



International Journal of Emerging Technologies in Computational and Applied Sciences (IJETCAS)

www.iasir.net

No-reference blur evaluation method for images based on edge analysis and segmentation using Canny Edge Detector

Md. Zakir Hossen, and Z. M. Parvez Sazzad

Department of Applied Physics, Electronics and Communication Engineering,
University of Dhaka, Dhaka-1000, Bangladesh

Abstract: In this paper, we develop a no-reference blur evaluation method for images based on edge analysis and segmentation in spatial domain. Blur mainly smoothes the image signal which causes the reduction of edge points. Consequently edge information of an image is estimated using canny edge detector. We believe perceptual blur of any image are strongly dependent on local features, such as plane and non-plane areas. Therefore, plane and non-plane areas based edge detection rate are evaluated in this method. Subjective experiments results are used to verify the method. The result indicates that the proposed method can efficiently predict blur of images.

Keywords: No reference, blur, segmentation, canny edge detector

I. Introduction

In the modern period of Internet along with digital imaging and communication system, perceptual image quality evaluation is very important task as consumers are getting more educated with these technologies. Blur is one of the important artifacts which degrade visual image quality very strongly [1], [2]. In order to improve the perceptual image quality several objective quality metrics such as full reference (FR), reduced reference (RR) and no reference (NR) are used [3]. Among these quality metrics, NR method is the most attractive since it does not need any prior knowledge about the reference image in order to quantify the degraded one [4]. Now there are many artifacts that may occur in a distorted image, such as blocking, ringing, noise, and blur. Here we are mainly concerned with NR blur assessment, which remains an important problem in many applications. On top of that, we incorporate an important human visual system characteristic based on local features such as different sensitivity of human on uniform (i.e., area with less edges) and non-uniform (i.e., area with more edges) areas [5]. Here under the assumption that human visual perception is very sensitive to edge information of an image and any kinds of artifacts create pixel distortion. Therefore, in this work, a NR perceptual blur evaluation model is proposed based on edge information for plane and non-plane areas.

Fig. 1: Reference image and its blur version.



The rest of the paper is spitted as follows. In the next section, we describe our approach and the techniques to construct an automatic NR objective method for perceptual image blur assessment. The results and the discussion of my algorithm are given in Section III. Finally, Section IV concludes the paper. Future works are also mentioned in the same section.

II. Proposed Model

An image appears blurred when its high spatial frequency values in the spectrum are attenuated. A reference image and its blur version are shown in Fig. 1. There are different types of blurs. For example, motion blur due to the relative motion between the camera and the scene, and out of focus blur due to a defocused camera and lens aberrations. Blur can also be introduced when processing the image data, such as performing compression. Here, image segmentation is an important task for the extraction of useful information from the image such as the color, shape, texture and the structure. Image segmentation is usually done using edge detection techniques which detect the edges depending upon the level of intensity difference in pixels and the level of discontinuity [6]. The block based segmentation algorithm which is used in our proposed model is explained in the following section.

A. Block Based Segmentation

In order to classify edge and non-edge blocks of an image, we use a simple block based segmentation algorithm [4]. First, we establish a simple pixel based segmentation to classify each pixel within the image into either an

edge, or non-edge pixel. Initially, standard deviation (STD) of each pixel is estimated within its 3×3 and 5×5 neighborhood pixels. For all corners pixels, we take into account only available pixels for the measures. Let $STD_{3 \times 3}(m, n)$, and $STD_{5 \times 5}(m, n)$ be the standard deviated image of 3×3 and 5×5 neighborhood, respectively. Then we calculate absolute difference, $D_a(m, n)$ by the following equation:

$$D_a(m, n) = |STD_{3 \times 3}(m, n) - STD_{5 \times 5}(m, n)| \quad (1)$$

where $m = 1, 2, \dots, M$ and $n = 1, 2, \dots, N$.

$$D = \sqrt{\frac{\sum_{i=1}^{M \times N} (\overline{D_a} - D_{ai})^2}{M \times N}} \quad (2)$$

Subsequently, we calculate STD of $D_a(m, n)$ by

We then use the following algorithm to classify edge and non-edge pixels of the image.

$$P(m, n) = \begin{cases} 1 & \text{if } D_a(m, n) \geq D \\ 0 & \text{Otherwise} \end{cases}$$

where “1” and “0” denote edge and non-edge pixels respectively. Secondly, we classify each block (8×8) of the image into either edge or non-edge block by using the segmentation algorithm. The block based segmentation algorithm:

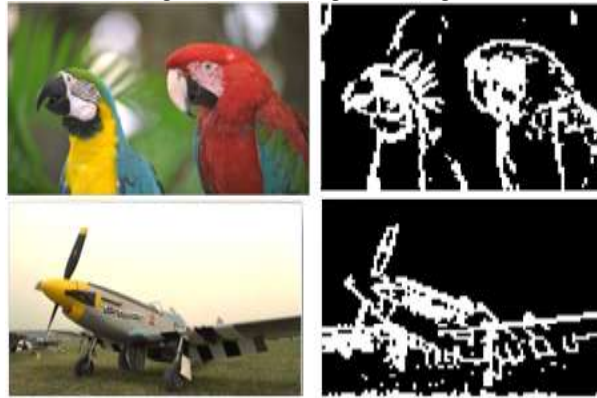
$$Sum = C_e + C_n \quad (3)$$

where C_e , and C_n are respectively the number of edge, and non-edge pixels per (8×8) block. Therefore, the “Sum” is the total number of pixels per block.

if $\left(\frac{C_e}{Sum} > th_n\right)$ then the block is “edge” else the block is “non-edge”

where “ th_n ” is the algorithmic threshold. The threshold estimation is described in details in Section III. In this segmentation algorithm, threshold “ $th_n = 0.19$ ” is estimated empirically. The threshold indicates that if more than 19% pixels within a block is edge, the block will be considered as “edge” block. Otherwise the block is defined as “non-edge” block.

Fig. 2: Block based segmented images



We consider two color test images of size 640×512 and 24 bits *RGB* and corresponding block based segmented images are shown in Fig. 2.

B. Canny Edge Detector

The purpose of edge detection in general is to significantly reduce the amount of data in an image while preserving the structural properties to be used for further image processing. The Canny detector is the most powerful edge detector provided by function `edge` [7]. In this method, firstly, image is smoothed to reduce noise using a Gaussian filter with a specified standard deviation, σ . The kernel of a Gaussian filter with a standard deviation of $\sigma = 1.4$ is shown in Equation (4).

$$B = \frac{1}{159} \begin{bmatrix} 2 & 4 & 5 & 4 & 2 \\ 4 & 9 & 12 & 9 & 4 \\ 5 & 12 & 15 & 12 & 5 \\ 4 & 9 & 12 & 9 & 4 \\ 2 & 4 & 5 & 4 & 2 \end{bmatrix} \quad (4)$$

Secondly, edge detection has been done through detecting meaningful discontinuities in intensity values by using first- and second-order derivatives. The first-order derivative of choice in image processing is the gradient [8]. The gradient of a 2-D function, $f(x,y)$, is defined as the vector:

$$\nabla f = \begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix} \quad (5)$$

The magnitude of this vector function is

$$\begin{aligned} \nabla f = \text{mag}(\nabla f) &= [G_x^2 + G_y^2]^{1/2} \\ &= \left[\left(\frac{\partial f}{\partial x} \right)^2 + \left(\frac{\partial f}{\partial y} \right)^2 \right]^{1/2} \end{aligned} \quad (6)$$

To simplify computation, this quantity is approximated sometimes by omitting the square-root operation

$$\nabla f \approx G_x^2 + G_y^2; \text{ or by using absolute values, } \nabla f \approx |G_x| + |G_y| \quad (7)$$

These approximations still behave as derivatives; that is, they are zero in areas of constant intensity and their values are proportional to the degree of intensity change in areas whose pixel values are variable

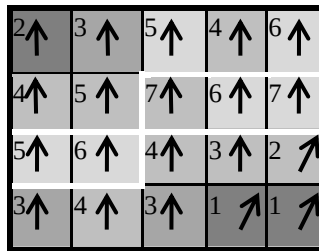
The gradient vector points in the direction of the maximum rate of change of f at coordinates (x,y) at which the angle, $\alpha(x,y) = \tan^{-1} \left(\frac{G_x}{G_y} \right)$ (8)

At each point, we computed the local gradient and edge direction. After that, determined edge points give rise to ridges in the gradient magnitude image. Thirdly, the purpose of non-maximum suppression is to convert the “blurred” edges in the image of the gradient magnitudes to “sharp” edges which is done by preserving all local maxima in the gradient image, and deleting everything else. The algorithm is for each pixel in the gradient image:

1. Round the gradient direction θ to nearest 45° , corresponding to the use of an 8-connected neighborhood.
2. Compare the edge strength of the current pixel with the edge strength of the pixel in the positive and negative gradient direction. That is if the gradient direction is north ($\theta = 90^\circ$), compare with the pixels to the north and south.
3. If the edge strength of the current pixel is largest; preserve the value of the edge strength. If not, suppress the value.

The algorithm then tracks along the top of these ridges and sets to zero all pixels that are not actually on the ridge top so as to give a thin line in the output. A simple example of non-maximum suppression is shown in Figure 3. Almost all pixels have gradient directions pointing north. They are therefore compared with the pixels above and below. The pixels that turn out to be maximal in this comparison are marked with white borders. All other pixels will be suppressed.

Fig. 3: Illustration of non-maximum suppression. The edge strengths are indicated both as the degree of shadow colors and numbers, while the gradient directions are shown as arrows. The resulting edge pixels are marked with white borders.



And, the ridge pixels are then threshold using two thresholds, T_1 and T_2 , with $T_1 < T_2$. Ridge pixels with values greater than T_2 are said to be “strong” edge pixels. Ridge pixels with values between T_1 and T_2 are said to be “weak” edge pixels. Finally, the algorithm performs edge linking by incorporating the weak pixels that are 8-connected to the strong pixels. The syntax for the Canny edge detector:

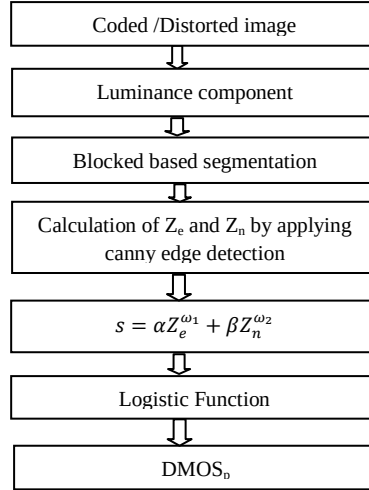
$$[g,t] = \text{edge}(f, 'canny', T, \text{sigma}) \quad (9)$$

where T is a vector, $T = [T_1, T_2]$, containing the two thresholds explained in step 3 of the preceding procedure, and sigma is the standard deviation of the smoothing filter. Here, t is included in the output argument; it is two-element vector containing the two threshold values used by the algorithm where thresholds have been chosen arbitrarily. Our implementation uses the iterative approach. First, all weak edges are scanned for neighbor edges and joined into groups. At the same time it is marked which groups are adjacent. Then all of these markings are examined to determine which groups of weak edges are connected to strong edges (directly or indirectly). All weak edges that are connected to strong edges are marked as strong edges themselves. The rest of the weak edges are suppressed. Figure 4 shows the edge detection with different threshold values where sigma is always 1.5.

Fig.4: Edged images by applying Canny edge detector. (a) Original; (b) $th_n[0.04, 0.1]$; (c) $th_n[0.06, 0.12]$ (d) $th_n[0.01, 0.07]$.



Fig. 5: NR blur evaluation model.



The proposed NR blur evaluation model is shown in Fig. 5. In this work, we consider only luminance component of color images for simplicity. Firstly, we apply the block (8×8) based segmentation algorithm which is discussed in Section II.A to classify edge and non-edge blocks in the image. Secondly, we apply the Canny edge detector algorithm to calculate strong Canny edge rate for each edge and non-edge block of the image. Thirdly, we average the Canny edge rate separately for edge and non-edge areas of the image. The average edge value of edge and non-edge blocks are calculated by the following equations:

$$Z_e = \frac{1}{N_e} \sum_{n=1}^{N_e} Z_{be} \quad (10)$$

$$Z_n = \frac{1}{N_n} \sum_{n=1}^{N_n} Z_{bn} \quad (11)$$

where Z_e and Z_n are respectively the average Canny edge rate value of edge and non-edge blocks of the images. And, also N_e , and N_n are respectively the number of edge and non-edge blocks of the image. The features are combined by the following equation:

$$S = \alpha Z_e^{\omega_1} + \beta Z_n^{\omega_2} \quad (12)$$

where α , β , ω_1 , and ω_2 are the model parameters. The model parameters are optimized by the Particle Swarm Optimization (PSO) algorithm with the subjective test data [9]. As, the model has not take into account the nonlinearity between the human perception and the physical features, therefore we consider a logistic function as the nonlinear property [4]. Finally, the differential mean opinion score prediction ($DMOS_p$) is evaluated by the following equation:

$$DMOS_p = \frac{99}{1 + \exp[-1.0217(S - 50)]} + 1 \quad (13)$$

Table 1: Optimum threshold estimation based proposed model's performance.

Th _n	Training			Testing		
	AAE	SRO	CC	AAE	SRO	CC
0.16	7.46	0.93	0.79	16.48	0.73	0.70
0.19	7.05	0.94	0.84	14.07	0.79	0.80
0.25	7.26	0.93	0.81	14.18	0.79	0.80
0.35	7.31	0.93	0.81	15.12	0.76	0.80

III. Results

A. Threshold Estimation and Performance of Our Proposed Model

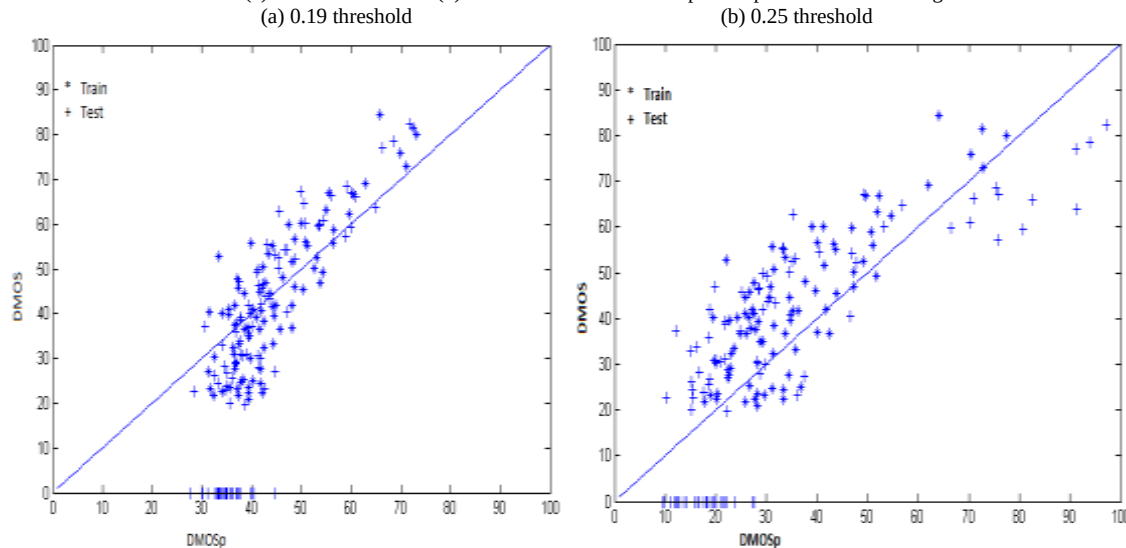
It is cleared from the segmentation algorithm (see section II .A) that the threshold value (“ th_n ”) must be within 0-1. We select four different “ th_n ” (0.16, 0.19, 0.25, 0.35) for all images which indicates the suitable threshold band of the proposed segmentation algorithm. As the target of this segmentation is used in perceptual blur assessment, the best suitable threshold (“ th_n ”) is estimated among the four values based on Spearman rank-order correlation coefficient (SRO) between differential mean opinion score (DMOS) and differential mean opinion score predictions (DMOSP). As, JPEG2000 coded images are mainly degraded by blurring. Therefore, perceptual quality of the coded images is indirectly representing blur prediction. Subsequently, in this work, we use JPEG2000 images database to verify the performance of our proposed model. Here, DMOS and DMOSP are respectively indicated subjective blur and predicted blur. The DMOS scale in the database is in the range 0-100. Higher value indicates less blur and high perceptual quality and vice versa. In order to evaluate and verify the best suitable algorithmic threshold among the four values for image blur assessment, we have considered LIVE database (JPEG2000 coded images) and divided randomly the database into two parts for training and testing [10]. As verification, we can compare the performances of the algorithm using the different threshold’s values. Since the target of the segmentation is used to improve the blur prediction, we have estimated all method parameters (α , β , ω_1 , and ω_2) for each threshold value separately by the optimization algorithm with the subjective test data. Here, we have calculated average absolute error (AAE), Spearman rank-order correlation coefficient (SRO), and linear correlation coefficient (CC) between DMOS and DMOSP for all training and testing images that are shown in Table 1. The optimum threshold is estimated by comparing the performances of the algorithm with the different threshold values based on minimum value of AAE and maximum value of SRO. Therefore, in all cases (in Table 1), the performance of the segmentation algorithm is sufficient with the threshold value of “ th_n ” = 0.19. The obtained method parameters by the PSO optimization algorithm for the threshold value of the training database (DMOS scale,0-100) are shown in Table 2.

Table 2: Method parameters and weighting factors for quality scale, 0-100.

$\alpha = -13.3801$	$\omega_1 = .0047$
$\beta = 59.3801$	$\omega_2 = 0.2012$

The DMOS versus DMOSP for all images (training, and testing) are shown in Fig. 6 with two threshold, th_n = 0.19, and 0.25. Fig. 6 indicates sufficient correlation between DMOS and DMOSP. And, the average value of CC is 0.82 for threshold value 0.19 and CC is 0.80 for the threshold value of 0.25. Therefore, the best suitable threshold value is 0.19.

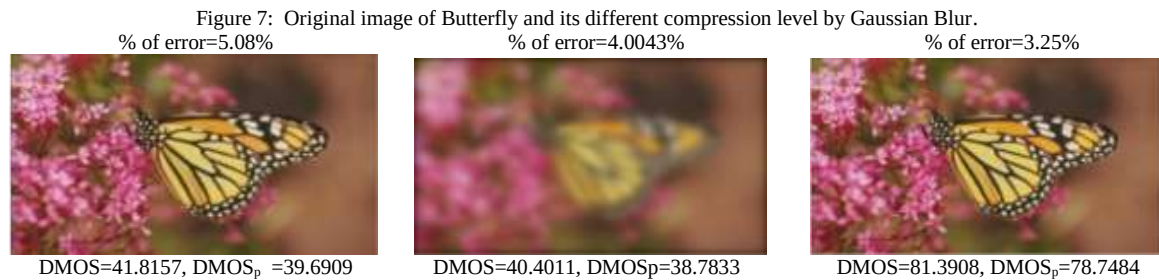
Figure 6: Scatter plot of difference mean opinion score (DMOS) versus DMOS prediction of the blurred image dataset in LIVE database contain (a) 0.19 threshold and (b) 0.25 threshold. Each data point represents one test image.



B. Performance Evaluation on Gaussian Blur Images

We also verify the performance of our proposed model on Gaussian blurred images with the same parameters and threshold value of 0.19. We considered twenty Gaussian blur images of different scene content. Out of all, five were reference images and each of the reference images were blurred by different level of Gaussian blur. Our proposed model’s CC on the Gaussian images is 0.86. The higher CC indicated that the prediction performance of our proposed model is also sufficient on the Gaussian images. In order to show the result, we

have taken three images of butterfly which are compressed according to different compression level of Gaussian blur. The DMOS and DMOS_p of the three images with percentage of errors are shown in Fig. 7. The figure shows that the DMOS_p scores are very much closed to the DMOS scores and the error values are within 5%. The small error indicates that our proposed model's performance is sufficient.



IV. Conclusion

A computational no-reference blur evaluation model is proposed JPEG2000 coded images based edge information measure on plan and non-plane areas. The plane and non-plane areas of an image are separated by a block based segmentation algorithm and edge information are calculated by Canny detection algorithm. The generalization ability of the proposed model is verified by JPEG2000 images database and Gaussian blurs images. The model's prediction is sufficient on the both types of images. In future, color information can be incorporated to the model to improve the prediction performance.

References

- [1] A. Ciancio, A. L. N. Targino da Costa, Eduardo A. B. da Silva, A. Said, R. Samadani, and P. Obrador, "No-Reference Blur Assessment of Digital Pictures Based on Multifeature Classifiers", IEEE Trans. on Image Processing, vol. 20, no. 1, pp. 64-75, 2011.
- [2] Pina Marzaliano, Frederic Dufox, Stephen Winkler and Touradj Ebrahimi, "A No-Reference Perceptual Blur metric" in Proc. IEEE ICIP vol.3, pp.57-60, 2002.
- [3] Z. M. Parvez Sazzad, Y. Horita, "Local region-based image quality assessment independent of JPEG and JPEG2000 coded color images" Journal of Electronic Imaging, Vol. 17, Issue 3, pp. 033002-033018, 2008.
- [4] Z. M. Parvez Sazzad, "No-Reference Image Quality Assessments for JPEG and JPEG2000 Coded image", PhD thesis, March 2008.
- [5] Roushain Akter, Z.M. Parvez Sazzad, J.Baltes, and Y. Horita, "No-Reference Stereoscopic Image Quality Assessment", in Proc. SPIE, vol.7524, Jan 18-22, Sun Jones, USA, 2010.
- [6] Yun-Chung-chung, Jung-ming Wang, Robert R BailySei wang, ChenShyang-Lih Chang, "A Non-Parametric Blur Measure Based on Edge Analysis for Image Processing Application" in Proc. IEEE Cybernetics and Intelligent Systems, Singapore, Dec 1-3, 2004.
- [7] Rafael C. Gonzalez, "Digital Image Processing Using MATLAB", ISBN 81-7758-898-2.
- [8] Ming Jun Cheng, Alan C Bovik, "No-Reference Blur Assessment using Multiscale Gradient" EURASIP Journal of Image and Video Processing, pp. 1-11, 2011.
- [9] J. Kennedy, and R. Eberhart, "Particle Swarm Optimization," Proc. IEEE ICNN, Pearth, Australia, pp. 1942-1948, 1995.
- [10] Laboratory for Image and Video Engineering, (LIVE database), The university of Texas, Austin.

V. Acknowledgement

Author would like to thank the lab of the University of Texas, Austin for providing the LIVE database.