Efficient Barcode Localization Method for Low-Quality Images

Xiang Pan
Key Laboratory for Biomedical Engineering of Ministry of Education, Zhejiang University, Hangzhou, China
xiangpan0111@zju.edu.cn

Dong Li
Key Laboratory for Biomedical Engineering of Ministry of Education, Zhejiang University, Hangzhou, China
21715042@zju.edu.cn

Weijia Wu
Key Laboratory for Biomedical Engineering of Ministry of Education, Zhejiang University, Hangzhou, China
wwj123@zju.edu.cn

Hong Zhou*
Key Laboratory for Biomedical Engineering of Ministry of Education, Zhejiang University, Hangzhou, China
zhouh@mail.bme.zju.edu.cn

ABSTRACT
Barcode technology is widely applied to industrial automatic identification field for its low cost and high reliability as well as the fast real-time performance. Due to the complexity of low-light, rotation and blur in the industrial field, several barcode localization approaches which are superior in speed or accuracy fail to accurately locate and even detect the barcode. This paper proposes a real-time barcode approach that can effectively cope with the above problems when dealing with low-quality images. First, we rely on the gradient information of the pixels to obtain both orientation map and magnitude map. Then, we use the Shannon entropy theorem to get a salient map for the sake of segmenting salient patches of the high score. Later, we utilize the smoothing filter to remove the noise and connect the salient patches to form the barcode candidate BLOBs (Binary Large OBject). Finally, the correct barcode is selected from the above candidate BLOBs with a covariance matrix. We obtained 500 experimental images covering the conditions of reflection, rotation, and low illumination from the industrial site. The experimental results based on the dataset show that our method exceeds significantly the other three advanced methods in accuracy.

CCS Concepts
• Computing methodologies → Object detection

Keywords
Barcode; localization; gradient information; covariance matrix

1. INTRODUCTION
As modernization of industrial fields booms such as logistics tracking and warehousing [1], barcode technology has become an indispensable part and there is a tendency that more precise localization of barcode is required when we need to know the information carried by the objects in conveyor belts and automatic production [2, 3]. However, since the lens is fixed focus and its resolution is low, the images acquired by the camera are more likely low-quality in practical. What's worse, other conditions such as motion blur, noise, and reflection may often occur, and the object carrying barcode information is arbitrarily placed, thus it will pose a great challenge for barcode localization. In fact, all existing image-based barcode localization methods are affected to varying degrees when dealing with low-quality images. Of course, a polarizer can be used on the lens, which will reduce the effect of the reflection on the localization to a certain extent, but it does not have much effect. Thus, a efficient barcode localization method for low-quality images taken in poor conditions appears to be meaning for logistics industry.

ICGSP '19, June 1–3, 2019, Hong Kong, Hong Kong

ACM ISBN 978-1-4503-7146-9/19/06…$15.00
https://doi.org/10.1145/3338472.3338474

2. RELATED WORKS
Barcode localization has been researched and optimized for many years. Previous work has mainly solved one of the specific problems such as ambiguity [4], so it can only be robust in specific situations. Certainly, there are other algorithms such as Sörös's method [5] that can cope with most of the above when processing high-quality images, but fail to do so with low-quality images. These methods are mainly based on the spatial domain and the frequency domain. Spatial domain research includes Hough Transforms [6, 7, 8], orientation histograms [9, 10] and morphological operators [2, 3], etc., while frequency domain contains Gabor filters [11, 12], wavelet transform [13], etc. Gallo [14] is a spatial domain method that obtains the gradient map of the image by calculating the difference between the horizontal derivative and the vertical derivative of each pixel. After smoothing the gradient map, the author uses Otsu's method [15] to convert it to a binary image and the BLOB, which maximizes the local window in the gradient map is considered to be the region where the barcode is located. However this method does not work for slanted barcodes. Oktem [13] uses wavelet transform to address the problem of segmentation and texture extraction, and utilizes wavelet transform and morphological operations to locate the position of the barcode, but this method is complicated to implement. Katona [2] first removes noise and enhances edges through Gaussian kernels, then chooses the candidate regions and keeps the possible regions respectively by depending on bottom-hat filtering and distance mapping. Finally, the author uses the square structuring element and the morphological method to

remove the high-density text area around the barcode, but the method leads to more noise and results in a decrease in accuracy when the number of barcodes in the image increase. Sörös [5] proposes an algorithm based on covariance matrix. The edge detector derived from the covariance matrix is insensitive to illumination and rotation, while sensitive to scale size. Therefore, the author increases the scale invariance by repeating the calculation of the structure matrix on larger and larger image blocks and uses the saturation in the HSV to increase the robustness to the fuzzy barcode afterwards, but the biggest weakness is much time-consuming. Bodnár [16] combines probabilistic Hough transforms, distance transforms, contrast measuring, local clustering, and minimum-maximum morphological operations into an integrated detector to better locate barcodes. The aggregated weights can be obtained by calculating only one feature to maximize the recall value, but the recall rate is not high when confronting with low-quality images. Kutiyanawala [12] uses multi-channel Gabor filtering technique to extract directional textures, the author then utilizes the histogram to analyze the line segment pairs to locate the barcode region, but it is time-consuming.

To address the problems existing in the methods above, this paper proposes an algorithm that can effectively locate the barcode while dealing with undesired conditions such as rotation, different scale and blur, which is more suitable for barcode localization of low quality images and its resolution time is within the average. Our algorithm is based on the visual cognition of barcodes of texture and shape. The Shannon entropy theorem is used to determine multiple patches with low entropy, then several patches with the same dominant orientation component are grouped into candidate BLOBs. Finally, we construct a covariance matrix to determine the BLOB that contains the barcode.

The remainder of the paper is organized as follows. Section 2 details the approach presented in this paper. Section 3 shows the experimental results of the proposed method and the other three methods on the public dataset and the private dataset. Section 4 presents the summary and future work.

3. PROPOSED METHOD

This section describes a barcode localization algorithm suitable for low-quality images. The algorithm consists of image pre-processing, salient patches detection, candidate BLOBs generation, and barcode BLOB verification. First, we generate the global orientation histogram and magnitude histogram of the image through the Sobel operator. Second, we compute each patch of the image with the Shannon entropy theorem and select the significant patches with low entropy. Later, we utilize the smoothing filter to remove the noise in the image and generate some barcode BLOBs by connecting the patches whose entropy is low with the same dominant orientation. Finally, we construct a covariance matrix of the gradient vectors in each small window of each BLOB and apply the canonical correlation to determine the BLOB whose edge property is highly strong and the most directional components are in the unidirectionally varying region (UNIVAR).

3.1 Pre-Processing

The most notable feature of barcode is that it contains many parallel stripes in the same direction, edge extraction does not need to consider multi-channel information of the image. Therefore, we first convert the original image I into a grayscale image G, which can greatly reduce the computational cost. Figure 1 shows a grayscale image of an express with a barcode.

Figure 1. An express with a barcode.

To define the edges in the image, we use the Sobel operator to obtain the global gradient map of the image and get orientation and magnitude of the pixel p by the following equations (1) and (2). The orientation map O and the magnitude map M are respectively generated by calculating the overall pixels in the image (see Figure 2). The dark pixels in the orientation map are useful orientation data while the light pixels are useless data. The light pixels on the magnitude map will be used for further computational processing while the dark pixels will be discarded. In order to remove some irrelevant gradient information, only the pixels whose gradient value exceed the threshold are considered.

$$o(p)=\begin{cases}\arctan\left(\frac{|\nabla I_y(p)|}{|\nabla I_x(p)|}\right) & if\ |\nabla I(p)|\geq T_{mag}\\ 0 & else\end{cases} \quad (1)$$

$$m(p)=\begin{cases}|\nabla I(p)| & if\ |\nabla I(p)|\geq T_{mag}\\ 0 & else\end{cases} \quad (2)$$

Where $\nabla I_x(p)$ and $\nabla I_y(p)$ are respectively the gradients of the pixel p in the x and y coordinates, and $|\nabla I(p)|$ can be determined with $|\nabla I(p)|=\sqrt{|\nabla I_y(p)|^2+|\nabla I_y(p)|^2}$. To be able to detect the barcode within an acceptable resolution, we divide all the pixels into N_0 bins covering $[0,180°)$ degrees to generate a global orientation histogram h_G which derives from the orientation map O, the angular resolution sets to 10 degrees. In order to select the principle orientation components from h_G, we count the number of pixels of each orientation component in the histogram, the orientation components exceeding 40% of the maximum orientation component are marked as the strong orientations o_s, otherwise they are regarded as the weak orientations o_w.

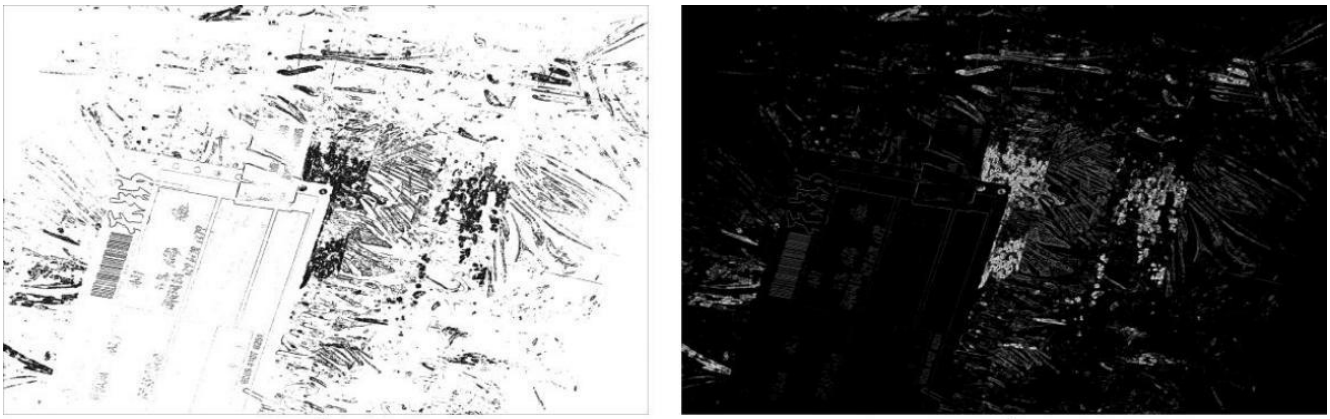


Figure 2. Represents the orientation map and magnitude map.

(a) the orientation map O of the image G (b) the magnitude map M of the image G

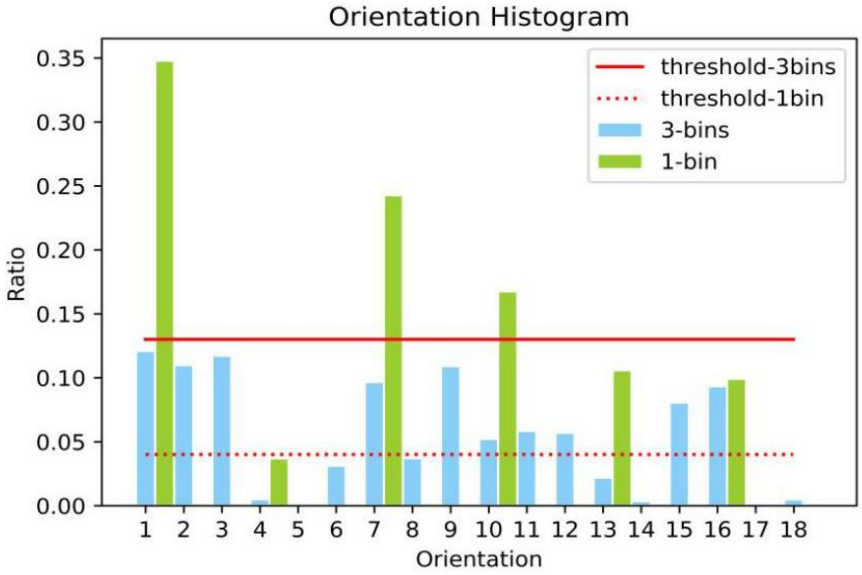


Figure 3. The azure bars represent the orientation components only with a bin, the orientation components exceeding the red (dotted) line are considered strong orientations, otherwise as weak orientations. The yellow-green bars represents orientation components with three consecutive bins, and the orientation components exceeding the red (solid) line are the strong orientations while the orientation component lower than the red line are the weak orientations.

Subsequent experiments denote that the test results are not satisfying based on the statistics of the count of pixels of the orientation component covering a single bin. Hence, we make each orientation component cover three consecutive bins in linear mapping in this paper. The details can be seen in Figure 3. The experimental results show that the computation time can be reduced and the detection accuracy is greatly improved both for low resolution and high resolution images.

3.2 Salient Patches Detection

In order to detect possible barcode regions, we use a voting scheme. The orientation map is divided into patches that do not overlap each other, and the Shannon entropy in each patch is calculated by the method of Chang [17]. Concretely, we first calculate the largest orientation component in each patch and then get the absolute difference between each orientation component and the maximum orientation component. Afterwards, we sum the value of Shannon entropy for each patch in the orientation map O. The equation (3) is given as followed.

$$Sum_{entropy} = \begin{cases} \sum_{i=0}^{N_0-1} | \mathrm{h}_i - h_{\max} | & if\ h_{\max} \in O_s \\ 0 & else \end{cases} \quad (3)$$

Where the $h_{\max}$ represents the maximum bins in the modified histogram in a patch. Intuitively, if the entire patch is within the barcode region, the value of Shannon entropy is relatively low. However, when the patch are full of the weak orientation, its value is low as well. Therefore, we only focus on the patch whose maximum orientation belonging to the strong orientaions. After that, we normalize the value of average Shannon entropy of each patch, the patches are considered not to be within the barcode region if the value of average Shannon entropy of the patch is below the threshold $\mathrm{T}_{\mathrm{entropy}}$, which is used to set the saliency patchs with high saliency scores. By the equation (4), we generate the salient map S.

$$S(f)=\begin{cases} Sum_{\text{entropy}} & if\ Ave_{entropy} > T_{\text{entropy}} \\ 0 & \text{else} \end{cases} \quad (4)$$

Where $Ave_{entropy}$ denotes the average Shannon entropy. In our experiments, T_{entropy} is set to 0.3 for the low-quality images.

3.3 Candidate BLOBs Generation

There are multiple patches with non-zero in the salient map S. Considering that the pixels of the barcode share the same orientation component, we need to connect the patches with the same orientation components to form the barcode candidate BLOBs. To eliminate the noise, we use a filter to smooth the salient map. Next, we use Ostu's method [15] to convert map S into binary map B and connect multiple patches into candidate BLOBs with the same orientation components. Figure 4 shows an image with barcode candidate BLOBs, which is derived from the image in Figure 1.

Figure 4. The image with barcode candidate BLOBs.

3.4 Barcode Candidate BLOB Verification

In order to reduce the running time, we remove the small size BLOBs. A threshold based on the image size is set in the experiment. When the image size is higher than 800*800, the threshold is set to 1/2 of the largest BLOB, otherwise it is set to 1/4 [2]. Figure 5 shows an image with the remaining barcode candidate BLOBs.

According to Harris's [18] and Ando's [19] researches, a covariance matrix can be used to categorize an image field into three regions: a unidirectionally varying region (UNIVAR), an omnidirectionally varying region (OMNIVAR), and a nonvarying region. For our purpose, we focus on the unidirectionally varying region.

In order to apply the covariance matrix we defined a small sliding window to analyze each candidate BLOB, then we apply the UNIVAR Detector to categorize that window. The definition of UNIVAR Detector is shown in equation (5).

Since the sliding window is binary and rectangular, the response of detector is noisy. Therefore, we apply a seven by seven Gaussian filter window and its standard deviation is set to one in order to reduce noise. When the value of reaches 1, it indicates that the grayness value changes in one dimension. Depending on the size of the BLOB, the sliding window can be swift "n" number of times inside the BLOB. As a result, we count the number of equal to one and divide it by the number of times we applied the sliding window, obtaining a success rate. The BLOB containing the barcode will have the highest success rate.

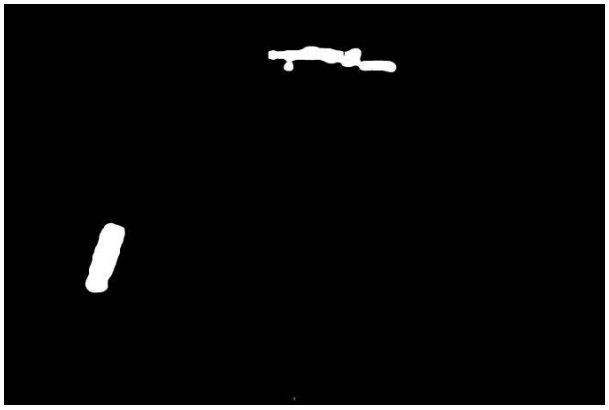


Figure 5. The image with barcode BLOBs after removing small size BLOBs.

$$P_{\text{eg}}=\frac{(S_{\text{xx}}-S_{yy})^2+4S_{\text{xy}}{}^2}{(S_{\text{xx}}+S_{yy})^2} \quad (5)$$

where

$$<\begin{bmatrix} f_x \\ f_y \end{bmatrix}[f_x \quad f_y]>=\begin{bmatrix} <f_x^2> & <f_xf_y> \\ <f_xf_y> & <f_x^2> \end{bmatrix}=\begin{bmatrix} S_{xx} & S_{xy} \\ S_{xy} & S_{yy} \end{bmatrix}$$

and $<\bullet>$ stands for $\iint_{\tau} \bullet\, dxdy$.

To remove the $P_{\text{eg}}=\frac{0}{0}$ condition, we add a small constant named const in the denominator, as shown in equation (6).

$$P_{\text{eg}}=\frac{(S_{\text{xx}}-S_{yy})^2+4S_{\text{xy}}{}^2}{(S_{\text{xx}}+S_{yy})^2+\text{const}} \quad (6)$$

Due to the low quality of the image, there are many edges that are not sharp enough, and the value of may return a value other than one. Hence, we set a threshold, so that if is greater than one, it is considered as edge whose grayness varies one-dimensionally. The equation is given in (7).

$$E(f)=\begin{cases} \text{edge} & if \quad P_{EG} > T_{\text{eg}} \\ not\ edge & elif \quad P_{EG} \le T_{\text{eg}} \end{cases} \quad (7)$$

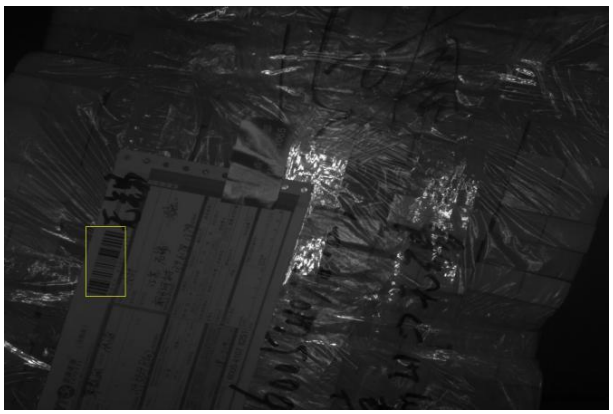


Figure 6. Barcode localization result of our algorithm drawn in the yellow bounding box.

As mentioned above, the BLOB with the highest success ratio is elected as the barcode BLOB. Figure 6 shows the barcode localization result of our algorithm.

4. EXPERIMENTS

In order to implement and test the proposed algorithm, we employed the Opencv framework inside a C++ development environment, running on a hardware PC configured with 12G RAM, an Intel Core i3-4170 processor at 3.7GHz of speed clock. The purpose of this experiment is to compare the accuracy and speed response of our method against the other three algorithms respectively proposed by Yun [10], Sörös [5] and Gallo [14].

4.1 Dataset

In order to measure the robustness of the above algorithms of barcode localization, we selected images from different sources and grouped them into several test datasets. The test datasets consisted of two parts.

First, we randomly selected 900 images with a resolution of 640*480 from Muenster BarcodeDB source created by Wachenfeld et al. [20], which can be of public access and name this dataset as Dataset_900.

Considering that Gallo's method [14] only had a good positioning effect on vertically placed barcodes and the barcodes placed in any other orientation could hardly be correctly located, we randomly chose 600 images from the same source with a resolution of 640*480 with a vertical oriented barcode and named the dataset as Dataset_600.

The images from the last dataset came from STO industrial site, considered as private access, the images were captured under low light at night and its resolution was 3072*2048. We randomly chose 500 test images and named the dataset as Dataset_500.

It is worth to say that all of the above images contain only one barcode.

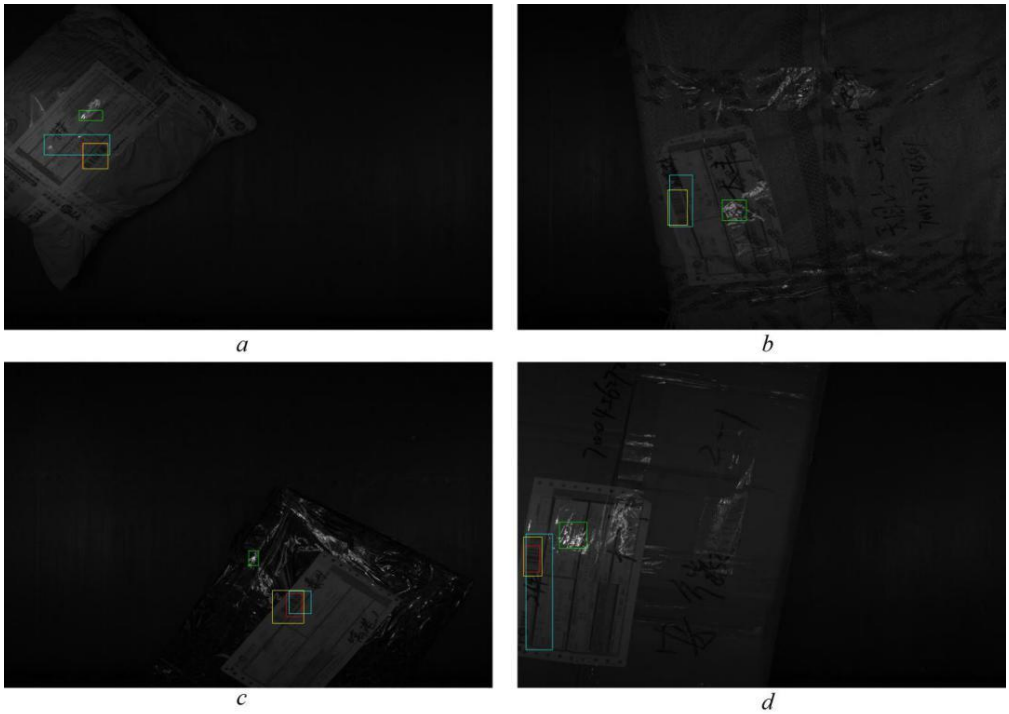


Figure 7. The azure bars represent the orientation components only with a bin, the orientation components exceeding the red (dotted) line are considered strong orientations, otherwise as weak orientations. The yellow-green bars represents orientation components with three consecutive bins, and the orientation components exceeding the red (solid) line are the strong orientations while the orientation components lower than the red line are the weak orientations.

4.2 True Positive and Accuracy

We manually labeled all the test images and defined the region drawn in the bounding box as the ground truth. To measure the accuracy of the localization, we used the Jaccard similarity coefficient in equation (8) to estimate the overlap between the actual bounding box and the bounding box generated by the above four algorithms.

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|} \tag{8}$$

Where A represented the Ground Truth we draw in the image and B stood for the bounding box of the detection region drawn by the algorithms.

In addition, we supposed that the value of Jaccard similarity coefficient exceeding 0.6 indicated a good match, concretely, which means that barcode is correctly located and is considered as True Positive. We tested in the above three datasets, the True Positive and accuracy are respectively shown in Table 1 and Table 2. As can be seen from Table 2, on the Dataset_900 and Dataset_600 dataset, the barcode localization accuracy of the above four algorithms is all above 80%, and the difference between them is not distinct. But on the Dataset_500 dataset, our method has more obvious localization accuracy than the other algorithms. Figure 7 shows the localization of our method and the other three methods.

The three line graphs in Figure 8 summarize the simulation results considering the Jaccard similarity coefficient ranging from 0.1 to

0.9 for each dataset. In this way we can more easily evaluate the performance of each method for any given threshold.

Intuitively, we can conclude that, for high-quality images (a) and (b) in Figure 8, there is no apparent advantage of our method in comparison with the other three. However, for low-quality images (c) in Figure 8, it can be seen that our method outperforms.

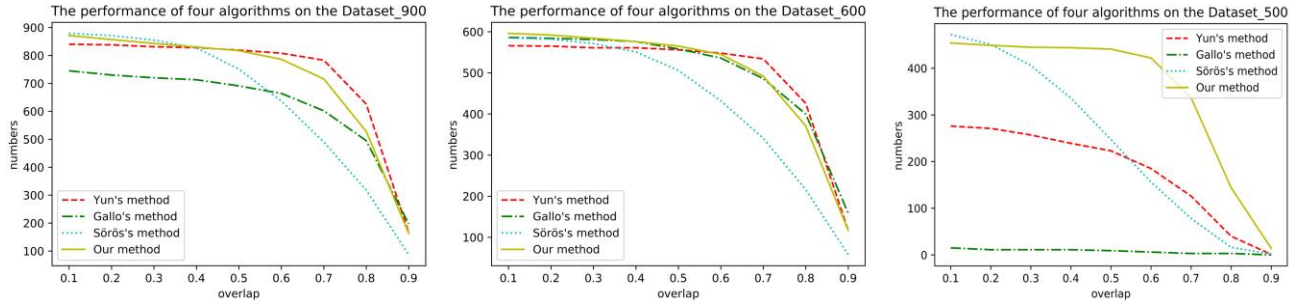
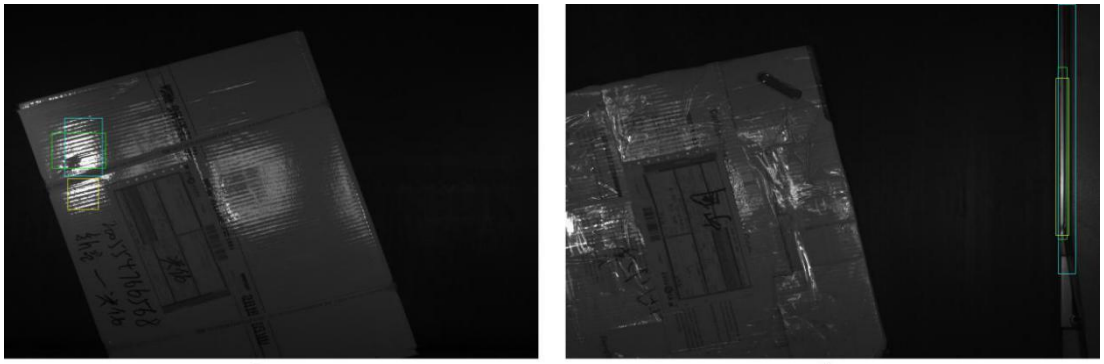


Figure 8. The performances of the four algorithms on the three datasets.

(a) experimented on the Dataset_900
(b) experimented on the Dataset_600
(c) experimented on the Dataset_500



a *b*

Figure 9. Examples of barcode localization for the above four algorithms.

(a)(b) show Gallo's method [14] (green box in the image), Sörös's method [5] (blue box in the image) and our experimental algorithm (yellow box in the image) barcode localization method results. The Yun's barcode localization method failed to locate the barcode, because the image don't meet its requirement.

Table 1. True Positive for barcode localization algorithms on three datasets

Methods	Dataset_900 (640*480)	Dataset_600 (640*480)	Dataset_500 (3072*2048)
Yun [10]	808	548	185
Gallo[14]	664	536	6
Sörös [5]	637	432	156
Ours	786	545	422

4.3 Average Resolution Time

Speed is a key performance indicator in industrial sites. We tested on dataset_900, dataset_600 and dataset_500 respectively, and the average resolution time is shown in Table 3. As can be seen from the Table 3, Gallo's method [14] takes the shortest time on each dataset, and Sörös's [5] takes the longest time. In large-scale images, the resolution time of our method is close to the Yun's method [10], but is much better than Sörös's method and lower than Gallo's method.

Table 2. Accuracy for barcode localization algorithms on three datasets

Methods	Dataset_900 (640*480)	Dataset_600 (640*480)	Dataset_500 (3072*2048)
Yun [10]	89.8%	91.3%	37.0%
Gallo[14]	73.8%	89.3%	1.2%
Sörös [5]	70.8%	72.0%	31.2%
Ours	87.3%	90.8%	84.4%

Table 3. Average Resolution Time for barcode localization algorithms on three datasets

Methods	Dataset_900 (640*480)	Dataset_600 (640*480)	Dataset_500 (3072*2048)
Yun [10]	16.8 ms	17.8 ms	257.1 ms
Gallo[14]	6.2 ms	6.5 ms	119.3 ms
Sörös [5]	41.5 ms	44.3 ms	1016.6 ms
Ours	24.6 ms	25.5 ms	261.9 ms

4.4 Limitations

Due to the existence of reflections and other adverse factors in the industrial site, sometimes the edges of certain areas of the image might not be strong enough and some misleading stripes might also appear in some areas, which easily leads to mistake the wrong BLOB for the barcode region. Figure 9 shows wrongly-detected images.

5. CONCLUSION AND FUTURE WORK

This paper proposes a novel barcode localization algorithm, which can better solve the barcode localization problem for low-quality images. Experimental results on public datasets show that our approach is close to other methods in terms of accuracy. Furthermore, in the private dataset, even with more severe adverse factors such as rotation, reflection and blur in the barcode, our proposed method is much better than the other three methods, and it sustains good robustness. Also, the average resolution time of the proposed method is acceptable compared to other methods, which is possible to be applied in the industrial fields.

In the future, as the logistics industry continues to be prosperous, packages might contain more than one barcode at different sizes and shapes. Meanwhile, the materials and ink for printing barcodes might be changed for different functions, which also pose a big challenge for barcode localization.

Hence, we will conduct further investigation to focus on the error detection by reflection, and find other solutions to enhance our method to address the coming challenges.

6. REFERENCES

[1] Sun, H. 2009. The Application of Barcode Technology in Logistics and Warehouse Management. First International Workshop on Education Technology and Computer Science, (May. 2009), 732-735. DOI=10.1109/ETCS.2009.698.

[2] Katona, M., Nyúl, L.G. 2013. Efficient 1D and 2D Barcode Detection Using Mathematical Morphology. Mathematical Morphology and Its Applications to Signal and Image Processing(ISMM, 2013, Lecture Notes in Computer Science, vol 7883, Springer, Berlin, Heidelberg).

[3] Katona, M., Nyúl, L.G. 2012. A novel method for accurate and efficient barcode detection with morphological operations. The 8th International Conference on Signal Image Technology, (Nov. 2012), 307-314. DOI=10.1109/SITIS.2012.53.

[4] Yahyanejad, S., Ström, J. 2010. Removing motion blur from barcode images. IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Workshops, (Jun. 2010), 41-46. DOI=10.1109/CVPRW.2010.5543258.

[5] Sörös, G., Flörkemeier, C. 2013. Blur-resistant joint 1D and 2D barcode localization for smartphones. MUM '13 Proceedings of the 12th International Conference on Mobile and Ubiquitous Multimedia, (Dec. 2013), Article 11. DOI=10.1109/ICIIS.1999.810282.

[6] Muniz, R., Junco, L., Otero, A. 1999. A Robust Software Barcode Reader Using the Hough Transform. Proceedings 1999 International Conference on Information Intelligence and Systems, (Oct.-Nov. 1999), 313-319. DOI=10.1109/ICIIS.1999.810282.

[7] Safran, M.I., Oktem, R. 2007. A Fast Hough Transform Approximation and Its Application for Barcode Localization. 2007 IEEE 15th Signal Processing and Communications Applications, (Jun. 2007), 1-4. DOI=10.1109/SIU.2007.4298852.

[8] Zamberletti, A., Gallo, I., Albertini, S., et al. 2015. Neural 1D barcode detection using the Hough transform,Information and Media Technologies, 2015, 10, (1), 157-165. DOI=https://doi.org/10.11185/imt.10.157

[9] Tekin, E., Coughlan, J.M. 2012. Blade: Barcode localization and decoding engine, Tech Report 2012-RERC.01, 2012.

[10] Yun, I., Kim, J. 2017. Vision-based 1D Barcode Localization Method for Scale and Rotaion Invariant. TENCON 2017 - 2017 IEEE Region 10 Conference, (Nov. 2017), 2204-2208. DOI= 10.1109/TENCON.2017.8228227.

[11] Wang, M., Li, L., Yang, Z. 2010. Gabor filtering-based scale and rotation invariance feature for 2D barcode region detection. 2010 International Conference on Computer Application and System Modeling (ICCASM 2010), (Oct. 2010), volume 5, 34-37. DOI=10.1109/ICCASM.2010.5619370.

[12] Kutiyanawala, A., Qi, X., Tian, J. 2009. A Simple and Efficient Approach to Barcode Localization, 2009 7th International Conference on Information, Communications and Signal Processing (ICICS), (Dec. 2009), 1-5. DOI=10.1109/ICICS.2009.5397472.

[13] Oktem, R. 2004. Bar code localization in wavelet domain by using binary morphology. Proceedings of the IEEE 12th Signal Processing and Communications Applications Conference, (Apr. 2004), 499-501. DOI=10.1109/SIU.2004.1338574.

[14] Gallo, O., Manduchi, R. 2011. Reading 1D barcodes with mobile phones using deformable templates, IEEE Transactions on Pattern Analysis and Machine Intelligence, 2011, 33, (9), 1834-1843. DOI=10.1109/TPAMI.2010.229.

[15] Otsu, N. 1979. A Threshold Selection Method from Gray-Level Histograms, IEEE Transactions on Systems, Man and Cybernetics, 1979, 9, (1), 62-66. DOI=0.1109/TSMC.1979.4310076.

[16] Bodnár, P., Nyúl, L.G. 2012. Improving Barcode Detection with combination of Simple Detectors. 2012 Eighth International Conference on Signal Image Technology and Internet Based Systems, (Nov. 2012), 300-306. DOI=10.1109/SITIS.2012.52.

[17] Chang, S.K., Yang, C.C. 1983. Picture information measures for similarity retrieval, Computer Vision, Graphics, and Image Processing, 1983, 23, (3), 366-375. DOI=https://doi.org/10.1016/0734-189X(83)90034-8.

[18] Harris, C., Stephens, M. 1988. A combined corner and edge detector, Proceedings of the Fourth Alvey Vision Conference, Manchester, England, August-September 1988, 147-151.

[19] Ando, S. 2000. Image field categorization and edge/corner detection from gradient covariance, IEEE Transactions on Pattern Analysis and Machine Intelligence, 2000, 22, (2), 179-190. DOI=10.1109/34.825756.

[20] Pattern Recognition and Image Analysis, https://www.uni-muenster.de/PRIA/en/forschung/index.shtml, accessed 11 August 2018.