Restoration of Spatially-Varying Motion-Blurred Images

Shereen El-Shekheby, Rehab F. Abdel-Kader
Electrical Engineering Department
Faculty of Engineering, Port Said University
Port Said, Egypt
sherenzakaria@eng.psu.edu.eg,
rehabfarouk@eng.psu.edu.eg

Fayez W. Zaki
Department of Electronics and Communications
Engineering, Faculty of Engineering
Mansoura University
Mansoura, Egypt
fwzaki@mans.edu.eg

Abstract— **Restoration of spatially-varying blurred images is a challenging task needed in various computer vision systems and applications. In this paper, we propose a novel spatially varying blur detection and restoration method. Motion blur is detected automatically from a single image. Initially, the blur kernel length and direction are estimated by finding the kernel that maximizes the likelihood of a blurred local window. This is achieved by incorporating either vertical or positive diagonal kernels with various lengths. Then, initial blur regions are estimated using a kernel specific feature. Next, the initial blur regions are refined with the support of the image segmentation (CCP) method and neighboring information. Finally, Blurred regions are recovered using the best-estimated kernel. Comparisons with state of the art methods reported in the literature demonstrate accuracy improvements in the image blur detection and restoration results.**

Keywords—Image Restoration; Space-varying; Motion Blur; Image Segmentation; Blur Detection.

I. INTRODUCTION

Image restoration is extensively utilized in most practical image processing applications such as surveillance, astronomy, criminal detection, medical imaging, remote sensing, and the analysis of traffic accidents [1].

Image restoration is the process of recovering a clear version of the original image from an initially blurred image. Image blurring can be mathematically modeled as a convolution function between the proposed clear image and the point spread function (PSF) known as the blurring kernel

$$y = k \otimes x + n \tag{1}$$

Where y represents the initial blurry image, x is the clear image required to recover, k is the blur kernel, n represents the noise, and $' \otimes '$ represents the convolution operator. Both clear image to be recovered *(x)* and the blur kernel (k) are unknown.

Object motion and camera defocus are the most common causes of image blur. Object motion blur is produced by a moving object while the image is being captured. It can be modeled as a line characterized by angle and length. Defocus blur is caused by depths variations of the scene. Defocus blur can be modeled mathematically by a circular disk that characterized by the disk radius.

Image restoration; also called deconvolution/deblurring; is performed by reversing the process that caused the image blur in the first place. Image deconvolution problems can be classified into non-blind deconvolution and blind deconvolution. In the non-blind deconvolution, the blur kernel is estimated and is used to restore the original clear image x. Richardson-Lucy (RL) and Wiener deconvolution [2-4] was previously the most successful deblurring methods reported in the literature. These methods are still widely used because of their simplicity and effectiveness nevertheless; they may suffer from unpleasant artifacts. Non-blind deconvolution is not suitable for practical applications, wherefore blind deconvolution is required. In blind deconvolution, the blur kernel and the clear image must be estimated. Blurring kernel could be estimated first then used to evaluate the clear image or simultaneously estimate both of blur kernel and the clear image. Very successful blind deconvolution methods based on alternating minimization are introduced in [5-7].

Image blur kernel can be classified into a uniform blur (spatially-invariant) and non-uniform blur (spatially-variant). Uniform blur affects the entire image, but non-uniform blur affects different parts of the blurred image. It is challenging to perform space-varying (non-uniform) deblurring. To simplify space-varying image deblurring, some methods rely on the approximation of uniform blur [8-12]. However, these methods are ineffective when the spatial variability of the blur increases.

Goldstein and Fattal [12] estimate the blur kernel in motion blurred images. This method estimates the blurring kernel by modeling statistical irregularities in the power spectrum of the blurred images. It relies on spatially invariant blur and fails when the blur spatial variability escalates.

An alternative approach to deal with space-varying blur is to divide the blurred image into blocks. Each block is processed individually; each kernel is estimated and de-blurred independently [13]. However, the process remains problematic as there can be multiple objects within a single block.

In this paper, a robust spatially varying motion blur detection and restoration approach is proposed. Spatially varying blur affects various regions of the image differently; therefore, using a single blur kernel for all image regions will cause unpleasant artifacts in the restored image. The blur kernel for the blurred regions is precisely estimated and can then be used to restore a clear image with a considerable visual quality. Blur regions are detected automatically from a single blurry image without the need for any information about the blur kernel, the camera settings, or the input image. Comparison with the most successful methods reveals accuracy improvements in the image blur detection and restoration results.

The structure of this paper is organized as follows: in the next Section, a comprehensive survey of the related work is provided. In Section III, the proposed blur detection and de-blurring method is presented; Computational results and comparisons to the most successful existing algorithms are reported in Section IV. Finally, concluding remarks are summarized in Section V.

II. RELATED WORK

Images blur detection and restoration is a crucial topic in digital image processing research. Motion blur algorithms can work either with a single image [7, 8] or with multiple images [14, 15]. The use of more than one image for motion deblurring reduces applicability.

Most spatially-varying motion blur approaches are based on local image features [16-20]. Liu et al. [16] use some blur features and Bayes classifier to classify blur regions. This method suffers from low performance when applied to natural images. Chakrabarti et al. [17] determine the motion direction of moving objects using the likelihood of a small blurred window. This method is time-consuming and is evaluated using images with only a single moving object. Shi et al. [18] incorporate four discriminative blur features and a naïve Bayesian classifier. This method incorrectly interprets non-blurred pixels as blurred ones. Pang et al. [19] identify blur by presenting a new kernel specific feature vector and training the Support Vector Machine (SVM) classifier to distinguish blurred regions from non-blurred ones. Their experimental results outperform results reported by Liu et al. [16], Chakrabarti et al. [17], and Shi et al. [18]. Yang and Qin [20] use the method proposed by Shi et al. [18] to initially identify the blurred regions. Next, the classification of the blurred areas is done by using a blur classification algorithm. Finally, the blur kernel is estimated, and the blurred parts are restored using the Total Variation (TV) based on a non-blind de-blurring algorithm.

In this paper, we propose a spatially varying motion blur detection and restoration approach. Motion blur regions are automatically detected from a single image without requiring any additional information. Blur kernel is estimated, and a kernel septic feature is used to recognize blur regions. Blur regions refinement is performed with the support of Contour-guided Color Palettes (CCP) image segmentation method and neighboring information. Blur kernel for the detected blurred regions is used to restore a clear image with considerable visual quality. The framework of the proposed blur estimation and de-blurring approach is presented in Fig. 1.

III. PROPOSED WORK

A. Kernel Initialization

Motion blur is modeled as a straight line as follows:

$$k(x,y) = \begin{cases} \frac{1}{len} & if\ \sqrt{x^2+y^2} \le len\ ,\ tan\theta = \frac{y}{x} \\ 0 & otherwise \end{cases} \quad (2)$$

Where k is the blur kernel, *x, y* are the horizontal and vertical coordinates, *Len* is the kernel length and θ is the kernel angle.

Motion blur kernels affect blurred images in four directions: vertical, horizontal, positive diagonal and negative diagonal. However, both the vertical and horizontal motion blur kernels are equivalent to model motion blur in the used dataset [19]. Moreover, our experimental results verify that the effect of positive and negative diagonal motion blur kernels are equivalent to model motion blur in the used dataset. Therefore, we consider motion blur kernels in both directions: vertical and positive diagonal.

At any pixel l, blur is modeled as follows:

$$y[l] = (k_{in} * x)[l] + n[l] \quad (3)$$

Where $y[l]$ represents the blurred image, $x[l]$ is the corresponding clear image and $n[l]$ characterizes the noise at every pixel *l*. Noise is supposed to be a white Gaussian noise $N(0,\sigma_x^2)$, and k_{in} represents the blur kernel at each pixel location l [17]. For the sake of blur kernels k_{in} estimation we use one hypothesis that kernel is constant in any local neighborhood.

If $y[l]$ is blurred with an initial blur kernel k_{in}, then the gradient filtered $y_i[l]$ can be expressed as

$$y_i[l] = \big(x * (k_{in} * f_i)\big)[l] \quad ,i = 1{:}j \quad (4)$$

and $$f_i[l] = w[l] \times e^{(-j\langle w_i, l\rangle)}, i = 1{:}j \quad (5)$$

where $f_i[l]$ is the set of local orthogonal filters, $w[l]$ is a window function, w_i are the frequencies, and j is the number of filters [19]. The motion blur kernel k_{in} is one of a discrete set of possible candidates, corresponding to the vertical or the positive diagonal box filters.

$$k_{in} \in \{k_{vin}, k_{din}\} \quad (6)$$

$$y_i^v[l] = (f_{iv} * \nabla_v * y)[l] \quad (7)$$

$$y_i^d[l] = (f_{id} * \nabla_d * y)[l] \quad (8)$$

where ∇_v and ∇_d are the vertical and positive diagonal gradient filters, and $\{\ f_{iv}\}$, $\{f_{id}\ \}$correspond to the sub-band decompositions based on vertical and positive diagonal 1-D windows. We need to analyze the blurry image to detect movement diagonally or vertically. The likelihood of the local window which is blurred by either vertical or positive diagonal kernels is estimated by:

$$p_m(k_{vin}) = \log p(\ y_i^v[l] | k_{vin}) \quad (9)$$

$$p_m(k_{din}) = \log p(\ y_i^d[l] | k_{din}) \quad (10)$$

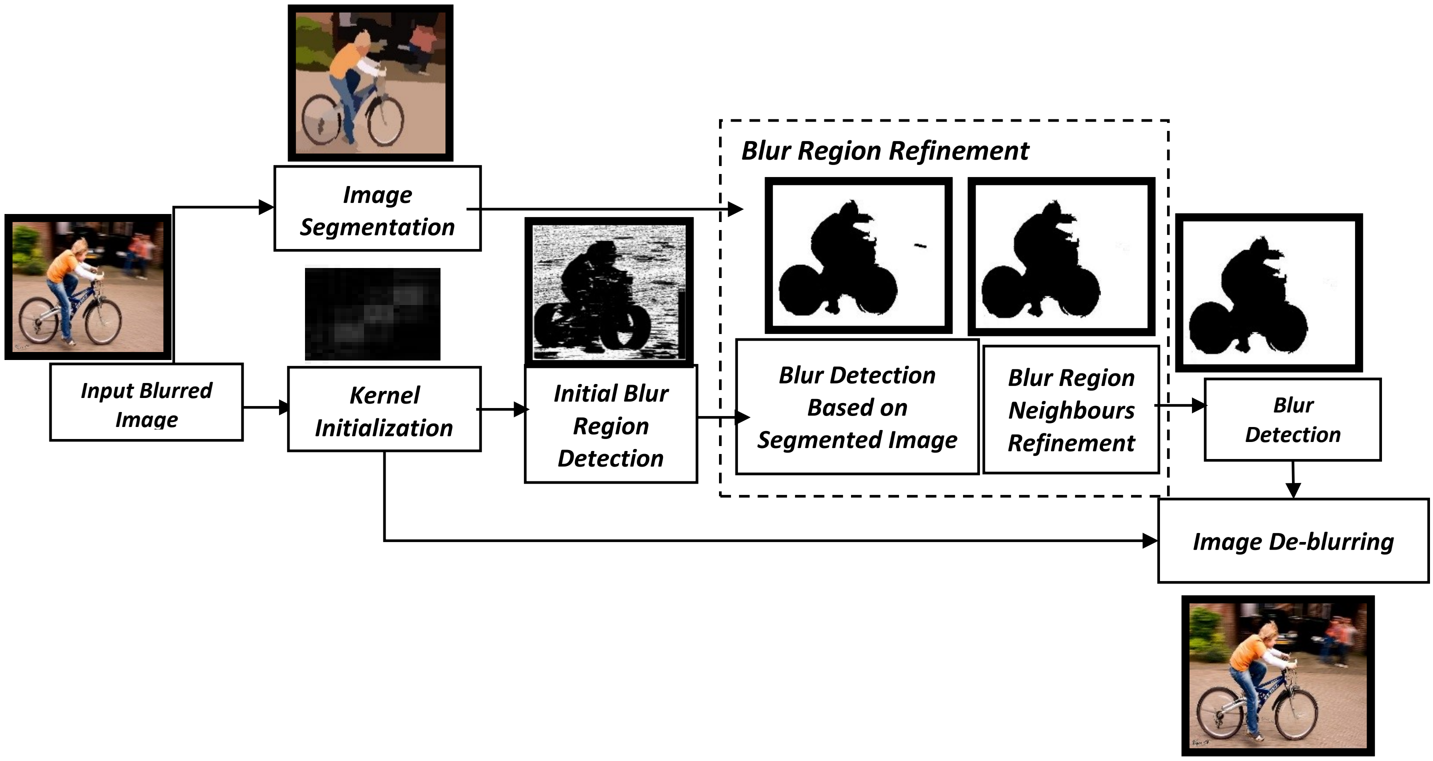

Fig. 1. The framework of the proposed blur estimation and de-blurring approach

The initial blur kernel $k_{in}$is selected from the candidate pool given in (7) such that likelihood in is maximized as follows:

$$theta = maximize\ (p_m\ (k_{vl}, k_{dl}\ )) \quad (11)$$

Theta is the optimal motion direction (0 or 45) in degrees. For optimal blur kernel lengthk_{in}, we discretize the range of motion length into samples from *Len*= 3 to 50 with a step size of 4.

B. Initial Blur Region Detection

After the initial kernel estimation, a kernel specific feature [19] is utilized to initially detect the blurred regions. The feature offers information required to train the Support Vector Machine (SVM) classifier [21, 22] to distinguish the blurred region from non-blurred ones. The kernel-specific feature vector V_i is the multiplication of the variance σ_{yi}^2 of filtered gradients y_i and the variance σ_{ki}^2 of the initial blur kernel k_{in} filtered by the filter f_i [19].

$$\sigma_{yi}^2 = {}^1\!/_W \sum_l |y_i[l]|^2 \quad (12)$$

where W is the window length.

$$\sigma_{ki}^2 = E(|(k_{in} \otimes f_i)[l]|^2) \quad (13)$$

$$V_i = \sigma_{ki}^2 \sigma_{yi}^2\ , \ i = 1,2, \dots \dots j \quad (14)$$

In a small region, the feature vector V_k is a j-dimensional vector that can be expressed as follows [19]:

$$V_k = [V_1, V_2, V_3, \dots \dots \dots, V_j] \quad (15)$$

Classification of the blurred regions by using the LIBSVM [22]. The kernel-specific feature vectors V_k are extracted by training using a set of clear image patches and blurred image patches by kernel K [19]. After training, a supervised nonlinear SVM classifier is learned.

$$S_k(V_k) = \sum_{u=1}^{p} \alpha_u y_u\, K(s_u, V_k) + B \quad (16)$$

where α_u are Lagrange multipliers which represent the influence of the classifier constraints, p is the number of the support vectors, y_u are the binary class labels, s_u are the training samples and B is the bias. $K(s_u, V_k)$ is SVM kernel the Radial Basis Function (BRF), which maps the data to higher dimensions to make the linear SVM works in nonlinear settings. It can be represented as follows:

$$K(s_u, V_k) = exp[-\gamma \|s_u - V_k\|^2] \quad (17)$$

where γ is the kernel bandwidth.

The optimal motion kernel is given as follows:

$$k_{optimal} = arg\left(\max_{k\epsilon K} S_k(V_k)\right) \quad (18)$$

The SVM classifier decision rule can be expressed as:

$$D(v_k) = sgn\left(\sum_{u=1}^{p} \alpha_u y_u\, K(s_u, V_k) + B\right) \quad (19)$$

Blur detection is performed by checking the sign of the value of the SVM classifier decision rule [21]. Each pixel is characterized as blurred or non-blurred based on the SVM result. Pixel-based blur detection typically yields coarse results. Therefore, SVM classifier results are preceded by a blur region refinement approach to enhance blur detection accuracy.

C. Blur Region Refinement

Initially detected blur regions must be refined using Contour Guided Color Palette (CCP) image segmentation method and neighboring information as described in the following subsections.

1) Blur Detection Based on Segmented Image

In CCP image segmentation method, image contour and color cues are integrated to provide image color palettes [23].
CCP method introduces color samples along lengthy contours between the input image regions by using the Mean-Shift (MS) algorithm [24]. This color palette considered to be an initial segmentation which will be improved by successful post-processing techniques as contours leakage avoidance, fake boundary removal, and the emergence of the small regions [23].
The result segmented image is integrated with the pixel-based initial blur detection results to enhance the blur detection accuracy. The ratio of the total blurred pixels in each image segment to the total number of pixels in that segment is calculated. If the ratio exceeds a predefined threshold, the segment is considered as blurred; otherwise, it will be a non-blurred. All pixel values in the blurred segment are set to 1; otherwise, it is set to 0. The blur detection algorithm based on the segmented image is shown in Algorithm 1.

Algorithm 1. Blur Detection & Segmentation Algorithm.

Input: Blurred image y, Initial blur detection I, blur detection threshold ϵ
Output: Blur regions detection based on the segmented image (B_s).
$B_s = CCP(y)$ %Contour guided color segmented image
For j=1: n % n: number of segments
sumw=0 % number of blurred points in a segment
sumb=0 % number of non-blurred points in a segment
For i=1:m % m: the total number of points in a segment
IF (I (j, i) ==0)
sumb=sumb+1
Else
sumw=sumw+1
End IF
End For
Blurred_ratio=(sumw/m) *100
IF (Blurred_ratio)>= ϵ
B_s(j)=1
Else
B_s(j)=0
End IF
End For

2) Blur Region Neighbours Refinement

The blur detection based on the segmented image may have some inaccurate classified regions. Consequently, we use neighbor's information to enhance the blur detection accuracy. For each segmented region, the neighboring regions are set Otherwise, the classification of this region remains unchanged.

Algorithm 2. Blur Region Neighbour Refinement Algorithm.

Input: Blurred image y, Blur detection based on the egmented image B_s
Output: Blur region neighbour refinement(D)
$[S_l] = CP(y)$ % S_l: segment image labels map.
strel_obj = strel ('disk', 2) % disk-shaped structuring elements
n = max(S_l) %n: number of segments
For L = 1: n
cur_reg = (S_l == L) % current segment
neighb_reg = imdilate (cur_reg, strel_obj) - cur_reg
% neighbor segments
neighb_index= S_l(neighb_reg==1)
N=size(neighb_index) %number of neighbours
IF $\left(\sum_{k=1}^{N} B_s(k)/N\right) == 0$ & (B_s(L) ==1)
D(L) =0
Else IF $\left(\sum_{k=1}^{N} B_s(k)/N\right) == 1$ & (B_s(L) == 0)
D(L) =1
End IF
End For

D. Image Deblurring

The deconvolution of the whole image using the same kernel often lead to unpleasant artifacts in the non-blurred regions. Therefore, our approach restores partially motion blurred images by deblurring only segmented blur regions using the non-blind deconvolution Lucy–Richardson algorithm [2, 3].

1) Blur Region Segmentation and Deblurring

Motion blur kernels parameters (direction and length) are estimated automatically from the input blurred image as explained in section A. Subsequently, the blurred region is segmented then de-blurred using Lucy–Richardson algorithm. The non-blurred region is combined with the de-blurred region to obtain the restored image as presented in Fig 2.

Fig. 2. (a) blurred image (b) De-blurring result (c) Segment un-blurred region from Input image (d) Segmented un-blurred region.

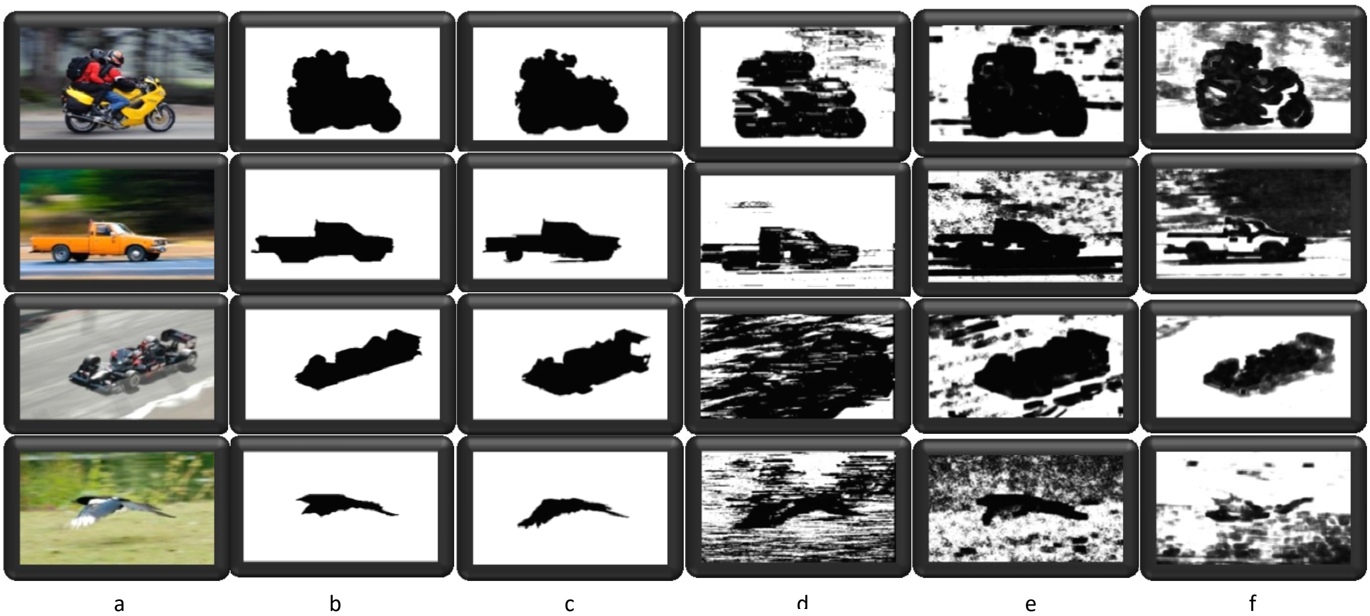

Fig. 3. (a) Motion blurred images (b) Ground truth. (c) Proposed method (d) Pang et al. (e) Shi et al. (f) Liu et al.

2) Border Refinement

The direct combination of the deconvolution of the blurred region and the non-blurred will lead to unpleasant artifacts in the intersection between de-blurred and non-blurred regions. We handle this problem by extending the detected blurred region border to smooth out border boundaries and reduce the resultant border artifacts.

IV. Experimental Results

The proposed algorithm was implemented in MATLAB R2017a on an Intel Pentium i7-2670QM CPU 2.20GHz, with 8GB of RAM. In this section, we compare the proposed approach with the state of the art methods reported by Liu et al. [16], Shi et al. [18], and Pang et al. [19].

A. Dataset and parameter analysis

The proposed approach is evaluated on the public dataset accessible at [25]. The dataset has motion blurred images with the equivalent ground-truth blur labels. Datasets I contain 296 motion blurred images. The nonlinear SVM classifier is trained with a subcategory of the datasets [19, 26] differ from the testing dataset.

Blur detection threshold (ϵ) utilized in algorithm 1 has been estimated experimentally. For accuracy evaluation, the results accomplished from the proposed approach are compared to ground truth to detect how many pixels are precisely classified. Accuracy is evaluated using the following equation:

$$Accuracy = \frac{\{i|O_i = G_i\}}{M * N} \quad (20)$$

Where M, N represents image height and width, respectively, O_i represents that current estimate blur or non-blur and G_i is the ground-truth [27].

B. Comparative Results

The performance of the proposed method is compared to the most effective methods stated in the literature. Table 1 shows the accuracy of the Naïve Bayes (DBDF) [18], SVM classification (MVV) [19], MVV+GrabCut [19], by Liu et al. [16], and the proposed approach. Table 1 shows the progress of our proposed approach over other comparative methods. For the motion blurred images, the proposed method accuracy is improved compared to other methods. The combination of blur kernel estimation and blur region segmentation may lead to some missing or extra details. Fig. 3 shows that the proposed approach is robust for detecting motion blur, especially for single object images. Images with a single object have an accuracy of approximately 98%. Fig. 4 shows the de-blurring results obtained using the proposed approach.

Table 1. Accuracy comparison of DBDF, MVV, MVV+ GrabCut, NBNN, and the proposed method.

Method	Blur detection accuracy%
DBDF [18]	41.2
MVV [19]	49.8
MVV+ GrabCut [19]	56.5
Liu et al. [16]	62.0237
Proposed approach	71

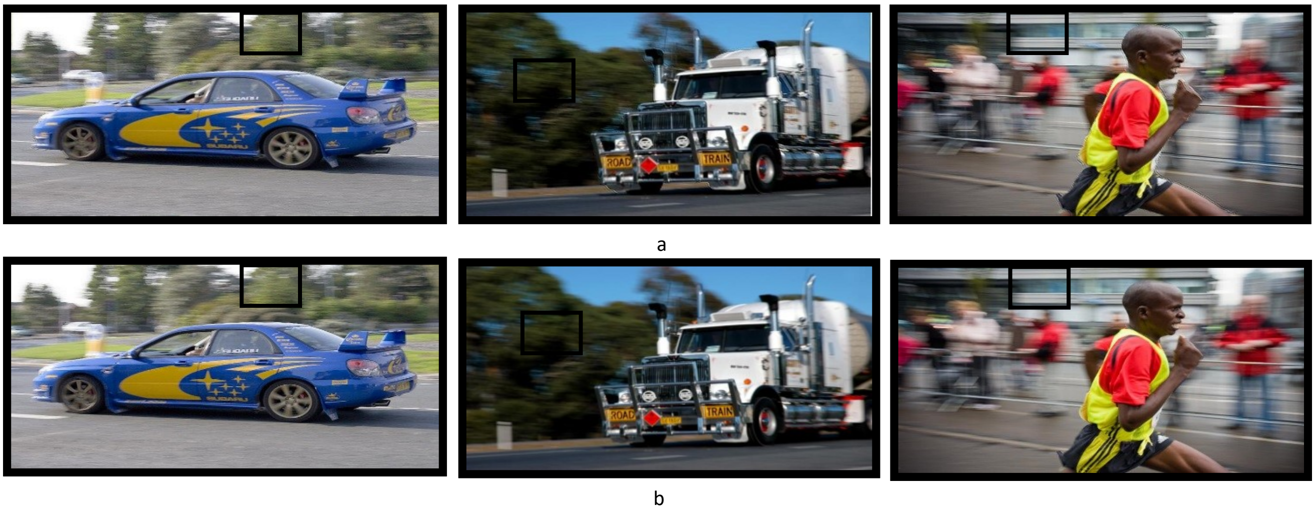

Fig. 4. (a) Deblurring result (b) Original motion-blurred image

V. CONCLUSION

In this paper, a robust spatially-variant blur detection and restoration method is proposed. This paper has five main contributions. First, the optimal motion blur kernel length and direction is estimated by maximizing the Likelihood of blurred local window by vertical and positive diagonal kernels. Second, the initial blur detection is implemented using a kernel specific feature. Third, the initial blur detection is refined with the support of image segmentation (CCP) method. Fourth, the detected blur regions are refined using neighbor information. Finally, the blurred regions are de-blurred using the computed kernel and the standard Lucy–Richardson algorithm. The performance of the proposed approach is verified using many images from the dataset accessible at [25]. Experimental results validate the efficiency of the proposed approach in detecting blur for motion blurred images. The proposed approach outperformed state-of-the-art methods such as DBDF, and MVV+GrabCut. In the future, the proposed approach can be extended to handle defocus blurred images.

REFERENCES

[1] B. K. Gunturk and X. Li, Image Restoration Fundamentals and Advances. 2013.

[2] W. Richardson, “Bayesian-based iterative method of image restoration,” Journal of the Optical Society of America, vol. 62, no. 1, pp.55–59, 1972.

[3] L. Lucy, “An iterative technique for the rectification of observed distributions,” Astronomical Journal, vol. 79, no. 6, pp. 745–75, 1974.

[4] Wiener and Norbert, Extrapolation, Interpolation, and Smoothing of Stationary Time Series, New York: Wiley, 1949.

[5] Q. Shan, J. Jia, and A. Agarwala, “High-quality motion deblurring from a single image,” ACM Transactions on Graphics, vol. 27, no. 3, pp. 73:1–73:10, August 2008.

[6] S. Cho and S. Lee, “Fast motion deblurring,” ACM Transactions on Graphics, vol. 28, no. 5, pp. 145:1–145:8, December 2009.

[7] L. Xu and J. Jia, “Two-phase kernel estimation for robust motion deblurring,” in Proceedings of the European Conference on Computer Vision, vol. 1, pp. 157–170, 2010.

[8] R. Fergus, B. Singh, A. Hertzmann, S. T. Roweis, and W. T. Freeman, “Removing camera shake from a single photograph,” ACM SIGGRAPH 2006 Papers on - SIGGRAPH ’06, p. 787, 2006.

[9] A. Levin, R. Fergus, F. Durand, and W. T. Freeman, “Image and depth from a conventional camera with a coded aperture,” ACM Transactions on Graphics, vol. 26, no. 3, p. 70, 2007.

[10] D. Krishnan, T. Tay, and R. Fergus, “Blind deconvolution using a normalized sparsity measure,” Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 233–240, 2011.

[11] Q. Shan, J. Jia, and A. Agarwala, “High-quality motion deblurring from a single image,” ACM Transactions on Graphics, vol. 27, no. 3, p. 1, 2008.

[12] Goldstein, A., Fattal, R.: ‘Blur-kernel estimation from spectral irregularities’Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), 7576 LNCS, (Part 5), pp. 622–635, 2012.

[13] Y. P. Guo, H. P. Lee, and C. L. Teo, “Blind restoration of images degraded by space-variant blurs using iterative algorithms for both blur identification and image restoration,” Image and Vision Computing, vol. 15, no. 5, pp. 399–410, 1997.

[14] F. Sroubek and J. Flusser, “Multichannel blind deconvolution of spatially misaligned images,” IEEE Transactions on Image Processing, vol. 14, pp. 874–883, July 2005.

[15] A. R. S. Peleg, “Restoration of Multiple Images with Motion Blur in Different Directions,” 2000.

[16] R. Liu, Z. Li, and J. Jia, “Image partial blur detection and classification,” 26th IEEE Conference on Computer Vision and Pattern Recognition, CVPR, 2008.

[17] A. Chakrabarti, T. Zickler, and W. T. Freeman, “Analyzing spatially-varying blur,” Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 2512–2519, 2010.

[18] J. Shi, L. Xu, and J. Jia, “Discriminative blur detection features,” Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 2965–2972, 2014.

[19] Y. Pang, H. Zhu, X. Li, and X. Li, “Classifying Discriminative Features for Blur Detection,” IEEE Transactions on Cybernetics, vol. 46, no. 10, pp. 2220–2227, 2016.

[20] D. Yang and S. Qin, “Restoration of the degraded image with partial blurred regions based on blur detection and classification,” in 2015 IEEE International Conference on Mechatronics and Automation (ICMA), pp. 2414–2419, 2015.

[21] Burges, C.J.C.: ‘A Tutorial on Support Vector Machines for Pattern Recognition’, pp. 1–43,1997.

[22] Chang, C., Lin, C.: 'LIBSVM: A Library for Support Vector Machines', pp. 1–39, 2013.

[23] Fu, X., Wang, C., Chen, C., Wang, C., Kuo, C.J.: 'Robust Image Segmentation Using Contour-guided Color Palettes', In ICCV, 2015.

[24] Comaniciu, D., Meer, P.: 'Mean shift: A robust approach toward feature space analysis'*IEEE Trans. Pattern Anal. Mach. Intell.*, 24, (5), pp. 603–619, 2002.

[25] Blur Detection Dataset. [Online]. Available: http://www.cse.cuhk.edu.hk/∼leojia/projects/dblurdetect/dataset.html, accessed Jun. 23, 2015.

[26] Source Codes of Classifying Discriminative Features for Blur Detection. [Online]. Available: http://yanweipang.com/papers, accessed Jun. 23, 2015.

[27] Wang, W., Zheng, J. jin, Zhou, H. jun: 'Segmenting, removing and ranking partial blur'*Signal, Image Video Process.*, 8, (4), pp. 647–655, 2014.