A Human-Robot Interactive Mahjong Playing System Based on Visual Recognition Using a Convolutional Neural Network

Zhifang Wu
School of Automation Science and Engineering, Faculty of Electronic and Information Engineering, Xi'an Jiaotong University, Xi'an, Shaanxi 710049, China
wuzf2019@stu.xjtu.edu.cn

Jiuqiang Han
Guangdong Artificial Intelligence and Digital Economy Laboratory, Guangzhou, Guangdong 510335, China
jqhan@mail.xjtu.edu.cn

Erhu Liu
School of Automation Science and Engineering, Faculty of Electronic and Information Engineering, Xi'an Jiaotong University, Xi'an, Shaanxi 710049, China
liuerhu@stu.xjtu.edu.cn

Hongqiang Lyu
School of Automation Science and Engineering, Faculty of Electronic and Information Engineering, Xi'an Jiaotong University, Xi'an, Shaanxi 710049, China
hongqianglv@mail.xjtu.edu.cn

ABSTRACT

Mahjong is a popular tabletop game in China, Japan, and other Asian countries. As an incomplete information game, it is more complicated than other complete information games, such as Go, chess, and shogi. With the rapid development of service robot, there have been several human-robot interactive playing systems for complete information games so far, but mahjong is not the case. In this paper, a human-robot interactive mahjong playing system (HRMPS) was developed. HRMPS consists of five modules, including central host, robot players, visual kit, mahjong conveyor, and interactive software. The central host serves the communication between four robot players, which grab a tile on mahjong conveyor and recognize its face via visual kit, and make an action decision by themselves or by human opponents with the help of the interactive software. In HRMPS, to visually recognize a total of 27 different mahjong faces in uncontrolled conditions, a deep convolutional neural network was adopted to achieve an accuracy of 99.71% with a running time of 29ms. The experimental results tell that HRMPS is applicable in human-robot interactive mahjong game.

CCS CONCEPTS

• **Human-centered computing**; • **Human computer interaction**; • **Interactive systems and tools**;

KEYWORDS

Human-robot interaction, Mahjong playing system, Visual recognition, Convolutional neural network

ACM Reference Format:
Zhifang Wu, Jiuqiang Han, Erhu Liu, and Hongqiang Lyu. 2021. A Human-Robot Interactive Mahjong Playing System Based on Visual Recognition Using a Convolutional Neural Network. In *The 5th International Conference on Computer Science and Application Engineering (CSAE 2021), October 19–21, 2021, Sanya, China.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3487075.3487188

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
CSAE 2021, October 19–21, 2021, Sanya, China
© 2021 Association for Computing Machinery.
ACM ISBN 978-1-4503-8985-3/21/10…$15.00
https://doi.org/10.1145/3487075.3487188

1 INTRODUCTION

Tabletop games are popular all over the world. Generally speaking, most tabletop games can be roughly sorted into two categories according to the available information within the game, including complete information games and incomplete information games [1]. In a complete information game, the game information about every player is available to all the players, such as Go, chess, shogi, etc. In an incomplete information game, players do not possess full information about their opponents and the available information that one participant possesses is acquired from cards of her/his own, such as mahjong, poker, Texas hold'em, bridge, etc [2]. Take mahjong for an example, four players take turns to draw and discard tiles (i.e. cards in poker games) to form four melds (or sets) and a pair (eye) but they don't know some information of other players, including their "type", strategies and preference [3]. With the acceleration of the rhythm of people's life, tabletop games become one of the most popular ways of entertainment. Compared to virtual online games, human-robot tabletop game playing systems are closer to reality and help to increase the player's interest. They also have a prospect application in commercial use, e.g. in a chess room or a mall. Customers can get a modern and technical sense when they play these tabletop games with physical robots. Many emerging techniques such as speech interface [4], affect recognition [5], and hand gesture recognition [6, 7], have been applied to these systems to make users feel more natural and user-friendly. Furthermore, a human-robot interactive tabletop game playing system can be considered as a testbed for human-robot collaboration, since such a system usually provides rich interaction opportunities that arise when humans and robots play collaboratively as a team. Research on it may pave the way for human-robot Interaction (HRI) in a more complex and less constrained situation.

Over the years, numerous studies have been conducted in the human-robot interactive tabletop game playing system. Urting D et al. [8] proposed a chess-playing robot named MarineBlue. It determines whether a square was occupied through classifying the

squares according to their color, and the type of pieces was determined by analyzing the moving chess. No further piece recognition step is needed since the chess piece can be located by comparing two frames before and after a move when the initial board situation is known. Matuszek C et al. [9] designed Gambit, an autonomous chess-playing robot system, where a chessboard was located by detecting board corner points, point cloud consisting of all points above the surface of the board plane were used to determine the positions of occupied chess squares, and the scale-invariant feature transform (SIFT) algorithm was used to identify the 6 types of chess pieces (King, Queen, Pawn, Rook, Bishop, and Knight). Guillermo L et al. [10] designed a computer vision system for chess-playing robots, compared three classic algorithms in detecting whether a square is occupied, and concluded that the multisampling and voting classification algorithm was the most robust but also the slowest. Chinese chess and shogi are the tabletop games for two players and each piece has a character pattern printed on. WenYuan. C. [11] developed a method against noise and rotation for 7 types of Chinese-chess character (General, Advisor, Elephant, Horse, Chariot, Cannon, and Soldier) recognition based on feature comparison techniques. Gou K. et al. [12] developed a robust vision system for shogi robots. The 10 types of pieces (Pawn, Silver, Gold, Bishop, Rook, King, Promoted Pawn, Promoted Silver, Horse, and Dragon) were recognized by cross-correlation matching. In the mahjong series, there are few studies on human-robot interactive game playing system and vision-based method for mahjong recognition. Tang et al. [13] developed an Anti-swindle Mahjong Leisure System (AMLPS) which used mature radio-frequency identification (RFID) technology to identify the mahjong tiles (27 classes) and prevent cheating in a game. In their system, the cost of mahjong tiles became higher because the redesigned mahjong tiles whose back contained a unique RFID tag were needed.

Different from the tabletop games mentioned above, mahjong is an incomplete information one, and usually, the tiles are arranged with the back upwards to keep the information private to others. It is significant for robots to acquire the game information (i.e. recognize all the tiles) effectively and accurately in a human-robot interactive mahjong playing system. However, it is difficult to recognize the 27 classes precisely using traditional methods that combine feature extraction and feature comparison since the features effective in one suit are likely inapplicable for another suit. Besides, uncontrolled conditions, such as illumination variations, make it challenging to achieve high-accuracy recognition utilizing traditional methods. Deep learning, a subset of machine learning, has revolutionized many exciting tasks in recent years, ranging from image classification to natural language processing [14]. In the field, there are some typical neural networks show the superiority in multi-class classification task, such as AlexNet [15], GoogLeNet [16], VggNet [17], ResNet [18] and SqueezeNet [19]. SqueezeNet, a smaller convolutional neural network (CNN) structure, is able to achieve AlexNet-level accuracy on ImageNet with 50x fewer parameters. As [19] pointed, this small structure with equivalent accuracy requires less training time and is more feasible to deploy on other hardware with limited memory.

In this paper, a human-robot interactive mahjong playing system (HRMPS) based on visual recognition was designed and completed. The four robot players realize the task of gripping tiles, discarding tiles, and making strategies with the help of a visual kit, in which the images used for detection and recognition of tiles are captured by 2 cameras, respectively. And a CNN-based method is adopted to recognize tiles for high accuracy. The structure of the rest of the paper is organized as follows: Section 2 introduces the overall design of this system, including the architecture of HRMPS and details of 5 designed modules; In Section 3, the methods that are applied to detect tiles, recognize tiles and make strategies are introduced. Section 4 describes the hardware and software components of HRMPS. Section 5 presents the results, including performance of the CNN model and the experiments on HRMPS. In Section 6, the conclusions are given.

2 OVERALL DESIGN OF HRMPS

The overall architecture diagram of HRMPS is shown in Figure 1. HRMPS is primarily comprised of five modules, including central host, robot players, visual kit, circular conveyor, and interactive software (Figure 1(a)). The central host is designed to control each process of HRMPS. It serves the communication between four robot players, shares the game data with interface software, and controls the speed of circular conveyor. The robot players aim to grip the tile on the circular conveyor, recognize the face with the help of the visual kit, and make decisions by themselves. The visual kit takes 3 tasks of capturing mahjong images by cameras, detecting mahjong tiles on circular conveyor, and recognizing the tiles. The interactive software is employed to establish communication between robots and humans, including displaying the game states, refreshing the system reactions, and inputting strategies from human opponents.

The detailed design of HRMPS is shown in Figure. 1(b). Since the robot arm is in limited length, a circular conveyor instead of traditional tables is adopted to convey tiles to the accessible area of robot players. Each robot player completes vision tasks with the help of a visual kit that consists of a client and two cameras. And one camera (camera 1) is installed on the robot arm for mahjong back detection while the other (camera 2) is fixed on the countertop for mahjong face recognition. When the game starts, the robot player collects the image information on the circular conveyor continuously by camera 1, so that the tile near the gripper can be detected, and its spatial coordinates are calculated. Then the gripper waiting above conveyor is lowed and closed to pick mahjong tile up while the tile enters into its area, and moves this tile just above camera 2. Meanwhile, the face image of the tile is captured and recognized accurately utilizing the CNN-based model. Next, the robot player analyses the recognition result to decide which tile should be discarded, and the decision is sent to central host. After that, the interactive software receives the information from the host, refreshes the game state, and displays the discarded tile. At the same time, central host gives the information to all robot players and instructs next robot player to take action. If a person participates in the game, he or she makes strategies through interactive software. In a Mahjong game, four players take turns taking actions including picking a tile, recognizing the pattern on it, and making a decision according to the available information and the rules of mahjong game. In HRMPS, the opponents are robot players in default, and one or more human players can participate in the game by switching the corresponding robot from autonomous decision mode to manual

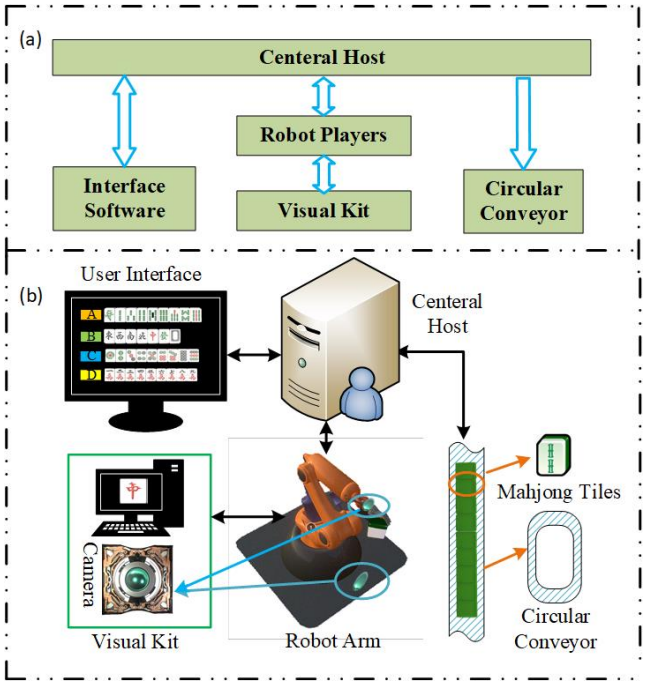

Figure 1: The Overall Architecture of Designed HRMPS. (a) The Architecture Diagram of Five Modules in HRMPS; (b) The Details of HRMPS.

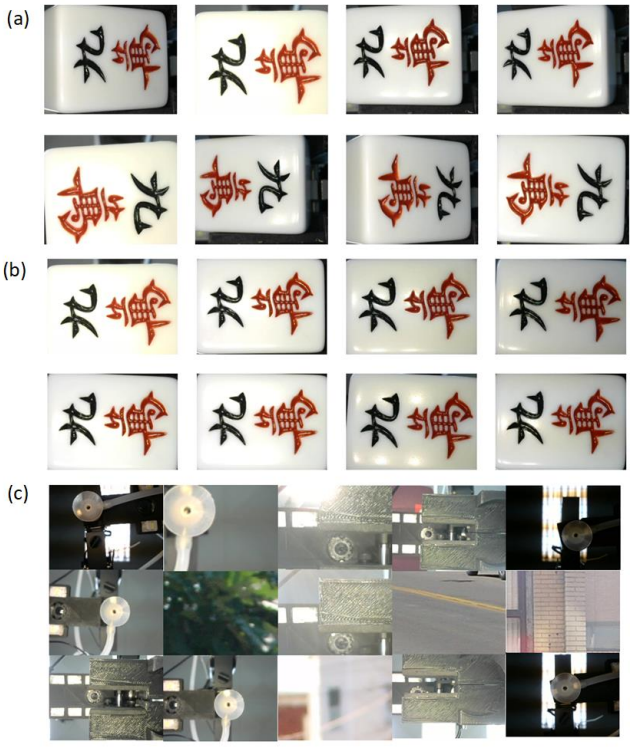

Figure 2: Some Positive and Negative Samples of Mahjong Dataset. (a) Some Samples of Tile Character 9 under Different Illumination Conditions; (b) Some Samples of Tile Character 9 under Different Inclination Angles; (c) Some of the Negative Samples, Such as Images of Ceilings and Halls.

decision mode. Game state and decisions made by players will be displayed on the user interface.

3 METHODS

3.1 Back Detection

To guide the robotic arm to grab mahjong tile from the conveyor in real time, visual detection of mahjong back is necessary. Considering the features of Mahjong tiles whose back is of one color, such as green, yellow, and red, the following image processing is employed to achieve the task. Firstly, RGB channel separation is used to detect mahjong tiles in the image captured by Camera 1. Next, the central coordinate of the mahjong tile in the image is calculated after the area filtering of this image. Finally, according to the relationship of the image coordinates, camera coordinates, and gripper coordinates, the accurate position of the mahjong tile can be calculated.

3.2 Face Recognition

After a mahjong tile is grabbed from the conveyor, an accurate and efficient visual recognition of mahjong face becomes one of the most important issues for HRMPS. Deep neural networks have shown an ability to outperform traditional feature-based approaches in computer vision tasks [20]. And SqueezeNet, which is comprised mainly of Fire modules, can achieve performance comparable with other CNN frameworks such as AlexNet while requiring fewer parameters. In this study, a CNN model based on Fire module is employed to recognize 27 classes of tiles under uncontrolled conditions, such as illumination variation.

3.2.1 Datasets. The CNN-based method usually benefits from a large amount of data [21], and a balanced dataset is essential to how the classifier performs since these models are sensitive to training sample distribution [22]. To collect enough and balanced mahjong images, some data augmentation techniques are used to enlarge the data set, and samples of each class are made in the approximately same size. In the details, this mahjong dataset, a total of 42900 images including 38610 positive samples and 4290 negative samples, are collected by the following steps. For positive samples, 4158 images of mahjong tiles (154 images each class, 27×154) under different inclination angles and illumination conditions are collected manually. Then data augmentation techniques, including rotation and scale, are employed to get more samples. After that, each class has 1430 samples so that the dataset contains a total of 38610 mahjong images. Herein, some samples of the tile *Character 9* under different inclination angles and illumination conditions are shown in Figure 2(a) and Figure 2 (b), respectively. Besides, for negative samples, considering the case that the gripper fails to grip mahjong tile, some samples such as images of chuck, gripper, and ceiling are collected. Also, indoor images such as halls and venues are added to enhance the system's adaptability to the environment

Table 1: The Network Architectural Dimensions

Layer name	Output size	Filter size/stride	Depth	$S_{1\times1}$	$e_{1\times1}$	$e_{3\times3}$
image	227×227×3	/	/	/	/	/
conv1	113×113×64	3×3/2(×64)	1	/	/	/
pool1	56×56×64	3×3/2	0	/	/	/
fire2	56×56×128	/	2	16	64	64
fire3	56×56×128	/	2	16	64	64
pool3	28×28×128	3×3/2	0	/	/	/
fire4	28×28×256	/	2	32	128	128
fire5	28×28×256	/	2	32	128	128
pool5	14×14×256	3×3/2	0	/	/	/
fire6	14×14×384	/	2	48	192	192
fire7	14×14×384	/	2	48	192	192
fire8	14×14×512	/	2	64	256	256
fire9	14×14×512	/	2	64	256	256
conv10	14×14×28	1×1/1	1	/	/	/
pool10	1×1×28	14×14/1(×28)	0	/	/	/

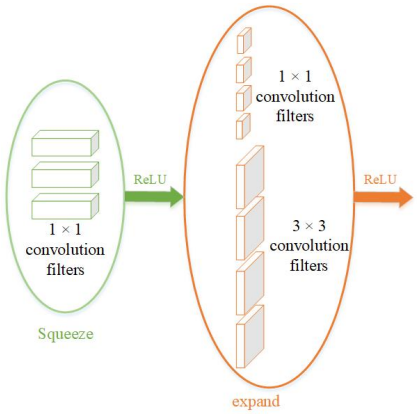

Figure 3: The Fire Module in SqueezeNet.

changes. And 4290 negative samples are prepared finally. Some of the negative samples are shown in Figure 2 (c).

3.2.2 Structure of the CNN Model. SqueezeNet is a smaller CNN architecture that uses fewer parameters while maintaining competitive accuracy [23]. One of the main features of SqueezeNet is Fire modules in which a squeeze convolution layer with only 1×1 convolution filters is fed into an expand layer with a mix of 1×1 and 3 × 3 convolution filters. And a total of 3 strategies that can be used to customize SqueezeNet are described as: (1) replace 3 × 3 filters with 1 × 1 filters, (2) decrease the number of input channels to 3 × 3 filters, (3) downsample late in the network so that the convolution layers have large activation maps. The convolution filters organization in a Fire module is shown in Figure 3

The architecture of the CNN-based model for face recognition in HRMPS is shown in Figure 4. We design this network based on Fire module and modify it corresponding to our application by trial and error. The network starts with a standalone convolution layer(conv1), followed by eight fire modules (fire 2-9) that are designed to hold a small total number of parameters, ended with a final convolution layer (conv10). The number of filters per fire module is increased gradually from the beginning to the end of the network. The max-pooling with a stride of two is performed after conv1, fire3 and fire5. And the average pooling is performed after conv10. ReLU is adopted as the activation function and Dropout with a ratio of 0.5 is applied after the fire9 module. In the final, Softmax is selected as the classifier of CNN. In Table 1, the parameters of CNN architecture are described in detail.

3.2.3 Training Setup. The CNN is trained on the mahjong dataset. According to the rule of thumb, that is, 80% of the dataset is selected as the training set and 20% as the validation set. During the training procedure, the pre-training weights of SqueezeNet are taken as the initial weights and the loss function (i.e. mean squared error) is defined as,

$$L = \frac{1}{2n}\sum_{i=1}^{n} y_i - \theta\left(\mathbf{x}_i\right) \|_2^2 \tag{1}$$

where $\theta(\mathbf{x}_i)$ is the output of the CNN for an input $\mathbf{x}_i$, and $\mathbf{y}_i$ is the actual label. Note also that n is the number of training samples.

The mini-batch gradient descent algorithm [24] (batch size: 512) is adopted to solve the optimal problem mentioned above (minimize the loss function). The rule of parameters updating is given as,

$$\omega_{j+1} \leftarrow \omega_j - \frac{1}{B_n}\eta\sum_{i=1}^{B_n}\frac{\partial L^{(i)}}{\partial \omega_j} \tag{2}$$

where ω_j is the current weight, B_n is the batch size, η is the learning rate, and $L^{(i)}$ is the loss of i_th sample. η is a very important hyper parameter which affect the convergence of the training and the performance of the network [25]. And in our work, the best value 0.006 are determined through a grid search method.

3.3 Action Decision

An action decision strategy based on Belief-Desire-Intension (BDI) model [26] is employed in HRMPS. The essential assumption of the BDI model is that actions are derived from a process named

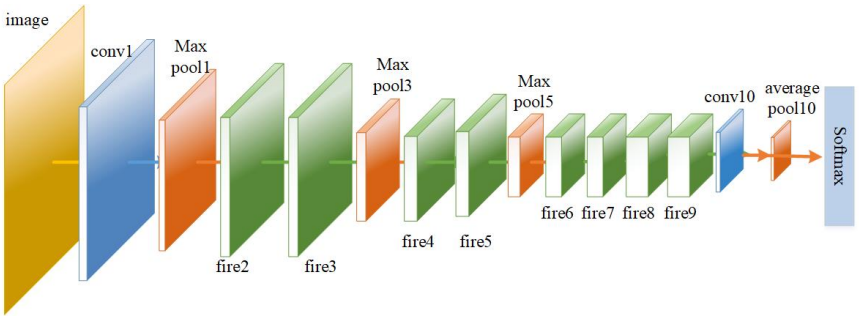

Figure 4: The Architecture of CNN for Recognition of Mahjong Tiles.

practical reasoning, which is composed of two steps. In the first step, deliberation (of goals), a set of desires is selected to be achieved, according to the current situation of the agent's beliefs. The second step is responsible for the determination of how these concrete goals produced as a result of the previous step can be achieved by means of the available options for the agent [27]. In the mahjong game, the memory set, rule set, behavior set, and state set constitute the robot's belief. The memory set stores the events that occurred, including the tiles that the robot has got and other robots have discarded. The rule set stores mahjong game rules, and behavior set is the actions of the robot, such as discarding a tile, *Chow*, *Pong*, and *Kong*. The state set contains the robot's knowledge, i.e. the tiles to be dropped and drawn for winning the game, which is the basis of forming the goal. The actions of a robot player are decided by the following steps. Firstly, update the robot's belief according to the environmental information. Then, form the goal set by selecting the state in which the number of dropped and drawn tiles is fewest. Next, decompose the goals to obtain the states that can be reached in one step. Finally, match these states with the behaviors to determine the current action.

4 COMPOSITION OF HRMPS

Hardware and software are completed in light of the designed HRMPS. The hardware is comprised of robots, cameras, computers, and a circular conveyor. The robot here is a modified robotic arm that takes tasks of physical movement, gripping tiles, and dropping them. Cameras are devoted to capturing the images for back detection and face recognition of tiles. Computers are used for managing the game process, processing the captured images, and communicating with humans. And a circular conveyor conveys mahjong tiles to the accessible area of grippers. The software consists of two parts including software on central host and clients. The former contains a game logic module that manages the game process and a GUI module provides users an intuitive game interface. The software in clients consists of an algorithm engine and an interactive module.

4.1 Hardware

There are four robots in HRMPS and these robots play an important role in the task of gripping mahjong tiles successfully. To make the robotic arm stable, the grippers are modified both in the appearance and structure based on a second-generation *Dobot* manipulator which is selected as the prototype (Figure 5(a)). In terms of details,

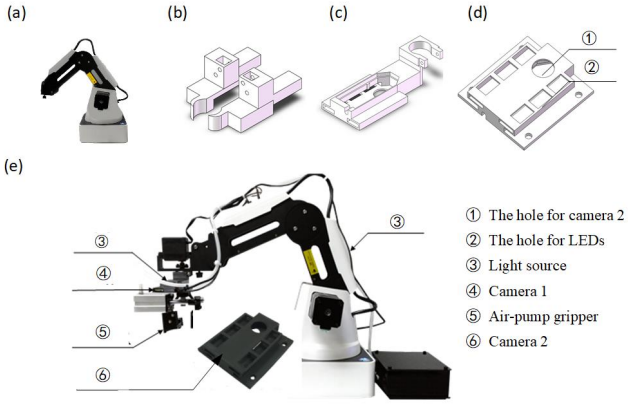

Figure 5: The Main Hardware of Robot. (a) The Prototype of Robot; (b) The Model of 3D Printed Air Pump-Driven Gripper; (c) The Model of Designed Housing for Camera 1; (d) The Model of Designed Housing for Camera 2; (e) The Modified Robot.

the redesigned gripper (Figure 5(b)) is an air pump-driven gripper printed by 3D printing technology and a pipe with an inner diameter of 4mm connected between the air pump and the cylinder.

As for cameras, HRMPS needs 4 suit cameras (i.e. 4 camera 1 and 4 camera 2). And considering the actual installation of them in HRMPS, two different types of housing were designed and manufactured with the help of a 3D printer. The circuit boards of the two cameras were embedded inside. One housing is mounted on the end of the robot arm for camera 1 (Figure 5(c)) and the other is fixed on the countertop (Figure 5(d)) for camera 2. Besides, two lines of LED lights are arranged on both sides of the two types of cameras to resist the ambient light to some extent. The schematic diagram of the modified arm is shown in Figure 5(e).

Five computers are used in HRMPS. Among them, one is the host and the other four play the roles of brains of the four robots respectively. Each computer of the four processes the information to locate and identify mahjong tiles, communicate with both the arm and the host, and displays the game interface.

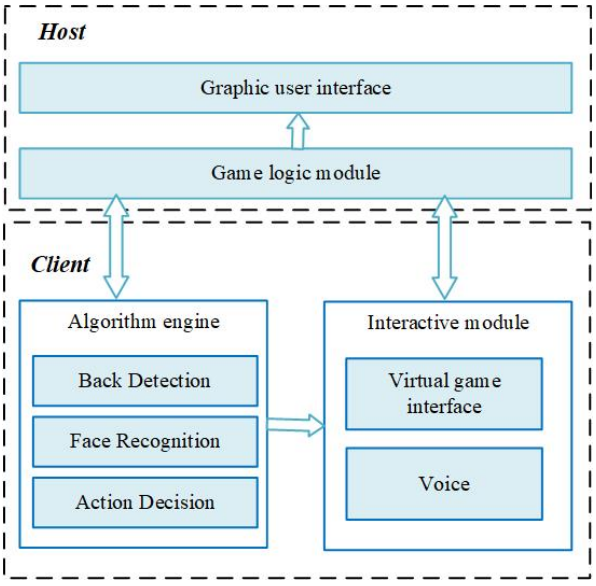

Figure 6: The Schematic Diagram of Software in HRMPS.

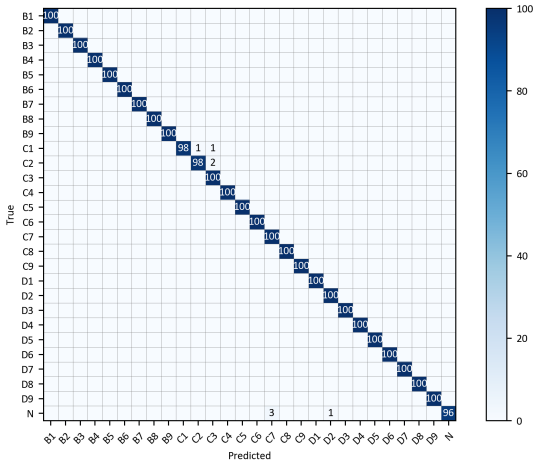

Figure 7: The Confusion Matrix of Recognition Results.

4.2 Software

The software of central host mainly consists of two modules, one is the game logic module and the other is the GUI module. The game logic module manages the game process, such as deciding whose turn to take action. The GUI module is responsible for displaying the tiles of all game players. The software on the clients contains two modules, i.e. algorithm engine and interactive module. The algorithm engine integrates three algorithms including back detection method, face recognition method, and action decision method. And two functions are integrated into the interactive module. One is a virtual game interface to display game status of the current player in real-time, and the other is to broadcast robots' or human players' decisions by voice, such as which tile is discarded, and which player is winning. The schematic diagram of the software is shown in Figure 6

5 RESULTS AND DISCUSSION

5.1 Performance of the CNN Model

The performance of the proposed CNN model for mahjong face recognition is evaluated in terms of accuracy and efficiency. And a comparable experiment with traditional HOG+SVM has been conducted. The test results show that the accuracy of the CNN model reached 99.71%, while the classical method HOG+SVM was only 96.18%. Considering the running speed of one frame, the CNN model only needs 29ms, while HOG+SVM requires 53ms, 2 times of the former or so. The running time was obtained on a computer with Intel ®Core™ i5-4570 processor and 4GB RAM. These results (Table 2) demonstrate that this CNN model satisfies the requirements of both accuracy and real time, and it is more suitable than traditional method in HRMPS.

Moreover, the system with the CNN model deployed is tested, which aims to evaluate the performance of the CNN model in a relatively real environment. In the test, Mahjong tiles are put on the circular conveyor and then a robot player detects, grasps, and recognizes them. Each mahjong tile of 27 classes and negative samples are tested 100 times which means a total of 2800 times. Figure 7 presents the confusion matrix that shows the details of this classifier. The result indicates 25 classes are identified with a true positive rate of 100% and less than 5 samples are confused in *Character 1(C1)*, *Character 2(C2)*, and negative samples (*N*). This confusion matrix indicates that such a model performs well in mahjong classification and it is applicable in HRMPS.

5.2 Experiments

The experiments are conducted on the completed HRMPS (Figure 8(a)) and the four robot players work collaboratively in a mahjong game. When a robot player receives the instruction of drawing a tile, a series actions including detecting the accessible tile (Figure 8(b)), grasping it (Figure 8(c)), moving it to the above of camera 2, and recognizing it (Figure 8(d)) are carried out smoothly.

The software on HRMPS provides users an intuitive game interface. Figure 9(a) shows the GUI on central host, which displays

Table 2: Results of Comparable Experiments on Test Set

Methods	Total number of samples	Number of true prediction	Accuracy	Running time
HOG+SVM	2800	2693	96.18%	53ms
CNN	2800	2792	99.71%	29ms

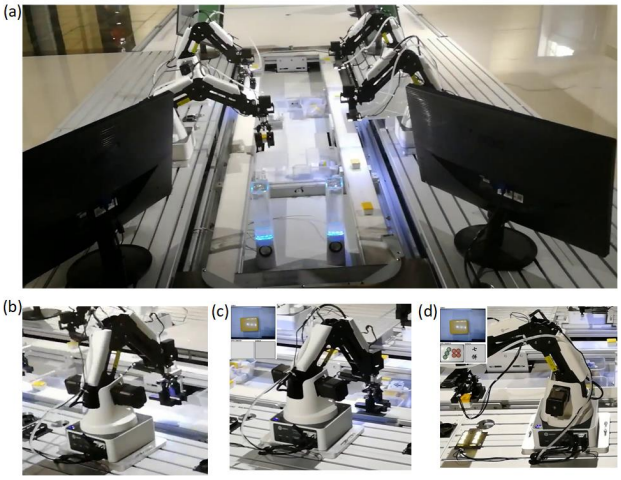

Figure 8: Experiments on HRMPS. (a) The Physical Appearance of HRMPS; (b) A Robot Player is Detecting Mahjong Tiles; (c) Gripping the Mahjong Tile; (d) Recognizing the Mahjong Tile.

the game status including all tiles of four robot players in hand, the tiles were discarded, and the picked tile currently. The client's interactive software (Figure 9(b)) contains two panels, the left one presents the recognition result of the drawn tile, and the right is a virtual mahjong game interface with real-time data.

6 CONCLUSIONS

In this paper, a human-robot interactive mahjong playing system named HRMPS was proposed. It provides a platform where human opponents can play a mahjong game with robots that realize the tasks of gripping tiles, recognizing tiles, and making strategies by themselves. Five modules of HRMPS, including central host, robot players, visual kit, circular conveyor, and interactive software, were designed and completed. In the visual kit, the mahjong dataset was built with 42900 images, and the CNN-based method was employed to make the recognition in high accuracy and real-time. To evaluate the performance of this method, comparable experiments with the HOG + SVM method were conducted. The results demonstrated this method outperforms HOG+SVM in the recognition task with an accuracy of 99.71% and a running time of 29ms. Moreover, it indicated that combining the deep learning model to a human-robot interactive game system can improve the vision ability of robot players and the efficiency of HRI. A primary goal of research in HRI has been to investigate 'natural' means by which a human can interact and communicate with a robot. In terms of HRMPS, a fun robot with personalized strategies is significant to deliver a 'natural' gaming experience to human opponents. Thus in future work, we will focus on game strategy optimization which aims to make robots do much of what a human player would do.

ACKNOWLEDGMENTS

This research was funded by the National Natural Science Foundation of China, grant number 61602367.

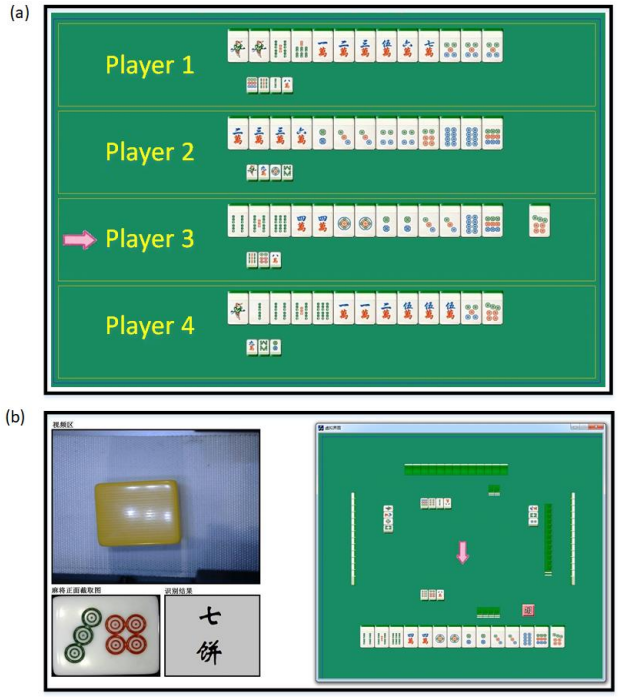

Figure 9: The Experimental Results Displayed on (a) the Monitoring Software for Central Host and (b) the Interactive Software for Clients.

REFERENCES

[1] R Gibbons (1992). *A Primer in Game Theory.* Hertfordshire, U.K.: Harvester Wheatsheaf.
[2] Milgrom P R, Weber R J (1981). Distributional strategies for games with incomplete information. Mathematics of Operations Research, 10(4), 619-632.
[3] Wang M, Yan T, Luo M, and Huang W (2019). A novel deep residual network-based incomplete information competition strategy for four-players mahjong games. Multimedia Tools and Applications (6218), 1-25.
[4] Zhang J; Zhao J; Bai S and Huang Z (2004). Applying speech interface to Mahjong game. In Proceedings of 10th International Multimedia Modelling Conference.
[5] Castellano G, Leite I, Pereira A, Martinho C and Mcowan P W (2010). Affect recognition for interactive companions: challenges and design in real world scenarios. Journal on Multimodal User Interfaces, 3(1), 89-98.
[6] Lee D H and Hong K S (2011). Game interface using hand gesture recognition. Computer Sciences and Convergence Information Technology (ICCIT), 2010 5th International Conference on IEEE.
[7] Sriboonruang Y, Kumhom P, and Chamnongthai K (2006). Visual Hand Gesture Interface for Computer Board Game Control. In Proceedings of 2006 IEEE International Symposium on Consumer Electronics.
[8] Urting D and Berbers Y (2003) MarineBlue: A Low-cost Chess Robot. In Proceedings of Robotics and Applications.
[9] Matuszek C , Mayton B , Aimi R and Deisenroth M P (2011). Gambit: An autonomous chess-playing robotic system. In Proceedings of 2011 IEEE International Conference on Robotics and Automation.
[10] G Larregay, F Pinna, L Avila, and D Morán (2018). Design and implementation of a computer vision system for an autonomous chess-playing robot. Journal of Computer Science & Technology,18, 1-11.
[11] Chen W Y (2014). Chinese-Chess Image Recognition by using Feature Comparison Techniques. Applied Mathematics & Information Sciences, 8, 2443-2453.
[12] Koutaki G and Uchimura K (2015). Fast and Robust Vision System for Shogi Robot. J Robot Mechatron, 27, 182-190.
[13] Tang J (2014). Designing an Anti-swindle Mahjong Leisure Prototype System using RFID and ontology theory. Journal of Network and Computer Applications, 39, 292-301.
[14] Shrestha A and Mahmood A (2019). Review of Deep Learning Algorithms and Architectures. IEEE Access, 7, 53040-53065.

[15] Krizhevsky A, Sutskever I, Hinton, G E (2018). Imagenet classification with deep convolutional neural networks. In Proceedings of Advances in neural information processing systems.
[16] Szegedy C, Liu W, Jia Y, Sermanet P, Reed S and Anguelov D A (2014). Going deeper with convolutions. In Proceedings of the IEEE conference on computer vision and pattern recognition.
[17] Simonyan K and Zisserman A (2014). Very deep convolutional networks for large-scale image recognition. arXiv:14091556.
[18] He K, Zhang X, Ren S, and Sun J (2016). Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition.
[19] Iandola F N, Han S, Moskewicz M W, and Ashraf K (2016). SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and< 0.5 MB model size. arXiv: 1602.07360.
[20] Guo Y M, Liu Y, Oerlemans A, Lao S Y and Wu S (2016). Deep learning for visual understanding: A review. Neurocomputing, 187, 27-48.
[21] Sun C, Shrivastava A, Singh S and Gupta A (2017). Revisiting Unreasonable Effectiveness of Data in Deep Learning Era. 2017 IEEE International Conference on Computer Vision (ICCV).
[22] Khan S H, Hayat M, Bennamoun M, F Sohel and Togneri R (2018). Cost sensitive learning of deep feature representations from imbalanced data. IEEE Transactions on Neural Networks & Learning Systems, 29(8), 3573-3587.
[23] Zheng H, Gu N and Zhang X (2018). An Efficient and Slight Convolutional Neural Network for Vehicle Type Classification. Journal of Physics: Conference Series. IOP Publishing, 1069(1): 012116.
[24] Hinton G, Srivastava N and Swersky K (2012). Neural networks for machine learning lecture 6a overview of mini-batch gradient descent. 2012, 14.
[25] Smith L N (2018). A disciplined approach to neural network hyper-parameters: Part 1–learning rate, batch size, momentum, and weight decay. arXiv: 1803.09820.
[26] Georgeff M, Pell B, Pollack M, Tambe M and Wooldridge M (1998). The belief-desire-intention model of agency. In International workshop on agent theories, architectures, and languages. Springer, Berlin, Heidelberg.
[27] Wooldridge M J K (2000). Reasoning about Rational Agents. Kybernetes, 5, 161-174.