An Improved Algorithm of DPM for Two-dimensional Barcode

Tao Gao
North China Electric Power University
Department of Automation
Baoding, Hebei Province, 071003
China
gaotao81@foxmail.com

Xiao-cheng Du
North China Electric Power University
Department of Automation
Baoding, Hebei Province, 071003
China

Iztok Fister Jr.
University of Maribor
Faculty of Electrical Engineering and Computer Science
Smetanova 17, 2000 Maribor
Slovenia
iztok.fister1@um.si

ABSTRACT

Data Matrix is a two-dimensional matrix code which has many advantages like as large information density, high capacity, and small size and so on. It is widely used in industrial automation, logistics, transportation and other fields. A two-dimensional barcode includes printing and DPM barcodes, which depending on the differences of the application at backgrounds of the two-dimensional barcode. DPM is the abbreviation for Direct Part Mark and has some weaknesses, like low contrast, more noise jamming, complicated background, uneven illumination during the data appreciation process. The main mission of this paper was to put forward a series of image preprocessing methods which links binaryzation and morphological transformation based on handheld device, and achieve adaptive smoothing fuzzy and adaptive morphological transform through the DPM detection. The experimental results show that the method can overcome problems such as too large middle gap, the uneven illumination and noise.

Keywords

Data Matrix Barcode, Binary Image Processing, Direct Part Mark

1. INTRODUCTION

Two-dimensional barcode is a neotype barcode technology based on a one-dimensional barcode, and according to criteria stores data and symbolic information on special geometries using black and white, which are distributed along the horizontal and vertical areas of two-dimensional planar space. Due to some specialties regarding significant information storage capacity, good robustness and low cost [1] two-dimensional barcoding has been applied gradually within the commercial, transportation, finance, health care and other fields.

2. DATA MATRIX BARCODE

Presently, the more commonly used international two-dimensional barcode includes the Data Matrix, PDF417, QR code, etc. The Data Matrix has the minimum size of all the barcodes, and is especially suitable for small parts logos and can be printing on to the entity directly, so that the sign is widely used for small objects like drugs, integrated circuits, and manufacturing processes from production lines [2].

2.1 The structure of the data matrix 2D barcode

The Data Matrix of a two-dimensional barcode is shown in Figure 1. Its symbol's structure is composed of position area and data areas. The position area includes an 'L' solid boundary and an 'L' dotted boundary as shown in Figure 2, and has a dead zone with one data unit. Position area is the Data Matrix of border the two-dimensional barcode, and is mainly used for limiting the physical size, orientation and symbol distortion of DM. The 'L' dotted boundary is mainly used for finite unit structure of a symbol as well, which can also help to determine physical size and distortion. As shown in Figure 3, the data area contains the symbols which to be encoded, and contain coded information, like Numbers, letters and Chinese characters according to certain encoding rules. The DM code is the lattice and consists of two kinds of color, a black and white combination, and every black or white square with the same size is a unit of data, representing a binary 1 and a binary 0 [5].

Figure 1: Data matrix of two-dimensional barcode

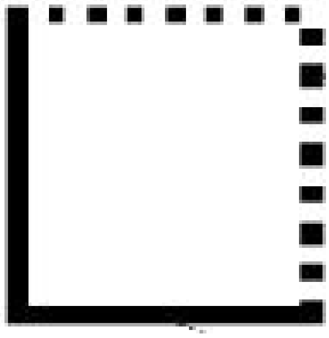

Figure 2: Position area of DM

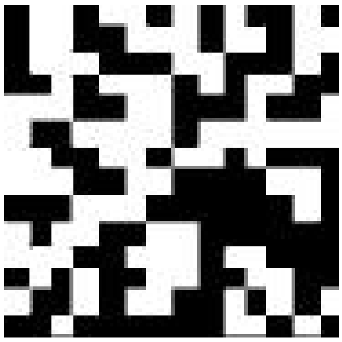

Figure 3: Data area of DM

2.2 DPM barcode

DPM is the abbreviation of Direct Part Mark. It was originally used for machinery and electronics industrial parts, and can record lots of information about parts like production, quality inspection. The factory embosses the DPM into an image by means of an etching method like the laser etching is usage can be extended to automobile manufacturing, pharmaceutical, medical, military firearms management, and so on. So the DPM barcode is a kind of important information within the Internet of technology. The DPM mark is the main carrier of the two-dimensional barcode because the two-dimensional barcode has certain characteristics like large coding capacity, high density, high information security. Compared with the two-dimensional barcode printed on paper, the generating method of the DPM bar code varies, like ink-jet printing. In addition, it can also be generated by certain other methods like laser etching, striking a hitting machine, and electrochemical corrosion. The material of the parts with DPM barcode sculpture varies from cast iron, aluminum, glass, hard plastic, to wood.

3. THE SOURCE CODE

Barcode recognition technology has been widely used, and the main open source code includes 'zxing' open source code and 'zbar' open source code. In this section, the improvement is based on the 'zxing' open source code. In the open zxing source code, there are certain advantages as follows: it can be installed within intelligent phones, and it not only has high speed for identification but also has a short recognition run-time. In addition, it can identify the one-dimensional and two-dimensional barcodes at the same time, and can also search online for products according to the recognized barcode. So it is highly convenient for developers to use this barcode.

3.1 The Decoding Process of The 'Zxing' Open Source Code

In the open 'zxing' source code, it needs to open its build camera first, and then photograph the barcode to be identified. The color model of the image obtained is a RGB color model. However, the RGB color space has some disadvantages like: No intuitive, asymmetric, and dependency on the hardware device [3]. So, the RGB color model should be converted to YUV color model, for the reason that it is important that the luminance Y is separate from the chroma U and V in the YUV color space. The formula for the RGB color model converting into YUV color model is as follows.

- Y=0.299R+0.587G+0.114B
- U=-0.147R-0.289G+0.436B
- V=0.615R-0.515G-0.1B

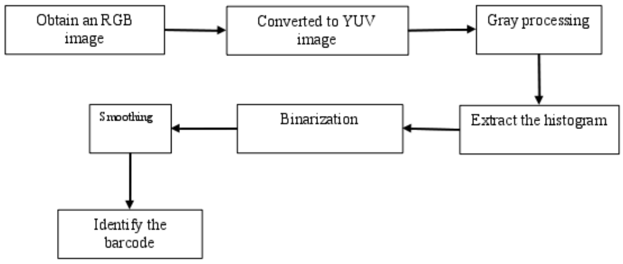

Figure 4: The basic process of zxing source code

After obtaining the YUV image, it is necessary to convert the image into a grayscale image, and then convert it into a binary image. In addition, the histogram needs to be obtained at the same time, for then to identifing the barcode. The description above is the whole barcode image processing of the open source code identification of 'zxing'. The basic process of 'zxing' source code is shown in Figure 4.

3.2 Program flowchart

The method put forward in this paper is based on "Z-Xing" project which is an open source two-dimensional code recognition project of Google, its main purpose is to do some pretreatment to the initial two-dimensional code image. It is achieved the image binarization by referring to the binarization of Yang Shu, and attaching some appropriate process to dotted data matrix bar code processing. The size of the point module will be get by the process of blob detection. Combined with morphological processing the image can be identified, the program flow chart shown in Figure 6. Among them, it is necessary to focus on these folders mainly with the names "android", "camera", "encode", and "result". The process of the initialization program is as follows: firstly, to load the main activity and create the object of the Capture Activity Handler in this class, and then is object starts the camera to realize the automatic focusing. Secondly, to create Decode handler threads, and then create the object of a

Decode handler, and this object then obtains original byte data from the camera, all the above during the first stage. After obtaining the data, it parses out the two-dimensional barcode, and then parses out the character of the barcode, at the same time removing the characters without analysis to be handle by 'Capture Activity Handlers'. This class is called the 'decode function' of the main activity used for completing the analyse of the characters, and the image is finally displayed on the screen. In this way, it completes the parsing of the barcode.

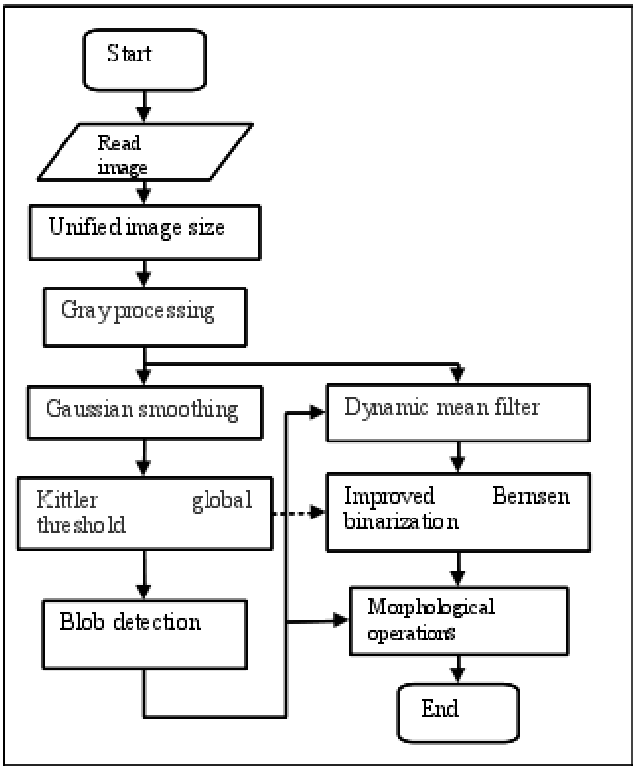

Figure 5: The basic process of zxing source code

4. BINARY PROCESSING OF DATA MATRIX CODES

4.1 The binary image processing

The key to binarization is to define the boundary between black and white. The image has been transformed into gray image, with every point represented by a gray value, which is needed for defining a gray value (namely threshold). If the value of the point is greater than that value, it can be defined as white (1), otherwise, it is defined as black (0).

4.2 The improvement of a binarization algorithm

As shown in Figure 6, in 'zxing' barcode, the binarization processing is implemented in Binarizer class, where the one-dimensional code uses the **get-BlackRow** method, and the two-dimensional code uses **getBlackMatrix** method. There are two classes generated from Binarizer class: **Global-Histogram-Binarizer** and **Hybrid-Binarizer**. The implementations of the **getBlackMatrix** method for these two classes are different, so we modify the binarization process briefly for the **Global-Histogram-Binarizer** class. In the improved **Global-Histogram-Binarizer** class, the threshold is calculated by the kittler algorithm. Using the kittler binarization algorithm, we deal with the gray images of the DataMatrix barcodes during binarization processing, and then the global threshold (T) can be obtained. The kittler binarization processing formula is: For pages other than the first page, start at the top of the page, and continue in double-column format. The two columns on the last page should be as close to equal length as possible.

$$T = \sum_x \sum_y e(x, y) f(x, y) / \sum_x \sum_y e(x, y) \quad (1)$$

In Equation (1), $f(x, y)$ is the original gray image, $e(x, y) = max\{|e_x|, |e_y|\}$ represents the maximum gradient. In $e(x, y), e_x = f(x-1, y) - f(x+1, y)$ represents the horizontal gradient, $e_y = (x, y-1) - f(x, y+1)$ represents the vertical gradient [4]. The relevant procedure is shown in Figure 7.

```
public BitMatrix getBlackMatrix() throws NotFoundException {
  LuminanceSource source = getLuminanceSource();
  int width = source.getWidth();
  int height = source.getHeight();
  BitMatrix matrix = new BitMatrix(width, height);
  byte[] localLuminances = source.getMatrix();
  int threshold = estimateThreshold(localLuminances,width,height);
  for (int y = 0; y < height; y++) {
    int offset = y * width;
    for (int x = 0; x< width; x++) {
      int pixel = localLuminances[offset + x] & 0xff;
      if (pixel < threshold) {
        matrix.set(x, y);
      }
    }
  }

  return matrix;
}
```

Figure 6: The improved binarization process

```
private static int estimateThreshold(byte[] localLuminances,int width,
        int height) throws NotFoundException
{
    int E=0;
    int EF=0;
    for(int y=1;y<height-1;y++)
    {
        int offset = y * width;
        for(int x=1;x<width-1;x++)
        {
            int grey = localLuminances[offset + x] & 0xff;
            int grey1 = localLuminances[offset + x-1] & 0xff;
            int grey2 = localLuminances[offset + x+1] & 0xff;
            int grey3 = localLuminances[offset + x-width] & 0xff;
            int grey4 = localLuminances[offset + x+width] & 0xff;
            int Ex=grey1-grey2;
            int Ey=grey3-grey4;
            int ex=Math.abs(Ex);
            int ey=Math.abs(Ey);
            int exy=Math.max(ex,ey);
            E+=exy;
            EF+=exy*grey;
        }
    }
    int threshold;
    threshold=((EF/E)-1)/255;
    return threshold;
}
```

Figure 7: Kittler algorithm

5. EXPERIMENTS

The process of test uses the Huawei smartphone as a carrier. The main parameters of the phone are as follows: the CPU is mediatek MT6592M, the frequency is 1433 MHZ, memory is 2 GB. We deal with Data Matrix barcode on different specifications, color, clarity, and 34 DPM pictures are tested. In this section, we identified the Data Matrix barcode by soft wares created from the 'Z-xing' package which have been improved and haven't been improved respectively, and compare the recognized effects with each other. The results are shown

Software	Original	Improved
Number of images	34	34
Number of successful images	19	25
Success rate	55.9%	73.5%
Number of images with the brightness difference	17	17
Number of successful images	7	11
Success rates	41.2%	64.7%
Number of DM barcode images (2*2)	5	5
Number of successful images	2	4
Success rates	40%	80%
Number of images sampled from artifacts directly	12	12
Number of successful images	7	9
Success rate	58.3%	75%

Table 1: Comparison

in Table 1. Because the method will be used in handheld device finally, so the method need have more instantaneity. So in the experiment, shoot the image from different angles for 100 times, then count average correct recognition rate and speed. Through analyzing the experimental result, it can be known that the elapsed time is relevant to the contrast, roughness and illumination of image. The method sacrifices efficiency for accuracy because of using improved Bernsen method. The elapsed time is acceptable through comparing the time with the elapsed time of standard DataMatrix. Counting the time that each part spends through simulation, it can be reached that the improved Bernsen method takes more time when the spot detection takes second place. The size of image does not much affect the result because the program adds size unitizing method.

6. DISCUSSION AND CONCLUSION

It can be seen from Table 1, that after improving 'zxing' binarization, the success rates of various types of Data Matrix barcode recognition were improved, especially in the test with the brightness difference, and the effect was very obvious, which embodies the characteristics of the kittler algorithm. It can also more effectively find the area of uneven illumination. The improved algorithm is more stable and adaptive, so as to improve the success rate of recognition. In addition, the improved 'zxing' still has some problems:

- It can be seen from Table 1, that the differences between the success rates are higher, and the difference is around 20 percents, some even higher, and this is due to the pictures being a slightly less. Table 1 can reflect the effect of improvement, but it is not as great as the numbers reflected in the Table.
- In the identification process of 2 X 2 DM code or 1 X n DM code, the recognition rate had improved after the improvement, but certain process of image recognition took a long time, so this kind of barcode recognition effect of the improved 'zxing' is still unsatisfactory.
- When we recognized the pictures that sampled from the artifacts directly, we needed to extract the tested sections manually, otherwise, the recognition effect was bad, in other word, and it was easily affected by the background.

7. ACKNOWLEDGEMENTS

This work is supported by National Natural Science Foundation of China (No. 71102174 and No. 51306058), the Fundamental Research Funds for the Central Universities (No. 2014QN46).

8. REFERENCES

[1] C.-H. Cheung and L.-M. Po. Novel cross-diamond-hexagonal search algorithms for fast block motion estimation. *Multimedia, IEEE Transactions on*, 7(1):16–22, 2005.

[2] T. Jinzhao and Q. Jie. Fast and precise detection of straight line with hough transform. *Journal of Image and Graphics*, 13(2):234–237, 2008.

[3] Q. Q. Ruan and Y. Z. Ruan. *Digital image processing (second edition)*. Electronic Industry Press, Beijing, 2005.

[4] S. Yang and Z.-h. Shang. A new binarization algorithm for 2d bar code image. *Journal of Kunming University of Science and Technology (Science and Technology)*, 1:011, 2008.

[5] Y.-x. Zou and G.-b. Yang. Research on image pre-processing for data matrix 2d barcode decoder. *Computer Engineering and Applications*, 34, 2009.