ISSN:0975-9646

Rupali M. Bora et al, / (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 3 (4) , 2012,4696-4700

Detecting Image Forgeries using Discrepancies in Motion Blur

Rupali M. Bora, N. M. Shahane, Umesh K. Gaikwad

K. K. Wagh Institute of Engineering Education & Research, Nashik, Maharashtra, India

Abstract: **It is easy to manipulate images for malicious purposes due to widespread availability of photo manipulation software. Image splicing is one such form of tampering. Various methods have been proposed for detecting such splicing.**

An attempt has been made to detect splicing in images by searching discrepancies in motion blur. The motion blur estimation is through image gradients in order to detect inconsistencies between the spliced region and the rest of the image. A Blur Estimate Measure (BEM) is used to assist in inconsistent region segmentation in images that contain small amounts of motion blur and a no-reference Perceptual Blur Metric (PBM) is used to detect directional blurs in images. The selection of PBM is based on threshold, which should needs to be set. Based on these measures the regions of images are identified with consistent and inconsistent blurs.

To study the effect of Recursive Otsu's method, which deal with threshold-based segmentation and may give better results, is aim.

Keywords- **Image gradient, image forgery detection, motion blur estimation, Blur Estimate Measure, Perceptual Blur Metric, Recursive Otsu's method**

I. INTRODUCTION

A. Motivation

Forged and manipulated images have been produced in the media-driven society with great skills. Visual medium's power is compelling and such malicious tampering can have significant impact on people's perception of facts and events. Fake and misleading images can be used for introducing sensationalizing news, political propaganda, psychological bias, and propagating urban myths.

If the original image contains motion blur, how to detect the spliced region that had motion blur? Introducing motion blur into a spliced object, depends on the perception of the person creating the forgery and hence, is not likely to be completely consistent with the blur in the rest of the image. This fact is used to present a solution to this tampering detection problem. Specifically, splicing in a motion-blurred region is addressed. With the artificial blur introduced in the spliced part similar to the background blur, the inconsistency is difficult to perceive visually.[1]

For such inconsistency identification, some technique is required. So, here the technique is presented based on image gradients and gives better result to identify inconsistencies.

B. Existing Techniques

i) Spectral matting

1. Spectral matting basics

The spectral matting algorithm as developed by [5] segments an image into a number of soft matting components. Each pixel in an image is assumed to be a convex combination of K image layers $F_1, \ldots, F_K$, such that: $I_i = \sum_{k=1}^{K} \alpha_i^k F_i^k$

where α^k are the matting components of the image I with i as the pixel index. These specify the extent to which each layer is present in the color of the pixel observed. They are non-negative and sum to 1 at each pixel.[5]

2. Blur estimation

In order to estimate the motion blur, the α-motion blur constraint model as proposed in [4]. This method uses the gradient of the alpha-channel (matting components) to estimate the amount of blur present in various regions of the image. Since most of the pixels in the matte are opaque, the gradient in those regions is 0. It is reasonable to assume that wherever the gradient is non-zero, it satisfies the α-motion blur constraint: $\nabla \alpha_b . b = \pm 1$ where α_b is the blurred matting component and b is the blur vector.[5]

3. Methodology and algorithm

In order to identify tampered regions, the user first selects a region that is suspected to have been spliced into the image, in order to reduce the computation time required. This also helps avoid considering regions having very different motion blur (such as in-focus objects) along with a spliced region having a similar blur to the background, which would not result in the desired segmentation. Alternatively, instead of requiring the user to select a subimage, multiple blur model estimation with MRF segmentation, as described in [4], may be used.

This method works for regions with significantly different blurs, and each region can then be examined for consistency using the technique. The image is divided into a number of overlapping blocks, and the motion blur estimates are calculated for each block. Such an operation is made necessary by the memory required by the spectral matting process. However, using overlapping blocks has two major benefits: 1) Motion can be estimated at a number of points, as opposed to just the boundary between foreground and background, giving better resolution and 2) Space-invariance of motion blur can be assumed over each block, allowing for simpler calculations. Note that unsupervised grouping of the matting components is feasible because accurate segmentation is not a concern, but in fact, matte edge strengths.[2]

ii) *Image gradients*

1. Blur Estimation

The spectral characteristics of the image gradients are used as proposed in [8]. Such approach has been shown to be more robust to noise and somewhat non-uniform motion than just using the cepstrum of the image. For the case of uniform motion blur, the blurring process is modeled as the convolution of a sharp image with a blurring kernel:

$$I(x,y)=(H*P)(x,y))+N(x,y) \quad (1)$$

where I is the blurred image, H is the sharp image, P is the blurring kernel, and N is the noise present and (x,y) are the pixel coordinates.

For a horizontal uniform velocity motion blur, the blurring kernel can be modeled as $P_u=\frac{1}{L}[1,1,1,1\ldots.]_{1XL}$, where L is the length of the kernel.

Note that a directional blurring kernel $P_{\theta 0}$ can be formulated by rotating Pu by θ_0 degrees about the x-axis. Taking the Fourier transform of (1),

$$\hat{I}=\hat{H}\hat{P}+\hat{N} \quad (2)$$

A periodic pattern that is easier to detect also exists in the blurred image's gradient in the spectral domain. Differentiating (1),

$$I'=H'P'+N' \quad (3)$$

Taking the Fourier transform and omitting the noise term

$$\hat{I}'(\omega)=\hat{H}'(\omega)\hat{P}'(\omega) \quad (4)$$

The Radon transform, which is widely used for detecting straight lines in noisy images, is used. For a motion-blurred image, there are periodic large negative lines in $\log|\hat{I}'|$ with slope $P_{\theta 0}$ and periodicity 1/L. Denoting the Radon transform by R, $R(\log|\hat{I}'|)$, will have periodic peaks located at $(\pm 1/L, 90-\theta_0^\circ)$, $(\pm 2/L, 90-\theta_0^\circ)$, $(\pm 3/L, 90-\theta_0^\circ)$……..Therefore, this should correspond to a peak in the Fourier transform of $R(\log|\hat{I}'|)$.Denoting this Fourier transform of the Radon transform by F, let the peak in (ω_0,φ_0) $F(\log|\hat{I}'|)$ occur at,

Then

$$\varphi_0=90-\theta_0^{\;0} \qquad \omega_0=\frac{1}{L} \quad (5)$$

This estimated motion blur is represented as a two-element vector,

$$\phi=\left[\phi^{mag}\phi^{dir}\right] ,$$

Where $\phi^{dir}=\theta_0^{\;0}$ and $\phi^{mag}=L$

An issue arises if the original image has multiple motion blurs present. This is often the case when a moving object is imaged against a still background or vice versa. In [24], segmentation of an image into various regions depending on the estimated blur model for each pixel has been proposed. The technique is useful if the original image contains multiple motion blurs. In this case, the blur models could be separated. While this is appropriate for images with significantly different blurs, it is not feasible for image forgeries. As the region spliced into an image is desired to be concealed from detection, the blur added to it is likely to be similar to that of the surrounding region. In such a case, it would be difficult to estimate a separate blur model for the suspected region at a global level. A localized technique is necessary to separate blur models in case of forgeries. However, it is important to note that the spliced region is not segmented due to having a different blur compared to its surrounding region in any of these images. Therefore, such a technique is useful in dealing with cases in which multiple motion blurs are present in the original image itself, but not for detecting the forged region. Each segment can then be analyzed for consistency.

The technique assumes that the spliced region has a different intensity than its immediate background, which is reasonable.

II. Literature Review

A. Block-Level Analysis

Given an image having an artificially motion-blurred spliced region, it is not possible to extract multiple blur models over the whole image from its gradients, especially when the blurs are quite similar to each other. Hence, the blur is estimated at a local level allowing for different blur models to be estimated, without being lost in noisy data at a global level. The image is divided into M_b X N_b overlapping blocks $b_{m,n}$, $m=1toM_b$ and $n=1toN_b$, and the motion blur estimate $\phi_{m,n}$ for each block is calculated. $\phi_{m,n}$ is a two-dimensional vector $[\phi_{m,n}^{mag}\phi_{m,n}^{dir}]$ consisting of the motion blur estimate magnitudes and directions. The image subdivision has two major benefits:

1) Motion blur can be estimated at a number of points, as opposed to just a single estimate for the entire image, giving improved resolution, and
2) Space-invariance of motion blur can be assumed over each block, allowing for simpler calculations.[1]

B. Smoothing

The components of the motion blur estimates $\phi_{m,n}$ can be represented in magnitude and direction estimate matrices $\boldsymbol{\phi}^{mag}$ and $\boldsymbol{\phi}^{dir}$, respectively, each of size M_b X N_b, i.e.,

$$\boldsymbol{\vartheta}^k=\begin{pmatrix} \phi_{1,1}^k & \phi_{1,2}^k \ldots\ldots\ldots\ldots & \phi_{1,N_b}^k \\ \phi_{2,1}^k & \phi_{2,1}^k \ldots\ldots\ldots & \phi_{2,1}^k \\ . & . & . \\ . & . & . \\ \phi_{M_b,1}^k & \phi_{M_b,1}^k \ldots\ldots\ldots & \phi_{M_b,1}^k \end{pmatrix} \quad \ldots..(6)$$

where k=mag, dir

Since $\phi_{m,n}$ is calculated independently for each block $b_{m,n}$, a smoothing operation is used to correct for small variations in the estimated blurs. The magnitudes and the directions of the estimates are smoothed:

$$\boldsymbol{\phi}_{smooth}^k=\boldsymbol{\phi}^k * h^k \qquad k=mag,dir \quad (7)$$

Where $\boldsymbol{\phi}_{smooth}^k$ represent the respective smoothed estimates and is the smoothing filter employed. A disk filter was used. The estimates were obtained for a region containing different motion blurs. As can be seen, the local estimates display some variability at the border between the two regions due to the multiple blurs present in the same block for each local estimate. Smoothing allows the local estimates to vary in a non-abrupt manner, which results in better segmentation of the image in subsequent steps.[1]

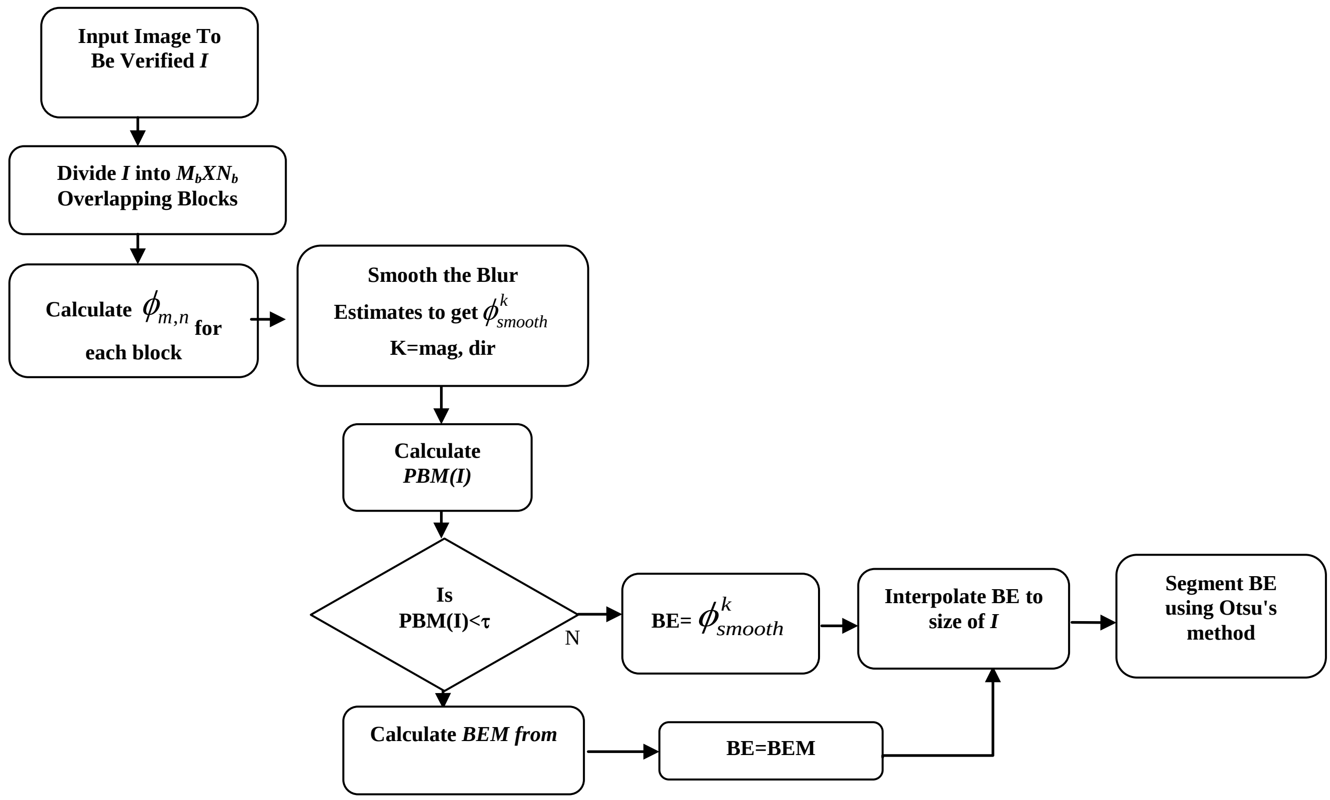

Fig. 1 Flowchart of existing technique

C. Blur Estimate Measures

For images in which certain regions appear to have little perceptible motion blur, the gradients of *I* along with the motion blur estimates ϕ^{mag}_{smooth} are used, generating a new *blur estimate measure (BEM)*, in order to improve robustness:

$$BEM(m,n) = \frac{\nabla I \cdot W_{i,j}}{\phi^{mag}_{smooth}(m,n)} \qquad (8)$$

Where $W_{i,j}$ is a neighborhood window located at the center (i ,j), in pixel coordinates, of each block $b_{m,n}$ of the image *I* and of the same size as the block. As formulated above, (8) can distinguish between the cases where there is little motion blur due to better focusing and where there is a small amount of motion blur due to the lack of enough texture to give significant information about the motion blur present. Only the magnitudes of the motion blur are considered as the direction estimates are perpendicular to the image gradients. Similar to $\mathbf{\Phi^k}$ in (6), *BEM(m,n)* can be arranged in a M_b X N_b matrix, as $\phi^k_{m,n}$ and *BEM(m,n)* are calculated block-wise.[1]

In order to automate the decision of using BEMs or just motion estimates, a modification is employed and the no-reference perceptual blur metric (*PBM*) presented in [6]. This metric is intended for Gaussian and compression blurs, necessitating a modification to deal with motion blur, which is not isotropic. Let S_α be the set of edge pixels in the binary edge map of obtained by applying the Sobel operator in the direction. A new metric, named as *oriented blur metric* PBM_α, is defined as

$$PBM_\alpha(I) = \frac{\sum_{p \in S_\alpha} E(p)}{|S_\alpha|} \qquad (9)$$

where *E(p)* is the width of the edge along the direction perpendicular to α at the edge pixel p and |.| denotes cardinality. In order to define E(p) in (12), let the two sides of the line in the direction α at some p be denoted by a and b, as shown in Fig. 2. Also, let the pixel locations of the first local maximum and minimum from this p, along the above perpendicular line, on the side a be denoted by $MA_a(p)$ and $MI_a(p)$, respectively. $MA_b(p)$ and $MI_b(p)$ are defined similarly. Then, the edge width is defined as

$$E(p)=min(|MA_a(p)- MI_b(p)|, |MA_b(p)- MI_a(p)|) \qquad (10)$$

The oriented PBMs are computer for orientations α_i, i=1 to t, where t is the number of orientations evaluated and then define the overall PBM as

$$PBM(I)=max(PBM_{\alpha i}) \qquad \forall \qquad \alpha_i \qquad (11)$$

This metric is used to determine the overall blurriness of the image as large edge widths, indicating blurred edges, give high values of *PBM(I)*. *BEM*s are chosen when it is below a predetermined threshold. The chosen blur estimate **BE** is given by

$$\mathbf{BE} = \begin{cases} \mathbf{BEM} & , if PBM \le \tau \\ \left[\phi^{mag}_{smooth} \phi^{dir}_{smooth} \right] & , otherwise \end{cases} \qquad (12)$$

It is to be noted that **BE** is a M_b X N_bX2 matrix in the latter case, as both magnitude and direction estimates are employed. In this case, all subsequent operations are carried out independently on each of the M_b X N_b component matrices.

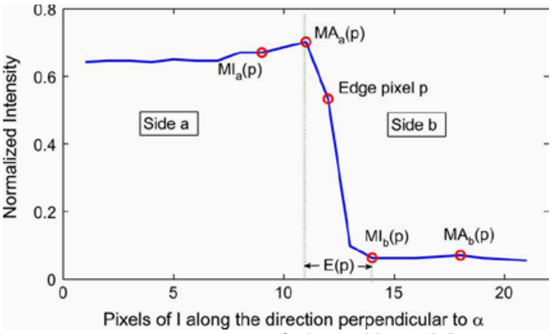

Fig. 2 Computation of edge width *E(p)*[1]

D. Interpolation

The motion blur estimate, **BE** are then upsampled to the size of *I* using bicubic interpolation, in order to have an estimate of the blur at each pixel. The accuracy of the estimate depends on the amount of upsampling done. Bicubic interpolation provides better results than nearest neighbor and bilinear interpolation. [1]

E. Segmentation

The image is segmented into two regions that exhibit different motion blurs. This is done by adaptively thresholding the upsampled **BE** using Otsu's method [7]. This method also provides an effectiveness metric which is used to discard images which show consistent directions and/or magnitudes in their motion blur estimates and hence cannot be segmented effectively. The result of segmenting the magnitude and direction of the estimates provides with an indication of regions with dissimilar motion blur.

In order to accomplish such segmentation, the mean values of the motion blur estimates of the two regions obtained by Otsu's method are used. [1]

III. SYSTEM ARCHITECTURE

A. Flow/Block Diagram

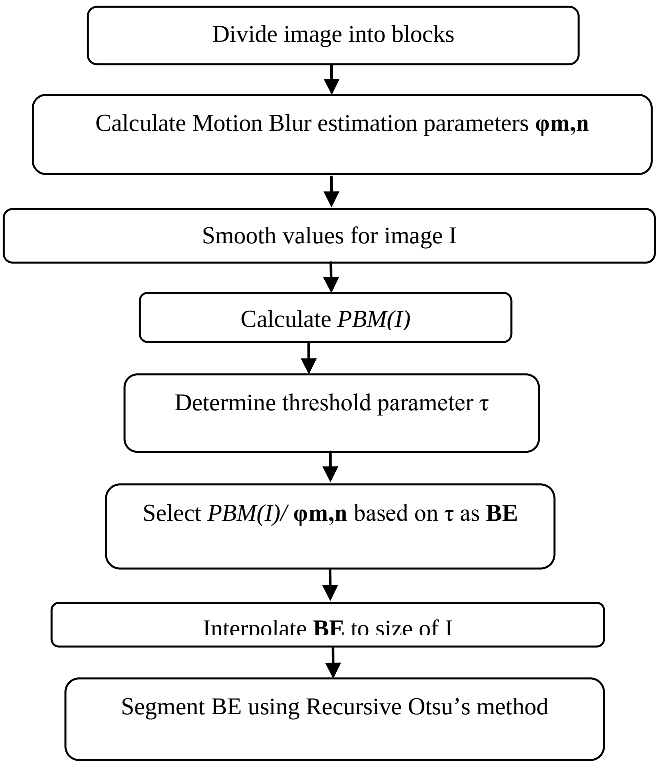

Fig. 3 Flow diagram of proposed technique

1. Divide image into blocks

Image is divided into subblocks in order to calculate the parameters at small region. Subblocks can be varied sizes which could be an evaluation criterion.

2. Calculate Motion Blur estimation parameters **φm,n**

For these parameters various methods are to evaluated which are mostly based on image gradients viz.using Frequency and Cepstral Domain [10][1], radon transform to find direction and t-norm operator to find its extend [9][11], spectral matting of image gradients[2].

The best method out of these methods and their values for parameters is selected for further process.

3. The further blocks are simplified to get the final segmentation of image objects as per motion blur parameters. Further blocks viz. Smoothing, calculating PBM values, determining threshold parameter, selection of BE based on this threshold, interpolation, can be simplified to reduce the time complexity and improve the speed of process with acceptable results.

Segmenting the image objects based on obtained parameters for motion blur can be done directly using Recursive Otsu's method.

B. Recursive Otsu's method

Otsu's method determines a global threshold for an image by minimizing the within-class variance for the resulting classes (foreground pixels and background pixels). This can be done equivalently maximizing the between-class variance for given threshold T. The technique fails in case where foreground and background are similar, so recursive Otsu's method is introduced.[3]

The key idea of recursive Otsu's method is to take the set of background pixels as determined by standard Otsu thresholding and rethreshold these pixels alone in order to extract more of next.[3]

C. Selection of threshold τ

The limiting factor τ for PBM should be selected accurately. Comparisons could be done for different values of τ.

D. Evaluation Criteria

Segmentation cost varies with respect to the block size and with respect to the block overlap. Block sizes could be 80X80, 100X100, 120X120, 140X140 and 160X160 pixels with overlaps of 10, 15, 20, 30 and 40 pixels. Performance could be evaluated based on these different values. It is required to check whether the results without overlapping are accurate than with overlapping [1].

IV. DESIGN AND WORK DONE

The flow of project starts with image subdivision into subblocks and its calculations for direction and blur extent.

Fig.4. Image with motion blur (magnitude=20, derection=45^0)

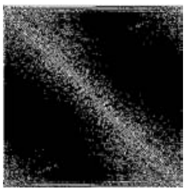

Fig.5. Fourier transform of Image of fig. 4

The image with motion blur can be created using fspecial function in MATLAB 7.10v for testing or using GIMP editor, the fourier transform shows the periodic large negative lines as shown in fig. 5. Radon transform gives the peaks of these lines as parameter for motion blur extent and direction.

These parameters are to be calculated using different approaches as previously stated and functionality is to be checked for the same against time complexity. And attempts will be made to make the entire process simplified.

V. CONCLUSIONS

A technique for detecting spliced images using discrepancies in motion blur is presented. Motion blur is observed in images of fast-moving subjects and may also appear due to camera shake. The fact is introducing motion blur into a spliced object, in general, depends on the perception of the person creating the forgery and hence, is unlikely to be completely consistent with the blur in the rest of the image.

The approach has been based on the method of spectral analysis of image gradients. The image gradients of a blurred image in the spectral domain display some periodic characteristics which are correlated with the amount and direction of motion blur present.

The suspected image is divided into overlapping blocks and the motion blur for each block is estimated. The various approaches are compared for the calculation of these parameters with reduced time complexity. This is followed by a post-processing step of smoothing the blur estimates and upsampling to the size of the image. The regions of the image which show inconsistent blur are then segmented from the image and displayed to the user. A blur estimate measure (BEM) is developed to provide robust segmentation, in the case of little perceptible blur. The presence of low blur is determined by using a perceptual blur metric.

The entire post-processing will be simplified using segmentation for the parameters obtained as a result of first two blocks. Segmentation will be achieved using Recursive Otsu's method.

REFERENCES

[1] P. Kakar, N. Sudha, Wee Ser, "Exposing Digital Image Forgeries by Detecting Discrepancies in Motion Blur", *IEEE TRANSACTIONS ON MULTIMEDIA*, VOL. 13, NO. 3, JUNE 2011, pp.-

[2] P. Kakar, N. Sudha, and W. Ser, "Detecting digital image forgeries through inconsistent motion blur," in *Proc. IEEE Int. Conf. Multimedia & Expo*, 2010, pp. 486–491.

[3] Nina, B. Morse, W. Barrett, "A recursive Otsu thresholding method for scanned document binarization", *in IEEE Workshop Applications of Computer Vision (WACV),* 2011,pp. 307-314

[4] S. Dai and Y.Wu, "Motion from blur," *in Proc. IEEE Conference on Computer Vision and Pattern Recognition CVPR* 2008, pp. 1–8.

[5] A. Levin, A. Rav-Acha, and D. Lischinski, "Spectral matting," *in Proc. IEEE Conference on Computer Vision and Pattern Recognition CVPR '07, June 17–22*, 2007, pp. 1–8.

[6] P. Marziliano, F. Dufaux, S.Winkler, and T. Ebrahimi, "A no-reference perceptual blur metric," *in Proc. IEEE Int. Conf. Image Processing, Citeseer*, 2002, vol. 3, pp. 57–60.

[7] N. Otsu, "A threshold selection method from gray-level histograms," *Automatica*, vol. 11, pp. 285–296, 1975.

[8] H. Ji and C. Liu, "Motion blur identification from image gradients," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2008, pp. 1–8.

[9] Ting-Ting Zhang, Gang Xu , "Identification Of Motion Blurred Parameter By Using T-Norm Operator", *In IEEE Proceedings Of The Sixth International Conference on Machine Learning and Cybernetics* 19-22 August 2007,pp. 1606-1610

[10] Ashwini M. Deshpande, Suprava Patnaik "A Non-uniform Motion Blur Parameter Identification and Restoration using Frequency and Cepstral Domain", *International Conference & Workshop on Recent Trends in Technology, (TCET) 2012 Proceedings*, pp.-11-17

[11] Hongwei SUN, Michel Desvignes, Yunhui YAN, Weiwei LIU, "Motion Blur Parameters Identification from Radon Transform Image Gradients", *IEEE 2009*