IET Image Processing

Research Article

IET Journals
The Institution of Engineering and Technology

Blur parameter locus curve and its applications

ISSN 1751-9659
Received on 13th May 2019
Revised 14th September 2019
Accepted on 14th October 2019
E-First on 19th December 2019
doi: 10.1049/iet-ipr.2019.0577
www.ietdl.org

Himanshu Kumar[1] ✉, *Sumana Gupta*[1], *Venkatesh K. Subramanian*[1]

[1]*Department of Electrical Engineering, IIT Kanpur, Kanpur 208016, UP, India*

✉ *E-mail: kumarh@iitk.ac.in*

Abstract: Conventionally, the point spread function (PSF) is understood as a characteristic function of any optical system. It captures the information about the amount of blur present along all the directions for a point in the scene. However, the dependence of blur on the PSF is in the form of convolution for any object other than a point source present in the scene and hence their relationship is less explicit. The authors propose a blur parameter locus curve (BPLC) as a system representation which has a one to one relationship with blur. BPLC simply is a chart of blur amounts in all directions of a given PSF with respect to the selected measurement function. They further characterise the PSF by decomposing the variation of BPLC across all directions based on the study performed for different possible forms of the blur kernels. Such decomposition provides powerful tools for various analysis. As PSF can be anisotropic, the computation of BPLC becomes an essential intermediate step to obtain the scale map as at the same scale, blur is different in different directions. Furthermore, they demonstrate the use of BPLC to obtain other system characteristics function such as PSF.

1 Introduction

We begin by making precise definition of certain terms that are the basis of our development. All subsequent use of the respective terms will be with these definitions in mind.

PSF: Point spread function (PSF) in the imaging plane of a three-dimensional (3D) scene point. In the image plane, it is defined as a 2D function for each point in the image plane that is defined by the location of the corresponding source point in scene as well as the properties of the imaging system. Thus, the PSF relates the 3D scene to the image plane, by associating with each point source location in 3D space, a certain image which we shall call the respective PSF for that particular point source location. The scene dimensions, therefore, manifest as parameters of such a PSF.

Blur: We define 'Blur' as the spread of an object image in the image plane due to the underlying PSF of the imaging system. So, if the blur kernel is a Kronecker delta function, there would be no blur corresponding to that point/object in the scene.

Blur amount: Blur amount is a measurement of the blur present. Blur can be measured in terms of blur inducing phenomena such as the radius of spread, variance etc. It is a user-defined quantity, e.g. for Gaussian PSF $g(x, y; \sigma)$ blur amount could be σ, or alternatively, blur radius R often defined as $\sqrt{2}\sigma$.

Blur parameters: Parameters of the underlying PSF that control the blur amounts are called 'Blur Parameters', e.g. σ for defocus-induced Gaussian, blur length L for motion blur, radial distance r for rolling blur.

Scale: Scale refers to physical quantities of the imaging system-scene combination that result in the underlying PSF by determining the respective blur parameters. The physical quantities may affect PSF either directly or indirectly as a combination of two factors: for example, rotation blur depends upon the distance from the rotating axis of the point source, in addition to depth, the PSF of diffraction is dependent on aperture size and wavelength (but not on the depth). We refer to all these quantities as 'scale' of corresponding PSFs.

Scale parameters: Parameters such as depth that relates scale to corresponding blur parameters of underlying PSF are referred to as 'Scale Parameters'.

Scale map: Scale parameter for each pixel for a given image is referred to as scale map. Thus, with each pixel we associate the scale parameters of the corresponding source point(s). In many situations, the point spreading will result in a particular pixel in the image being associated with multiple scene points, and when this is the case, the scale map will associate with that pixel the scale parameters of the most dominant scene point involved.

Blur kernel: Blur kernel is obtained by fixing the point source location variables X, Y, Z, v to specific values. The blur kernel therefore reduces, under the particular constraints, to a function of just (x, y). Its form is thus frozen for a given imaging system and scene/object. For convenience, we choose a PSF with unity blur parameters as the blur kernel, e.g. for Gaussian PSF $g(x, y; \sigma)$, the corresponding blur kernel is defined as $g(x, y; 1)$. It is a generic representation of the PSF.

The PSF is a characteristic function of any optical/imaging system. Essentially, it is an energy redistribution of a point source of the scene in the image plane. The degree and the form of energy redistribution depend on the scale and the nature of the PSF. The fundamental notion of nature of PSF is captured in the blur kernel which takes on different canonical forms for different engendering physical phenomena: we have a defocus blur kernel that serves to represent all defocus caused blur phenomena, a different 'linear motion' blur kernel for all instances of linear motion blur etc. Thus, blur amount at any image point, is a function that is dependent upon both the nature of the blur kernel as well as blur parameters. The blur kernel for a given source remains invariant of scale – however, the blur parameters vary with the scale and they determine the precise final realisation of the PSF.

The PSF can be utilised for many applications such as deblurring, system characterisation, scale map estimation etc. Scale map estimation is an important field of research. The variations of scale are captured in the scale map and can be used to obtain the underlying information of the corresponding physical quantities. Thus, the scale map can be evaluated by estimating the corresponding PSF at each pixel location.

The PSF is usually defined over $N \times N$ points for each image-point in the image plane. However, we encounter edges more frequently than the isolated points in the image. The blurring at the edges is in the form of convolution. Thus, the relation of blurring to the PSF is not direct. Hence, blurring of an edge is the superposition of all the individual points constituting the edge,

often referred to as edge spread and the corresponding blurring function is referred to as the edge spread function (ESF).

The recovery of PSF requires $N \times N$ independent observations. This results in an ill-posed problem due to the presence of noise which is often solved by employing various priors such as [1–4]. However, blur amount is a scalar quantity and can be estimated accurately at edge points. Kumar *et al*. [5] introduced the concept of blur measurement function. In this paper, we further extend this concept to characterise and decompose the PSF using variation of blur amount across the edge orientation unlike state-of-the-art methods which evaluate it at $N \times N$ points. We refer to this as the blur parameter locus curve (BPLC). We utilise BPLC for calibrating the scale map for an anisotropic PSF (PSF after unequal vertical and horizontal image resizing, asymmetrical (astigmatic) PSF of optical/imaging systems) which induces different amounts of blur at different directions at a given scale. We also develop the properties/constraints for PSFs based on BPLC. We use these constraints to discard the erroneous blur values due to shadows and false depth edges. The main contributions of the paper are as follows:

- We present a novel classification of blur kernels that can be used to analytically calculate the blur amount along any given direction θ for a given PSF.
- We introduce the concept of the BPLC and address the ambiguity of different blur amounts at different directions for a given scale. This addresses the fundamental limitation of scale map estimation application for the case of anisotropic PSF, which is usually not considered in the state-of-the-art methods.
- We also demonstrate the effects of image resizing on the blur and derive closed-form analytical expression for it using BPLC. We apply these constraints to remove erroneous blur values due to shadow and false edge to obtain more accurate (Reliable) defocus map.
- We also present a novel methodology and analysis tools based on BPLC for PSF characterisation from a single image.

The paper is organised as follows: Section 2 discusses related works. Section 3 presents the classification of the blur kernels and PSF. Section 4 introduces BPLC along with the analytical relationship between image resizing and the blur kernel. We present the applications of the proposed BPLC theory in Section 5 and conclude the paper in Section 6.

2 Previous works

Richardson [6] estimated the blur PSF using an iterative approach in a Bayesian framework. The recent state-of-the-art methods [1–4] also estimate the PSF using an iterative procedure for the underlying blurring optimisation problem along with different priors like sparsity and constraint blur type. However, priors used in these state-of-the-art methods may not hold, and the accuracy of the estimated PSF is sensitive to window size. Even in case of the accurate PSF estimation, parameterisation of estimated PSF and relating it to the scale is required to address the scale ambiguity. Hence, scale map estimation using spatially varying PSF extraction is not a reliable and practical approach for general applications. Similarly, state-of-the-art methods [7–9] employ the Gaussian nature of defocus blur kernel for estimating the blur amount. These state-of-the-art methods of scale map estimation for defocus blur are quite fast and accurate. However, these methods are valid only when the PSF is isotropic in nature.

Trentacoste *et al.* [10] chose the Gaussian blur PSF and studied the effect of downsampling on the PSF. They derived the relationship between downsampling and image blur PSF in a single dimension and generalised it for higher dimensions. They created perceptually similar images after downsampling by matching the blur. However, they did not discuss the effect of downsampling on oriented edges. Also, downsampling in random directions and closed-form relationship for non-identical downsampling along orthogonal directions have not been discussed in their work. In the patent of Hong [11], downsampling is used to reduce the amount of the defocus blur. The patent states that estimation of large blur parameters is inaccurate. Hence, they claim that the use of downsampling reduces the blur amount and improves the estimation accuracy. However, no analytical justification or relations are provided.

Qin *et al.* [12] use downsampling and upsampling blocks to calculate the error between the high-resolution and low-resolution images. Using this error, the high-resolution image is estimated. They report that defocus plays an important role in estimating high-resolution images. However, the effect of downsampling on defocus blur has not been discussed. Pang *et al.* [13, 14] used directional filter responses for distinguishing between motion blur and defocus blur. These methods highlight the importance of the use of PSF characteristics. However, for non-isotropic PSFs, which commonly result from asymmetrical resizing of images or in images captured through an asymmetrical optical/imaging system, this would not be effective.

We observe that the state-of-the-art methods does not relate blur parameter of PSF with scale while estimating scale map. Also, these methods do not address scale ambiguity for anisotropic PSF. Hence, these methods cannot be applied when PSF is not isotropic. We introduce the concept of the BPLC to address this and provide fundamental tools for PSF related analysis.

Yang and Qin [15] estimated the underlying blur kernel from a single partially blurred image by classifying the blurred regions. The sharp regions corresponding to blurred regions are restored by utilising the estimated underlying blur PSF. Lu *et al.* [16] presented an unsupervised method for domain-specific single-image deblurring based on disentangled representations by splitting the content and blur features in a blurred image using content encoders and blur encoders. Such methods employ specific information for recovery of underlying PSF and sharp image, however, such information may not be available.

Xu *et al.* [17] proposed a deep convolutional neural network trained after suppressing extraneous details and enhancing sharp edges for extracting sharp edges from blurred images. Nimisha *et al.* [18] proposed deep neural networks-based approach for end-to-end motion deblurring by capturing the data prior for generating and to discriminating between clean and blurred features. Nah *et al.* [19] proposed a multi-scale convolutional neural network-based end-to-end deblurring scheme that restores sharp images in an end-to-end manner where blur is caused by various sources. Madam-Nimisha *et al.* [20] proposed a Generative Adversarial Network-based approach for an end-to-end deblurring network. Wang *et al.* [21] estimated blur kernel in a transform domain, whereas the deconvolution model is decoupled into deblurring and denoising stages via variable splitting. The method utilises benefits of both model optimisation and deep learning. These methods utilise learning-based techniques to restore the sharp image from a blurred image, however, performance of such methods is dependent upon training images, type of blur PSF etc.

Kotera and Šroubek [22] utilised sparsity priors to recover a sharp image from a single blurred image by maximising the aposteriori information. Lai *et al.* [23] utilise the normalised colour-line prior to estimate the single-image blur kernel. Bai *et al.* [24] proposed a graph-based blind image deblurring algorithm by interpreting an image patch as a signal on a weighted graph. The method uses a reweighted graph total variation prior that can efficiently promote a bi-modal edge weight distribution given a blurry patch. Liu *et al.* [25] used image prior for image deblurring based on a Super-Gaussian field model with adaptive structures. Chen *et al.* [26] used local maximum gradient prior for recovering the sharp image from the blurred image. Aljadaany *et al.* [27] proposed an end-to-end deblurring deep networks system Dr-Net based on the data fidelity and the image prior. Bai *et al.* [28] used coarse-to-fine priors to estimate the underlying PSF and to restore the deblurred image. These methods recover sharp image from the blurred image by utilising various priors which may not be true in general.

Sun *et al.* [29] estimated the underlying PSF by imposing a patch prior specifically tailored towards modelling the appearance of image edge and corner primitives. Vasu and Rajagopalan [30] investigated the relation between the edge profiles present in a motion-blurred image and the underlying camera motion

responsible for causing the motion blur. Yang and Ji [31] presented an interpretation of edge selection/reweighting in terms of variational Bayes inference, and therefore developed a novel variational expectation maximisation algorithm with built-in adaptive edge selection for blind deblurring. These methods demonstrate the importance of edge blur for determining/characterising the underlying PSF. We utilise this concept and propose BPLC for characterising the PSF. Similarly, Chakrabarti *et al.* [32] modelled gradients in an image using an infinite mixture of zero-mean Gaussian distributions to segment out blurred region and Lin *et al.* [33] proposed the use of a separable coded aperture to estimate depth from a single defocused image. This highlights the importance of decomposing the PSF into separable form which we have introduced in this paper by utilising the BPLC.

Hong and Park [34] recovered a sharp image from a single motion-blurred image by estimating the underlying PSF using anisotropic regularisation by an iterative process. Deshpande and Patnaik [35] presented a uniform motion blur PSF estimation by estimating the underlying parameters of the PSF using radon transform. Similarly, Shi and Liu [36] proposed PSF estimation algorithm based on gradient cepstrum analysis and Mosleh *et al.* [37] estimated PSF using frequency spectrum. Dong and Ma [38] proposed a novel variational method for single image blind deblurring based on the fractional-order differential to overcome the staircase effect produced by the total variation regularisation. Pan *et al.* [39] used auto-correlation of the absolute phase-only image in frequency domain to obtain underlying motion blur PSF from single blurred image. These methods utilise specific features of known blur type to estimate the underlying PSF and restore the sharp image, thus these methods cannot be used for generalised cases.

We observe from the state-of-the-art methods PSF estimation for underlying blur is an ill-posed problem and require some prior knowledge. We also observe that edges are important attributes for blur PSF estimation. We propose the analysis and characterisation of PSF based on the amount of blur at the edges using the introduced concept of BPLC.

3 Blur kernel classification

We first discuss some properties related to PSFs and blur kernels. We use these properties to classify the blur kernels. We later use these properties to develop useful mathematical tools for analysis.

3.1 Isotropy

A blur kernel is said to be isotropic if its spread amount (degree of energy distribution) is circular in shape irrespective of the possible variation of magnitude within the support so long as the variation is not a function of θ. Equation (1) shows an example of isotropic PSF $f_1(x, y)$ which has identical blur parameter σ for all directions. Whereas (2) is an example of anisotropic PSF $f_2(x, y)$ having blur parameters σ_1 and σ_2 along X and Y directions which induces direction-dependent blur.

$$f_1(x, y) = \frac{1}{2\pi\sigma^2} \mathrm{e}^{-((x^2+y^2)/2\sigma^2)} \tag{1}$$

$$f_2(x, y) = \frac{1}{2\pi\sigma_1\sigma_2} \mathrm{e}^{-((x^2/2\sigma_1^2)+(y^2/2\sigma_2^2))} \tag{2}$$

The degree of energy distribution is usually measured in terms of the blur parameter. However, the blur parameter itself has different definitions for different blur kernels, e.g. spread length for motion blur, the standard deviation for defocusing. Any suitable definition can be taken for calculating the blur parameter for PSF. Thus, the isotropy of a PSF is with respect to the chosen definition of blur parameter, i.e. the PSF may be isotropic for one blur parameter (e.g. variance) and may not be isotropic with respect to the other (e.g. spread length).

3.2 Degeneracy and closed-form

We define properties such as degeneracy and closed-form to efficiently characterise the blur computation and representation of the blur kernels.

Definition 1: (Degeneracy): Any given blur kernel $\hat{f}_\theta(r)$ that has energy distribution only along any one single direction θ is called a degenerate blur kernel and direction θ is referred to as the principal direction.

A typical example of such a blur kernel is the (linear) motion blur kernel. For the linear motion blur kernel, all the blurring energy is oriented in the motion direction. As an example, for uniform linear motion, $\hat{f}_\theta(r)$ will be a rectangular pulse in r, with the pulse width proportional to the velocity.

Definition 2: (Closure): Any given blur kernel that can be expressed as convolution of two or more degenerate blur kernels is called a closed-form blur kernel and corresponding directions of the component degenerate blur kernels are referred as the principal directions.

The ideal defocus blur kernel is usually expressed as a symmetrical Gaussian kernel. This kernel can be decomposed into the convolution of two degenerate blur kernels in any two perpendicular directions. Hence, the defocus blur kernel is one example of a closed-form blur kernel, and, in this case, the principal directions are not unique. For any general optical system, the blur kernel can be arbitrary shaped with directional dependent blur parameter σ.

3.3 Completeness and blur kernel model

The blur kernels can be different for various sources of blurs, e.g. uniform for motion blur, Gaussian for defocus blur etc. Thus, the blur parameter has a different definition for different blur kernels. Measurement of blur is performed on the basis of assumed nature of underlying blur PSF, e.g. [7–9]. Thus, measurement function may not be same as underlying blur kernel of the PSF. However, underlying blur PSF can be modelled and analysed using measurement function if the blur parameters of two have one to one correspondence. We refer to this property as *Completeness* of blur kernel with respect to measurement function as in Theorem 1.

Theorem 1: (Completeness): If parameter η for PSF $f(x;\eta)$ has a linear relation with spread d as well as equivalent blur parameter σ, then the representation of $f(x;\eta)$ with respect to the measurement function is complete.

Proof: The spread d varies linearly with η for PSF $f(x;\eta)$ and equivalent parameter σ has linear relationship with η

$$\eta = k_1 d; \quad \sigma = k_2 \eta \tag{3}$$

From (3)

$$\sigma = kd$$

Under scale and space transformations $\mathfrak{f}$ of blur kernel, the corresponding spread d has a one-to-one correspondence to equivalent blur parameter σ.

$$\mathfrak{f}(d)k = \mathfrak{f}(\sigma)$$

Hence, the equivalent representation of $f(x;\eta)$ with respect to the measurement function is complete. □

Using Theorem 1, the underlying blur kernel which is complete with respect to measurement function can be modelled and analysed with the help of measurement function. We choose a measurement kernel for blur measurement as a Gaussian kernel and have derived the results accordingly from now-on. These results can be modified easily for other kinds of measurement functions.

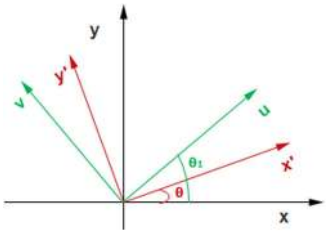

Fig. 1 *Reference axis*

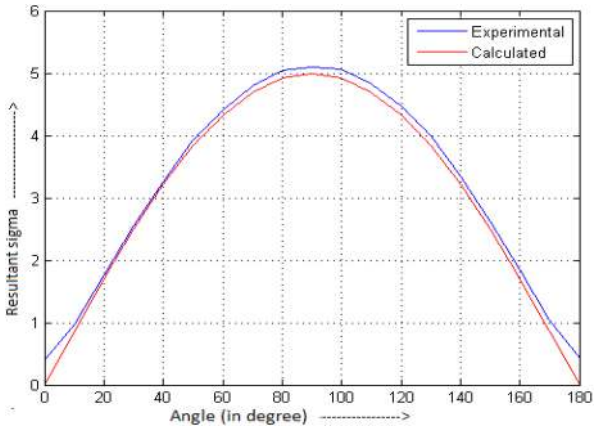

Fig. 2 *Blur parameter for degenerate blur kernel; Here angle denotes edge orientation*

3.4 Blur computation from PSF

Blur measurement can be performed in many ways, e.g. blur parameters, variance, blur radius etc. In this paper, we estimate the blur amount caused by underlying PSF in terms of associated blur parameters. We present analytical blur computation for complete degenerate and closed-form PSF in following propositions.

Proposition 1: Blur parameter with respect to Gaussian measurement function in some given direction θ for a complete degenerate PSF $\hat{f}_{\theta_1}(r)$ having principal direction θ_1 can be obtained as the projection of equivalent blur parameter σ_1 of corresponding Gaussian function model for PSF $\hat{f}_{\theta_1}(r)$ along the given direction as given in the following equation:

$$\sigma_\theta^2 = \sigma_1^2 \cos^2(\theta - \theta_1) \tag{4}$$

Proof: We take a step edge $AU(x') + B$ along y' such that its perpendicular makes angle θ with x axis. The principal direction for degenerate Gaussian kernel is along u direction which makes angle θ_1 with x axis. Since, PSF $\hat{f}_{\theta_1}(r)$ is complete with respect to Gaussian measurement function, hence can be analysed using equivalent Gaussian function model for PSF. Thus, the blurred edge I_b is the convolution of the step edge and the Gaussian function model for PSF with equivalent blur parameter as σ_1. Reference axis diagram is shown in Fig. 1.

$$I_b = (AU(x') + B) * \frac{e^{-(u^2/2\sigma_1^2)}}{\sqrt{2\pi\sigma_1^2}}$$

Taking Fourier Transform

$$\begin{aligned}\mathcal{F}\{I_b\} &= \mathcal{F}\{AU(x') + B\}\frac{\mathcal{F}\{e^{-(u^2/2\sigma_1^2)}\}}{\sqrt{2\pi\sigma_1^2}} \\ &= \mathcal{F}\{AU(x') + B\}e^{-\pi^2 u^2 2\sigma_1^2}\end{aligned}$$

Decomposing into x' and y' coordinate system and putting $y' = 0$

$$\begin{aligned}\mathcal{F}\{I_b\} &= \mathcal{F}\{AU(x') + B\}\, e^{-\pi^2(x'\cos(\theta-\theta_1) - y'\sin(\theta-\theta_1))^2 2\sigma_1^2} \\ &\equiv \mathcal{F}\{AU(x') + B\}\, e^{-\pi^2(x'\cos(\theta-\theta_1))^2 2\sigma_1^2}\end{aligned}$$

we note that $y' = 0$ as there is no variation in y' direction, leading to the above simplification. Now comparing with Fourier Transform of standard blur kernel

$$\sigma_\theta^2 = \sigma_1^2 \cos^2(\theta - \theta_1) \quad \square$$

Fig. 2 shows the blur parameter variation with the angle for the case of the degenerate blur kernel. We observe from the figure that estimated and calculated values are close. There is large error in case of smaller blur amounts as the estimation process becomes erroneous while estimating very low amount of blur. This verifies the blur kernel characterisation for degenerate case as given in Proposition 1.

Proposition 2: Blur parameter with respect to the Gaussian measurement function in some given direction θ for a complete closed-form PSF having principal directions θ_1 and θ_2, can be obtained as sum of projections of equivalent blur parameters σ_1 and σ_2 as given in (5). σ_1 and σ_2 are equivalent blur parameters corresponding to Gaussian function model for PSF with respect to Gaussian measurement function.

$$\sigma_\theta^2 = \sigma_1^2 \cos^2(\theta - \theta_1) + \sigma_2^2 \cos^2(\theta - \theta_2) \tag{5}$$

Proof: Proceeding same as in Proposition 1, we obtain

$$I_b = (AU(x') + B) * \frac{1}{2\pi\sigma_1\sigma_2} e^{-(u^2/2\sigma_1^2)} * e^{-(v^2/2\sigma_2^2)}$$

u and v are not necessarily perpendicular direction

$$\begin{aligned}\mathcal{F}\{I_b\} &= \mathcal{F}\{AU(x') + B\} \\ &\quad \times \frac{1}{2\pi\sigma_1\sigma_2}\mathcal{F}\left\{e^{-(u^2/2\sigma_1^2)}\right\}\mathcal{F}\left\{e^{-(v^2/2\sigma_2^2)}\right\} \\ &= \mathcal{F}\{AU(x') + B\}e^{-\pi^2 u^2 2\sigma_1^2}e^{-\pi^2 v^2 2\sigma_2^2}\end{aligned}$$

As the direction is preserved in Fourier Transform, we decompose in perpendicular directions and taking components in that direction

$$\begin{aligned}\mathcal{F}\{I_b\} &\equiv \mathcal{F}\{AU(x') + B\}\, e^{-\pi^2(x'\cos(\theta-\theta_1))^2 2\sigma_1^2} \\ &\quad \times e^{-\pi^2(x'\cos(\theta-\theta_2))^2 2\sigma_2^2}\end{aligned}$$

Now comparing this with standard blur kernel Fourier Transform, we get

$$\sigma_\theta^2 = \sigma_1^2 \cos^2(\theta - \theta_1) + \sigma_2^2 \cos^2(\theta - \theta_2)$$

Thus, using Propositions 1 and 2, the blur amount can be computed efficiently at any direction in terms of associated blur parameters of equivalent Gaussian function model for given complete PSF with respect to Gaussian measurement function. □

4 BPLC

Except for some special cases, e.g. microscopic, astronomical imaging, edges are common structural features appearing more often in any scene than the isolated points. Blurring of an edge is superposition of all the individual points constituting the edge often referred to as edge spread and corresponding blurring function is referred to as ESF. However, all blur PSF that produces the same amount of blur with respect to measurement function are equivalent as shown in Theorem 2.

Theorem 2: Given a blur PSF $f(x, y; \sigma)$, the equivalent blur spread at an edge with orientation perpendicular to θ can be given by the equivalent blur PSF $g(x; \sigma)$ with blur parameter σ_θ equal to the resultant blur present at the edge due to $f(x, y; \sigma)|_\theta$ with respect

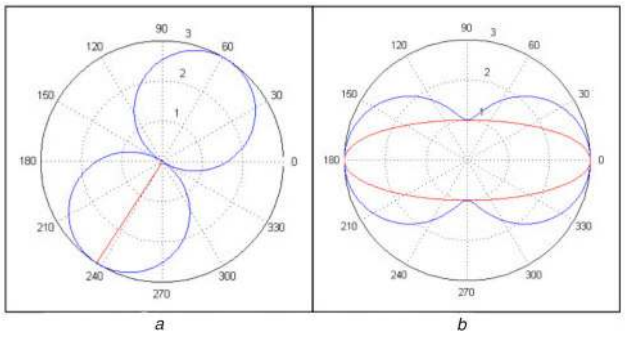

Fig. 3 *BPLC and PSF for*
(a) Degenerate kernel, ***(b)*** Elliptical kernel. Red curve denotes PSF and blue curve denotes corresponding BPLC

to measurement function as in (6). Here, x' denotes direction along θ orientation.

$$g(x';\sigma_\theta) \equiv f(x,y;\sigma)|_\theta \tag{6}$$

Proof: If PSF of a given system is $f(x,y;\sigma)$ and the given edge is oriented in perpendicular to θ direction (along y' direction), then we write the blurred edge I_b as a convolution of the step edge model $AU(x')+B$ with parameters A and B and blur PSF $f(x,y;\sigma)$.

$$\begin{aligned} &x' = x\cos\theta + y\sin\theta; \quad y' = x\sin\theta - y\cos\theta \\ &I_\mathrm{b} = (AU(x')+B) * f(x,y;\sigma) \end{aligned} \tag{7}$$

If edge is along $\theta+90$, the resultant blur amount present at the edge can be modelled with standard blur PSF $g(x';\sigma_\theta)$ which is the only function of x' and constant along edge direction y' and having blur parameter σ_θ. Then

$$I_\mathrm{b}(\theta) = (AU(x')+B) * g(x';\sigma_\theta) \tag{8}$$

Now applying measurements on both the original and equivalent models of the blur edge as given in (7) and (8), respectively, we get

$$\mathbb{M}(I_\mathrm{b}) = \sigma_\theta, \quad \mathbb{M}(I_\mathrm{b}(\theta)) = \sigma_\theta \tag{9}$$

From (9), we deduce that the two models given in (7) and (8) are equivalent with respect to measurement. Hence

$$g(x';\sigma_\theta) \equiv f(x,y;\sigma)|_\theta \quad \square$$

The equivalence of PSF is with respect to measurement function. The equivalent blur kernel for a specific source can be estimated from knowledge of underlying original kernel, for example in case of defocus (Gaussian kernel) f and g both are Gaussian. This is because the only constraint on the equivalent blur kernel at any edge is equal blur amount in a perpendicular direction to edge with respect to the measurement function. Thus, we can choose any equivalent blur kernel such that it produces equal blur amount at the edge as to original.

Since, the nature of defocus blur kernel is Gaussian, thus we choose the measurement function as Gaussian. Also, we choose equivalent blur kernel as measurement function, i.e. Gaussian for simplicity utilising Theorem 2. Thus, the estimated blur parameter σ using Gaussian measurement function is directly employed to model the underlying PSF with Gaussian function.

To be able to model the underlying blur PSF with measurement function to further analyse, its completeness with respect to measurement function is must as shown in Theorem 1. The completeness theorem enables us to model the underlying blur PSF at an edge with measurement function with parameter taken as an equivalent blur parameter σ. From Theorem 2, we conclude that specifying ESF rather than PSF is more useful for specifying the blur amount for edge structures than for point structures for commonly encountered image data (barring images from microscopic imaging and astronomy). However, since the ESF is also dependent upon the edge orientation, it becomes necessary to record ESF for all possible edge orientations for a given PSF to specify the blur amount in all directions.

However, we instead construct a blur parameter (σ) locus which gives us every information about the blur amount for all edge orientations with respect to a selected measurement function. We call it the BPLC. This curve carries the normalisation information of the blur parameter σ with respect to orientation for any given scale. We construct the BPLC from the PSF as follows:

Construction of BPLC

Step 1: Obtain PSF for some fixed scale (say 1).
Step 2: Adopt the specification for σ.
Step 3: Take a step edge in orientation $\theta+90°$ and blur it with given PSF.
Step 4: Determine σ for the edge using criteria in Step 2.
Step 5: Mark point at σ distance from the origin in θ direction.
Step 6: Repeat Steps 3, 4 and 5 for all orientation θ.
Step 7: Interpolate the points to get continuous BPLC.

We refer to BPLC at unity scale as normalised BPLC. Now, from the normalised BPLC, we can obtain the blur parameter σ for other scales in any direction by just multiplying by scale factor. For the case of defocus, the scale corresponds to the depth of objects from the lens. In case of rotational blur, scale corresponds to perpendicular distance from axis of rotation. Fig. 3 shows few examples for the BPLC along with the corresponding PSFs. The radial distance of any point on BPLC specifies the resultant blur in corresponding direction in the figure.

In general, only blurred image is provided in place of PSF. In such cases, we segment the image such that segments belong to the same scale using colour, contours, e.g. [40–42]. Then, we obtain the BPLC for different segments as discussed above.

The methods given in Propositions 1 and 2 can be used for generating the BPLC efficiently. The equivalent blur at any edge point can be equivalently written as sum of projections of PSF components, which is the same as the contribution of blur amount from the neighbouring points lying on the edge. These decompositions provide a powerful tool for various forms of analysis. In case of the complete closed-form PSFs and complete degenerate PSFs, BPLC is computed very efficiently using these methods. BPLC is obtained from the PSF, however, BPLC is a more useful representation than the PSF itself for blur related analysis as edges are primarily occurring features in images. The key differences of BPLC over PSF representation of blur are as follows:

- The BPLC is easy to estimate from a blurred image as it only requires blur amount in various directions – in short, it is only a *chart* of blur amount in all orientations and at all locations and measured using our selected measurement function (Gaussian kernel in our case) while, in general, the estimation of the PSF from a given image is an ill-posed problem as it is specified over the entire support of PSF, which is an entire 2D function.
- The BPLC yields a direct relation between the blur amount and the blur parameter of PSF, whereas PSF may not have such an explicit relation with the blur amount. Completeness of PSF with respect to measurement function as given in Theorem 1 allows us to model the PSF using associated measurement function.
- The BPLC contains the blur amount variation in all directions at some given scale, hence, carries normalisation information for the blur amount in various directions. This is essential information for scale map estimation in case of anisotropic PSFs. Again, the PSF does not contain such information explicitly.

Fig. 4 shows a test image blurred with Defocus Blur PSF with blur parameter σ as 1 in the horizontal direction and 3 in the vertical direction. We observe from the figure that the two rectangles appear at different depths in depth map obtained without resolving

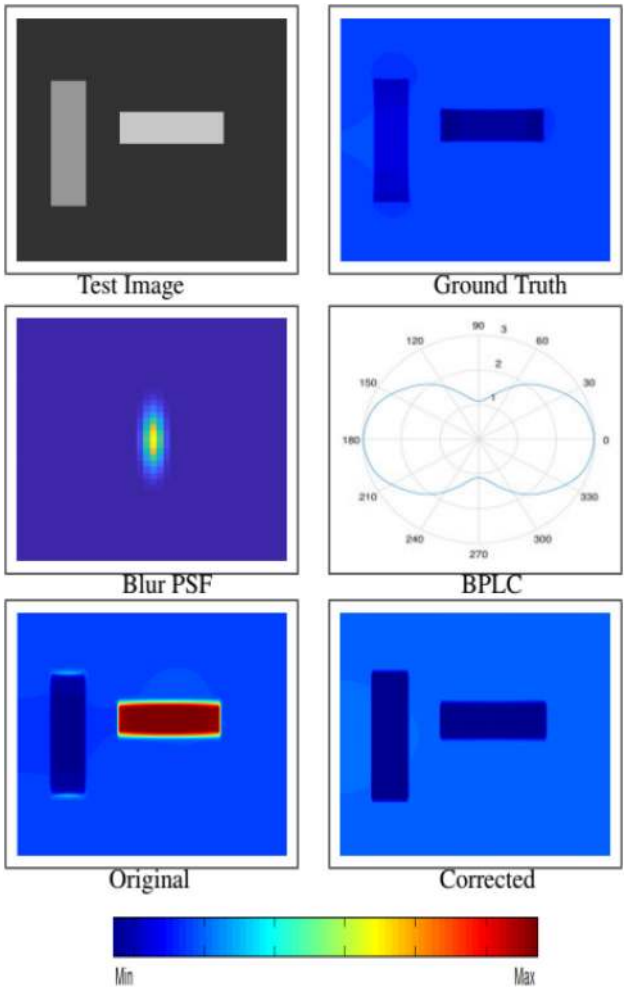

Fig. 4 *Effect of anisotropic PSF on scale map*

scale ambiguity. The two rectangles appear at the same depth in depth map obtained after correcting scale ambiguity using BPLC.

- The BPLC can be used to characterise and compare different PSFs with respect to a measurement function. Some of these properties are described in Section 5.2.
- Even a non-parametric PSF can also be parameterised using the BPLC with respect to a selected measurement function. Few examples of such PSFs are shown in Fig. 5. We observe that these PSFs do not have the parametric form, however using BPLC these PSFs can be expressed as orientation varying blur parameter.

Thus, we observe that the BPLC is a more convenient representation compared to PSF for blur analysis.

4.1 Image resizing and BPLC

As discussed above, image resizing is a very common operation in image processing. We compute the change in blur amount due to image resizing as shown in Theorem 3.

Theorem 3: The effect of image downsampling on the blur parameter can be equivalently obtained by downsampling the BPLC by the same factors.

Proof: The blur parameter along any direction can be obtained from the BPLC. Since blur parameter σ is the distance from the origin of the BPLC in the given direction

$$x^2 + y^2 = \sigma^2$$

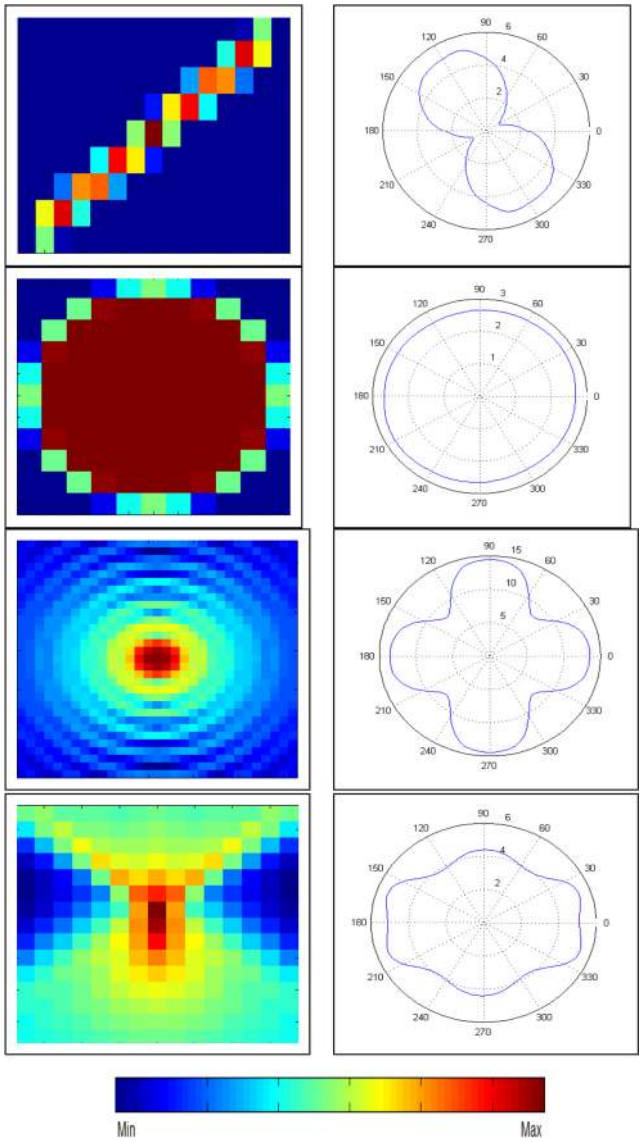

Fig. 5 *Examples of observable and non-observable blur kernels and corresponding BPLCs, first two rows show the example of observable PSFs and last two rows show example of non-observable PSFs. Here angle denotes edge angle*

Downsampling image by a factor M in horizontal and by factor N in the vertical direction the new axes become x' and y', which are related to axes x and y as given below.

$$x' = \frac{x}{M}; \quad y' = \frac{y}{N}$$

The equivalent downsampled BPLC on modified axes becomes as follows:

$$\frac{M^2 x'^2}{\sigma^2} + \frac{N^2 y'^2}{\sigma^2} = 1$$

To compute the blur values in direction having slope m, we substitute $y' = mx'$ to find the new location (x', y').

$$\frac{M^2 x'^2}{\sigma^2} + \frac{m^2 N^2 x'^2}{\sigma^2} = 1$$

$$\Rightarrow x'^2 = \sigma^2 \frac{1}{m^2 N^2 + M^2}$$

The new blur parameter σ_f is the distance of the new intersection point (x', y') from the origin.

$$\sigma_f^2 = x'^2 + y'^2 = \sigma^2 \frac{m^2 + 1}{m^2 N^2 + M^2} \quad (10)$$

The σ_f obtained from downsampling the BPLC as given by (10). Hence, blur parameter of the downsampled image can be equivalently obtained by downsampling the BPLC itself by the same factors in the corresponding directions. □

Thus, we can downsample the BPLC to get the new blur parameters for all directions. However, the downsampled BPLC is not valid for the downsampled image as the directions of the edges will also change with downsampling. However, change in direction can be obtained using downsampling factors M and N. Using these relations, new BPLC can be obtained from old BPLC. Fig. 6 shows downsampled BPLC and BPLC for downsampled image by factor 2 in the vertical direction. We observe from the figure that the downsampled BPLC and BPLC for downsampled image are not the same.

Fig. 6 shows downsampled BPLC and BPLC for a downsampled image by factor 2 in the vertical direction. Now, if we apply non-identical downsampling on the orthogonal directions, then σ_{scaled} is dependent on orientation of the edge. We downsample the image by a factor 2 only in the vertical direction. Blur estimates using (10) match with those observed using experiments. Matching error is large in case of small amount of blur as estimation of small blur results in greater inaccuracies. Fig. 7 shows the relation between σ_{scaled} and σ_{orig} for various oriented edges (0°, 15°, 30° and 90°).

The estimated blur values σ_{orig} and σ_{scaled} follow a linear relationship as expected. Besides that, the slope is same as that obtained from (10) for various orientations which verifies the correctness of the proposed theory for the downsampling along with correctness of our blur kernel decomposition methodology.

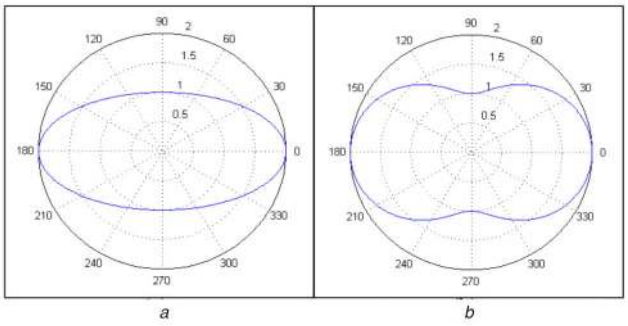

Fig. 6 *BPLC for circular blur kernel with $\sigma = 2$ for*
(a) Downsampled BPLC, ***(b)*** Downsampled image by factor 2 in vertical direction

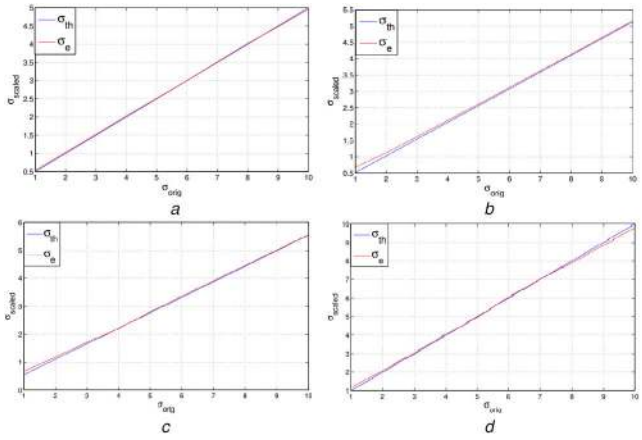

Fig. 7 *Non-identical downsampling (downsampling by 2 vertically)*
(a) 0°, ***(b)*** 15°, ***(c)*** 30°, ***(d)*** 90°

5 Applications of BPLC

5.1 Reliable defocus map estimation

Depth from defocus is an active area of research. Edges are high frequency contents of an image and are most suitable for defocus extraction because they provide accurate localisation. Accurate edge localisation is essential for accurate blur estimation. However, defocus estimation often becomes erroneous due to the presence of non-ideal edges, false edges, texture edges or other types of blurs than defocus, e.g. motion blur. Elder and Zucker [43] proposed a local scale (σ of the Gaussian filter) control for edge detection and defocus blur estimation. Using this, local minimum reliable scale, accurate edges are obtained. These edge points are used to estimate the amount of defocus blur for depth information.

The state-of-the-art methods [3, 44–48] rely on machine learning techniques for inferring depth map of a given image. These techniques, utilise texture features to infer depth. However, performance of these techniques heavily depends upon whether training and testing images are of similar scenes and whether they are both captured using similar imaging systems. Also, these techniques produce a coarse depth of the scene and fine depth-structures are not preserved in the depth map, as opposed to edge-based defocus map estimation techniques.

Various methods such as [7–9, 49, 50] estimate defocus blur at the edge locations by employing different strategies. Zhuo and Sim [9] produce reliable defocus map in the presence of noise employing gradient ratio at the edges. Srikakulapu *et al.* [51] use reliable local scale using Elder and Zucker's method [43] to obtain the edge points at which defocus is estimated. However, the presence of non-ideal edges, false edges and other types of blurs affect the performance of these techniques.

Karaali and Jung [52] increased the accuracy of the estimated defocus map of [9] by selecting consistent edge points across scale in a 'Canny' edge detector [53]. Then, the scale for defocus estimation (re-blurring σ in [9]) is adaptively taken as half of the maximum detectable edge-scale for consistent edge points. This method mainly focuses to achieve accuracy in the estimated defocus map by taking appropriate adaptive scale in defocus measurement using Zhuo's method [9] for different edges. However, errors due to shadow, false edges may still persist.

All the above methods are edge-based methods, in which defocus blur parameter σ is estimated at the edges. This map containing defocus blur information only at edges is referred to as 'Sparse Defocus Map'. The estimated blur parameter σ for defocus blur corresponds to the depth, i.e. higher defocus values represents higher depth. So, we propagate this information present at edges in Sparse Defocus Map over the entire image to generate the 'Full Defocus Map' using Levin's [54] closed-form colourisation scheme based on texture, colour correlation.

Usually, the defocus blur kernel is modelled as Gaussian function. The inaccuracies arising due to the presence of false and non-ideal edges can be corrected by exploiting the blur kernel properties. Direct kernel fitting may not always be accurate due to the presence of noise. We use BPLC characteristics for downsampling to filter out ambiguous defocus values. This not only yields a more accurate defocus map but also measures the reliability of the estimated defocus values.

In this section, we introduce the framework of the proposed method for reliable defocus map estimation. Defocus blur is Gaussian in nature. Higher depth induces higher defocus blur. Hence, depth information can be captured by measuring the blur parameter σ for underlying defocus blur kernel. However, measurement of defocus blur using σ is often erroneous due to the presence of non-ideal, false edges, noise etc. This error can be limited by imposing properties of associated defocus blur kernel. Effect of downsampling or scale-space related constraint for defocus blur kernel is one such property. Trentacoste *et al.* [10] derived the effect of downsampling on the defocus blur kernel as given by the following equation:

$$\sigma \xrightarrow{\downarrow d} \frac{\sigma}{d} \quad (11)$$

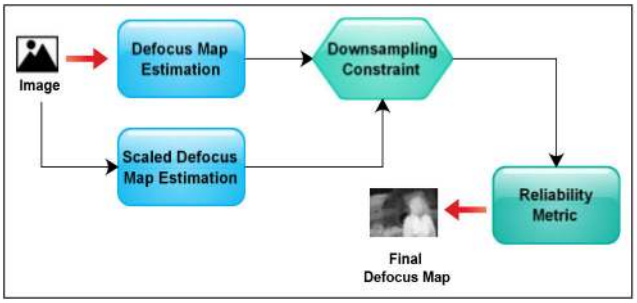

Fig. 8 *Flow diagram for reliable blur estimation*

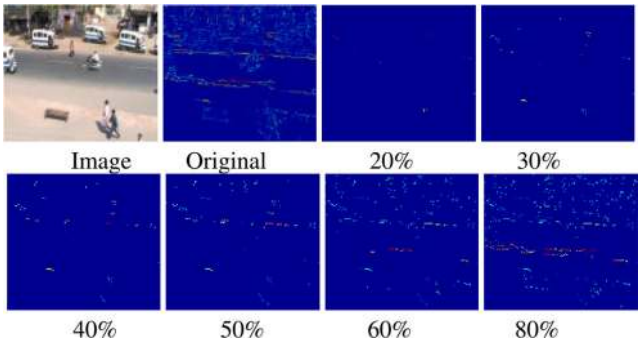

Fig. 9 *Effect of various reliability score threshold on sparse defocus map*

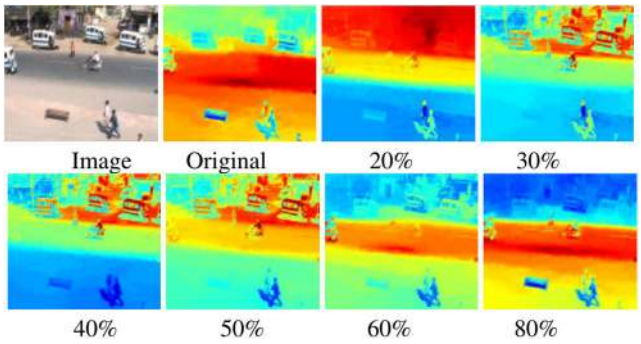

Fig. 10 *Effect of various reliability score threshold on full defocus map*

Blur parameter σ reduces to σ/d after downsampling by factor d, which is a simple case of generalised result derived in (10). We impose this constraint to remove the non-ideal, false edges attributed to the erroneous defocus map.

5.1.1 Proposed reliable defocus map creation: In this subsection, we describe the proposed method for reliable defocus map estimation. Fig. 8 shows the flow diagram for the proposed reliable defocus blur estimation framework. First, we estimate the sparse defocus map S and S_2 of the original image I as well as the resized image I_2, respectively. We have used Zhuo's method [9] for sparse defocus map creation and a downsampling factor of 2 along both horizontal and vertical directions. We then downsample the sparse defocus map S using nearest neighbour filter to obtain sparse defocus map S_d. The calculated sparse defocus map S_d is compensated for downsampling as in (11) to obtain the sparse defocus map S_e. Then, we calculate the reliability metric R at every point in the image where the sparse defocus map is non-zero, i.e. at the edges in edge map e. We create the final defocus map based on the reliability metric.

The reliability metric $R(x)$ at the point x is defined as the relative error between the defocus blurs of the original and resized images after imposing the blur transformation at each edge point as given in (11), i.e. between S_e and S_2 as given in the following equation:

$$R(x) = \frac{|S_2(x) - S_\mathrm{e}(x)|}{S_\mathrm{e}(x)} \tag{12}$$

Input: Image I, Threshold t
Output: Reliability Map R

Estimate edge map e of I and Sparse Defocus Map S
Obtain I_2 by downsampling I by factor 2
Estimate Sparse Defocus Map S_2 from I_2
Calculate Sparse Defocus Map S_d by downsampling S with downsampling factor 2 using nearest neighbor filter
Calculate Sparse Defocus Map S_e using (11)
for $\forall x \in e$ **do**
  if $\sigma < 0.5$ **then**
    continue
  else
    $R(x) = |S_e(x) - S_2(x)|/S_e(x)$
  end if
end for
for $\forall x \in e$ **do**
  if $R(x) < t$ **then**
    continue
  else
    Omit x from e
  end if
end for

Fig. 11 *Algorithm 1: Reliability map estimation*

We retain defocus blur values at only those edge points where the reliability is greater than a threshold (in other words, the relative error is less than a chosen threshold). Figs. 9 and 10 highlight the effect of various reliability threshold on the sparse defocus map and full defocus map for selected sample image.

We observe from the figure that the lower threshold values of reliability score limit the number of edge points in sparse defocus map drastically. This may result in poor quality of full defocus map as shown in Fig. 10. Whereas, large values of the reliability threshold result in inclusion of erroneous edge points in the sparse defocus map, which in-turn produce erroneous full defocus map as shown in Fig. 10. So, we choose a mid-range value 40% for the reliability threshold. We also skip relative errors for blur lesser than a blur threshold 0.5, as blur estimation is usually inaccurate for lower blur values. Both the reliability threshold and the blur threshold govern the number of edge points that get included in the reliable defocus map. Algorithm 1 (see Fig. 11) shows sample pseudo code for generating Reliability Map R. The proposed method calculates the reliability map based on the scaling constraint as in (10) to reduce the error of erroneous blur detection at the non-ideal, false edge points to obtain more accurate defocus map.

Holes are another main source of error in edge-based defocus map estimation [55]. In this paper, we have not applied hole filling in order to have a one to one comparison with the state-of-the-art Karaali method [52].

5.1.2 Proposed defocus map evaluation framework: Defocus map cannot be directly compared with depth map as depth and defocus are related to each other by (13). Conversion of defocus to depth requires the associated camera parameters k_1 and k_2. However, usually, these parameters are not available. Also, estimated defocus value results in value proportional to original defocus value due to effect such as downsampling as shown in (11), matting [54] required for full defocus map estimation from sparse defocus map, etc. We use optics to relate defocus parameter σ to depth d as given by (13) based on framework discussed in [56]. Here, k_1 and k_2 are camera parameters that are always positive

$$\sigma = k_1 - \frac{k_2}{d} \quad \Rightarrow \quad d = \frac{k_2}{k_1 - \sigma} \tag{13}$$

Estimation of defocus is relative in nature, i.e. only ordering of defocus with respect to depth is retained. So, we estimate the parameters k_1 and k_2 for the given image. Equation (14) shows measurement model of depth and defocus. Here, w represents the measurement error.

$$\sigma = k_1 - k_2\frac{1}{d} + w \tag{14}$$

Now, we stack all the depth and defocus values and obtain system of linear equations given by (15)

$$\begin{aligned}\sigma_1 &= k_1 - k_2\frac{1}{d_1} + w_1\\ &\vdots\\ \sigma_n &= k_1 - k_2\frac{1}{d_n} + w_n\end{aligned}$$
$$\underbrace{\begin{bmatrix}\sigma_1\\ \vdots\\ \sigma_n\end{bmatrix}}_{\Sigma} = \underbrace{\begin{bmatrix}1 & \frac{1}{d_1}\\ \vdots & \vdots\\ 1 & \frac{1}{d_n}\end{bmatrix}}_{\mathbf{H}}\underbrace{\begin{bmatrix}k_1\\ -k_2\end{bmatrix}}_{\mathbf{K}} + \underbrace{\begin{bmatrix}w_1\\ \vdots\\ w_n\end{bmatrix}}_{\mathbf{W}} \tag{15}$$

$$\mathbf{\Sigma} = \mathbf{HK} + \mathbf{W} \;\Rightarrow\; \hat{\mathbf{K}} = (\mathbf{H}^T\mathbf{H})^{-1}\mathbf{H}^T\mathbf{\Sigma} \tag{16}$$

Using (15) and (16), we estimate the parameters k_1 and k_2 which minimises the least-square error. Thus, the estimated parameters k_1 and k_2 measures the distortions in the relative defocus-depth correspondence. The positive values of parameters k_1 and k_2 denote that the camera is focused at the nearest object in the scene from the camera and defocus blur increases with the depth. In such a case, maximum defocus blur equals to be k_1 at the farthest object in the scene. While positive value of k_1 and negative value of k_2 denote that the camera is focused at the farthest object in the scene and defocus blur value decrease with depth. Negative value of k_1 is not possible and denotes serious errors in estimation process. These two types of focusing condition give rise to Focal Plane Ambiguity and can be removed using method such as [57, 58]. Generally, images of the scene with focus at nearest object are taken for full defocus map quality evaluation. In such cases, distortions in defocus-depth pair appear as erroneous values (negative values) of k_1 and k_2. Thus, accuracy of an estimated defocus map can be assessed using this framework.

5.1.3 Results: We carried out a quantitative evaluation of proposed reliability method for few images from [44] where ground truths are available. Our test environment contains MATLAB 2014 in Linux environment with processor Intel(R) i5 CPU 650@3.20GHz. Fig. 12 shows the colourmap representation for both ground truth depth maps and estimated defocus maps.

We transform estimated defocus values into depth values using (13) and evaluate the accuracy of depth estimation using estimated k_1 and k_2. We use relative error 'rel', logarithmic error 'log10' and root mean square error 'RMSE' defined in [44] for depth map error calculation. Relative error 'rel' is defined as mean fraction error with respect to depth as given in (17). Here, d_e is estimated depth, d is ground truth depth and N is number of sample points at which depth is estimated, i.e. image size

$$\text{rel} = \sum_{i=1}^{N}\frac{1}{N}\frac{|d_e - d|}{d} \tag{17}$$

Logarithmic error 'log10' is defined as mean of Logarithmic of the ratio of estimated depth to ground truth depth as given in (18).

$$\log 10 = \sum_{i=1}^{N}\frac{1}{N}\left|\log_{10}\frac{d_e}{d}\right| \tag{18}$$

Higher values of 'rel', 'log10' and RMSE represent a higher error in the estimated defocus values. These errors are with respect to estimated k_1 and k_2 which minimises the RMSE as discussed in Section 5.1.2. Following are two examples of such an evaluation for images from [44]. The ground truth in these examples represents the depth maps, while state-of-the-art methods estimate defocus maps.

Table 1 shows error metrics for the example images shown in Fig. 13. We observe that the method of Zhuo and Sim [9] and Karaali and Jung [52] estimate parameter k_2 as a negative quantity which must be positive as discussed in the previous section (camera is focused at the nearest object of the scene). Also, the proposed method produces defocus map with least error metrics. We thereby conclude from Table 1 that the proposed method produces a more accurate defocus map compared to state-of-the-art methods [9, 52].

We also generated the full defocus map for a few test images with and without correction and performed a qualitative comparison. The results are shown in Fig. 14. First column shows the test images, the second column shows the defocus map results for Zhuo's method [9], third column shows defocus map obtained using Srikakulapu's method [51] without applying the hole filling, fourth column shows defocus map for Karaali's method [52] by utilising edge-aware matting given in [52], fifth column shows defocus map for Karaali's method [52] using Levin's [54] closed-form matting and last column shows defocus map obtained with the proposed correction. Sky region in the third image introduces large error in estimated defocus map as no defocus information is present for such regions. These regions are referred as 'Hole regions' and can be efficiently corrected using [55]. We observe from the figure that the defocus map obtained after correction is far more accurate and preserves the depth order of the scene in defocus maps compared to state-of-the-art methods. Karaali's method [52] by utilising edge-aware matting Karaali's method is more suited to images with less finer texture/details and does not work well for common images. While Karaali's method [52] using Levin's [54] closed-form matting preserves the fine details in the defocus map corresponding to scene structure. However, Karaali's method does not effectively remove edge locations corresponding to erroneous defocus values which in turn result in the erroneous defocus maps. Similarly, defocus maps obtained using Zhuo's method [9] and Srikakulapu's method [51] also contain erroneous regions. Thus, the proposed method results in more accurate and reliable defocus map creation compared to state-of-the-art methods. More results can be found in [59].

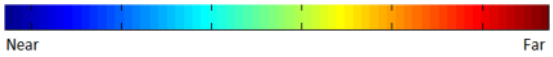

Fig. 12 *Colourmap for defocus map representation*

Table 1 Example 1: Depth estimation accuracy

Image	Method	k_1	k_2	rel	log10	RMSE
	Zhuo [9]	0.693	−0.869	1.224	1.764	51.315
Img1	Karaali [52]	0.783	−0.316	1.062	2.069	50.505
	Proposed	1.186	0.866	0.916	1.322	49.403
	Zhuo [9]	0.655	0.018	0.997	2.759	47.51
Img2	Karaali [52]	0.540	−0.601	1.179	1.804	48.45
	Proposed	0.988	0.650	0.921	1.344	46.96

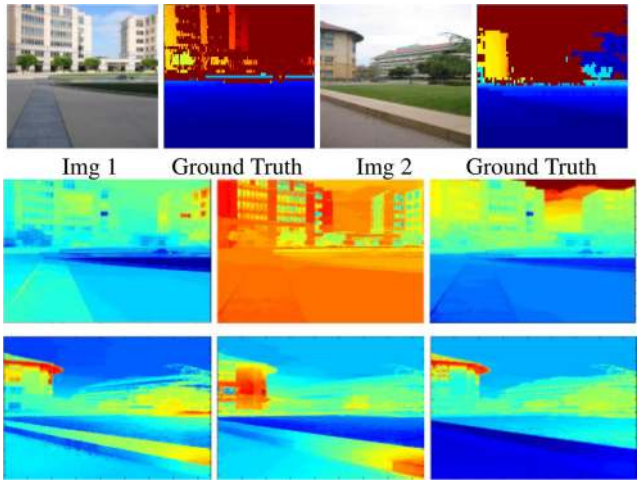

Fig. 13 *Sample quantitative comparison of defocus map estimation, defocus maps in second row correspond to Img 1 and its ground truth given in first and second columns of first row, respectively, and defocus maps in third row correspond to Img 2 given in first row*

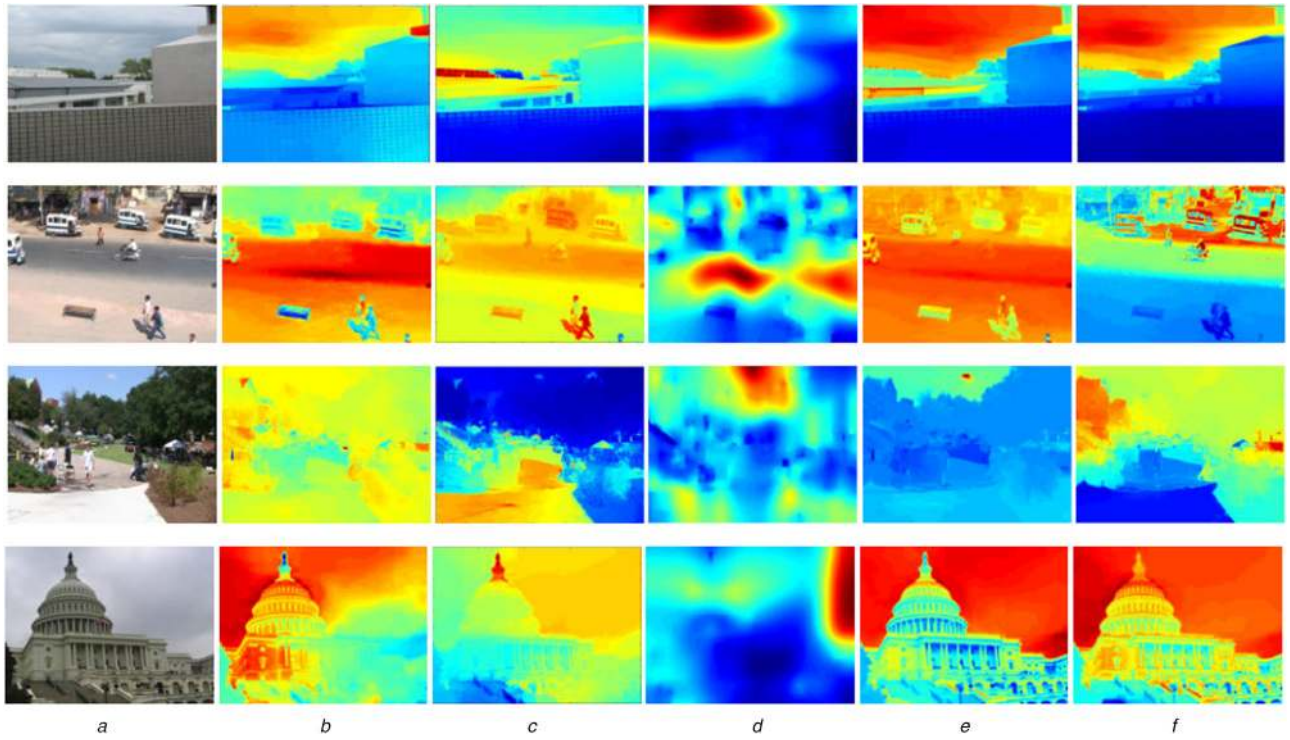

Fig. 14 *Defocus map comparison, Column (a) shows test images, Columns (b), (c), (d) and (e) show defocus maps obtained using Zhuo [9], Srikakulapu [51], Karaali[1] [52], Karaali[2] [52] and proposed methods respectively; Karaali[2] utilises edge-aware matting given in [52] to obtain full defocus map from sparse defocus map, Karaali[2] utilises Levin's [54] closed-form matting to obtain the full defocus map from the sparse defocus map*

(a) Image, ***(b)*** Zhuo [9], ***(c)*** Srikakulapu [51], ***(d)*** Karaali[1] [52], ***(e)*** Karaali[2] [52], ***(f)*** Proposed

5.2 PSF characterisation using BPLC

In this section, we develop and describe some more results that can be used for PSF characterisation. Firstly, the closed-form blur PSF can be expressed with two perpendicular degenerate blur kernel components as in Theorem 4. Thus, only these two perpendicular components are observable.

Theorem 4: Any closed-form blur kernel which is complete with respect to Gaussian measurement function can be equivalently expressed as another closed-form blur kernel with two perpendicular degenerate blur kernel components.

The PSF can be estimated by estimating the two perpendicular degenerate blur kernel components. The nature of energy distribution for a given degenerate PSF can be inferred from gradient profile at the edge perpendicular to the principal direction of the degenerate PSF.

$$\begin{aligned} I &= (AU(x)+B) * f(x;\sigma) \\ \nabla I &= \nabla((AU(x)+B) * f(x;\sigma)) \\ &= (\nabla(AU(x)+B)) * f(x;\sigma) \\ &= (A\delta(x)) * f(x;\sigma) \\ \nabla I &= A f(x;\sigma) \end{aligned} \tag{19}$$

Equation (19) shows that the gradient profile at step edge yields the resultant blur kernel in perpendicular direction. We use this result in Proposition 3 to estimate the full closed-form PSF.

Proposition 3: Any closed-form blur PSF can be obtained using gradients I_x and I_y^1 in two perpendicular principal directions of equivalent closed-form blur PSF as given by (20).

$$F(x,y) = \frac{I_x * I_y^1}{||I_x * I_y^1||} \tag{20}$$

Proof: As given in (19), gradient gives profile of underlying equivalent blur kernel at edge in perpendicular direction. If blur PSF $F(x, y)$ has two principal components $f_1(x; \sigma_1)$ and $f_2(y; \sigma_2)$ with parameters σ_1 and σ_2 in x and y directions, respectively, the blur PSF can be recovered as follows using gradient edge profile at the step edge with parameters A_1 and B_1.

$$\begin{aligned} F(x, y) &= f_1(x; \sigma_1) * f_2(y; \sigma_2) \\ I &= (A_1 U(x) + B_1) * F(x, y) \\ I_x &= A_1 \delta(x) * f_1(x; \sigma_1) * f_2(y; \sigma_2) \\ I_x &= A_1 f_1(x; \sigma_1) \end{aligned}$$

Similarly, from gradient profile at step edge with parameters A_2 and B_2 in the perpendicular direction

$$\begin{aligned} I^1 &= (A_2 U(y) + B_2) * F(x, y) \\ I_y^1 &= A_2 f_2(y; \sigma_2) \end{aligned}$$

Now, the convolution of I_x and I_y^1 along with normalisation, blur PSF $F(x, y)$ can be recovered

$$F(x, y) = \frac{I_x * I_y^1}{\left\| I_x * I_y^1 \right\|}$$

Thus, any closed-form blur PSF can be obtained using gradient in two perpendicular principal directions of the equivalent closed-form blur kernel. □

Fig. 15 shows the gradient profiles of example blur kernels in the principal direction. Size of the PSF can be retrieved from the length of the gradient. The underlying PSF can be recovered by convolving the two blur gradient profiles in two principal directions. We refer to a PSF as *Observable,* if the PSF can be equivalently expressed as closed-form blur kernel with two perpendicular degenerate PSFs as in Theorem 4. Such a PSF can be recovered using Proposition 3. However, principal directions and completeness of the PSF may not be available at all in general as we are provided only the blurred image observation. Convexity of BPLC can be used to determine the observability of a PSF as shown in Theorem 5.

Theorem 5: BPLC of a complete observable blur kernel inscribes a convex region.

Proof: Theorem 4 shows that any complete closed-form blur PSF can be equivalently expressed as a closed-form PSF with two perpendicular components. So, the resultant blur σ_f in direction θ can be expressed in terms of and blurs of degenerate components with blur parameters σ_1 and σ_2.

$$\sigma_f^2 = \sigma_1^2 \cos^2\theta + \sigma_2^2 \sin^2\theta$$

Thus, BPLC Γ can be written as a parametric form given by (21).

$$\Gamma(\theta) = (\sigma_1 \cos\theta, \sigma_2 \sin\theta) \tag{21}$$

'Signed Curvature' κ for parametric $(r(t), s(t))$ is given by (22).

$$\kappa = \frac{r's'' - s'r''}{(r'^2 + s'^2)^{3/2}} \tag{22}$$

Thus, the curvature of the BPLC Γ is

$$\kappa = \frac{\sigma_1 \sigma_2}{(\sigma_1^2 \sin^2\theta + \sigma_2^2 \cos^2\theta)^{3/2}}$$

So, 'Signed Curvature' κ is positive throughout hence the curvature is in the same direction. Thus, BPLC Γ inscribes a convex region. □

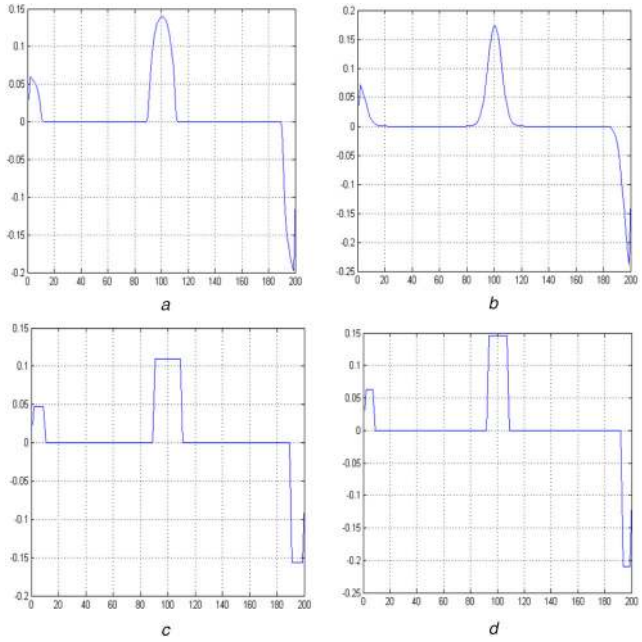

Fig. 15 *Gradient profile along principal directions of blur kernel*
(a) Disk blur kernel, ***(b)*** Gaussian blur kernel, ***(c)*** Square blur kernel, ***(d)*** Motion blur kernel

Table 2 Blind deconvolution metric

Method	Proposed	Sun *et al.* [29]	Dark channel [61]
SSIM	0.8802	0.9211	0.8947
PSNR	28.63	30.27	27.92
time, s	39.97	860.87	267

In general, determining the convexity is an NP-hard problem. However, the convexity of any curve can be easily determined from number of extrema as given in Proposition 4.

Proposition 4: BPLC of any blur kernel is observable if number of minima is either two or uncountable.

From Theorem 5, either BPLC of an observable PSF is constant, i.e. uncountable minima as all values are same or has two minima. Using this property, we determine from the BPLC if the PSF is observable. However, it is only a necessary condition and is not sufficient for the observability.

Fig. 5 shows examples of observable and non-observable blur PSFs. We observe that the nature of BPLC is convex in either side of axis joining the minima points on BPLC for observable PSFs in first two rows. Thus, all these PSFs are observable with respect to underlying selected measurement function, i.e. Gaussian kernel. While PSFs in the last two rows result in non-convex BPLC, hence, these kind of PSFs are not complete with respect to measurement basis function, i.e. Gaussian kernel and thus cannot be analysed. Different suitable measurement function which fulfils the completeness requirement can be used instead. Using these tools, we can characterise an underlying PSF from the given blurred image.

We performed the PSF characterisation using for PSFs in [2]. Although these PSFs are not complete with respect to measurement function, yet PSF generated using proposed theoretical framework results in PSF estimates with one-to-one correspondence. These results are given in [60]. We also used these PSF estimates as initial estimates in [29] for blind deconvolution purpose on [2] database as given in Table 2.

We observe from the table that the proposed method produces deconvolved output of comparable quality to that of state-of-the-art methods with much less time complexity. Thus, we conclude that the BPLC can be used for PSF characterisation and analysis even when no prior information is available.

6 Conclusion

In this paper, we present a novel concept of blur point locus curve to represent, analyse and characterise the underlying PSF. We also present a classification of the PSFs that allows us to compute the blur amount in any orientation efficiently. We also introduce notion of completeness and measurement function which allows to perform PSF related analysis in alternate domain (Measurement function domain). These tools can be used in various applications such as characterisation of an optical system, PSF decomposition and generation, blind deconvolution, image tampering detection etc.

7 References

[1] Joshi, N., Szeliski, R., Kriegman, D.J.: 'PSF estimation using sharp edge prediction'. 2008 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR 2008), Anchorage, USA, 2008, pp. 1–8

[2] Levin, A., Weiss, Y., Durand, F., *et al.*: 'Understanding blind deconvolution algorithms', *IEEE Trans. Pattern Anal. Mach. Intell.*, 2011, **33**, (12), pp. 2354–2367

[3] Zhu, X., Cohen, S., Schiller, S., *et al.*: 'Estimating spatially varying defocus blur from a single image', *IEEE Trans. Image Process.*, 2013, **22**, (12), pp. 4879–4891

[4] Gong, D., Tan, M., Zhang, Y., *et al.*: 'Blind image deconvolution by automatic gradient activation'. Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition, Las Vegas, USA, 2016, pp. 1827–1836

[5] Kumar, H., Gupta, S., Venkatesh, K.: 'A novel method for image compression using spectrum'. 2017 Ninth Int. Conf. on Advances in Pattern Recognition (ICAPR), Bangalore, India, 2017, pp. 1–6

[6] Richardson, W.H.: 'Bayesian-based iterative method of image restoration', *JOSA*, 1972, **62**, (1), pp. 55–59

[7] Bae, S., Durand, F.: 'Defocus magnification', in Cohen-Or, D., Slavík, P.: (Eds.) '*Computer graphics forum*', vol. 26 (Wiley Online Library, 2007) pp. 571–579

[8] Tang, C., Hou, C., Song, Z.: 'Defocus map estimation from a single image via spectrum contrast', *Opt. Lett.*, 2013, **38**, (10), pp. 1706–1708

[9] Zhuo, S., Sim, T.: 'Defocus map estimation from a single image', *Pattern Recognit.*, 2011, **44**, (9), pp. 1852–1858

[10] Trentacoste, M., Mantiuk, R., Heidrich, W.: 'Blur-aware image downsampling', in Chen, M., Deussen, O.: (Eds.) '*Computer graphics forum*', vol. 30 (Wiley Online Library, 2011), pp. 573–582

[11] Hong, L.: 'Device and method for estimating defocus blur size in an image'. 2011. Google Patents. US Patent 8,068,688

[12] Qin, F., Cao, L., Zhu, L., *et al.*: 'Blind single-image super resolution reconstruction with defocus blur', *Sens. Transducers*, 2014, **169**, (4), pp. 77–83

[13] Pang, Y., Zhu, H., Li, X., *et al.*: 'Classifying discriminative features for blur detection', *IEEE Trans. Cybern.*, 2015, **46**, (10), pp. 2220–2227

[14] Pang, Y., Zhu, H., Li, X., *et al.*: 'Motion blur detection with an indicator function for surveillance machines', *EEE Trans. Ind. Electron.*, 2016, **63**, (9), pp. 5592–5601

[15] Yang, D., Qin, S.: 'Restoration of partial blurred image based on blur detection and classification', *J. Electr. Comput. Eng.*, 2016, **2016**, p. 1

[16] Lu, B., Chen, J.C., Chellappa, R.: 'Unsupervised domain-specific deblurring via disentangled representations'. Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition, Long Beach, USA, 2019, pp. 10225–10234

[17] Xu, X., Pan, J., Zhang, Y.J., *et al.*: 'Motion blur kernel estimation via deep learning', *IEEE Trans. Image Process.*, 2017, **27**, (1), pp. 194–205

[18] Nimisha, T.M., Kumar-Singh, A., Rajagopalan, A.N.: 'Blur-invariant deep learning for blind-deblurring'. Proc. of the IEEE Int. Conf. on Computer Vision, Venice, Italy, 2017, pp. 4752–4760

[19] Nah, S., Hyun-Kim, T., Mu-Lee, K.: 'Deep multi-scale convolutional neural network for dynamic scene deblurring'. Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition, Honolulu, Hawaii, 2017, pp. 3883–3891

[20] Madam-Nimisha, T., Sunil, K., Rajagopalan, A.: 'Unsupervised class-specific deblurring'. Proc. of the European Conf. on Computer Vision (ECCV), Munich, Germany, 2018, pp. 353–369

[21] Wang, B., Liu, G., Wu, J.: 'Blind deblurring of saturated images based on optimization and deep learning for dynamic visual inspection on the assembly line', *Symmetry*, 2019, **11**, (5), p. 678

[22] Kotera, J., Šroubek, F.: 'Blind deconvolution of images with model discrepancies using maximum a posteriori estimation with heavy-tailed priors'. Proc. SPIE/IS&T Electronic Imaging, San Francisco, USA, 2015 (Digital Photography XI, vol. 9404)

[23] Lai, W.S., Ding, J.J., Lin, Y.Y., *et al.*: 'Blur kernel estimation using normalized color-line prior'. Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition, Boston, USA, 2015, pp. 64–72

[24] Bai, Y., Cheung, G., Liu, X., *et al.*: 'Graph-based blind image deblurring from a single photograph', *IEEE Trans. Image Process.*, 2018, **28**, (3), pp. 1404–1418

[25] Liu, Y., Dong, W., Gong, D., *et al.*: 'Deblurring natural image using super-Gaussian fields'. Proc. of the European Conf. on Computer Vision (ECCV), Munich, Germany, 2018, pp. 452–468

[26] Chen, L., Fang, F., Wang, T., *et al.*: 'Blind image deblurring with local maximum gradient prior'. Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition, Long Beach,USA, 2019, pp. 1742–1750

[27] Aljadaany, R., Pal, D.K., Savvides, M.: 'Douglas-Rachford networks: learning both the image prior and data fidelity terms for blind image deconvolution'. Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 2019, pp. 10235–10244

[28] Bai, Y., Jia, H., Jiang, M., *et al.*: 'Single image blind deblurring using multi-scale latent structure prior', *IEEE Trans. Circuits Syst. Video Technol.*, 2019, doi:10.1109/TCSVT.2019.2919159

[29] Sun, L., Cho, S., Wang, J., *et al.*: 'Edge-based blur kernel estimation using patch priors'. IEEE Int. Conf. on Computational Photography (ICCP), Cambridge, USA, 2013, pp. 1–8

[30] Vasu, S., Rajagopalan, A.: 'From local to global: edge profiles to camera motion in blurred images'. Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 2017, pp. 4447–4456

[31] Yang, L., Ji, H.: 'A variational EM framework with adaptive edge selection for blind motion deblurring'. Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition, Long Beach, USA, 2019, pp. 10167–10176

[32] Chakrabarti, A., Zickler, T., Freeman, W.T.: 'Analyzing spatially-varying blur'. 2010 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), San Francisco, SA, 2010, pp. 2512–2519

[33] Lin, J., Lin, X., Ji, X., *et al.*: 'Separable coded aperture for depth from a single image', *IEEE Signal Process. Lett.*, 2014, **21**, (12), pp. 1471–1475

[34] Hong, H., Park, I.K.: 'Single-image motion deblurring using adaptive anisotropic regularization', *Opt. Eng.*, 2010, **49**, (9), p. 097008

[35] Deshpande, A.M., Patnaik, S.: 'Single image motion deblurring: an accurate psf estimation and ringing reduction', *Optik-Int. J. Light Electron Opt.*, 2014, **125**, (14), pp. 3612–3618

[36] Shi, M., Liu, S.: 'PSF estimation via gradient cepstrum analysis for image deblurring in hybrid sensor network', *Int. J. Distrib. Sens. Netw.*, 2015, **11**, (10), p. 758034

[37] Mosleh, A., Green, P., Onzon, E., *et al.*: 'Camera intrinsic blur kernel estimation: a reliable framework'. Proc. of the IEEE Conf. on Computer Vision and pattern Recognition, Boston, USA, 2015, pp. 4961–4968

[38] Dong, F., Ma, Q.: 'Single image blind deblurring based on the fractional-order differential', *Comput. Math. Appl.*, 2019, **78**, (6), pp. 1960–1977

[39] Pan, L., Hartley, R., Liu, M., *et al.*: 'Phase-only image based kernel estimation for single image blind deblurring'. Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition, Long Beach, USA, 2019, pp. 6034–6043

[40] Chen, L.C., Papandreou, G., Kokkinos, I., *et al.*: 'Deeplab: semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFS', *IEEE Trans. Pattern Anal. Mach. Intell.*, 2018, **40**, (4), pp. 834–848

[41] Comaniciu, D., Meer, P.: 'Mean shift analysis and applications'. 1999 The Proc. of the Seventh IEEE Int. Conf. on Computer Vision, Kerkyra, Greece, 1999, vol. 2, pp. 1197–1203

[42] Tao, W., Jin, H., Zhang, Y.: 'Color image segmentation based on mean shift and normalized cuts', *IEEE Trans. Syst. Man Cybern. B (Cybern.)*, 2007, **37**, (5), pp. 1382–1389

[43] Elder, J.H., Zucker, S.W.: 'Local scale control for edge detection and blur estimation', *IEEE Trans. Pattern Anal. Mach. Intell.*, 1998, **20**, (7), pp. 699–716

[44] Saxena, A., Chung, S.H., Ng, A.Y.: 'Learning depth from single monocular images'. Advances in Neural Information Processing Systems, Vancouver, Canada, 2005, pp. 1161–1168

[45] Eigen, D., Puhrsch, C., Fergus, R.: 'Depth map prediction from a single image using a multi-scale deep network'. Advances in Neural Information Processing Systems, Montreal, Canada, 2014, pp. 2366–2374

[46] Liu, F., Shen, C., Lin, G.: 'Deep convolutional neural fields for depth estimation from a single image'. Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition, Boston, USA, 2015, pp. 5162–5170

[47] Liu, F., Shen, C., Lin, G., *et al.*: 'Learning depth from single monocular images using deep convolutional neural fields', *IEEE Trans. Pattern Anal. Mach. Intell.*, 2016, **38**, (10), pp. 2024–2039

[48] Yan, R., Shao, L.: 'Blind image blur estimation via deep learning', *IEEE Trans. Image Process.*, 2016, **25**, (4), pp. 1910–1921

[49] Namboodiri, V.P., Chaudhuri, S.: 'On defocus, diffusion and depth estimation', *Pattern Recognit. Lett.*, 2007, **28**, (3), pp. 311–319

[50] Namboodiri, V.P., Chaudhuri, S., Hadap, S.: 'Regularized depth from defocus'. 2008 15th IEEE Int. Conf. on Image Processing (ICIP 2008), San Diego, USA, 2008, pp. 1520–1523

[51] Srikakulapu, V., Kumar, H., Gupta, S., *et al.*: 'Depth estimation from single image using defocus and texture cues'. 2015 Fifth National Conf. on Computer Vision, Pattern Recognition, Image Processing and Graphics (NCVPRIPG), Patna, India, 2015, pp. 1–4

[52] Karaali, A., Jung, C.R.: 'Edge-based defocus blur estimation with adaptive scale selection', *IEEE Trans. Image Process.*, 2018, **27**, (3), pp. 1126–1137

[53] Canny, J.: 'A computational approach to edge detection', *IEEE Trans. Pattern Anal. Mach. Intell.*, 1986, **6**, pp. 679–698

[54] Levin, A., Lischinski, D., Weiss, Y.: 'A closed-form solution to natural image matting', *IEEE Trans. Pattern Anal. Mach. Intell.*, 2008, **30**, (2), pp. 228–242

[55] Kumar, H., Gupta, S., Venkatesh, K.: 'Hole correction in estimated depth map from single image using color uniformity principle'. 2017 22nd Int. Conf. on Digital Signal Processing (DSP), London, UK, 2017, pp. 1–5

[56] Kumar, H., Yadav, A.S., Gupta, S., *et al.*: 'Depth map estimation using defocus and motion cues', *IEEE Trans. Circuits Syst. Video Technol.*, 2019, **29**, (5), pp. 1365–1379

[57] Kumar, H., Gupta, S., Venkatesh, K.: 'Resolving focal plane ambiguity in depth map creation from defocus blur using chromatic aberration'. 2015 10th Int. Conf. on Information, Communications and Signal Processing (ICICS), Singapore, 2015, pp. 1–5

[58] Sellent, A., Favaro, P.: 'Which side of the focal plane are you on?'. 2014 IEEE Int. Conf. on Computational Photography (ICCP), Santa Clara, USA, 2014, pp. 1–8
[59] Kumar, H.: 'Reliable defocus map results', 2018. Available at https://goo.gl/6ZsYxS
[60] Himanshu-Kumar, S.G., Venkatesh, K.S.: 'PSF characterization results'. 2018. Available at goo.gl/6mFLLr
[61] Pan, J., Sun, D., Pfister, H., *et al.*: 'Blind image deblurring using dark channel prior'. Proc. of the IEEE Conf. on Computer Vision and Pattern, Recognition, Las Vegas, USA, 2016, pp. 1628–1636