QR Code Patterns Localization based on Hu Invariant Moments

Hicham Tribak, Youssef Zaz
Faculty of Science of Tetouan, Abdelmalek Essaadi University
Tetouan, Morocco

***Abstract*—The widespread utilization of QR code and its coincidence with the swift growth of e-commerce transactions have imposed the computer vision researchers to continuously devise a variety of QR code recognition algorithms. The latter performances are generally limited due to two main factors. Firstly, most of them are computationally expensive because of the implemented feature descriptor complexities. Secondly, the evoked algorithms are often sensitive to pattern geometric deformations. In this paper a robust approach is proposed, in which the architecture is based on three distinct treatments among others: 1) An image quality assessment stage which evaluates the quality of the captured image in consideration that the presence of blur decreases significantly the recognition accuracy. 2) This stage is followed by an image segmentation based on an achromatic filter through which only the regions of interest are highlighted and consequently the execution time is reduced. 3) Finally, the Hu invariant moments technique is used as feature descriptor permitting removing false positives. This technique is implemented to filter out the set of extracted candidate QR code patterns, which have been roughly extracted by a scanning process. The Hu moments descriptor is able to recognize patterns independently of the geometric transformations they undergo. The experiments show that the incorporation of the aforementioned three stages enhances significantly the recognition accuracy along with a notable diminution of processing time. This makes the proposed approach adapted to embedded systems and devices with limited performances.**

***Keywords*—QR code; Hu invariant moments; pattern recognition; image blur estimation**

I. Introduction

Quick Response (QR) code or QRC is considered as one of the most used type of 2D barcodes. It is distinguished by its low cost production and its high capacity of storage and robustness towards errors compared to the 1D barcodes and RFID tags. QRC is mainly exploited in several commercial purposes, such as publicity (paper-based hyperlink redirecting to a website), industry (e.g. spare parts monitoring in manufacturing process [1]), remote identification of solar panels [2], and patients' identification in hospitals [3]. In most cases, QRC recognition may be hampered by several factors, among others are: image blurriness and QRC pattern geometric deformations. The former factor is resulting from the fact that the captured images are often taken through cameras which are put onboard of moving devices (e.g. mobile phones, robots). This causes the appearance of blur motion upon the captured images and consequently degrades the image quality, i.e. the QR code available in the image become unrecognizable and undecodable. In order to upgrade the algorithm accuracy, the system captures successively a set of images which in turn undergoes a quality assessment. By doing so, the corrupted ones are discarded and only the one of the highest quality is taken into consideration. For this reason, the 2D Fourier transform is implemented, as it offers significant information about image blur proportion through its valuable properties. As regards the second factor, the geometrical deformations (e.g. rotation, resizing and translation) occur since QCRs are often captured from lateral angles of view and from different distances. This renders their localizations quite difficult. That means the devised algorithm must be able to identify the targeted QRCs even if they are rotated or of irregular sizes.

By analyzing the state of the art of the earlier proposed algorithms of QRC recognition, it turned out that most of them basically rely on a raw 2D scanning process of a captured image in which two categories of positioning symbols (forming QRC) are sought i.e. Finder patterns **FPs** and alignment patterns **APs,** namely, each QRC is exclusively localized thanks to three FPs (laid on its three corners) and at least one AP as depicted in Fig 1. Both FP and AP are characterized by a well-defined texture, i.e. a longitudinal or a transversal section of a FP or an AP is respectively defined by the following ratios 1:1:3:1:1 and 1:1:1:1:1. Generally speaking, a naive localization of the mentioned patterns remains insufficient since the implementation of the said scanning process may produce huge number of false positives, thus, the extracted candidate QRC patterns must be inevitably transmitted to an additional filtering stage. In order to assess the authenticity of the extracted QR code patterns, many approaches have already been proposed, Haar-like features is a powerful object recognition technique which is inspired from the mathematical theorem "Haar wavelets" and the popular Viola–Jones detector as well. In this regards, [4] used a large Haar-like features dataset describing all possible QRC patterns appearance. In fact, the enormous size and complexity of the evoked dataset render the processing time response relatively slow, and thus a variant streamlined method is highly recommended.

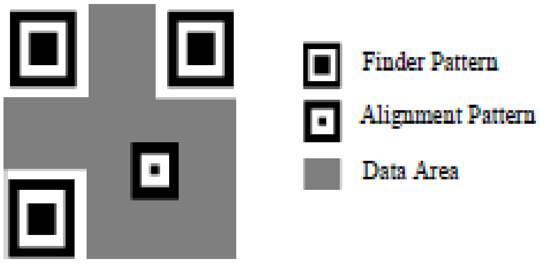

Fig. 1. QR code structure

Our contribution relies on the Hu invariant moments method which is mainly implemented to filter out the set of extracted candidate QRC patterns (resulted from the 2D scanning process). The provided characteristics of the proposed technique meet notably our needs, since it allows representing each pattern by only seven coefficients (which are unchangeable under rotation, scale change and translation) instead of its original structure (whole pattern image). The incorporation of the discussed technique has decreased dramatically the processing time, the limitations that we envisaged previously in our earlier proposed papers [5], [6] have been outperformed. A throughout overview of the proposed system is shown in Fig. 2.

The remainder of this paper is organized as follows. An overall description of the related works is discussed in Section II. The proposed approach is explained in Section III. The conclusions and further improvements are reported in Section IV.

II. Related Works

Sun et al. [7] used canny edge detector combined with an external contour detector which aims at locating the three corners of the QR code. S. Ming et al. [8] implemented an adaptive binarization thresholding to overcome lighting condition changes in addition to Hough transform to locate the corners of the QR code. J. Lin et al. [9] relied on a local binarization in order to simplify image analysis in case of uneven and complex background then a scanning process is performed in which the whole binary image is browsed to extract the candidate QRC patterns. The false positives are removed after calculating the angles that separate the preliminarily extracted QRC patterns. In our previous work [5] the Histogram of Oriented Gradients (HOG) has been implemented. The HOG aims at extracting texture features for the training patterns and the set of extracted ones. These features are then used to feed Support Vector Machine (SVM) classifiers, which indeed allow filtering out the false detected QRC patterns. Likewise, in [6] we used the well-known statistical procedure named Principal Components Analysis (PCA). The latter has been used to convert the pattern image to a set of correlated coefficients. Once the pattern decomposition is ensured, its resulting coefficients are afterwards compared separately with those related to training patterns using the Euclidian measurement as similarity metric. Each pattern having a low similarity are subsequently removed. The construction of a super resolution image has been proposed by Y. Kato et al. [10] with a view to improve QR code recognition in case of low resolution images. This approach is concretized by merging a set of low resolution images to get one of high resolution. L. Huijuan [11] used the Hough transform to extract the vertices points which characterize the four corners of the QR code. These obtained points are then transmitted to an additional stage in which a control point transform and spatial bilinear interpolation are combined. Once carried out, the area containing QR code is deduced. L. N. Zhong [12] proposed a different QR code recognition approach in which the Fourier transform is exploited. This approach is a twofold aim, since it allows both recognizing QRC edges and debluring the input image through the point spread function which is estimated by the Fourier space proprieties. This stage is further reinforced by implementing the edge strength histogram through which the invalid detected edges representing candidate QRC patterns are removed.

Although all the reported approaches advantages, there are still some limitations to deal with, especially alleviating computation time and overcoming geometric deformations sensitivity which hinders most of the QRC recognition algorithms.

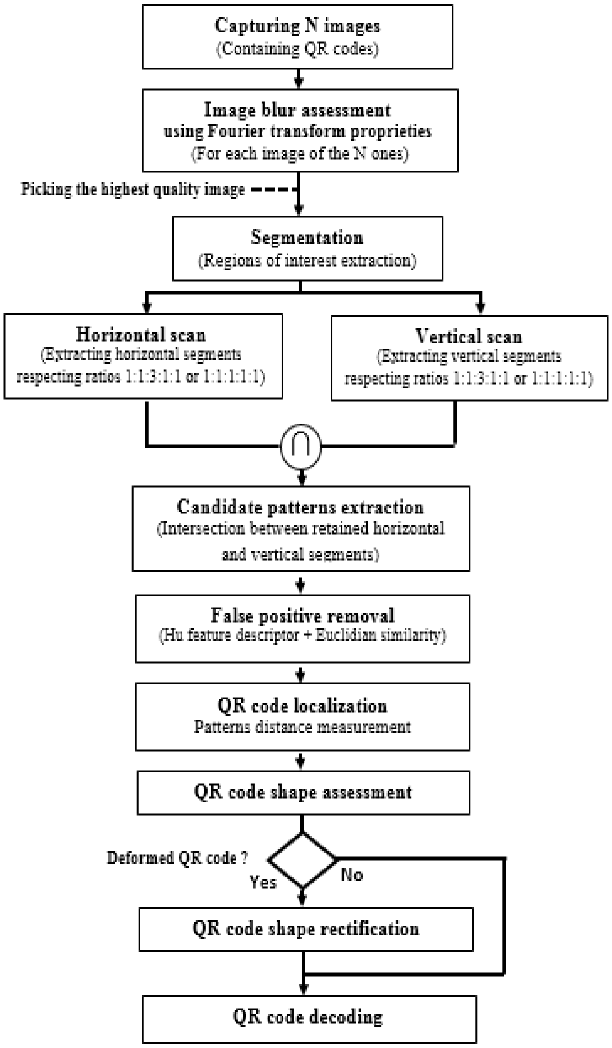
Capturing N images
(Containing QR codes)
Image blur assessment
using Fourier transform proprieties
(For each image of the N ones)
Picking the highest quality image
Segmentation
(Regions of interest extraction)
Horizontal scan
(Extracting horizontal segments
respecting ratios 1:1:3:1:1 or 1:1:1:1:1)
Vertical scan
(Extracting vertical segments
respecting ratios 1:1:3:1:1 or 1:1:1:1:1)
∩
Candidate patterns extraction
(Intersection between retained horizontal
and vertical segments)
False positive removal
(Hu feature descriptor + Euclidian similarity)
QR code localization
Patterns distance measurement
QR code shape assessment
Deformed QR code ?
Yes
No
QR code shape rectification
QR code decoding

Fig. 2. Overall chart of the proposed algorithm of QR code recognition.

III. Proposed Approach

A. Image Blur Measurement

One among the most important pre-processing stage that can be incorporated in pattern recognition systems is manifested in image blur measurement which guarantees selecting the image with the highest quality. The blur measurement algorithms are basically devised to deal with two common types of blurs i.e. motion and defocus blurs. As to enhance recognition accuracy, blurred images must be removed before proceeding to the recognition process. Generally speaking, blur metric approaches can be classified into three categories: Full-reference (FR), Reduced-reference (RR) and No-reference (NR). The former approach requires mainly a reference image with which the captured image is compared to estimate the blur proportion. This technique is not efficient, in consideration that QR codes can be placed in different environments thus the use of reference image turns out useless. In contrast, the no reference approach seems useful seeing that the captured image is directly assessed, whether through its spatial representation or from its frequency space. By returning to the state of the art, various image blur assessment approaches have been proposed, in which the conception is inspired from different techniques. P. Marziliano et al. [13] based on edge characteristics analysis in which the coordinates of the begging and ending points of each extracted edge are calculated. The difference between the two extremities allows determining the edge width, namely, an edge with large width is labelled as smoothed. Furthermore, an image containing several smoothed edges is automatically considered as blurred. The drawback of the discussed method is manifested on that in case of noise, a huge invalid edges will be detected, and consequently, a wrong estimation will be conducted. F. Roffet et al. [14] used an input image as reference along with its corresponding blurred versions. The latter are obtained after applying different low pass filter to the input image, then the intensity variations is calculated between the input image pixels with those related to each blurred image. The higher the variations are, the higher the blur proportion is. This method becomes useless in case of image artifacts. R. Bora et al. [15] proposed a block-based blur estimation, in which the image is divided into blocks. Afterwards, the blocks gradients and magnitude directions are calculated. X. Marichal et al. [16] studied a global blur metric through the high frequency coefficients of the Discrete Cosine Transform. A blurred image is the one whose is the high frequency coefficients are close to 0. K. De et al. [17] used the 2D Fourier transform to estimate blur proportion and image quality by means of an overall rate. The latter is obtained after dividing the total number of pixels that comply with a predefined threshold by the total number of image pixels. Giving that determining an optimal threshold turned out complicated, our method is based on a variant technique, i.e. the high and low frequencies of the image Fourier representation are separated by a circular separator as shown in Fig. 3(g). The presence of blur can be deduced if the proportion of low frequencies is much higher than the high ones. Basing on this principle, our approach is conducted as follows: The system captures a series of images targeting the same scene (containing QR codes). A sample of captured images with different blur rates is shown in Fig. 3(a), (b) and (c). In order to select the best quality image from the N captured ones, the latter are all converted to the frequency representation by means of the 2D Fourier transform. By doing so, each image is thereafter represented by a set of complex coefficients. By applying a specific function to each frequency image, the evoked coefficients are rearranged in such way that the low frequencies are shifted to the image center, whereas the high frequencies are placed away from the center. Afterward, the newly arranged coefficients are once again converted through the expression (1) as to obtain the corresponding magnitude images Fig. 3(d), (e) and (f) related to each frequency image. Once achieved, a circular mask is applied over each magnitude image separately. As aforementioned, the evoked mask aims at separating frequencies into two categories, high and low frequencies. For each magnitude image, the proposed algorithm counts the total number (denoted γ) of high frequency coefficients. The best quality image is the one with the highest factor γ.

$$m_{(x,y)} = \sqrt{a_{(x,y)}^2 + b_{(x,y)}^2} \tag{1}$$

Where, $m_{(x,y)}$ stands for a magnitude coefficient. a and b represent respectively the real and imaginary parts of the complex number.

By analyzing the obtained results in Fig. 3, one can observe that when the blur rate increases Fig. 3(a) to (c), the high-frequency number shrinks Fig. 3(d) to (f). The white inclined line in Fig. 3(e) and (f) provides significant information about the camera motion direction. More precisely, the motion direction is simply defined as the perpendicular line on the said inclined one. It should be noted that several algorithms rely on the evoked direction as to deblur and enhance the quality of the blurred image, however, this processing is computationally expensive and requires additional parameters as well.

B. Searching Space Reducing

In pattern recognition systems, the main challenging aim is manifested in how to boost recognition relevance while reducing computation time. The searching space limitation is one of the most important processing that must be beforehand incorporated into the recognition system. This processing aims at picking only the most relevant regions of the image that can contain QRCs. By deeply studying a collection of techniques which have focused on searching space reducing, they can be divided into two main categories: Shape-based approach and color-based-approach. The former is based on shape analysis by focusing on a particular geometric forms such as lines, vertex points, circles, etc. In this regard, several algorithms have been proposed, among others, [18] based on hull convex algorithm combined with vertex points extraction. This combination ensures detecting the areas containing acute edges which characterize QRC texture. G. Klimek [19] used the PClines line detector which allows extracting parallel straight lines. This method is intrinsically based on a parallel coordinate system that runs faster than Hough transform. As for the second technique, it is preferably implemented when the sought patterns are distinguished by a well-defined colors. In [20] the input RGB image is converted to the HSV color space. The latter permits finding the exact location of the sun through the values of the third component "V". This component allows finding the set of pixels having the higher brightness. The morphological filters (top-hat and bottom-hat) are used in [21] as to extract whether bright pixels inside dark background or vice versa. Other methods perform comparisons between the RGB color channels of each pixel of the image. These comparisons permit determining the most likely areas which may contain QRCs (black and white areas), i.e. only pixels whose three components R, G and B equal 255 or 0 are retained. Seeing that the QR code is characterized by two basic colors (white and black), it is interesting to use an achromatic filter. In this context, S.M. Bascon et al. [22] proposed a twofold aim filter which according to the set parameterization, it allows whether extracting achromatic areas or chromatic ones. The evoked filter is defined by (2). Once applied to the captured image, a set of achromatic regions are extracted. These regions are represented by the black areas in Fig. 4(b).

$$\mathrm{T}_{(\mathrm{X,Y})} = \frac{|R_{(\mathrm{X,Y})} - G_{(\mathrm{X,Y})}| + |G_{(\mathrm{X,Y})} - B_{(\mathrm{X,Y})}| + |B_{(\mathrm{X,Y})} - R_{(\mathrm{X,Y})}|}{K}$$

$$\mathrm{I}_{(\mathrm{X,Y})} = \begin{cases} 0 & if\ T(x,y) < 1 \\ 1 & otherwise \end{cases}$$

Where, R, G and B stand for red, green and blue RGB color channels, respectively. K is the color rate extraction (empirically set at 30). $\mathrm{I}_{(\mathrm{X,Y})}$ is the resulting classification value related to the treated RGB pixel. It is set to 0 if the pixel is achromatic otherwise it is set to 1.

After extracting a set of interest regions, the QRC recognition algorithm will focus only on the said regions instead of treating the whole image. By doing so, the processing time is dramatically decreased. As to illustrate the importance of the incorporation of the discussed filter, it has been tested on a sample of images of different sizes. The obtained results are shown in the curve below (Fig. 5), in which the abscissa axis represents the four tested images (of different sizes), whereas the ordinate axis refers to the corresponding required executing time. The orange curve reflects the average executing time when the whole images are treated. The blue curve represents the executing time when the QR code recognition scanning settles only for the interest regions. On the basis of the gotten results, one can observe that the processing time has notably been reduced when the scanning process focuses only on the interest regions.

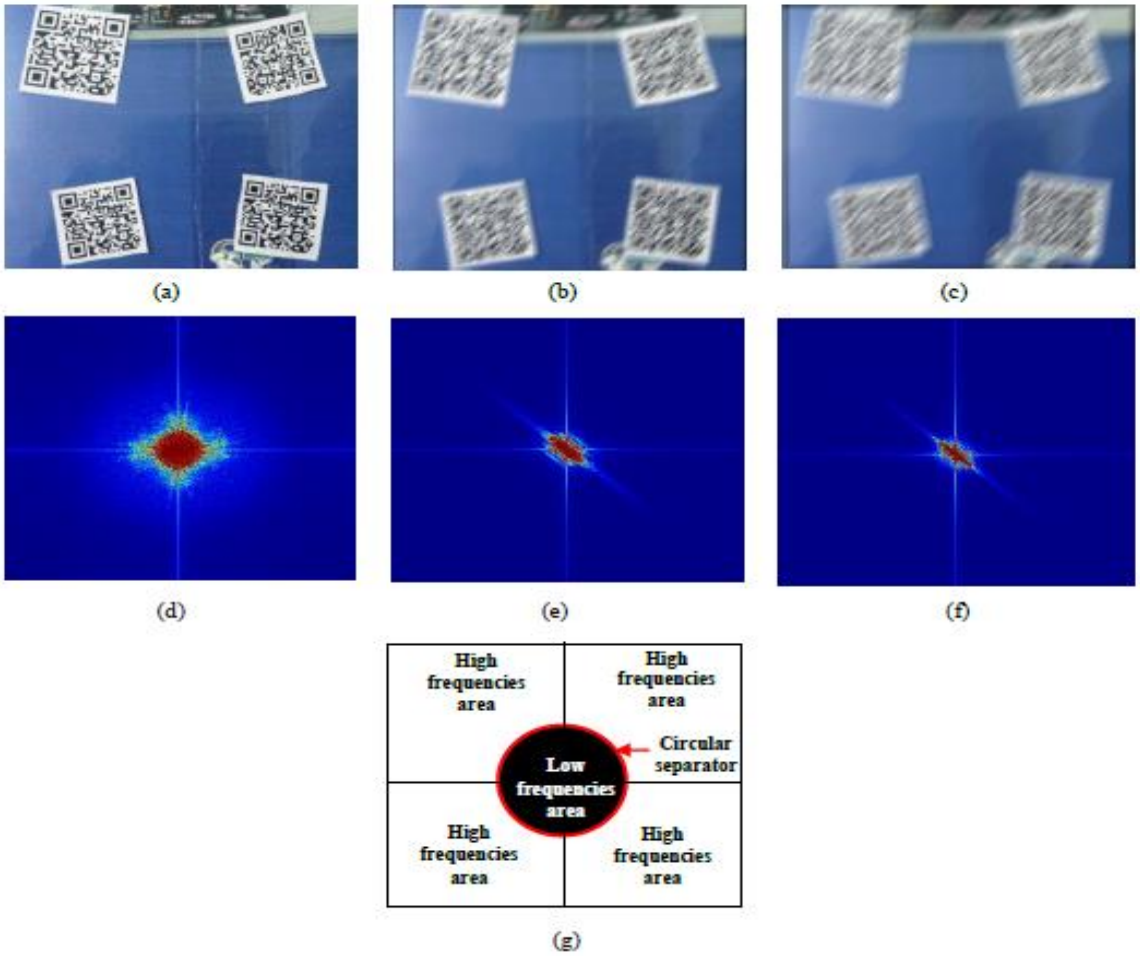

Fig. 3. Sample of captured images accompanied with their corresponding magnitude images. (a) Original image. (b) Blurred image with a blur factor α. (c) Blurred image with a blur factor 2α. (d), (e) and (f) Refer to the corresponding magnitude images related to the images (a), (b) and (c), respectively. (g) Represents the used circular mask which allows separating the high and the low frequencies.

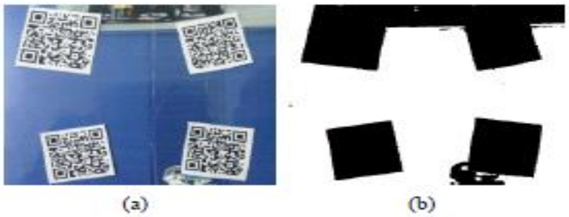

Fig. 4. Regions of interest extraction. (a) Original image. (b) Resulting binary mask in which the black areas refer to the regions of interest which may contain QR code.

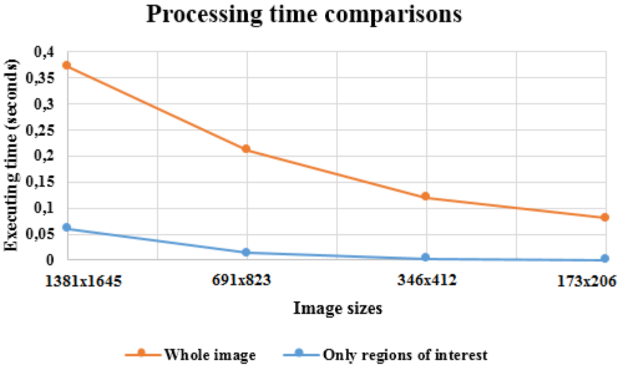

Fig. 5. Executing time comparisons between the whole image processing and interest regions one.

C. QR Code Patterns Recognition

As described in its patent, the QRC is generally localized, thanks to three finder patterns (FPs) and at least one alignment pattern (AP). The FPs are three position indicators which are placed on the three corners of the QRC. FPs are mainly used by the barcode scanner to preliminary detect the location of the QRC. The AP is an additional pattern which helps determining the orientation and geometric deformation of the QRC. Both FP and AP are distinguished by a specific textures i.e. a longitudinal or a transversal section of a FP or AP are respectively defined by the ratios B:W:BBB:W:B (also defined by 1:1:3:1:1) and B:W:B:W:B (i.e. 1:1:1:1:1), where B stands for a succession of black pixels, and W for a succession of white pixels.

Before skipping to the QRC localization stage, a series of important processing must be performed, in which the aim is manifested in simplifying image texture analysis. For this reason, each extracted interest region is converted from the RGB color space to the binary representation. This conversion is ensured by means of two consecutive conversions, i.e. grayscale conversion and binary one. These conversions are respectively defined by (3) and (5). Since the binarization conversion is mostly sensitive to the over and under illuminations, a contrast balancing is inevitably required to be performed for the grayscale representation of each interest region and this before skipping to its binary conversion. In order to attain this purpose, the Look-up table (LUT) transform (4) is implemented. This transform is widely used in case of poor contrast as it allows rearranged optimally the distribution of the grayscale intensities in a way that a set of intensities **are** shifted as possible to the maximum value 255 (for the set of intensities that are greater than the mean intensity) .The rest of intensities are shifted to the minimum one 0 (for the set of intensities which are less than the mean intensity), in other words, once applying the LUT transform to the grayscale histogram of an interest region, it will be observed that the distances between the grayscale intensities have been increased and consequently the image contrast is enhanced. In order to illustrate the impact of the LUT transform integration, let us take a real example (Fig. 6). In Fig. 6(a) a raw grayscale interest region is illustrated, in which the contrast is unbalanced due to the over-illumination. Its corresponding histogram is displayed in Fig. 6(c), wherein the grayscale intensities are clustered in a narrow range (80-190), whereas as two other ranges (0-80 and 190-255) are still empty (without any intensities). As shown in Fig. 6(e), when the contrast is unbalanced, the resulting binary image texture is highly degraded i.e. the appearance of black and white holes (noise) upon the generated binary image, the edges are weak and significantly corrupted as well, thus, the QRC patterns cannot be detected by the traditional scanning process (which looks for pixels section respecting the aforementioned ratios).

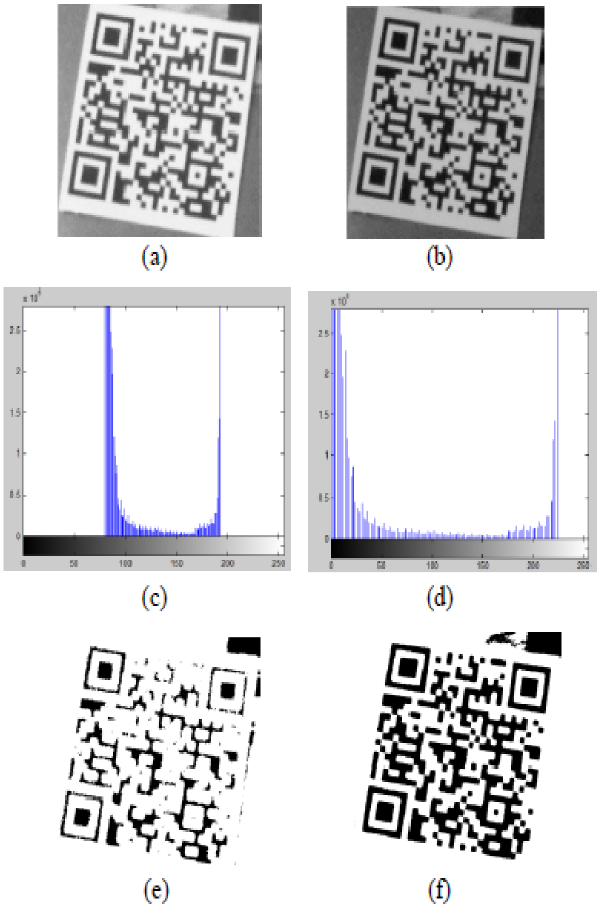

Fig. 6. An example of interest region binarization before and after implementing the LUT transform. The left column represents the grayscale interest region accompanied with its corresponding histogram and binary representation before performing the LUT. The right column refers to the same image after undergoing the LUT transform.

Fig. 6(b) shows the obtained enhanced interest region after applying the LUT transform to the corrupted image. Now, it is clear that the contrast has been improved. The corresponding histogram Fig. 6(d) in turn shows that the new intensities have been spread all over the abscissa axis and cover more area compared to the previous histogram (before performing the LUT). According to the newly generated binary image Fig. 6(f), the edges and texture are significantly enhanced, since they become strong and prominent enough to be easily analyzed by the evoked scanning process. Once all the interest regions are binarized, the image to be treated becomes as shown in Fig. 6.

$$I_{Gray}(x,y) = \frac{\alpha.R(x,y) + \beta.G(x,y) + \theta B(x,y)}{3} \qquad (3)$$

Where, α, β and θ stand for weighting coefficients which are respectively set at 0.2126, 0.7152 and 0.072. R, G and B stand for the three RGB color components of the treated pixel.I_{Gray} stands for the obtained grayscale pixel intensity.

$$I'_{Gray}(x,y) = \frac{255}{Max-Min}.(I_{Gray} - Min) \qquad (4)$$

I'_{Gray} stands for the new grayscale pixel intensity after implementing the LUT transform. Maximum and minimum refer, respectively to the maximum and minimum grayscale intensities of the raw grayscale interest region image.

$$I_{Binary}(x,y) = \begin{cases} 0 & if \ \ I_{Gray}(x,y) < Threshold \\ 1 & Otherwise \end{cases} \qquad (5)$$

I_{Binary} stands for the binary pixel value. The Threshold (as defined by (6)) represents the grayscale constant value which is used in binarization decision.

$$Threshold = \frac{1}{N}\sum_{x=0}^{l-1}.\sum_{y=0}^{c-1} I_{Gray}(x,y) \qquad (6)$$

N stands for the total number of pixel of the interest region to be treated. l and c represent respectively the number of rows and columns of the interest region.

So far, five interest regions have been extracted thanks to the aforementioned achromatic filter and subsequently binarized. In this stage, a scanning process is launched in which each interest region is browsed separately as to check if it contains QRCs. To achieve this task, each binary region undergoes two orthogonal (horizontal and vertical) scans. A horizontal scan browses each interest region row and retains each valid horizontal segment (denoted **Hi**) whose the structure is in conformity with the constraint (7), while accepting a slight difference, i.e. a valid segment is a segment whose the structure is similar to one of those in Fig. 8. Afterwards, the coordinates (beginning and ending pixels) of each retained segment are memorized in a specific matrix denoted $\mathbf{R_H}$. As shown in Fig. 9(a), in which a QRC model is used to illustrate the expected results, a throughout overview related to the set of retained horizontal segments is depicted by means of the red areas. As for the vertical scan, each binary interest region is scanned vertically. This process extracts the set of vertical valid segments (denoted **Vi** and having the same structure as represented in Fig. 8) which comply with the constraint (8). Idem, the retained vertical segments coordinates are saved in another matrix denoted $\mathbf{R_V}$. The expected retained vertical segments are defined by the red areas in Fig. 9(c). Once the scanning process is achieved, the system calculates the intersection (defined by the expression 9) between the two scans results, through which the candidate QRC patterns (FPs and APs) are preliminarily localized. The intersection results are alike depicted in Fig. 9(b), wherein the red areas stand for the square center of each QRC pattern. By performing the discussed scanning process to the earlier binarized image (Fig. 7), the candidate extracted patterns can be summarized by the red bounding boxes shown in Fig. 10. Despite the obtained results, the used scanning process still is a traditional and obsolete technique which lacks precision, since it analyzes naively the texture. By returning to the candidate extracted patterns in Fig. 10, it turns out that in addition to the true positives, number of false detected ones have been extracted and consequently the use of an additional filtering stage has to be inevitably integrated.

$$Hi \approx 1:1:3:1:1 \text{ or } 1:1:1:1:1 \qquad (7)$$

$$Vj \approx 1:1:3:1:1 \text{ or } 1:1:1:1:1 \qquad (8)$$

Where, Hi and Vi represent the valid horizontal and vertical segments respectively. The symbol ≈ means that a slight difference (with the two ratios) is accepted. i denotes the segment index.

$$R_H \cap R_V = \{p(x,y) \in P_{ROI} | \ (p(x,y) \in R_H) \wedge ((p(x,y) \in R_V)\} \qquad (9)$$

Where, R_H and R_V stand respectively for the matrices containing the retained horizontal and vertical segments. p(x,y) denotes a pixel which belongs to R_H and R_V. P_{ROI} represents an interest region.

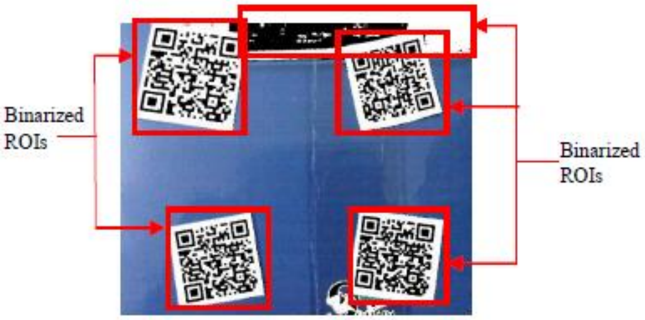

Fig. 7. Resulting regions of interest after being binarized.

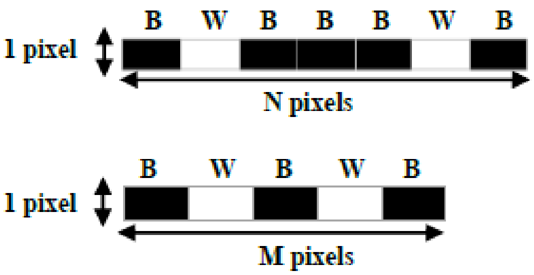

Fig. 8. Valid segments structure. (a) FP segment. (b) AP segment.

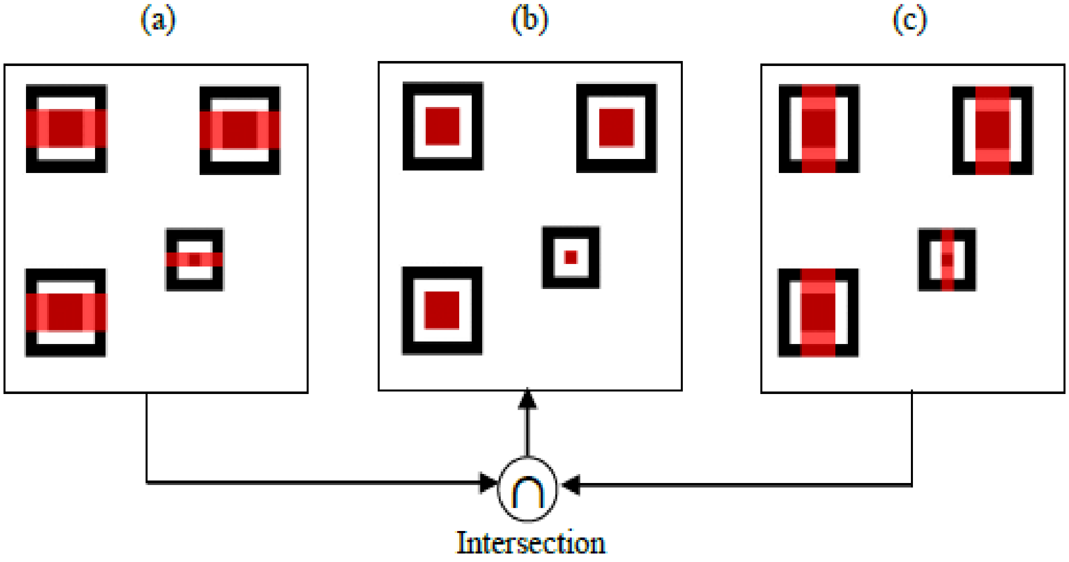

Fig. 9. Overview results related to the scanning process represented on a QR code model. (a) Set of retained horizontal segments (red rectangles). (b) Horizontal and vertical segments intersection allowing extracting the central squares of the potential FPs and AP (red squares). (c) Set of retained vertical segments (red rectangles).

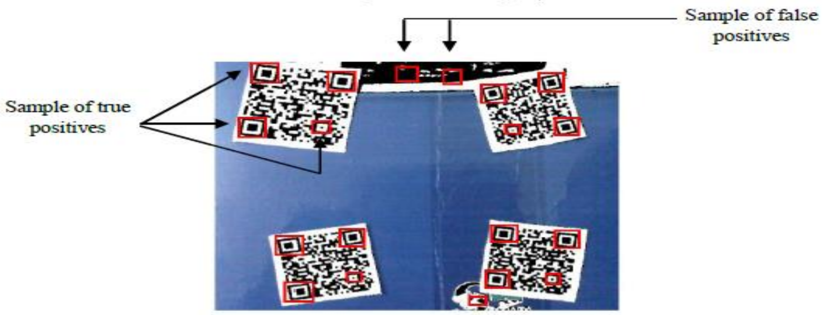

Fig. 10. Extracted candidate patterns.

D. *False Positives Removal Algorithm Based on Hu Invariant Moments and Pattern Similarity Measurement*

1) Overall description

In order to enhance their accuracies, most of pattern recognition algorithms incorporate an additional assessment stage after extracting a set of candidate patterns. This stage aims mainly at removing all sorts of patterns that do not match certain criterion. The most widespread approach consists of using tree classifiers among others, Random forests, K-d trees. Statistical-based approaches such as image correlation and Principal Components Analysis. Or, binary classifiers e.g. linear Support Vector Machine.

In the previous stage, number of candidate patterns (including true QR code patterns and false positives) have been extracted. In order to remove all irrelevant extracted patterns, the two-dimensional Hu invariant moments method combined with Euclidian similarity measurement have been implemented. Mathematically speaking, the seven Hu invariant moments are obtained after resolving a series of equations as explained in the next section (D.2). The evoked moments are widely implemented as image feature descriptor, in which the resulting seven features are invariant under three types of transformations, i.e. rotation, translation and scale change. That means whatever the pattern shape, it maintains the same seven Hu moments even if it undergoes the mentioned transformations. The combination between HU moments descriptor and Euclidian similarity measurement provides a robust and streamlined pattern classifier. The architecture of the latter can be divided into four basic steps (as displayed in Fig. 11). **Step 1** consists of preparing an input data compounded of training data and testing data. Due to the lack of any dataset describing QR code patterns, we had to create our own-made dataset adapted to our needs. The exploited training data is compounded by a set of images referring to two pattern classes, i.e. Finder patterns class and Alignment patterns class. The evoked pattern images have been taken under different lighting conditions and have undergone various geometrical deformations. By doing so, the classification precision is further boosted, since the training data becomes rich enough and credible to deal with different pattern aspect. In order to simplify its manipulation (comparison with extracted patterns), the training data images are all binarized. As regards the testing data, it contains the set of extracted patterns to be filtered. The testing data images are in turn converted to the binary representation. By passing to the **Step 2**, the binary images of the two datasets are transmitted to the Hu moments descriptor through which each binary image is decomposed into seven features (invariant moments). This decomposition allows significantly decreasing pattern comparison complexities, since the pattern image is henceforward represented by only seven coefficients instead of the whole image. The resulting Hu descriptor features are structured into two separate matrices i.e. training patterns features matrix and extracted patterns features matrix. The first matrix (denoted Hu1) is of size Nx7 wherein the index N (number of matrix rows) equals the total number of training data patterns, furthermore, each row of Hu1 refers exclusively to a given training pattern. The second index 7 (number of matrix columns) stands for the resulting seven moments related to each training pattern. Likewise, the second matrix (denoted Hu2) is of size Mx7, where M stands for the total number of extracted pattern. In the example above (Fig. 10), the index M equals 19. This matrix contains the obtained invariant moments related to the queried extracted pattern. More precisely, each row of Hu2 represents an extracted pattern. At the end of the Step 2, the two matrices are transmitted to the **Step 3**, in which each extracted pattern moments (i.e. each row of Hu2) are compared separately with the corresponding seven moments of each training pattern (i.e. each row of Hu1), namely, each two patterns comparison (extracted pattern moments with a training pattern ones) is accompanied with a similarity rate calculation. The patterns comparisons are conducted by means of Euclidian similarity measurement which is defined by (22). By achieving all patterns comparisons, each extracted pattern will be provided by a set of similarity rates. The number of rates of each extracted pattern equals the number of comparisons (i.e. number of training patterns). The Step 3 output is a sort of a matrix named Similarities Matrix (**SM**) whose the size equals MxN (M is the total number of extracted patterns and N is number of training patterns). Each row of SM refers to an extracted pattern whereas each column of it refers to a training pattern, furthermore, the intersection of each row and column represents a similarity rate between an extracted pattern and training one. The matrix SM is in turn transmitted to the **Step 4** wherein patterns classification and filtering are made, in such a way that each extracted pattern similarities are assessed independently. According to the proposed approach rule, a pattern is considered as a false positive if any of its similarity rates is less or equal to a fixed threshold. An overview diagram of the discussed stage is explained in Fig. 11.

2) *Database and training data*

Due to the lack of any database descripting QR code patterns (i.e. FPs and APs), we were obliged to create our own-made database. Indeed, the evoked database contains two distinct pattern classes. The first class is compounded by a set of FP images (hundreds of images) which have been taken under different conditions. These conditions allow describing the common deformations (e.g. under and over illuminations) that can disturb significantly QR code patterns localization. By doing so, the used database becomes rich enough to enhance the recognition accuracy. As to the second class, it contains a set of AP images which in turn have undergone the same tuning as the previous class.

3) *Mathematical definition of the Hu invariant moments*

Basing on the fundamental theorem which has been published in [23], the two-dimensional seven invariant moments are obtained after successively resolving a series of equations, the latter can be divided into four categories. **(a)** Ordinary moments of order i+j which are obtained by (10). **(b)** Centroid components which are represented by $\bar{x}$ and $\bar{y}$ whose corresponding equations are respectively defined by (11) and (12). **(c)** Central moments (denoted U_{ij}) which are defined by (13). These moments are characterized by the ability to remain invariant even in case of translation. **(d)** Normalized moments (Denoted η_{ij}) which are obtained after dividing each central moment by U_{00}^{Exponent}. This normalization is defined by (14). The normalized moments are invariant with respect to scale change i.e. a pattern maintains the same normalized moments n_{ij} even if it is rescaled. **(e)** The seven invariant moments denoted Ii (for i=1,…,7) are defined by the seven equations [(15) to (21)]. It is worthwhile to note that this category of moments is invariant under three types of transformations i.e. Translation, rotation and scale change.

$$M_{ij} = \sum_{x=1}^{m} \sum_{y=1}^{n} x^i y^j I(x,y) \tag{10}$$

Where, m and n are respectively the number of image rows and columns. I(x,y) stands for the binary pixel value.

$$\bar{x} = \frac{M_{10}}{M_{00}} \tag{11}$$

$$\bar{y} = \frac{M_{01}}{M_{00}} \tag{12}$$

Where, $\bar{x}$ and $\bar{y}$ stand for the gravity center coordinates related to the treated pattern.

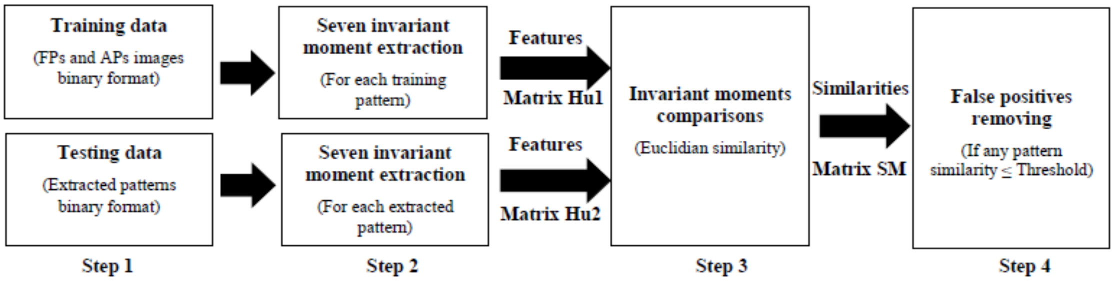
Training data
(FPs and APs images binary format)
Testing data
(Extracted patterns binary format)
Seven invariant moment extraction
(For each training pattern)
Seven invariant moment extraction
(For each extracted pattern)
Features
Matrix Hu1
Features
Matrix Hu2
Invariant moments comparisons
(Euclidian similarity)
Similarities
Matrix SM
False positives removing
(If any pattern similarity ≤ Threshold)
Step 1
Step 2
Step 3
Step 4

Fig. 11. Overview of the proposed false positives removing process.

$$U_{ij} = \sum_{x=0}^{m-1} \sum_{y=0}^{n-1} (x-\bar{x})^i (y-\bar{y})^j \, I(x,y) \qquad (13)$$

Where, U_{ij} refers to the central moments for i=0,…,3 and j=0,…,3

$$\eta_{ij} = \frac{U_{ij}}{U_{00}^{\left(\frac{i+j}{2}\right)+1}} \qquad (14)$$

η_{ij} refers to the central moments U_{ij} after being normalized.

$$I1 = \eta_{20} + \eta_{02} \qquad (15)$$

$$I2 = (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2 \qquad (16)$$

$$I3 = (\eta_{30} - 3\eta_{12})^2 + (3\eta_{21} - \eta_{03})^2 \qquad (17)$$

$$I4 = (\eta_{30} + \eta_{12})^2 + (\eta_{21} + \eta_{03})^2 \qquad (18)$$

$$I5 = (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12})\left[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2\right] + (3\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03})\left[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2\right] \qquad (19)$$

$$I6 = (\eta_{20} - \eta_{02})\left[(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2\right] + 4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{21} + \eta_{03}) \qquad (20)$$

$$I7 = (3\eta_{21} - \eta_{03})(\eta_{30} + \eta_{12})[\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] - (\eta_{30} - 3\eta_{12})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] \qquad (21)$$

Where, Ii for i=1,…,7 represent the seven Hu invariant moments.

$$S = \sqrt{\sum_{i=1}^{7} (P(I_i) - P'(I_i))^2} \qquad (22)$$

Where, S stands for the similarity rate between the two compared patterns, *P* and *P'* (referring to extracted pattern and training one, respectively).

4) Experiments and results

The aim of this example is to explain how exactly the proposed pattern filter works. This section reports an overview of the obtained results after implementing the proposed filter to a sample of training patterns (FP1,FP2,FP3, AP1, AP2, AP3 where FP and AP stand for finder pattern and alignment pattern, respectively) and a set of four extracted patterns (EP1,EP2,EP3,EP4) chosen randomly. The pattern classification is mainly based on a predefined similarity threshold which is set at 2.5 (experimentally, the value 2.5 allows obtaining the best results). The false positive removing is conducted as follows; **Rule 1**, each extracted pattern is classified either as a true positive (if at least one of its similarity rates is less or equal to 2.5) or as a false positive (if all its similarity rates are greater to the said threshold). **Rule 2**, once a pattern is declared as a true positive an additional verification is carried out in order to determine if the treated pattern belongs to FP category or AP one, i.e. it is considered as FP if all its similarity rates with training FPs are less compared to its similarity rates with training Aps or otherwise.

By returning to the resulting patterns moments related to the queried patterns, the obtained results are illustrated in Table 1, in which each column refers to a given pattern moments. As reported in section D.1, the patterns classification is conducted by comparing each extracted pattern moments with those of each training one separately, namely, the comparisons in question are based on Euclidian similarity measurement. After achieving all patterns comparisons, the obtained similarity rates can be summarized as shown in Table 2. Basing on these results, one can observe that the extracted pattern 1 (EP1) is considered as a true positive since the Rule 1 is respected, furthermore it is classified as a finder pattern in consideration that all its similarity rates with training FPs are less compared to its rates with APs (Rule 2). Concerning EP4, it is retained since the Rule 1 is verified, moreover, its corresponding similarity rates show that it is more similar to AP category than FP one and consequently categorized as AP. On the other hand, EP2 and EP3 are classified as false positives seeing that they do not respect the Rule 1 (i.e. all their similarity rates are greater than 2.5), thus, the two evoked extracted patterns are removed.

By applying the discussed filter to all the raw extracted patterns (Fig 10), all the irrelevant extracted ones will be removed. The newly obtained result is shown in Fig. 12.

TABLE I. CORRESPONDING SEVEN INVARIANT MOMENTS RELATED TO A SAMPLE OF TRAINING PATTERNS AND A SET OF FOUR EXTRACTED PATTERNS

	Training patterns sample						**Extracted patterns sample**			
Invariant moments	**FP1**	**FP2**	**FP3**	**AP1**	**AP2**	**AP3**	**EP1**	**EP2**	**EP3**	**EP4**
I1	1.2514	1.4165	0.9841	0.2054	0.2500	0.2430	1.3201	4	2.7921	0.2097
I2	0.0784	0.0603	0.4820	0.0013	0.0105	0.0090	0.0721	1.29	4.5402	0.0020
I3	0.2042	0.1896	0.6210	0.0081	0.0075	0.0079	0.2415	4.52	9.4000	0.0075
I4	0.1240	0.1778	0.0005	0.0630	0.0210	0.0001	0.5417	5.4	8.3331	0.0952
I5	0.0109	0.0067	0.9410	0.0004	0.0081	-0.0028	0.0119	3.52	16.4203	0.0093
I6	-0.0301	-0.0227	-0.5470	0.4215	0.0065	-0.0060	-0.9830	-2.5	-10.2108	-0.5201
I7	-0.0594	-0.0319	-0.6410	0.007	-0.019	-0.0001	-1.0125	-26.6	5.1024	-0.0151

TABLE II. COMPARISONS BETWEEN A SAMPLE OF TRAINING PATTERNS AND FOUR EXTRACTED ONES IN TERMS OF EUCLIDIAN SIMILARITY

	FP1	**FP2**	**FP3**	**AP1**	**AP2**	**AP3**
EP1	***1.4132***	***1.4242***	***1.3817***	2.1315	1.8550	1.8686
EP2	27.8981	27.9026	27.1743	28.1737	28.1098	28.1329
EP3	23.9540	23.9346	23.0700	24.3417	24.1717	24.1779
EP4	1.1716	1.3219	1.5735	***0.9425***	***0.5334***	***0.5243***

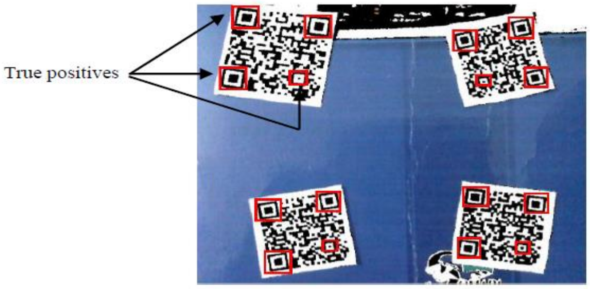

Fig. 12. Remaining patterns after removing false positives.

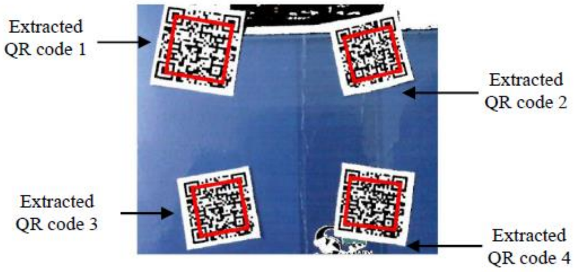

Fig. 13. Extracted QR codes after gathering the corresponding patterns into clusters.

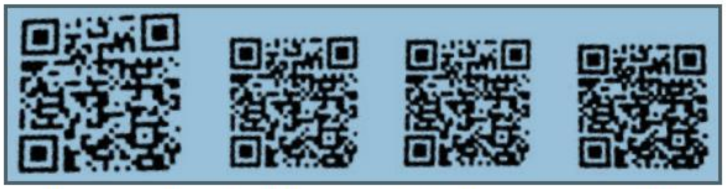

Fig. 14. Extracted QR codes after being cropped and undergone geometric rectifications.

E. *QR code localization and rectification*

So far, the used pattern filter has retained only 16 patterns from the total of 19 ones which have been roughly extracted by the traditional scanning process. That means, 3 extracted patterns have been considered as false positives. Basing on the remaining pattern, the QR code localizer algorithm localizes the position of each QR code. Namely, localizing a QR code requires finding a combination of three FPs and at least one AP. The three FPs have to be close to each other according to a strict distance constraint (23), i.e. the distance separating each couple of FPs must verify the mentioned constraint. The APs are easily found, seeing that they are basically laid inside the three FPs area. Once the constraint is respected, the three treated patterns are declared belonging to the same cluster and thus the corresponding QR code is found. By assembling the 16 patterns in their corresponding clusters Fig. 13, four QR codes can finally be localized. In order to further improve the proposed algorithm accuracy and facilitate the decoding stage, the four extracted QR code are cropped separately and undergo a geometric rectification in case of acute deformations such as rotation and perspective deformation. The obtained result after QR codes rectification is shown in Fig. 14.

$$D_{\min} \leq D(FPi, FPj) \leq D_{max} \tag{23}$$

Where, D_{max} and D_{min} stand respectively for the maximum and minimum acceptable distance separating two finder patterns whereas D(FPi,FPj) refers to the real distance between finder pattern i and j.

IV. Conclusion

In this paper an enhanced QR code recognition algorithm has been proposed. This system mainly aims at outperforming the earlier proposed QR code recognition in terms of response time and recognition precision. The existing QR code recognition systems are mostly based on huge feature descriptors (e.g. Haar-like features and Hough transform) in addition to complex pattern classifiers which require voluminous training data. The proposed algorithm uses a light training data thanks to the use of Hu invariant moments. This technique allows converting each image to only seven coefficients (geometrically invariant) and consequently alleviating image comparisons complexity.

The evoked system starts first by selecting the image with the best quality, this by eliminating all blurred images. Afterwards, the region of interests are selected using an efficient achromatic filter. The obtained regions are then scanned horizontally and vertically in order to find the emplacement of the QR code patterns (i.e. Finder and Alignment patterns) which are distinguished by a well-defined ratios. It should be noted that the evoked scan result is still unreliable due to the occurrence of false positives. In this regard, the Hu invariant moments descriptor combined with Euclidian similarity measurement are used to remove all false detected patterns. Once done, QR codes locations are found by adequately grouping together each corresponding three FPs with their corresponding APs. The integration of Hu moments and Euclidian similarity as pattern classifier has dramatically decreased the processing time while enhancing QR code recognition accuracy compared to the traditional methods which use the whole patterns images to perfume comparisons. Although its efficiency, Hu moments technique still less efficient compared to Zernike moments which are more efficient, flexible, in addition that their coefficients are easier to reconstruct than Hu ones [24]. Zernike moments and other settings will be integrated in our future work as to further enhance the performance of proposed system.

References

[1] M. Hara, M. Watabe, T. Nojiri, T. Nagaya, Y. Uchiyama, “Optically readable two-dimensional code and method and apparatus using the same”, Patent US 5726435 A, 1994.

[2] Hicham Tribak, Youssef Zaz, “Remote QR Code Recognition Application: Solar Panels Identification in Solar Plant”, International journal of imaging and robotics”, Volume 17, issue 2, pp. 17-31.

[3] V. Uzun, “QR-Code Based Hospital Systems for Healthcare in Turkey," 2016 IEEE 40th Annual Computer Software and Applications Conference (COMPSAC), Atlanta, GA, 2016, pp. 71-76.

[4] P. Bodnar, L. G. Nyul, “Improved QR Code Localization Using Boosted Cascade of Weak Classifiers”, Acta Cybernetica, 2015, pp. 21-33

[5] H. Tribak, S. Moughyt, Y. Zaz and G. Schaefer, “Remote QR code recognition based on HOG and SVM classifiers,” 2016 International Conference on Informatics and Computing (ICIC), Mataram, 2016, pp. 137-141.

[6] H. Tribak,Y. Zaz, “QR Code Recognition based on Principal Components Analysis Method” International Journal of Advanced Computer Science and Applications (IJACSA), 8(4), 2017.

[7] A. Sun, Y. Sun and C. Liu, “The QR-code reorganization in illegible snapshots taken by mobile phones,” 2007 International Conference on Computational Science and its Applications (ICCSA 2007), Kuala Lampur, 2007, pp. 532-538.

[8] S. Ming, F. L. Sheng,Y. X. Ting, Z. S. Huai, “ Image Analysis Method for QR Code's Automatic Recognition”, Journal of University of Electronic Science and Technology of China, 2009.

[9] Jeng-An Lin and Chiou-Shann Fuh, “2D Barcode Image Decoding,” Mathematical Problems in Engineering, vol. 2013, Article ID 848276, 10 pages, 2013.

[10] Y. Kato, D. Deguchi, T. Takahashi, I. Ide and H. Murase, “Low Resolution QR-Code Recognition by Applying Super-Resolution Using the Property of QR-Codes,” 2011 International Conference on Document Analysis and Recognition, Beijing, 2011, pp. 992-996.

[11] L. Huijuan, “Omnidirectional Recognition of Quick Response Code Image”, Chinese Journal of Scientific Instrument, 2006.

[12] L. N. Zhong, Y. J. Yu, “Recognition of Two-dimensional Bar Code Based on Fourier Transform”, Journal of Image and Graphics, 2003-08

[13] P. Marziliano, F. Dufaux, S. Winkler and T. Ebrahimi, “A no-reference perceptual blur metric,” Proceedings. International Conference on Image Processing, 2002, pp. III-57-III-60 vol.3.

[14] F. Crete-Roffet, T. Dolmiere, P. Ladret, M. Nicolas. “The Blur Effect: Perception and Estimation with a New No-Reference Perceptual Blur Metric.” SPIE Electronic Imaging Symposium Conf Human Vision and Electronic Imaging, Jan 2007, San Jose, United States. XII, pp.EI 6492-16, 2007.

[15] R. M. Bora and N. M. Shahane, “Image forgery detection through motion blur estimates,” 2012 IEEE International Conference on Computational Intelligence and Computing Research, Coimbatore, 2012, pp. 1-4.

[16] X. Marichal, Wei-Ying Ma and HongJiang Zhang, “Blur determination in the compressed domain using DCT information,” Proceedings 1999 International Conference on Image Processing (Cat. 99CH36348), Kobe, 1999, pp. 386-390 vol.2.

[17] Kanjar De, V. Masilamani, “Image Sharpness Measure for Blurred Images in Frequency Domain”, Elsevier Procedia Engineering, Volume 64, 2013, Pages 149-158.

[18] Kong Suran, “QR Code Image Correction based on Corner Detection and Convex Hull Algorithm”, JOURNAL OF MULTIMEDIA, VOL. 8, NO. 6, DECEMBER 2013.

[19] G. Klimek and Z. Vámossy, “QR Code detection using parallel lines,” 2013 IEEE 14th International Symposium on Computational Intelligence and Informatics (CINTI), Budapest, 2013, pp. 477-481.

[20] S. Moughyt, Y. Zaz, S. Fathi and O. El Kadmiri, “Sky status: A local analysis of ground based digital images,” 2016 5th International Conference on Multimedia Computing and Systems (ICMCS), Marrakech, 2016, pp. 667-670.

[21] F. Zaklouta ∗, B. Stanciulescu, “Real-time traffic sign recognition in three stages”, Journal of Robotics and Autonomous Systems, 62 (2014) 16–24.

[22] S.M. Bascon, S. Lafuente-Arroyo, P. Gil-Jimenez, H. Gomez-Moreno, “F. Lopez- Ferreras, Road-sign detection and recognition based on support vector machines”, IEEE Transactions on Intelligent Transportation Systems 8 (2) (2007) 264–278.

[23] M. K. Hu, “Visual Pattern Recognition by Moment Invariants”, IRE Trans. Info. Theory, vol. IT-8, pp.179–187, 1962.

[24] R. K. Sabhara1, C. P. Lee, and K. M. Lim, “Comparative Study of Hu Moments and Zernike Moments in Object Recognition”, Smart Computing Review, vol. 3, no. 3, June 2013.