Reliability Evaluation of Custom 2D Barcode OCR by using Monte-Carlo Simulation

Evgeny Gershikov[1] and Samuel Kosolapov[2]

Signal and Image Processing Laboratory,
ORT Braude Academic College of Engineering, Karmiel, Israel
[1] E-mail: eugeny11@braude.ac.il
[2] E-mail: ksamuel@braude.ac.il

Abstract. Modern smartphones enable fast and easy recognition of a single two-dimensional (2D) Barcode presented to the smartphone' camera. In our current research, which deals with a camera-based Instant Feedback System (IFS), it is required to recognize multiple 2D barcodes presented to the camera. Additional IFS requirement is to evaluate the orientation of the 2D barcode relative to the camera. In order to make recognition of a plurality of barcodes, presented at the same time to the digital camera, feasible and reliable, special custom format of 2D barcode was created. Well-known OCR (Optical Character Recognition) algorithms were adapted to the multiple target recognition. In order to evaluate the feasibility of the selected IFS approach and selected 2D barcode design, MATLAB-based simulations were performed. The simulations covered the whole process starting from the acquisition of an IFS image and up to the recognition of the 2D barcodes. Considering a big number of the simulation parameters, the usage of the classical Monte-Carlo approach is necessary in order to evaluate the accuracy and reliability of the selected 2D barcode design and selected OCR algorithms. A number of practically interesting IFS configurations were analyzed and compared with respect to reliability. We conclude that the proposed IFS method is feasible at simulation level, but requires higher reliability OCR techniques for practical use. The proposed and similar techniques are also examined in real images.

Keywords: Image Processing, Instant Feedback System, OCR, 2D Barcode, Monte-Carlo Simulation

1 Introduction

2D Barcodes are widely used to store short alphanumeric information. A number of 2D barcode standards are known. A typical compact barcode reader contains a small size digital camera, a digital signal processor and an alphanumeric display or certain means for transferring the extracted barcode information to a computer for future processing. Modern iPhones, Android or Windows8 smartphones and tablets have a high quality camera and processing means powerful enough to recognize 1D barcodes and 2D barcode of different formats.

Recently, a number of applications using 2D barcodes were created for smartphones and are available for free download. Most applications of that kind require manual positioning of the camera so that a target containing a 2D barcode will be located inside a certain rectangle of the smartphone's monitor. This approach enables recognition of a single barcode only.

In our current research, which deals with a camera-based Instant Feedback System (IFS), it is required to recognize multiple 2D barcodes presented to the camera.
In the frames of our exemplary IFS, every student in the class gets a flat card containing a 2D barcode. When asked to answer to a multiple-choice question, presented by the lecturer, students raise their cards in a specific orientation (orientation of the label encodes the number of the chosen answer). A digital camera in the lecturer's smartphone (or tablet) is used to take an image of the class and the smartphone' processor is expected to process the image in order to generate a "list of grades" (the list format is student ID, number of the answer, grade).
In order to make recognition of a plurality of barcodes, presented at the same time to the digital camera, feasible and reliable, a special custom format of 2D barcodes was created and well-known OCR algorithms were adapted to the task of multiple target recognition. The barcode format used is described next.

2 Custom 2D Barcode Label

Specific design of our custom 2D barcode contains a 2-digit number in a human-readable form, denoted as Short ID (SID) and a number of color elements (markers) designed to assist the Optical Character Recognition algorithm. The 2D barcode label (2DBCL) is printed on the surface of a flat card, denoted the IFS card. The OCR algorithm must isolate the surface of every 2DBCL from the background and from other labels, register the image of the isolated 2DBCL by using well-known image registration algorithms with the help of the markers and, finally, recognize the SID. Additionally, the orientation of each 2DBCL must be evaluated since it encodes a vote of the card holder, for example, an answer to a multiple choice question.
The 2DBCL used in the first experiments is shown in Fig. 1. As can be seen, the label consists of an outer circle of cyan color with an inner yellow circle. The SID is printed in the center of the barcode label and in this example the SID is 82. Note that the inner circle is truncated on its top side to create a straight line, which is horizontal when the barcode is not tilted. This straight line is used to detect the 2DBCL orientation, which encodes the student vote (chosen answer).

Fig.1 Appearance of the 2DBCL (front side of the IFS card).

Note that the chosen colors were taken due to their simplicity, the ease of the yellow-cyan edge detection and the assumption that these colors are rarely seen in natural photographs taken indoors. However, sometimes the yellow-cyan edge on the side of the 2DBCL is recognized as a straight line due its distortion by the camera optics when positioned at certain angles relative to the camera axis. Thus, we propose using an improved barcode label with another color on sides, as shown in Fig. 2. In this 2DBCL design the magenta color is used for the sides.

The 2DBCL is printed on the front side of the square IFS card. The back side of the card is shown in Fig. 3. Note that this card design corresponds to the case of a multiple-choice question with four possible answers. By positioning the card, so that the chosen answer number points vertically upwards, the student makes his vote. Since this side of the card is towards the student, the camera cannot see the chosen answer and has to decode it using the detected card orientation.

3 Simulation

In order to evaluate the feasibility of the selected IFS approach, MATLAB-based simulations were performed. The simulations covered the whole process starting from the acquisition of an IFS image and up to the recognition of the SIDs and the votes present in it. The block diagram of the simulation units is shown in Fig. 4. As can be seen, there are four main simulation units, which provide creation of synthetic IFS images, and all the steps of the recognition of the SIDs of the simulated voters and their votes. A detailed description of each simulation unit follows.

Fig.2 Improved 2DBCL (front side of the IFS card).

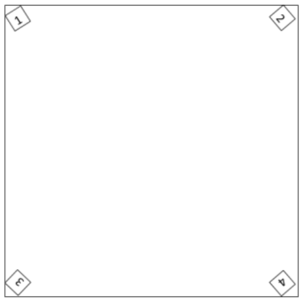

Fig.3 Appearance of the back side of the IFS card.

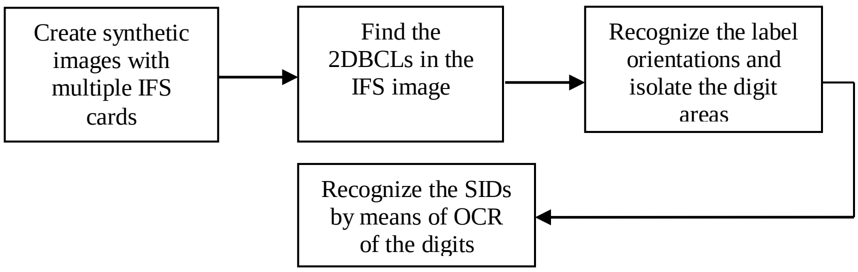

Fig. 4. Block chart of the simulation units. There are four main simulation units (see text for detailed units description).

The first implemented simulation unit creates images of the selected design as seen by a digital camera. Simulation parameters are: the distance from the camera to the 2DBCL (on the Z-axis), the position of the IFS card in the camera's field of view (in the XY plane) and its three 3D orientation angles. In addition to these, typical camera distortions can be also taken into account in the simulation module. For simplicity, we do not include their effects in our simulation. The resulting images can be stored as uncompressed BMP file or as compressed JPG file, whereas image resolution and compression parameters can be controlled by the user. An example of a simple simulated IFS image with two barcodes is shown in Fig. 5.

Fig.5 A simple simulated IFS image with two 2DBCLs.

Note that in Fig. 5 the upper 2DBCL is simulated as further from the camera and both patterns are both rotated (simulating the student votes) and distorted by a projective transform simulating a certain angle of view of the camera.

The second simulation unit we created searches for the markers that allow identifying the barcodes and isolates available 2DBCLs from the background. The simplest way to locate the barcode labels of the design proposed in Fig. 2 is to search for cyan-yellow edges, which are straight lines in the image. For this purpose the edges of the IFS image were calculated (e.g., by Canny's algorithm [1]) and then the straight lines in the edge image were located using the Hough transform [2]. The next stage was to iterate through these lines and keep only those that correspond to an edge between yellow and cyan regions by comparing the real colors on either side of each checked line and the expected colors. A simple way to compare the real and expected colors is to take the normalized inner product of the two RGB color vectors **RealColor** and **ExpColor**, i.e., $\frac{\mathbf{ExpColor}^T \cdot \mathbf{RealColor}}{\|\mathbf{RealColor}\| \cdot \|\mathbf{ExpColor}\|}$ and check if it is above a predefined threshold (for a match) or not. Such a threshold should be a number close to 1, e.g., 0.93.

Once the correct lines were detected, the areas of the 2DBCLs can be located and processed in the following units.

Our third simulation unit recognizes the orientation of the barcode labels and isolates the areas of the digits. The recognition of the orientation of each barcode is simple once the yellow-cyan straight edge has been detected in the previous step. For each such edge there are two possible orientations (for example, the yellow side is at the top and the cyan side is at the bottom or vice versa) and we check the colors to choose the correct one of the two possibilities. Once the orientation is extracted, we can rotate the barcodes so that the digits become horizontal. To detect the digit area correctly without knowing the distance to camera, its view angle and its resolution we apply a simple method based on watersheds [4] to locate the circular area of each 2DBCL (see Figs. 1-2) and use it to determine the bounding box of the digits. The watersheds algorithm is applied to the edges of the blue color component since this was found to provide good performance considering the colors chosen for the 2DBCL design.

The fourth simulation unit provides the last stage of OCR by using simple methods available in MATLAB programming environment. First, the bounding box containing the digits is converted to a binary image and the text in it is separated into its connected components. Each connected component is expected to be a digit and is compared to a database of the 10 digits using the normalized 2-D cross-correlation [3] after resizing all the digits to be of a predefined size. The examined digit is recognized by choosing the database digit providing the highest cross-correlation in the comparison. Two OCR algorithm versions were considered: without image registration of the recognized digit and the compared database digits prior to the calculation of the normalized cross correlation (OCR-1 method) and with it (OCR-2).

The results of the full simulation chain are the SIDs of the 2DBCLs and their original orientations.

4 Monte-Carlo Simulation

Considering a big number of practically required simulation parameters, the classical Monte-Carlo approach was necessary to evaluate the accuracy and reliability of the IFS approach. In the stage of Monte-Carlo simulations, the full chain of simulation units was operated a number of times, where at every simulation run the values of the selected set of setup parameters were modified in a pseudo-random way. The main setup parameters of the simulation are the number of 2DBCLs in the synthetic IFS image, the SIDs used, the orientations of the barcodes (that encode the chosen answers) as well as the viewing angles of the barcodes relative to the camera and the distances from it.

We can summarize our conclusions from the numerous simulations performed in the following bullets.

- The angle of the barcodes relative to the camera should be small, so that the barcode label is not very distorted. Otherwise, while it is still possible to recognize the circular area of the label, the SID digits recognition is not reliable.
- The distance of the barcode to the camera should not be too big and it depends on the camera resolution. Each barcode label should be at least of the size of around 40×40 pixels.
- Any orientation of each 2DBCL is allowed corresponding to the chosen answer of the simulated voter. The algorithm performance does not depend much on the orientations.
- Any SID of two digits can be used. This covers the case of up to 100 students. 3 digits SIDs for bigger classrooms were not tested, but we expect a similar performance since each digit is recognized separately.
- The number of the barcodes simulated in the IFS image does not affect the recognition performance, but has significant influence on the processing time.
- The reliability of the orientation recognition of the barcodes is high enough. Errors up to almost 90 degrees are theoretically allowed in the recognition when considering multiple-choice questions with four possible answers. In our simulations the errors were less than 15 degrees.
- The use of image registration techniques to improve the accuracy of the SID recognition did not justify the added complexity and running time. Although for individual images sometimes the OCR-2 method with registration provided better performance than OCR-1, on average both methods provided similar accuracy of the results.
- The simple OCR methods used are influenced by small distortions of the form of the digits. Other more complex OCR techniques can be considered for future research.

Conclusions

In this work we examined a new camera-based method for Instant Feedback, that can be used in classrooms in institutions of education, such as schools, colleges and universities. The proposed simple IFS system consists of a camera used to acquire a photo of the classroom and a collection of IFS cards with a specifically designed 2D barcode labels printed on them. The barcode was designed to allow reliable detection and recognition of the SID of the cardholder as well as the orientation of the label. This orientation encodes the student's vote when answering a multiple-choice question, asked by the lecturer. Thus, both the SIDs and the orientations have to be reliably extracted from the IFS image captured by the camera. The reliability of our proposed system was tested using Monte-Carlo method of simulations. We conclude that while the reliability of the orientation recognition was good enough, the reliability of the SID recognition was very dependent on the position of the barcodes relative to the camera – when the barcodes were simulated at a big angle relative to the camera optical axis, the recognition accuracy was low. We believe that this result can be improved by applying other more advanced OCR techniques or considering other barcode designs that do not include digits. These ideas are currently under research [5].

References

[1] J. Canny, "A computational approach to edge detection", IEEE Transactions on PAMI, 8(6), 679-697, 1986.

[2] R. O. Duda and P. E. Hart, "Use of the Hough Transformation to Detect Lines and Curves in Pictures", Comm. ACM, 15, 1, 11-15, 1972.

[3] R. M. Haralick and L. G. Shapiro, "Computer and Robot Vision", Volume II, Addison-Wesley, 316-317, 1992.

[4] F. Meyer, "Topographic distance and watershed lines", Signal Processing, 38, 113-125, July 1994.

[5]. S. Kosolapov, E. Gershikov and N. Sabag, "Feasibility of Camera-Based Instant Feedback System", *Think Mind*, Content 2013 Proceedings, 23-29, 2013