

# ROBUST IMAGE SEGMENTATION FOR DEFOCUS BLUR BY SHARPNESS METRIC AND ALPHA MATTING

R. Pruthvika (M.Tech.)<sup>1</sup>

D. Nikitha (M.Tech, Assistant Professor)<sup>2</sup>

<sup>1</sup>Department of ECE, Vijay Rural Engineering College, Nizamabad, Telangana, 503003, INDIA

[sgspruthvika@gmail.com](mailto:sgspruthvika@gmail.com)<sup>1</sup> [desainikitha1@gmail.com](mailto:desainikitha1@gmail.com)<sup>2</sup>

*Abstract---When an image is captured by any optical imaging devices in that image, defocus blurs is the common undesirable thing. It is either enhance or inhibit the visual perception of an image scene. In different image processing operations like image restoration and object detection we required to segment the partially blurred image into blurred and non-blurred regions. We proposed sharpness metric in this paper based on Local Binary Patterns and a robust segmentation algorithm for the defocus blur. The proposed sharpness metric exploits the observation that the majority local image patches in blurred regions have considerably fewer of bounds native binary patterns compared with those in sharp regions. Mistreatment this metric, beside image matting and multiscale inference, we have a tendency to obtain high-quality sharpness maps. Tests on many partly blurred pictures were accustomed measure our blur segmentation rule and 6 comparator ways. The results show that our proposed method for defocus blur achieves comparative segmentation results with the state of the art and have the massive speed advantage over the others.*

**Keywords:** Digital image processing, LBP pattern, Sharpness, image restoration, object recognition

## 1. INTRODUCTION

Defocus estimation plays a crucial role in computer vision and computer graphics applications together with depth estimation, image quality assessment, image deblurring and refocusing. Different conventional methods have implemented on multiple images for defocussing. A set of images of the same scene is captured using multiple focus settings. Then

the defocus is measured during a implicit or explicit deblurring process. Recently, image pairs captured using coded aperture cameras [5] are used for better defocus blur measure and all-focused image recovery. However, these methods suffer from the occlusion problem and require the scene to be static, which limits their applications in practice.

Estimating defocus blur is a challenging task mainly because the corresponding PSFs are spatially varying and cannot be represented by any global descriptor. Indeed, spatially varying defocus PSFs for a given camera can be pre-calibrated and described typically through a simple model (e.g. Disc, Gaussian) that is characterized by a single parameter indicating its scale (radius, standard deviation, etc.) For an image, we call the 2D map of the scale parameter the defocus blur map, which indicates the level of local blur at each pixel (see an example in Fig. 1). The main purpose of this paper is to provide an automatic way of estimating a defocus blur map from a single input image.

Defocus blur map estimation has several potential applications. For example, it can be employed to detect and segment in-focus subjects from the out-of-focus background, helping a photo editor to edit the subject of interest or the background, separately. Besides that, since defocus blur level is intimately related to the depth of the scene, a blur map also provides important information for depth estimation. The computation of depth information typically requires two photos of the same scene taken at the same time, but from slightly different vantage points, i.e. a stereo pair [6]. However, in most cases only one image is available. A blur map allows one to reconstruct a 3D scene from a single photograph as

long as the camera settings (focal length, aperture settings, etc.) are known. For image restoration applications, if both the defocus PSF calibration and blur map estimation are made, we can reconstruct an all-in-focus image through a non-blind spatially varying deblurring process.

This method locally selects the best PSF by evaluating its deconvolution errors. It requires a specially designed aperture filter for the camera, which strongly limits its domain of application. Instead of estimating the optimal blur scale in the continuous domain, it can only identify the most likely candidate from a finite number of calibrated PSFs with somewhat limited accuracy. Chakrabarti et al. suggested a method estimating the likelihood function of a given candidate PSF based on local frequency component analysis without deconvolution [8]. In their paper the method is applied to detect simple motion blur, but it can also be employed for defocus blur identification. Again it can only detect optimal PSFs from a finite number of candidates.

## 2. LITERATURE SURVEY

**Fergus et al. [1]** proposed a method that Camera shakes during exposure May causes to objectionable image blur and damage photographs. Conventional blind deconvolution techniques rarely assume frequency-domain parameters on images for the motion path while camera shake. Real camera motions can follow up the convoluted way and spatial domain prior can better retains visually image properties. They introduced a method to remove the effects of camera shake from blurred images. This method assumes that a uniform camera blur over the image and also negligible in-plane camera rotation. In order to calculate the blur from the camera shake, the person must specify an image region without saturation effects. They showed results for a wide variety of digital photographs which are taken from personal photo collections.

**Bae and Durand[2]** presented the image processing technique in which defocus magnification is used to perform blur estimation. To maximize defocus blur caused by lens aperture by taking a single image then estimate the size of blur kernel at edges and further they spread this technique to the whole image. In this

approach multi scale edge detector is used and model fitting that obtain the size of bur propagate the blur measure by assuming that blurriness is smooth where intensity and color are approximately similar. Using defocus map, they enhance the existing blurriness, which means that blur the blurry regions and keeps the sharp regions sharp. In comparison to other methods more difficult issues arises such as depth from defocus, so this proposed method do not need precise depth estimation and do not need to disambiguate texture less regions. The method models changes in energy at all frequencies with blur and not just very high frequencies (edges).

**Levin et al. [3]** evaluate Blind deconvolution algorithm which is the restoration of a sharp version from a blurred image when the blur kernel is not known. Most algorithms have dramatic progress; still many aspects of the problem remain challenging and difficult to understand. The goal of this method is to analyze and evaluate blind deconvolution algorithms both theoretically as well as experimentally. They had also discussed the failure of the MAP approach.

**Kee et al. [4]**, discussed that noticeable blur is generated due to the optical system of the camera, also with professional lenses. They introduce method to measure the blur kernel densely over the image and also across multiple aperture and zoom settings. It shown that the blur kernel can have a non-negligible spread, even with top-of-the-line equipment. the spatial changes are not gradually symmetric and not even left-right symmetric .In this method two models of the optical blur are developed and compared both having advantages respectively. It is shown that the model find accurate blur kernels that can be used to restore images. They demonstrated that they can produce images that are more uniformly sharp then those images which produced with spatially-invariant deblurring technique.

**Tai and brown in [5]** As Image defocus estimation is useful for several applications including deblurring, blur enlargement, measuring image quality, and depth of field segmentation. They proposed a simple effective approach for estimating a defocus blur map based on the relationship of the

contrast to the image gradient in a local image region and call this relationship the local contrast prior. The advantage of this approach is that it does not need filter banks or frequency decomposition of the input image; infact it only needs to compare local gradient profiles with the local contrast. They discuss the idea behind the local contrast prior and shown its results on a variety of experiments. And it's found that for natural in-focus images, this distribution follows a similar pattern. They verified this distribution by plotting the distribution of the LC in images suffered from different type of degradation. This prior is useful in estimating defocus blur, in segmenting in focus regions from depth-of-field image and in ranking image quality.

### 3. PROPOSED METHOD

Local Binary Patterns (LBP) have been successful for computer vision problems such as texture segmentation, face recognition, background subtraction and recognition of 3D textured surfaces [36]. The LBP code of a pixel  $(x_c, y_c)$  is defined as

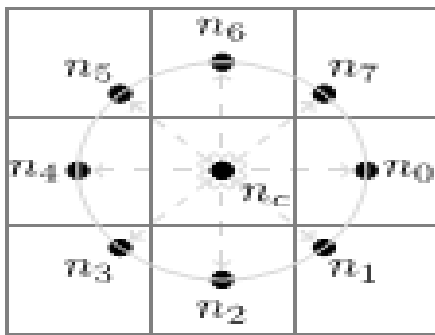


Fig. 1. 8-bit LBP with  $p=8, R=1$

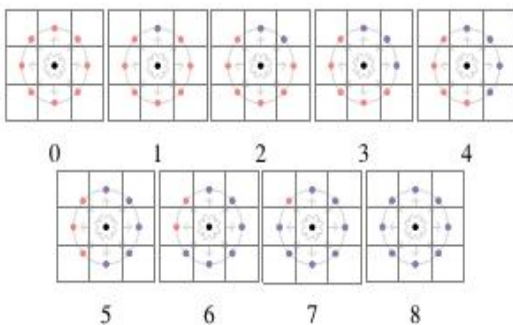


Fig. 2. The uniform rotationally invariant LBP.

$$LBP_{P,R}(x_c, y_c) = \sum_{p=0}^{P-1} S(n_p - n_c) \times 2^p$$

$$\text{With } S(x) = \begin{cases} 1 & |x| \geq T_{LBP} \\ 0 & |x| < T_{LBP} \end{cases} \quad (3.1)$$

Where  $c$  is the intensity of the central pixel  $(x_c, y_c)$ ,  $n_p$  corresponds to the intensities of the  $P$  neighboring pixels located on a circle of radius  $R$  centered at  $n_c$ , and  $T_{LBP} > 0$  is a small, positive threshold in order to achieve robustness for flat image regions as in [19]. Figure 4 shows the locations of the neighboring pixels  $n_p$  for  $P=8$  and  $R=1$ . In general, the point's  $n_p$  do not fall in the center of image pixels, so the intensity of  $n_p$  is obtained with bilinear interpolation. A rotation invariant version of LBP can be achieved by performing the circular bitwise right shift that minimizes the value of the LBP code when it is interpreted as a binary number.

In this way, number of unique patterns are reduced to 36. Ojala et al. found that not all rotation invariant patterns sustain rotation equally well [34], and so proposed using only uniform patterns which are a subset of the rotation invariant patterns. A pattern is uniform if the circular sequence of bits contains no more than two transitions from one to zero, or zero to one. The non-uniform patterns are then all treated as one single pattern. This further reduces the number of unique patterns to 10 (for 8-bit LBP), that is, 9 uniform patterns, and the category of non-uniform patterns. The uniform patterns are shown in Figure 5. In this figure, neighboring pixels are colored blue if their intensity difference from centre pixel is larger than  $T_{LBP}$ , and we say that it has been "triggered", otherwise, the neighbours are colored red.

Our proposed sharpness metric exploits these observations:

$$m_{LBP} = \frac{1}{N} \sum_{i=6}^9 n(LBP_{8,1}^{riu2} i) \quad (3.2)$$

Where  $n(LBP_{8,1}^{riu2} i)$  is the number of rotation invariant uniform 8-bit LBP pattern of

type  $i$ , and  $N$  is the total number of pixels in the selected local region which serves to normalize the metric so that  $m_{LBP} \in [0,1]$ . One of the advantages of measuring sharpness in the LBP domain is that LBP features are robust to monotonic illumination changes which occur frequently in natural images. The threshold TLBP in Equation (3.1) controls the proposed metric's sensitivity to sharpness.

There is a sharp fall-off between  $\sigma = 0.2$  and  $\sigma = 1.0$  which makes the intersection of response value range of sharp and blur much smaller than the other metrics. When  $\sigma$  approaches 2, responses for all patches shrinks to zero which facilitates segmentation of blurred and sharp regions by simple thresholding. Moreover, almost smooth region elicit a much higher response than smooth region compared with the other metrics. Finally, the metric response is nearly monotonic, decreasing with increasing blur, which should allow such regions to be distinguished with greater accuracy and consistency.

### NEW BLURSEGMENTATIONALGORITHM

This section presents our algorithm for segmenting blurred/sharp regions with our LBP-based sharpness metric it is summarized in Figure 12. The algorithm has four main steps: multi-scale sharpness map generation, alpha matting initialization, alpha map computation, and multi-scale sharpness inference.

#### A. Multi-Scale Sharpness Map Generation

In the first step, multi-scale sharpness maps are generated using  $m_{LBP}$ . The sharpness metric is computed for a local patch about each image pixel. Sharpness maps are constructed at three scales where scale refers to local patch size. By using an integral image [50], sharpness maps may be computed in constant time per pixel for a fixed  $P$  and  $R$ .

#### B. Alpha Matting Initialization

Alpha matting is the process of decomposing an image into foreground and background. The image formation model can be expressed as

$$I(x, y) = \alpha_{x,y}F(x, y) + (1 - \alpha_{x,y})B(x, y) \quad (3.3)$$

Where the alpha matte,  $\alpha_{x,y}$ , is the opacity value on pixel position  $(x,y)$ . It can be interpreted as the confidence that a pixel is in the foreground. Typically, alpha matting requires a user to interactively mark known foreground and background pixels, initializing those pixels with  $\alpha = 1$  and  $\alpha = 0$ , respectively. Interpreting "foreground" as "sharp" and background as "blurred", we initialized the alpha matting process automatically by applying a double threshold to the sharpness maps computed in the previous step to produce an initial value of  $\alpha$  for each pixel.

$$mask^s(x, y) = \begin{cases} 1, & \text{if } m_{LBP}(x, y) > T_{m_1} \\ 0, & \text{if } m_{LBP}(x, y) < T_{m_1} \\ m_{LBP}(x, y), & \text{otherwise.} \end{cases} \quad (3.4)$$

Where  $s$  indexes the scale, that is, masks  $(x,y)$  is the initial  $\alpha$ -map at the  $s$ -th scale.

#### C. Alpha Map Computation

The  $\alpha$ -map was solved by minimizing the following cost function as proposed by Levin

$$E(\alpha) = \alpha^T L \alpha + \lambda (\alpha - \hat{\alpha})^T (\alpha - \hat{\alpha}) \quad (3.5)$$

Where  $\alpha$  is the vectorized  $\alpha$ -map,  $\hat{\alpha} = mask^i(x, y)$  is one of the vectorized initialization alpha maps from the previous step, and  $L$  is the matting Laplacian matrix. The first term is the regulation term that ensures smoothness, and the second term is the data fitting term that encourages similarity to  $\hat{\alpha}$ . For more details on Equation 3.5, The final alpha map at each scale is denoted as  $\alpha_s, s=1,2,3$

#### D. Multi-Scale Inference

After determining the alpha map at three different scales, a multi-scale graphical model was adopted to make the final decision. The total energy on the graphical model is expressed as

$$E(h) = \sum_{s=1}^3 \sum_i |h_i^s - \hat{h}_i^s| + \beta \left( \sum_{s=1}^3 \sum_i \sum_{j \in N_i^s} |h_i^s - h_j^s| + \sum_{s=1}^2 \sum_i |h_i^s - h_i^{s+1}| \right) \quad (3.6)$$

Where  $\hat{h}_i^s = \alpha_i^s$  is the alpha map for scale sat pixel location  $i$  that was computed in the previous step, and  $h_i^s$  is the sharpness to be inferred. The first term on the right hand side is the unary term which is the cost of assigning sharpness value  $h_i^s$  to pixel  $i$  in scale  $s$ . The second is the pairwise term which enforces smoothness in the same scale and across different scales. The weight  $\beta$  regulates the relative importance of these two terms. Optimization of Equation 3.6 was performed using loopy belief propagation.

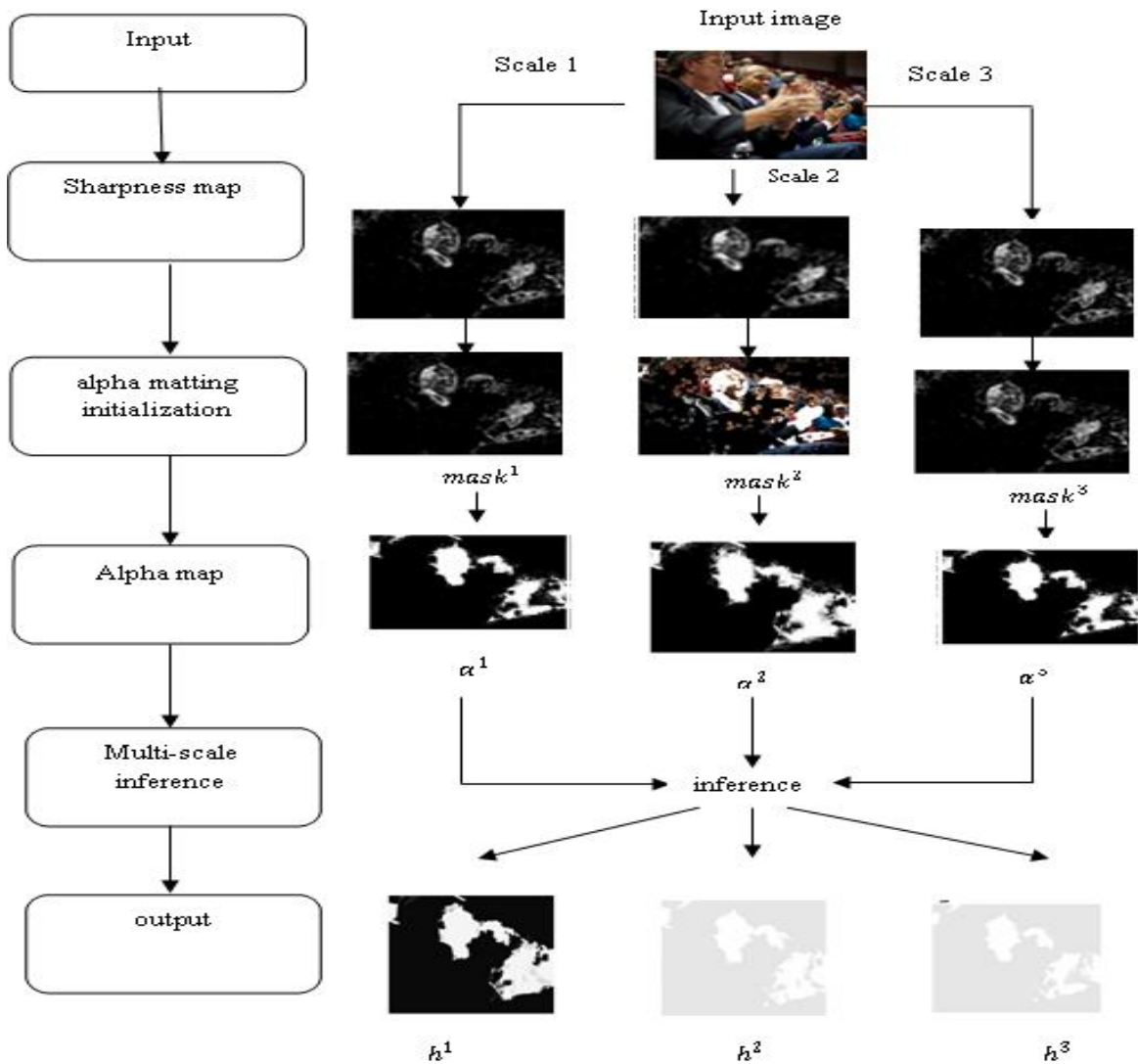


Fig. 3. Our blur segmentation algorithm. The main steps are shown on the left; the right shows each image generated and its role in the algorithm. The output of the algorithm is  $h^1$ .



The output of the algorithm is  $h^3$  which is the inferred sharpness map at the largest scale. This is a grayscale image, where higher intensity indicates greater sharpness.

#### 4. SIMULATION RESULTS



Figure: 1 Segmentation with LBP



Figure: 2 Segmentation with LLBP

#### 5. CONCLUSION

We successfully proposed easy and effective sharpness metric for the segmentation of partially blurred image into blurred and non blurred regions. This metric is mostly based on the distribution of uniform LBP patterns in blurred and non blurred regions. The direct use of some sharpness measure based on the sparse representation gives the comparative results to our proposed method. By integration the metric into a multiscale information propagation frame work, it is able compare results

with the progressively as well as with state of art. We've shown that the algorithm's performance is maintained once victimization an mechanically associated adaptively taken threshold Tseg. Our sharpness metric measures the amount of sure LBP patterns in the native neighborhood so is with efficiency enforced by integral pictures. If combined with period matting algorithms, such as GPU implementations of world matting [18], our methodology would have required high speed advantage over the other defocus segmentation algorithms.

#### REFERENCES

- [1] R. Achanta, S. Hemami, F. Estrada, and S. Susstrunk, "Frequency-tuned salient region detection," in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2009, pp. 1597–1604.
- [2] H.-M. Adorf, "Towards HST restoration with a space-variant PSF, cosmic rays and other missing data," in Proc. Restoration HST Images Spectra-II, vol. 1. 1994, pp. 72–78.
- [3] T. Ahonen, A. Hadid, and M. Pietikäinen, "Face description with local binary patterns: Application to face recognition," IEEE Trans. Pattern Anal. Mach. Intell., vol. 28, no. 12, pp. 2037–2041, Dec. 2006.
- [4] S. Bae and F. Durand, "Defocus magnification," Comput. Graph. Forum, vol. 26, no. 3, pp. 571–579, 2007.
- [5] K. Bahrami, A. C. Kot, and J. Fan, "A novel approach for partial blur detection and segmentation," in Proc. IEEE Int. Conf. Multimedia Expo (ICME), Jul. 2013, pp. 1–6.
- [6] J. Bardsley, S. Jefferies, J. Nagy, and R. Plemmons, "A computational method for the restoration of images with an unknown, spatially-

varying blur,” *Opt. Exp.*, vol. 14, no. 5, pp. 1767–1782, 2006.

[7] A. Buades, B. Coll, and J.-M. Morel, “A non-local algorithm for image denoising,” in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, vol. 2, Jun. 2005, pp. 60–65.

[8] G. J. Burton and I. R. Moorhead, “Color and spatial structure in natural scenes,” *Appl. Opt.*, vol. 26, no. 1, pp. 157–170, 1987.

[9] A. Chakrabarti, T. Zickler, and W. T. Freeman, “Analyzing spatially-varying blur,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2010, pp. 2512–2519.

[10] T. S. Cho, “Motion blur removal from photographs,” Ph.D. dissertation, Dept. Elect. Eng. Comput. Sci., Massachusetts Inst. Technol., Cambridge, MA, USA, 2010.

[11] F. Couzinie-Devy, J. Sun, K. Alahari, and J. Ponce, “Learning to estimate and remove non-uniform image blur,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2013, pp. 1075–1082.

[12] S. Dai and Y. Wu, “Removing partial blur in a single image,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2009, pp. 2544–2551.