The Design of Deep-Learning-Based Facial Recognition System for Smart Shopping Cart

Ching-Ya Liu, Ying-Jen Chen, *IEEE Member*

***Abstract*—This world is full of science and technology. With the rapid of development of mobile devices, applications and demands of the biometric identification technology are also expanding rapidly. Compared with the traditional identity authentication method, biometric identification technology is not easy to be forgotten and copied, and it is one step closer in terms of security and convenience. In recent years, the combination of the Artificial Intelligence and the Internet of Things can be seen in cities, which has a significant impact on stores, transportation and even foods. This article aims to design develop a deep learning based facial recognition system for the smart shopping cart. This system combines traditional shopping carts in shopping hypermarkets with tablet computers. Users can use tablet to browse all commodities in the hypermarket, and use facial recognition technology to enhance security and convenience in user login. This system uses the tablet to obtain the user's face samples and preprocesses the image for the important feature extraction. Moreover, the matching network learning with a small training set is applied for facial recognition to identify users when there are not enough user samples.**

I. Introduction

There are many researches on face recognition [1]-[5]. Since the facial recognition system is to be applied on the shopping cart in a hypermarket, there will not be many image samples for users. Therefore, using traditional convolutional neural networks can't play a big role. For this reason, our research uses matching network, which only needs a small number of samples, for the face recognition system. At first, the face sample acquired by the tablet is extracted by the Dlib library combined with OpenCV for important feature extraction. Then the processed face images are input into the neural network for the training. Moreover, we change the Long Short-Term Memory (LSTM) [6] neural network architecture of the matching network to GRU (Gate Recurrent Unit) [7]. This relatively lightweight architecture can save time during training and can almost maintain the recognition rate of the original architecture.

II. Related Work

In this research, we need to find facial features in the image to ensure that the training samples are correct. In addition, we need a model that can be accurately identified using a small amount of samples because the number of face images for each user is small for a hypermarket.

A. Facial landmarks with Dlib

We use the built-in camera lens of the tablet as the image sensor, and use Dlib for facial feature extraction of the acquired image. The pre-trained face landmark detector in the Dlib library is used to estimate the position of the 68 (x, y) coordinates mapped to the facial structure of the human face. The index of the 68 coordinates can be displayed as shown in Fig. 1. The Dlib library utilizes the algorithm mentioned in [8], which is a regression tree method based on gradient enhancement learning. The algorithm uses cascading regression factors. First, a series of calibrated face images are required as a training set, and then a model is generated. When a picture is obtained, the algorithm will generate an initial shape. A rough feature point position is estimated, and then the gradient boosting algorithm applied to reduce the sum of the square error of the initial shape and ground truth. In addition, the least square method is utilized to minimize the error and to get the cascaded regression factor for each level.

B. Matching Network for One Shot Learning

Matching Network is a new method that uses memory augment neural networks and metric learning. The feature extractors for the support set image and the input image are different. The similarity of the cosine distance is used to compare the input image with each image in the support set, and finally the softmax is used for classification. The model structure of matching network is shown in Fig. 2.

Figure 1. The index of 68 coordinates.

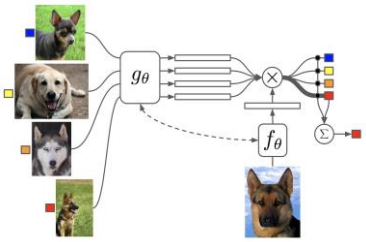


Figure 2. The structure of Matching Network.

Research supported by the Ministry of Science and Technology of Taiwan (MOST 109-2221-E-305-008-).

C. Y. Liu is with the Department of Electrical Engineering, National Taipei University, New Taipei City 237303, Taiwan (R.O.C.) (e-mail: angus09855069@gmail.com).

Y.-J. Chen is with the Department of Electrical Engineering, National Taipei University, New Taipei City 237303, Taiwan (R.O.C.) (corresponding author to provide phone: +886-2-8674-1111#68819; fax: +886-2-26736500; e-mail: yjcheng@mail.ntpu.edu.tw).

The input samples $\hat{x}$ of the model need to be matched with a support set S of k samples $S = \{(x_i, y_i)\}_{i=1}^{k}$. When given a new support set in the testing process, then the model that learned before can use to get the possible $\hat{y}$ for each test sample. The calculating of $\hat{y}$ is given as

$$\hat{y} = \sum_{i=1}^{k} a(\hat{x}, x_i) y_i \quad (2)$$

where $a(\hat{x}, x_i)$ is used to measure the matching degree between $\hat{x}$ and the training sample x_i, and the calculation of the test sample label through y_i is similar to the weight summation. The function $a(\hat{x}, x_i)$ of is shown below

$$a(\hat{x}, x_i) = e^{c(f(\hat{x}), g(x_i))} / \sum_{j=1}^{k} e^{c(f(\hat{x}), g(x_j))} \quad (3)$$

where f is defined to encode the test sample into a vector, and the g is the sample encoding of the support set. It is similar to the deep convolutional network for processing images, such as VGG and Inception.

III. Research Method

This section introduces how to obtain the user's face image for training and display the recognition result. The flowchart of overall process is shown in Fig. 3. First, when the user uses the shopping cart tablet to register, the tablet camera will capture the user's face image. Then the image is transmitted to the high-performance laptop on the shopping cart for image processing. The captured image will be stored in the back-end database. Other pre-created face images is applied as a support set to train a face recognition matching network model. Finally, the user can use face recognition for login next time.

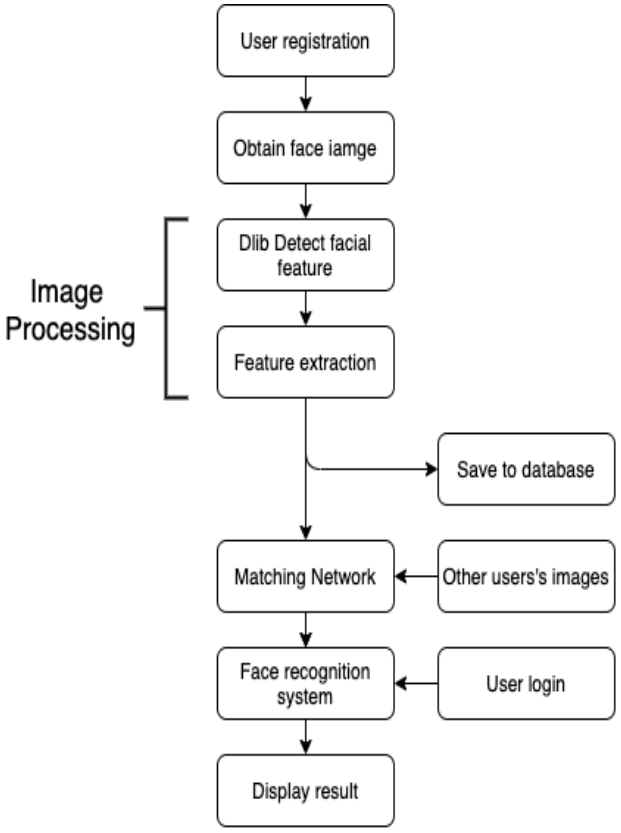


Fig. 3. The structure of smart shopping cart face recognition.

A. Image processing

As long as the user's face remains front, the Dlib library can correctly detect facial features, such as nose, eyes, eyebrows, mouth, etc.. Then the bounding box of the face selected by Dlib is converted to the OpenCV-style bounding box. In this way, the region of interest (ROI) of the face image can be obtianed. The important features of the user's face image are detected, and the 68 coordinate points are marked out and bounded with the bounding box as shown in Fig. 4. The detected result is cast into a new image according to the bounding box as shown in Fig. 5. Finally, the image is transmitted to the back-end database.

B. Neural Network Architecture

About the matching network for one shot learning, this architecture was proposed by the team members of Google DeepMind. From the architecture diagram shown in Fig. 6, it can be seen that the encoder g encodes the support set; the encoder f encodes the input are output, and finally the label belongs its output. The contents of g and f are embedding function of CNN and bidirectional LSTM. The support set S is regarded as a sequence. In the process of feature extraction, LSTM is used to allow all images to interact. The reason why the author uses LSTM in the article is that it can quickly learn from the data displayed in the sequence. LSTM contains a block that is marked as a smart network unit, because it can memorize the value of a variable length of time. In addition, there is a threshold in the block to determine whether the input is important enough to be remembered and whether to output. However, It still takes a lot of time during the training process. Hence we transform LSTM into another type, GRU, to speed up execution. Compared with LSTM, the structure of GRU is much lighter.

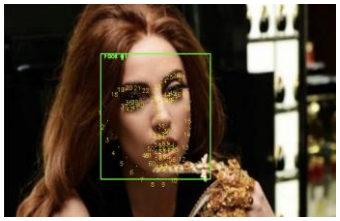


Figure 4. Facial feature coordinates.

Figure 5. Capture ROI.

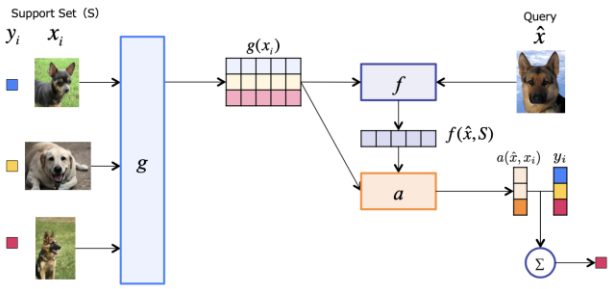


Figure 6. Encoder process architecture.

C. User Interface and Backend Database

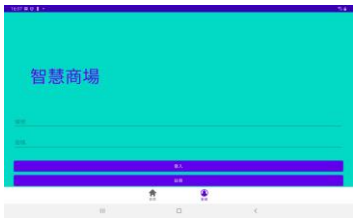
We use a tablet and the Android system for the user interface, and apply Android Studio for development. The "smart shopping cart store" APP includes user registration, login, and product browsing. The information of commodities identified by the camera lens with deep learning in the shopping cart will also be displayed on the main page on which there are the commodity information, price, and the total price of all commodity in the shopping cart. The back-end database is created by MySQL. The users information includes personal data, face images, shopping history records, such as commodities, expenditures, etc. will be written into the database.

IV. Experiment Results

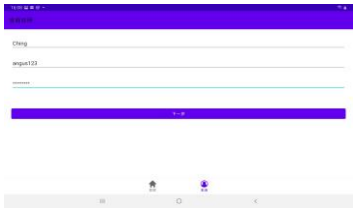
This section is divided into several steps to show the experimental results. This experiment of the face recognition system uses a tablet computer to obtain images and use a high-performance computer on a smart shopping cart for face recognition. Therefore, through the results of this experiment, we can see the possibility of this idea.

A. Image Capture

For the face recognition system, the first step is to obtain training samples. Use the tablet computer to obtain the user's image, and then apply the method in section III. A. to perform feature extraction of the image and then save it to the back-end database. The user use the tablet to login, register, and take a photo of face image for face recognition login system as shown in Fig. 6.



(a)



(b)

(c)

Figure 7. Face recognition login system: (a) login, (b) register and (c) take a photo of face image.

B. Training and identification accuracy

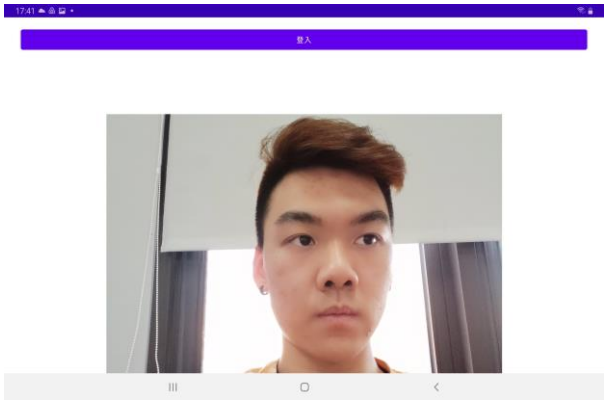
In the previous step, we have training samples to the backend database. After that, we can produce labels for all users and support set from the other training samples. The selected way is divided into N-way K-shot where N is the number of selected categories and K is the number of samples selected for this category. The training process will be divided into 5-way 1-shot and 5-way 5-shot and divided into two different neural network architectures LSTM and GRU, respectively. The identification results are shown in Table 1. It can be seen that the LSTM method has better recognition results. However, applying GRU, which is a much lightweight system, can save lots of training time that is more suitable for practical applications.

C. Face Recognition Login

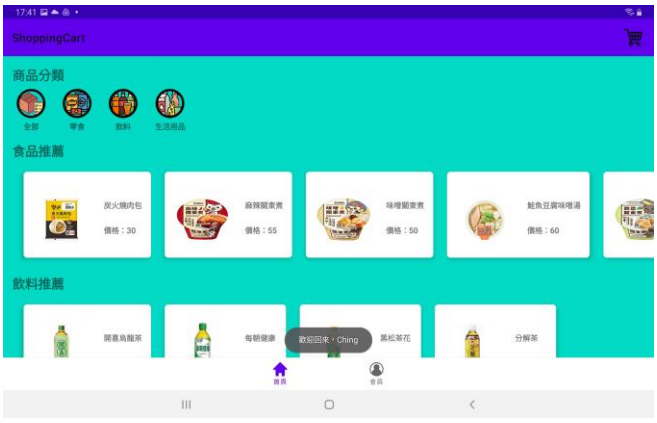
After completing the registration, the user can directly take a photo and quickly login when using the smart shopping cart next time. After logging in, the related functions of the smart shopping cart can be directly used as shown in Fig. 8.

TABLE I. Table Type Styles

	5-way 1-shot	5-wat 5-shot
LSTM	90.8%	93.8%
GRU	83%	89.5%



(a)



(b)

Figure 8. Face recognition logging in: (a) login by face recognition (b) related functions of the smart shopping cart.

V. Conclusion

The purpose of this research is to design a face recognition system on a smart shopping cart. It not only combines IoT technology, but also applies machine vision and deep learning. In this system, the connection of each step is very important. A successful process is to obtain several more correct face images, perform image processing, and then use these images as training samples to train the deep learning neural network model for the face recognition system. Finally, you can use face recognition for quick login at next time you use it. In this system, there are many things that can be improved and optimized, such as the structure configuration. Environmental conditions also have an impact on machine vision, such as light and reflections.

References

[1] O. M. Parkhi, A. Vedaldi, and A. Zisserman, "Deep face recognition," in Proceedings of the British Machine Vision Conference (BMVC 2015), Swansea, UK, 2015.

[2] Y. Sun, X. Wang, and X. Tang, "Hybrid Deep Learning for Face Verification," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 38, no. 1, pp. 1997-2009, 2016.

[3] Weiyang Liu, Yandong Wen, Zhiding Yu, Ming Li, Bhiksha Raj, Le Song, "SphereFace: Deep Hypersphere Embedding for Face Recognition," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 212-220

[4] J.Deng, J.Guo, N.Xue, S.Zafeiriou, "ArcFace:AdditiveAngularMargin Loss for Deep Face Recognition," in CVPR, Jun. 2019, pp. 4685-4694.

[5] Xi Yin and Xiaoming Liu, "Multi-Task Convolutional Neural Network for Pose-Invariant Face Recognition," IEEE TRANSACTIONS ON IMAGE PROCESSING, VOL. 27, NO. 2, FEBRUARY 2018.

[6] S Hochreiter, J Schmidhuber, "Long Short-Term Memory," Neural computation, 1997 - MIT Press.

[7] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, Yoshua Bengio, "Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling," arXiv preprint arXiv:1412.3555, 2014-arxiv.org

[8] Vahid Kazemi, Josephine Sullivan, "One Million Face Alignment with an Ensemble of Regression Trees," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2014, pp. 1867-1874.