

Recognition of the smart card iconic numbers

Xue Shi Xin^{1,a}, Qing Song¹, Lu Yang¹ and Wen He Jia¹

¹Automation School, Beijing University of Posts and Telecommunications, Beijing, China

Abstract. Manual way of recognizing and inputting the smart card iconic numbers leads to much time consuming, high resources consuming and low accuracy. This paper presents a new automatic method based on the technology of computer vision, including Harr operator for face detection, Fast operator for number location, image binarization, character segmentation and BP neural network for number recognition. Experimental results on 100 ID cards show that by adjusting the parameters of the BP neural network properly, the recognition accuracy is more than 99%, and the recognition time for each ID card is within 0.05 seconds.

1 Instruction

As information technology brought up the onset of intelligence age, smart cards play a significant role in our daily life. Commonly used smart cards like ID cards, credit cards play a significant role in our daily life, each smart card is distinguished by a series of unique numbers information (signature). These iconic numbers is widely used in public places like banks and airports, which calls for a high accuracy of identification and recording. Aimed at identifying and recording the numbers iconic effectively and efficiently we study the identification of Chinese ID cards.

Based on the highly developed computer vision technology, the identification and recording of ID cards are now solved by computer recognition. Yet there exists several difficulties in the process of recognition:

- Complex context. ID card's context usually consists of text in colour pattern counterfeit labels, thus a fliter of useless information is necessary during the identification.
- Abrasion and besmirch. A robust algorithm is needed to ensure accuracy.
- Various illumination environments. The identification system needs to adapt to different lighting situation.

To solve these problems, we handled the ID card image according to the computer vision techniques. We first locate the face position on the ID card by Haar face detection. Then target the ID card number area by considering the positions of text, number, face and FAST detectors. Next, we segment this target area into single characters and normalized them one by one. Finally, recognise each character with BP neural network recognition algorithm.

2 Scheme

In order to realize the ID card digitization, we take pictures of ID card and upload it to computer.

2.1 Image pre-process

Pre-process is necessary for all collected images. We mainly deal with noises and tilts in this essay, which are commonly caused by some inevitable factors like light or camera issues during the image collection. We effectively solve the noise by filtering and the tilt with Hough-Transform [1] method.

2.2 Image locating

As one of the most important characteristics, facial image has fixed size and position in an ID card, which benefits the ID card numbers locating. Based on Adaboost algorithm and Haar-like feature, the ID card number position can be inferred roughly by the fixed facial image position.

By using FAST detector [2], all the corners in the image can be found. Than a filter algorithm is used to all the corners to find the position of all words and numbers in the ID cards .Combine the result of face detect, the area of ID card numbers can be determined.

Segment the image of numerical strings into 18 independent character images after the area of numbers is inferred. To be used for the recognition, each of segmented images must be normalized.

2.3 Image recognition

A dataset including 5500 images is created to recognition the printed numbers.

In the process of printed numbers recognition, the printed number recognition is presented by neural

^a Corresponding author: songqing512@126.com

network based on BP [3]. Statistical test results of BP neural network in different parameters.

Figure 1 is a flow chart of scheme.

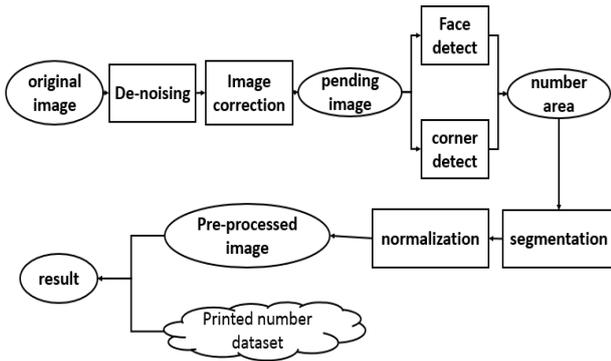


Figure 1. Flow chart of scheme.

3 Implementation

3.1 Image pre-process

3.1.1 Denoising

In order to make the image of the ID card smoother, the first processing is Gaussian filtering, which helps eliminate the noise and interference.

Two-dimension Gauss function:

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \quad (1)$$

Image of two-dimension Gauss function:

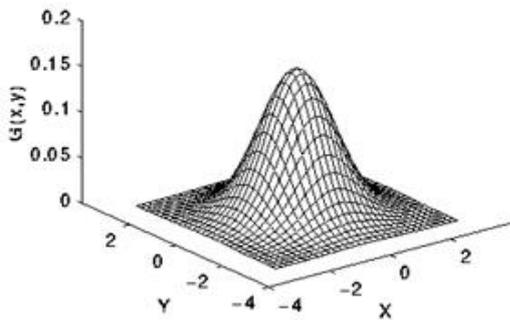


Figure 2. Image of two-dimension Gauss function.

Images in a computer are stored in the form of a two-dimension matrix. Gauss filtering makes use of two-dimension convolution operators of Gaussian kernel to convolve with the whole image successively. Therefore eliminate the noise and interference.

3.1.2 Image correction

Hough transform [1] can be applied to detection of objects of any shape. This method works well when being used to detect straight lines. The correction has high precision. The edges of ID cards have high linearity. We can figure out the angle of inclination of the photo by

using the straight line on the edge and then calibrate the photo.

3.2 Image locating

3.2.1 Face localization

ID cards have uniform size and style. Every ID card has a full-faced photo, which is 32mm*26mm, in the right side of the card. The position and size of the images are fixed. According to this characteristic, the position of other text messages can be roughly determined by using the relative position of the image.

This article locate the image by using face detection technology based on Adaboost and Haar-like [4] feature.

Haar-like feature includes central feature, line feature and edge feature. They are shown below:

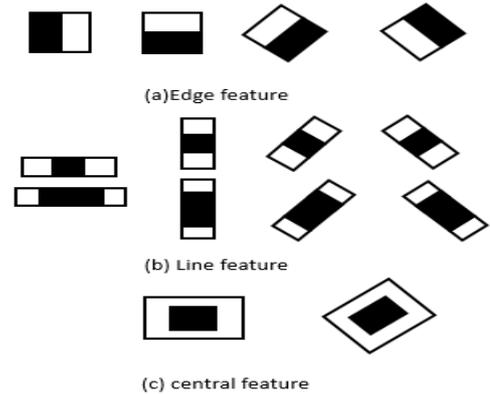


Figure 3. Haar-like feature.

In the process of faces detection, place random feature rectangles on the image in turn. The difference between sum of pixel of the white region and sum of pixel of the black region serves as a feature. If the rectangle contains faces, this eigenvalue serves as a face feature. Otherwise, this eigenvalue serves as a non-face feature. Different sizes of rectangles on different positions of the image provide large amounts of features. With Adaboost algorithm, several strong classifiers are trained by these features. After input images are detected by multiple strong classifiers, it can be confirmed that the image is a face. The effect of ID card images being detected by Haar-like tech:



Figure 4. The result of the face detect.

3.2.2 Corner detection

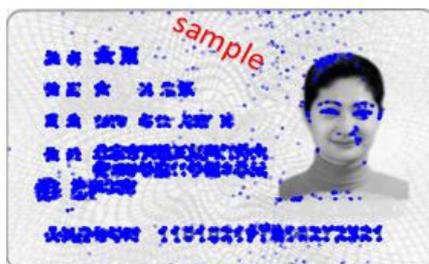
Corner detection, which is in the computer vision system, is a method used to acquire the feature of images. It's widely used in fields like motion detection, image matching, video tracking, 3-D modeling and target recognition. It's also called feature points detection. In the ID card's image, corner points focus on the edge of characters, figures and faces. Characters and figures' position can be detected by analyzing the distribution of corner points.

There are various kinds of method to detect corner points. This article adopts optimized fast corner detection algorithm to locate characters and figures in the image.

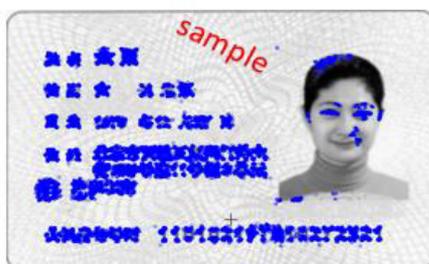
FAST was put forward by Edward Rosten and Tom Drummond in 1996. It's a simple and fast method to detect corners. According to this algorithm, a pixel is defined as a corner point if it has plenty of pixels, which are in different regions in its neighbourhood.

When directly using FAST detector, it generate many FAST detector around colour textures and stains so as to cause interference. Therefore, this paper intends to optimize the corner detection.

After detection by FAST, record location information of all corners. These corners includes large amount of noise points. A position pattern of corner can be got after inputting all the positions into a blank image. With image convolution method, convolve with the whole corner image. Then employ binarization to process the image and get a new image of corner position. New positions of corners correspond to the positions of characters and figures after denoising. Comparison chart of effect before and after optimizing in Figure 5.



(a) Before optimization



(b) After optimization

Figure 5. Figures before and after optimization.

3.2.3 Locating and segmentation of ID number

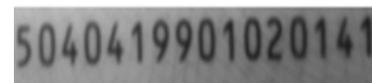
• Locating:

By locating the face and detecting corners, positions of faces, characters and figures in the image can be obtained. According to the positional relation between ID numbers

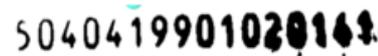
and the face, we can accurately conclude that the image to the bottom left of head (bottom left of the ID card), where corners are dense, is the image of ID number. Based on steps above, location of ID numbers can be acquired.

• Segmentation:

After accurately locate the ID number and extract its region, segmentation of figures shall be conducted. Conduct binarization with the extracted image first. Due to the uneven light, it does not work well by using fixed threshold method(Figure 6 (b)). This paper employs the adaptitave local threshold(Figure 6 (c)) to achieve binarization of the image with different threshold values at different positions.



(a) original image



(b) Fixed threshold



(c) Adaptive threshold

Figure 6. The images before and after binarization

After binarization, projecting the image vertically. Count up the number of dark spot of every row and get the figure 7.

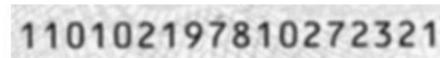


Figure 7. The vertical projection of the image

Select the centre of each blank area as segmentation point. The number segment is divided into 18 independent character images.

3.3 Recognition of ID card numbers

3.3.1 Image normalization

Pre-process always be used before the features of the numbers image are extracted. The pretreatment mainly refers to the normalization process. The process of the normalization include size normalization and position normalization.

• Position normalization:

Normalization method based the border positions is used for position normalization, find the position and center of

the border by calculation in the image of single number, than move the center to a specified position of a new blank image, position normalization is implemented by above processes.

• Position normalization:

The process of transforming the images of different size to the same size is the size normalization. first, calculate the length of minimum external positive square of the image, then change the length of the image by scale. Size normalization is implemented by above processes.

The result after the normalization is shown in Figure 8.

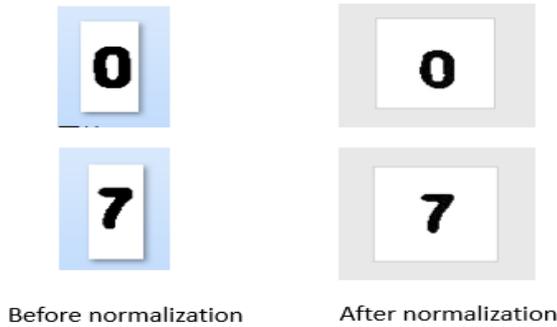


Figure 8. The normalization of the image

3.3.2 Dataset : Printed Number-11

In order to solve the problem of ID card numbers recognition, the methods of machine learning were used. In the processes of recognition, a dataset is needed to support the experiment. A dataset named NMIST was created by Corinna Cortes who works in Google Labs and Christopher J.C. Burges who works in Microsoft Research as shown in Figure 9 (a). The NMIST database is publicly available, has a training set of 60,000 examples, and a test set of 10,000 examples. It is a good database for people who want to try learning techniques and pattern recognition methods on real-world data while spending minimal efforts on preprocessing and formatting. However NMIST dataset is unsuited to printed numbers. Printed numbers has fixed font, and we need a high accuracy in the process of recognition, so this dataset is created for the printed numbers recognition.

The new dataset named Printed Number-11 as shown in Figure 9 (b) include the common fonts that the printed numbers used. Not only is the accuracy improved by using Printed Number-11, but also the situation that the numbers with wear and tear can be improved.

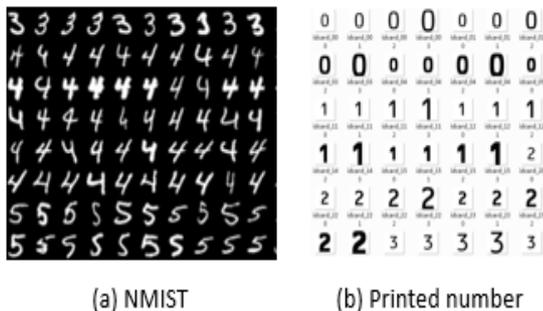


Figure 9. Datasets.

Printed Number-11 include 10 kinds of common fonts ,such as FARRINGTON-7B-Qiqi , Times New Raman , Arial etc. In each kind of font, ten numbers and a capital character “X” are included, each character saved in an image, resized to 100*100 pixels. In the process of creating dataset, different size and position of a same character are saved as different image as is shown in Figure 10 (a), (b), (c), then different degree of covering are used in each image which is to simulate the number with wear as is shown in Figure 10 (d). Eventually each character generated 500 images, a dataset include 5500 images is created to recognition the printed numbers.

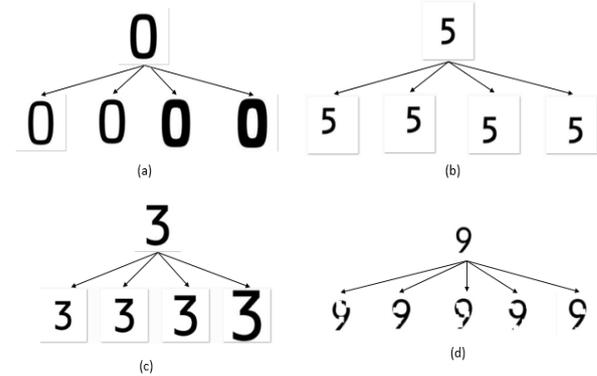


Figure 10. The process of creating dataset.

3.3.3 Recognition

In the section of printed numbers recognition, the printed number recognition is presented by neural network based on BP. As one of the important feed-forward neural network models, BP (Back-propagation) neural network is a kind of multilayer feed forward network with the nature of error back propagation.

In training stage, NMIST and Printed Number-11 is used to train different classifier, the highest accuracy used the NMIST is 95% and the highest accuracy used Printed Number-11 is more than 99%.The result show that Printed Number-11 is more suitable for printed numbers because the examples in Printed Number-11 is more similar to the printed numbers.

Recognition accuracy is affected by using different activation functions during training the neural network. For a high accuracy, this paper try to use 3 kinds of activation functions, logistic sigmoid, Tanh (hyperbolic tangent) and ReLu(rectified linear units). The plots of the functions is shown in Figure 11.

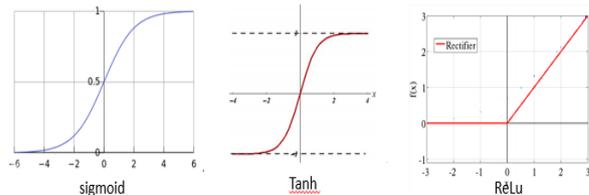


Figure 11. The plots of the activation functions.

Three kinds of different activation functions were selected for training different classifier in Single hidden layer neural network, then we use 100 ID cards to test, the result of recognition as shown in table 1.

Table 1. The performance by the different activation functions.

activation functions	correct /sum	accuracy
Sigmoid	2570/2700	95.2%
ReLu	2595/2700	96.1%
Tanh	2546/2700	94.3%

Recognition accuracy is affected by using different activation functions. The performance of Sigmoid and Tanh are unsatisfactory, the recognition results are 95.2% and 94.3%. ReLu has the best performance, the accuracy is 96.1%. The reason is that without pre-training, Sigmoid and Tanh can't reach the global optimum because the gradient vanishing problem, ReLu doesn't has this problem. Besides, the sparsity of the ReLu is can be adjusted, this is similar to the human' brain.

The neural network is composed of input layer, hidden layer and output layer. Input layer is the image we input, output layer is the recognition result. The hidden layer can be one or multi hidden layers. The result can be influenced by the number of hidden layer and the number of hidden layer nodes. This paper uses 2700 characters in 100 ID cards for validation set, test the accuracy in different number of hidden layers and different number of hidden layer nodes. The result was shown in table 2.

Table 2. The performance by different parameters.

Number of hidden layers	Number of hidden layer nodes.	correct /sum	accuracy
1	30	2591/2700	96.0%
	50	2622/2700	97.1%
	70	2635/2700	97.6%
2	30	2667/2700	98.78%
	50	2670/2700	98.89%
	70	2673/2700	99.00%
3	30	2687/2700	99.52%
	50	2692/2700	99.70%
	70	2690/2700	99.63%
4	30	2580/2700	95.56%
	50	2576/2700	95.41%
	70	2575/2700	95.37%

Analysis table data we can know that increasing the number of hidden layers and the number of hidden layer nodes does not necessarily gain a better accuracy. The results are best when the number of hidden layers is 3, the number of hidden layer nodes is 50 in the experiment. When the number of hidden layer is too much, over-fitting occurs frequently, resulting in reduced accuracy recognition. The number of hidden layer nodes can influence the result, but not greatly.

According to the results, select the appropriate number of hidden layers and number of hidden layer nodes in the process of recognition can improve the correct rate of recognition. When the number of hidden layers is 3, the number of hidden layer nodes is 50, the accuracy of ID card numbers recognition can reach 99.70%.

The speed of recognition can be influenced by complexity of the BP neural network. It takes more time if the network is more complex. This research uses 3 layers of hidden layers (50 nodes in each layers) BP neural network to recognition 100 ID cards in turn and records the time required by system clock in program. It take 4.57 seconds to recognition 100 ID cards, and the time for each ID card is within 0.05 seconds.

4 Conclusion

This paper introduces an algorithm of locating and recognition for the printed numbers. A new dataset named Printed Number-11 was created for printed numbers. In the process of printed numbers locating, FAST detector and image convolution are combined. In the process of printed numbers recognition, by adjusting parameters in BP neural network, the final accuracy is better than 99%, the time for each ID card is within 0.05 seconds.

Acknowledgments

Especial thanks to The Natural Science Foundation of China (NSFC) and Beijing Higher Education Young Elite Teacher Project for the financial support.

References

1. Ballard D.H.Generalizing. The Hough Transform to detect arbitrary shapes [J]. *Pattern Recognition* **13**, 111-122 (1981)
2. Rosten E, Drummond T. Machine learning for high-speed corner detection [M]. *Computer Vision-ECCV 2006*. Graz: Springer Berlin Heidelberg. 430-443 (2006)
3. Rowley H A, Baluja S, Kanade T.Neural network-based face detection [J]. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, **20**, 23-38 (1998)
4. Lienhart Process R, Maydt J.An extended set of haar-like features for rapid object detection [J]. *2002 International Conference on. IEEE*. **1**, 900-903 (2002)