A Camera Based Digit Location and Recognition System for Garment Tracking

Anli Sun, Xi Wang, and Hong Wei
School of Systems Engineering, University of Reading, Reading, UK RG6 6AY
h.wei@reading.ac.uk

Abstract

Garment information tracking is required for clean room garment management. In this paper, we present a camera-based robust system with implementation of Optical Character Reconition (OCR) techniques to fulfill garment label recognition. In the system, a camera is used for image capturing; an adaptive thresholding algorithm is employed to generate binary images; Connected Component Labelling (CCL) is then adopted for object detection in the binary image as a part of finding the ROI (Region of Interest); Artificial Neural Networks (ANNs) with the BP (Back Propagation) learning algorithm are used for digit recognition; and finally the system is verified by a system database. The system has been tested. The results show that it is capable of coping with variance of lighting, digit twisting, background complexity, and font orientations. The system performance with association to the digit recognition rate has met the design requirement. It has achieved real-time and error-free garment information tracking during the testing.

1. Introduction

Automatic or semi-automatic *optical character recognition (OCR)* has been recognised as a successful technique and widely used in various applications for decades [1, 2]. Although OCR for typewritten text is considered as a solved problem generally, it is still challenging to develop a robust, efficient, and error free system, which not only deals with all noises and faded prints over time, but also has to distinguish between the target digits and other alphabetic characters in a same image in real-time manner. A unique system has to be designed and implemented to meet such requirements.

To the general purpose of OCR, many efforts have been made to solve problems of noise removal, text detection, and character recognition. Numerous algorithms have been developed to accomplish OCR with their own strengths and weakness, such as template matching, neural network, Gabor filters, wavelet based methods etc. [3, 4, 5]. Oliveira, *et. al.* developed a modular system using segmentation based verification strategies on automatic recognition of handwritten numerical strings for bank cheque authentication with an accuracy rate over 99% [2]. There are different types of noises or image defects (*i.e.* low contrast, blur, shading or out of focus) which affect performance of OCR as described in [3]. In order to improve robustness of a system to adapt wide variety of images, proper noise removal becomes important. Adaptive threshold algorithms can produce a global thresholding to convert color images with less complex background to binary images [5]. CCL has been used to detect text in images [4, 5, 6]. It may encounter difficulties with complex background where touching objects exist [7]. This requires additional contextual and structural features to be employed in the object search. ANNs have commonly been used as a recognition engine in OCR based systems [8]. Smagt compared three different neural networks on OCR and declared that ANNs with the BP learning algorithm had its capability of nonlinear projection and flexible network structure, and they were efficient and had a high recognition rate when the optimum number of layers and neurons was chosen against the number of characters to be recognised [9].

This paper presents a robust system for automatic digit location and recognition from real-time camera captured images as the purpose of clean room garment information tracking. The system employs a web camera as a sensor to capture images containing garment labels. The ROI consists of eight same font digits which are required to be recognised and recorded for garment tracking. In this paper, methodology used for image capturing, digit location, digit segmentation, recognition, and verification is discussed in Section 2. In Section 3, the experimental results are demonstrated. Section 4 concludes the paper and points out future work to optimise the system.

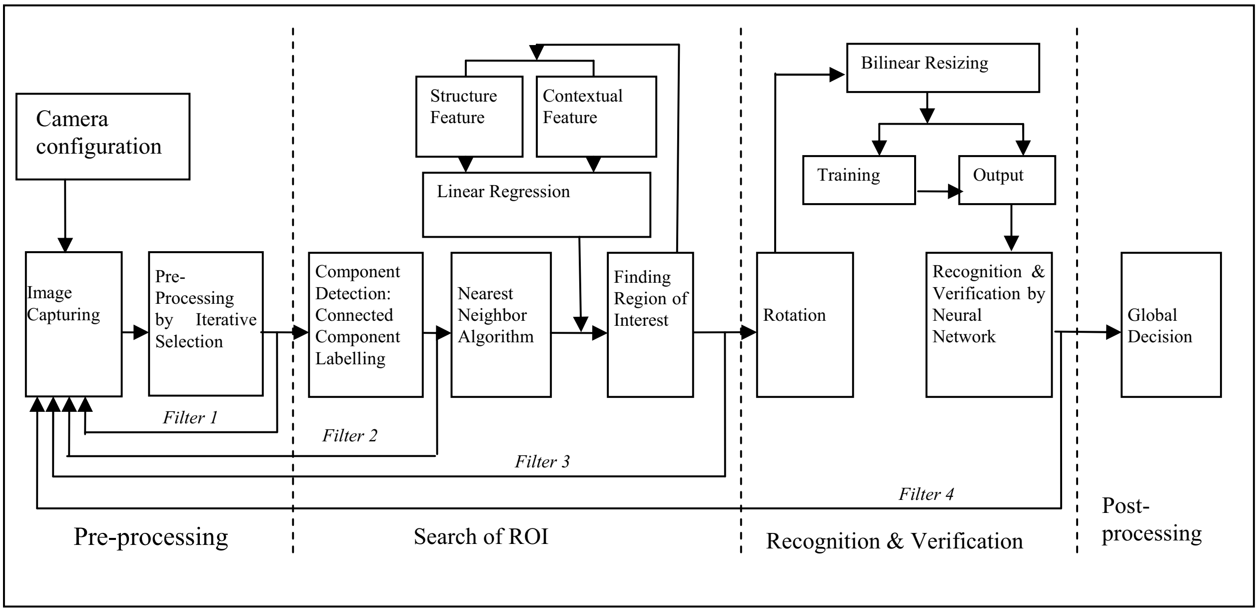


Figure 1. Schematic construction of the automatic digit location and recognition system

2. Methodology

The system layout is demonstrated in Figure 1. It consists of four parts: pre-processing, search of ROI, recognition and verification, and post-processing. In the pre-processing phase, a captured colour image is converted into a binary image using a proper thresholding value. Search of ROI is dedicated to processes of locating and segmentating the eight connected digits. The recognition and verification phase works with the neural network. The post-processing phase makes a decision on whether the processed digits should be accepted or rejected by the system and the database is accordingly updated. There are four filters involved in the system, which are used to control the capturing process. The details of the system are descibed as follows.

A. Image Capturing

The real-time image capturing process outputs images with 640x480 pixels in a speed of 30-50 fps (frames per second). It is controlled by a series of system filters (filtes 1, 2, 3 and 4). Filter 1 examines the overall grey level change between images with a timing interval to decide whether or not the captured image is a system targeted image, which should be passed to the next process stage. Filter 2 checks the number of objects produced from the CCL algorithm. If the number is not in a designated range (too many or too few objects are detected), the image under processing is abandoned, and new capturing is required. If the system does not find the ROI (eight connected digits), filter 3 will stop further process of the image and start the system from new images. When the system completes the recognition, a verification process is carried out with comparison of the recognised digits with garment labels stored in the database. If there is no match, filter 4 will trigger the image capturing process. The four filters play an important role in maintaining the system stability, and improving system efficiency and recognition accuracy.

B. Image pre-processing to generate binary images

Image pre-processing is the essential step before the other algorithms can be applied in the further processing phases in the system. The purpose of the pre-processing is to convert colour images captured by the web-cam to binary images. Adaptively finding the best single threshold value is vital to effectively separate text prints and image background. There are a number of methods to find a threshold, such as using edge pixels, iterative selection, grey-level histograms, entropy, fuzzy sets, minimum error thresholding *etc.* [10]. Among them, an iterative selection algorithm is a refining process on an initial guess at a threshold by consecutively passing through the image. The system adapted this algorithm. Starting with the initial estimate of the threshold T_0, the *kth* estimate of the threshold can be written as

$$T_k = \frac{\sum_{i=0}^{T_{k-1}} i \cdot h[i]}{2\sum_{i=0}^{T_{k-1}} h[i]} + \frac{\sum_{j=T_{k+1}+1}^{N} j \cdot h[j]}{2\sum_{j=T_{k-1}+1}^{N} h[j]} \tag{1}$$

where h is the histogram of the grey levels in the image, N is the maximum of greyscale value. When $T_k = T_{k+1}$, then T_k as the optimum threshold is found.

In practice, some garment labels have texture grids in the background and all printed texts are black. For example Figure 2(a) is a 640x480 colour image of garment barcode label captured by a web-cam. The iterative selection ensures that background grids were eliminated with the optimum threshold, as shown in Figure 2(b). In this case, the foreground and background was clearly separated with $T = 89$. Control of lighting conditions could be important to support the iterative selection algorithm in finding the optimum threshold.

(a) Original image **(b) After pre-processing**

Figure 2. Captured garment barcode label and its pre-processing result

Due to faded prints, lighting condition change, and inaccuracy of camera lens focus, the thresholding value generated by the iterative selection method could introduce problems of that the digits would be broken into small objects in confusing ROI search. To cope with such difficulties and increase system's robustness, the system takes the digit size as a feature to rule the final decision.

C. Detection of the ROI

As stated before, the ROI is 8 consecutive same font digits in an image outstanding of other prints (*e.g.* characters and barcode) in the same image. The process of ROI search operates at two steps: connected component labelling and object grouping based on the binary image generated in the pre-processing. Initially the CCL algorithm was used to search for all objects including all barcode, texts, digits, and other independent connected marks in the image**,** and then a nearest neighbour algorithm was applied to find the 8 nearest connected objects. There could be more than one object group with 8 consecutive objects in an image. New features of objects were used to decide the ROI. One of the features is that the expected ROI is an eight object group which is closest to the barcode with similar object size and similar gap between objects in the group. In this case, the position of barcode was used as a reference, and it has to be located firstly. The unique feature of these bars was identified in locating them in an image, *i.e.* they are a group of consecutive, highest and paralleled objects.

C.1. CCL algorithm

The CCL aims to label all pixels belonging to a same connected object. In the algorithm, a pixel P(x, y) can be labelled based on its 8 neighbours, which are defined as N8(P) ∈ {(x+1, y), (x-1, y), (x, y+1), (x, y+1), (x, y), (x+1, y+1), (x+1, y-1), (x-1, y+1) ,(x-1, y-1)}. The pixel 8-connectivity describes the relation among 8 neighbour pixels. Rules were followed to determine the pixel label based on a raster scan to the 8 neighbour pixels. At the end of processing, all pixels in the same object had a same label name. Figure 3 shows all the detected objects with colours in the image.

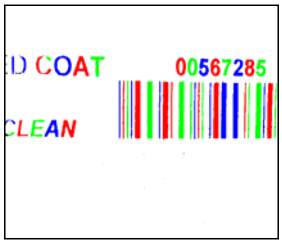


Figure 3. Different objects detected in the image

C.2. Nearest neighbour algorithm

The K-nearest-neighbour searching algorithm was employed to find the k nearest objects (in the case, the 8 consecutive digits) based on Euclidean distance from all the detected objects in the image [12]. The ROI was the combination of 8 nearest neighbour objects with similar size. If there were more than one object group with 8 consecutive objects, two further rules were applied to decide which group is the ROI. The first rule is that the Euclidean distance between the 8 objects must be similar; and the second rule exploited the barcode as a reference, *i.e.* the ROI has the nearest distance to the central line of the bars. Based on these rules, the ROI was located. Meanwhile, the 8 digits in the ROI were segmented and ready for recognition, as shown in Figure 4.

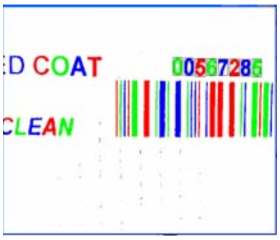


Figure 4. The ROI in the bounding boxes

C.3. Linear regression: a non-CCL method

The CCL based method can detect objects efficiently when the background is less complex. However it may fail when touching objects exist [13]. The background noises (*e.g.* embedded grids), image distortion (*e.g.* skew angles of fibres), lighting and shading condition, and faded prints cause the CCL based method failure in locating the ROI and segmenting digits. Under these circumstances, a complementary solution was used to reduce system errors and ensure that it is reliable and robust. Previous research has shown techniques to deal with such problems, for example a method treated texts as distinctive texture and used unsupervised clustering to classify each pixel as text or non text [14]. In our case, the special relationship (structural and contextual features) between the expected ROI and the barcode was taken into account. A simple geometrical method was designed, which is called the non-CCL method. A linear regression algorithm created a line through the centres of all bars. Moving the central line along the barcode in both dimensions, the gap between the digits and bars can be found as a unique feature to locate the ROI. The regression line also indicates the image orientation, with which the ROI can be rotated into the horizontal position. Figure 5 shows a line through the centres of bars. Image orientation was measured through the gradient of the line as 79.82 degree against the vertical coordinate axis in Figure 5. Eight digits were detected as shown in bounding boxes.

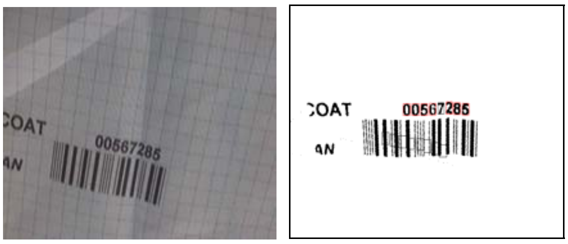
Figure 5. A line through centres of bars

D. Image rotation

The ROI found in images may have various orientations. Since the ANN recognition engine requires all digits in the horizontal to generate an array as inputs to the multilayer perceptron (MLP) neural network, the process of rotating the ROI is added into the system. The formula used is shown in Equation (2).

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos\partial & -\sin\partial \\ \sin\partial & \cos\partial \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \qquad (2)$$

where ∂ is the rotation angle when pixel $P(x,y)$ is rotated to $P(x', y')$ as the horizontal orientation. The example in Figure 6(a) is an original image captured by the camera. The orientation is random. After the rotation process, it is shown in Figure 6(b).



(a) Orginal Image **(b) after rotation**

Figure 6 Image rotation

With all eight digits are well segmented and refined, it is ready to be processed by the ANN recognition engine.

E. ANNs for digit recognition

ANNs have been applied for document analysis and recognition with a high recognition rate [15]. They process data by learning from training samples and have remarkable ability of coping with complicated and imprecise data with noises [16].

Our system adapted a one hidden layer fully connected MLP Neural Network with the BP learning algorithm. The MLP structure was configured with 6×8 input neurons and 10 output neurons which stand for 10 digits. To meet the input requirement for the MLP network, the bilinear interpolation algorithm was used to re-scale all digits in the size of 6x8 pixels to match the number of input neurons. The elements of resized 6×8 matrix were taken as 48 input neurons. The input values were bipolar (either 0s or 1s) which prepresent black and white in images, respectively. As one important factor for the whole training process, a training set which includes 10 groups of digits in various fonts was used. Among these training samples, digit twisting and distortion were taken into account.

Another vital parameter which could influence the training time and recognition accuracy is the number of hidden neurons in the hidden layer. Too many hidden neurons might exceed the optimal ANNs size due to the overfitting, which can lower the recognition ability, while too few hidden neurons may introduce large training errors [17]. Ideally, the least number of hidden neurons should be used, as it would be computational cost-effective, whilst still give the required performance. To find the optimum number, the network was trained and tested in turn by different numbers of hidden neurons, starting with five and adding another five each time.

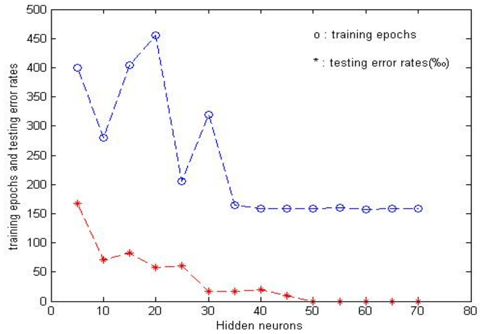


Figure 7. Epochs trend and testing errors against number of hidden neurons

The average training epoch values and testing error rates are demonstrated in Figure 7 based on 100 testing samples. It can be seen that circles which stand for training epochs converge to 160 when the number of hidden neurons was configured above 35. However, the error rate of recognition plotted by stars would not reduce to zero until 50 hidden neurons were used. Thus, to achieve both high recognition accuracy and fast training process, 50 hidden neurons were selected to construct the ANNs for the digit recognition purpose.

F. Verification

A verification mechanism was taken to ensure that the recognised digit sequence belongs to the unique counterpart in the system database. The system compares the recognition results with those strictly categorised in the database. If the database does not validate the recognised digit sequence, the system has to adjust its parameters to redo the whole process from the beginning until the digit sequence is accepted. Otherwise, an operator is involved in the process. It guarantees that the system is reliable and error free.

3. Experimental Evaluation

The system was designed to work in a commercial environment. It is not only required to be reliable, robust, and real-time, but more importantly is error-free. Therefore training and testing samples were carefully selected. Large amount of images captured from a variety of garments by a web camera (Logitech QuickCam Pro 4000, 1.3 Megapixel photo resolution, 640x480 digital video capture resolution) were used in testing. We present the testing results in three parts: detecting ROI (including detected by CCL and non-CCL), digit recognition, and verification. A variety of samples includes all orientations (0^o-360^o), different fonts, skewed images, embedded grids in the background, different colours of garments, lighting and shading, and faded prints. The recognition rate is defined as

$$\text{Recognition Rate} = \frac{\text{number of correctly recognised digits}}{\text{number of all testing digits}}$$

Table 1 shows that the detection of the ROI by the CCL was affected by sample skewing angles, embedded grids, lighting & shading, and faded prints. The non-CCL method was complementarily used in detecting the ROI where the CCL failed. The verification was performed to guarantee that all digit sequences output from the recognition system match those in the database. The error free for the system was achieved during the testing.

Table 1. System Performance Evaluation

Samples with variance of	No.	Detecting ROI		Recognition Rate %	Verified Rate %
		CCL %	Non-CCL %		
Orientations (0^0-360^0)	450	100	0	100	100
Fonts	250	100	0	100	100
Skew Angles	150	84	16	100	100
Emb. Grids	150	64	36	97.4	100
Light.& Shad.	150	76	24	98.2	100
Faded Print	150	64	36	99.6	100

4. Conclusion

The camera-based automatic digit location and recognition system presented in this paper has been designed and developed specifically for the clean room garment management purpose. A web-camera is used for image capturing with consideration of cost-effective. The experimental tests have shown that the camera quality is satisfactory to the system usage. Various algorithms are adapted in the system for image pre-processing, object detection, segmentation, as well as digit recognition. The testing results have demonstrated the robustness of the system. Since the project is in its mid-stage, there are time spaces for further investigation of image pre-processing, object segmentation, *etc.*. More algorithms may be developed to complement the current algorithms used in the system to cope with more complex situations. Hardware support will also be taken into account in future development. Optical filters would be exploited to remove majority of visible lights to improve performance under light reflection caused by plastic covers of garment. Control of environmental lighting condition can be achieved by using an enclosed operating box with a fixed setup for a camera and lighting or more 'adaptive' thresholding algorithms.

5. Acknowledgement

The project is sponsored by the KTP/Micronclean Ltd. with project number of J3106800/01. The authors would like to thank Micronclean (Newbury, UK) Ltd. for the data supply.

References

[1] Victor Wu, R.Manmatha, and Edward M. Riseman, "Textfinder: an automatic system to detect and recognize text in images", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 21, No. 11, pp1224-1229, 1999

[2] Luiz s. Oliveira, Robert Sabourin, Flavio Bortolozzi, and Ching Y. Suen, "Automatic recognition of handwritten numerical strings: A recognition and verification strategy", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol.24, No.11, pp1438-1454, 2002

[3] George Nagy, Thomas A. Nartkerb, and Stephen V. Ricec. "Optical character recognition: an illustrated guide to the frontier", *Proceedings of Document Recognition and Retrieval VII*, Vol.3967, pp58-69, San Jose, *SPIE*, January 2000

[4] O. Matan and C. Burges, "Recognizing overlapping hand-printed characters by centered-objects integrated segmentation and recognition", *Proc. Int'l Joint Conf. Neural Networks*, pp. 504-511, 1991.

[5] B. Yu and A. Jain. "A generic system for form dropout". *IEEE Transactions on Pattern Analysis and Machine Intelligence,* Vol. 18, pp1127-1134, 1996

[6] Anil K. Jain and Bin Yu. "Automatic text location in images and video frames", *Pattern Recognition*, Vol.31, No.12, pp2055-2076, 1998

[7] Huiping Li, David Doermann and Omid Kia, "Automatic text detection and tracking", *IEEE Transaction on Image Processing*, Vol.9, No.1, pp147-156, 2000

[8] Jie Xi, Xian-Sheng Hua, Xiang-Rong Chen, Liu Wenyin, and Hong-Jiang Zhang, "A Video Text Detection and Recognition System", *Proc. of IEEE International Conference on Multimedia and Expo* (ICME 2001), August 22-25, Tokyo, Japan, pp1080-1083. 2001

[9] P. Patrick van der Smagt, "A comparative study of neural network algorithms applied to optical Character recognition", *Proc. of International Conference on Industrial and Engineering Applications of Artificial Intelligence and Expert Systems*, pp1037-1044, 1990

[10] J. R. Parker, *Algorithms for Image Processing and Computer Vision*, John Wiley & Sons Inc, New York, 1996

[11] Rosenfeld, A., and Pfaltz, J. L. Sequential "Operations in digital picture processing", *J. ACM*, Vol.13, No.4, pp471-494, 1966

[12] Ting Liu, Andrew W. Moore, Alexander Gray, and Ke Yang. "An investigation of practical approximate nearest neighbour algorithms", *Proc. of Conference on Knowledge Discovery in Data*, pp 629-634, 2004

[13] Chiou-Shann Fuh, Shun-Wen Cho, and Kai Essig, "Hierarchical colour image region segmentation for content-based image retrieval system", *IEEE Transactions on Image Processing*, Vol. 9, No. 1, pp156-162, 2000

[14] A. K. Jain and S. Bhattacharjee, "Text segmentation using Gabor filters for automatic document processing", *Mach. Vis. Appl.*, Vol.5, pp169-184, 1992

[15] S. Marini, M. Gori, and G. Soda, "Artificial neural networks for document analysis and recognition", *IEEE Transactions on Pattern Analysis and Machine Intelligence* Vol.27, No.1, pp23-35, 2005

[16] P. Picton, *Neural Networks, 2nd edition,* PALGRAVE, New York, 2000

[17] I. V. Tetko, D. J. Livingstone, and A. I. Luik, "Neural network studies 1. comparison of overfitting and overtraining", *J. Chem. Inf. Comput. Sci.,* Vol.35, pp826-833, 1995