A Hybrid Approach to Detection and Recognition of Dashboard Information in Real-time

Yue Tao, Yue Yong, Paul Craig
Department of Computer Science and Software Engineering,Xi'an Jiaotong-Liverpool University, Suzhou,China, 215000
E-mail: Yu.Tao16@student.xjtlu.edu.cn
E-mail: Yong.Yue@xjtlu.edu.cn
E-mail: Paul.Craig@xjtlu.edu.cn

Abstract: The digital industrial dashboards are developing rapidly. But only to display numbers in dashboards has not met industrial demands, more text information are added into them. Therefore, this also increases the workload for the instrument testers. Traditionally, for testing the stability and accuracy of the displays, most testing work relies on human observation. In addition, some errors would be caused by fatigue, carelessness and other uncertain factors during human observation. In order to improve the efficiency of instruments reading and recording, this paper proposes an end-to-end real-time instruments information location and recognition method based on OpenCV and a popular optical character engine (OCR) engine, called "tesseract". We also design an online testing strategy to verify what is the performance about this system. Finally, the result would be generated as a test report for the display tester.

Key Words: Industrial dashboards, end-to-end text detection and recognition, digital image OCR

1 INTRODUCTION

Nowadays, industrial digital dashboards are used widely in manufacturing and many electronic products. They belong to the field of human-computer interaction and can direct users to use machines or products properly by reflecting some key information. Although digital dashboards have such a huge significance, there is not an effective method to present how to automatic test them. Traditionally, testing instrument panels mainly depends on manually watching, but it is time-consuming. Moreover, there are also some special occasions, such as high temperature, high pressure, chemical metallurgy, mountain cliffs, nuclear radiation and other places which human body cannot adapt. Computer vision has a stronger ability to solve work on automatic instruments reading and recording than human eyes recognition. Widely using machines to replace human resource, not only this can save labor costs and ensure personal safety, but also improves work efficiency.

Many scholars have researched automatic meters reading, but mostly concentrated on numeral recognition based on computer vision. From different aspects of image preprocessing, skew correction, character segmentation and numeral identification, many advanced theories and methods are proposed. For instance, Lin et al. propose a method which use Canny edge detection and Hough transform to correct the tilt angle of instrument images collected by the camera[1].

Text objects localization and recognition on instrument usually needs a stable detection environment to ensure high-accuracy. However, mostly advanced algorithms in OCR field concentrate on extracting text information in an open scene(e.g.street view) and the highest precision is only up to 88% with a low recall rate of 66.4%[2, 3, 4]. Hence, we refer to some precocious algorithms related to OCR filed and add some optimization methods for obtaining a higher precision at such situation. In this paper, a system which is able to implement detection and recognition of key text objects from screens in real-time based on OpenCV and the open source OCR engine. Moreover, an online testing method is created to verify the correctness of system recognition by comparing with ground truths. Ground truths is produced by "Canoe", a signal simulating controller.

This paper also makes the fellowing contributions:

- We change textual localization method of he art-of-the-state algorithm, the Extremal Region (ER) detectors and superinduce a auto layout analysis for the instrument.
- After comparing ground truths, a result report would be generated for the display tester. According to this report, testers just inspect some specific samples with inconsistent recognition results.

The paper is organized as following section: section 2 provides an overview of the dashboard information detection and recognition system including framework introduction and some specific method illustration. Then, in the section 3, we analyze the experimental results and evaluate the performances of this system. Finally, the conclusion and future work are given in section4.

2 DASHBOARDS INFORAMTION DETECTION and RECOGNITION SYSTEM

2.1 The framework of this system

Based on many approaches on textual detection and recognition on industrial digital instruments, an integral characters extraction and recognition system should be mainly composed by image skew correction, image binarization, text localization and text recognition. However, we adopt an algorithm with well generalization to extract textual regions. Thus, image skew correction is removed at the beginning of the image preprocessing stage. This system is designed as the fellow chart and composed of five stages, as fellows (see in Figure 1):

- Image preprocessing: The input images are acquired from a web-camera and preprocessed as binarization.

- Text localization: This stage finds text object regions and groups those into information modules which will display fixed information like speed region, milage region and so on.

- Text recognition: In this stage, the textual modules mentioned in the second stage are recognized by using poplar OCR engine (e.g. Google Tesseract[5]).

- Result storiation and comparison: This stage the results produced in the 3rd stage will be written into a log file. The output from canoe vector is ground truth. The testers will compare these two files.

- Output final report: After comparing two files, the accuracy and other performance of system can be concluded and analyzed.

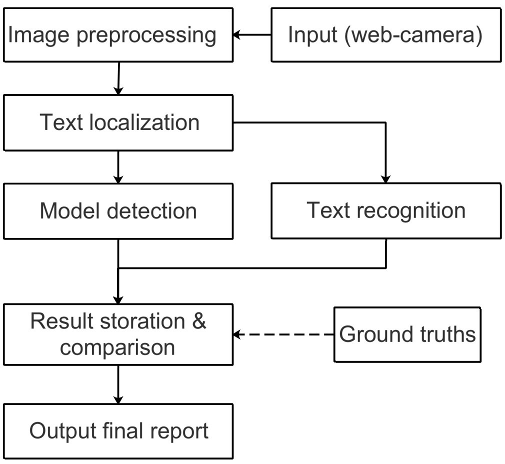


Figure 1: The architecture of text localization and recognition system for the industrial digital dashboard.

Furthermore, the experimental object is a MONO LCD digital dashboard with resolution of 320*240 supplied by Bosch company, a LCD dot matrix diagram display, called as "HMI". This HMI uses that sort of graphic LCDs to display the text information on running e-bikes, such as speed, driving mode, total distance traveled, duration and energy, as shown in Figure 2.

Figure 2: The example of a graphic LCD about HMI

2.2 Textual detection and localization

The main idea of ER detectors is to find extremal regions, then filter these by a sequential classifier which consists of a Real AdaBoost classifier[**?**] with decision trees and SVM classifier with RBF kernel[8]. In the last step, the remaining ERs are grouped into words and merged with same location in different channels. some detail information about ER detectors can be referred to [6]. In this part, we just introduce how this algorithm to do the text-line detection and an improved algorithm we design for the specific text-line localization.

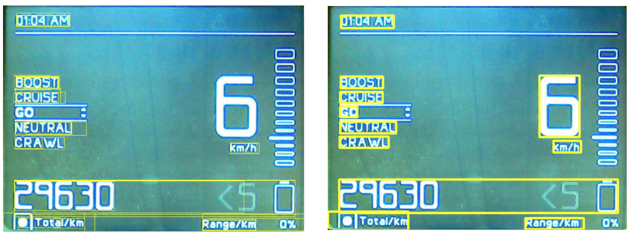
The method on text line detection proposed by Neumann et al. can be split into two phases. In the first phase, the character candidates in all channels which are detected and selected from the sequential classifier are searched by iteratively finding all region triplets. The process of finding region triplets is that to group two adjacent region r^1 and region r^2 into a pair, if r^2 is on the right side of r^1 and the distance between these two regions is measured with the distance of their centroids. Then, In the search of region pairs, a region pair(r^1, r^2) and another region pair (r^2, r^3) would be competed by a trained topological model and verified whether these two pairs can be combined into a triplets(r^1, r^2, r^3). The trained topological model is an AdaBoost classifier and features which would be trained in prophase are height ratio and region distance normalized by region width. In the second phase, every triplet is first transformed into a text line with 3 length and an initial bottom line direction b is prospected by Least Median of Square. This method is used to detect horizontal text line and the bounding box which contains the triplet is computed with its coordinates $\underline{x}$(left),$\bar{x}$(right) and $\bar{h}$(maximal height). In addition, in this project, we set d_{max} = 0.2 which is the threshold of distance between any two text lines. If the distance of any two triplets? text line are less d_{max}, these two triplets are merged together and a new bottom line direction and bounding box coordinates are all updated.

Although Neumann's text line formation method is low computational complexity and well-generalization with an agglomerative clustering approach. There is still a problem which is to only detect character regions length over 3. In the digital industrial dashboard, many words are displayed just in two-length format and some numbers are shown in single format. Thus, many character regions are not able to be extracted(see in Figure 3(a)). For solving such problem, we modify the method of the first stage, refer to Algorithm 1. After every stage of finding amalgamable regions, we back up the left ones. For example, $C1 = \{r_1 \in R \wedge r_2 \in P\}$, where C1mans the sets of one-length regions and P means the sets of region pairs. Then, $C2 = \{r \in P \wedge r \in T\}$ where T denotes the sets of region triplets. Finally, these backups with their similar region location are merged together. We should promise only one words occurs in one region. The reult is displayed in Figure 8(b).

Algorithm 1 The changed algorithm about text line formation in the first stage.

Require: The set of regions R (textual region candidates);
Ensure: A set of triplets T; A set of independent region pairs C_2; A set of back-up region pairs P; A set of isolated character regions C_1;

```
T ← ∅;
C2 ← ∅;
P ← ∅;
C1 ← ∅;
for r^1 ∈ R do
    for r^2 ∈ N(r^1) do
        if v({r^1, r^2}) = 0 then
            P ← {r^1, r^2};
            continue;
        end if
        for r^3 ∈ N(r^2) do
            if v({r^2, r^3}) = 0 then
                P ← {r^2, r^3};
                continue;
            end if
        end for
        C2 ← {P − p ⊊ {r^k, r^{k+1}, r^{k+2}} ∈ T};
    end for
    C1 ← {r ∈ R \ ⋃ N(r^n) : r^n ∈ R};
end for
```



(a) (b)

Figure 3: The contrast group of the orignial ER detectors and the changed one.

2.3 Model detection and tracking

Detecting and Tracking the selected mode (e.g. Figure 4) in the dashboard video is different from some traditional character recognition. Only using above method and Tesseract cannot resolve such problem. Hence, in the first step, we adopt the Sobel operator[9] to extract the outline on the x-direction projection. This is able to strengthen the horizontal lines edges on the instrument and helpful for Hough transform to find them. Then, some irrelevant detected lines would be eliminated by a filter rule while panel smearing disturbs the correctness of detection. Thus, we refer the idea of Markov chain[10] to predict the result of the smearing frame.

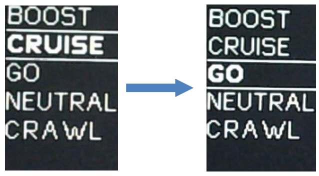


Figure 4: The mode region in HMI dashboard and the statue transfers from "cruise" to "go".

2.4 Character recognition training

By using "tesseract" to recognize some samples captured from a dashboard image, we find that some numbers with a special font (Bosch designs) cannot be identified correctly by "tesseract",shown as Figure 5. Thus, we collect a dataset of these special numbers pictures with a "tif" file format. This dataset consists of 40 imitated digital tube numbers and 20 normal numbers with different size and background color (whit-on-black and black-on-white) besides some notations (e.g. ":", "%" and ".") . Some examples in the dataset are displayed in Figure 5. The "tesseract" own training tool can trains this dataset to produce a datafile. Then, we use this datafile to verify whether it is useful to identify those numbers in the original dataset. The whole processing belongs to a supervised learning. Because the size of learning samples is small, The training set and the testing set are the same. The testing result is that accuracy rate of the imitated digital tube numbers is 100%, while some normal numbers like "6,8,9" are all identified as "8".

Figure 5: Some examples in the training samples.

2.5 Validation design

For checking whether this system works successfully. The online testing we design is to verify the correctness of results. Through comparing the ground truths which produced by canon vector, a simulating signal controller, we static the number of error outputs not equal to ground truths frame by frame. Finally, the accuracy of this algorithm can be got and a detailed report about which a frame is recognized falsely is also generated.

An intact processing of the online testing is illustrated in Figure 6. In order to test the display stability, the interval of characters variation is extended to 5 seconds. This can decrease the interference caused by the display smear. Thus, in the simulator which produce the ground truths, one phase generate only one record. However, this system processes the dashboard every 5 frames. To do this is to save more computational resource and reduce random recognition errors. Then, the result $x^*$with the highest occurrence probability is elected as the final result within this phase

Table 2: the results of accuracy on the whole system and different regions.

Video Clip (minute)	Global Accuracy	Model Accuracy	Charge Info Accuracy	Other Regions Accuracy
1	100%	100%	100%	100%
10	95.4%	95.4%	97.5%	99%
30	94.5%	94.5%	95%	99%
60	94.2%	94.2%	95%	99%

to contrast with its corresponding ground truth, see as formula (1). The false results are record to count the error rate in the current video phase.

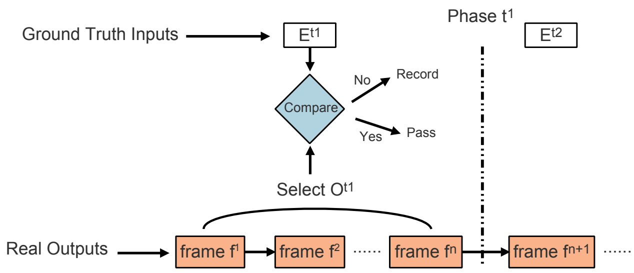


Figure 6: A flow chart of an on-line testing method.

$$x^* = \underset{x_i 1 \leq i \leq n}{\arg\min} \ \|\mathrm{P}(x_i|x_1, ..., x_n) \bullet \mathrm{P}(x_1, ..., x_n)\|. \quad (1)$$

3 PERFORMANCE EVALUATION

The experiments on the dashboards information detection and recognition system are mainly organized in two aspects, computational speed appraisal and accuracy testing respectively. Firstly, we note down precisely how long it takes at some specific stages. For some detail data, Table 1 is shown. We find that the phase of textual detection takes the longest time among all works in Table 1. In the textual recognition part, a parallel computing idea is used to save more time, only occupies 20ms.

Table 1: The computing time on different stages

Stage Work	Time(ms)
Textual Detection	330.60
Auto Layout Generation	12.56
Model Detection	45.71
Texual Recognition	20.78

About the experimental testing groups, we set 4 batch to verify the accuracy of this system, a batch of one minute video clips, ten minutes video clips, thirty minutes clips and one hour clips separately. Each model value to be tested is changed by a random rule which is settled by us. Table 2 shows the results of accuracy on the whole system and different regions. In overall, a global accuracy of this system decreases over time. The selected model detection still occur some errors, its accuracy only 95% at last. As for the normal character recognition, its accuracy is lowest in this table. Because the size of characters in this region is too small, Tesseract cannot identify this region and some outputs are empty. That means that the words segmentation of Tesseract is invalid.

4 CONCLUSION and FUTURE WORK

In this paper, we have proposed a system used to extract and recognize textual information on the industrial digital instrument. Based on the results of experiments, it is known that this system correspond with the speed of characters changing in the instrument. After an on-line testing work, the accuracy of the whole system reaches at 94.2% finally. For future study, we plan to design an algorithm for words segmentation instead of Tesseract for improving the accuracy of the charge region.

ACKNOWLEDGE

This work has been support by Bosch Automative Components Company in Suzhou. It is also a school-enterprise cooperative project. We acknowledge the department of PJ-MM-CN in this company for providing us experimental devices and scene. In addition, some valuable suggestion also make contributes to improve the progress of this project.

REFERENCES

[1] Jianping Lin and Yipeng Liao. Automatic recognition of digital meter readings based on opencv and lssvm. Microcomputer its Applications, 36(2):37-40, 2017.

[2] Alessandro Bissacco, Mark Cummins, Yuval Netzer, and Hartmut Neven. Photoocr: Reading text in uncontrolled conditions. In The IEEE International Conference on Computer Vision (ICCV), December 2013.

[3] Kai Wang, B. Babenko, and S. Belongie. End-to-end scene text recognition. In 2011 International Conference on Computer Vision, pages 1457-1464, Nov 2011.

[4] J. J. Lee, P. H. Lee, S. W. Lee, A. Yuille, and C. Koch. Adaboost for text detection in natural scene. In 2011 International Conference on Document Analysis and Recognition, pages 429-434, Sept 2011.

[5] R. Smith. An overview of the tesseract ocr engine. In Ninth International Conference on Document Analysis and Recognition (ICDAR 2007), volume 2, pages 629-633, Sept 2007.

[6] L. Neumann and J. Matas. Real-time scene text localization and recognition. In 2012 IEEE Conference on Computer Vision and Pattern Recognition, pages 3538-3545, June 2012.

[7] Chang Huang, Bo Wu, Haizhou AI, and Shihong Lao. Omnidirectional face detection based on real adaboost. In Image Processing, 2004. ICIP '04. 2004 International Conference on, volume 1, pages 593-596 Vol. 1, Oct 2004.

[8] K. R. Muller, S. Mika, G. Ratsch, K. Tsuda, and B. Scholkopf. An introduction to kernel-based learning algorithms. IEEE Transactions on Neural Networks, 12(2):181-201, Mar 2001.

[9] F. Li J. Kan. Fast Two-dimensional Maximum Entropy Threshold Segmention Method Based on Sobel Operator. In Computer Science, vol. 42 pp. 209-210+220 2015.

[10] J. R. Morrison P. R. Kumar. Linear programming performance bounds for Markov chains with polyhedrally translation invariant transition proabilities and applications to unreliable manufacturing systems and enhanced wafer fab models. In Proc. IMECE2002, 2002.