Automatic Reading of Intermodal Containers ID Numbers (Ownership Codes)

Fernando Martín-Rodríguez[(1)], Xulio Fernández-Hermida[(1)], J.C. Torres-Barragáns[(1)].

[(1)] Departamento de Teoría de la Señal y Comunicaciones. Universidad de Vigo.
E.E.T. C/ Maxwell S/N. 36310 Vigo (Pontevedra).
Phone: +34-986-812151, Fax: +34-986-812116, E-mail: fmartin@tsc.uvigo.es, xulio@tsc.uvigo.es .

Abstract- This paper is about the design of an automated system to read the "Ownership Code" [1] of intermodal containers as they are being loaded down from the ship. The system used machine vision with cameras attached to the crane to get images of container to be processed. Usefulness of this system lies in the implementation of an early detection system for handling errors. Nowadays, it is a huge problem that crane operator picks up the wrong container. If we discover it late, when the ship has gone, it becomes almost a catastrophe for the handler. Nowadays, there are human operators checking manually the correctness of loading/unloading. This new system arrives to help them, generating alarms that could be issued before the crane releases its load. Human supervision will still be necessary but will become more effective.

Keywords: container, handling, machine vision, mathematical morphology, O.C.R.

Fig. 1. Example container photo, taken at the terminal.

I. INTRODUCTION

As we have commented at the abstract, our main aim here is the automatic reading of container ownership code. Cameras are to be installed in the handling cranes, so that we get as soon as possible images useful for recognizing the required number. With this number, we can check if that particular container was scheduled to be unloaded in our terminal and in the opposite case, we will issue an alarm for checking. If this indeed is an error, we can correct it before releasing the container from the crane (or, at least, shortly after). So this system gets an important efficiency improvement in the terminal work. We also get an important error protection measure.

This system was jointly designed by "University of Vigo" and "Tranformaciones Globales S.L." (Transglobal Inc.) that is the corporation responsible for container handling at Vigo terminal. Cameras were attached to Transglobal's cranes and test images belong to containers handled by them.

The main camera is now placed in the side part of a crane, capturing images like that in figure 1. An electronic signal is issued by the crane to trigger camera system. Camera system is built by a commercial camera and an Odroid [2] computer. When the Odroid receives the trigger signal, a HD (1920x1080) image is captured.

Crane also produces a TCP/IP connection that is used to inform the Odroid of the container size present in the image. Crane knows this information because of the position of its spreader (the part that claps the container). We are currently supporting the most frequent sizes: 20, 40 and 45 feet long.

II. IMAGE PREPROCESSING

As software knows camera position and also container size, a region of interest (ROI) can be defined into the image so that the desired number appears in the new (smaller) image and in this manner we remove much of the unwanted information of the whole HD capture.

In figure 2 (left) we can see the original image cropped to the defined ROI (defined for a 40 feet container from lateral camera). In the right we can see the same image where we have made a contrast enhancement via an automatic histogram stretching [3].

Fig. 2. ROI: original (left) and enhanced (right).

At this point, image is ready to be processed.

III. IMAGE SEGMENTATION

First, image is analyzed to determine if we have bright characters in a dark background or vice versa. This is performed through histogram analysis [3,4].

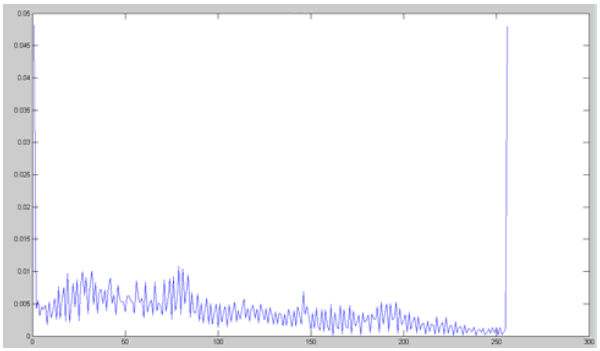

Fig. 3. Gray histogram of figure 2 (right).

Characters will be darker than background when accumulated sum of the gray histogram is greater below 128 (50% gray) than above (figure 3).

Afterwards, mathematical morphology [3] is used to: first, emphasize characters and, afterwards, to segment them. Detailing these processes, we have:

- Characters are emphasized with a bottom-hat (top-hat for bright characters) that detects thin objects in opposite background (figure 4).
- Afterwards, image is binarized. Instead of using the classic method (Otsu threshold [4]), we find the threshold that leaves 5% of the pixels above it (figure 5).
- Then image is cleaned erasing objects too big or too small to be characters. Remaining objects are converted into rectangles (equal to their bounding boxes) giving rise to a pre-segmentation (figure 6).

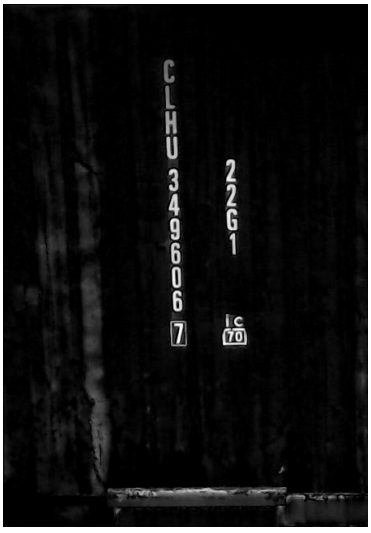

Fig. 4. Bottom-hat, white and thin objects get emphasized, rest of the image tends to get black.

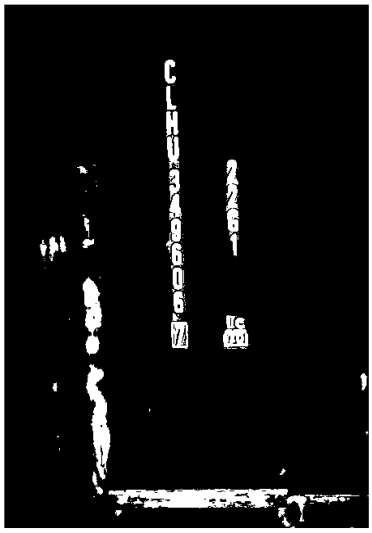

Fig. 5. Binarization: threshold is computed to yield 5% of pixels white.

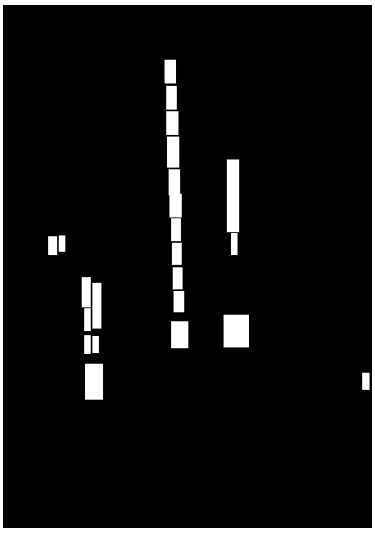

Fig. 6. Pre-segmentation: ROI's suspicious of being legitimate characters are selected to be checked by O.C.R.

The last image (figure 3, right) defines some rectangles that are "possible characters". Now, we must crop the corresponding sub-images from the enhanced ROI (figure 2, right) and a O.C.R. must determine if we have a true character and, in that case, recognize it. This process is aided by the fact that the true characters are always lined up (horizontally or vertically), making possible to remove some false ones now.

IV. CHARACTER RECOGNITION

Recognition is performed by an open source O.C.R. We use "Tesseract O.C.R." [5] with appropriate configuration. We provide Tesseract with individual, pre-segmented characters. Each is accompanied by an appropriate white list.

A white list can be used to instruct Tesseract about the "possible" characters. We can get much information from the ownership code format. Seeing the characters in, for example, figure 4, we see:

- A first component made always of 4 letters, last letter is U, J or Z. This component is not very important.
- A second component made of 6 digits. This one is the most important because it can be used to identify the container within the freight list (we have available the complete container list for the ship besides the list of those that need to be unloaded).
- A digit inside a box is a redundancy check. Important to check consistency of the result.
- A last component of 4 characters combining letters and numbers. This is only the container type and it is not relevant.

V. CONCLUSIONS AND FUTURE LINES

We have implemented an automatic reader system for ownership codes of intermodal containers. System is already functional and it is just now in the process of being integrated with the other terminal systems (cranes and their control systems). So, this last one is the main future line now.

ACKNOWLEDGEMENT

Authors want to acknowledge the "Centro para el Desarrollo Tecnológico Industrial" (CDTI, www.cdti.es) for their funding (project "MAR-TEC").

REFERENCES

[1] ISO 6346: standard defining ownership codes for intermodal containers.

[2] http://www.hardkernel.com/main/main.php

[3] R.C. González, R.E. Woods, "Digital image processing using MATLAB", Steven L. Eddins, Upper Saddle River: Gatesmark Publishing, cop. 2009.

[4] N. Otsu, "A Threshold Selection Method for Gray-Level Histograms", IEEE Transactions on System, Man and Cybernetics, vol. 9, N.1 (1979).

[5] https://code.google.com/p/tesseract-ocr/