Just Noticeable Defocus Blur Detection and Estimation

Jianping Shi[†]   Li Xu[‡]   Jiaya Jia[†]

[†] The Chinese University of Hong Kong

[‡] Image & Visual Computing Lab, Lenovo R&T

jpshi@cse.cuhk.edu.hk   xulihk@lenovo.com   leojia@cse.cuhk.edu.hk

Abstract

We tackle a fundamental problem to detect and estimate just noticeable blur (JNB) caused by defocus that spans a small number of pixels in images. This type of blur is common during photo taking. Although it is not strong, the slight edge blurriness contains informative clues related to depth. We found existing blur descriptors based on local information cannot distinguish this type of small blur reliably from unblurred structures. We propose a simple yet effective blur feature via sparse representation and image decomposition. It directly establishes correspondence between sparse edge representation and blur strength estimation. Extensive experiments manifest the generality and robustness of this feature.

1. Introduction

Photos awaken our pictorial memories. A good photo generally contains clear and sharp objects that are important. With the prevalence of high-resolution imaging sensors, blurriness and its spatial change become perceivable. Our experiments show that a typical 13 mega-pixel mobile phone camera could produce blur spanning up to 5-8 pixels even we put these points in focus. We name the commonly occurred small defocus blur *just noticeable blur* (JNB), which is formally defined as blur spanning about 3-9 pixels and losing a quantitatively insignificant level of structures.

JNB commonly exists in images. It actually gives us useful information to understand the scene. A typical example is shown in Fig. 1, where sight blurriness implies foreground and the salient object we should notice.

Impact These facts motivate us to study the new JNB detection and estimation problem. It finds loads of potentially interesting applications. For example, it could avail computer-aided image quality assessment. It can be used to manipulate images and generate special effects, including background blur magnification and partial image de-
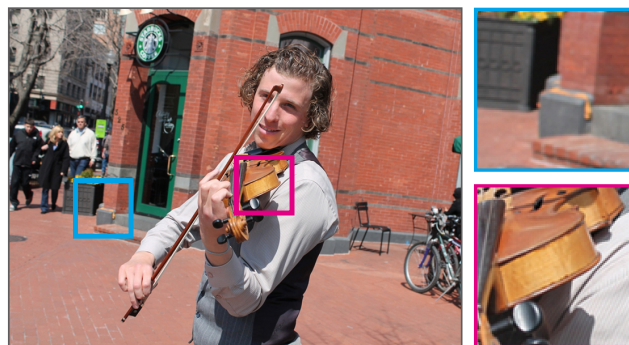

Figure 1. This image found over internet was captured with aperture size f/5.6 and exposure time 1/500s. When this supposedly clear image is viewed in its original resolution, slight blurriness can still be noticed. It is a general phenomenon.

blurring.

With regard to blur estimation, our work contributes in spatially-varying blur strength estimation in the pixel level. The relative blurriness estimation actually carries the scene depth estimation, which avails many tasks. Besides, depth itself forms vital data to help object recognition and classification. Our work does not need to capture extra depth data. As long as slight blur exists, a single image is enough to infer usable depth. We will show several results later.

Challenges Though valuable in research, this JNB problem faces its unique challenges compared to traditional blur estimation. It is not feasible for existing methods to trivially address it. We explain it with respect to two possible solutions.

First, advanced deblurring algorithms [5, 17, 25, 12, 28, 13] can estimate blur kernels and deconvolve input images. But it is still difficult to handle spatially varying blur, especially when it is not caused by camera motion. Existing non-uniform deblurring methods assume camera motion models and are generally computationally expensive even for a small-resolution image.

Second, there are a limited number of methods to handle explicit blur detection [8, 14, 19, 24, 31, 30, 29, 22]. Nearly
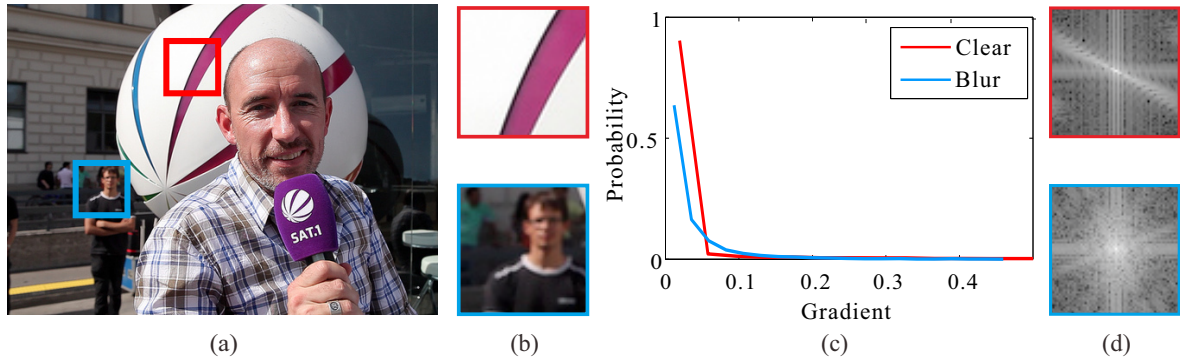
Figure 2. A natural image example. (a) An image with clear and blurry regions in red and blue respectively. (b) Close-ups. (c) Gradient distributions for these regions. The clear region has generally smaller gradients. (d) Spectrum map after Fourier transform on logarithmic intensities. The JNB patch surprisingly has more high frequency components than the clear one.

all of them make use of local patch information. In the next section, we discuss that with only local information, the blur features may lack basic ability to differentiate between JNB and clear pixels.

Our Method We show a new direction to understand small image blur via sparse representation based on external data. Specifically, we found that when decomposing local image patches into dictionary atoms in an additive manner, clear and JNB dictionaries show quantitatively and visually different results. The diverged effect manifests that dictionary atoms can characterize structure in just noticeable blur images, thus amplifying the inherent difference between slight blur and clear regions. Based on it, we propose our simple but expressive JNB feature. It is verified on image data in accordance with our finding.

Our main contributions are as follows. First, we introduce a new framework for small blur identification. Second, we propose a sparsity-based feature, which can produce usable results in blur strength estimation. Moreover, we verify our approach on two image blur detection datasets with one containing all JNB images. We also apply our results to applications of image deblurring, image refocus, and relative depth estimation, to demonstrate its potential.

2. Existing Blur Descriptors

To know where and how strong the blur happens, a number of solutions were proposed in this field. Different from traditional camera motion blur estimation [5, 27, 9, 26, 18] where blur is significant and anisotropic, just noticeable blur mainly deals with slight defocus blur. Starting from Elder and Zucker [8], who utilized the first and second order gradient information for local blur estimation, various methods have been proposed along this line to detect and estimate defocus blur.

Defocus blur analysis can be generally categorized into gradient and frequency based solutions. Gradient meth-

ods [8, 14, 19, 24, 31] exploit the fact that blur suppresses gradients. Thus the gradient distribution in a clear region should have more heavy-tail components and be flat to avoid strong peaks. In the frequency point of view [19, 30, 29], blur attenuates high frequency components and increases low frequency ones.

However, all above approaches were designed to estimate defocus blur for narrow depth of field images or images containing large blur, where the difference of edge sharpness in in-focus and out-of-focus regions are significant. They do not work similarly well at JNB level. The reason is that these descriptors collect local information. On relatively small regions, local information may not be stable enough for descriptor construction. So the local statistics on sharp and JNB regions are hardly distinguishable, making their classification challenging.

A natural image example in Fig. 2 illustrates this problem. The red patch is clear and the blue one is blurred. Although both patches contain edge information, their local gradient distributions contradict the common sense that clear patch should contain more large-magnitude gradients shown in (c). The local Fourier spectra reveal similar ambiguity in (d). It is due to the diverse complexity of latent structures in the two patches, where the blurred region here has more edges than the clear one. Since such ambiguity regions appear commonly in JNB images, traditional blur features could fail on these images. Note local gradient distribution features were used in [8, 19, 14, 31] and local frequency based metrics include slope of average power spectrum [19, 22], wavelet response [29] and Gabor filter. For these methods, only using information within a local region is not enough for JNB. Other methods, e.g., matting [7, 6] and local appearance model maximum saturation [19, 23], encounter similar issues.

To solve our JNB problem, external data should be referred to. In our approach, we extract a set of templates containing rich structural and statistical information from

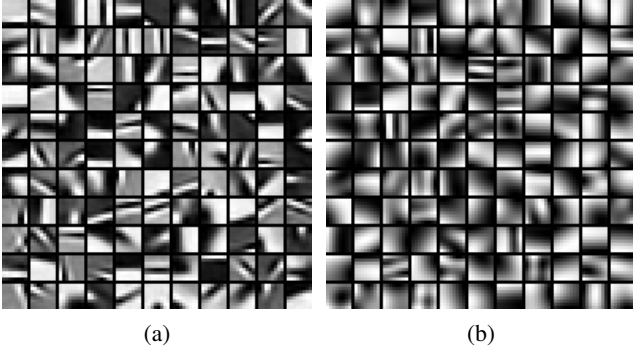|	
(a)	(b)

Figure 3. Visualizing the dictionary for natural image sparse representation. (a) Clear natural image dictionary. (b) Dictionary trained on a JNB image set.

external data. Then the templates are used to evaluate the possibility of existence of JNB. We provide comparison and analysis in Section 4. Hardware solutions for defocus analysis [15, 11, 3] are out of the scope of this paper. Optical aberrations correction [21] is also related to estimating small blur. But it handles images in special situations.

3. Our Blur Detection Features

Our human visual system has the ability to distinguish blur from clear regions. It implies a possible foundation to construct an automatic system based on seeing a plenty of natural image examples in clear or blurry states. We follow the data-driven strategy via sparsity based natural image decomposition on extra data.

3.1. Sparse Dictionary

Sparse representation [1, 20] commonly works as follows. Given a set of n signals $Y = \{y_1, \ldots, y_n\} \in \mathbb{R}^{d \times n}$, each signal y_i can be represented by a sparse number of dictionary atoms as

$$\min_{x_i} \|y_i - Dx_i\|_2^2 \qquad s.t. \ \|x_i\|_0 \le k, \qquad (1)$$

where $D \in \mathbb{R}^{d \times m}$ is an over-complete dictionary capturing all atomic information lying in Y; x_i is the coefficient to reconstruct y_i. The ℓ^2 norm in the objective function makes representation error small, so that the sparsely recovered signal Dx_i is close to the original one. The ℓ^0 norm for x_i in the constraint induces sparsity, which allows a small number of dictionary atoms in D to reconstruct y_i. Basically, sparse representation is to use dictionary atoms to capture elementary information.

In natural image decomposition, we collect overlapped image patches as input. Each image patch is vectorized as y_i in Eq. (1). A dictionary D is trained on all image patches. Based on the constructed dictionary, each image patch is decomposed into a few atoms together with

their non-zero coefficients, forming the reconstructed feature Dx_i via Eq. (1).

3.2. Clear and JNB Dictionaries

As sparse representation can decompose each image patch into several elementary dictionary atoms, do these atoms represent clear and JNB input differently? We conduct experiments to evaluate it. Our method extracts image patches each with size 8×8, forming a 64D vector. Then we train a natural image dictionary with 128 atoms using clear images following the procedure of [1]. The resulting dictionary is illustrated in Fig. 3(a). Each atom is an edge-like component, reasonably representing natural image structure.

In the meantime, we use the same procedure to train a dictionary on images blurred slightly by Gaussian with $\sigma = 2$. The corresponding image dictionary is shown in Fig. 3(b), which presents obviously different structures containing nearly no sharp patterns.

The contrast between dictionaries shows how blur, even slightly, influences the fundamental atoms in image decomposition. It also manifests that JNB and clear dictionaries are not interchangeable when performing sparse patch reconstruction.

A Naive Method So a straightforward strategy is to learn the dictionary for each set of patches. If it contains smooth elements, the patches are possibly blurred ones. This scheme has a few blatant limitations. It assumes all patches in an image are either blurred or clear in order to learn dictionaries correctly. It is also costly to learn dictionaries again and again for different input data. Finally, it may be possible to tell whether the set of patches are blurred or not; but it is difficult to estimate blur size in terms of blurring strength.

Our proposed method is different from this naive scheme. It only needs to construct one dictionary via offline training. It is then used afterwards to classify new patches individually without assuming that all or a group of patches are in the same class. Blur degree estimation is achievable as well in this simple framework.

3.3. Sparsity JNB Feature

Considering inherent discrepancy between the two types of dictionaries, we propose a new blur metric. We first learn a blur dictionary D following Eq. (1). The dictionary D is trained over 100,000 patches randomly cropped from 1,000 natural images blurred by the Gaussian kernel of $\sigma = 2$. We have tried other choices, including increasing and decreasing Gaussian variance, and replacing Gaussian with other types of blur. It is found that this configuration is sufficient for our feature construction. The maximum number k cor-

(a) Clear input, 59 atoms   (b) Gaussian blur σ=0.6, 46 atoms   (c) Gaussian blur σ=1, 33 atoms

(d) Gaussian blur σ=1.2, 24 atoms   (e) Gaussian blur σ=1.5, 20 atoms   (f) Gaussian blur σ=2, 19 atoms
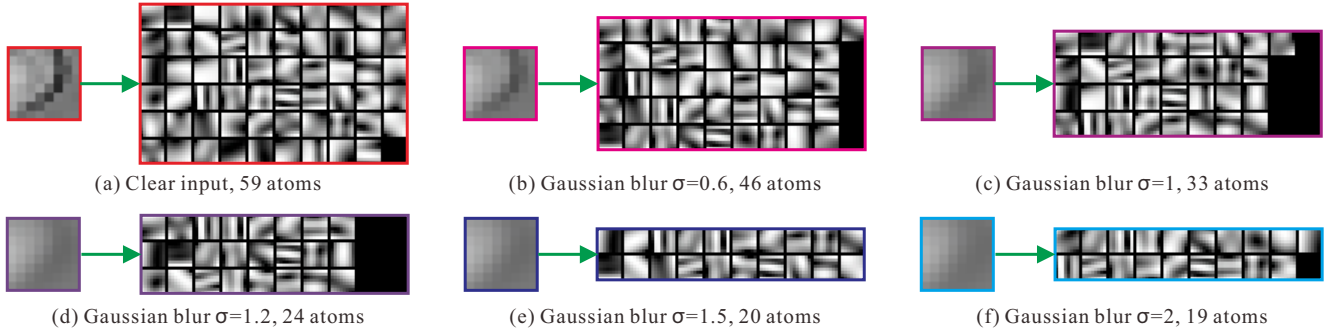
Figure 4. Sparsity features for different blur degrees. As blurriness gets strong, variation of patches decreases. The number of atoms used to represent images correspondingly drops.

responding to the used dictionary atoms is set to 5 in patch decomposition and the total dictionary size is 128.

After D is learned, it is applied to all image patches, both JNB and clear, for blur identification. For each new patch input y_i, we use another spare representation to decompose it into basic atoms. It is expressed as

$$\min_{x_i} \|x_i\|_1 \quad s.t. \ \|y_i - Dx_i\|_2 \le \epsilon, \tag{2}$$

where ϵ is a constant (0.07 in our experiment). Different from the traditional form that selects a relatively large ϵ to resist noise and outliers, we set this value small to make the resulting averaged PSNR between the original and reconstructed patches over 50. This special setting is based on the consideration that detail-level structural information is central to image blur assessment in human perception.

The output atoms and corresponding coefficients reflect whether the input is blurred or not and how strong it is. We build our sparsity feature f_a for input y_i as the number of non-zero elements in x_i, expressed as

$$f_a = \|x_i\|_0. \tag{3}$$

Note these patches should not be flat in color in order to avoid classification ambiguity. Actually it does not matter that much if we label one flat patch as blur or clear for many applications such as deblurring and blur magnification.

Understanding and Verification   Why is the number of atoms to decompose each patch essential in blur identification? The rationale is that sharp edges have more high-frequency sharp components than the JNB regions. In frequency decomposition, to fully encode a high-frequency edge, many small low-frequency components need to be recorded. They are added with their respective weights.

Taking a 1D signal as example, to represent the clear box signal well, many smoothed atoms need to be added together with their respective coefficients, as illustrated in the first row of Fig. 5. Contrarily, a blurred signal requires
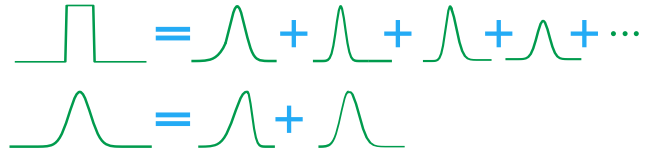

Figure 5. Major atom difference when using the blur dictionary to represent inputs in 1D. The first-row clear input signal needs many blurred atoms in representation due to structure divergence. The bottom blurred signal, contrarily, can be reconstructed sparsely.

much less atoms for optimal reconstruction, illustrated in the second row of Fig. 5.

We choose to use the blurry-image dictionary but not the clear one for image reconstruction. It is because in the blurry-image dictionary, there is barely sharp component because it is established from blurry training data. So to represent a clear patch using blurry bases, a large number of atoms must be used. This situation differs greatly from that to represent a blur patch.

It is notable using the clear-image dictionary does not achieve similar effect, as there inevitably exist flat or smooth regions in clear images. They make the dictionary have comparable ability to reconstruct blurry and clear images using similar numbers of atoms.

Our blurry-image dictionary D is shown in Fig. 3(b). A clear image patch and those blurred by 5 different kernels, are decomposed into atoms contained in Fig. 4. Since the blurred dictionary mainly contains various smooth structures, to represent a clear input, the used dictionary atoms are almost as many as the patch dimension. With blur degree increasing, edges become less sharp. Thus a small number of elements can already form the basis. The trend can be statistically obtained: more atoms generally correspond to less blurry patches. We thus use this clue as our small blur indicator.

We use the blurred images to create the dictionary instead of blurring the clear dictionary directly because the clear-image dictionary in Fig. 3(a) contains many edge-like structures with three or more major value clusters, while
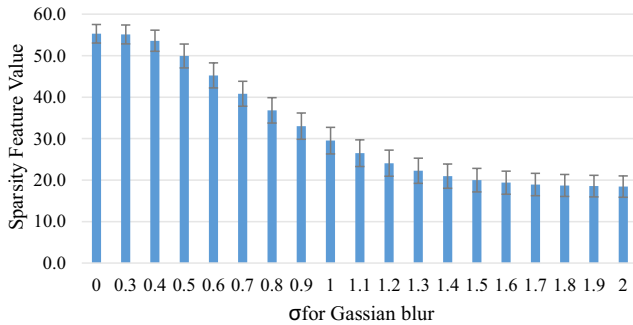
660

Figure 6. Sparsity values v.s. blur strength. Height of each bar indicates the average sparsity value corresponding to a particular blur strength measured by blur standard deviation. The short gray lines represent the standard deviation. It statistically proves that our sparsity measure is strongly and stably correlated with blur strength.

most blur dictionary atoms are smooth boundaries with two main value sets in (b). The atoms generated by blurring the clear dictionary are still different from those in (b) and do not work similarly well in our experiments. Our current blur dictionary thus captures the elementary information to represent JNB images.

Final Blur Strength Measure   We verify the generality of the phenomenon that less used dictionary atoms correspond to stronger blurriness. We capture about 200 images with different levels of defocus blur, where their blurriness can be roughly matched with the Gaussian kernel of variance ranging from 0.3 to 2, in a controlled laboratory environment. Along with natural images blurred by Gaussian defocus kernels, in total we collect 5,000,000 sampled patches variously smoothed in different degrees. We remove flat regions to avoid ambiguity.

Fig. 6 lists our sparsity values f_a and their corresponding standard derivation under different blurriness levels. The sparsity decreases as blur increases. Moreover, the sparsity feature values for each particular blur level are rather consistent under a small standard deviation. This manifests the effectiveness and usability of the sparsity blur feature.

The statistically stable correspondence between blur standard deviation σ and sparsity feature values f enables us to fit a logistic regression function as

$$f = \frac{a}{1 + \exp(b\sigma + c)} + d, \tag{4}$$

where a, b, c and d are the fitted variables with corresponding values 39.49, 4.535, -3.538, and 18.53 respectively. Eq. (4) allows our system to even estimate the degree of blurriness for each patch even if it is small, and empowers spatial-varying blur strength estimation.

4. Experiments and Comparisons

Our method does not handle flat regions due to their inherent ambiguity. As aforementioned, it does no matter to determine them as sharp or blur. We simply mask them out to indicate uncertain pixels. We fill in these holes using closed form matting [16]. The final blur map is bilateral filtered to remove noise and preserve sharp boundaries.

We provide an example in Fig. 7. (a) is the input image, and (b) is its corresponding ground-truth mask to indicate the clear region with respect to the just noticeable blur region. Our raw feature in (j) is already powerful enough to classify the background toy as blurry. The final map in (k) is perceptually more reasonable. Given the color input and the blur map, we apply a graph-cut algorithm to label the blur region in (l). It is close to the ground-truth.

The JNB detection task is actually not easy. We compare our sparsity based method with other blur estimation approaches including [2, 19, 4, 31, 30, 23, 22] in (c)-(i). We either implement their methods or directly use public codes for comparison. Our method outperforms others. All the results are normalized to [0, 1]. In particular, the method of [4] (result shown in (c)) is for directional blur via local Fourier transform from a set of candidates. It shows directly applying local FFT is not that discriminative for small blur.

The methods of (d) and (e) estimate blur at strong edge regions, and then propagate them to get final results [2, 31]. These methods rely on accurate estimate of edge blur. The Gaussian kernel and noise-free assumptions could be unsuitable when dealing with our problem. The background toy in (d) and the star region in (e) are not correctly labeled. The result of [30] is shown in (f). It provides reasonable estimate but computation is time consuming. Also graph-cut optimization makes the final result only have a few discrete values.

Finally, the methods of [19, 22] with results shown in (g) and (h) generate local blur estimates based on local gradient or frequency statistics. They do not produce significant difference between clear and JNB regions. So there are clear errors in the results. A decomposition based approach is also employed via local SVD in [23] with the result shown in (i). Without additional training data, this method does not perform similarly well on textured regions such as the head of the front toy. Our method is specially designed to detect JNB. The final result after filtering in (k) is visually compelling. The generated binary map in (l) can be regarded as direct foreground segmentation.

To further evaluate our approach, we collect 8 JNB images, where blurry regions are masked out as ground-truth. On these data, we compare our sparsity based feature with other blur estimation approaches including [2, 19, 4, 31, 30, 23, 22]. Other blur analysis methods do not generate blur maps and thus are not included. We show quantitatively comparison on our data via *precision-recall* (PR) in
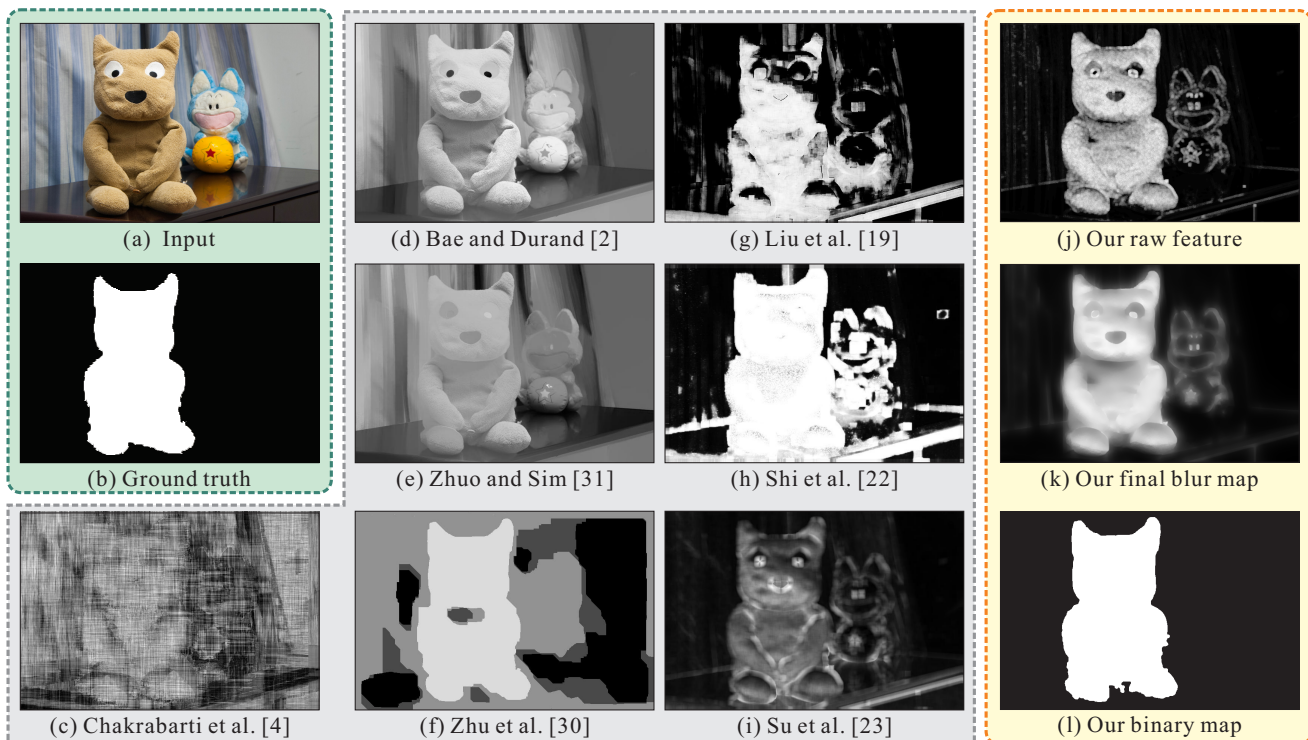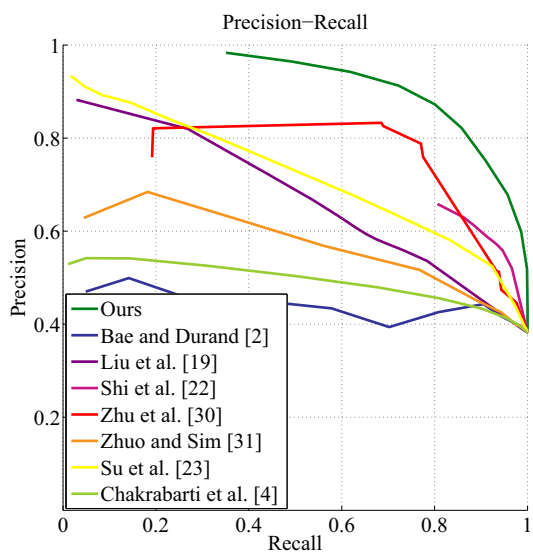
(a) Input
(b) Ground truth
(c) Chakrabarti et al. [4]
(d) Bae and Durand [2]
(e) Zhuo and Sim [31]
(f) Zhu et al. [30]
(g) Liu et al. [19]
(h) Shi et al. [22]
(i) Su et al. [23]
(j) Our raw feature
(k) Our final blur map
(l) Our binary map

Figure 7. Blur map comparison.

Figure 8. Precision-recall comparison on the JNB data.

Figure 9. Precision-recall comparison on the blur dataset [22].

Fig. 8. The final maps to calculate the PR curve are chosen as binary at possible thresholds within range [0, 255]. Our method achieves the highest precision in the entire recall range in [0, 1].

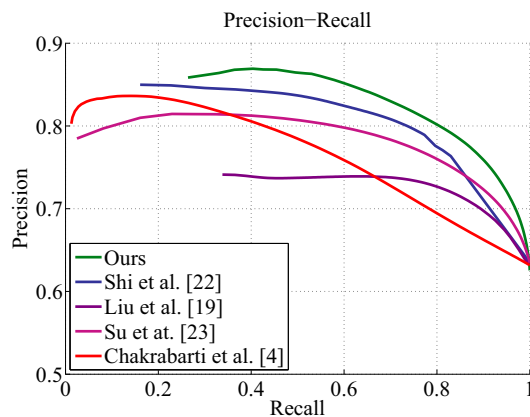We also test our method on images proposed in [22] where the input image are not restricted to just noticable blur. Based on the results in Fig. 9, our method surprisingly outperforms others [19, 4, 23, 22] in the full range, which indicates that our method is also able to handle general blur in most cases. To further investigate the latent information, we manually select 62 images out of the 1000. All images contain not-so-obvious out-of-focus blur and background still has visible structures. These data are difficult to handle in general. Our results are shown in Fig. 10.

5. Applications

In what follows, we apply our blur map results to several applications including deblurring, refocus, and depth

(a) Input

(b) Deblurring result

(c) Ground truth mask

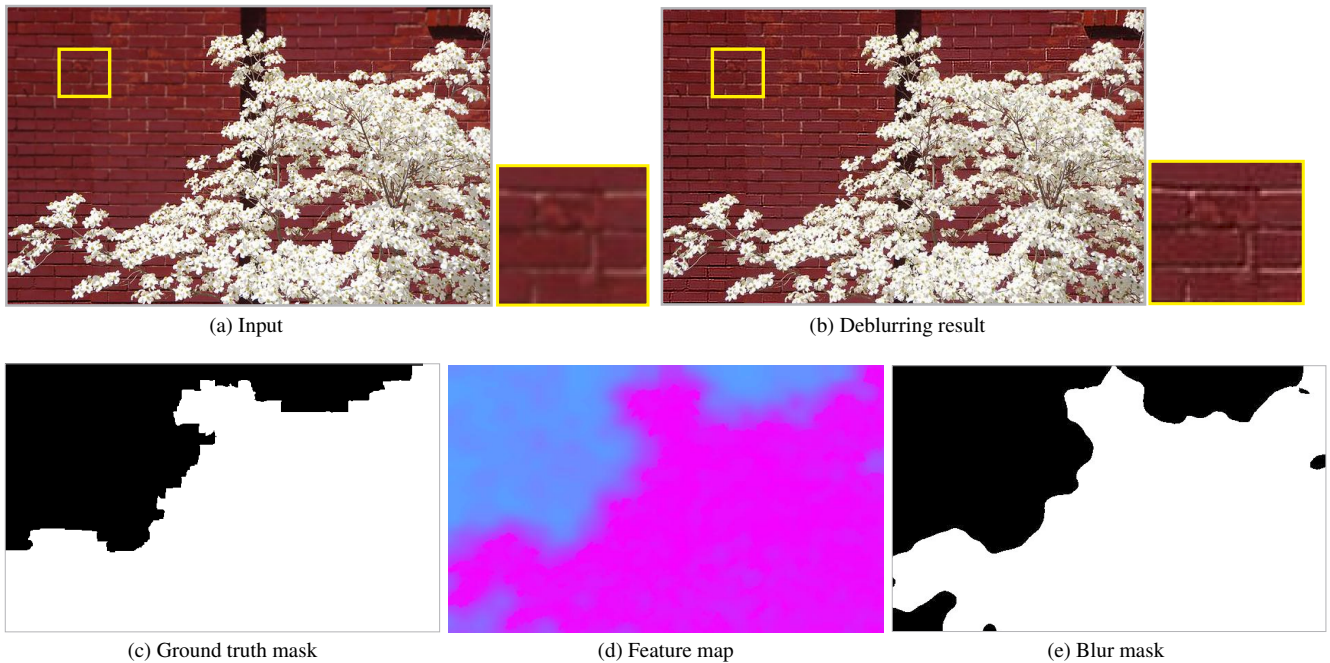
(d) Feature map

(e) Blur mask
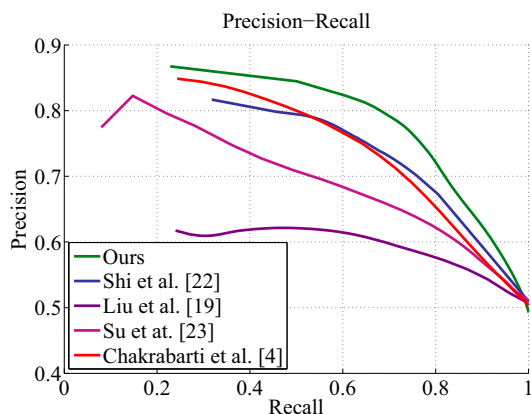
Figure 11. Deblurring using our blur estimate.

Figure 10. Precision-recall comparison on small blur data in [22].

estimation.

5.1. Deblurring Using the Blur Estimate

We handle spatially varying blur in this section. The input JNB image is shown in Fig. 11(a) with ground-truth clear mask in (c). We create the blur region mask by cropping our blur feature map with values over 45. The obtained mask is shown in Fig. 11(d). It is close to the ground truth mask. Then we deconvolve it [10]. The deblurring result is shown in Fig. 11(b), which manifests that our estimated blurriness is usable to recover a clear image.

5.2. Refocus Using Blur Estimate

The estimated blur maps can be regarded as a coarse representation for depth maps, which are used to produce the refocus effect. We first generate 20 different blurred versions for an image, and quantize the blur map into 20 different levels. For each pixel, according to the distance between its blur feature value and the referenced blur feature in the current round, we choose one result from previous 20 blurred images. An input image and its corresponding feature map are shown in Fig. 12(a) and (b). Two different refocus effects are shown in (c) and (d).
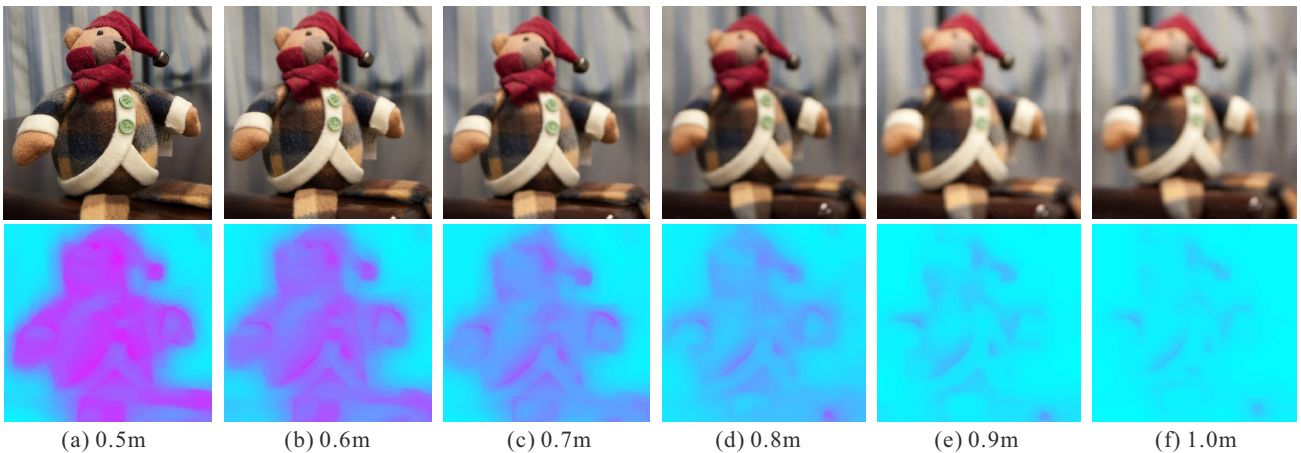
5.3. Depth Estimation

A set of images taken under the same aperture size but with different distances are used to verify how blur estimate relates to depth. Fig. 13 shows our 6 reference images with their corresponding blur feature values in different distances. As distance increases, feature values for the central object decreases. We use the median feature value on the toy region as reference to describe the image, which excludes the influence from the confusing totally blurred background. The six images have corresponding feature values 55, 48, 41, 32, 20 and 16. For the test image, we also select a reasonable region to calculate the feature value. Three test cases are shown in Fig. 14(a), (c), and (e). The feature values in (b), (d), and (f) are 44, 55, and 36 respectively, which are consistent with the feature values in their reference distances.

(a) Input and feature map   (b) Refocus result   (c) Refocus result

Figure 12. Refocusing using our blur map. (a) is the input image. (b) is our learned feature map. (c) shows different refocusing result given our blur feature map. The arrow highlights the focus point.

(a) 0.5m   (b) 0.6m   (c) 0.7m   (d) 0.8m   (e) 0.9m   (f) 1.0m

Figure 13. Calibrated image with their corresponding blur features. The set of images are taken using aperture size f/5.0 in different distances.
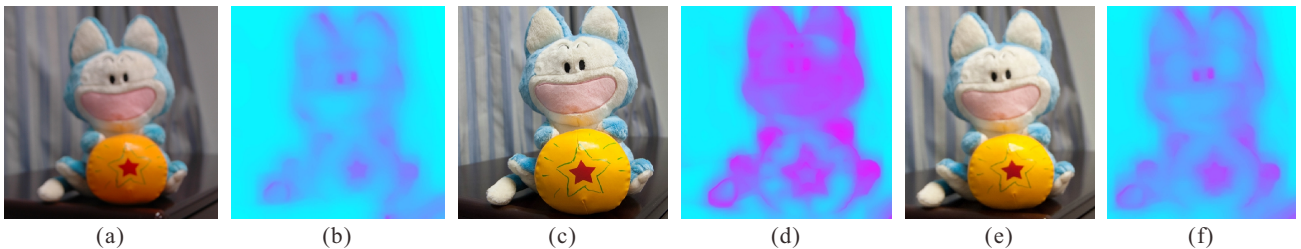

(a)   (b)   (c)   (d)   (e)   (f)

Figure 14. (a), (c), and (e) are three test examples with distances 0.8m, 0.5m, and 0.7m. (b), (d), and (f) are their corresponding features.

6. Conclusion and Discussion

We have explored an important topic to estimate just noticeable blur. We first analyzed previous local feature methods. Then, a new sparse feature was developed for just noticeable blur detection. We showed that this feature directly corresponds to blur strength. The learned blur features can benefit various applications, including image deblurring, image refocus and depth estimation.

Our current framework does not consider strong noise as it could be taken similarly as details in clear regions.

Our possible future work includes developing a more robust propagation method adaptive to blur feature applications, as well as using the blur map to assist high level vision applications, such as image classification and detection.

Acknowledgments

References

[1] M. Aharon, M. Elad, and A. Bruckstein. K-svd: An algorithm for designing overcomplete dictionaries for sparse representation. *IEEE Transactions on Signal Processing*, 54(11):4311–4322, 2006.

[2] S. Bae and F. Durand. Defocus magnification. 26(3):571–579, 2007.

[3] A. Chakrabarti and T. Zickler. Depth and deblurring from a spectrally-varying depth-of-field. In *European Conference on Computer Vision (ECCV)*, pages 648–661. 2012.

[4] A. Chakrabarti, T. Zickler, and W. T. Freeman. Analyzing spatially-varying blur. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2512–2519, 2010.

[5] S. Cho and S. Lee. Fast motion deblurring. 28(5):145, 2009.

[6] S. Dai and Y. Wu. Estimating space-variant motion blur without deblurring. In *IEEE International Conference on Image Processing (ICIP)*, pages 661–664, 2008.

[7] S. Dai and Y. Wu. Removing partial blur in a single image. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2544–2551, 2009.

[8] J. H. Elder and S. W. Zucker. Local scale control for edge detection and blur estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 20(7):699–716, 1998.

[9] R. Fergus, B. Singh, A. Hertzmann, S. T. Roweis, and W. T. Freeman. Removing camera shake from a single photograph. 25(3):787–794, 2006.

[10] D. Fish, A. Brinicombe, E. Pike, and J. Walker. Blind deconvolution by means of the richardson–lucy algorithm. *Journal of the Optical Society of America A*, 12(1):58–65, 1995.

[11] N. Joshi, S. B. Kang, C. L. Zitnick, and R. Szeliski. Image deblurring using inertial measurement sensors. *ACM Transactions on Graphics (TOG)*, 29(4):30, 2010.

[12] N. Joshi, R. Szeliski, and D. Kriegman. Psf estimation using sharp edge prediction. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8, 2008.

[13] D. Krishnan and R. Fergus. Fast image deconvolution using hyper-laplacian priors. In *The Conference on Neural Information Processing Systems (NIPS)*, pages 1033–1041, 2009.

[14] A. Levin. Blind motion deblurring using image statistics. *The Conference on Neural Information Processing Systems (NIPS)*, 19:841, 2007.

[15] A. Levin, R. Fergus, F. Durand, and W. T. Freeman. Image and depth from a conventional camera with a coded aperture. *ACM Transactions on Graphics (TOG)*, 26(3):70, 2007.

[16] A. Levin, D. Lischinski, and Y. Weiss. A closed-form solution to natural image matting. pages 228–242, 2008.

[17] A. Levin, Y. Weiss, F. Durand, and W. T. Freeman. Understanding and evaluating blind deconvolution algorithms. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1964–1971, 2009.

[18] H. T. Lin, Y.-W. Tai, and M. S. Brown. Motion regularization for matting motion blurred objects. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 33(11):2329–2336, 2011.

[19] R. Liu, Z. Li, and J. Jia. Image partial blur detection and classification. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8, 2008.

[20] J. Mairal, F. Bach, J. Ponce, and G. Sapiro. Online learning for matrix factorization and sparse coding. *The Journal of Machine Learning Research (JMLR)*, 11:19–60, 2010.

[21] C. J. Schuler, M. Hirsch, S. Harmeling, and B. Schölkopf. Blind correction of optical aberrations. In *European Conference on Computer Vision (ECCV)*, pages 187–200. 2012.

[22] J. Shi, L. Xu, and J. Jia. Discriminative blur detection features. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

[23] B. Su, S. Lu, and C. L. Tan. Blurred image region detection and classification. In *ACM international conference on Multimedia*, pages 1397–1400, 2011.

[24] Y.-W. Tai and M. S. Brown. Single image defocus map estimation using local contrast prior. In *IEEE International Conference on Image Processing (ICIP)*, pages 1797–1800, 2009.

[25] Y.-W. Tai, X. Chen, S. Kim, S. J. Kim, F. Li, J. Yang, J. Yu, Y. Matsushita, and M. S. Brown. Nonlinear camera response functions and image deblurring: Theoretical analysis and practice. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 35(10):2498–2512, 2013.

[26] Y.-W. Tai, H. Du, M. S. Brown, and S. Lin. Correction of spatially varying image and video motion blur using a hybrid camera. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 32(6):1012–1028, 2010.

[27] L. Xu and J. Jia. Two-phase kernel estimation for robust motion deblurring. In *European Conference on Computer Vision (ECCV)*, pages 157–170. 2010.

[28] L. Yuan, J. Sun, L. Quan, and H.-Y. Shum. Progressive inter-scale and intra-scale non-blind image deconvolution. 27(3):74, 2008.

[29] Y. Zhang and K. Hirakawa. Blur processing using double discrete wavelet transform. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1091–1098, 2013.

[30] X. Zhu, S. Cohen, S. Schiller, and P. Milanfar. Estimating spatially varying defocus blur from a single image. *IEEE Transactions on Image Processing (TIP)*, 22(12):4879–4891, 2013.

[31] S. Zhuo and T. Sim. Defocus map estimation from a single image. *Pattern Recognition*, 44(9):1852–1858, 2011.