HANDWRITTEN CHINESE CHARACTER RECOGNITION THROUGH A VIDEO CAMERA

Xiaoou Tang[1], Cheung Yung Chan[1], and Jianzhuang Liu[2]

[1]Department of Information Engineering
[2]Department of Electronic Engineering
The Chinese University of Hong Kong
Shatin, Hong Kong

ABSTRACT

In this paper, we propose a handwritten Chinese character recognition system using a video camera. The system combines the advantages of both online and offline approaches. It allows users to write on any regular paper just like using an off-line system. At the same time, using a video camera attached on the computer, the system can capture the stroke temporal information similar to an on-line system.

1. INTRODUCTION

Chinese characters are used by nearly one-quarter of the population in the world. However, several decades after the computer was introduced into the Chinese community, the input of Chinese characters into computers is still a difficult problem, primarily because the Chinese language is not alphabetic. Each Chinese character is a beautiful but complicated square graph composed of different strokes organized in a highly artistic and imaginative manner. Handwritten character recognition is an efficient alternative for Chinese character computer input. However, with over 5000 categories of characters, the Handwritten Chinese character data are complex, diverse, and deformable, thus are very challenge to be automatically recognized.

Handwritten character recognition has been studied for many years. Research has been divided on the basis of data input: optical (off-line) [1-3] and pen-based (on-line) [2-5]. Off-line systems usually use scanned character images as input data. Because of the difficulty in extracting reliable stroke information and the missing of temporal information, classification accuracy of most off-line systems remains too low for practical usage. Online-based systems on the other hand have attained high recognition rates for natural handwriting data, and many commercial systems are available. This is mostly because the stroke temporal information is recorded for each written word. However, one has to use an extra writing board as the input equipment. Writing on the slippery surface of the writing board is cumbersome and time consuming. The extra cost of the pen and board, as well as the inconvenience of using them greatly hindered the popularity of pen-based systems.

In this paper, we propose a new system that combines the advantages of both online and offline approaches. Since writing on paper is the most natural way for handwriting, we allow users to write on any regular paper just like using the off-line system. However, instead of waiting for a whole page to be written and then scanned into a computer, we use a video camera attached on the computer to capture a sequence of the character images while it is being written on the paper. Then using a set of video processing techniques, we extract time sorted individual pixels on the character strokes from the video sequence. Thus we can capture the stroke temporal information similar to the on-line system. Unlike the online system writing board that has to be purchased separately, most computers nowadays have a small video camera attached for net meeting type of application, therefore no additional special equipment is required for our system.

We describe the system in the following three sections. In section 2, we describe the basic system configuration and a stroke sequence extraction method. We then proceed to an improved approach and show experimental results in section 3. A brief summary is given in section 4.

2. SYSTEM CONFIGURATION

The basic system is described in Figure 1. As a user writes on a piece of paper, a video camera captures the evolving

0-7803-6297-7/00/$10.00 © 2000 IEEE

character continuously. A sample sequence of the captured images is shown in Figure 2 (a). These images are then converted into binary images through thresholding. As we can see from the binary images shown in Figure 2 (b), the background noise including the hand, pen and their shadows occupies significant sections of the binary image. Fortunately, after the character is finished and the pen moves away from the current character to the next one, the finished character is left alone. In the third step of processing, using this clean character image, we can mask out only the character part of each binary image in the video sequence. Figure 2 (c) shows the results of such a masking operation. Now, from the processed image sequence we can clearly see how each stroke is gradually written out. By computing the difference between neighbor images, we can locate both the time (frame number) and location of each written pixel, which give us the similar type of information that used to be available only to traditional online systems.

The approach is quite straightforward. However we encounter some difficulties in implementing the last step in the system, i.e. differenciation. As we can see from Figure 2 (c), because of the background noise, some pixels on the character will appear before their turns. For example, at the beginning of the sequence, instead of the first few pixels on the character, many noise pixels appear because of the pen and hand movement in the image. Simply computing the image difference cannot produce the intended pixel list. Even in the middle of the image sequence, we can also see some noise pixels appear at the tips of the stroke caused by the slight misalignment of images. Such misalignment of images will produce even more noisy pixels after we compute the image differences. In addition, to compute the difference between all neighbor images and later combine all the difference images are not trivial in computation. To derive a stable pixel sequence, we propose an improved approach in the next section.

3. PIXEL TRACING

Instead of computing the image differences, we first conduct a thinning operation on the last finished binary character to reduce the line width to one pixel. Then for each pixel on the thinned character, we look at a 3x3 neighborhood around the same location in each binary image in the video sequence. If any one value of the nine pixels is nonzero in a image we assign a switch value 1 to that image frame, otherwise a value 0 is assigned. Using a 3x3 neighborhood can avoid the problem of slight misalignment between images. Tracing through the video sequence from the first image to the last image where the final character is recorded, we obtain a binary switch line shown in Figure 3 for each pixel on the thinned character. To find the frame number that a pixel is first written, we trace from the end of the binary line in Figure 3. Since the pixel must be on the finished character, all the binary lines end with switch value one. As we trace back the frames along the binary switch line, sometimes the switch value will turn to zero and remain zero until the first frame at the beginning of the line, as shown in Figure 3 (a). In this case, we can clearly identify the frame (marked by a circle in Figure 3) at the turning point as the one in which the pixel first appears.

Sometimes, because of noise, a pixel may disappear from one or a few frames even after it is already written, as shown in Figure 3 (b). To get around such situations, we continue to check the next five frames. If they are all zeros, we confirm the point as the turning point, otherwise the search continues. In Figure 3 (b), we also see that the pixel appeared several times even before it is written. These apparently are due to the hand and pen shadows in the image. We sort the pixels on the character by their first appearance frame number and then stack the binary lines of all the pixels into a binary matrix according to the sort order. So the first row of the matrix is the binary switch line of the first pixel on the character, and the last row of the matrix is the binary line of the last pixel on the character. Displaying the binary switch matrix as a black and white image in Figure 4 (b) for characters in (a), we can clearly see the trace line (circles in the image) dividing the one region and zero region. Each monotonic increasing section of the trace line corresponds to one stroke in the character. Each horizontal section of the trace line represents a jump in frame numbers from one pixel to the next, thus corresponds to the transition period from one stroke to the next. Finally, we show the extracted spatial and temporal information of the characters in Figure 4 (c), where the x and y axis represent the location of each pixel, the z axis represents the first appearance frame number of the pixel. We can see that the frame number increases as a stroke is written in one direction, and the frame number jumps from one stroke to the next.

4. SUMMARY

We have demonstrated a character recognition system that combines the advantage of both on-line and off-line systems. Using a regular video camera, we can capture temporal information of written characters as they are written on a sheet of regular paper. Thus high accuracy on-line recognition algorithm can be used for recognition. As a new concept, there are still many issues need to be resolved before the system becomes of practical usage. We need to carefully study many parameter settings in the system as well as continuous writing of long paragraphs on paper.

5. REFERENCES

[1] T. H. Hildebrandt and W. Liu, "Optical recognition of handwritten Chinese characters: advances since 1980", Pattern Recognition, 26(2), 205_225, 1993.

[2] *Handbook of Character Recognition and Document Image Analysis*, Eds. H. Bunke and P. S. P. Wang, World Sci. Publishing Company, 1997.

[3] M. Uneda, "Advances in recognition methods for handwritten Kanji characters," IEICE Trans. Inf. & Syst., Vol. E79-D, No. 5, May, 1996.

[4] J. Liu, W. K. Cham and M. M. Y. Chang, "Online Chinese character recognition using attributed relational graph matching", IEE Proc. Vision Image Signal Process., 143(2), 125_131, 1996.

[5] J. Liu, W. K. Cham and M. M. Y. Chang, "On-line Chinese character recognition by incorporating human knowledge", Int. J. Uncertainty Fuzziness Knowledge-based System, 5(1), 13_29, 1997.

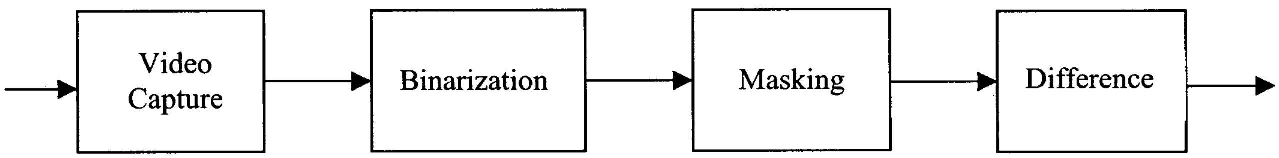
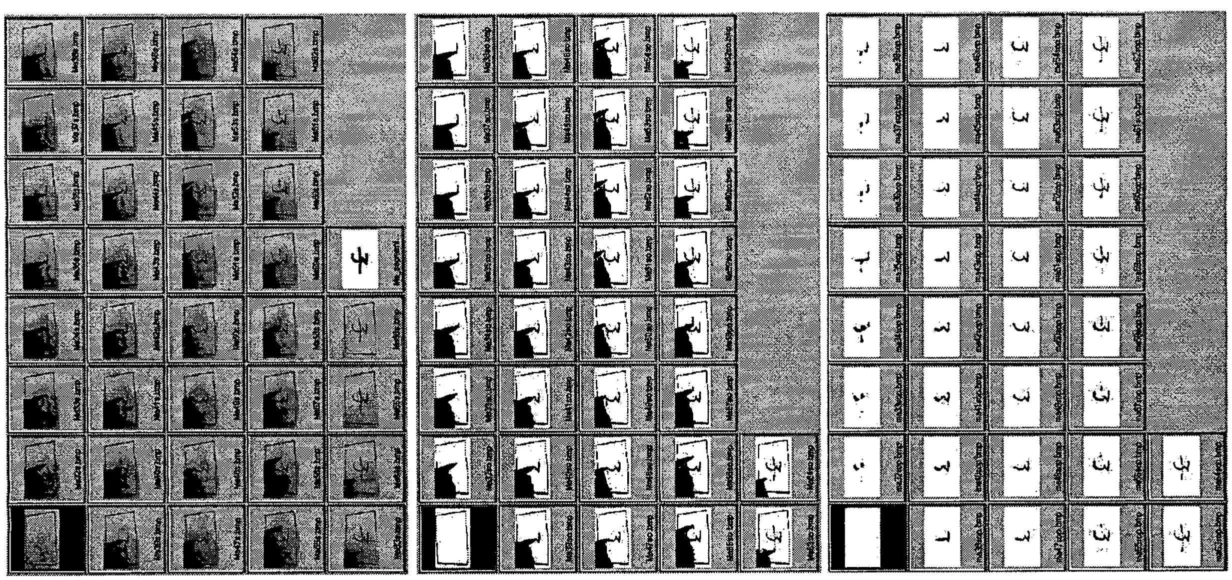


Figure 1: A general video handwritten character pixel sequence extraction system.



(a) (b) (c)

Figure 2: Video sequence processing example: (a) original video images; (b) binary images; (c) binary images masked by the finished character.

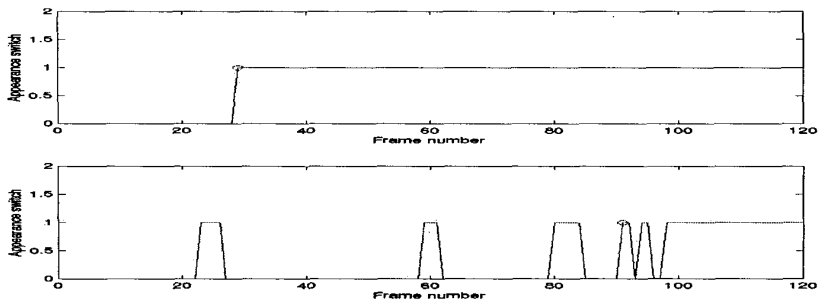


Figure 3: Sample binary switch lines of two pixels on the character. Symbol 'o' marks the frame number that the pixel first appears in the video.

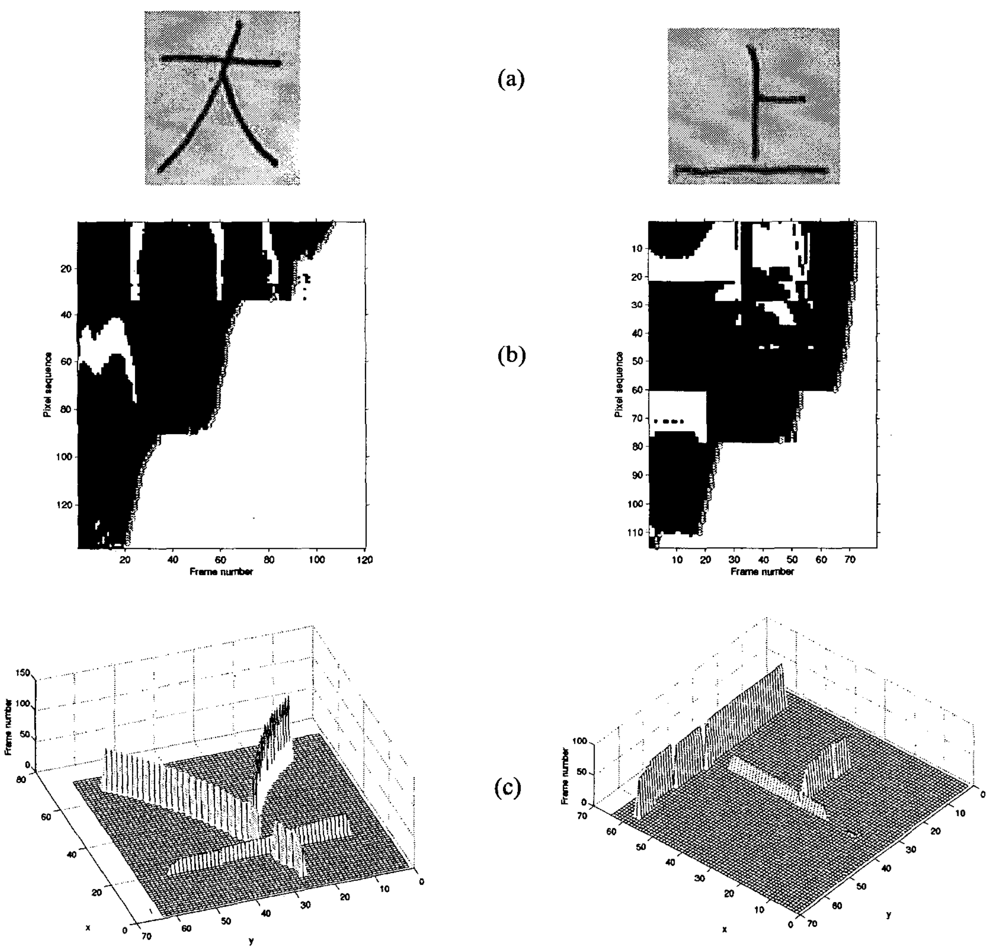


Figure 4: Character tracing results: (a) original images; (b) pixel appearance frame number ('o' marks) trace plot; (c) appearance frame number plots of character pixels.