2020 IEEE/ACM 28th International Symposium on Quality of Service (IWQoS)

A General Approach to Robust QR Codes Decoding

Jiamei Lv, Yuxuan Zhang, Wei Dong, Yi Gao* and Chun Chen
College of Computer Science, Zhejiang University.
Alibaba-Zhejiang University Joint Institute of Frontier Technologies.
Email: {*lvjm, zhangyx, dongw, gaoy*}*@emnets.org,chenc@zju.edu.cn*

***Abstract*—With the continued proliferation of smart mobile devices, Quick Response (QR) code has played an important role in daily life. They may be distorted and partially invisible due to bright spots, folding and stains When they are printed on soft materials such as plastic bags. Existing scanners may fail in detecting and decoding QR codes due to distortion.**

In this paper, we propose a simple but effective approach to decoding distorted and partial QR codes. First, we improve an existing QR code detection algorithm to extract QR codes. Then based on the structural features of QR codes that white and black modules are staggered, we propose a novel distortion correction mechanism that uses an adaptive window to match each module. In order to tackle the problem of invisibility, we print multiple QR codes and capture them in an image. Considering confidence of each module in separate, we reconstruct a relatively complete QR code.

Extensive experiments have been conducted to evaluate the performance of our approach. The results show that our approach improves the decoding rate by 50% - 60% compared to the other two baselines.

I. Introduction

With the increasing popularity of smart mobile devices, Quick Response (QR) code [1] has been widely used in a variety of domains across project tracking, item identification, mobile advertising, payment, and social networking, to name a few. They deliver information through the arrangement of small black and white modules.

Although there exist many commercial scanners for QR code decoding, they basically follow the steps: first, identify the QR codes by the special arrangement of black and white modules; second, get data in order according to the color of each module. At present, decoding QR codes requires people to hold a scanner and face QR codes directly for maintaining their original shapes, which is laborious. When QR codes are printed on soft materials such as plastic bags and become distorted and partial, it is necessary to flatten them artificially, which brings poor quality of user experience. Therefore, a robust mechanism to decode distorted and partial QR code is desperately needed.

In general, there are following two issues.

1) Due to the softness of the plastic bag and the angle between scanner and QR code, the QR code can be severely distorted.
2) Owing to folding, bright spots and stains, some parts of the QR code will not be visible.

There are already some works focus on the robustness of QR code decoding. Some use deep learning algorithms to recover QR codes [2]–[4]. Through a lot of training, they can quickly recover the distorted QR codes. However, their performance will be visibly deteriorating when there are new untrained QR codes. Besides, some use mathematical methods to reconstruct QR codes [5], [6]. They correct distorted QR codes by modeling the distortion mathematically. It is inefficient and error-prone to model every possible distortion of QR codes on plastic bags. In addition, when QR codes are partial, they are ineffective because of the lack of information.

In this paper, we propose a mechanism to decode distorted and partial QR codes on plastic bags which can be divided into three parts.

QR codes extraction. We improve an existing light code detection mechanism to extract QR codes. The subregion extracted not only contains QR code completely, but also retains the shape information of QR code.

Distortion correction. QR codes use black module and white module to represent 0 and 1 respectively. Existing scanners usually use a fixed-size window to match each module in QR codes. When QR codes are distorted, the recorded data may be wrong, resulting in decoding failure. To this end, we propose a mechanism that uses a size-adaptive window to match each module. The key basis is *data masking* defined in QR code specification. In order to make QR more readable, QR code specification defines some mask patterns to change the outputted matrix. After data masking, the numbers of black and white modules in QR codes are equal as possible and the modules of the same color will try to avoid adjacent. Therefore, the window position and size can be adjusted adaptively according to the boundary between modules of different colors.

QR codes combination. As to the problem of partial invisibility of QR codes, the basic idea is printing multiple identical QR codes and combining their visible parts to get complete data. The key point is how to identify the missing area of these QR codes. There are about three causes for data missing, bright spots, stains and folding. For bright spots, we propose a method based on the change of chromaticity to identify the potential area of bright spots. It is easy to identify colored stains. As to black/white stains, we can only infer them by the black/white area. For folding, we utilize boundary slope and the chromaticity change to judge the crease of QR codes.

*Corresponding author. This work is supported by the National Science Foundation of China (No. 61872437 and No. 61772465), and Zhejiang Provincial Natural Science Foundation for Distinguished Young Scholars under No. LR19F020001.

978-1-7281-6887-6/20/$31.00 ©2020 IEEE

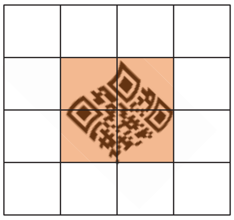

Fig. 1. QR code extracted by the original algorithm [9]

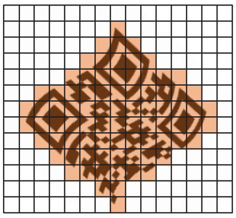

Fig. 2. QR code extracted by improved algorithm

We use a table where each element is the probability of the corresponding data error. Finally, we get the most probable data flow based on the confidence table.

We summarize our key contributions as follows:

- We propose a novel distortion correction mechanism that use a sliding window to match each module in QR codes.
- We propose a QR code reconstruction mechanism that combine multiple QR codes to solve the problem that some parts of QR codes are invisible.
- We conduct extensive experiments to evaluate performance of our mechanism. The results show that our scanner's decoding rate reaches 85%-95%, which is 50% - 60% higher than other two baselines.

The rest of the paper is organized as follows. Section II describes the design of QR code extraction followed by the description of distortion correction in Section III. Section IV is about the step of QR codes combination. Section V shows the evaluation results. Section VI talks about the related work. We conclude in Section VII.

II. QR CODE EXTRACTION

The main workflow of our scanner consists of three parts: first, extract QR codes from the captured image; second, reconstruct QR codes to correct the distortion; finally, combine multiple QR codes to solve the problem of information loss. We will introduce the QR codes extraction in this section.

At present, general scanners use the specific ratio between black and white modules to detect QR codes, which may be invalid in case of distortion. There are some learning-based object detection mechanisms [7], [8] that require high performance computational platform and may bring high cost. We use a light code detection mechanism [9] to extract QR codes.

The mechanism divides the picture into $M \times N$ cells and judge whether each cell is a subregion of QR codes by RGB chromaticity. It can identify QR codes of different sizes by adjusting algorithm parameters adaptively. However, the algorithm is designed to extract each QR code in the batch QR codes image and it may lose some information about shapes that is essential to reconstruct QR codes, as shown in Fig. 1.

Therefore, we make some improvements. We reduce the size of each cell and re-do the QR code extraction in the part extracted by the original algorithm. The lower bound of cell size is related to the version of QR codes. For example, for version-1 QR codes who comprise 21 modules per side, when the lower bound is set to $1/21$ of the side length of the extracted area, the information of shapes is almost satisfying. Since the size of the cell is reduced, it is highly probable that the portion of images in the cell is composed entirely of white pixels, resulting in a high false positive ratio. To this end, we perform k-means clustering on neighbor cells. After improvement, the extraction of the QR code is shown in Fig. 2.

III. DISTORTION CORRECTION

The data in QR code are arranged in a linear format, the change of relative position caused by distortion will lead to a large number of errors. Therefore, we need to correct distortion and put modules on right coordinates. After analysis and verification, we find that most modules in distorted QR codes have the following characteristics.

1) When the fold does not exist, the size change between most adjacent modules is small.
2) The modules remain substantially rectangle though distortion.

Besides, white and black modules are scattered according to the QR code specification. Therefore, we propose an effective distortion correction mechanism that uses a sliding window to roughly reconstruct QR codes. In order to improve accuracy, we first rotate the QR code to the horizontal direction by the shape of subregion extracted in the previous section.

A. Basic Procedure

We use a window that has a similar size to the modules to match each module and record the value in a map. If the currently matching module is black, we record 1 and otherwise record 0. While sliding, the size and position of the window are changing to adapt to distorted modules.

We infer the initial size of the window by the re-extracted area by formula 1.

$$\begin{aligned} l &= \frac{L}{C} \\ h &= \frac{W}{C} \end{aligned} \tag{1}$$

where L, H is the side length of re-extracted area, and C is the number of modules on each side of the QR code. Window with initial size (l, w) slides until there is a module that has a similar size with it and there are no modules of the same color around. Take it as the center, we use breadth-first search (BFS) to traverse its around modules until all the pixels of the code area are traversed. So far, the true rotation angle of QR code is still unknown, which is essential to find the head of data flow. We detect specific patterns, finder patterns, on the corner of QR codes that are constant in different codes to help us determine the rotation angles.

However, they may be partial due to the folding, light spots and stains. Therefore, we locate them by calculating the cosine similarity between normal pattern and the reconstructed image. We divide the reconstructed QR code graph into four subgraphs: upper left, lower left, upper right and lower right and then calculate the cosine similarity between them and the normal finder pattern separately. When the similarity exceeds the threshold th (70 in this paper), the subgraph is considered

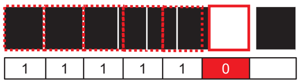

Fig. 3. A data bit is missing because of continuous black modules.

to contain a finder pattern. Have determined the positions of finder patterns, we can get the rotation angle of QR codes to read the data stream.

B. Issues in Practical Scenarios

We implement the above method on 1,000 distorted QR code and the average bit accuracy is only 63.53%, which is far from enough to decode QR codes. Fold and the cumulative error caused by modules of the same color are to blame for the low bit accuracy. We use different mechanisms to solve these two problems.

Fold. When the QR code is folded, modules may be misplaced. In order to address the problem, we use the folded area as a boundary and reconstruct the QR code on both sides respectively. When there are two or more fold areas, we make the modules in the middle of the two folded regions invalid, because it is difficult to determine their correct coordinates. How to determine the folding area will be introduced in section V.

Cumulative error. Actually, our method uses boundary between black module and white module to adjust the size and position of the window. But in the QR codes, continuous black modules or white modules are very common which leads to the accumulation of errors. As shown in Fig. 3, when the width of the black module decreases due to distortion, the width of the window does not change, resulting in missing a data bit. We use two tricks to solve this problem.

- **Anchors strategy.** In a certain version of QR code, the positions of some modules have been determined, such as those belonging to finder pattern, timing pattern, etc. We call these position-determined modules anchors and use them to correct the location of nearby modules. Using anchor correction requires us to know the coordinates of the current module in the original QR code, so we try to use anchor as the initial module, such as module at the finder pattern corner.
- **Feedback mechanism.** We use the feedback from the later modules to correct the sizes and positions of previous modules. We set a threshold for the size change of the window. If the difference in size before and after the window adjustment is greater than the threshold, we believe that a cumulative error may have occurred. We will go back and update the size and position of modules that have not been updated.

  The new module size is defined by the following formula:

  $$S' = S + \delta S/(B_N + W_N) \tag{2}$$

  where S' is the width of next window, S is the width of current window, δS is the width difference between the current window and the module, and B_N and W_N is the number of continuous black and white modules respectively.

IV. QR Codes Combination

A. Overview

We will solve the problem of partial QR codes in this section. Since the missing parts of varying QR codes are almost different, we propose the method of QR code combination to address the problem. We identify the missing parts of QR codes firstly and then calculate the confidence for each module in QR codes. It is worth noting that the confidence values of the modules in missing areas will decrease. Finally, we combine multiple QR codes considering the confidence values. Experiments in section VI have shown that combining 3 QR codes strikes a good balance among computational cost and the decoding rate. Below, we will detail these parts.

B. Missing Area Identification

There are roughly three factors that make the QR codes to be partially invisible, namely bright spots, stains and folding.

1) Bright spots: At present, there have been some work [10], [11] focus on locating bright spots in a single image. The luminance-based methods require some prior knowledge, such as the incident angle of light, which limits their application. In contrast, chrominance-based methods are more widely used. However, almost all of them have two demands making them inapplicable in QR codes. First, the images are required to be colored. Second, images cannot be strongly textured. Therefore, We are committed to propose an algorithm to locate bright spots on QR codes.

According to the Fresnel's law of reflection [12], when light is projected onto a non-homogeneous object, the RGB color $I(x) = [I(x)_r, I(x)_g, I(x)_b]^T$ at the pixel x is the linear combination of the diffuse reflection $I^D(x)$ and specular reflection $I^S(x)$ as shown in formula 3.

$$I(x) = I^D(x) + I^S(x) = m_d(x)\Lambda + m_s(x)\Gamma \tag{3}$$

where $m_d(x)$ and $m_s(x)$ are the surface-geometry-related weights for the diffuse and specular reflections respectively. Λ is the diffuse chromaticity and Γ is the illumination chromaticity. Γ is equal to 0 when there is no specular reflection. Λ is identical for uniform color surfaces, but different for nonuniform or textured surfaces. In other words, when the light is even, the white modules of QR codes have the same Λ while all black modules have a same as well.

Since the diffuse chromaticity Γ and weights for the diffuse reflection $m_s(x)$ are both greater than or equal to 0. It is apparent that a pixel with a specular component has a greater intensity than a pixel without it when their colors are the same. Due to the sensitivity of the camera color filter and the reflected light from surrounding objects, RGB channels of the white modules in the captured image are generally less than 255. In fact, $I_{max}(x)$ is usually less than 200 in natural light, which allows us to distinguish between white pixels with and without specular reflection. If the maximum and average RGB values of a pixel are both larger than the average values of all pixels by a threshold, then pixel x is considered to be a white pixel with specular reflection. We use pixel x as cluster center to cluster. The chromaticity distance between pixels x and pixel y is defined as follows:

$$d(x,y) = \|I(x), I(y)\| \tag{4}$$

If the chromaticity distance is less than a threshold T_C, x and y are belong to the same cluster.

Unfortunately, it is difficult to distinguish between white pixels without specular reflection and black pixels with specular reflection. Thus, we identify bright spots in units of cell whose size obtained in section II. When bright spots appear in some cell, we consider chromaticity of all pixels in this cell to be the combination of specular reflection and diffuse reflection.

2) Stains: The causes of stains may be varied compared with bright spots. We divide all stains into two categories based on their effects.

Colored stains. Colored stains can be easily distinguished from chromaticity. The values of the black/ white RGB channels are very close. Therefore, we use intensity ratio [13], as defined in Formula 5, to help identify colored stains.

$$Q(x) = \frac{I_{max}(x)}{I_{max}(x) - I_{min}(x)}; \tag{5}$$

where

$$\begin{aligned} I_{max} &= \max\{I_r(x), I_g(x), I_b(x)\}; \\ I_{min} &= \min\{I_r(x), I_g(x), I_b(x)\}; \end{aligned} \tag{6}$$

Black stains. When stains color is black, it is difficult to separate them from black modules. Fortunately, QR code is designed to avoid the cluster of numerous modules of the same color in order to improve the decoding rate. Therefore, we can identify potential black stains by judging the area of the black modules. When the black area is larger than the threshold (7 modules in this paper), we mark the modules and lower their confidence values.

3) Fold: Folding destroys the structure of QR code, so we have to reconstruct QR codes considering folding area. It has been observed that fold is usually caused by tying or extrusion, which usually cause a sharp change at the edge of the QR codes. We can roughly judge the position of the folding.

Because folding usually results in changes in the height and orientation of the components on both sides of the crease, there will be differences in the chromaticity between the both sides of the fold.

Band-pass filter is used here to reduce the effect of bright spots and black modules. In our mechanism, we use the two factors of the boundary slope and the image chromaticity change to judge the creases of the QR codes.

C. Confidence Calculation

Missing areas in QR code not only affects the module in it, but also affect its surrounding modules. Therefore, we consider the distance between the module and the missing area closest to it when calculating the confidence of the module. The formula is as follows:

$$c_{i,j} = C - \frac{z}{d+1}. \tag{7}$$

TABLE I
PERFORMANCE UNDER DIFFERENT NUMBERS OF QR CODES.

Number	Attributes	Min	Max	Mean	Media
1	Bit Correct Rate (%)	77.079	100	90.982	90.249
	Latency (s)	0.311	0.593	0.485	0.498
2	Bit Correct Rate (%)	77.778	100	93.769	95.692
	Latency (s)	0.753	1.126	0.889	0.878
3	Bit Correct Rate (%)	84.580	100	96.789	97.279
	Latency (s)	1.138	1.808	1.559	1.585
4	Bit Correct Rate (%)	88.4354	100	97.052	97.506
	Latency (s)	1.729	2.401	1.966	1.903
5	Bit Correct Rate (%)	90.0227	100	98.129	98.979
	Latency (s)	2.319	2.869	2.432	2.355

where i and j are the coordinates of the module, C are fixed values, Z is a variable parameter, d is the distance between the $module_{i,j}$ and the missing area closest to it.

For modules in black stains area, z is the smallest, and for modules in folding area, z is the biggest.

D. Combination

In this step, we combine multiple QR codes considering their confidence values and finally reconstruct a relatively complete QR code. There are two possibilities for each module, black or white. We calculate the probability that they are white and the probability that they are black respectively. The formulas are as follows.

$$\begin{aligned} P1_{i,j} &= \sum_{x=1}^{n} m_{i,j} \cdot q_{i,j} \\ P0_{i,j} &= \sum_{x=1}^{n} (1 - m_{i,j}) \cdot q_{i,j} \end{aligned} \tag{8}$$

where $P1_{i,j}$, $P0_{i,j}$ is the probability of black and white respectively, $m_{i,j}$is the value of reconstructed QR code on coordinates (i,j), and $q_{i,j}$ is the corresponding confidence, n is the number of combined QR codes. When $P1_{i,j}$ is larger than $P0_{i,j}$, we think the current module is black, otherwise it is white.

V. EVALUATION

A. Setup

The testing images are collected by an ordinary mobile phone in a laboratory whose sizes are 550 pixels × 550 pixels.

Decoding rate, the percentage of QR codes successfully decoded, is used here to measure the performance of different scanners. We compare our scanner with a commercial QR code scanner Alipay scanner that has over 1 billion users and the 2D barcode scanner proposed in [5] which use mathematical methods to correct the QR code.

The default configuration of the experiments is as following: the environmental luminance level is around 200 lx; the QR codes are version 1-H; plastic bags are full of things.

B. Number of QR Codes

We conduct experiments to find the appropriate number of QR codes in order to achieve a good trade-off between accuracy and latency. We try to decode QR codes combining

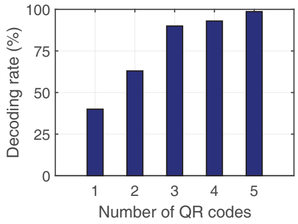

Fig. 4. Decoding rate of different number of QR codes.

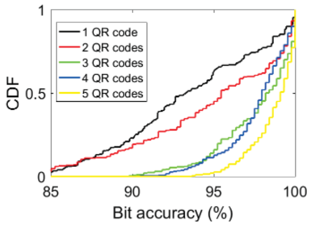

Fig. 5. CDF of bit accuracy of different number of QR codes.

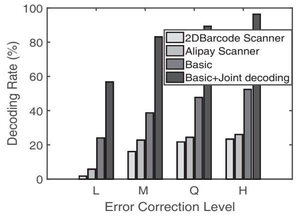

Fig. 8. Decoding rate of different error correction levels.

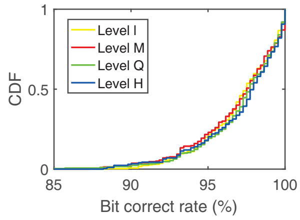

Fig. 9. CDF of bit accuracy of different error correction levels.

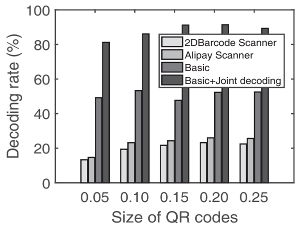

Fig. 6. Decoding rate of different relative size of QR codes.

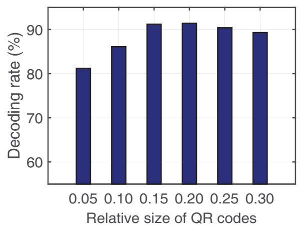

Fig. 7. Average bit accuracy of different relative size of QR codes.

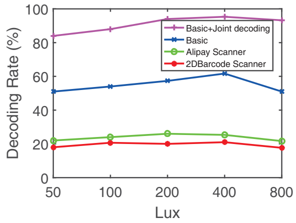

Fig. 10. Decoding rate of different light intensity.

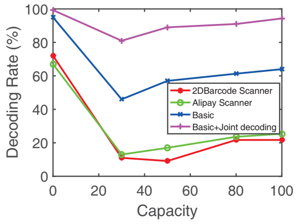

Fig. 11. Average bit accuracy of different light intensity.

two, three, four and five QR codes respectively. Results are shown in Fig. 4 and Table I.

As shown in Fig. 5 , when the number of QR codes for combination increases, so does the bit accuracy. When only using one QR Code, the decoding rate is about 40%. When adding one QR code, there are 30% increase. By adding another QR code, the decoding rate has been increased by another 33%. When the number of QR codes goes up to four or five, the decoding rate increase by only 3% and 5.7% respectively. At the meanwhile, the delay increased by 100%. Considering the trade-off between latency and decoding rate, we believe it is reasonable to combine three QR codes.

C. Different Size

We conduct experiments to verify the effect of QR codes size on our method. We make the width of the plastic bags as 1 and use relative width represents the size of QR code. The results are shown in Fig. 6 and Fig. 7.

The results show that ours can at least keep 60% improvement compared with the other two baselines. The decoding rate and bit accuracy of our method increase as the size increases within a certain range. However, when the relative size exceeds 0.2, the performance is getting worse. After analysis, we believe that the deformation of plastic bags can explain. When the deformation of the plastic bag is constant, the smaller the sizes of QR codes, the larger the proportion of the affected area, thus making the performance worse. However, when the QR codes are too large, there is a risk that they won't be completely captured due to curvature of the plastic bag, reducing the decoding rate. We believe that our method works better when the relative sizes of QR codes are between 0.15 and 0.2.

D. Different Error Correction Levels

QR codes has 4 error correction levels, namely Level L, Level M, Level Q, Level H. The error correction capability increases as the level increases. In order to verify our method's robustness of decoding QR codes of different error correction levels, we experiment with a total of 1200 QR codes images with different correction levels. The results are shown in Fig. 8. The average decoding rate of our scanner is 81.33% that is much higher than the rate of Alipay scanner (19.67%) and 2D Barcode Scanner (15.64%).

It can be clearly seen from the figure that as the error correction rate increases, the decoding rate also increases. As shown in Fig. 9, the bit accuracy variation of different error correction levels is not very obvious. Therefore, the primary factor affecting the decoding rate is the improvement of error correction capability of the QR code itself.

E. Different Luminance Conditions

To evaluate how the environmental luminance conditions affect our method, we do these experiments on 1500 version-1 different QR codes. Under different luminance conditions, our scanners can decode at least 85% QR codes, while the decoding rates of the other two baselines are only about 20%.

When the light is dark, the image distortion will be more obvious due to the sensitivity of the camera, so low decoding rate is inevitable. But when the illumination intensity is about 800 lx, the decoding rate decreases. This is because strong light causes the problem of bright spots, which affects decoding rate to a certain extent. Combination of QR codes can effectively alleviate the influence of bright spots. When the illumination is stronger, the decoding rate of baselines is reduced by about 10% and our method is only reduced by 2%.

F. Different Bag Capacity

When plastic bags are filled with things of different capacities, the shape of QR codes may be affected. In this case, we pack different volumes, 0%, 30%, 50%, 80%, 100% volume of plastic bag, of waste in a plastic bags and test our scheme with 1,500 images.

The results are shown in Fig. 11. The average scan success rate is 93.7% which is much higher than two other scanners.

As shown in Fig. 11, when the bag is not full, the decoding rate is lower. Because when the bag is empty or filled, the surface is flatter and the QR code is less distorted.

VI. Related Work

In recent years, QR codes are widely used in daily life and gain increasing research interests.

Novel barcode design. Some work are aiming at beautifying the visual-unpleasant appearance of QR codes. ARTcode [14] leverages novel techniques on pointillism, shuffling and adaptive color selection to carry information as well as preserve better image quality. Lin et al. [15] use Gauss-Jordane limination procedure and a rendering mechanism to generate QR codes with high quality visual content. Xu et al. [16] propose an approach to automatically produce robust art style QR code by leveraging the CNN-based style transformation network.

At the meanwhile, some work are devoted to improve the robustness of barcode, which are more similar to ours. ShiftCode [17] introduces a two-level reliability technique for intra-frame error correction and inter-frame redundancy. RDCode [18] propose a tri-level error correction scheme: intra-blocks, inter-blocks and inter-frames error correction and provide high reliability to recover the lost block and frames. Megalight [19] propose a novel solution framework for color barcode stream recognition basing on machine learning techniques and achieve high accuracy in color barcode recognition. RainBar [20] propose a progressive code locator detection and localization scheme and a robust color recognition scheme are to hence the decoding rate.

Our work differs from them. We make QR codes robust by printing multiple QR codes in stead of changing them. Since our method follows QR code specification, it can be well compatible with the current QR codes system.

Robust barcode scanner design. Some work has been done to improve the design of scanners rather than barcode. There are some devoted to improving the performance of QR code extraction. some work [8], [21] use neural networks, which require high performance computational platform. Some [22] apply basic morphological operations to extract QR codes, which may be ineffective when QR codes are partial and distorted.

Although these work have their contributions, there are some shortcomings. Firstly, most of the existing work is relatively one-sided and lacks a complete process. Moreover, they are ineffective when decoding QR codes of arbitrary distortions.

VII. Conclusion

QR Code is increasingly used in diverse fields and it is now printed on soft material to deliver information. However, QR codes can be distorted and partial and existing scanners cannot perform poorly.

In this paper, we develop a new mechanism to solve the problem of low decoding rate of QR codes on plastic bags. We reconstruct it using sliding window to deal with the distortion and combine multiple QR codes to solve the problem of information loss. Extensive experiments have been performed to verify the its effect and the results show that it has a very great improvement in decoding rate compared with the other two scanners.

References

[1] "Qr code wikipedia." http://en.wikipedia.org/wiki/QR_code.

[2] C. Ledig, L. Theis, F. Huszár, J. Caballero, A. Cunningham, Acosta *et al.*, "Photo-realistic single image super-resolution using a generative adversarial network," in *Proc. of IEEE CVPR*, 2017.

[3] D. Gong, J. Yang, L. Liu, Y. Zhang, I. Reid, C. Shen, A. Van Den Hengel, and Q. Shi, "From motion blur to motion flow: a deep learning solution for removing heterogeneous motion blur," in *Proc. of IEEE CVPR*, 2017.

[4] M. Zhou, H. Chen, J. Paisley, L. Ren, L. Li, Z. Xing, D. Dunson, G. Sapiro, and L. Carin, "Nonparametric bayesian dictionary learning for analysis of noisy and incomplete images," *IEEE Transactions on Image Processing*, vol. 21, no. 1, pp. 130–144, 2011.

[5] J.-A. Lin and C.-S. Fuh, "2d barcode image decoding," *Mathematical Problems in Engineering*, vol. 2013, pp. 1–10, 12 2013.

[6] K.-T. Lay, L.-J. Wang, P.-L. Han, and Y.-S. Lin, "Rectification of images of qr codes posted on cylinders by conic segmentation," in *in Proc. of IEEE ICSIPA*, 2015.

[7] J. Li, Q. Zhao, X. Tan, Z. Luo, and Z. Tang, "Using deep convnet for robust 1d barcode detection," in *in Proc. of Springer IISA*, 2017.

[8] H. Zhang, G. Shi, L. Liu, M. Zhao, and Z. Liang, "Detection and identification method of medical label barcode based on deep learning," in *in Proc. of IEEE IPTA*, 2018.

[9] B. Jiang, Y. Ji, X. Tian, and X. Wang, "Batch reading densely arranged qr codes," in *Proc. of IEEE INFOCOM*, 2019.

[10] J. Suo, D. An, X. Ji, H. Wang, and Q. Dai, "Fast and high quality highlight removal from a single image," *IEEE Transactions on Image Processing*, vol. 25, no. 11, pp. 5441–5454, 2016.

[11] W. Ren, J. Tian, and Y. Tang, "Specular reflection separation with color-lines constraint," *IEEE Transactions on Image Processing*, vol. 26, no. 5, pp. 2327–2337, 2017.

[12] Shafer and S.A, "Using color to separate reflection components," *Color: Research and Application*, vol. 10, no. 4, pp. 210–218, 1985.

[13] H.-L. Shen and Z.-H. Zheng, "Real-time highlight removal using intensity ratio," *Applied optics*, vol. 52, no. 19, pp. 4483–4493, 2013.

[14] Z. Yang, Y. Bao, C. Luo, X. Zhao, S. Zhu, C. Peng, Y. Liu, and X. Wang, "Artcode: preserve art and code in any image," in *in Proc. of ACM UbiComp*, 2016.

[15] S.-S. Lin, M.-C. Hu, C.-H. Lee, and T.-Y. Lee, "Efficient qr code beautification with high quality visual content," *IEEE Transactions on Multimedia*, vol. 17, no. 9, pp. 1515–1524, 2015.

[16] M. Xu, H. Su, Y. Li, X. Li, J. Liao, J. Niu, P. Lv, and B. Zhou, "Stylized aesthetic qr code," *IEEE Transactions on Multimedia*, vol. 21, no. 8, pp. 1960–1970, 2019.

[17] T. Zhan, W. Li, X. Chen, and S. Lu, "Capturing the shifting shapes: Enabling efficient screen-camera communication with a pattern-based dynamic barcode," *in Proc. of ACM UbiComp*, 2018.

[18] A. Wang, S. Ma, C. Hu, J. Huai, C. Peng, and G. Shen, "Enhancing reliability to boost the throughput over screen-camera links," in *Proc. of ACM MobiCom*, 2014.

[19] T. Zhan, W. Li, X. Chen, and S. Lu, "Megalight: Learning-based color adaptation for barcode stream recognition over screen-camera links," *in Proc. of ACM UbiComp*, 2019.

[20] Q. Wang, M. Zhou, K. Ren, T. Lei, J. Li, and Z. Wang, "Rain bar: Robust application-driven visual communication using color barcodes," in *in Proc. of IEEE ICDCS*, 2015.

[21] D. K. Hansen, K. Nasrollahi, C. B. Rasmussen, and T. B. Moeslund, "Real-time barcode detection and classification using deep learning," in *IJCCI*, 2017, pp. 321–327.

[22] H. Hu, W. Xu, and Q. Huang, "A 2d barcode extraction method based on texture direction analysis," in *in Proc. of ICIG*, 2009.