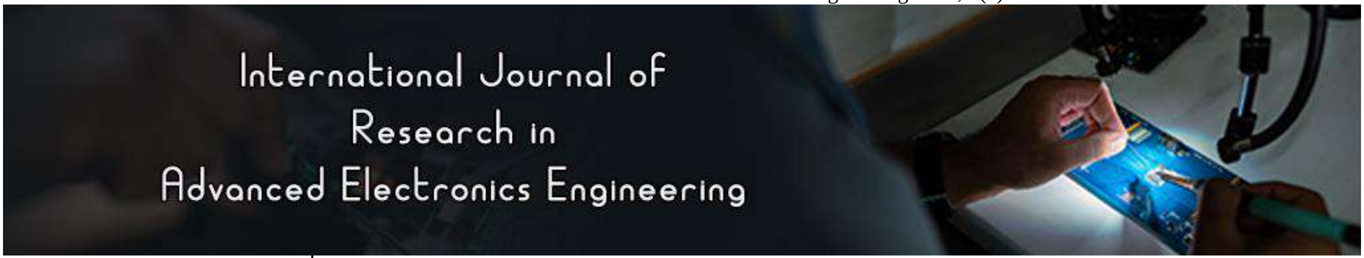


E-ISSN: 2708-4566
P-ISSN: 2708-4558
IJRAEE 2022; 3(1): 29-34

www.electrojournal.com
Received: 02-01-2022
Accepted: 09-03-2022

Dr. Manoj Kumar Mittal
Associate Professor, Shri Ram College Muzaffarnagar, Uttar Pradesh, India

Correspondence
Dr. Manoj Kumar Mittal
Associate Professor, Shri Ram College Muzaffarnagar, Uttar Pradesh, India

Simulation of camera motion paths and effect of different parameters of image stabilization including PSF and number of deconvolution iteration

Dr. Manoj Kumar Mittal

Abstract
Different camera motion paths are simulated and effect of different parameters of image stabilization observed, which includes PSF or kernel sizing, PSF shape, PSF normalization and Number of deconvolution iteration. The PSF simulated here will be estimated in real time scenario by noting camera motion in 3D using different motion sensors including accelerometer, gyroscope etc. Also spatially varying blur simulation and deblurring is to be simulated.

Keywords: Kernel sizing, PSF shape, PSF normalization, image degradation

Introduction
Image degradation takes place because of two factors [1]: Noise and Blurring. In image degradation image become glossy or lose clear vision of an image. It may add some type of additive noise like salt and pepper noise, Gaussian noise etc.
[2]. Image quality is a characteristic of an image that measures the perceived image degradation (typically, compared to an ideal or perfect image)
A blurred or degraded image can be approximately described by the equation given below:

$$g(x,y)= h(x,y)\otimes f(x,y)+\eta(x,y) \quad (1)$$

where, G is the degraded image, h is the distortion operator, also called the point spread function (PSF) or Kernel function. This function, when convolved with the image, creates the distortion, f *is* the original true image. The image f really doesn't exist. This image represents what you would have if you had perfect image acquisition conditions, η is the Additive noise, introduced during image acquisition, that corrupts the image $\eta \sim N(0, \sigma^2)$ is the noise. As convolution in special domain is analogous to multiplication in frequency domain, hence Eq. (1) can be rewritten in frequency domain as [3-6]:

$$G(u,v)= H(u,v)F(u,v)+N(u,v) \quad (2)$$

Terms in capital are Fourier transforms of corresponding terms in Eq. (1). By using definition of convolution, we can express same equation in vector matrix form also [7-9].

Blur Image Degradation
Sometime intentional blur can be used to great artistic effect in photography or computer vision. However, in many common imaging situations, blur is a nuisance. Camera motion blur often occurs in light-limited situations and is one of the most common reasons for discarding a photograph. If the blur function is known, the image can be improved by deblurring it with a non-blind deconvolution method. However, for most images, the blur function is unknown and must be recovered. Recovering both the blur or “point-spread function” (PSF) and the desired deblurred image from a single blurred input (known as the blind-deconvolution problem) is inherently ill-posed, as the observed blurred image provides only a partial constraint on the solution. Prior knowledge about the image or kernel can disambiguate the potential solutions and make deblurring more tractable [5, 10, 11]. The blurring or degradation, of an image can be caused by many factors like:

- **Motion Blur:** Movement during the image capture process by the camera or when long exposure times are taken by the subject.
- **Out-of-focus:** optics, use of a wide-angle lens, atmospheric turbulence or a short exposure time, which reduces the number of photons captured.
- Atmospheric turbulence [8].

There are different types of blur in image such as: Fragment blur, Gaussian Blur, Motion Blur, Radial Blur, Surface Blur, Unfocus blur, and Zoom Blur. Among all the types of image degradation mechanisms motion, blur is more important and recurrent hence the major focus of work carried out is on motion blur [12].

Camera motion blur

Image deblurring is the combination of two tightly coupled sub-problems: PSF estimation and non-blind image deconvolution. Both are longstanding problems in computer graphics, computer vision, and image processing [10, 13]. Blind deconvolution is an inherently ill-posed problem due to the loss of information during blurring [14]. Most image deblurring work assumes a spatially invariant kernel; however, this often does not hold in practice. Motion blur is the result of the relative motion between the camera and the scene during the integration time of the image.

There are many properties of camera and scene that can lead to spatially-varying blur:

1. Depth dependent defocus blur.
2. Defocus blur due to focal length variation over the image plane.
3. Depth dependent blur due to camera translation.
4. Camera roll motion.
5. Camera yaw and pitch motion when there are strong perspective effects [8].

The spatially varying blur due to first two factors are; generally introduced by operators mistakes and hence can be ignored**.** Blur due to last three factors is having maximum effect on image quality [14]. The spatially invariant kernel in Eq. (1) is now replaced by a spatially-variant blur kernel, i.e., *A(d)* represented by a sparse-matrix or vector matrix for as:

$$A(d) = \pi \int_{D}^{3} A_t(d)\mathrm{dt} \tag{3}$$

where, $A_t(d)$ is a sparse re-sampling matrix that implements the image warping and re-sampling due o the homography and Spatially-varying blur model [14, 20]. It is given by:

$$\vec{g} = A(d)\vec{f} + n, \tag{4}$$

In Figure 1, the unblurred scene shown in (a) is blurred using three different simulated camera rotations about the "X" and "Y" axes. These blurring functions are depth invariant and for long focal lengths also shift invariant. In (b) and (c), the scene is blurred by linear horizontal and vertical motions, respectively. In (d), the scene is blurred due to circular motion. In practice, the space of possible motion paths is very large, which makes the problem of motion deblurring without prior knowledge of the motion, very hard to solve [10].

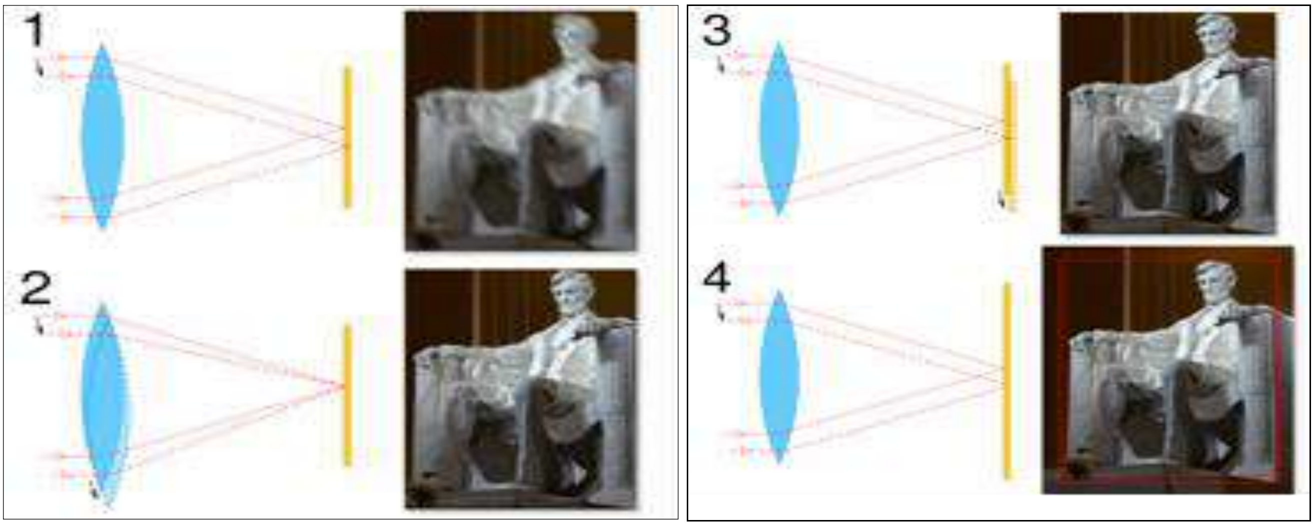


Fig 1: Different Camera Motions Lead to Different Motion Blurs.

Yaw, Pitch and Roll Motion of Camera

Camera motions are shown below in Figure 2.

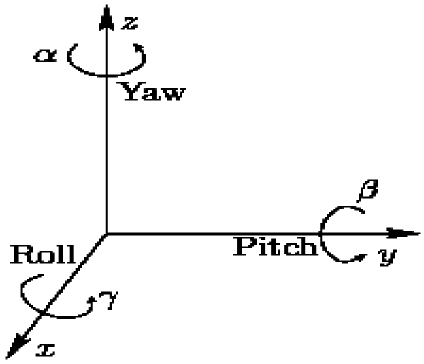


Fig 2: Yaw pitch and Roll Motion of Camera.

Yaw

Yaw motion is also known as panning. Panning refers to the horizontal scrolling of an image that is wider than the display. It creates folding effect in an image. Edges of image look like coming out of screen [16]. Mathematical model for representing yaw effect is [17]:

$$R_z(\alpha) = \begin{pmatrix} \cos\alpha & -\sin\alpha & 0 \\ \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{pmatrix}. \tag{5}$$

Pitch

Pitching, the camera up or down causes each image to become "keystoned." Keystoning causes rectangles, (e.g., window frames and door frames) to become trapezoids that are wider at top (camera pitching down) or at bottom (camera pitching up) [16]. Mathematical model for representing Pitch effect is:

$$R_y(\beta) = \begin{pmatrix} \cos\beta & 0 & \sin\beta \\ 0 & 1 & 0 \\ -\sin\beta & 0 & \cos\beta \end{pmatrix}. \tag{6}$$

Roll

Pitching, the camera up or down causes each image to become blurred at edges [16]. Mathematical model for representing Pitch effect is:

$$R_x(\gamma) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\gamma & -\sin\gamma \\ 0 & \sin\gamma & \cos\gamma \end{pmatrix}. \tag{7}$$

Combine Effect of Yaw, Pitch and Roll

Mathematical model for representing Pitch effect [16] is:

$$R(\alpha,\beta,\gamma) = R_z(\alpha)\,R_y(\beta)\,R_x(\gamma) = \begin{pmatrix} \cos\alpha\cos\beta & \cos\alpha\sin\beta\sin\gamma - \sin\alpha\cos\gamma & \cos\alpha\sin\beta\cos\gamma + \sin\alpha\sin\gamma \\ \sin\alpha\cos\beta & \sin\alpha\sin\beta\sin\gamma + \cos\alpha\cos\gamma & \sin\alpha\sin\beta\cos\gamma - \cos\alpha\sin\gamma \\ -\sin\beta & \cos\beta\sin\gamma & \cos\beta\cos\gamma \end{pmatrix}. \tag{8}$$

The over all effect of yaw, pitch and roll can be simulated with a PSF. Any among these movement can be modeled by simple linear or circular movment or motion path of camera.

PSF simulation for different motion paths

This section highlights observations and results of MATLAB work carried out. Simulation for different motion path is based on mathematical model proposed in Eq. (9). Said equation is based on assumption that image is free of any additive noise. Hence to blur image only convolution operation of original image with PSF or kernel is required [18].

PSF is the point-spread function with which image was convolved. PSF or kernel is nothing but a filter mask with ability to spread energy of a pixel with its neighbours. De-convolution is the process of reversing the effect of convolution.

$g(x,y)= h(x,y) \otimes f(x,y)$ (9)

Different Possible Motion Paths

The simulated PSF explained above is obtained or estimated here by analysing motion of camera or relative motion of camera and object being captured [19].

Horizontal Blur

In this blur camera motion expected, is in horizontal direction with pixel displacement of three pixels in said direction. Hence kernel used is of size 3X3 matrix as shown below in Figure 3a. It shows effect of this type of blur [19].

Vertical Blur

In this blur camera motion expected is in vertical direction with pixel displacement of three pixels in said direction. Hence kernel used is of size 3X3 matrix as shown below in Figure 3b, shows effect of this type of blur [19].

Linear Diagonal (Shift Left)

In this blur camera motion expected is in diagonal direction in 45 degrees towards left with pixel displacement of three pixels in said direction. Hence kernel used is of size 3X3 matrix as shown below in Figure 3c. It shows effect of this type of blur [19].

Linear Diagonal (Shift Right)

In this blur camera motion expected is in diagonal direction in 45 degrees towards right with pixel displacement of three pixels in said direction. Hence kernel used is of size 3X3 matrix as shown below in Figure 4a. It shows effect of this type of blur [19].

Disk or Ring or O Shaped

In this blur camera motion expected is in ring shaped or disk shaped 7 pixels. Hence kernel used is of size 7X7 matrix as shown below in Figure 4b. It shows effect of this type of blur. This is one of the most complex motion path for deblurring problem [19].

Ramp or V Shaped

In this blur camera motion expected is in diagonal direction in 315 degrees towards right followed by 45 degree towards right with pixel displacement of three pixels in each direction. Hence kernel used is of size 4X6 matrix as shown below in Figure 4c. It shows effect of this type of blur [19].

Inverted Ramp or A Shaped

In this blur camera motion expected is in diagonal direction in 45 degrees towards right followed by 315 degree towards right with pixel displacement of three pixels in each direction. Hence kernel used is of size 4X6 matrix as shown below in Figure 5a. It shows effect of this type of blur [19].

S Shaped

In this blur camera motion expected is in S shaped path. This is most complex camera motion path and hence very difficult for deblurring. Kernel used is of size 11X8 matrix as shown below in Figure 5b. It shows effect of this type of blur. Ringing effect observed in deblurred image is very high [19].

Effect of PSF Normalization

Normalization of kernel or PSF means dividing PSF matrix by absolute value. The sum of mask values is always ensured to be 1, so that the smoothened image intensity is equal to the original image intensity in a uniform intensity region. Same effect is clearly visible in Figures 6 and 7 shown below.

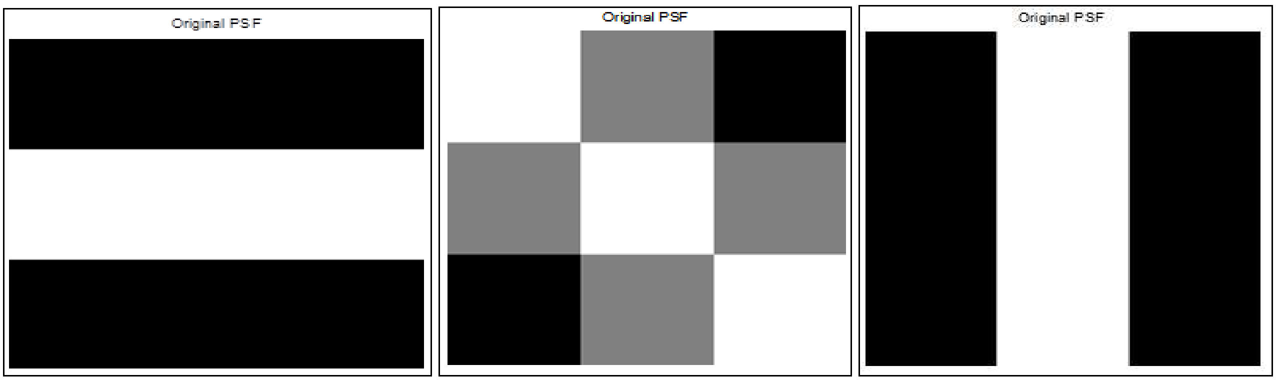


Fig 3: PSF, Original Image, Blurred Image and Restored Image for Horizontal, Vertical and Linear Shift Left Diagonal Motion Paths, Respectively.

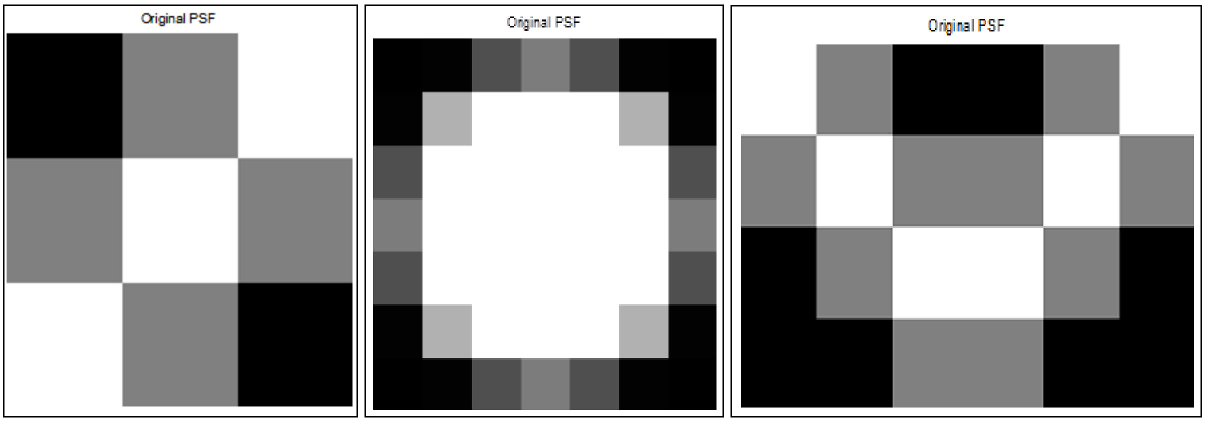


Fig 4: PSF, Linear Shift Right Diagonal, O Shaped and V Shaped Motion Paths, Respectively.

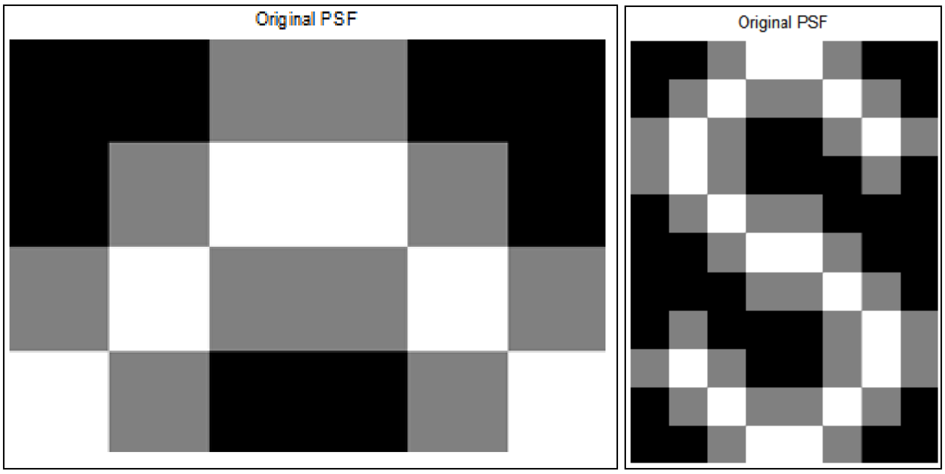
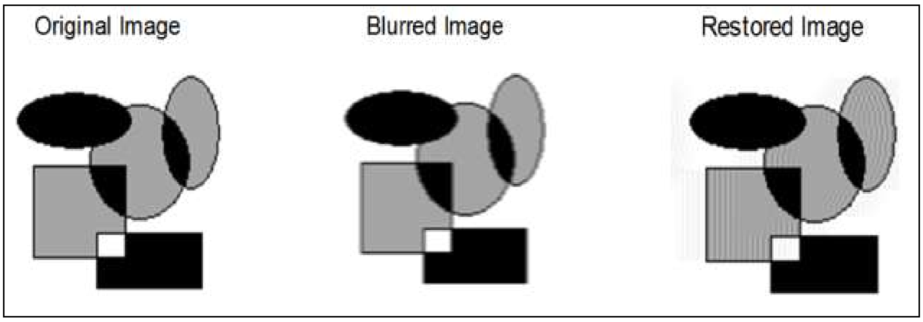
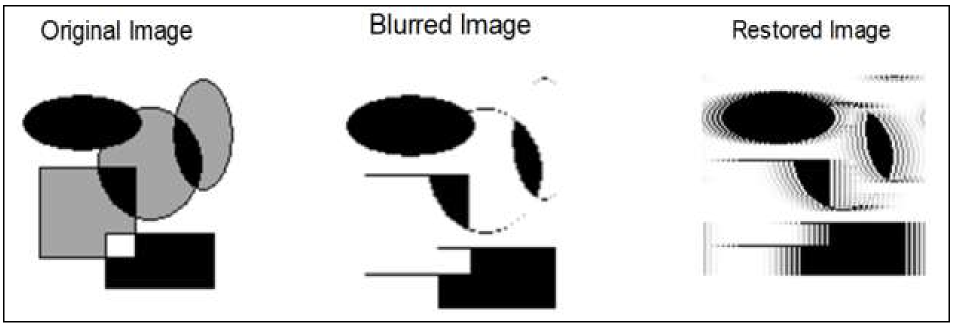


Fig 5: PSF, Inverted V Shaped and S Shaped Motion Paths, Respectively.



(a) (b) (c)

Fig 6: Original Image, Output of Blurring Function i.e., Imfilter and Restored Output with Normalized Deblurring Kernel.



(a) (b) (c)

Fig 7: Original Image, Output of Blurring Function i.e., Imfilter and Restored Output if Non-Normalized Deblurring Kernel.

Effect of Kernel Size or PSF Size

Having a bigger kernel size; makes computations faster at the expense of greater smoothening or blurring. This is because, in the latter case, the intensity of one pixel is dependent on more number of neighbouring pixels [20].

Effect of Number of Iterations

More number of iterations improves performance of deblurring algorithm. In MATLAB deconvblind () function shows improvement in deblurred output if number of iterations increases [20].

Conclusion

Different camera motion paths are simulated and effect of different parameters of image stabilization observed, which includes PSF or kernel sizing, PSF shape, PSF normalization and Number of deconvolution iteration. The PSF simulated here will be estimated in real time scenario by noting camera motion in 3D using different motion sensors including accelerometer, gyroscope etc. Also spatially varying blur simulation and deblurring is to be simulated.

References

1. Neel Joshi, Sing Bing Kang, C. Lawrence Zitnick, *et al.* Image Deblurring using Inertial Measurement Sensors, *in* ACM T. Graphics. 2010;29(4).
2. Ben-Ezra M, Nayar SK. Motion-based Motion Deblurring, IEEE T. Pattern Anal. 2004;26(6):689-698.
3. http://en.wikipedia.org/wiki/Image_noise
4. Rafael Gonzalez C, Richard Woods E. Digital Image Processing. 3rd Ed. CT: Pearson Education, New Jersey, 2008.
5. http://jnm.snmjournals.org/content/45/9/1519.full.
6. http://www.mathworks.in/help/toolbox/images/f9-5358.html
7. Raskar R, Agrawal A, Tumblin J. Coded Exposure Photography: Motion Deblurring using Fluttered Shutter, ACM T. Graphics. 2006;25(3):795-804.
8. Richardson WH. Bayesian-Based Iterative Method of Image Restoration. J. Opt. Soc. Am, 1972, 55-59.
9. Fergus R, Singh B, Hertzmann A, *et al.* Removing Camera Shake from a Single Photograph, in ACM T. Graphic. 2006, 787-794.
10. http://en.wikipedia.org/wiki/Canon_EOS_450D
11. Steve Webster, Andrew Dorrell. Improved Motion Invariant Imaging with Time Varying Shutter Functions in Digital Photography VII, P. SPIE-IS and T Electronic Imaging, SPIE, 2011, 7876.
12. http://www.getpaint.net/doc/latest/EffectsNoiseMenu.html
13. Levin A, Weiss Y, Durand F, *et al.* Understanding and Evaluating Blind Deconvolution Algorithms, in Computer Vision and Pattern Recognition. CVPR IEEE Conference, 2009.
14. http://www.getpaint.net/doc/latest/EffectsBlursMenu.html
15. Sahoo RK, Sabat AK, Nayak RK, Sahoo LN. On a method of estimating variance of the product estimator. International Journal of Statistics and Applied Mathematics. 2021;6(6):16-23.
16. Yuan L, Sun J, Quan L, *et al.* Image Deblurring with Blurred/Noisy Image Pairs. ACM T. Graphic. 2007;26(3).
17. http://sidvind.com/wiki/Yaw,_pitch,_roll_camera
18. Arthur Weeks Jr. Fundamentals of Digital Image Processing, 2nd Ed. PHI Publications, 1996.
19. Jain AK. Fundamentals of Digital Image Processing, 3rd Ed: ISBN0780334108, Pearson Education, 1989
20. http://www.mathworks.in/help/toolbox/images/bqqhld4.html