QR Code Recognition based on Principal Components Analysis Method

Hicham Tribak, Youssef Zaz

Abdelmalek Essaadi University, Faculty of Science
Tetouan-Morocco

Abstract—**QR (Quick Response) code recognition systems (based on computer vision) have always been challenging to be accurately devised due to two main constraints: (1) QR code recognition system must be able to localize QR codes from an acquired image even in case of unfavorable conditions (illumination variations, perspective distortions) and (2) The system must be adapted to embedded system platforms in terms of processing complexity and resources requirement. Most of the earlier proposed QR code recognition systems implemented complex feature descriptors such as (Harris features, Hough transform which aim at extracting QR code pattern features and subsequently estimating their positions. This process is reinforced by pattern classifiers e.g. (Random forests, SVM) which are used to remove false detected patterns. Those approaches are very computationally expensive. Thus, they are not able to be run in real-time systems.**

In this paper, a streamlined QR code recognition approach is proposed to be efficiently operable in systems characterized by a limited performance. The evoked approach is conducted as follows: the captured image is segmented in order to reduce searching space and extract the regions of interest. Afterwards a horizontal and vertical scans are performed to localize preliminarily QR code patterns, followed by Principal Component Analysis (PCA) method which allows removing false positives. Thereafter, the remaining patterns are assembled according to a constraint so as to localize the corresponding QR codes. Experimental results show that the incorporation of PCA decreases notably the processing time and increase QR code recognition accuracy (96%).

Keywords—*QR code; Image segmentation; Principal Components Analysis; Perspective rectification; Pattern similarity measurement*

I. INTRODUCTION

Nowadays, Quick response code (QCR) has become one of the most relying data storage tools by coinciding the enormous growth of the E-Commerce and mobile phone market. QRC combines notably between robustness and cost effectiveness. Thence, it knows increasingly a widespread demand in diverse fields (e.g. industry, commerce, etc.) and for various purposes, such as objects tracking and products labeling [1,2] for which it is attached, this by giving their corresponding information. The latter is stored using a reliable error correcting codes (Reed-Solomon) as well as a vision-based decoder for data extraction. The QRC decoder must be rigorously designed while taking into account number of unavoidable factors. By nature, QCRs are mostly placed on complex environments which are characterized by irregular textures, and illumination changes. This renders QRC

recognition quite challenging. In this regard, many competitions have been conducted by inviting computer vision researchers to invent accurate QRC recognition algorithms which can prove their ability in overcoming the evoked obstacles.

QRC recognition algorithms are mostly devised to be implemented in mobile phones or in embedded system platforms, the two latter are distinguished by limited resources as well as a real-time execution. By such factors, the architecture of the proposed algorithms must not be computationally expensive. In contrast, when we analyze the earlier proposed approaches which exist in the literature, we notice that most of them rely on voluminous feature descriptors which generate pattern descriptions of high dimension. These features are used to feed complex pattern classifiers (implemented as false positives removal). Although the accuracy of the earlier proposed QRC recognition algorithms, most of them turned out inappropriate for embedded systems due to their high resource requirements and time consuming.

In this paper a variant QCR recognition system is devised exclusively with accordance to embedded system performance limitation. The proposed approach is based on Principal Components Analysis which is implemented as pattern classifier which basically helps reducing the original data dimension before proceeding to the effective processing. In addition, PCA function receives a set of training pattern images as well as those related to the extracted candidate patterns as input data and subsequently generates loading coefficients matrix in which each row refers to a given pattern. Afterwards, each extracted pattern coefficients are compared separately with those of the training patterns by means of Euclidian similarity measurement. An extracted pattern is considered as a true positive if it has at least one similarity which is less or equal to a predefined threshold.

The remainder of this paper is organized as follows. In section II the existing methods of QR code recognition systems are discussed. The background of this paper is presented in section III. The proposed approach is explained in section IV. The obtained results are displayed in section V. The conclusion accompanied with an outlook of the future improvements are reported in section VI.

II. RELATED WORKS

Many QR code recognition approaches have been earlier proposed in the literature, basing upon different distinctive

techniques. In this regard, G. S. Vardhan et al [3] used QRC as RFID tag which is therefore recognized through frequency domain reflectometry. N. Bhardwaj et al [4] and Z. Yang et al [5] proposed RGB color segmentation along with luminance enhancement to reduce searching space of QRC. Y. L. Lin et al [6] implemented the histogram of oriented gradients as QRC feature descriptor combined with AdaBoost classifier.
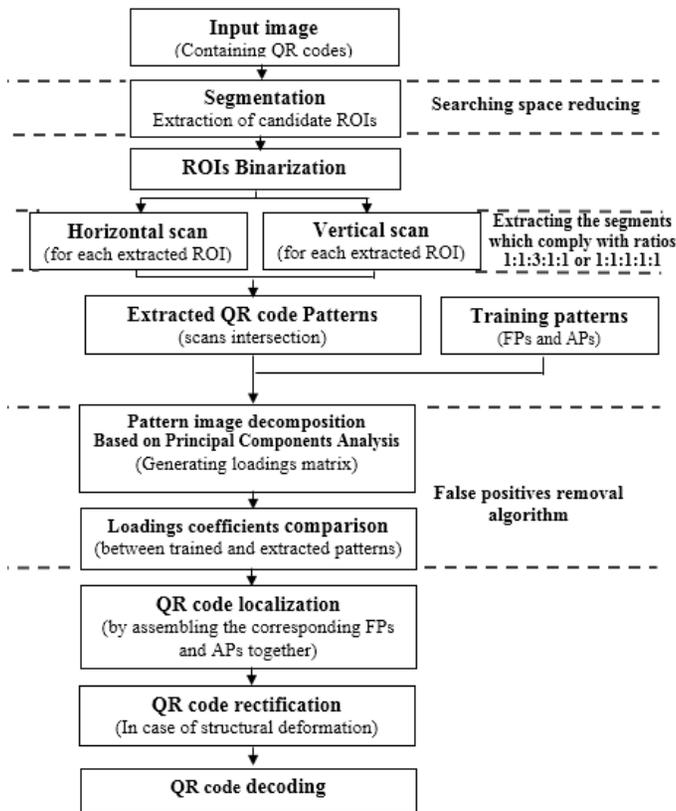

Fig. 1. Overall diagram of the proposed QR code recognition system

M. Ostkamp et al [7] used the raytracing technique which allows recognizing and retrieving QRC even with curved distortions. M. Ahn et al [8] recognition approach is based on cloud-based pre-generated image matching. Y. Kato et al [9] used basic QRC features and a sliding window detector to recognize QRC from image of low resolution, this by constructing an image of high resolution. Z. l. Liao et al [10] relied on vertex point characteristics which aim at localizing the corners related to the finder patters of the QRC. Y. Liu et al [11] focused mainly on the binarization stage using multi-level thresholding as well as a global scanning of the binary image. Once achieved, QRC positions can be estimated. I. Szentandrasi et al [12] implemented Hough transform since they considered QRC as a set of perpendicular segments. Hough transform and one dimensional wave scanning are also implemented by Liu Huijuan [13] for both QRC recognition and structure retrieval in case of geometric distortions. L. Zhong et al [14] exploited spectral space analysis by means of Fourier Transform through which both QCR recognition and deblurring are achieved. P. Bodnar et al [15] implemented a combination of cascading classifiers trained separately by Haar-like features, Local Binary Patterns and Histogram of

Oriented Gradient features. K. Suran [16] based on Harris corner detector and convex hull algorithm.

III. BACKGROUND

QR code is an efficient type of 2D barcode (standardized by the norm ISO/IEC 18004). It allows storing and reading an important amount of data of different types: up to 7089 numeric values, 4296 alphanumeric characters, 2953 binary values and 1817 kanji. QRC is also available in 40 different versions (varying between 1 and 40) as well as various module sizes (21x21 up to 177x177). Each module can handle up to four data error correction levels (L, M, Q and H). These levels can respectively retrieve up to 7%, 15%, 25% and 30% of QRC structure in case of structural damage [17]. Moreover to its technical advantages, QRC is economically viable in consideration that it can be printed in a small piece of paper or exploited directly in electronic format, unlike RFID tags which require specific and costly equipment.

Mainly, QR code localization is conducted through a well-defined patterns characterized by distinctive texture. These patterns help the scanner to preliminarily locate a QR code even if it is placed on complex texture. The evoked patterns can be categorized into two categories: **Finder pattern** (FP) and **Alignment pattern** (AP). The former allows a preliminary estimation of the QRC position, whereas the latter is used to determine the orientation and geometric deformations of the QRC. A structural survey of QRC is displayed in Fig 2.
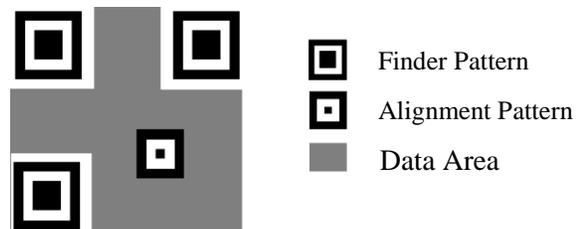

Fig. 2. Structural survey of QR code

IV. PROPOSED APPROACH

A. *Searching-space reducing*

Most of QCR recognition systems are devised to be implemented either in handheld devices (e.g. mobile phone, tablet) or in embedded system platforms (Arduino, Raspberry Pi). These equipment are characterized by resources of limited capacity and basically run in real-time. To overcome these restrictions, a searching-space reduction is inevitably required before proceeding to the QRC recognition processing. In this context, various approaches have been proposed. P. Moallem et al [18] implemented edge features and disparity gradient limit technique so as to reduce searching-space during the correspondence of stereo vision areas. E. Cho et al [19] used Quad-tree structure to extract the foreground layer of an image which is assumed containing the regions of interest. The evoked process is a twofold aim, on the one hand it allows reducing computation time along with alleviating resource requirements, and on the other hand it helps increasing recognition accuracy by discarding areas which do not satisfy a predefined constraint. In the current paper, a powerful color-

based filter is implemented, which is already proposed by F. zaklouta et al [20] through which a binary mask Fig 3(b) is obtained after scanning all RGB pixels related to the input image and extracting all the areas which are distinguished by an achromatic color (in consideration that QRC texture is characterized by black and white colors). By doing so, a set of candidate regions of interest are extracted. The obtained results are depicted in Fig 3.

$$T_{(X,Y)} = \frac{|R_{(X,Y)} - G_{(X,Y)}| + |G_{(X,Y)} - B_{(X,Y)}| + |B_{(X,Y)} - R_{(X,Y)}|}{K} \quad (1)$$

$$I_{(X,Y)} = \begin{cases} 0 & if \ T(x,y) \leq 1 \\ 1 & otherwise \end{cases}$$

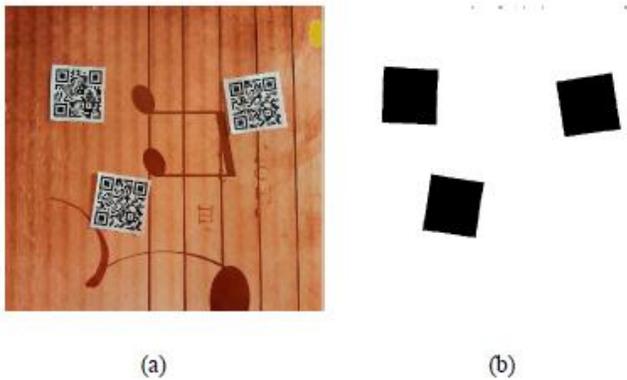Where K is the achromatic color extraction rate. K is set at 30.

Fig. 3. Regions of interest extraction. (a) Original image (b) Binary mask wherein the black areas refer to the regions of interest

So far, a set of potential QRC areas have been preliminary located by means of the aforementioned binary mask. In order to check out the validity of the obtained ROIs, the latter are passed down to an additional assessment process, in which each ROI is scanned separately into two directions (vertical and horizontal). A valid ROI must contain exactly three FPs and at least one AP. Before proceeding to the evoked process, the set of extracted ROIs are binarized so as to facilitate texture analysis. The algorithm starts first by converting each of them to the grayscale representation using a powerful method which is defined by (2). This equation emphasizes the mean value between the dominant and the minimum intensity related to the three RGB color channels. The mentioned transform outperforms other grayscale ones given that it provides the best contrast. As regards the binary conversion, it is defined by (3). The reached results are displayed in Fig 4.

$$I_{Gray} = \frac{[Max \ P(R,G,B) + Min \ P(R,G,B)]}{2} \quad (2)$$

$$I_{Binary}(x,y) = \begin{cases} 0 & if \ I_{Gray}(x,y) \leq T \\ 1 & Otherwise \end{cases} \quad (3)$$

Where P(R,G,B) stands for RGB pixel. I_{Gray} represents a grayscale pixel. I_{Binary} refers to a binary pixel. T is a grayscale threshold.
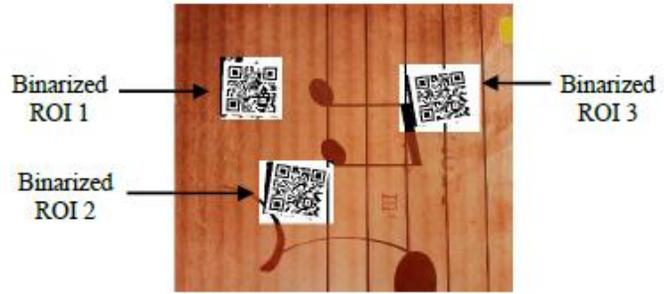

Fig. 4. Candidate regions of interest after Binarization

B. QR code pattern localizations

In order to find the exact location of FPs and APs within the extracted ROIs, three successive processes are performed for each ROI separately. A horizontal scan is launched first, in which the binary area related to each ROI is scanned horizontally. The scan process retains and saves the coordinates (coordinates of the beginning pixel and the end one) of each valid horizontal segment (on a specific matrix denoted **H**) whose the structure complies whether with the ratio 1:1:3:1:1 (related to FP) or 1:1:1:1:1 (corresponding to AP). The structure of the horizontal and vertical segments related to both FP and AP are displayed in Fig 5(a) and (b) respectively. The red perpendicular segments stand for a horizontal and vertical section of a FP and AP. Once the horizontal scan is achieved, a vertical one is launched in the same way as the previous. The vertical scan saves the coordinates of all valid vertical segments (on a specific matrix denoted **V**) which respect the aforementioned ratios.
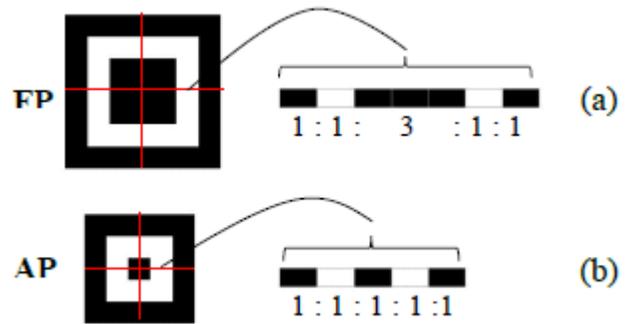

Fig. 5. Longitudinal and transversal sections of a FP and AP. (a) Sections of a FP. (b) Sections of an AP

At the end, the scanning algorithm computes the intersection between the two scan results (**H ∩ V**) i.e. the intersection between the retained vertical segments and the horizontal ones, this process indeed allows extracting the location of the central pixel related to each extracted pattern. Besides true positives (FPs and APs) a set of false ones are extracted. The latter disturb significantly the QR code localizations. In this reason all the extracted patterns are transmitted to a filtering process based on Principal Components Analysis which is used as pattern feature

descriptor combined with Euclidian similarity measurement which aim at comparing the resulting pattern features. The implemented filter is sufficiently trained by number of FPs and APs images which are taken under different illumination conditions. The preliminarily extracted patterns are displayed in Fig 6.
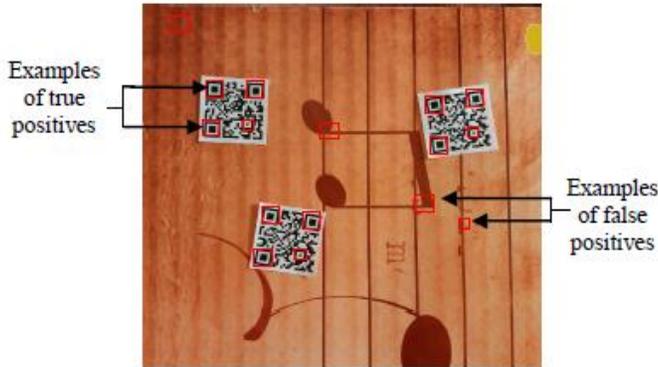

Fig. 6.   Preliminarily extracted patterns before removing false positives

C. False positives removal algorithm

Most of the false positive removal algorithms are mainly based on three distinct steps. (1) Pattern feature extraction (2) Classifier training. (3) Prediction and Classification. Although the efficiency of this architecture, it is very computationally expensive in terms of computation time and resource requirements, in consideration that the majority of the feature descriptors which are implemented in QR code recognition systems such as SIFT [21], SURF [22], MSER [23] generate huge pattern descriptions, the latter are characterized by high dimension. In addition to that, most of the evoked architectures implement complex classifiers among others, Random forests, K-d trees. These classifiers generally require the building up of an enormous combination of trees which conduct to form a huge decision paths. In order to enhance the performance of the tree classifiers various constraints must be dealt with, especially when it comes to: **(1)** Finding a valid stopping criterion. **(2)** Finding an optimal trees splitting. **(3)** Determining the best trees selection which allows gaining an accurate classification. **(4)** Browsing all the forest paths i.e. starting from the root (main node) until the deepest leaves (terminal nodes). For each browsed path, the probability reflecting the number of feature occurrences must be computed. On the basis of the evoked constraints, the discussed approach turned out inconsistent for embedded system platforms. To get around this problem, a variant approach is proposed which aims at alleviating the QRC recognition complexity. This by implementing the well-known statistical procedure "Principal Components Analysis (PCA)" which is earlier invented by [24] accompanied with the Euclidian similarity measurement. As aforementioned, PCA allows decreasing fairly the original data dimension and generating alternatively a reduced output data in the form of two streamlined matrices i.e. Loading matrix and Score matrix. These matrices are henceforth used to manipulate and analyze the original data. The use of PCA has notably alleviated both complexity and processing time while guarantying a high accuracy. Finally, it is worthwhile to note

that the main advantage of the PCA is manifested in that it does not require to be trained with false positives, which decreases notably the size of the training data, unlike SVM and tree classifiers which must be fed with a huge sample of false positives.

It should be noted that the idea of this paper has been inspired from the face recognition system which has known a considerable success in human face identification. In the proposed system, the same principle has been maintained. Namely the evaluation of the extracted patterns validity is divided into three steps. **(1)** Preparing a sample of pattern images as training data e.g. Fig 7(a) and (b) which refer to FP and AP categories respectively. The trained pattern images have been taken under different illumination conditions and have undergone various geometric deformations (e.g. affine and projective deformations). Afterwards, the training patterns accompanied with the extracted ones (e.g. Fig 7 (c) and (d) which refer to a true positive and a false one respectively) will be transmitted to PCA function. The latter will in turn generate the corresponding loading matrix whose each row represents a given pattern. The discussed procedure is explained in detail in the next section.
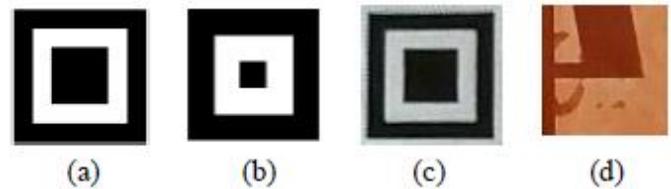

Fig. 7.   Sample of training and extracted patterns. (a) Training image of a FP (b) Training image of AP (c) Extracted true positive (FP) (d) False positive

- **Proposed approach steps**

The proposed approach is conducted as follows:

1) At first, the size of each extracted pattern image is rescaled so that it becomes compatible with those of the training patterns. Moreover, each pattern image is converted to a column vector (i.e. by successively concatenating its). The evoked column vectors will be denoted CV_i (for i=1,2,…X) where X stands for the total number of the queried patterns and i represents the column index. If the pattern image size is NxN, then the corresponding column vector size is N^2x1. The process is applied to all the queried patterns (training and extracted patterns). By doing so, a global matrix (denoted A) is obtained whose the size is N^2xX. The matrix A is defined as follows:

$$A = \begin{pmatrix} p_{11} & p_{12} & p_{13} & \cdots & p_{1X} \\ p_{21} & p_{22} & p_{23} & \cdots & p_{2X} \\ p_{31} & p_{32} & p_{33} & \cdots & p_{3X} \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ p_{N^2 1} & p_{N^2 2} & p_{N^2 3} & \cdots & p_{N^2 X} \end{pmatrix} \quad (4)$$

$$CV_1 \quad CV_2 \quad CV_3 \quad\quad CV_X$$

2) Calculating the mean value (denoted mi) of each

column of A. i.e. computing the sum of all coefficients related to each column CVi and dividing the resulting sum by the total number of column coefficients N^2. The evoked mean is defined by the expression (5).

$$m_i = \frac{1}{N^2} \sum Coef\ (CV_i) \quad for\ i = 1,2,\dots,X$$
$$(5)$$

Where Coef (CVi) stands for the set of coefficients related to the ith column of *A*. X stands for the total number of columns contained in *A*, and m_i is the mean value of the ith column.

3) Subtracting each column of A by the corresponding mean m_i as defined by the expression (6). This operation allows centering each column of A around its mean. The resulting centered matrix is denoted *A'* and its corresponding structure is defined by (7).

$$CVN_i = CV_i - m_i\ for\ i = 1,2,\dots,X$$
$$(6)$$

Where CVN_i represents the new ith centered column vector of *A'*.

$$A' = \begin{pmatrix} p'_{11} & p'_{12} & p'_{13} & \dots & p'_{1X} \\ p'_{21} & p'_{22} & p'_{23} & \dots & p'_{2X} \\ p'_{31} & p'_{32} & p'_{33} & \dots & p'_{3X} \\ \vdots & \vdots & \vdots & \dots & \vdots \\ p'_{N^21} & p'_{N^22} & p'_{N^23} & \dots & p'_{N^2X} \end{pmatrix} \qquad (7)$$

$$CVN_1 \quad CVN_2 \quad CVN_3 \qquad CVN_X$$

In this section, the decomposition of the original matrix A is explained from the mathematical point of view basing on PCA method. As aforementioned, the main purpose of the use of PCA is manifested on false positives removal while alleviating the complexity of the comparisons which are performed between the detected patterns and the training ones. In this regard, the matrix A is transmitted as an input to the PCA function which thereafter generates an output in form of two distinctive matrices. The latter are generally known as loading matrix (denoted C_M) and score matrix (denoted S_M). The loading matrix C_M is considered as the main matrix on which the proposed approach relies on to ensure pattern comparisons. The coefficients of C_M reflect the principal variances along the eigenvectors related to the original data as well as the correlation between the queried patterns. Furthermore, the sum of squares of each column of C_M is 1. The loading matrix C_M is obtained after extracting the eigenvectors of the covariance matrix related to the input matrix A. This process is defined by the expression (9). As to the score matrix S_M, it mainly represents the two main orthogonal directions (i.e. the two main principal components) which split optimally the input data through two perpendicular straight lines. The score matrix is obtained by multiplying the centered matrix **A'** by the loading matrix C_M as defined by the equation (10). On the other hand, the decomposition of the original matrix A is conducted according to the equation (8).

As previously mentioned, the proposed false positives removal algorithm will exclusively rely on loading matrix coefficients. In consideration that these coefficients provide significant information about the treated patterns. Henceforth, each pattern will be represented by a given row of C_M. Basing on this rule, each extracted pattern coefficients will be compared separately with the coefficients of the training patterns. This comparison is conducted by means of the Euclidian similarity measurement. An extracted pattern is considered as a true positive if it is similar to at least one training pattern.

$$A = S_M C_M^T + E_M \qquad (8)$$

$$C_M = Eigenvectors\ (Cov(A)) \qquad (9)$$

$$S_M = A'.C_M \qquad (10)$$

Where S_M stands for the score matrix. C_M^T refers to the transposed of the loading matrix. E_M is the error matrix which is added to the multiplication result of S_M and C_M^T to reconstruct the original matrix A. Cov(A) stands for the covariance matrix of A. and finally, A' refers to the centered matrix.

D. Extracted pattern assessment and classification

To better understand the proposed approach, a simple demonstration seems useful. In this example the queried data is assumed to contain only 4 training patterns and 2 extracted ones. The two extracted patterns are divided into a true positive and a false one. As previously mentioned, the six patterns are transmitted to the PCA decomposition function after being converted to column vectors. This decomposition in turn yields the corresponding loading matrix C_M as defined in (11). The first four rows of C_M refer to the four training patterns whereas the fifth row (true positive) and the sixth one (false positive) represent the extracted patterns. The classification of the latter requires a comparison of their corresponding loading coefficients with those related to the training ones basing on the Euclidian similarity rate which is denoted E_D. The latter is in turn defined by the expression (12) whilst the pattern classification is obtained through the constraint (13). An extracted pattern is considered as a true positive if and only if it has at least one similarity rate which is less or equal to a predefined threshold (denoted T_D). Otherwise, the pattern is considered as a false positive and consequently removed from the set of extracted candidate patterns. The Table 1 illustrates the obtained results after comparing each extracted pattern coefficients with the training ones. The similarity threshold T_D is set at 0.1. By analyzing the obtained similarity rates, one can observe that the extracted pattern 1 (defined by the fifth row) obtains a similarity rate equals to 0 (according to its comparison with row 1 which refers to a finder pattern), since the evoked similarity is less than 0.1, the extracted pattern 1 is considered as a true positive. On the other hand, the extracted pattern 2 (represented by the sixth row) is considered as a false positive in consideration that none of its obtained similarity rates comply with the classification constraint. This candidate pattern is subsequently removed.

$$C_M = \begin{pmatrix} 0.5803 & 0.3776 & 0.1407 & -0.0238 & 0.0174 \\ -0.2941 & 0.3995 & 0.2070 & 0.7147 & 0.4475 \\ -0.4627 & 0.5750 & 0.3665 & -0.3293 & -0.4610 \\ -0.1566 & 0.2561 & -0.2292 & -0.5887 & 0.7147 \\ 0.5803 & 0.3776 & 0.1407 & -0.0238 & 0.0174 \\ 0.0366 & -0.3989 & 0.8548 & -0.1819 & 0.2752 \end{pmatrix} \quad (11)$$

Where C_M stands for the loading matrix coefficients whose each row refers exclusively to a given pattern of the six queried ones.

$$E_D(i,j) = \sqrt{(a_1 - b_1)^2 + (a_2 - b_2)^2 + \cdots + (a_i - b_j)^2} \quad (12)$$

$$\text{Classification decision} = \begin{cases} True\ positive & if\ E_D(i,j) \leq T_D \\ False\ positive & Otherwise \end{cases}$$
(13)

Where $E_D(i,j)$ is the Euclidian similarity rate between the extracted pattern i and the training one j. a_i and b_j stand for the loading coefficients. T_D represents the classification threshold.

TABLE. I. PATTERN CLASSIFICATIONS BASED ON THE COMPARISON OF LOADING COEFFICIENTS

Extracted candidate patterns	Loading rows comparisons	Similarity rate
Extracted pattern 1 (row 5 of C_M)	Row 5 with 1	0
	Row 5 with 2	1.22
	Row 5 with 3	1.22
	Row 5 with 4	1.22
Extracted pattern 2 (row 6 of C_M)	Row 6 with 1	1.22
	Row 6 with 2	1.41
	Row 6 with 3	1.41
	Row 6 with 4	1.41

Given how crucial the implementation of PCA method turns out, the obtained results after applying the false positives filter are displayed in Fig 8. It clearly appears that all the false detected patterns (which were present in Fig 6) have been successively removed. Henceforth the QR code localizations can be performed by grouping adequately the remaining patterns.
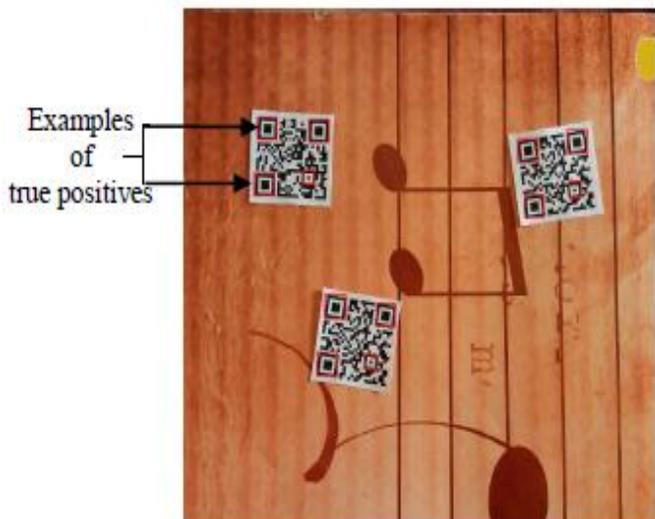

Fig. 8. Obtained results after removing false positives

E. QR code localizations

In this stage, QR code localizations are carried out through the measurement of distances which separate the remaining patterns. To make that possible, these patterns are divided into groups, in which each group must exactly contain three FPs and at least one AP. Basing on this constraint, the algorithm considers that three FPs belong to the same QR code, if the distances separating them satisfy the constraint (14). Once satisfied, the corresponding APs are automatically inferred basing on the principle that APs are usually located within the FPs area. The obtained results are shown in Fig 9 wherein three QR codes are correctly localized.

Fig. 9. QR code localizations based on pattern distance measurements

$$D_{ij} = D_T \pm \omega \quad (14)$$

Where D_{ij} stands for the distance which separates two FPs i and j. D_T is a distance threshold and ω designates a marginal acceptable difference.

F. Geometric and color rectifications of QR code

QR code shape evaluation and correction are often deployed in systems wherein the 2D barcode decoder is fixed on a movable object. e.g. the case when mobile phone is used as a QR code decoder, a simple hand motion induces shape deformations of the targeted QR codes, which consequently become unreadable. The most common deformations which may alter the captured QR code can be divided into three categories: (a) Affine deformation, (b) Projective deformation and (c) Lens deformation. As to the affine deformation, the QR code shape is simply rotated or transformed to a well-defined geometric shape e.g. parallelogram, rhombus, etc. Given its linearity, the affine transformation is distinguished by its ability to preserve the main geometric characteristics (parallelism, straight lines) of the transformed objects. For this reason, it does not require a complex processing which aims at retrieving their original structure. On the other hand, the Projective deformation is more complicated to be handled, since it is a nonlinear transformation. This type of deformation renders the QR code trapezoid or in the form of an undefined geometric shape. Both affine and projective transformations are defined by the linear system (15). The only thing that makes difference between them is manifested on the used coefficients f_i related to the Homography matrix. As regards to the lens deformations, they are frequently caused by the use of inappropriate lens calibration, the evoked deformations are

generally represented by barrel, pincushion and Mustache distortions wherein the image becomes curved because of its magnification which varies irregularly compared to the optical axis i.e. these deformations occur when the field of view of the lens is much wider or smaller than the size of the image sensor.

The proposed QR code rectification algorithm infers the type of deformation which alters each detected QR code through its four corner coordinates. Once determined, the corresponding Homography matrix is computed (15) and thereafter the corresponding geometric correction is applied to the QR code. This by mapping its pixels to the new positions in the square area.

$$\begin{pmatrix} f_1 & f_2 & f_3 \\ f_4 & f_5 & f_6 \\ f_7 & f_8 & f_9 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = \begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} \qquad (15)$$

The sub matrix $\begin{pmatrix} f_1 & f_2 \\ f_4 & f_5 \end{pmatrix}$ is used when the QR code has undergone an affine deformation such as rotation (as shown in Fig 10). In case wherein the QR code is tilted by a given angle θ. The latter is calculated through the basic rules (SOH-CAH-TOA) related to the right triangle. The corresponding Homography matrix is defined as illustrated in (16)
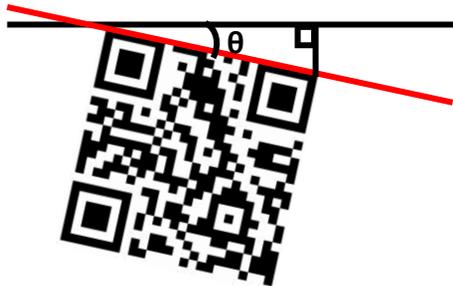

Fig. 10. Tilted QR code by an angle θ

$$\begin{pmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = \begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} \qquad (16)$$

When it comes to the projective deformation, a rigorous process must be conducted. The first step consists of defining a square area in which the corrected QR code will be placed. This area is defined by the four corners b1, b2, b3 and b4 as shown in Fig 11(b). Afterwards the projective transformation matrix is computed basing on the relationship between the four corner coordinates (a1, a2, a3 and a4 as shown in Fig 11(a)) related to the deformed QR code with their corresponding ones in the square area in Fig 11(b). Once the projective matrix is computed, all the pixels of the deformed QR code are mapped to the square area, By doing so, the corrected QR code is obtained as shown in Fig 11(b).
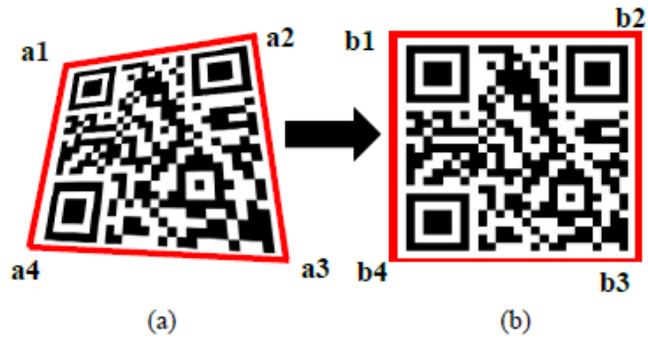

Fig. 11. Shape rectification of the deformed QR code based on projective transformation. (a) Deformed QR code (b) QR code after shape rectification

V. RESULTS AND DISCUSSION

The proposed system has been tested in:

- Laptop characterised by an AMD C-60 processor running at up to 1,33 GHz with 2Go of RAM, and an HD camera with USB interface.

- Raspberry pi 2 characterised by quad-core ARM Cortex-A7 processor running at up to 900MHz with 1Go of RAM, and an HD camera offering a resolution of 2560x1536, the evoked equipment are put onboard a robot.

Table 2 displays the obtained results after comparing both the required executing time and performance related to the Principal Components Analysis (**PCA**) with those of the random forests (**RF**), the support vector machine (**SVM**) and the correlation method (**CR**). In order to ensure a good prediction, both RF and SVM require a huge sample of training patterns accomagned with the set of patterns to be classified. Although their high accuracy, the size of their required training data increases notably their processing time (as reported in Table 2). As regards to the correlation method, the pattern comparisons is performed through the correlation between the pixels of the extracted pattern with those of the training one. this method is characterized by its low accuracy and high sensitivity towards image deformations. On the other hand, it clearly appears that the Principal Components Analysis outperforms all the aforementioned methods in terms of accuracy and its low processing time. The tests have been conducted using different pattern image sizes. The latter are set at 11x11, 22x22, 88x88, 176x176 and 352x352.

TABLE. II. COMPARISON BETWEEN PRINCIPAL COMPONENTS ANALYSIS PERFORMANCE AND OTHER METHODS OF PATTERN CLASSIFICATIONS

Methods	352x352	176x176	88x88	22x22	11x11	Accuracy
	Processing time (seconds)					
PCA	0.62	0.41	0.30	0.25	0.20	96%
RF	3.67	3.19	2.86	2.38	2.16	92%
SVM	2.60	2.16	1.97	1.85	1.82	85%
CR	1.33	0.71	0.59	0.42	0.39	68%

VI. CONCLUSION

In this paper, a streamlined QR code recognition system has been proposed to be efficiently run in embedded systems. The implementation of the Principal Components Analysis is opted in order to reduce the dimension of the original data. i.e. the extracted patterns are analyzed and filtered through their corresponding Loading matrix coefficients (generated by the PCA) instead of their original images. The proposed system has been tested under different criteria and taking into account several image deformations. The obtained results were promising.

The proposed QR code recognition system performs a series of treatments. It starts first by the input image segmentation based on an achromatic filter. This segmentation aims at extracting the candidate regions of interest (ROI) which may contain QR codes. Afterwards, the extracted ROIs are binarized and scanned horizontally and vertically in order to localize the candidate QR code patterns within the resulting ROIs. The extracted patterns are then filtered by means of the PCA accompanied with the Euclidian similarity measurement. By doing so, all the false detected patterns are removed. Therefore, the QR code localizations are launched by assembling the remaining patterns into groups. Each group (compounded exclusively by three FPs and at least one AP) allows localizing a QR code. Finally, the obtained QR codes are assessed and rectified in case of structural deformations.

Future directions of research concern the integration of an additional processing which will be implemented at the beginning of the system. This process aims at evaluating the quality of the captured image in terms of blur and photometric distortion. The corrupted images are automatically discarded.

REFERENCES

[1] B. Tiryakioglu, G. Kayakutlu and I. Duzdar, "Medical device tracking via QR code and efficiency analyze," 2016 Portland International Conference on Management of Engineering and Technology (PICMET), Honolulu, HI, 2016, pp. 3115-3128.

[2] D. Jagodić, D. Vujičić and S. Ranđić, "Android system for identification of objects based on QR code," 2015 23rd Telecommunications Forum Telfor (TELFOR), Belgrade, 2015, pp. 922-925.

[3] G. S. Vardhan, N. Sivadasan and A. Dutta, "QR-code based chipless RFID system for unique identification," 2016 IEEE International Conference on RFID Technology and Applications (RFID-TA), Foshan, 2016, pp. 35-39.

[4] N. Bhardwaj, R. Kumar, R. Verma, A. Jindal and A. P. Bhondekar, "Decoding algorithm for color QR code: A mobile scanner application," 2016 International Conference on Recent Trends in Information Technology (ICRTIT), Chennai, 2016, pp. 1-6.

[5] Z. Yang, Z. Cheng, C. C. Loy, W. C. Lau, C. M. Li and G. Li, "Towards robust color recovery for high-capacity color QR codes," 2016 IEEE International Conference on Image Processing (ICIP), Phoenix, AZ, 2016, pp. 2866-2870.

[6] Y. L. Lin and C. M. Sung, "Preliminary study on QR code detection using HOG and AdaBoost," 2015 7th International Conference of Soft Computing and Pattern Recognition (SoCPaR), Fukuoka, 2015, pp. 318-321.

[7] M. Ostkamp, S. Luzar and G. Bauer, "QR codes on curved media facades: Two approaches for inverse distortion based on raytracing and image warping," 2014 International Conference on Computer Graphics Theory and Applications (GRAPP), Lisbon, Portugal, 2014, pp. 1-6.

[8] M. Ahn, S. Hong and S. Lee, "A research on the QR Code recognition improvement using the cloud-based pre-generated image matching scheme," 2015 International Conference on Information Networking (ICOIN), Cambodia, 2015, pp. 356-357.

[9] 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.

[10] Z. l. Liao, T. l. Huang, R. Wang and X. y. Zhou, "A method of image analysis for QR code recognition," 2010 International Conference on Intelligent Computing and Integrated Systems, Guilin, 2010, pp. 250-253.

[11] Y. Liu and M. Liu, "Automatic Recognition Algorithm of Quick Response Code Based on Embedded System," Sixth International Conference on Intelligent Systems Design and Applications, Jinan, 2006, pp. 783-788.

[12] Istvan Szentandrasi, Adam Herout, Marketa Dubska, "Fast Detection and Recognition of QR codes in High-Resolution Images", ACM Digital Library, May 2012, pp. 129-136

[13] Liu Huijuan, "Location and Segmentation of Quick Response Code Image", Journal of Electronic Measurement and Instrument, 2006,

[14] LIU Ning-zhong, YANG Jing-yu, "Recognition of Two-dimensional Bar Code Based on Fourier Transform", Journal of Image and Graphics, 2003.

[15] Peter Bodnar, Laszlo G. Nyul, "Improved QR Code Localization Using Boosted Cascade of Weak Classiers", acta cybernetica journal, 2015, pp. 21-33.

[16] Kong Suran, "QR Code Image Correction based on Corner Detection and Convex Hull Algorithm", Journal of Multimedia, 2013, pp 662-668.

[17] Gautam Garg, (2015), QR Code Capacity in Number of Characters. Available. [Online]: http://scanova.io/blog/blog/2015/04/22/qr-code-capacity/

[18] P. Moallem and K. Faez, "Search space reduction in the edge based stereo matching by context of disparity gradient limit," 2nd International Symposium on Image and Signal Processing and Analysis. In conjunction with 23rd International Conference on Information Technology Interfaces (IEEE Cat., Pula, 2001, pp. 164-169.

[19] E. Cho and D. Kim, "Foreground soft segmentation for search space reduction," 2013 10th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI), Jeju, 2013, pp. 249-250.

[20] Fatin Zaklouta, Bogdan Stanciulescu, "Real-time traffic sign recognition in three stages", Journal of robotics and autonomous systems (Elsevier), 2014, pp. 16-24.

[21] Method and apparatus for identifying scale invariant features in an image and use of same for locating an object in an image", David Lowe's patent for the SIFT algorithm, March 23, 2004

[22] Herbert Bay, Andreas Ess, Tinne Tuytelaars, Luc Van Gool, SURF: Speeded Up Robust Features", Computer Vision and Image Understanding (CVIU), Katholieke Leuven University, Vol. 110, No. 3, pp. 346–359, 2008

[23] Matas, J., O. Chum, M. Urba, and T. Pajdla. "Robust wide baseline stereo from maximally stable extremal regions." Proceedings of British Machine Vision Conference, pages 384-396, 2002.

[24] Jolliffe, I. T. Principal Component Analysis, Springer, 2002.