SEGMENTATION METHOD IN CONTAINER AUTO RECOGNITION SYSTEM

Zhiwei He, Jilin Liu

Institute of Information and Communication Engineering, Zhejiang University
Hangzhou, China 310027

Abstract. In this paper, we introduce the segmentation processes in container auto recognition system. The segmentation method uses image processing.

I. Introduction

With the globalization of the economy, more and more goods are transported by container. As a result, thousands of containers need to be registered at shipsides and depots. If this registration process is done manually, it is not only prone to error but also too slow to meet the increasing volume of containers. If this process can be done automatically with the help of the computer, we can improve the efficiency rapidly. For this reason, we develop a system called Container auto Recognition System (CRS). This system use a camera to take the photo of containers, then we use some image processing methods to extract the container identity numbers, with a pattern recognition to recognize them in succession. Many methods have been proposed for character recognition[2,3], so the recognition algorithm is not the main purpose of this paper. We just concentrate on the most difficult part of the CRS——the location and segmentation of the individual characters which compose the identity numbers of a container. Some methods have been proposed to locate and segment the characters[1,2], in this paper, we introduce a new method and its process.

In order to achieve robustness and efficiency, we must consider the following issues in the real life. Firstly, although there are only three types of container in size, the painted container ID numbers can appear in a wide range of sizes, fonts and locations. The characters can be aligned in one, two, three horizontal lines or one vertical row. The order of the characters is important to the recognition result, which means we must get the right information of the align mode and extract the characters orderly. Secondly, the surface of a container may be accidented and the illumination in a real scene varies according to the time of the day and the changing weather, so the contrast of the grabbed container image may change rapidly. And the contrast between the characters and the background of the container may be affected by their colors. Thirdly, the grabbed image may be obscured by many noises such as mud, rust, peeling paint, etc. And there may be some other unwanted objects such as company logos and container weight information characters(see Fig. 1). A perfect system should get rid of all these interferences to locate and segment the characters correctly.

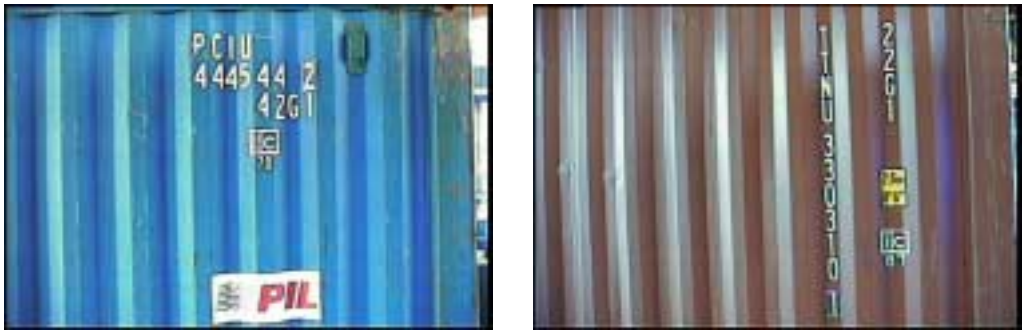

Fig. 1. Container pictures.

II. Process of location and segmentation

As we have mentioned, location and segmentation is the most difficult part of CRS, in some sense, it is also the most important part. In CRS, we do it in some successive steps.

1. Location

In order to reduce the influence of noises, we sample the gray-level image, which is 768×512 in size, every other line and every other column. The location here is just to cut out some useless parts of the container image. The ideal result is to reserve the part in the image that contains the container identity number and eliminate everything else. First, we differentiate the sampled image. As the characters in the

image are composed of both short horizontal and vertical line segments while the background is not, so in the differentiated image the up and down edge of the image which contains the ID number can be detected. This can be done as the following: firstly, from the horizontal and vertical projection of the differentiated image we can get some peaks and some valleys, we take the position of the upmost and downmost peak of the horizontal projection as the up and down boundary of the result location image. And in the same way, we can get the left and right boundary. Then with the relationships up=up/2, left=left/2, right=right×2, down=down×2 we can get the boundary in the original image. Then cut this rectangle out of the image we can get the result location image. Because of the influence of the noises, the location result is not accurate so we need to do the following step.

2. Binarization

This step plays an important role in CRS. Theoretically speaking, if you can find an effective and robust enough binarization algorithm, which can binarize the characters to white objects and the others to black, there is no difficulty to segment them. But as the real image is distorted by the reasons we mentioned before, we can not find a perfect algorithm. What we need to do is to find an algorithm, which can differ most of the characters from the background. We use the following binarization algorithm.

This method is called Gradient Binarization. We divide the located image horizontly and vertically to some small images. For each different small image, obtain the histogram of it and then we can get a threshold which is in the middle of the gray-level of the characters and the background. So we can binarize that part with the corresponding threshold, that is, if the gray-level of a pixel is greater than the threshold, binarize it to 255 and 0 if not. When each of the small image have been binarized, the whole container image is binarized also.

At last, filtering the binarized image we can get a good result.

3. Obtain align information

Get horizontal and vertical projections of the binarized image. We use this data to obtain the align information. If the align mode is horizontal, then the vertical projection has much more peaks than the horizontal one. On the other hand, if the horizontal projection has much more peaks than the vertical one, we may infer that the align mode is vertical. If there is no distinct difference between them we must use the convolution projections for further decision. If the align mode is horizontal, we must do horizontal segmentation first and then the vertical segmentation in the successive processes, otherwise vertical segmentation first and then horizontal segmentation.

The following two steps may change the order according to the align information we get in the last process, that is, if the align mode of the characters is horizontal we do horizontal segmentation first, and vertical segmentation first if the align mode is vertical.

a. The result of binarization b. The result of segmentation(first line)

Fig. 2. Results.

4. Horizontal segmentation

If the align mode is horizontal, the purpose of horizontal segmentation is to extract orderly each line of the ID number. It is also done with the binarized image. Firstly, do horizontal transition histogram. Filtering the transition histogram to smooth the histogram and then get the peaks and valleys of it Secondly, determine a variable local threshold to segment each line of the characters. This threshold is variable because the image is real and it needs to be adaptive. It is determined based on: the position of the highest peak and its nearest valley, the average height of the characters, height of the background between two lines of characters and so on. Thirdly, get the up and down boundary of each line of characters and then cut out and save them orderly.

If the align mode is vertical, the purpose of horizontal segmentation is to extract orderly each character from each result obtained in vertical segmentation process. This is also done with the help of histogram. For a single row of character, there is sure to be some black space between two characters and in the histogram this black space exhibit as a valley. So we can just use these valleys to extract each character. And of course the average height of a character is useful to determine whether a region contains a character. And if one row contains less than 4 characters, we infer that it is not a useful row to the ID numbers and we just throw it out.

5. Vertical segmentation

This is generally like the horizontal segmentation process. If the align mode is vertical, we need to do vertical segmentation first. Here, the selection of variable thresholds needs to be based on the average width of characters.

III. Results and Conclusion

All the module form the first version of an industrial system developed in our lab. This process has been applied on 132 all kind of container pictures. 118 are correctly segmented but 14 are not: 9 can not be segmented because of the strong glisten and 5 are “strongly” segmented(retained area cut a part of characters). The first issue can be solved through hardware. That means we can take some measures to compensate the strong glisten.

The segmentation method can be used on other ITS, for example, the plate auto-recognition system. Our future job is to develop a more robust system, which takes less executing time but have a highly right recognition rate.

References:

[1]: R.Mullot,C.Olivier :”Automatic extraction Method of container Identity Numbers and registration Plates of Cars”,1991 IEEE,pp1739-1744

[2]: Dr. John C.M.Lee: “Automatic Character Recognition for Moving and Stationary Vehicles and Containers in Real-life Images”,1999 IEEE,pp2824-2828

[3]: Ho. C. Lui, Chung M. Lee and Fang Cao:“Neural Network Application to Container Number Recognition”,1990 IEEE,pp190-195