# The Realization of Identification Method for DataMatrix Code


[1]YanGe Dai, [1]Lizhen Liu*, [1]Wei Song, [1]Chao Du, [2]Xinlei Zhao

[1]Information and Engineering College, Capital Normal University, Beijing, 100048,P.R.China

[2]College of Foreign Languages, Capital Normal University, Beijing, 100048,P.R.China

Email: 767930847@qq.com, lzliu_cnu@sina.com



***Abstract*—The minimum size of the DataMatrix code is 0.0002 square inches and this size is the smallest of the current one-dimensional and two-dimensional codes. So it is particularly suitable for printing on metal parts of circuit boards. However, compared with the general two-dimensional code, there are still many technical difficulties on metal surface DataMatrix code identification technology. The reflection on metal surface is not conducive to the DataMatrix binarization. The large spacing between the codes is not conducive to positioning and identification. There is a distorted or incomplete DataMatrix code on metal surface. In this paper, a DataMatrix code recognition method is proposed by digital image processing technology. Firstly, Binarization is achieved by improving the traditional Otsu algorithm. The binarized image is dilated to solve the problem of large code interval. Secondly, the DataMatrix code region is located. The Hough transform is used to detect the four vertices of the DataMatrix code. The vertex is used to correct the twisty DataMatrix code. Finally, according to the structural characteristics of the DataMatrix code, the incomplete code is completed. According to the encoding rules, the DataMatrix code is decoded. After the above series of steps, we can achieve the purpose of identifying DataMatrix code.**



***Keywords—DataMatrix; image processing; binarization***


## I. Introduction

The minimum size of DataMatrix code is the least of all the bar code especially suitable for marking small parts. It is this advantage that makes DataMatrix code more widely used in electronic components, semiconductor crystals, aerospace components and military property identification [1]. There are some differences between the DataMatrix code in this paper and the DataMatrix code printed on the metal. The DataMatrix code printed on the metal is a small rectangle of black and white, and the structure information is intact. In this paper, the DataMatrix code pneumatic printed on the metal. The surface is uneven, so the light reflection effect is different. The pictures captured by the ordinary camera are uneven illumination and the code spacing is very large, which are not good for identification. The metal in the industry is prone to oil pollution, incomplete, and so on. It also brings some difficulties for the identification. If we use the traditional method of DataMatrix code identification, the recognition rate will be greatly reduced, and even the DataMatrix code can not be identified.

Based on the general two-dimensional code recognition method, the traditional DataMatrix code recognition method is improved in this paper. There are many traditional binaryzation methods, such as global threshold method and Otsu threshold method. But it is not suitable for the environment of this paper. The image is divided into blocks and use Otsu method on each blocks to achieve binarization. On the basis of the traditional recognition method, morphological operations are added to reduce the space between the codes. Next, locate the area of the DataMatrix. There are many ways to locate the area. In this paper, we use Hough transformation to detect the straight lines to locate the area. The intersection point is calculated and the region of the DataMatrix code is corrected according to the intersection point. Draw mesh lines in the corrected region. According to the mesh line drawing standard DataMatrix codes and complete DataMatrix codes. Finally, initial data is obtained by decoding .

The rest of the article is organized as follows. In the next section the previous work is specifically introduced. Section III introduces the method in detail, and experiment results are reported and discussed in section IV. Finally, Section V is about conclusions.

## II. Related Work

In the field of DataMatrix code identification, many researchers have proposed their own methods. The general process of the method is preprocessing, positioning and rectification, and identification. Many methods are to improve and optimize the location and correction algorithm of the DataMatrix code region.

In 2007, L Beng proposed a method to identify DataMatrix code. A distorted image of a DataMatrix code was obtained by simulating actual environment. The barcode area in the image was acquired by barcode localizing. After the barcode area being analyzed, these barcode characteristic parameters were abstracted and the distorted image fitting formula was gained by exercise, which set up the correspondence between distorted barcode image point coordinates and the standard one [2]. Xu W, Mccloskey S proposed a method to locate the DataMatrix region by Harris corner detection [3]. But the dilated DataMatrix structure is not standard. It is easy to detect many corners so that the vertices cannot be judged. Liu F, Liu A, Wang M, Yang Z proposed a new method to the localization of DataMatrix. The method for DataMatrix barcode localization based on boundary tracking and Radon transform [4]. Huang Q, Chen W, Huang X, Zhu Y presents a novel algorithm for


**978-1-5386-1978-0/17/$31.00 ©2017 IEEE**

locating DataMatrix code based on finder pattern detection and bar code border fitting [5]. Kulshreshtha R, Kamboj A, Singh, S compared the decoding robustness of QR Code and DataMatrix barcode subjected with varying degree of noise. From their experiment it is inferred that DataMatrix code is more robust to noise and decodable than the QR code with the same level of noise [6]. In 2014, Kruchinin A Y, a scholar of the Russian Academy of Sciences, proposed a DataMatrix recognition method for arbitrary angles in complex backgrounds [7]. WANG, Weiping, Gaifang present a method for extracting the best candidate edge combination based on multi-features fusion, aiming at the challenges of accurately locating the DataMatrix code with pollution and perspective deformation [8]. In 2016, Wang G, Xiangkun W U, Kun H U use the cross correlation matching algorithm to locate the code, Then they use the pyramid-layered algorithm to reduce the calculation cost. The improved Hough transform is used to carry out the 'L' boundary line of the image [9].

The above identification methods for DataMatrix are aimed at the integrity of the code information structure, without intervals and metal surface leveling. In this paper we improved the binaryzation algorithm in the preprocessor to adapt to the environment of this paper.

## III. Methods of Binaryzation And Location

### A. Binaryzation

Due to the low camera pixel and the light reflection of the metal surface, the collected images are blurred and the noise is serious. The characteristic information of the image is concealed, and the code information of the DataMatrix is lost, which makes the recognition rate very low. In order to alleviate the interference of the above situation, more key information is retained. In this paper, we improve the Otsu algorithm and divide the image into blocks. Each block uses Otsu algorithm to realize the binaryzation in the uneven illumination.

Suppose the width and height of the DataMatrix code Image are W, H. The gray level of the image is $L$, and the range of the gray value of the pixel is $\{0, 1, \ldots L-2, L-1\}$, $N_i$ represents the number of pixels of the gray scale $i$. $C$ is the total number of pixels in the image. After the normalized histogram, the probability $p_i=N_i/C$. Now, we choose an initial threshold $K$ and the $K$ in the range from 0 to $L-1$, According to this threshold $K$, the original image is classified, the threshold less than $K$ is divided into a kind of point set $S_1$, the threshold greater than or equal to $K$ is divided into another kind of point set $S_2$, then the cumulative probability of pixels in the $S_1$ is:

$$P_1(K) = \sum_{i=0}^{k} p_i . \tag{1}$$

The cumulative probability of pixels in the $S_2$ is:

$$P_2(K) = \sum_{i=k+1}^{L-1} p_i = 1 - P_1(K). \tag{2}$$

The average gray value in $S_1$ is:

$$m_1(k) = \sum_{i=0}^{k} ip\left(\frac{i}{s_1}\right) = \sum_{i=0}^{k} \frac{ip\left(\frac{s_1}{i}\right)p(i)}{p(s_1)} = \frac{1}{P_1(k)}\sum_{i=0}^{k} ip_i . \tag{3}$$

Similarly, the average gray value in $S_2$ is:

$$m_2(k) = \frac{1}{P_2(k)}\sum_{i=k+1}^{L-1} ip_i . \tag{4}$$

The cumulative mean value obtained by the threshold $K$ is:

$$\mathrm{m}(\mathrm{k}) = \sum_{i=0}^{k} ip_i . \tag{5}$$

And the average intensity of the entire image is given by

$$m_G = \sum_{i=0}^{L-1} ip_i . \tag{6}$$

And $\sigma^2$ is the between-class variance, defined as [10]:

$$\sigma^2 = p_1(m_1 - m_G)^2 + p_2(m_2 - m_G)^2 . \tag{7}$$

This expression can be written also as:

$$\sigma^2 = p_1 p_2 (m_1 - m_2)^2 = (m_G p_1 - m)^2 / p_1(1 - p_1) . \tag{8}$$

Reintroducing $K$, we can get the between-class variance related to $K$, and get the following equation:

$$\sigma^2(\mathrm{K}) = \frac{\left[m_G p_1(\mathrm{k}) - \mathrm{m}(\mathrm{k})\right]^2}{p_1(k)\left[1 - p_1(k)\right]} . \tag{9}$$

Find the best threshold $K^*$, $K^*$ is used to maximize the value of $\sigma^2(k)$.

### B. Morphological Operation

After binaryzation, there is a large interval between the DataMatrix code, which is not conducive to the location and recognition. It can reduce the space with morphological dilate algorithm.

With $A$ and $B$ as sets in $Z^2$ The dilation of $A$ by $B$ denoted $A \oplus B$ [11]:

$$A \oplus B = \{Z \mid (\hat{B})_Z \cap A \neq \varnothing\}. \quad (10)$$

This equation (10) is based on reflecting B about its origin, and shifting this reflection by Z. The result of the dilate algorithm is a collection of all shifting.

### *C. Localization and Correction*

To better locate the DataMatrix region, the boundary of the DataMatrix region must be detected. Hough transform is a classical method to recognize geometric shapes in images [12]. In this paper, Hough transform is used to detect lines and locate regions Because of the random shooting angle, DataMatrix code must exist distortion. Therefore, it is necessary to do geometric distortion correction for the image [13].

The pixel coordinates are represented by (x, y) before distortion, and the pixel coordinates are represented by (x', y') after distortion. The correspondence between them are denoted as follows:

$$x' = s(x, y). \quad (11)$$

$$y' = t(x, y). \quad (12)$$

s(*x, y*) and t(*x, y*) are the spatial transformation functions of geometrically distorted images, if they are the simplest linear distortion. Then s(*x, y*) and t(*x, y*) can be obtained by the follows:

$$s(x, y) = k_1 x + k_2 y + k_3. \quad (13)$$

$$t(x, y) = k_4 x + k_5 y + k_6. \quad (14)$$

However, most distortions are nonlinear, then s(*x, y*) and t(*x, y*) can be obtained by the follows:

$$s(x, y) = k_1 + k_2 x + k_3 y + k_4 x^2 + k_5 xy + k_6 y^2. \quad (15)$$

$$t(x, y) = k_7 + k_8 x + k_9 y + k_{10} x^2 + k_{11} xy + k_{12} y^2. \quad (16)$$

Knowing the equation (15) and equation (16), we can restore the image by inverting it. In fact, neither of these two equation is known. We need to find the point where the location of the recovery Image and distortion Image are known. These points are called constraint corresponding points [18]. The coefficients in equation (15) and equation (16) are calculated by the coordinate positions of these points so as to establish the corresponding spatial position of the distortion Image and the recovery Image.

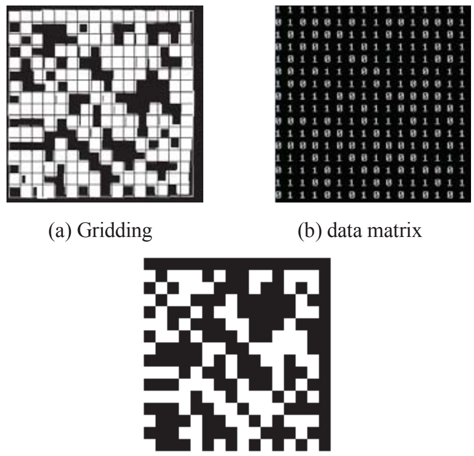
### *D. Decoding*

Before decoding, the standard network structure of the DataMatrix code is obtained, and then the pixel matrix of the DataMatrix code is used to decode.

Set the width and height of the image to be *w*. The upper-left point coordinate is (*a, a*). The coordinates of the other three vertices are (*a+w, a*), (*a+w, a+w*), (*a, a+w*). Then use four lines to scan the four boundaries, and we can get the code number *N* of rows or columns. The length of each code is *w/n*. The X direction and the Y Direction are plotted N lines respectively, and the interval between each line is *w/n*. As shown in Figure 1 (a), *a = 10, w = 250, n=16*. So the spacing between each line is 250/16. Scan each small area of the image and count the number of white pixels. Set the number to be $n_{ij}$ Where (*i, j*) represents the location of each small area in the Image. The data $A_{ij}$ of the data matrix region is denoted as follows:

$$A_{ij} = \begin{cases} 0, n_{ij} / N < 0.5 \\ 1, n_{ij} / N \geq 0.5 \end{cases}. \quad (17)$$

The N in equation (17) is the total number of pixels in each of the small rectangular regions. As shown in Figure 1 (b), the data region matrix of the data matrix code is obtained. If $A_{ij}$=1, draws a small black matrix of 10*10 pixels. If $A_{ij}=0$, draws a small white matrix of 10*10 pixels. Figure 1 (c) is a standard DataMatrix code Image.



(a) Gridding (b) data matrix

(c) standard DataMatrix

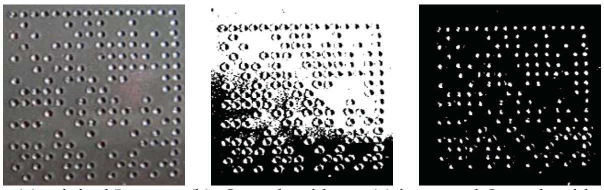
Fig. 1. Location and recognition results

The encoding process involves conversion from initial data to DataMatrix code. First, according to the length and type of the code, the ECC200 encoding table is selected to set the corresponding size. Then, the error correction code is computed according to the Reed-Solomon algorithm and the error correction code is placed in the data code. Finally, the code are arranged according to the standard in the ISO/IEC-16022DataMatrix protocol [14]. Plus the boundary constitutes the standard DataMatrix code. Decoding is the inverse process of encoding. After the data region is obtained, all the code are extracted, including both the data code and the error correcting code .Correct the wrong code according to the error correction code. Then, according to the size of the DataMatrix code, select the data code. The initial data can be calculated according to the decoding table.

## IV. Experiments And results Analysis

In this section, a contrast of the experimental results and the results of each step of the experiment are shown. Fully demonstrates the feasibility of the method in this paper.

### A. *Effect of Improved Otsu Algorithms under Uneven Illumination*



(a) original Image (b) Otsu algorithm (c) improved Otsu algorithm

Fig. 2. Otsu algorithm and improved Otsu algorithm

As shown in Figure 2, (a) is a DataMatrix code Image of uneven illumination. (b) is the binaryzation of Otsu algorithm. It can be seen that the effect is very poor. The DataMatrix and the background area are not correctly separated. And most of the metal surface exist reflective, so this method is not suitable for the subject. (c) is the binaryzation of improved Otsu algorithm. The improved algorithm has better effect on DataMatrix image with uneven illumination. At the same time, different DataMatrix images automatically calculate the threshold. No manual change threshold are required, and they can be adapted to different lighting. Therefore, this paper adopts this method.

### B. *Effect of Different Denoising Methods*

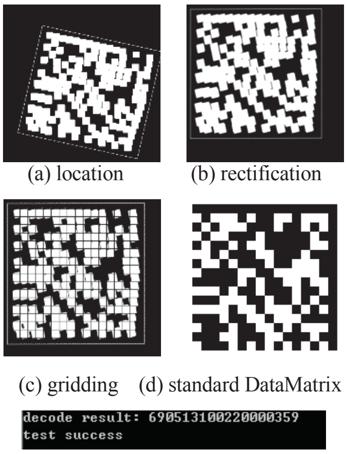
The noise can be removed by filtering, and the median filtering is adopted in this paper. Median filter has a good filtering effect on salt&pepper noise.

(a) binaryzation Image (b) Mean filter (c) median filtering

Fig. 3. median filtering and mean filtering

As shown in Figure 3. (a) is DataMatrix code after binaryzation. (b) uses mean filter to denoise. and (c) uses median filtering to denoise. Compared with (b) and (c), it can be found that the image is blurred after mean filtering. The median filter denoising can get better results. Median filtering denoising is adopted in this subject. So the method has good effect in noisy environment.

### C. *Location and Recognition*




(a) location (b) rectification

(c) gridding (d) standard DataMatrix code

(e) recognition result

Fig. 4. Location and recognition results

As shown in Figure 4. (a) is a DataMatrix code that is positioned. The white line in the picture is the area to be fixed. (b) corrected by figure (a). (c) is obtained by gridding in (b). (e) is the result of recognition. However, the DataMatrix code may incomplete. Judge the four sides of the DataMatrix, If the number of the black small rectangle is more than 3/5 of the number of lines, we consider that the line is the L edge of DataMatrix code. Then use a small black rectangle instead of a small white rectangle on the edge.

TABLE I. Recogntion Rate of DataMatrix Code on Metal Surface

| **Rotation angle** | 0 | 30 | 45 | 60 |
|---|---|---|---|---|
| **Sample size** | 200 | 200 | 200 | 200 |
| **Total time(ms)** | 8120 | 8240 | 8460 | 8300 |
| **Average time(ms)** | 40.6 | 41.2 | 42.3 | 41.5 |
| **Correct identification number** | 188 | 183 | 181 | 185 |
| **Recognition rate** | 0.940 | 0.915 | 0.905 | 0.925 |

As shown in table 1. The first line in the table is the DataMatrix code with different angles. This table mainly shows the average recognition rate and average recognition time of DataMatrix code images in different angles. The method proposed in this paper has higher recognition rate and less time consuming. It can also be seen that the angle has little effect on the recognition rate and the recognition time. It is proved that the method has certain stability.

## V. Conclusions

This paper proposes a method for recognizing DataMatrix codes on metal surfaces. The basic process includes binarization, denoising, morphological operations, Hough transform, geometric correction, drawing and complete DataMatrix codes, decoding. By improving the traditional Otsu algorithm, the problem that the reflection of metal surface is not easy to binaryzation is solved. Through morphological dilate algorithm, the problem of large spacing between the DataMatrix codes and difficult to identify are solved. Through the grid, complete DataMatrix code and solves the problems that oil pollution and incomplete DataMatrix codes unable to be identified. Combined with DataMatrix decoding library, we successfully identify the DataMatrix code on metal surface. Experimental results show that the proposed method is feasible for the identification of DataMatrix codes on metal surface. Other schemes mostly use expensive photosensitive components to solve these technical difficulties through physical methods. Therefore, the method in this paper has the advantages of low economic cost and better popularization.

## Acknowledgment

This work was supported in part by National Science Foundation of China under Grants No. 61402304, the Beijing Educational Committee Science and Technology Development Planned under Grants No. KM201610028015, Beijing Science and technology innovation platform.

## References

[1] page. The 2D data matrix barcode[J]. Computing & Control Engineering Journal, 2006, 16(6):39.

[2] Leng B. A DataMatrix-based Mutant Code Design and Recognition Method Research[C]// International Conference on Image and Graphics. IEEE Computer Society, 2007:570-574.

[3] Xu W, Mccloskey S. 2D Barcode localization and motion deblurring using a flutter shutter camera[C]// Applications of Computer Vision. IEEE, 2011:159-165.

[4] Liu F, Liu A, Wang M, et al. Robust and Fast Localization Algorithm for Data Matrix Barcode[M]. IEEE, 2010.

[5] Huang Q, Chen W, Huang X, et al. Data Matrix Code Location Based on Finder Pattern Detection and Bar Code Border Fitting[J]. Mathematical Problems in Engineering,2012,(2012-4-10), 2012, 2012(2):199-210.

[6] Kulshreshtha R, Kamboj A, Singh S. Decoding robustness performance comparison for QR and data matrix code[C]// International Conference on Computational Science, Engineering and Information Technology. 2012:722-731.

[7] Kruchinin A Y. Industrial DataMatrix barcode recognition for an arbitrary camera angle and rotation[J]. CO, 2014, 2014, 38(4):865-870.

[8] WANG, Weiping, Gaifang. Polluted and Perspective Deformation DataMatrix Code Accurate Locating Based on Multi-features Fusion[J]. Chinese Journal of Electronics, 2014, 23(3):550-556.

[9] Wang G, Xiangkun W U, Kun H U, et al. Embedded Recognition of DataMatrix Code by Using Cross Correlation Matching Algorithm and Hough Transform[J]. Journal of Jilin University, 2016, 34(3):419-426.

[10] Xu X, Xu S, Jin L, et al. Characteristic analysis of Otsu threshold and its applications[J]. Pattern Recognition Letters, 2011, 32(7):956-961.

[11] Pye C J, Bangham J A. A fast algorithm for morphological erosion and dilation[C]// European Signal Processing Conference, 1996. Eusipco 1996. IEEE, 2010:1-4.

[12] Leavers V F. Which Hough Transform?[J]. Cvgip Image Understanding, 1993, 58(58):250-264.

[13] Wang J, Ge Y, Heuvelink G B M, et al. Effect of the sampling design of ground control points on the geometric correction of remotely sensed imagery[J]. International Journal of Applied Earth Observation & Geoinformation, 2012, 18:91-100.

[14] Lenk B. Data Matrix ECC 200[J]. Lenk Monika Fachbuchverla, 2007.