RESEARCH ARTICLE   OPEN ACCESS

Deblurring, Localization and Geometry Correction of 2D QR Bar Codes Using Richardson Lucy Method

Manpreet Kaur*, Amandeep Singh Bhandari**, Charanjit Singh***
*(Student, Department of Electronics and Communication Engg., Punjabi University, Patiala)
** (Assistant Professor, Department of Electronics and Communication Engg., Punjabi University, Patiala)
*** (Assistant Professor, Department of Electronics and Communication Engg., Punjabi University, Patiala)

Abstract
This paper includes the recognition of 2D QR bar codes. This paper describes the deblurring, localization and geometry correction of 2D QR bar codes. The images captured are blurred due motion between the image and the camera. Hence the image containing the QR barcode cannot be read by QR reader. To make the QR barcode readable the images are need to be deblurred. Lucy Richardson method and Weiner Deconvolution Method is used to deblurr and localize the bar code. From both of the methods Lucy Richardson Method is best because this method takes less time for execution than the other method. Simulink Model is used for the Geometry correction of the QR bar code. In future, we would like to investigate the generalization of our algorithm to handle more complicated motion blur.
KEYWORDS: - 2D QR bar code image, smart phone camera, Richardson Lucy Method, Vision assistant module from lab view.

I. INTRODUCTION

QR code is a short stand for quick response code as the creator intended the code to allow its contents to be decoded at high speed. QR Code is a two-dimensional symbol. It was invented in 1994 by Denso, one of major Toyota group companies, and approved as an ISO international standard (ISO/IEC18004) in June 2000. This two-dimensional symbol was initially intended for use in production control of automotive parts, but it has become widespread in other fields. Now QR Code is seen and used everyday everywhere. The primary objective was to create a "code read easily by the scanner equipment". The QR Code is an information matrix that carries meaningful in the vertical direction as nicely as the horizontal, hence the two-dimensional term. By carrying in each direction, QR Codes can carry up to several hundred times the amount of information carried by 1dimensional bar codes. QR Codes, otherwise known as "quick response codes" are part of a whole new way of connecting with consumers via their smart phones and mobile devices.
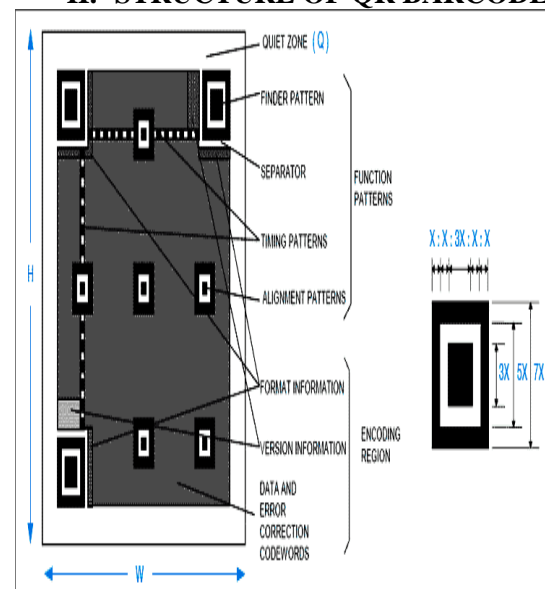
II. STRUCTURE OF QR BARCODE

Figure 1: Structure of Bar Code

Problem definition

We can see the QR barcode now a days can be seen everywhere. QR barcode is a 2D barcode and it can store more information in comparison to 1D barcode. QR barcodes are been used by many marketing companies for advertisements. Just one click of a QR barcode with a phone having barcode scanner application, can decode the barcode and one can retrieve information stored in it. Now the problem is when the image is captured may be the image is been captured in low contrast, uneven

illumination and blur motion can be added. One more problem may arise that is the geometry of the code may be distorted if while capturing the image projection angle is not correct. Now if the images are blurred and the geometry of the code has been distorted it is not possible to read the code as it is the requirement of the code reader that the image should be correct in geometry means the QR code should be a perfect square and image should not be blurred. Now algorithm should be such that it can deblur the blurred images and script can read the code properly and the second thing is algorithm should be able to correct the geometry of the code, it should make the code a perfect square so that it can be recognized by the script.

Experiments steps and measurement
Step1: The images captured are blurred due motion between the image and the camera. Hence the image containing the QR barcode cannot be read by QR reader. To make the QR barcode readable the images are need to be deblurred. Richardson Lucy Method and Wiener Deconvolution method is used to deblurr the image. For this step MATLAB has been used. Deblurring of the image is done in MATLAB.

Deblurring with the Lucy-Richardson Algorithm
Use the deconvlucy function to deblur an image using the accelerated, damped, Lucy-Richardson algorithm. The algorithm maximizes the likelihood that the resulting image, when convolved with the PSF, is an instance of the blurred image, assuming Poisson noise statistics. This function can be effective when you know the PSF, but know little about the noise. To illustrate a simple use of deconvlucy, a blurred, noisy image by convolving a Gaussian filter PSF with an image (using imfilter) and then adding Gaussian noise of variance V to the blurred image (using imnoise).
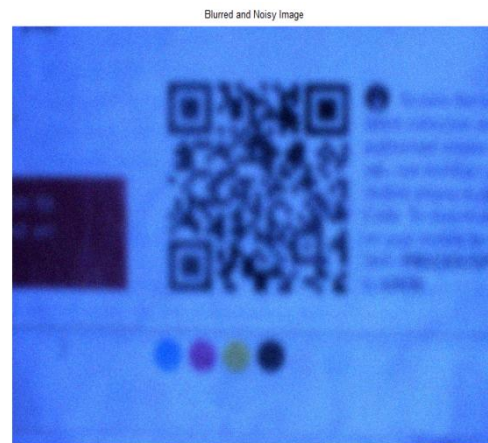

Figure 2: Original Image

Figure 3: Blurred Image

Deblurring the Image
Use deconvlucy to restore the blurred and noisy image, specifying the PSF used to create the blur, and limiting the number of iterations to 5.
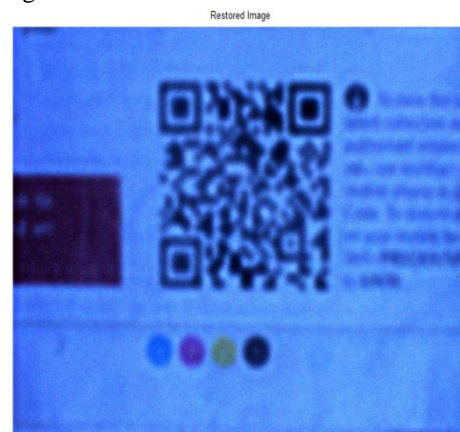

Figure 4: Restored Image

Refining the Result
The deconvlucy function, by default, performs multiple iterations of the deblurring process. You can stop the processing, after a certain number of iterations, to check the result, and then restart the iterations from the point where processing stopped. To do this, pass in the input image as a cell array, for example, {Blurred Noisy}. The deconvlucy function returns the output image as a cell array which you can then pass as an input argument to deconvlucy to restart the deconvolution.
The output cell array contains these four elements:
- output{1} -- The original input image
- output{2} -- The image produced by the last iteration
- output{3} -- The image produced by the next-to-last iteration

- output{4} -- Internal information used by deconvlucy to know where to restart the process
- The deconvlucy function supports several other optional arguments you can use to achieve the best possible result, such as specifying a damping parameter to handle additive noise in the blurred image. To see the impact of these optional arguments, view the Image Processing Toolbox Deblurring Demos.
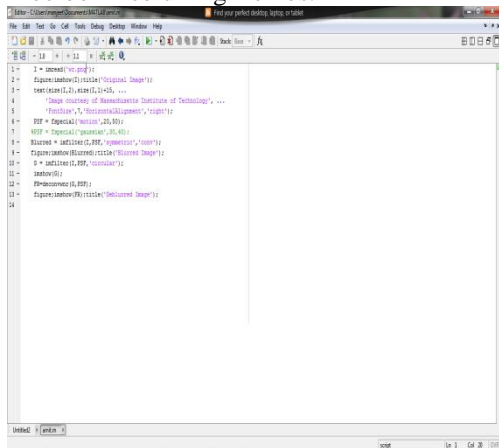

Figure 5: Matlab's Editor Window

Step2: after the image has been deblurred the QR code can be detected using vision assistant. So now we will use vision assistant to read QR code for that we will use few features of the software which are discussed below one by one with proper results. When we will start software the start window will appear like the figure below:
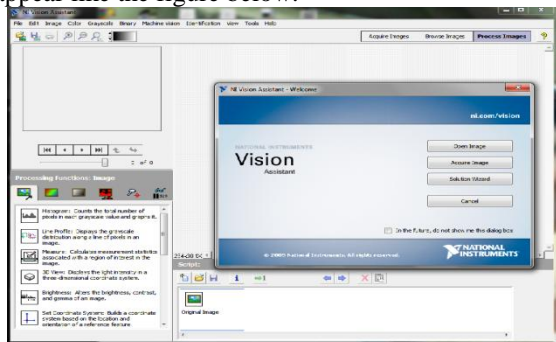

Figure 6: Vision demo

Geometry correction:

Simulink is an environment for multi-domain simulation and Model-Based Design for dynamic and embedded systems. It provides an interactive graphical environment and a customizable set of block libraries that let you design, simulate, implement, and test a variety of time-varying systems, including communications, controls, signal processing, video processing, and image processing. Simulink is integrated with MATLAB®, providing immediate access to an extensive range of tools that let you develop algorithms, analyze and visualize simulations, create batch processing scripts, customize the modeling environment, and define signal, parameter, and test data.

III. TO CREATE SIMULINK MODEL

Type simulink('open') in command window. It will open the simulink library browser window from where the necessary blocks can be selected to design simulink model. The window will appear like the figure below
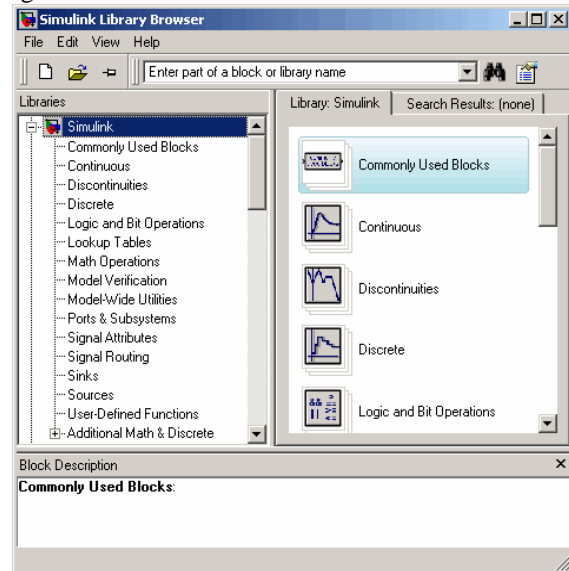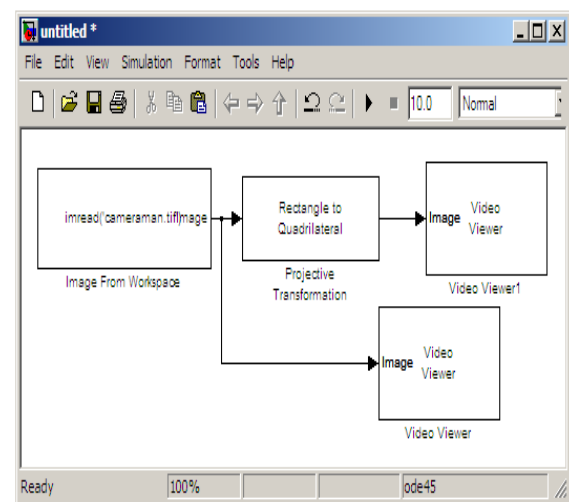

Figure 7: Simulink Module

Blocks Used-

Select the shape you want to convert. Your choices are Rectangle to quadrilateral, Quadrilateral to rectangle, or Quadrilateral to quadrilateral. Use the Projective Transformation block to transform your rectangular image into a quadrilateral. Set the Quadrilateral vertices [r1,c1,...,r4,c4] parameter. Use the Image from Workspace block to import an image into your model.

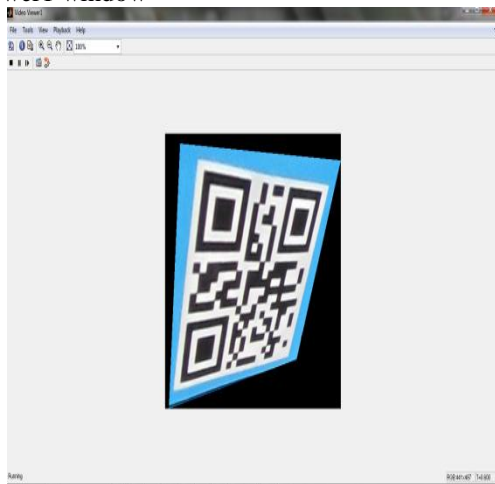The original quadrilateral image appears in the Video Viewer1 window

Figure 8: Input image

The square image appears in the Video Viewer2 window.

Figure 9: Output image

Results:-
Deblurring Of Images:-

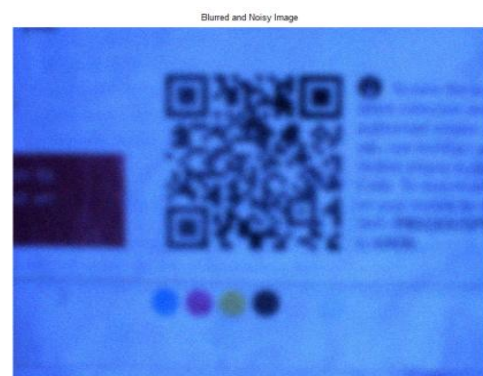**Figure 10:** Original image

Figure 11: Blurred image

Use deconvlucy to restore the blurred and noisy image, specifying the PSF used to create the blur, and limiting the number of iterations to 5.
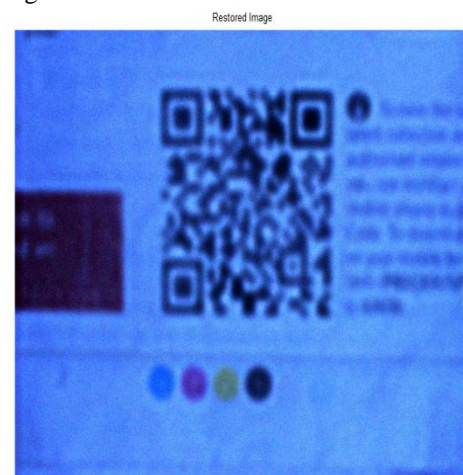

Figure 12: Deblurred image

Database of 50 deblurred images at different PSF values were taken and the estimated time has been noted and it conclude that 50 deblurred images can be read properly and the total estimated time taken by the algorithm to read 50 images is 14.1 seconds.

Geometry Correction: the QR barcode is in the form of a quadrilateral it is not a perfect square. If we try to read this barcode then code reader will fail to read it. So that we have to do is to first correct the geometry of the distorted code to make it a square so that the code can be read by our script. Now what we will do is, will run a MATLAB program. The program will find the four vertices of the quadrilateral and using that vertices in the Simulink model we will provide these coordinates to the projective transformation block as an input and accordingly the Simulink model will provide a square. Now this square code can be read by our script.
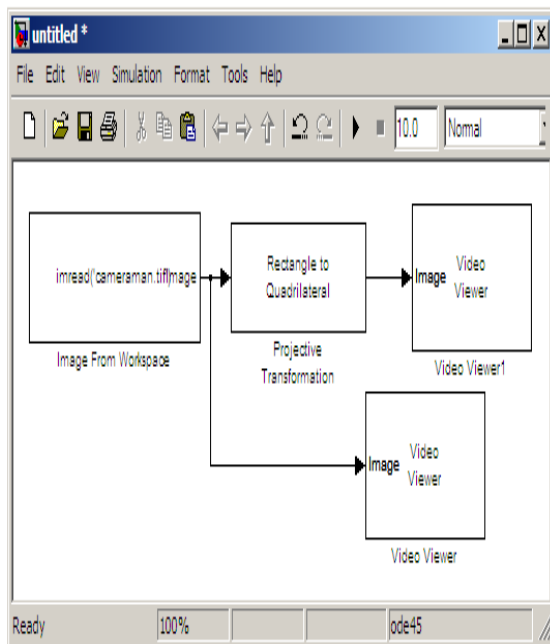
Run The Simulink Model:
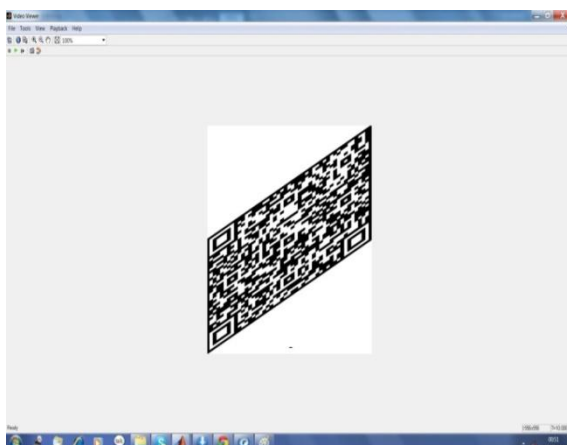

Figure 13: Simulink model for geometry correction

Figure 14: Output of video viewer: same as input

Figure 15: Output of video viewer1:-Required output.

IV. Conclusion

In this thesis we have to recognize the QR barcode which is blurred. First we have to deblurr the image using Richardson Lucy Method in MATLAB. For deblurring the image we choose different values of blur length (Len) and blur angle (theta) and we examine that at which value of Len and theta the image is deblurred and the code is readable. After deblurring the image the deblurred image is read in vision assistant and the time taken to read the code is estimated. The method is relatively accurate for lengths between 10 to 40, but breaks down at large blur lengths.

We deblurr three images using different values of Len and Theta and in this way we collect a database of 50 images and passed it through our script in vision assistant. The time estimated to read 50 deblurred images is 14.1secs. We observe that most of the computing time is spend on testing large blur sizes. Second thing we did is to correct the geometry of the distorted QR code using MATLAB, Labview and Simulink model. We write a program in MATLAB to find the four vertices of distorted code and with the help of these vertices in Simulink model we corrected the geometry. Now this corrected code is passed through our script and the QR code can be read.

V. Future Scope

In future, we would like to investigate the generalization of our algorithm to handle more complicated motion blur. In particular, we are interested in developing a localize version of our algorithm to identify the block-wise blurring kernels of a given blurred image. With such an algorithm in hand, we will be able to restore the motion blurred image caused by fast-moving objects in the scene.

References

[1]. Yanming Zhou, et al. (2006) "*Camera Readable 2D Bar Codes Design and Decoding for Mobile Phones*" in Image Processing, 2006 IEEE International Conference on Digital Object Identifier: 10. 1109/ICIP.2006.312495 Publication Year: 2006, Page(s): 469 – 472.

[2]. Hyotaek Lim, et al.(2010) "*Online banking authentication system usingmobile-OTP with QR-code*" in International Conference Computer Sciences and Convergence Information Technology (ICCIT), 2010 5th International Conference on Digital Object Identifier:10.1109/ICCIT. 2010.5711134 Publication Year: 2010, Page(s): 644 – 648.

[3]. Aidong Sun, et al.(2007) "*The QR-de reorganization in illegible snapshotstaken by mobile phones*" in Computational Science

and its Applications, 2007. ICCSA 2007. International Conference on Digital Object Identifier: 10.1109/ICCSA.2007.86 Publication Year: 2007, Page(s): 532 - 538 Cited by: 4.

[4]. Hanaizumi, et al.(2004) *"Barcode readers using the camera device in mobile phones"* in Cyber worlds, 2004 International Conference on Digital Object Identifier: 10. 1109/CW.2004.23 Publication Year: 2004, Page(s): 260 -265 Cited by: 11.

[5]. Hui Ji, et al. (2008) *"Motion blur identification from image gradients"* in Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on Digital Object Identifier: 10.1109/CVPR. 2008.4587537 Publication Year: 2008, Page (s): 1 - 8 Cited by: 5

[6]. Changsheng Chen, et al.(2010) *"quality measure of mobile phone captured 2Dbarco de images"* in Image Processing (ICIP), 2010 17th IEEE International Conference on Digital Object Identifier: 10.1109/ICIP. 2010.5654007 Publication Year: 2010, Page(s): 329 – 332.

[7]. ZhaoxuanYang, et al.(2010) *"robust and fast localization algorithm for data matrix barcode"* in optoelectronics and image processing(ICOIP), 2010 International conference on Volume:2 Digital Object Identifier: 10.1109/ICOIP.2010.299 Publication Year: 2010 , Page(s): 356 – 359.

[8]. Puchong Subpratatsavee, et al.(2014) *"HC2D Barcode Reader using Embedded Camera in Android Phone"* in 11th International Joint Conference on Computer Science and Software Engineering (JCSSE).

[9]. Bingsheng Zhang, et al.(2014) *"SBVLC: Secure Barcode-based Visible Light Communication for Smartphones"* in IEEE INFOCOM 2014 - IEEE Conference on Computer Communications.

[10]. Yu Xiaoyang, et al.(2013) *"Research and Achievement of QR Code Encryption Based on Cellular Automata"* in 2nd International Conference on Measurement, Information and Control.

[11]. Pei-Yu Lin, et al. (2013) *"Secret Hiding Mechanism Using QR Barcode"* in International Conference on Signal-Image Technology & Internet-Based Systems.

[12]. Marco Querini and Giuseppe F. Italiano, et al.(2013) *"Color Classifiers for 2D Color Barcodes"* in Proceedings of the 2013 Federated Conference on Computer Science and Information Systems pp. 611–618..

[13]. Saber Dinparast, et al.(2013) "A Mobile Payment Scheme Using 2D-Barcode" in

2013 5th Conference on Information and Knowledge Technology.

[14]. Toshihiko Wakahara, et al.(2011) *"Image Processing of 2-Dimensional Barcode"* in International Conference on Network-Based Information Systems.

[15]. Chanwoo Chun, et al.(2012) *"Pilot-based 2D-Barcode Image Restoration using Successive Interference Cancellation"*.