BULGARIAN ACADEMY OF SCIENCES

CYBERNETICS AND INFORMATION TECHNOLOGIES • Volume **17**, No 1
Sofia • 2017 Print ISSN: 1311-9702; Online ISSN: 1314-4081
DOI: 10.1515/cait-2017-0011

Research on Quick Response Code Defect Detection Algorithm

Guo Yanhua[1], *Zhou Sihua*[2], *Zhou Xiaodong*[1], *Chen Bojun*[2], *Wang Shaohui*[2]

[1]*School of Mechanical and Electrical Engineering, Zhoukou Normal University, Zhoukou 466001, China*
[2]*School of Physics and Telecommunication Engineering, Zhoukou Normal University, Zhoukou 466001, China*

Abstract: *Defect Detection is one of the most important parts of Automatic Identification and Data transmission. Quick Response code* (*QR code*) *is one of the most popular types of two-dimensional barcodes. It is a challenge to detect defect of various QR code images efficiently and accurately. In this paper, we propose the procedure by a serial of carefully designed preprocessing methods. The defect detection procedure consists of QR code identification, QR code reconstruction, perspective transformation, image binarization, morphological operation, image matching, and Blob analysis. By these steps, we can detect defect of different types of QR code images. The experiment results show that our method has stronger robustness and higher efficiency. Moreover, experiment results on QR code images show that the prediction accuracy of proposed method reaches 99.07% with an average execution time of 6.592 ms. This method can detect defect of these images in real time.*

Keywords: *QR Code, defect detection, perspective transformation, blob analysis, correlation matching.*

1. Introduction

Barcodes are widely used because of its advantages of automatic identification. Two-dimensional barcode compared with the one-dimensional barcode has the following advantages: (1) high data capacity; (2) no additional storage; (3) great error correction ability. In recent years, the two-dimensional code technology, which is a kind of electronic media, promotes great development of information transmission. Quick Response code (QR code) is one of the most popular types of two-dimensional barcode developed in Japan by Denso Corporation. Although QR code was first designed for the automotive industry, they are now applied over much wider range of applications, including commercial tracking, transport ticketing and identity verification, etc.

There are certain researches to QR code. Lin and Fuh [1] proposed a serial of carefully designed preprocessing methods to decode QR code images. Belussi and Hirata in [2, 3], proposed a fast component-based two-stage approach for detecting QR codes in arbitrarily acquired images. Moreover Ohbuchi, Hanaizumi and Hock [4] and Y. Liu, J. Yang and M. Liu [5] focused on enabling mobile phone to recognize QR code under poor conditions in real time. Yong, Wang and Ai [6] proposed a recognition approach for QR code based on correlation match. Fan, Jiang and Liu [7] proposed an approach for localization of barcode according to gradient and morphological mathematics. Sun et al. [8] proposed a recognition method of QR Code based on sparse representation. However, researches of two-dimensional code is restricted to the encode and decode, location, identification, etc., while the defect detection of two-dimensional code was not involved.

We focus on revising preprocessing methods in traditional QR code processing procedure on the base of the analysis of previous researches. An algorithm which can be used in industrial on-line defect detection is proposed in this paper, in order to complete the defect detection. The algorithm steps consist of QR code identification and reconstruction, perspective transformation, morphological operation, image binarization, image matching and Blob analysis. The experimental results show that the algorithm is robust enough, with 99.07% accuracy. The average execution time of the algorithm is 6.592 ms, which has verified the feasibility of the algorithm.

2. Algorithm theory

2.1. QR code recognition and reconstruction

According to certain rules, QR code symbol uses a particular geometric figure in the plane with the distribution of the black and white graphic symbols to record information data.

The QR code symbol has a general structure that comprises encode strings, version, marking, error correction level, symbol size, Finder Patterns (FP), Timing Patterns (TP) and Alignment Patterns (AP) [1]. FP, TP, AP and QR code recognition algorithms, based on encode algorithm, obtain the encoding information of QR code, and finish the reconstruction of QR code. QR code image sample and reconstruction image are shown in Fig.1.

(a) (b)

Fig. 1. QR code image sample (a); QR code reconstruction image (b)

2.2. Perspective transformation

Due to taking a QR code image in the incorrect direction, geometric deformation and other factors, there is 2D spatial distortion in barcodes such as spatial rotation and skew shown on Fig. 2a.

After detecting four corner points, a linear transformation to rectify any spatial distortion can be used in the QR code without any given skewed information such as rotation angles and elevation angles [9]. Since subsequent processing of this algorithm needs matching processing, adopting coordinate-rotation transform will cause the size changes of the image after correction for cases where the geometric deformation is serious compared with template image; this can lead to mismatching. So, in this algorithm, we take perspective transformation for QR code correction. The principle of Perspective transformation is shown on Fig. 2.

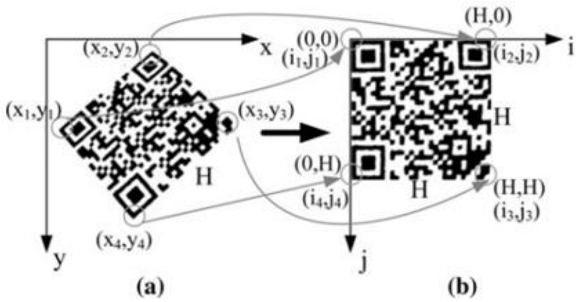


Fig. 2. Schematic of perspective transformation

(1) $$\begin{bmatrix} x' & y' & \omega' \end{bmatrix} = \begin{bmatrix} i & j & 1 \end{bmatrix} \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix},$$

where $x = x'/\omega'$ and $y = y'/\omega'$. In (1), each original coordinate (i, j) can be transformed to a new coordinate (x, y) by using the 3×3 matrix. Based on (1), x and y can be written as

(2) $$\begin{cases} x = \dfrac{x'}{\omega'} = \dfrac{a_{11}i + a_{21}j + a_{31}}{a_{13}i + a_{23}j + a_{33}}, \\ y = \dfrac{y'}{\omega'} = \dfrac{a_{12}i + a_{22}j + a_{32}}{a_{13}i + a_{23}j + a_{33}}. \end{cases}$$

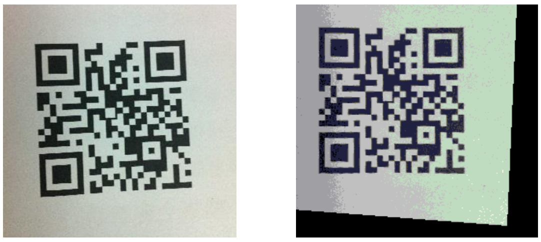
Without loss of generality, the 3×3 matrix can be normalized, so that $a_{33} = 1$. Equation (2) can be written as

(3) $$\begin{cases} x = a_{11}i + a_{21}j + a_{31} - a_{13}ix - a_{23}jx, \\ y = a_{12}i + a_{22}j + a_{32} - a_{13}iy - a_{23}jy. \end{cases}$$

Applying four pairs of correspondence points to (3), we can get a linear system as follows:

(4) $$\begin{bmatrix} i_0 & j_0 & 1 & 0 & 0 & 0 & -i_0x_0 & -j_0x_0 \\ i_1 & j_1 & 1 & 0 & 0 & 0 & -i_1x_1 & -j_1x_1 \\ i_2 & j_2 & 1 & 0 & 0 & 0 & -i_2x_2 & -j_2x_2 \\ i_3 & j_3 & 1 & 0 & 0 & 0 & -i_3x_3 & -j_3x_3 \\ 0 & 0 & 0 & i_0 & j_0 & 1 & -i_0y_0 & -j_0y_0 \\ 0 & 0 & 0 & i_1 & j_1 & 1 & -i_1y_1 & -j_1y_1 \\ 0 & 0 & 0 & i_2 & j_2 & 1 & -i_2y_2 & -j_2y_2 \\ 0 & 0 & 0 & i_3 & j_3 & 1 & -i_3y_3 & -j_3y_3 \end{bmatrix} \times \begin{bmatrix} a_{11} \\ a_{21} \\ a_{21} \\ a_{12} \\ a_{22} \\ a_{32} \\ a_{13} \\ a_{23} \end{bmatrix} = \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \\ y_0 \\ y_1 \\ y_2 \\ y_3 \end{bmatrix}.$$

The value of unknown coefficients will be determined by solving the linear system in (4). Correction results are shown in Fig. 3.



(a) (b)

Fig. 3. QR code sample (a); correction result (b)

2.3. Morphological processing

The basic morphological operators are erosion, dilation, opening and closing [10]. In morphology, structure elements are the most basic and important concept. The role of structural elements in morphological transform is equivalent to the signal processing of the “filtering window”.

The input image $f(x,y)$ is inflated by structural elements $b(x,y)$, which is defined as, and $f \oplus b$ expressed by

(5) $(f \oplus b)(s,t) = \max\{f(s-x,t-y)+b(x,y) \mid (s-x,t-y) \in D_f;(x,y) \in D_b\}$.

Input image $f(x,y)$ corroded by $b(x,y)$ is defined as $f \ominus b$, and it’s expression is

(6) $f \ominus b\ (s,t) = \min\{f(s+x,t+y)-b(x,y) \mid (s+x,t+y) \in D_f;(x,y) \in D_b\}$.

To corrode $f(x,y)$ is to translate structural elements $b(x,y)$ in order to form a set of points, which include all results of $b(x,y)$ contained in $f(x,y)$. Using $b(x,y)$ to inflate $f(x,y)$ is to translate structural elements $b(x,y)$ in order to form a set of points, which include the nonempty intersection between $b(x,y)$ and $f(x,y)$.

2.4. Binarization

The image binarization is used to obtain QR Code, and Otsu method is a common and efficient binary algorithm [11-14]. We assume that the QR grey level images have L pixel-levels, P_g is the frequency of each grey level value appeared; t is optimal threshold for foreground and background. The proportions of foreground and background pixels are as follows:

$$\begin{cases} \theta(t) = \sum_{g=1}^{L} P_g, \\ 1-\theta(t) = \sum_{g=t+1}^{L} P_g. \end{cases} \tag{7}$$

Average gray of foreground and background pixels are as follows:

$$\begin{cases} \mu_f = \dfrac{\sum_{g=1}^{T} gP_g}{\theta(t)} = \dfrac{\mu(t)}{\theta(t)}, \\ \mu_b = \dfrac{\sum_{g=t+1}^{L} gP_g}{1-\theta(t)} = \dfrac{\sum_{g=1}^{L} gP_g - \sum_{g=1}^{t} gP_g}{1-\theta(t)} = \dfrac{\mu-\mu(t)}{1-\theta(t)}. \end{cases} \tag{8}$$

As defined $\mu(t) = \sum_{g=1}^{T} gP_g$, μ represented the average grey level value of QR grey level images. Hence, the variances of the two parts are determined by the threshold t as follows:

$$\begin{cases} \sigma_f^2 = \dfrac{\sum_{g=1}^{t} (g-\mu_f)^2 P_g}{\theta(t)}, \\ \sigma_b^2 = \dfrac{\sum_{g=t+1}^{L} (g-\mu_b)^2 P_g}{1-\theta(t)}, \\ \sigma_t^2 = \sum_{g=1}^{L} (g-\mu)^2 P_g, \end{cases} \tag{9}$$

t is optimal threshold when the variance σ_B^2 is maximum,

$$\sigma_B^2 = (\mu_b-\mu)^2(1-\theta(t)) + (\mu_f-\mu)^2\theta(t). \tag{10}$$

3. Proposed method

3.1. Morphological processing based on edge pixels

The traditional morphology processing degree of QR code cannot be controlled, so it cannot be better adapted to different illumination and noise acquisition environment. According to this phenomenon, we proposed a morphology processing method, which can be described as the control ability of edge pixel number and operation degree.

In the proposed approach, we take morphology processing to N edge pixels of template image (N is integer as in equation (11)). And at the same time, the processing of pixel can be controlled. (In the following we called the adjustment coefficient β). According to the test and analysis of the 10 000 samples, we suggest that the values can be 0.5, 0.6, 0.8, 1.0, 1.3, 1.0. This method does not need to process the sample image, which can retain the original image features to a maximum extent. Adjusting gray is given in the equation (12)：

(11) $$N = M_{\text{pixel}}/2,\ M_{\text{pixel}} \in [3, 20].$$

Here, M_{pixel} is the pixel width (or length) of a single module of QR code; 20 is the biggest size module coding of QR code,

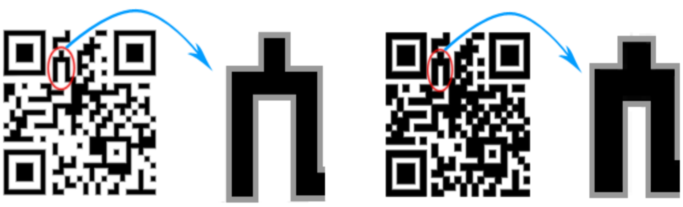
(12) $$V'_{N\_\text{gray}} = \beta_N \cdot V_{N\_\text{gray}},$$

$\beta_N = \beta_{[1-10]} = \{0.5, 0.6, 0.8, 1.0, 1.3, 1.5, 1.5, 1.5, 1.5\}$; V'_{N_gray} is the edge pixel gray value after morphological processing, β_N is the adjustment coefficient of edge pixels, V_{N_gray} is the gray value of the N-th pixel distance from edge before processing.

High threshold images are generated by corrosion operation for template images, denoted as I_{h_thres}, and low threshold images are generated by inflate operation for template images, denoted as I_{l_thres}, as shown in Fig. 4. Local characteristics as shown in Fig. 5.

(a) (b)

Fig. 4. I_{h_thres} (a); I_{l_thres} (b)



(a) (b)

Fig. 5. I_{h_thres} local features (a); I_{l_thres} local features (b)

3.2. Template matching method

Template matching method based on using the template was proposed; it is used to search the tracked object in the searched image. The size of the template *T* is assumed *K***L*, which is stacked on searched image to translate, and the size of searched image is *M***N*. The covered subgraph by the template is defined as $S_{(i,j)}$, (i, j) is the coordinate for top left corner of the subgraph in the searched image [15, 16]. So, comparing the substance of *T* and $S_{(i,j)}$ $(1 \le i \le M - K + 1, 1 \le j \le N - L + 1)$, if they are same, then the difference between *T* and $S_{(i,j)}$ is zero. Similarity of *T* and $S_{(i,j)}$ is measured by

$$R(i,\ j) = \frac{\sum_{k=1}^{K}\sum_{l=1}^{L} S_{(i,j)}(k,l) \times T(k,l)}{\sqrt{\sum_{k=1}^{K}\sum_{l=1}^{L}\left[S_{(i,j)}(k,l)\right]^2} \times \sqrt{\sum_{k=1}^{K}\sum_{l=1}^{L}\left[T(k,l)\right]^2}}. \tag{13}$$

From (13) we can know that $0 \le R(i, j) \le 1$, when $R(i, j) = 1$, *T* and $S_{(i,j)}$ are matched.

In this algorithm, each of defect detection needs to match two times. The black line defect or black block defect is located in the result of the low threshold image matching correction image. The white line defect or white block defect is located in the result of the high threshold image matching correction image. The matching method can effectively filter interference and there is no need to judge the gray value of defect area after matching; it's accuracy is high.

3.3. Blob analysis

Blob analysis is to analyze the same pixel gray value in the connected domain of image. The connected domain is called Blob. In this paper, each defect is a Blob. For QR code in different acquisition environment, if the threshold method is inappropriate, it can reduce defect detection accuracy. In this section, we obtain foreground and background segments of QR code.

Blob Analysis can be used to measure the morphological parameters of arbitrary shape object. In this processing, Blob analysis isn't for a single pixel but row or column of image. Compared with the algorithm based on pixel, this method has the advantage of faster processing speed. Analyzing and location defect, we can get some features, such as minimum circumscribed rectangle, length, width and center coordinates, area value, line length and the number etc. This process is called feature extraction. Assume the binary images as *I*, it's number of row or column denoted by *R* or *C*. Blob analysis algorithm steps are as follows:

Step 1. Initialize $n = 0$, set the middle row $r_0 = R/2$ as the scan line, start scanning from $c_0 = 1$ to $c_0 = C$, if $I(r_0, c_0) \ne 0$, then turn to the next step.

Step 2. Search the connected domain of the points (r_0, c_0) and set the pixel value as 0 of the coordinate *I* in connected domain. Then return the connected domain coordinates as (r, c).

Step 3. Calculate the area of the connected domain or line length. If the area or line length are in the threshold range then return the connected domain and go to the next round of cycle; If not, then directly skip to the next round of cycle.

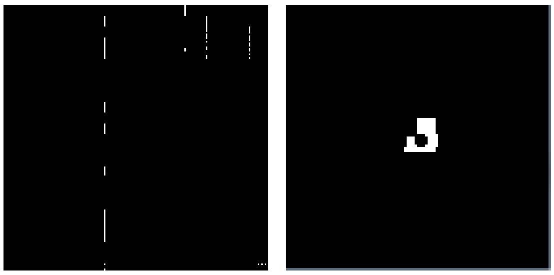
4. Experimental results

4.1. Image source

Part of our test images come from samples database, the others are from the simulation generated QR Code samples, which are produced according to the environment of sampling. Simulation generated QR Code samples mainly from three aspects, (1) to change the brightness, contrast, noise within a certain range, in order to generate random collection environment; (2) to randomly generate QR Code samples according to the encode of QR Code strings, version, marking, the error correction level, symbol encoding information; (3) to add defect types randomly, including white line defect, black line defect, white block defect and black block defect.

4.2. Accuracy analysis

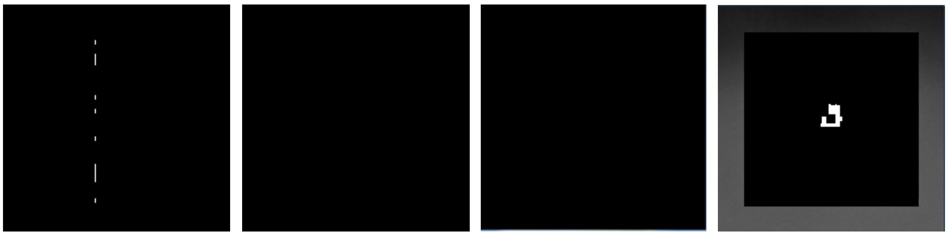
In this section, the experimental results include two parts: the (typical) sample processing result contrast and robustness test of the algorithm. The results of traditional methods matching are shown in Fig. 6. From Fig. 6a we can conclude that the traditional processing method has certain interference. Because of that the acquisition environment is relatively appropriate, and there is no interference after processing (Fig. 6b); our algorithm is able to detect defects.



(a) (b)

Fig. 6. Bina_White line (a); Bina_white block (b)

The proposed detection method, and two samples of the matching results are shown respectively in Fig. 7. We can see from Fig. 7, that this method can effectively filter out interference and detect the defect accurately.



(a) (b) (c) (d)

Fig. 7. B _white line_H (a); B_white line_L (b); B_black block_H (c); B_black block_L (d)

Based on 10 000 QR Code test sample, the traditional method omission rate is 2.1%, error rate is 10.49%, and the accuracy is 87.41%. The reasons of low accuracy in traditional method are mainly attribute to the processing is unsuitable in different environment and the automatic segmentation threshold is not appropriate. Then mistakenly identified targets will reduce the accuracy.

According to our method, we set the high threshold image's threshold as *H*_thres, the low threshold image threshold as *L*_thres, the block defect threshold as *B*_thres, and the line defect threshold as *l*_thres. Table 1 illustrates that *H*_thres is the optimal threshold when it is 150, while the omission rate and error rate are 0.22% and 0.30% respectively. And the reason of residual defects and mistakenly identified mainly is that the gray and QR code coding region of gray is very close, the defect types is moon's path and white block and after threshold caused residual and mistakenly identified. Table 1 illustrates that *L*_thres is the optimal threshold when it is equal to140. In this time the omission rate and error rate were 0.21% and 0.20% respectively. And what causes omission and error is that the defect gray and gray of QR code coding region is very close, the defect types are white lines and white blocks.

Table 1. Omission rate and error rate of high/low gray-level threshold

*H*_thres	Omission rate	Error rate	*L*_thres	Omission rate	Error rate
120	0.0011	0.0227	120	0.0010	0.0190
130	0.0012	0.0110	130	0.0021	0.0078
140	0.0022	0.0086	140	0.0021	0.0020
150	0.0022	0.0030	150	0.0049	0.0020

Table 2 illustrates that *L*_thres and *H*_thres all set the optimal threshold and when the *B*_thres is equal to 54, the omission rate and error rate were 0.42% and 0.51% respectively; at this time they are all at the lowest level. The main reason for mistaken identifications is that the defect area is too small and defect gray and gray of image coding region contrast is not obvious. Table 2 indicates that when *L*_thres and *H*_thres are the best threshold and the *l*_thres is equal to [24,36], the omission rate and error rate are 0 respectively in this case.

Table 2. Omission rate and error rate of lack of block threshold

H_thres/L_thres	B_thres	Omission rate	Error rate	l_thres	Omission rate	Error rate
150/140	27	0.0010	0.0197	12	0.0000	0.0132
150/140	36	0.0024	0.0153	18	0.0000	0.0030
150/140	45	0.0029	0.0111	24	0.0000	0.0000
150/140	54	0.0042	0.0051	36	0.0000	0.0000
150/140	63	0.0071	0.0051	42	0.0011	0.0000

All the data in the table above indicates that when the parameters take the optimal value, the omission rate and error rate were 0.42% and 0.51% respectively, and the comprehensive accuracy of defects is 99.07%.

4.3. Efficiency analysis

In this section, we show the execution time for detecting defect for different sizes of QR code images in our method. We use ten thousand different sizes of QR code images to calculate the execution time of the algorithm. Table 3 shows the results for each step of the defect detection time.

Our algorithm only needs 6.592 milliseconds average for detecting defect of QR code images. Hence, it is able to detect about 546,115 QR code image hourly.

Table 3. Each step execution time for algorithm (ms)

Number	Decode	Reconstruction	Perspective transformation	Morphological processing	Match	Blob analysis
1	3.889	0.074	0.672	1.496	0.397	0.401
2	3.809	0.051	0.600	1.523	0.408	0.586
3	3.788	0.057	0.554	1.518	0.400	0.397
4	3.816	0.051	0.723	1.537	0.406	0.410
5	3.809	0.051	0.620	1.513	0.406	0.455
6	3.886	0.054	0.569	1.523	0.395	0.400
7	3.793	0.051	0.718	1.517	0.394	0.394
8	3.860	0.051	0.566	1.489	0.397	0.558
9	3.859	0.049	0.818	1.591	0.396	0.394
10	3.826	0.058	0.563	1.528	0.397	0.396
Average time	3.834	0.055	0.640	1.524	0.400	0.439

5. Conclusion

In this paper, we have presented a method of QR code defect detection in acquired images. The detecting steps consist of QR code reconstruction, perspective transformation, image binarization, morphological processing, image matching and Blob analysis. Image rectifying, morphological processing and matching are necessary steps for accurately detecting defect. The best of knowledge and solving this challenging problem require real-time processing and robustness in respect to illumination, rotation and perspective distortions. Our carefully designed preprocessing methods improve the accuracy of the detection procedure. Based on QR code mapping relation of four sets of vertex coordinates, we adapt perspective transformation of QR code image for correction. We efficiently resolve the uneven illumination and noise problems by improved morphological processing method for

QR code image matching. This method has the advantages of high accuracy, better ability to filter out interference than traditional methods and matching gray judgment without defects. We apply the Blob analysis to extract defect information according to connected domain characteristics of defects.

The proposed approach was validated and tested on image database, resulting in QR code defect detection rates superior to 99.07% by keeping an average detecting time of 6.592 ms on test images. Part of the undetected defects are failing to reconstruct QR code. Another part is that geometrical distortions of the QR code is beyond maximum degree in the FIP aggregation algorithm.

***Acknowledgments*:** This work was supported by Henan Provincial Department of Science and Technology Research Project (No 162102210400), Education Department of Henan Province (No 16B140006) and National Nature Science Foundation of China (No 11405280).

References

1. L i n, J.-A., C.-S. F u h. 2D Barcode Image Decoding. – Mathematical Problems in Engineering, Vol. **113**, 2013, No 2, pp. 292-319.
2. B e l u s s i, L. F. F., N. S. T. H i r a t a. Fast Component-Based QR Code Detection in Arbitrarily Acquired Images. – Journal of Mathematical Imaging and Vision, Vol. **45**, 2013, No 3, pp. 277-292.
3. B e l u s s i, L. F. F., N. S. T. H i r a t a. Fast Qr Code Detection in Arbitrarily Acquired Images. – Proceedings of the Conference on Graphics, Patterns and Images, Vol. **45**, 2011, No 3, pp. 281-288.
4. O h b u c h i, E., H. H a n a i z u m i, L. A. H o c k. Barcode Readers Using the Camera Device in Mobile Phones. – In: Proc. of International Conference on Cyberworlds, IEEE, 2004, pp. 260-265.
5. L i u, Y., J. Y a n g, M. L i u. Recognition of QR Code with Mobile Phones. – In: Proc. of Control and Decision Conference (CCDC'2008), 2008, pp. 203-206.
6. Y o n g, X., L. W a n g, Q. A i. QR Code Recognition Method Based on Correlation Match. – Chinese Journal of Scientific Instrument, Vol. **32**, 2013, No 3, pp. 571-576.
7. F a n, Z. Y., F. J i a n g, Z. W. L i u. Recognition of PDF417 Barcode Based on Captured Images. – Transactions of Beijing Institute of Technology, Vol. **28**, 2008, No 12, pp. 1088-1092.
8. S u n, D. D., J. Z h a o, R. W a n g et al. QR Code Recognition Based on Sparse Representation. – Journal of Computer Applications, Vol. **33**, 2013, No 1, pp. 179-181.
9. C h u, C. H., D. N. Y a n g, Y. L. P a n et al. Stabilization and Extraction of 2D Barcodes for Camera Phones. – Multimedia Systems, Vol. **17**, 2011, No 2, pp. 113-133.
10. W a n g, G., J. W a n g, M. L i et al. Hand Vein Image Enhancement Based on Multi-Scale Top-Hat Transform. – Cybernetics and Information Technologes, Vol. **16**, 2016, No 2, pp. 125-134.
11. Z h o u, X. W. The Research of Two-Dimension Barcode Recognition Technology. Shanghai Jiao Tong University, 2007.
12. M u n o z-M e j i a s, D., I. G o n z á l e z-D í a z, F. D i a z-de-M a r i a. A Low-Complexity Pre-Processing System for Restoring Low-Quality QR Code Images. – IEEE Transactions on Consumer Electronics, Vol. **57**, 2011, No 3, pp. 1320-1328.
13. S u n, M., L. S. F u, X. T. Y a n g et al. Image Analysis Method for QR Code's Automatic Recognition. – Journal of University of Electronic Science and Technology of China, Vol. **38**, 2009, No 6, pp. 1017-1020.
14. L u o, J., Y. C h e n. Colour Restoration of Image Obtained from CCD Sensor Directly. – Cybernetics and Information Technologes, Vol. **14**, 2014, No 2, pp. 81-91.
15. C h e n, C., P. M a r z i l i a n o, A. C. K o t. 2D Finite Rate of Innovation Reconstruction Method for Step Edge and Polygon Signals in the Presence of Noise. – IEEE Transactions on Signal Processing, Vol. **60**, 2012, No 6, pp. 2851-2859.
16. D a i, Y., S. L. Y u. Two Dimensional Bar Codes Decoding Algorithm Based on Projection and Filtering-Restoring Principles. – Journal of University of Electronic Science and Technology of China, Vol. **34**, 2005, No 4, pp. 537-540.