Discriminative Blur Detection Features

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
http://www.cse.cuhk.edu.hk/leojia/projects/dblurdetect/

Abstract

Ubiquitous image blur brings out a practically important question – what are effective features to differentiate between blurred and unblurred image regions. We address it by studying a few blur feature representations in image gradient, Fourier domain, and data-driven local filters. Unlike previous methods, which are often based on restoration mechanisms, our features are constructed to enhance discriminative power and are adaptive to various blur scales in images. To avail evaluation, we build a new blur perception dataset containing thousands of images with labeled ground-truth. Our results are applied to several applications, including blur region segmentation, deblurring, and blur magnification.

1. Introduction

Blur is one type of photo degradation that leads to loss of details. In many special cases, it can also be a visual effect purposely generated by photographers to give prominence to foreground persons or other important objects based on defocus or camera/object motion.

With the fast development of computer vision techniques, it becomes important and practical to understand information immersed in blurred images or regions. We address a central *blur detection* problem in this area, since quickly and effectively finding blur pixels can naturally benefit many applications including but not restricted to image segmentation, object detection, scene classification, image quality assessment, image restoration, and photo editing [6, 23, 21], given the fact that many blurred images exist online or are produced from personal cameras.

There have been a series of methods directly solving blind [4, 25, 7, 15] and non-blind [27, 12] deconvolution problems. They aim at explicitly inferring latent images and/or blur kernels. Our goal in blur detection is not to follow this line using deconvolution [11]. Instead, we will focus on finding and constructing *blur feature representations* directly from input images and making them potent enough to differentiate between blurred and unblurred regions, which are of high importance in feature understanding.

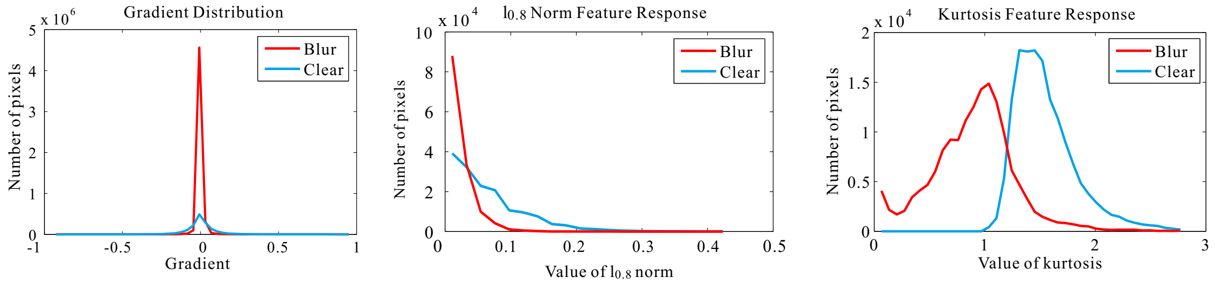
A few previous methods relate to explicit blur detection. Levin [14] used image statistics to identify partial motion blur. Lin *et al.* [16] also explored natural image statistics for blur analysis. Liu *et al.* [17] designed four local blur features for blur confidence and type classification. Chakrabarti *et al.* [3] analyzed directional blur via local Fourier transform. Dai and Wu [5] developed a two-layer image model on alpha channel to estimate partial blur. Different from these approaches directly fitting natural image statistics, we in this paper analyze feature discrepancy in gradient and Fourier space. We also propose a few features that are with decent discrimination ability theoretically and empirically.

In addition to feature construction, we explore a data-driven solution, which learns local filters. We build a new blur detection dataset that contains 1000 images with human labeled ground-truth blur regions. These data not only make detection results convincing, but also provide useful resource to understand blur with respect to structure diversity in natural images. It enables training and testing, which are traditionally hard to implement without suitable data.

Our contribution is three-fold. First, we design a set of blur features in multiple domains. Second, we develop a multi-scale solution for blur perception that avoids scale ambiguity. Third, we build a blur detection dataset with ground-truth labels on 1000 images, which provides a reasonable evaluation platform for blur analysis. We apply our results to several applications, including blur region segmentation, image debluring and blur magnification.

2. Blur Features

We deal with challenging partially blurred images where the point spread function (PSF) varies across the image.



(a) (b) (c)

Figure 1. Gradient level statistics. (a) Blur-pixel gradient distribution has a strong peak and a small tail. (b) Resulting value distribution using $\ell_{0.8}$ norm on gradient. (c) Value distributions using our kurtosis measure. Both the blurred and unblurred patches are extracted from our dataset with one million samples.

Following tradition, blur formation within a window can be expressed as convolution like

$$B = I * k, \tag{1}$$

where I is the local latent patch, k is the local PSF, $*$ models 2D convolution, and B is the blur observation. Note we do not aim to restore the local PSF k, which is difficult to be accurate for small patches. Instead, we study the structural difference between corresponding clear/blur regions to devise local blur feature representations.

2.1. Image Gradient Distribution

Natural images vary from scene to scene. The general principle that gradient follows a heavy-tailed distribution has been known in this community for years. But do these distributions make much difference on blurred and unblurred image regions? Intuitively, blurred patches seldom contain sharp edges, which lead to distributions containing small values. We plot gradient distributions in Fig. 1(a). There is clear visual difference. We are thus interested to propose effective measures to model it.

In blur image restoration, ℓ_p norm $(0.7 \leq p \leq 1)$ [9], ℓ_1/ℓ_2 norm [13], to name a few, are successfully employed as regularizers or priors. These terms however do not tell the major difference between blur and clear patches in detection. We plot the feature response of $\ell_{0.8}$ norm on gradient using one million sample points with blur/clear ground truth in Fig. 1(b). The resulting two distributions largely overlap, making these two types of patches not easily separable. Other ℓ_p norms or ℓ_1/ℓ_2 function present similar performance. Different from these metrics, we characterize features by peakedness and heavy-tailedness.

Peakedness Measure We measure the peakedness of a distribution by *kurtosis*, which is defined as

$$K(a) = \frac{E[a^4]}{E^2[a^2]} - 3, \tag{2}$$

where $E[\cdot]$ is the expectation operator for input data vector a. The kurtosis is defined on the forth and second order moments and measures peakedness of a distribution. The -3 operator is to make normal-distribution kurtosis approach zero. For natural images, a gradient distribution has an acute peak around zero and a heavy tail. It corresponds to a category, namely, the leptokurtic distribution, with a positive kurtosis value.

The blur process widens the gradient distribution of a natural image and therefore decreases kurtosis. We denote by (I_x, I_y) and (B_x, B_y) gradients of I and B in two orthogonal directions. Assuming I_x and I_y are i.i.d., we derive the following relationship.

Claim 1. *Given the local blur model and kurtosis measure defined in Eqs. (1) and (2), it is guaranteed to have $K(B_x) \leq K(I_x)$ and $K(B_y) \leq K(I_y)$.*

Proof. The heavy-tailed gradient distributions ensure $K(I_x) > 0$ and $K(I_y) > 0$. The second moment of the blur gradient B_x for pixel (i, j) can be expressed as

$$\begin{aligned} E[B_x^2(i,j)] &= E[(\sum_{l,m} I_x(i-l, j-m)k(l,m))^2] \\ &= \sum_{l,m,l',m'} E[I_x(i-l, j-m) I_x(i-l', j-m')] k(l,m) k(l',m') \\ &= E[I_x^2(i,j)] \sum_{l,m} k(l,m)^2. \end{aligned} \tag{3}$$

The last equation comes from the i.i.d. assumption on I_x. Similarly, by expanding $E[B_x^4]$, we get

$$\begin{aligned} E[B_x^4(i,j)] &= E[I_x^4(i,j)] \sum_{l,m} k(l,m)^4 + \\ & 3E^2[I_x^2(i,j)] \Big((\sum_{l,m} k(l,m)^2)^2 - \sum_{l,m} k(l,m)^4 \Big). \end{aligned} \tag{4}$$

Substituting Eqs. (3) and (4) into Eq. (2), we get

$$K(B_x(i,j)) = \frac{\sum_{l,m} k(l,m)^4}{(\sum_{l,m} k(l,m)^2)^2} K(I_x(i,j)). \tag{5}$$

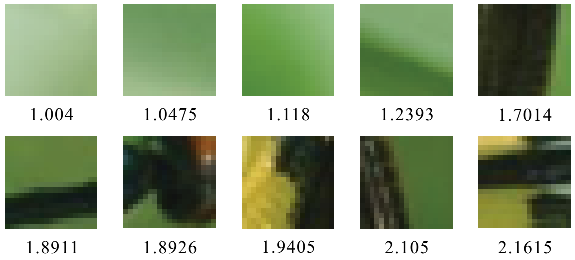


Figure 2. An illustration of kurtosis for different patches. The kurtosis feature value f_1 is given in Eq. (7). Unblurred patches yield larger values than blurred ones.

Further considering the blur PSF constraints $\sum_{l,m} k(l,m) = 1$ and $k(l,m) \geq 0$ yields

$$\sum_{l,m} k(l,m)^4 \leq \left(\sum_{l,m} k(l,m)^2 \right)^2 . \tag{6}$$

In this regard, $K(B_x(i,j)) \leq K(I_x(i,j))$. Similar conclusion applies to $K(B_y) \leq K(I_y)$. □

This claim presents the fact that kurtosis varies in blurred and unblurred regions. It is applied to gradients in different directions and thus has extra directional information. Given the input patch B, which could be blurred or unblurred, we define the first feature as

$$f_1 = \min(\ln(K(B_x) + 3), \ln(K(B_y) + 3)). \tag{7}$$

The logarithm is to map the feature to a suitable range. The $\min(\cdot)$ operator selects the smaller score between values in x- and y-directions. Larger values correspond to less blurred patches. To quickly verify how useful this feature is, we plot the feature values on one million patches that have already ground truth labels. The kurtosis distributions for blurred and unblurred patches are shown in Fig. 1(c).

The plotted two distributions are with quite different means and the overlapping region is small. This manifests the potential discriminative ability when applying this feature to detection. We show a few patches along with their feature responses in Fig. 2. Kurtosis for a blurred patch is much smaller than that of an unblurred one.

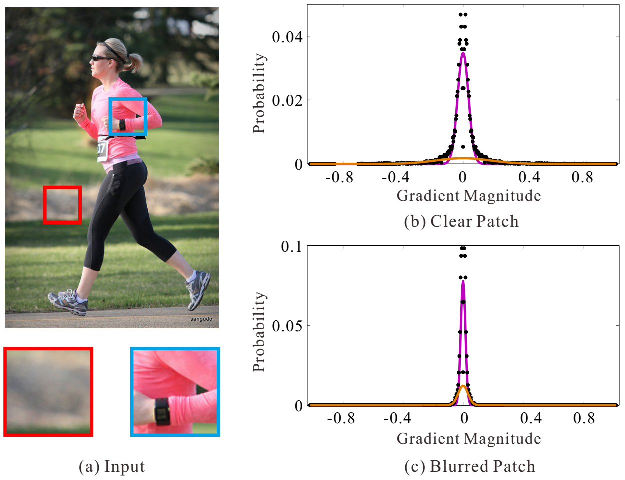
Heavy-Tailedness Measure While Kurtosis describes a general distribution property of peakedness, it is a bonus to also know the level of tailedness of a distribution since blur largely reduces gradient magnitudes. We fit a Gaussian mixture model for gradient magnitude ∇B using two components, yielding

$$\nabla B \sim \pi_1 G(\nabla B | \mu_1, \sigma_1) + \pi_2 G(\nabla B | \mu_2, \sigma_2), \tag{8}$$

where σ_1 and σ_2 are the standard deviations. One example is shown in Fig. 3. Between the two distributions, one fits most of the peak and the other contains primarily the heavy tail. We denote σ_1 as the larger variance between the two. Because the tail distribution variance in the clear patch is much bigger than that of the blur one, the tailedness feature is set as

$$f_2 = \sigma_1. \tag{9}$$

It is useful as one feature dimension to generally mark the difference between blurred and unblurred patches.



(b) Clear Patch

(a) Input

(c) Blurred Patch

Figure 3. Illustration of heavy-tailedness. (a) Input blur and clear patches. (b)-(c) Gradient magnitude distributions. The black dots are original magnitudes. They are fitted by two Gaussian distributions in solid curves in different colors.

2.2. Spectra in Frequency Domain

In frequency domain, it was observed that the average power spectrum of natural images $J(\omega)$ is with the form $1/\omega^\alpha$ [2, 8, 24] given $\alpha = 2$. The average power spectrum $J(\omega)$ is defined as

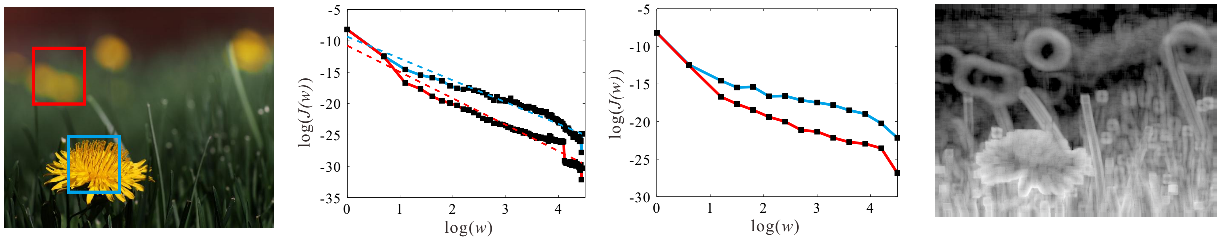
$$J(\omega) = \frac{1}{n} \sum_{\theta} J(\omega, \theta) \simeq \frac{A}{\omega^\alpha}, \tag{10}$$

where n is the number of different θ; (ω, θ) is the polar coordinate for pixel (i,j); A is an amplitude scaling factor; and $J(\omega, \theta)$ is the square magnitude of discrete Fourier transform (DFT).

Averaged power spectrum, intuitively, represents the strength of change. Blur attenuates high frequency components and therefore makes the power spectra fall off much faster than its sharp counterpart. We prove it as follows based on two common types of kernels.

Claim 2. *Given a natural image patch x and its Gaussian or box blurred version y by PSF k, the fall-off speed of the average power spectrum on y is several orders faster than that of x. It is expressed as*

$$\lim_{\omega \to \infty} \omega^2 J_y(\omega) = 0. \tag{11}$$



(a) Input (b) Liu et al. (c) Ours (d) Feature response

Figure 4. Spectrum feature illustration. (a) Input image. We label two regions in red and blue for analysis in (b) and (c). (b) Feature representation of Liu *et al.* [17]. The slope of fitted dash lines is used to discriminate between two patch types. (c) Our feature is more reliable by computing the area size below the curves. (d) Our feature map for the whole image. Smooth regions have larger values than sharp ones.

Proof. Given that convolution changes to multiplication after Fourier transform, we obtain

$$\begin{aligned}\lim_{\omega\to\infty}\omega^2 J_y(\omega) &= \lim_{\omega\to\infty}\omega^2 J_x(\omega)J_k(\omega)\\ \simeq \lim_{\omega\to\infty}\omega^2\frac{A}{\omega^2}J_k(\omega) &= \lim_{\omega\to\infty}AJ_k(\omega).\end{aligned}\qquad(12)$$

The *optical transfer function* (OTF) of a Gaussian filter remains a Gaussian; and the OTF of a box filter is a sinc function. So the average spectrum $J_k(\omega)$ of the kernel k with polar coordinates becomes $Be^{-c\omega^2}$ and B sinc$(c\omega)$ respectively. Both functions converge to zero when ω is infinitely large. Eq. (12) leads to the conclusion that the average power spectrum of a blurred patch under these kernels falls off faster than its clear counterpart. □

The above proof is on two types of kernels. Empirically, we also test available motion and defocus kernels by blind deconvolution, and unexceptionally get the same conclusion. This property is thus a general one. Instead of fitting a linear model, we sum power spectra as

$$f_2 = \sum_{\omega}\log(J(\omega)),\qquad(13)$$

which can be used directly to distinguish between blurred and unblurred patches. Its effectiveness is proved as follows.

Claim 3. *Given a natural image patch x, which is blurred by a PSF to form patch y, the cumulated average power spectrum for the blurred patch is smaller than that for the sharp patch, i.e.,*

$$\sum_{\omega}\log(J_y(\omega)) \le \sum_{\omega}\log(J_x(\omega)).\qquad(14)$$

Proof. After Fourier transform, we get

$$\sum_{\omega}\log(J_y(\omega)) = \sum_{\omega}\log(J_x(\omega)J_k(\omega)).\qquad(15)$$

The average power spectrum for the PSF satisfies

$$J_k(\omega) = (\sum_n k(n)e^{-i\omega n})^2 \le (\sum |k(n)|)^2 = 1,\qquad(16)$$

based on the definition of Fourier bases. Putting Eqs. (15) and (16) together, we get Eq. (14). □

We note our feature is more general and robust than the property described in [17] where only a line relationship is considered, as shown in Fig. 4(b). The black squares are the sample points. These two lines may easily over-fit input data because there are more samples at the high-frequency end. They are also vulnerable to outliers when small patches only contain a few spectrum samples.

In comparison, we uniformly sample $\log(\omega)$ to reconstruct frequency curves, as shown in Fig. 4(c). The $\log(\omega)-\log(J(\omega))$ curve is stable with respect to high frequency variation. Our final feature map is shown in Fig 4(d). Sharper regions yield larger values. It is one clue for blur detection.

2.3. Local Filters

Above features are based on natural image statistics. We also study how spatial filters such as Gabor [10] and Laplacian, can be used in this detection problem. They capture local band-pass or high-pass information that supplements frequency and gradient domain features. There is nearly no prior work to study how these handcrafted features behave in blur detection.

Based on our new dataset and ground-truth labels, we denote the labeled blur patch set as $\mathcal{B} = \{B_1,\ldots,B_p\}$ and unblurred patch set as $\mathcal{I} = \{I_i,\ldots,I_q\}$. Our goal is to obtain a group of linearly independent filters to best separate these two sets. In this regard, we denote data scatter for the blur set as

$$S_B = \sum_{x\in\mathcal{B}}(B-\mu_B)(B-\mu_B)^T,\qquad(17)$$

where $\mu_B = \frac{1}{p}\sum_{B\in\mathcal{B}}B$ is the mean. The data scatter for the other set S_I is defined similarly. Based on single-class

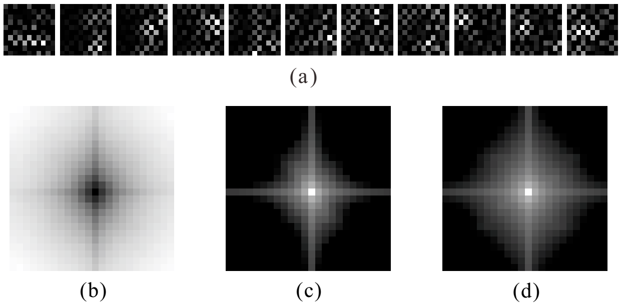


Figure 5. Our learned local linear filters. (a) Top 11 learned features. (b) Spectra of DFT for the learned linear filters. (c)-(d) Spectra for blurred and unblurred patches respectively.

data scatter measures, the intra- and inter-class scatters are written as $S_w = S_B + S_I$ and $S_b = (\mu_B - \mu_I)(\mu_B - \mu_I)^T$. We compute an invertible mapping matrix W to make the mapped feature response most discriminative. It is expressed as

$$\max_{W} \frac{tr(W^T S_b W)}{tr(W^T S_w W)}. \tag{18}$$

It is equivalent to the generalized eigenvalue problem

$$S_b w_i = \lambda_i S_w w_i, \tag{19}$$

with w_i being the generalized eigenvector and λ_i being its corresponding eigenvalue in an descending order. Each w_i is a learned local filter. The final blur feature is denoted as

$$f_3^n = \{w_1^T B, \ldots, w_n^T B\}, \tag{20}$$

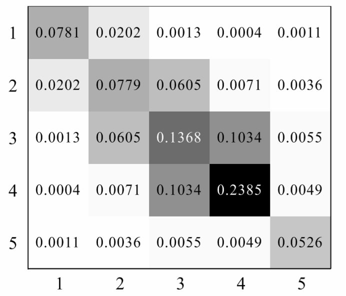
given the generalized eigenvectors corresponding to the n largest eigenvalues.

We analyze the usefulness of the learned filters in Fig. 5. We randomly sample one million blurred and unblurred patches from our dataset. The top-score learned filters are demonstrated in Fig. 5(a). Their structures are not intuitive. There is obvious difference from handcrafted gradient and Laplacian filters. By plotting the average log-square magnitude of DFT for the first 100 filters in Fig. 5(b), we notice the function is a special high-pass.

The spectrum maps for blurred and unblurred patches are shown in Fig. 5(c) and (d) after filtering. The two maps make a significant change in the mid- and high-frequency regions, manifesting that our learned filters enhance the difference specific for natural images under blur.

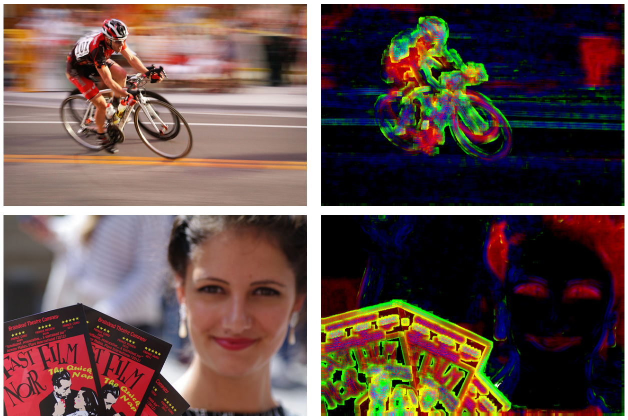
2.4. Final Feature Construction and Analysis

The above deliberately developed local blur features, including distribution measure, Fourier domain descriptor, and local filters, depict different aspects of blur. We plot the cross feature correlation by feature covariance in Fig. 6. Most feature pairs perform quite independently, which indicate features supplement each other. To better understand them, we visualize our features in 3D by PCA in Fig. 7. The condensed 3D features are mapped into RGB channels. The resulting feature map highlights different effective blur properties locally in the input images.



	1	2	3	4	5
1	0.0781	0.0202	0.0013	0.0004	0.0011
2	0.0202	0.0779	0.0605	0.0071	0.0036
3	0.0013	0.0605	0.1368	0.1034	0.0055
4	0.0004	0.0071	0.1034	0.2385	0.0049
5	0.0011	0.0036	0.0055	0.0049	0.0526

Figure 6. Feature covariance. Features of kurtosis, heavy-tailedness, spectrum area, the 1st local filter, and the 2nd local filter are indexed from 1 to 5.



(a) Input. (b) Features in 3 dimensions.

Figure 7. Visualizing feature in 3 dimensions.

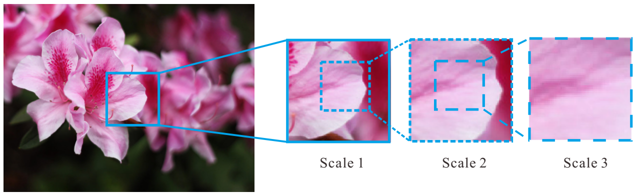


Figure 8. Illustration of multi-scale blur perception. The blur confidence is highly related to patch scales.

To combine all these features, we use a naive Bayesian classifier to learn the posterior for the set of features. The posterior score is used as our final representation. The naive Bayesian classifier naturally integrates the features in a discriminative way.

3. Multi-Scale Perception

Besides feature development and learning, we also contribute a unified blur confidence map by considering scales for detecting blur, since this is a perceptually sensitive pro-

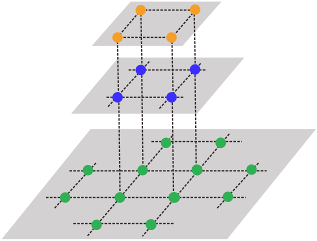


Figure 9. Our multi-scale graphical model.

cess according to the illustration in Fig. 8. Looking from only one resolution, it may not be accurate to know whether an image or patch is blurred or not. The scale ambiguity has been studied in various applications [26, 18]. We resort to a multi-scale model to fuse information from different levels.

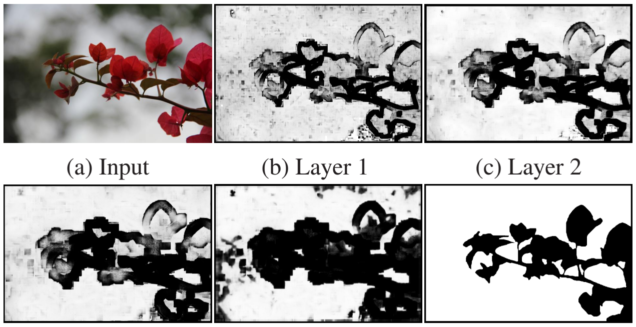
Our model extracts local blur features from three different scales. Given an input image, for each scale, we first divide the image into patches and compute local blur feature responses (i.e., the posterior score in Section 2.4) on them. Then a multi-scale structure is constructed as Fig. 9. Specifically, a blur response b_i^s is calculated on the patch centered at pixel i at a particular scale s. Our model connects the blur score of each pixel with those of the surrounding pixels. Inter-scale correlation is also built among patches centered at the same corresponding pixel in different levels.

Given local blur response $\{\hat{b}_i^s\}$ in each scale s and for each pixel i, the total energy on the graphical model is expressed as

$$E(b) = \sum_{s=1}^{3}\sum_{i} |b_i^s - \hat{b}_i^s| + \alpha \sum_{s=1}^{3}\sum_{i}\sum_{j\in\mathcal{N}_i^s} |b_i^s - b_j^s| + \beta \sum_{s=1}^{2}\sum_{i} |b_i^s - b_i^{s+1}|, \tag{21}$$

where b_i^s is the score we need to infer for each pixel. The first data term is unary to preserve the overall feature structure in image space. The second term is the spatial affinity, where $\mathcal{N}_i^s$ is the four-neighbor set for pixel i in scale s. The last term is the inter-scale affinity, which bridges feature responses in different levels. b_i^s and b_i^{s+1} have the same center pixel in two scales. α and β are weights. All the terms in Eq. (21) use the ℓ_1 norm distance for robust inference.

Eq. (21) can be optimized via loopy belief propagation [19]. It starts from an initial set of propagation messages, and then iterates through each node by applying message passing until convergence. The final blur response map in the top layer is our result. An inference example is shown in Fig. 10. Though the blur indicator in each layer contains errors, our final response map is much better than any of them after scale influence in blur detection.



(a) Input (b) Layer 1 (c) Layer 2

(d) Layer 3 (e) Final response (e) Ground truth

Figure 10. Blur response maps in three layers and our final representation.

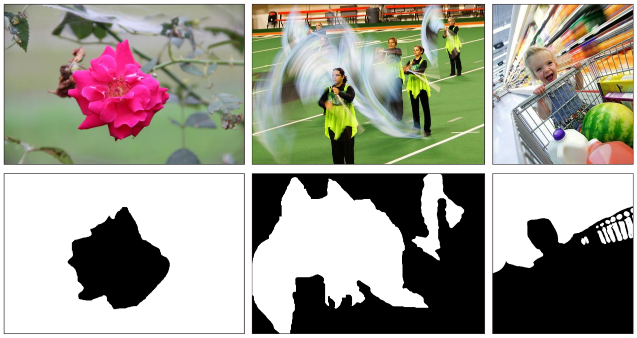
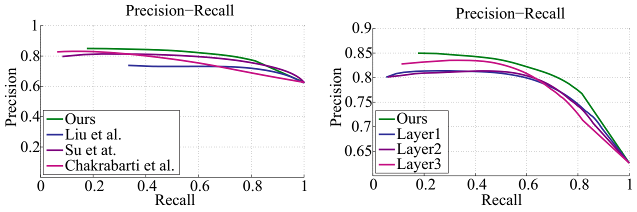


Figure 11. Representative images in our dataset.



(a) (b)

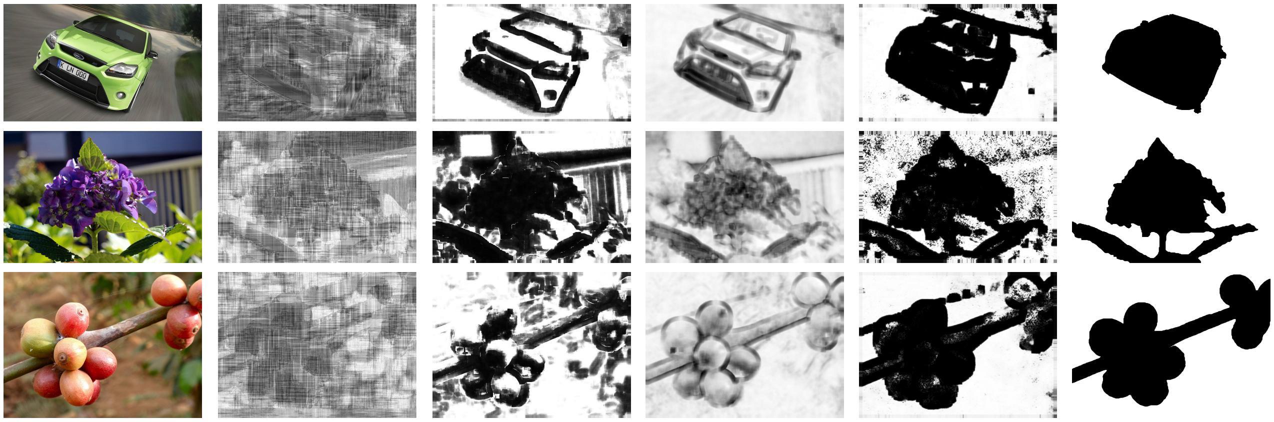
Figure 12. Quantitative comparison. (a) Precision-recall curves for different methods. (b) Precision-recall curves between our single-resolution and multi-scale results.

4. Experimental Results

In our experiments, both parameters α and β in the multi-scale model are set to 0.5. To conduct fair and statistical comparison, we construct a blur detection dataset with 1000 images. It consists of images with out-of-focus blur and partial motion blur. We ask helpers with good understanding of blur to cross label the blur regions in each image. Several examples are shown in Fig. 11. The whole dataset is downloadable from the project website.

4.1. Method Evaluation

We compare our method with state-of-the-arts [17, 22, 3] using existing or our (if the executable is not available online) implementation. Previous work introduced image fea-



(a) Input (b) Chakrabarti *et al.* (c) Liu *et al.* (d) Su *et al.* (e) Ours (f) Ground truth

Figure 13. Visual comparison on our data for local blur detection.

tures different from ours in terms of construction procedure and discrimination ability consideration. Our multi-scale blur information is also important for high quality blur estimation.

We provide quantitative comparison on our dataset via *precision-recall curve* in Fig. 12(a), where the final blur maps are binary ones within range $[0,\ 100]$. Our approach achieves the highest precision within almost the entire recall range $[0,\ 1]$. This is mainly due to the adaptive selection of discriminative local blur features, as well as the multi-level blur propagation. All the recall values in our results are larger than 0.5, which indicate a small chance to miss true positive samples in all thresholds.

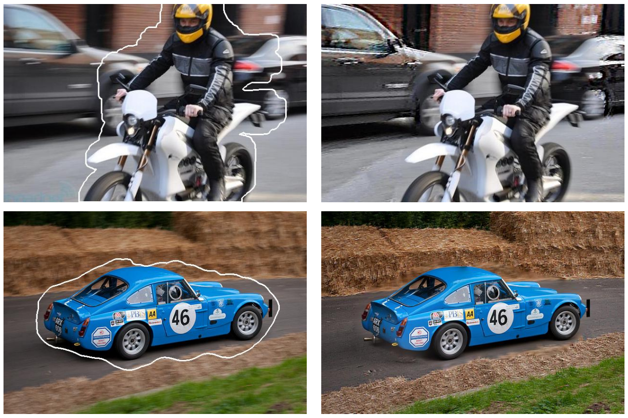
To analyze the effectiveness of the multi-scale scheme, we compare the precision-recall curves generated on our three single-layer maps and our final one in Fig. 12(b). Considering all level information via inter-layer confidence passing is better than only using one scale for blur detection.

A few of our results are compared to those of previous methods in Fig. 13. Our method handles well images with complex foreground and background under various blur causes. Our blur detection maps contain many high confidence values close to the ground truth. More are included in our supplementary file.

4.2. Applications Based on Blur Detection

Several computer vision applications can be benefitted from our blur detection task. We show two in what follows and more examples in the project website.

Blur Segmentation and Deblurring With our learned blur maps, it is possible to segment images into blur and clear regions. We adopt the graph-cut method in [20] and set the S and T nodes in it to pixels with blur confidence over 0.9 and below 0.1 respectively. Two segmentation results are shown in Fig. 14(a).



(a) (b)

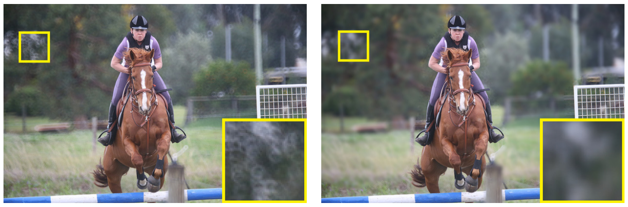
Figure 14. Spatially varying motion deblurring. (a) Input images with blur region masks. (b) Deblurring results.

Further with the segmented blur regions, we can possibly restore partial blurred images. Without usable blur masks, non-blind deconvolution mixes foreground and background under different motion. Our method is to deblur pixels only inside blur masks similar to the procedure described in [25]. Finally we put the original unblurred region back. A few results are shown in Fig. 14(b).

Blur Magnification Given the blurred image region, we can perform blur magnification [1], which produces a higher level of defocus. We show an example in Fig. 15. The resulting image is visually pleasing.

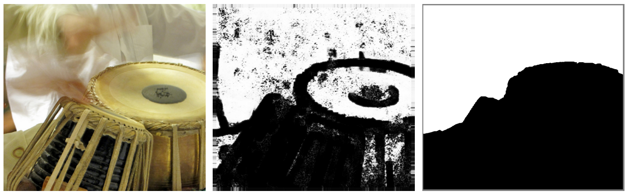
5. Conclusion

We have proposed a few effective local blur features. They describe different blur properties and are integrated into a multi-scale inference framework to handle scale variation. Another major contribution is that we have built a partial blur dataset with ground-truth blur labels, availing



(a) Input image. (b) Editing result.

Figure 15. Blur magnification.



(a) Original image (b) Our result (c) Ground truth

Figure 16. One failure example.

future research along this line.

Our method could occasionally fail. For example, when the background is textureless and foreground is motion blurred, pixels on both of these regions could be detected as blur as shown in Fig. 16. Thus further study in the semantic level will be our future work.

Acknowledgements

We thank Liwei Wang, Di Lin, and Xin Tao for their help and insightful discussion. This work is supported by a grant from the Research Grants Council of the Hong Kong SAR (project No. 413110) and by NSF of China (key project No. 61133009).

References

[1] S. Bae and F. Durand. Defocus magnification. *Computer Graphics Forum*, 26(3):571–579, 2007.

[2] G. Burton and I. R. Moorhead. Color and spatial structure in natural scenes. *Applied Optics*, 26(1):157–170, 1987.

[3] A. Chakrabarti, T. Zickler, and W. T. Freeman. Analyzing spatially-varying blur. In *CVPR*, pages 2512–2519, 2010.

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

[5] S. Dai and Y. Wu. Removing partial blur in a single image. In *CVPR*, pages 2544–2551, 2009.

[6] K. G. Derpanis, M. Lecce, K. Daniilidis, and R. P. Wildes. Dynamic scene understanding: The role of orientation features in space and time in scene classification. In *CVPR*, pages 1306–1313, 2012.

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

[8] D. J. Field et al. Relations between the statistics of natural images and the response properties of cortical cells. *J. Opt. Soc. Am. A*, 4(12):2379–2394, 1987.

[9] A. Gupta, N. Joshi, C. L. Zitnick, M. Cohen, and B. Curless. Single image deblurring using motion density functions. In *ECCV*, pages 171–184, 2010.

[10] J. P. Jones and L. A. Palmer. An evaluation of the two-dimensional gabor filter model of simple receptive fields in cat striate cortex. *Journal of Neurophysiology*, 58(6), 1987.

[11] L. Kovacs and T. Sziranyi. Focus area extraction by blind deconvolution for defining regions of interest. *PAMI*, 29(6):1080–1085, 2007.

[12] D. Krishnan and R. Fergus. Fast image deconvolution using hyper-laplacian priors. In *NIPS*, pages 1033–1041, 2009.

[13] D. Krishnan, T. Tay, and R. Fergus. Blind deconvolution using a normalized sparsity measure. In *CVPR*, 2011.

[14] A. Levin. Blind motion deblurring using image statistics. *NIPS*, 19:841, 2007.

[15] A. Levin, Y. Weiss, F. Durand, and W. T. Freeman. Understanding and evaluating blind deconvolution algorithms. In *CVPR*, pages 1964–1971, 2009.

[16] 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*, 33(11):2329–2336, 2011.

[17] R. Liu, Z. Li, and J. Jia. Image partial blur detection and classification. In *CVPR*, pages 1–8, 2008.

[18] C. Lu, J. Shi, and J. Jia. Abnormal event detection at 150 fps in matlab. In *ICCV*, 2013.

[19] K. P. Murphy, Y. Weiss, and M. I. Jordan. Loopy belief propagation for approximate inference: An empirical study. In *Proceedings of the Fifteenth conference on Uncertainty in artificial intelligence*, pages 467–475, 1999.

[20] C. Rother, V. Kolmogorov, and A. Blake. Grabcut: Interactive foreground extraction using iterated graph cuts. *TOG*, 23(3):309–314, 2004.

[21] T. Serre, L. Wolf, S. Bileschi, M. Riesenhuber, and T. Poggio. Robust object recognition with cortex-like mechanisms. *PAMI*, 29(3):411–426, 2007.

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

[23] A. Toshev, B. Taskar, and K. Daniilidis. Shape-based object detection via boundary structure segmentation. *IJCV*, 99(2):123–146, 2012.

[24] A. van der Schaaf and J. H. van Hateren. Modelling the power spectra of natural images: statistics and information. *Vision research*, 36(17):2759–2770, 1996.

[25] L. Xu and J. Jia. Two-phase kernel estimation for robust motion deblurring. In *ECCV*, pages 157–170, 2010.

[26] Q. Yan, L. Xu, J. Shi, and J. Jia. Hierarchical saliency detection. In *CVPR*, 2013.

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