RTS game system in the work adopts classical SIFT matching algorithm as the original SIFT matching method, manually determining the accuracy and mismatch rate of each match, that is, describing the matching performance by accuracy

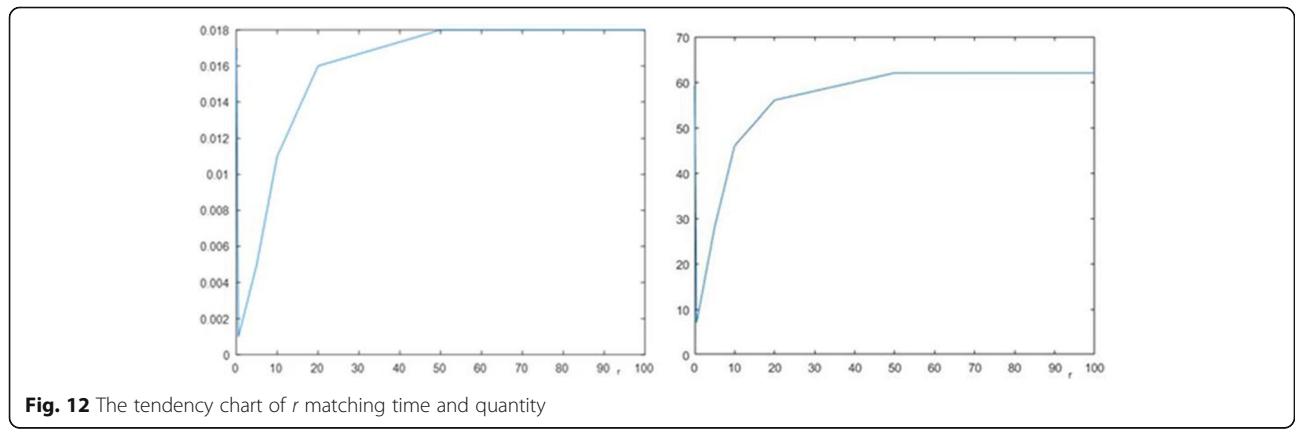


Fig. 12 The tendency chart of r matching time and quantity

Table 6 The matching time and quantity of NBP

Parameter	NBP	Invalid	Invalid	Initial value			
Values	2	X	X	4	8	16	32
Matching quantity	47	X	X	46	28	6	0
Matching time	0.004	X	X	0.011	0.019	0.02	0.002

and misrecognition rate. The accuracy is defined by the ratio of number of valid matches retained to total objective. The misrecognition rate is determined by the ratio of the result (the total number of key points minus the match points) to the total number of key points. The identification process includes image pre-processing, module on feature extraction, module on image description, training module, and identification module. A package to identify atlas is exported with the algorithm in Easy AR.

1. Pre-processing module

The pre-processing module outputs the area containing object in the image by operations, such as filtering the object containing image, searching for the contour, and establishing the minimum surrounding rectangle.

- ① Gray degree transforms the image by the formula of $L = 0.299 * R + 0.587 * G + 0.114 * B$.
- ② The median filter and Gaussian filter are used to filter the original image separately.
- ③ The image is reversed with grayscale and processes with binarization.
- ④ Searching contour is conducted to the binary image.
- ⑤ Establish a minimum surround rectangle for the obtained contour, and output the image inside the smallest rectangle.

2. Module on feature extraction

Module on feature extraction is divided into two steps. First, the key points are detected by reading the image in the grayscale pattern, then testing the key points with the output of key points assemblage. Second, features are described. The key code is as follows:

```
Sift feature detector extractor;
Mat descriptor;
extractor.compute(img, keypoints, descriptor);
```

Code in the first line declares a SIFT feature descriptor generator. The second line declares a Mat type of data used to store description. The third line directly uses the calculation method of descriptor generator to calculate the descriptor.

3. Module on image description

This module is to describe the image by spatial pyramid model. The specific process is as follows:

- Step 1: Extract the SIFT key points of the image by using the initial layer scale of 1, with the initial layer grid width of 6.
- Step 2: Extract the key points of SIFT feature of the original image.
- Step 3: Combine the key points assemblage generated in Steps 1 and 2 to form a new set of key points.
- Step 4: Describe the SIFT feature of the key points assemblage generated in Step 3, and output the image description vector.

4. Key point matching

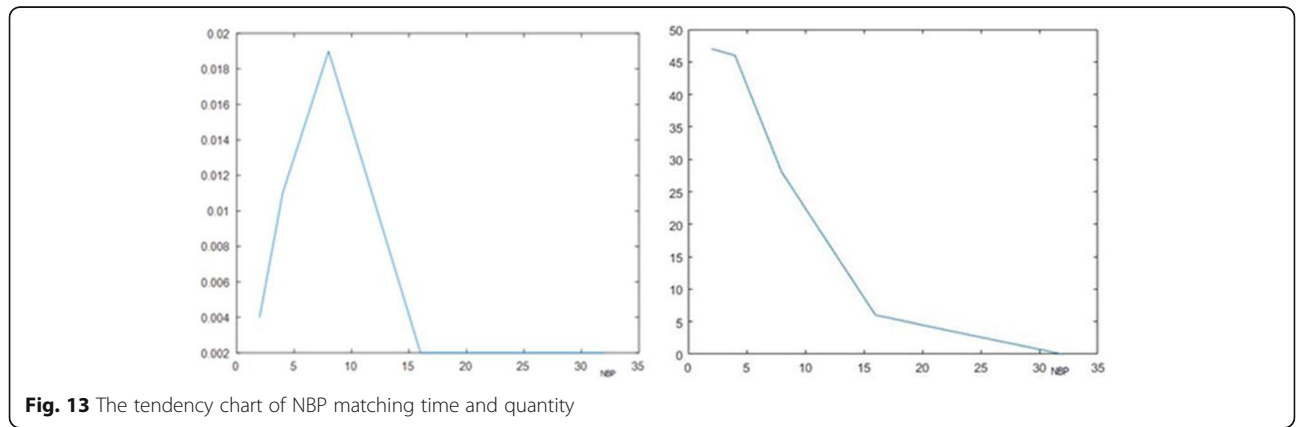


Fig. 13 The tendency chart of NBP matching time and quantity

Table 7 The matching time and quantity of NBO

Parameter	NBO Initial value						
	1	2	4	8	16	40	80
Values	1	2	4	8	16	40	80
Matching quantity	29	35	38	46	32	30	30
Matching time	0.002	0.003	0.006	0.011	0.022	0.054	0.107

For DoG function in scale space, curve fitting is conducted and the ratio of main curvature to feature value is calculated, eliminating edge feature points and feature points with low contrast. After calculating the gradient of extreme point, the adjacent region is divided based on gradient direction of feature points specified by gradient histogram. Then, the feature point descriptor is generated through integration, excluding feature points smaller than threshold value and counting the number of matching points. Setting a threshold, if the number of matching point is bigger than the threshold value, end the program and output match result; if the threshold value is bigger, the approximate component of the first layer image is used again to repeat Steps 2–6, then end the program after getting matching result.

4 System application

In order to verify the effectiveness of the improved SIFT algorithm, we developed a mobile AR game based on LastStandStan [18]. Integrating SIFT image recognition algorithm, AR technology and RTS game, the game focuses on using the proposed SIFT recognition technology to recognize pictures randomly taken by users.

The game is about a science fiction story that takes place in the future. As humans waste too much earth resources, the earth finally becomes deserted after 50,000 years due to depletion of resources. Human beings have to leave their homes and take spacecraft

drifting in space, while the space environment is not suitable for breeding the earthling. Therefore, the earth lives are completely caught in sleepy and energy-depleting spacecraft, leaving only the robots still running the instructions given by their developers. Due to the coming depletion of energy, the machine system of spacecraft starts the final program to create an unprecedented super robot. When the robot is about to wake up, its radio waves will be connected to the brain waves of players from the distant past. Players need to help the robot complete various tasks of production and combat readiness. The robot is a virtual image for player’s control of the spacecraft. Players are required to promote development of the game through ways of collection, creation, and war, ultimately completing the final purpose of life research.

4.1 Game design features

The game LastStandStan completes 3D modeling in the 3ds max 2014 environment as well as virtual interaction in Unity3D 5.0 engine. For achievement of AR function, a plug-in based on Easy AR has been developed to implement image recognition through the improved SIFT algorithm. The game output is based on screen of intelligent mobile terminal [19]. These are the following characteristics in the game design:

1. The game integrates task sequence and construction features of RTS game, blending characteristics of tower-defense game in battle scene. All operations are converted to the changes of various values.
2. A robot is selected by the player as the protagonist to provide operating instructions when the player switches game levels. Players click UI or corresponding objects

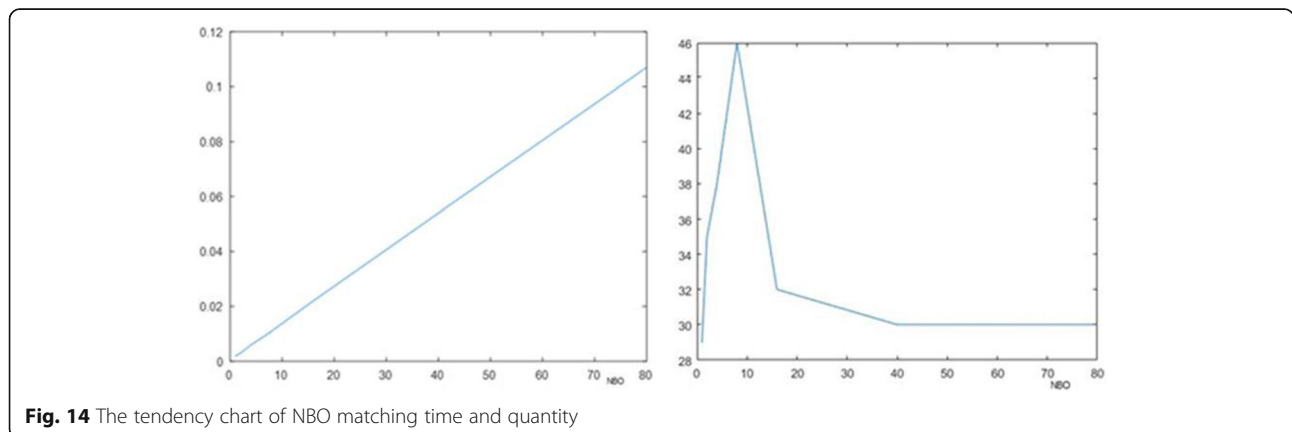


Fig. 14 The tendency chart of NBO matching time and quantity

Table 8 The matching time and quantity of magnif

Parameter	Magnif	Initial value						
Values	0.01	0.5	2	3	4	12	18	
Matching quantity	0	43	51	46	27	14	1	
Matching time	0.017	0.018	0.014	0.011	0.006	0.002	0.003	

through touch. Different values feedback will be returned to players if they click on different objectives. The first time players enter the game, they will receive a hint about whether they choose a robot or not [20–24]. When loading levels, the robot will prompt the player to operate the next step and complete daily collection of energy. Through the energy, all kinds of equipment are produced for wars, life research, and repair of spaceship.

3. Interaction is implemented on AR virtual objects. The interaction is achieved when the touching point coincides with the fixed point on virtual object. For example, in the spacecraft scene, when the player clicks the cabin in the middle of model, next-step command is actually triggered—cabin color changes the command. Then, the player can switch the level after clicking the confirmation button again.
4. Two outcomes are designed in the game. One is when the intelligence reaches 99%; the player needs to sacrifice the robot to complete life research, thus awakening the sleeping life of the earth. The other is to retain the robot when the intelligence reaches 99%. New mechanical civilization emerges after a few years.
5. Different operations are required in various cabins to promote game process. In collection cabin, energy collectors should be built to speed up energy collection. Repair cabin

can enhance durability of the spacecraft (the game failure failed at durability less than 0). The content of library cabin remains to be expanded, which allows players to view game background information. Combat cabin can be used to construct defensive equipment and create steed knight for warfare. Steed knight has six service lives, which need to be re-manufactured after the lives have been used up. The player can enter battle scene by clicking steed knight.

6. Combat scene is designed as a hexagonal terrain, generating enemies at random locations. With large and small sizes, the enemies will blow themselves when touching the defense equipment or base. Large monster has relatively higher moving speed and damage to the base. The defensive device detects surrounding enemies at regular intervals to destroy them. The durability at 0 results in game failure. The player will be rewarded with mineral, energy, and metal after all enemies have been destroyed to win battle victory.
7. Biological engineering cabin is the main task of the game, where players need to enhance the artificial intelligence through energy as well as minerals and metal obtained from wars. When artificial intelligence reaches 99%, final story of the game will be triggered to retain mechanical civilization or complete life.

4.2 Artificial intelligence

In order to enhance pleasure of the game, behavior logic is added in game design. Such logic is controlled by the system rather than the player, with feedback of numerical values [25]. When a single player fights against AI, AI is in random combinations and fight after AR recognition to players' image.

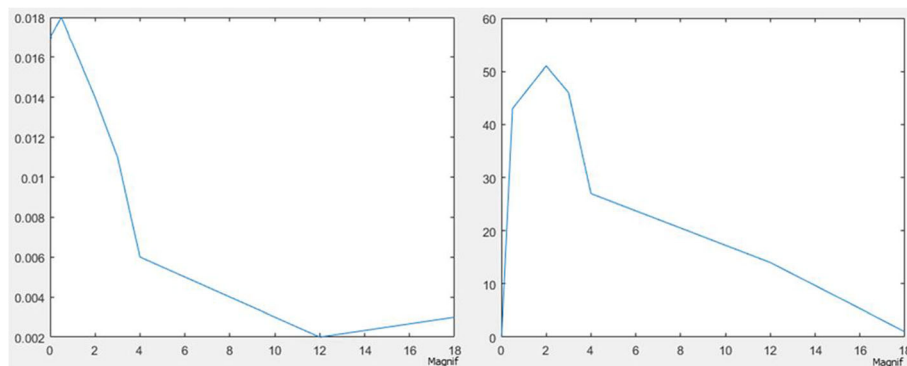


Fig. 15 The tendency chart of magnif matching time and quantity

Table 9 The matching quantity and time of Sigman

Parameter	Sigman	Initial value						
Values	0.01	0.16	0.32	0.5	1	2	4	
Matching quantity	11	11	11	11	11	11	11	
Matching time	0.002	0.002	0.005	0.002	0.002	0.002	0.002	

The characteristics of AI is that if AI takes advantage in the battle, it automatically chases players and launches attack after reaching a certain distance; if AI is in the weak side, it escapes in random directions. If they are the same branch of armed forces, two states are randomly generated—one is to actively pursue the player and attack in a certain distance, and the other is to escape in random directions. Table 1 shows specific artificial intelligence.

4.3 Numerical balance

Internal mechanism is the interaction among various values, while the values determine the time length of the game. In LastStandStan, the system needs to control basic values of energy, the damage to the base caused by monsters, as well as detection range and frequency of defense device. Basic values of the game will be balanced with revenue and spending values. Table 2 shows the detailed game balance values.

4.4 Main model design

Figure 8 shows main design model of the game.

5 Experimental test

5.1 Experiment of SIFT image matching algorithm

In order to evaluate the proposed algorithm, the experiment selected image sets containing view transformation in the data set. Figure 9 shows the selected images, with the change of viewing angles of each

group becoming larger. The proposed SIFT matching algorithm is used as the initial SIFT matching method. Matching accuracy and mismatch rate are manually determined for each match, namely describing matching performance with accuracy and error. The accuracy is defined by the ratio of the number of retained correct matches to the total number. Subtracting the number of matching points from the total number of key points, the error is calculated by the ratio of subtracting results to total number of key points. Figure 9 shows the matching result of SIFT algorithm with an image.

In order to study the influence of the values of relevant parameters in SIFT algorithm on matching results, four main steps were included as follows:

Step 1. Experiment sample was selected.

Considering workload of experimental study, only two representative samples were selected from the four samples in this part, namely illumination and head rotation samples, respectively. The influence of relevant parameters changes on light changing and angle changing images was explored.

Step 2. Relevant parameters were selected from SIFT algorithm. Six main parameters were selected from SIFT algorithm, including Sigman, r , thresh, NBP, NBO, and magnif.

Step 3. The range of parameters was studied. Regarding initial values of the six parameters in SIFT algorithm as midpoints, three valid values were taken to the left and the right, respectively (Due to the scope limit of parameters, it is not really for all that the left/right had three valid values.)

Step 4. Performance indicator parameters were determined according to the matching results generated from the change of parameter values, with parameter analysis.

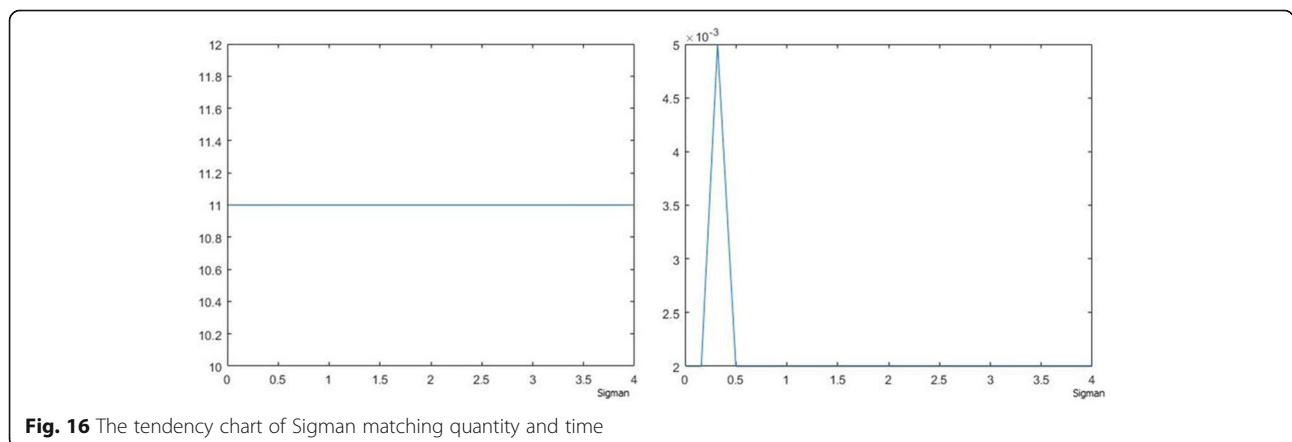


Fig. 16 The tendency chart of Sigman matching quantity and time

Table 10 The matching quantity and time of thresh

Parameter	Thresh	Initial value					
Values	0.01	0.02	0.03	0.04	0.1	0.3	0.5
Matching quantity	11	11	11	11	10	3	0
Matching time	0.007	0.002	0.002	0.002	0.002	0	0

Main work was included in Steps 3 and 4, with detailed process as follows:

Initial values of the six parameters in SIFT algorithm were as follows: $\text{Sigman} = 0.5$; $r = 10$; $\text{thresh} = 0.04$; $\text{NBP} = 4$; $\text{NBO} = 8$; and $\text{magnif} = 3$. Then, control variables were adopted: Only one parameter was changed per time for each of the two samples, while other parameters remained unchanged. In order to ensure visual display of experimental results, time data of 1000 magnifications was used in the experiment.

5.1.1 Experiment on parameter adjustment for illumination change image

1. Parameter Sigman

According to Table 3 and Fig. 10, experiment results are within the whole taking value interval after changing the parameter of Sigman. In terms of matching time index, the matching time of SIFT algorithm keeps unchanged. For matching quantity index, the matching number of initial values is 46, while that of other Sigman values remains unchanged.

Considering factors of experimental error, the change of Sigman values is almost irrelevant to matching time and quantity indexes of SIFT algorithm. Thus, Sigman is not considered as the key parameter of SIFT algorithm in improving matching rate of SIFT algorithm.

2. Parameter thresh

Based on Table 4 and Fig. 11, experiment results are within the whole taking value interval after changing the parameter of thresh. Both indexes of matching time and quantity of SIFT algorithm show a declining trend in the whole range.

Considering factors of experimental error, the change of thresh values is negatively correlated with indexes of matching time and quantity of SIFT algorithm. In other words, the matching time and quantity of SIFT algorithm gradually reduce with the increase of thresh values taken from left to right. Therefore, thresh can be regarded as a negative correlation parameter of matching performance in the test improving matching rate of SIFT algorithm.

3. Parameter r

According to Table 5 and Fig. 12, experiment results are within the whole taking value interval after changing the parameter of r. Both indexes of matching time and quantity of SIFT algorithm show a rising trend.

Considering influence of experimental error, the change of r values is positively correlated with matching time and quantity of SIFT algorithm. That is, the matching time and number of SIFT algorithm gradually decline with the increase of r values from left to right. Therefore, r can be regarded as a positive correlation parameter in the test improving matching rate of SIFT algorithm.

4. Parameter NBP

From Table 6 and Fig. 13, NBP value needs to be a positive; thus, only the value of 2 satisfies the requirement before the initial value of 4.

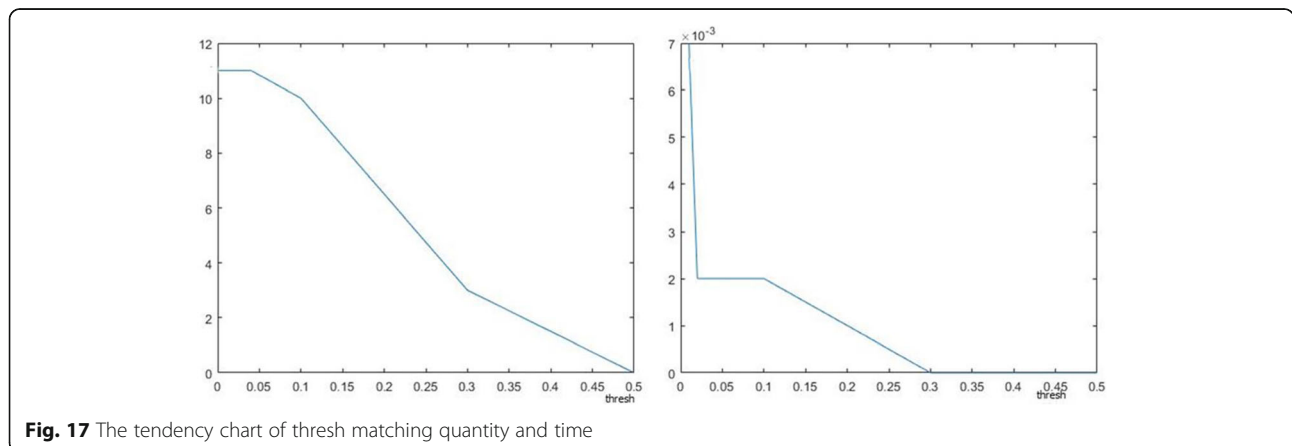


Fig. 17 The tendency chart of thresh matching quantity and time

Table 11 The matching quantity and time of r

Parameter	r	Initial value					
Values	0.01	0.5	5	10	20	50	100
Matching quantity	16	1	8	11	12	14	14
Matching time	0.003	0	0.001	0.002	0.003	0.003	0.003

Experiment results are within the whole taking value interval after changing the parameter of NBP. The matching time of SIFT algorithm firstly increases and then decreases, while the matched quantity shows an overall decreasing trend.

Considering effects of experimental error, the influence of NBP value change on matching time of SIFT algorithm firstly enhances and then decreases, with the increasing parameters of NBP values taken from left to right. There is an overall negative correlation between matching number and NBP value change. Thus, NBP is regarded as a negative-correlation parameter of matching performance in improving matching rate of SIFT algorithm.

5. Parameter NBO

Based on Table 7 and Fig. 14, experiment results are within the whole taking value interval after changing the parameter of NBO. The matching time of SIFT algorithm presents an increasing trend, while the matching quantity firstly increases and then declines.

Considering effect of experimental error, the change of NBO parameter value has a positive correlation with matching time of SIFT algorithm, with the increasing parameters of NBP values taken from left to right. For matching quantity, the correlation is firstly positive and then becomes negative on the whole. Therefore, in the test improving matching rate of

SIFT algorithm, a better match rate can be obtained when taking values within the range of 4–16 for set of NBO values.

6. Parameters magnif

Table 8 and Fig. 15 shows that experiment results are within the whole taking value interval after changing the parameter of magnif. The matching time of SIFT algorithm shows a decreasing trend, while the matching quantity firstly increases and then decreases.

Taking into account experimental error, the change of magnif values has a negative correlation with matching time of SIFT algorithm, with the increasing parameters of r values taken from left to right. For matching number, the correlation is firstly positive and then becomes negative on the whole. Therefore, when setting parameter values of magnif in the test improving matching rate of SIFT algorithm, a better match rate will be obtained within the value range of 0.5–3.

5.1.2 Experiment on parameter adjustment for angle change image

1. Parameter Sigman

Table 9 and Fig. 16 show that experiment results are within the whole taking value interval after changing the parameter of Sigman. The matching time of SIFT algorithm shows a steady trend at first, and then presents a sudden rise, finally returning to be steady after a decline. The number of matches basically remains unchanged.

When considering experimental error, the change of Sigman parameter value is positively at first and then becomes negatively correlated with matching time of

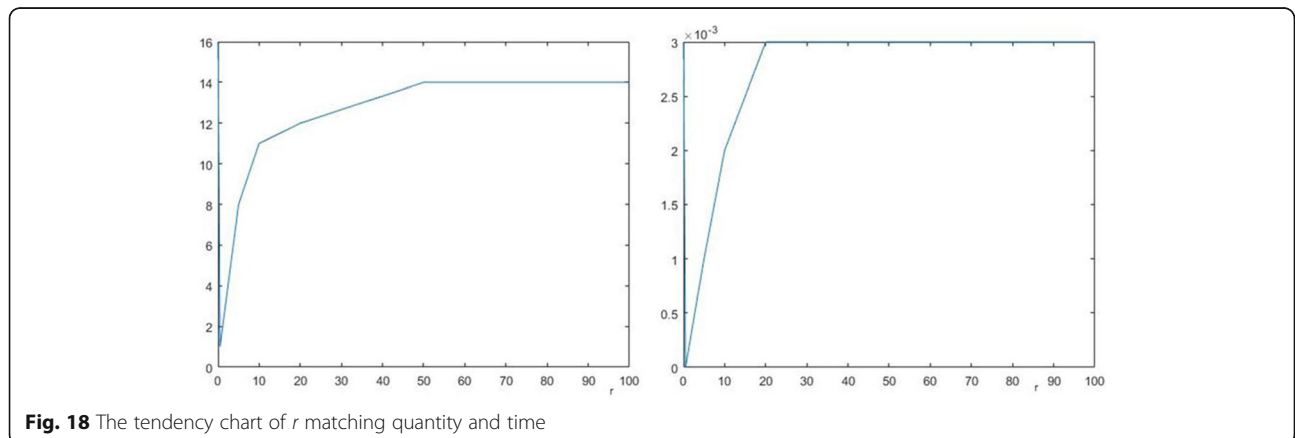


Fig. 18 The tendency chart of r matching quantity and time

Table 12 The matching time and quantity of NBP

Parameter	NBP	Invalid	Invalid	Initial value			
Values	2			4	8	16	32
Matching quantity	17	X	X	11	4	0	0
Matching time	0.001	X	X	0.002	0.005	0.006	0.001

SIFT algorithm, as the parameters of Sigman values increase from left to right. There is no correlation with the match number on the whole. Therefore, Sigman can be regarded as an irrelevant parameter in setting of Sigman parameter values during the test improving matching rate of SIFT algorithm.

2. Parameter thresh

Table 10 and Fig. 17 show that experiment results are within the whole taking value interval after changing the parameter of thresh. Both the matching time and quantity of SIFT algorithm show an overall declining trend.

Considering the influence of experimental error, the change of thresh values is negatively correlated with matching time and quantity indexes of SIFT algorithm. In other words, matching time and quantity of SIFT algorithm gradually decrease as thresh values increase from left to right. Therefore, thresh can be regarded as a negative correlation parameter of matching performance in the test improving matching rate of SIFT algorithm.

3. Parameter *r*

Table 11 and Fig. 18 show that experiment results are within the whole taking value interval after changing the parameter of *r*. Matching time SIFT

Table 13 The matching quantity and time of NBO

Parameter	NBO		Initial value				
Values	1	2	4	8	16	40	80
Matching quantity	6	8	10	11	11	12	12
Matching time	0.001	0.001	0.001	0.002	0.004	0.009	0.017

algorithm presents an overall rising trend, while the matching quantity firstly decreases and then increases, finally becoming steady.

Taking into account experimental error, the change of the *r* values is negatively at first and then becomes positively correlated with matching time and quantity indexes of SIFT algorithm. That is, matching time and quantity of SIFT algorithm firstly decrease and then increase, and finally become steady, with *r* values increasing from left to right. Therefore, *r* is considered as a positive correlation parameter of matching performance in the test improving matching rate of SIFT algorithm.

4. Parameter NBP

According to Table 12 and Fig. 19, NBP value needs to be a positive; thus, only the value of 2 can be taken before the initial value of 4. Experiment results are within the whole taking value interval after changing the parameter of NBP. The matching time of SIFT algorithm firstly increases and then declines, while the matching quantity presents an overall declining trend.

Considering effect of experimental error, the influence of NBP value change on matching time of SIFT algorithm firstly enhances and then decreases, with the increasing parameters of NBP values taken from left to right. There is an overall negative correlation

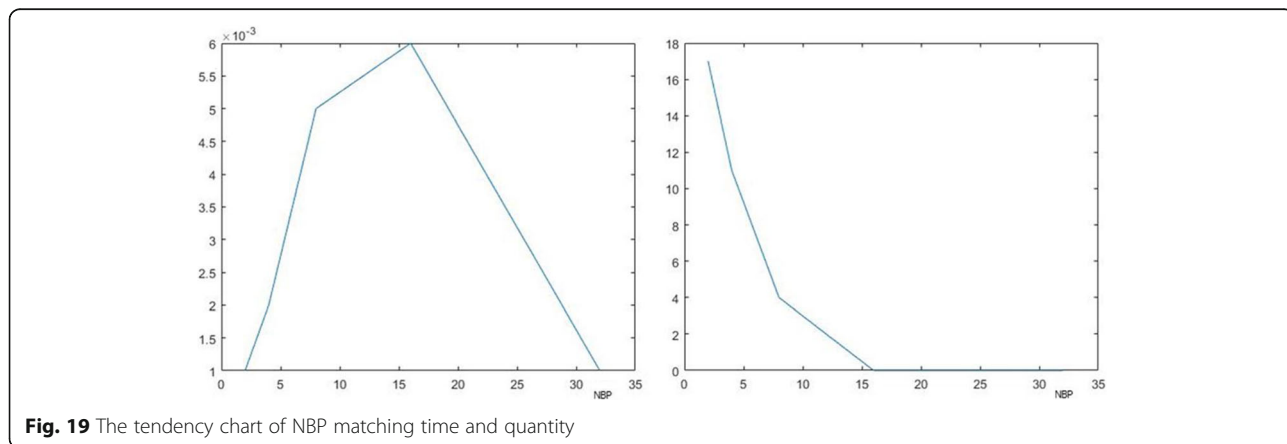


Fig. 19 The tendency chart of NBP matching time and quantity

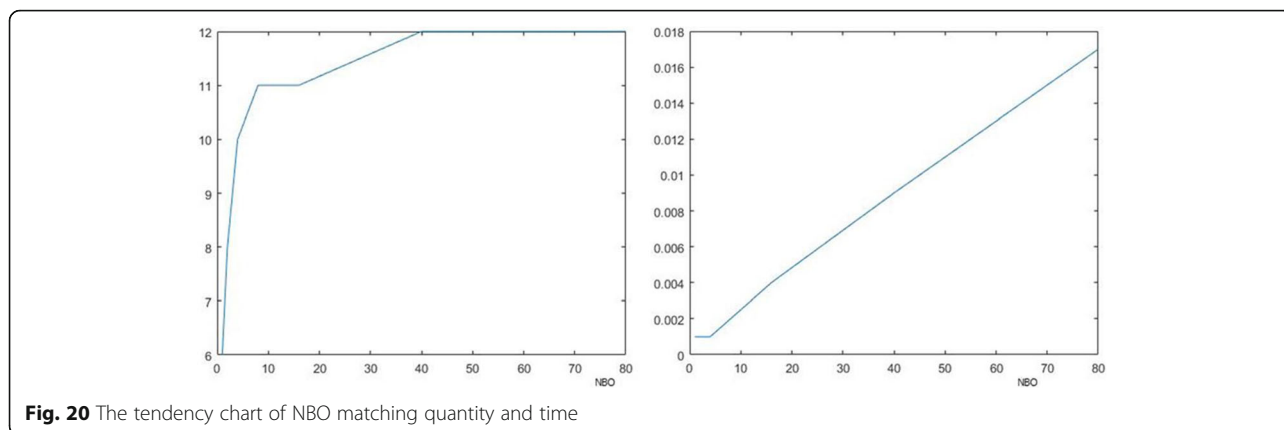


Fig. 20 The tendency chart of NBO matching quantity and time

between matching number and NBP value change. Thus, NBP is regarded as a negative correlation parameter of matching performance in improving matching rate of SIFT algorithm.

5. Parameter NBO

Table 13 and Fig. 20 show that experiment results are within the whole taking value interval after changing the parameter of NBO. The matching time of SIFT algorithm presents an increasing trend, while the matching quantity firstly increases and then becomes steady.

Considering the effect of experimental error, the change of NBO parameter value has a positive correlation with matching time and quantity of SIFT algorithm, with the increasing parameters of NBP values taken from left to right. Therefore, NBO is regarded as a positive correlation parameter in setting of NBO parameter values during the test improving matching rate of SIFT algorithm.

6. Parameters magnif

Table 14 and Fig. 21 show that experiment results are within the whole taking value interval after changing the parameter of magnif. The matching time of SIFT algorithm shows a decreasing trend, while the matching quantity firstly increases and then decreases.

Table 14 Matching quantity and time of magnif

Parameter	Magnif	Initial value					
Values	0.01	0.5	2	3	6	12	18
Matching quantity	0	16	13	11	3	1	1
Matching time	0.005	0.005	0.003	0.002	0.001	0.001	0

Taking into account experimental error, the change of magnif values has a negative correlation with matching time of SIFT algorithm, with the increasing parameters of r values taken from left to right. For matching quantity, the correlation is firstly positive and then becomes negative on the whole. Therefore, when setting parameter values of magnif in the test improving matching rate of SIFT algorithm, a better match rate will be obtained at the value of about 0.5.

The overall trend of the results is well demonstrated, though generation of errors cannot be avoided due to the limitations of experimental samples. Figure 22 shows the matching results.

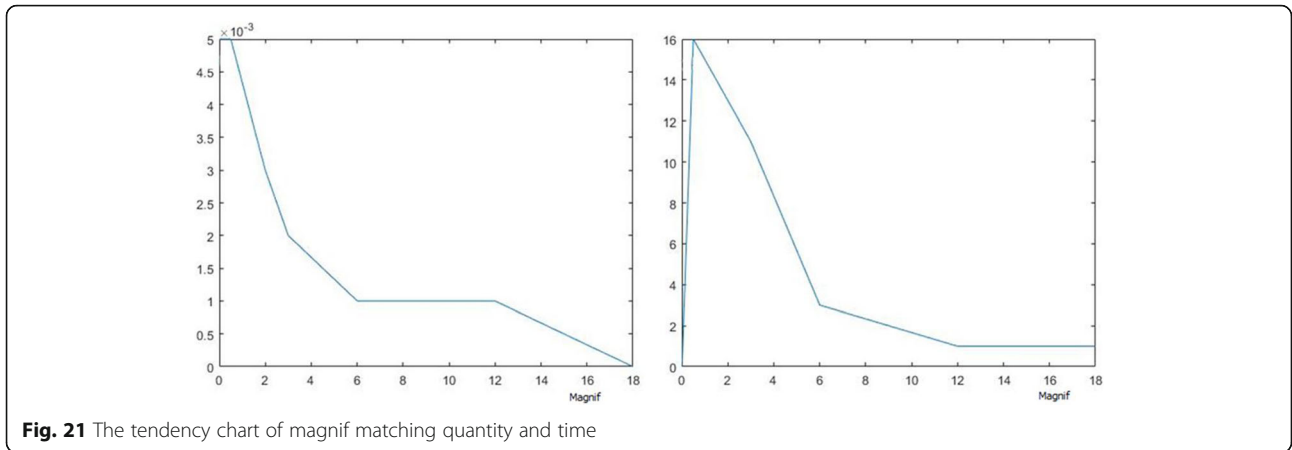
5.1.3 Summary of experimental results

In terms of the images of illumination and angle changes, Table 15 shows the influence of each parameter change on matching performance.

5.2 Game test

1. Class testing

The parameters in Easy AR need to be primarily modified to load different scenes when designing codes. A numerical manager (AllARguments function) was designed to manage all the codes. The loading of various scene models was changed through ray detection (Ray function) and level switch (Interim function). Using the overall UI management (TheAirshipUI function), we achieved image recognition, identification, deletion, and other general functions. The function values of such core code areas would be returned to other functions. The functions were running independently except for the link with core code block. Such design is consistent with design principles of “high cohesion and low coupling.” The designed code



framework was divided into core code block and other code blocks, where the former was used to control the code of other blocks. Figure 23 shows specific code framework.

Class testing was mainly performed for the following functions: Login interface (TheLoding function), spacecraft UI manager (ARairship function), energy tank UI (AREnergyCupola function), UI management of biological engineering cabin (ARbioengineeringCupola function), combat cabin UI (ARcombatCupola function), UI management of defense device (ARcombat function), UI manager of main compartment (ARmainCupola function), collection cabin of UI manager (ARgatherCupola function), library UI management (ARlibrARYCupola function), UI management of repair cabin (ARrepairCupola function), etc. In battle scene, the testing functions involved management generated by random enemies (RandomEnemy function), enemy AI (EnemyAll function), and weapon AI (WeaponAll function). In addition, the codes achieving other functions were tested, such as the particle scale effect

(ScalePARTicles function), text flicker effect (FlashBook function), and music manager (MusicManage function).

2. Running test in development environment of Unity3D

Unity 3D has run-time debugging as other development software. The achievements of the developed game were better displayed, with detection on realization of various functions of the game. The game was run by clicking the Run button (see Fig. 24).

3. Testing on mobile terminal

Mix series were used in the game development, where series of Mix 4 and 5 could run the game smoothly. Testing was also performed in other phone models. According to the testing results, phone models that can run the game include Mix 4, Mix 5s, LeEco 2x620, Vivo x7, and Noblue note. Figure 25 shows part of the phone

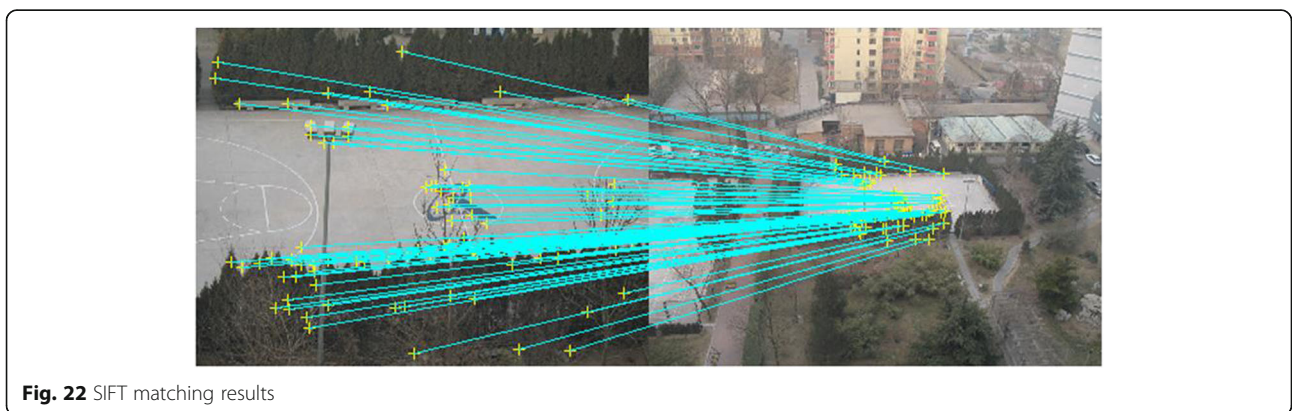


Table 15 Influence of parameters on SIFT matching performance

Parameter	Influence on SIFT matching performance
Sigman	The change of Sigman value is irrelevant with the matching quantity on the whole. Thus, Sigman can be regarded as an irrelevant correlation parameter in the test improving matching rate of SIFT algorithm.
Thresh	The change of thresh values is negatively correlated with matching quantity of SIFT algorithm. Therefore, thresh parameter is regarded as a negative correlation parameter in the test improving matching rate of SIFT algorithm.
<i>r</i>	The change of the <i>r</i> values is negatively at first and then becomes positively correlated with matching time of SIFT algorithm, presenting a positive correlation trend after the initial value of 10.
NBP	The change of NBP values is negatively correlated with matching quantity; thus, NBP is regarded as a negative correlation parameter of matching performance in improving matching rate of SIFT algorithm.
NBO	The change of NBO values has overall positive correlation with matching quantity. It can be regarded as a positive correlation parameter in setting of NBO parameter values during the test improving matching rate of SIFT algorithm.
Magnif	The change of magnif values is positively at first and then becomes negatively correlated with matching quantity. When setting parameter values of magnif in the test improving matching rate of SIFT algorithm, a better match rate will be obtained at the value of about 0.5.

properties. Currently, the phone model that fails passing the test is Vivo Y51A, caused by analytical package error resulting from too low Android version.

6 Results and discussion

The work developed a mobile AR game based on technology of SIFT image recognition, with main conclusions as follows.

1. Compared with the traditional image recognition, bold and new breakthroughs are implemented in terms of technology. The recognition algorithm is based on random pictures taken by gamers in the game according to their needs rather than fixed images. Experimental results indicate that such method can improve accuracy of image matching with similar structures. According to performance testing of algorithm, the algorithm can meet the requirement for precision and instantaneity of the system. AR management system of the cloud is beneficial to management on AR data of different applications.
2. In terms of game play method, the work changes the previous situation of single play method in RTS game. Features of tower defense and role play features are integrated in the game to provide rich and colorful game experiences.
3. For interactive mode of the game, touch-click way is adopted in interaction as there are not AR game specifications within the industry.

The main limitation of the work is that the matching algorithm is prone to errors in matching process of SIFT image, due to influence of factors such as light,

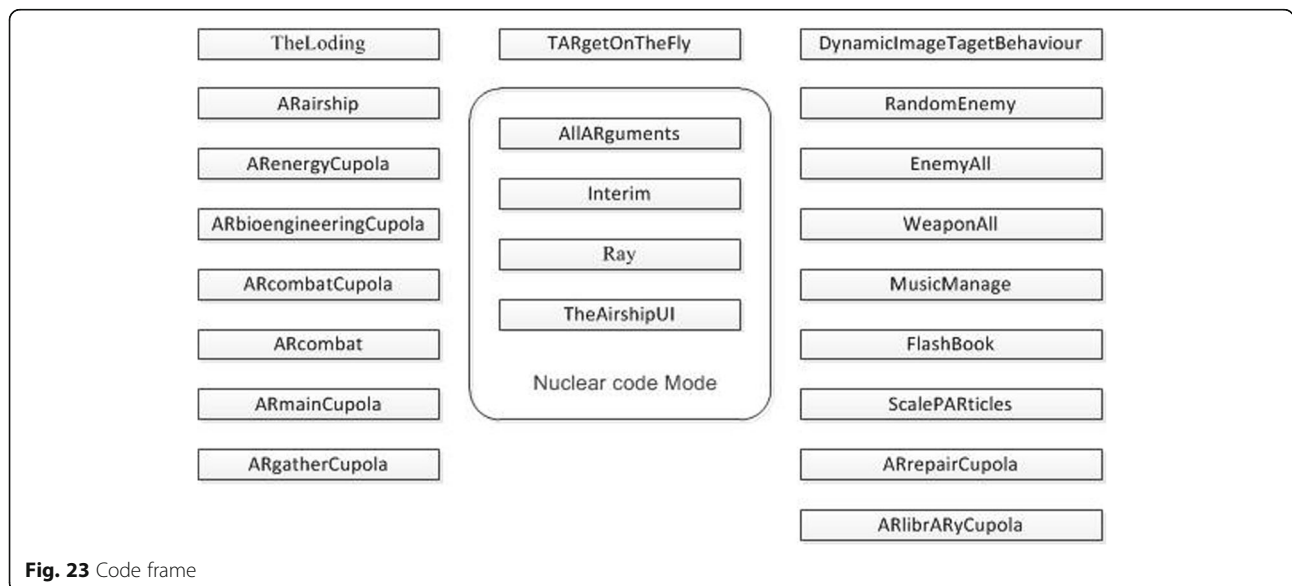


Fig. 23 Code frame



Fig. 24 Computational results in Unity 3D

occlusion, and camera movement. Sensor is considered to be used for correction. In addition, the game has not yet been available on iOS.

The main work in the work is conducted based on traditional SIFT algorithm. There are still many problems left to be improved due to personal ability and limited time. The future work can be taken from the following three aspects:

1. According to the conclusion of the study, the SIFT algorithm should be continuously improved to increase the matching rate and robustness.
2. The parallel operation of SIFT algorithm can be optimized to improve the speed of image processing, and the performance of SIFT can be optimized by combining with hardware equipment.
3. In the process of SIFT image matching, the matching algorithm is prone to errors due to many factors such as illumination, occlusion, and fast moving camera.

The sensor is used for correction in the later work. The game has not been released on iOS.

7 Conclusion

The present job designed an augmented reality game based on SIFT image recognition built on C/S structure. This game was divided into subsystems of cloud management and intelligent mobile terminal management. In this game, players can use both cloud database images and random images grabbed during game playing for image recognition which can be utilized for new game scene designing. To accelerate image recognition, in this paper, SIFT algorithm was improved by setting optimizing parameters and eliminating wrong matching points. The improved SIFT algorithm was integrated in Easy AR and working as a plug-in unit in the cloud server. The game is currently testing in laboratories before a hopeful public marketing.



Fig. 25 Test results for different mobile phone models. **a** The results of Spaceship scene **b** The results of Bioengineering space **c** The results of Fight scenes. 📷 Taking pictures Button for Players. ✅ The confirm button for Players. 💾 Save button. 🚪 Exit button

Acknowledgements

The author would like to thank the editors and anonymous reviewers for their valuable comments.

Availability of data and materials

Data will not be shared; reason for not sharing the data and materials is that the work submitted for review is not completed. The research is still ongoing, and those data and materials are still required by the author and co-authors for further investigations.

Author's contributions

BZ designed the research, analyzed the data, then wrote and edited the manuscript.

Funding

This work was supported by the science and technology key project of Henan Province, China. NO.172102210462.

Author's information

Boping Zhang is currently an Associate Professor at the School of Information Engineering, Xuchang University, China. She received master's degree from Zhengzhou University, China, in 2006. Her current research interests include computer vision, image processing, virtual reality, and pattern recognition.

Competing interests

The author declares that she has no competing interests.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Received: 21 September 2017 Accepted: 29 November 2017

Published online: 20 December 2017

References

1. A Ronald, A survey of augmented reality. *Presence Teleop. Virt.* **6**(4), 355–385 (1997)
2. Azuma, R. Recent advances in augmented reality[J]. *IEEE Comput. Graph. Appl.*, 2001,12(11):34-47.
3. Feiner, S., MacIntyre, B., Hollerer, T. et al., *A Touring Machine: Prototyping 3D Mobile Augmented Reality Systems for Exploring the Urban Environment*[A] *Proc. ISWC '97 (First Int. Symp. on Wearable Computers)*[C], vol. 1 (Cambridge, Personal Technologies, 1997), pp. 208. doi:10.1007/BF01682023
4. B Thomas, EB Close, J Donoghue, et al., *AR Quake:An Outdoor/Indoor Augmented Reality First Person Application*[a]. *Proceedings of 4th International Symposium on Wearable Computers*[C] (Cieeee Computer Society, Australia, 2000), pp. 139–146
5. Cheok, A. D. (2003) Human Pacman: A Mobile Entertainment System with Ubiquitous Computing and Tangible Interaction over a Wide Outdoor Area. In: Chittaro L. (eds) *Human-Computer Interaction with Mobile Devices and Services*. Mobile HCI 2003. Lecture Notes in Computer Science, vol 2795. Springer, Berlin, doi:10.1007/978-3-540-45233-1_16
6. KL Schrier, *Revolutionizing History Education: Using Augmented Reality Games to Teach Histories*[D] (Massachusetts Institute of Technology, Cambridge, 2005)
7. Google releases online reality augmented mobile games "Ingress"[OL]. <http://www.csdn.net/article/2012-11-16/2811943-google-launches-ingress>. Accessed 28 Jul 2017
8. M Villamizar, J Andradecetto, A Sanfeliu, et al., Bootstrapping boosted random ferns for discriminative and efficient object classification[J]. *Iberian Conf. Pattern Recognit. Image Anal.* **45**(9), 3141–3153 (2012)
9. L LIN, K YANG, Y-t WANG, et al., Key issues study for mobile augmented reality system[J]. *J. Image Graph.* **14**(3), 560–564 (2009)
10. D Lowe, Distinctive image features from scale invariant keypoints[J]. *Int. J. Comput. Vis.* **2**(60), 91–110 (2004)
11. Yan Lei,Yang Xiaogang et al. Mobile augmented reality system design and application based on image recognition [J], *J. Image Graph.*, 2016, 21(2):184-191.
12. DG Lowe, *Object Recognition from Local Scale-Invariant Features*// *Proceedings of the International Conference on Computer Vision* (Corfu, 1999), pp. 1150–1157
13. MINGA-LMA Hua-Dong, Region-SIFT descriptor based correspondence between multiple cameras[J]. *Chin. J. Comput.* **12**(4), 650–662 (2008)
14. K He, X Wang, G Yunfeng, Adaptive support-weight stereo matching algorithm based on SIFT descriptors[J]. *J. Tiajin Univ.* **49**(9), 978–984 (2016)
15. DG Lowe, Distinctive image features from scale-invariant key points. *Int. J. Comput. Vis.* **60**(2), 91–110 (2004)
16. C Guangxi, G Zhenting, et al., Fast image recognition method based on locality-constrained linear coding[J]. *Comput. Sci.* **43**(5), 308–314 (2016)
17. B Tingzhu, H Xibao, An improved image matching algorithm based on SIFT [J]. *Trans. Beijing Inst. Technol.* **33**(6), 622–627 (2013)
18. Z Wei, L Wei, P Jiexin, A image stitching method based on SIFT and region selection[J]. *Microelectron. Comp.* **27**(6), 205–207 (2010)
19. M Hincapie, A Caponio, Rios, *An Introduction to Augmented Reality with Applications in Aeronautical Maintenance. Consumer Electronics (ISCE)* (2013), pp. 197–198
20. Q Zhou, J Luo, Artificial neural network based grid computing of E-government scheduling for emergency management. *Comput. Syst. Sci. Eng.* **30**(5), 327–335 (2015)
21. Q Zhou, R Liu, Strategy optimization of resource scheduling based on cluster rendering. *Clust. Comput.* **19**(4), 2109–2117 (2016). <https://doi.org/10.1007/s10586-016-0655-9>
22. Q Zhou, Research on heterogeneous data integration model of group enterprise based on cluster computing. *Clust. Comput.* **19**(3), 1275–1282 (2016). <https://doi.org/10.1007/s10586-016-0580-y>
23. Zhou, Q. *Electron Commer Res* (2017). doi:10.1007/s10660-017-9265-8. Accessed 28 Jul 2017
24. Q Zhou, J Luo, The study on evaluation method of urban network security in the big data era. *Intell. Automation Soft Comput.* (2017). doi:10.1080/10798587.2016.1267444
25. Pirschheim C, Schmalstieg D, Reitmayr G. Handling Pure Camera Rotation in Key Frame-Based Slam[C]//2013 IEEE International Symposium on Mixed and Augmented Reality. Adelaide: IEEE 2013:229-238.

Submit your manuscript to a SpringerOpen[®] journal and benefit from:

- Convenient online submission
- Rigorous peer review
- Open access: articles freely available online
- High visibility within the field
- Retaining the copyright to your article

Submit your next manuscript at ► springeropen.com