

Learning Good Regions to Deblur Images

Zhe Hu¹ · Ming-Hsuan Yang¹

Received: 15 March 2014 / Accepted: 25 March 2015
© Springer Science+Business Media New York 2015

Abstract The goal of single image deblurring is to recover both a latent clear image and an underlying blur kernel from one input blurred image. Recent methods focus on exploiting natural image priors or additional image observations for deblurring, but pay less attention to the influence of image structure on estimating blur kernels. What is the useful image structure and how can one select good regions for deblurring? We formulate the problem of learning good regions for deblurring within the conditional random field framework. To better compare blur kernels, we develop an effective similarity metric for labeling training samples. The learned model is able to predict good regions from an input blurred image for deblurring without user guidance. Qualitative and quantitative evaluations demonstrate that good regions can be selected by the proposed algorithms for effective single image deblurring.

Keywords Image deblurring · Conditional random field · Kernel similarity · Region inference

Communicated by Cordelia Schmid.

Electronic supplementary material The online version of this article (doi:[10.1007/s11263-015-0821-1](https://doi.org/10.1007/s11263-015-0821-1)) contains supplementary material, which is available to authorized users.

✉ Ming-Hsuan Yang
mhyang@ucmerced.edu

Zhe Hu
zhu@ucmerced.edu

¹ School of Engineering, University of California at Merced,
Merced, CA 95343, USA

1 Introduction

Motion blur on an image often results from the relative motion between a camera viewpoint and the scene (e.g., caused by camera shake) during exposure time. It causes significant image degradation, especially in low light conditions where longer exposure time is required. Recovering the latent image from one single blurred image has been studied extensively in recent years. Typically, the blurred image formation process is modeled as a latent image convolved with a spatial-invariant blur kernel (i.e., a point spread function). Hence, the deblurring process is known as a two-dimensional deconvolution problem. When the underlying blur kernel is known or has been accurately estimated, the problem is reduced to non-blind deconvolution. On the other hand, if the blur kernel is unknown, the deblurring problem is known as blind deconvolution. The ill-posed nature of the single image deblurring setting makes the problem rather difficult.

To deblur an image, it has been shown that estimating the blur kernel first and then solving a non-blind deconvolution problem with the estimated kernel renders favorable results (Levin et al. 2009). For the single image deblurring problem, one would intuitively think that it is better to make full use of the input blurred image. However, not all pixels of the input blurred image are informative. Smooth regions, for example, do not contribute much for estimating the blur kernel. In this paper, we ask “What kind of image features or structures of a blurred image help effectively in kernel estimation?” By detecting good image features for blur kernel estimation, an accurately estimated blur kernel can then be used to recover a latent clear image with high visual quality.

Recent studies have shown that regions with strong edges tend to yield better deblurring results (Fergus et al. 2006; Levin et al. 2009). Some of the gradient-based methods favor salient edges with gradients of specific patterns (Joshi et al.



Fig. 1 Different regions lead to different kernel estimations and deblurred results. The *top left* image is the input blurred image with three sub-windows selected for estimating kernels. The other three images

are the recovered images and estimated kernels from these three sub-windows using [Fergus et al. \(2006\)](#)

2008; [Cho et al. 2010](#); [Xu and Jia 2010](#); [Bae et al. 2012](#); [Yousaf and Qin 2013](#)). On the other hand, based on one-dimensional signal examples, it is demonstrated that edges of short length could adversely affect the deblurring results ([Xu and Jia 2010](#)). In other words, if the whole image is used for image deblurring without deliberate selection of good features, negative impacts are likely to lead to inferior results. For this reason along with improving computational efficiency, it is preferable to select a region, rather than the whole image, for estimating blur kernels. [Figure 1](#) illustrates that different regions may lead to completely different kernel estimation results, and thereby different recovered images. This problem can often be partly alleviated by manual selection and visual inspection of the results. However, this requires tedious human inputs for deblurring images. In addition, the questions regarding which regions or what image structures are crucial for accurate blur kernel estimation remain unanswered.

In this paper, we address these questions for effective and efficient image deblurring.¹ We first propose a metric that measures the similarity between kernels quantitatively, which facilitates the process of labeling good estimated kernels. Instead of determining good image structures from empirical understanding and prior knowledge, we resort to learning for this task based on a collection of labeled data with the proposed kernel similarity measure. We pose the learning problem within the conditional random field (CRF) ([Laferty et al. 2001](#)) framework in order to exploit contextual constraints among image regions. In addition, we explore the importance of different features with structured output. We construct a dataset which covers a large variety of image structures and blur kernels following the technique described in [Levin et al. \(2009\)](#) for evaluation, and apply the learned models to select good image regions for deblurring. Experimental results demonstrate the effectiveness and efficiency of our approach for selecting good regions to deblur.

2 Related Work and Context

The problem of removing spatially invariant blur has been studied extensively and numerous algorithms have been proposed ([Fergus et al. 2006](#); [Raskar et al. 2006](#); [Levin 2006](#); [Tai et al. 2008](#); [Levin et al. 2009](#); [Cho and Lee 2009](#); [Xu and Jia 2010](#); [Cho et al. 2010, 2011](#)). Since blind deconvolution is an ill-posed problem, prior knowledge or additional information is often required for effective solutions. In the image deblurring literature, two types of additional information are often used: natural image priors and additional image observations.

One line of research focuses on exploring image priors for deblurring. In [Fergus et al. \(2006\)](#), the heavy-tailed gradient distribution of natural images is exploited as prior information. The mixture of Gaussians is used to fit the distribution of gradient magnitudes of natural images. The sparse gradient prior is also used to search for blur kernels in [Levin \(2006\)](#). In [Jia \(2007\)](#) a method is presented to exploit the underlying relation between motion blur and blurry object boundary, which is shown to facilitate better kernel estimation. In [Shan et al. \(2008\)](#), a deblurring algorithm is proposed in which prior knowledge regarding gradients of natural images is used with additional constraints on consistence of local smooth regions before and after blurring. The consistency constraints are shown to be effective in suppressing ringing effects. A method that uses sparsity constraints for both the blur kernel and the latent image in the wavelet domain is presented in [Cai et al. \(2009\)](#). In contrast to existing methods that exploit the heavy-tailed gradient distribution of natural images, an image restoration algorithm that applies adaptive priors based on texture contents is proposed in [Cho et al. \(2010\)](#). Experimental results on denoising and deblurring show that adaptive priors are important for deblurring results when patches at different image locations are manually selected.

Another line of research tackles image deblurring by leveraging additional image observations. With both a low-resolution video camera and a high-resolution still-image camera, an algorithm that utilizes both spatial and temporal information is proposed ([Ben-Ezra and Nayar 2003](#)) for effective image deblurring. On the other hand, noisy images also provide useful information for image deblurring. When

¹ Preliminary results of this work were presented in [Hu and Yang \(2012\)](#).

a pair of blurred and noisy images of the same scene are available, it has been shown that the blur kernel can be estimated using the sharp image structures in the noisy image (Yuan et al. 2007).

Numerous studies focus on exploiting additional information, e.g., priors and image sequences to facilitate image deblurring. Considerably less attention has been paid to exploiting image structure for kernel estimation and deblurring. By defining a pixel-wise measure in terms of non-straightness and edge size, an algorithm is proposed to select a few image patches for kernel estimation (Bae et al. 2012). In this paper, we aim to determine useful image structures for kernel estimation using learning techniques.

Recent deblurring algorithms can be categorized into four groups based on maximum a posteriori (MAP), variational Bayesian, edge prediction, and power spectrum methods. The MAP based methods (Shan et al. 2008; Cho et al. 2010; Krishnan et al. 2011; Xu et al. 2013) model the problem by maximizing the posterior probability with respect to the latent image and the blur kernel. The priors of the latent image and the blur kernel are incorporated as regularizations in the objective function. Since multiple solutions exist for the objective function, the success of these methods hinges on the choice of the regularizations. We note that MAP based methods may generate trivial solutions (i.e., delta kernel functions) (Levin et al. 2009) even with carefully designed priors. The variational Bayesian based methods (Fergus et al. 2006; Levin et al. 2009) approximate the posterior probability of the unobserved variables and compute the marginal likelihood. This approach has been shown to be theoretically more robust than the MAP based methods (Levin et al. 2009). The edge prediction based methods (Joshi et al. 2008; Cho and Lee 2009; Xu and Jia 2010) use explicit or implicit salient edge restorations for kernel estimations, which have been demonstrated as useful tools of alleviating delta kernel solutions with fast convergence. Despite demonstrated success (Köhler et al. 2012), these methods rely largely on heuristic edge selections for kernel estimations. The power spectrum based algorithms (Yitzhaky et al. 1998; Goldstein and Fattal 2012; Hu et al. 2012) are developed based on the observation that the power spectra of the sharp natural images can be well modeled by statistical regularities. It has been shown that the power spectrum of a blur kernel can be estimated from that of a blurry image. Phase retrieval techniques are used to estimate the phase component of the power spectrum which require a selected region containing a distribution of diverse edge orientations for kernel estimations.

Removing spatially variant blur is a challenging task since different regions within the image are degraded differently. This problem has attracted more attention nowadays due to its wide range of practical applications (Bardsley et al. 2006; Cho et al. 2007; Shan et al. 2007; Tai et al. 2008; Joshi et al. 2010; Whyte et al. 2010; Gupta et al. 2010; Hirsch et al. 2011;

Tai et al. 2011). Recently, a geometric model has been proposed to model the observed blurry image as the integration of all the intermediate images captured by the camera along the motion trajectory (Whyte et al. 2010; Tai et al. 2011). These intermediate images are modeled as the transformed ones (i.e., homographies) of the sharp image. Based on this model, image blur caused by camera motion can be well formulated as an optimization problem. It is also possible to remove spatially varying blur by estimating a general camera motion function. A similar model has been used to model three degrees of camera motion (with in-plane translation and rotation) in Gupta et al. (2010). In Hirsch et al. (2011), Hu and Yang (2012), fast patch-based non-uniform deblurring methods have been developed to speed up the optimization step. The back projection technique has been used to initialize the camera motion from estimated kernels of several image patches in Hu and Yang (2012). In this work, we show that our proposed region selection algorithm can be used for patch-based non-uniform image deblurring.

3 Kernel Similarity

Existing methods mostly resort to visual quality of deblurred images for empirical evaluation. While it is important to recover high visual quality images, it is neither reliable nor effective to evaluate recovered results visually since human vision is sensitive to noise and ineffective in telling subtle differences. As suggested in Levin et al. (2009), it is preferable to separate the image deblurring problem into two steps: first blur kernel estimation and then non-blind deconvolution. If the blur kernel can be accurately estimated, then the deblurred image can be recovered with non-blind deconvolution algorithms. Therefore, the ensuing question is how to identify kernels effectively.

The difficulty of comparing kernels in the image deblurring task arises when kernels vary in terms of shift and support size. Two kernels K_1 and K_2 are considered shift and support size invariant if the dominant (i.e., non-zero) parts of them are the same, regardless of differences in locations and sizes of the kernel windows. Thus, a good metric for kernel similarity should be shift and support size invariant. Figure 2 shows three kernels that vary in shift and support size.

The commonly used root mean square error (RMSE) metric is not effective in computing the similarity between two kernels. Typically, the RMSE metric favors smooth estimated kernels (comparing with the ground-truth), due to the use of Euclidean-norm and the fact that the entries of the blur kernel sum up to one. To deal with the aforementioned problems, we propose a kernel similarity metric to effectively compare estimated kernels with the ground truth. We compute the maximum response of normalized cross-correlation to represent the blur kernel similarity (KS) $S(K, \hat{K})$ of two kernels,

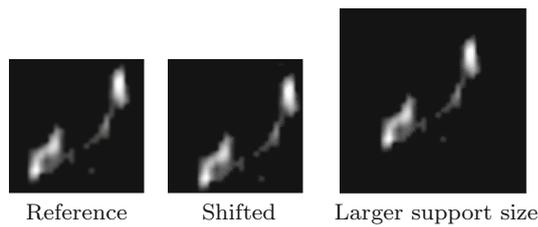


Fig. 2 Estimated kernels vary in shift and support size, but the dominant parts are the same. These kernels should be considered the same by a good kernel measurement in the image deblurring task

K and \hat{K} ,

$$S(K, \hat{K}) = \max_{\gamma} \rho(K, \hat{K}, \gamma), \quad (1)$$

where $\rho(\cdot)$ is the normalized cross-correlation function and γ goes through the possible shifts between the two kernels. With τ representing image coordinates, $\rho(\cdot)$ is given by

$$\rho(K, \hat{K}, \gamma) = \frac{\sum_{\tau} K(\tau) \cdot \hat{K}(\tau + \gamma)}{\|K\| \cdot \|\hat{K}\|}, \quad (2)$$

where $\|\cdot\|$ is the Euclidean norm, $K(\tau)$ and $\hat{K}(\tau)$ are zeros when τ is out of the kernel range. The maximum response of the normalized cross-correlation, similar in nature to the convolution operator, can handle the shift and support size problems mentioned above. With this metric, larger kernel similarity values reflect more accurate kernel estimation results. To demonstrate the effectiveness of the proposed kernel similarity metric, we compare it with the RMSE of the kernels shown in Figs. 3 and 4. We note that RMSE measures the reconstruction error and KS measures the similarity. In Fig. 3, the kernel in (c) is closer to the ground truth kernel compared with the kernel in (b) and the kernel in (d), since the kernel in (b) is over-smoothed and the kernel in (d) misses some parts. However the RMSE of the kernel in (c) is larger than the other two and thus demonstrates that low RMSE does not necessarily indicate good results.

For Fig. 4, the kernel in (c) and the kernel in (d) maintain more kernel structures and are more similar to the ground

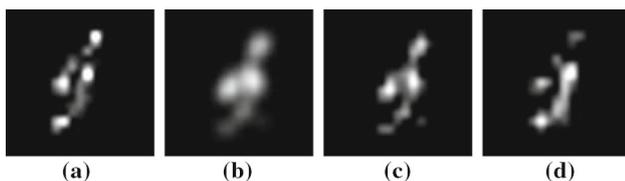


Fig. 3 Comparison on root mean square error and kernel similarity. **a** The ground truth blur kernel. **b** The smooth estimated kernel. **c** The estimated kernel has an extra part. **d** The estimated kernel misses some parts. The kernel in **c** is closer to the ground truth kernel. Although the RMSE of the kernel in **c** is larger than the other two estimated kernels, the KS value shows that it is closer to the ground truth

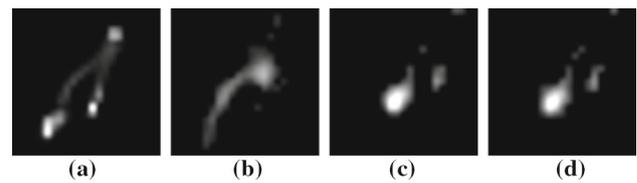


Fig. 4 Comparison of root mean square error and kernel similarity. **a** The ground truth blur kernel. **b** The estimated kernel is far away from the ground truth kernel. **c, d** Two similar estimated kernels with some missing components. The kernels in **c** and **d** are similar to the ground truth kernel. Although the RMSE of these three kernels are similar, the KS metrics show that the kernel in **c** is more similar to the ground truth one

truth. In this case, the RMSEs of the estimated kernels are of the same magnitude while the KS values of the kernel in (c) and the kernel in (d) are higher than those of the kernel in (b). These two examples illustrate that the proposed KS is a better metric for comparing kernels.

4 Learning Good Regions to Deblur

Existing deblurring methods discuss some potential features that may be useful for kernel estimation, mainly based on empirical experimental results. In this paper, we address this problem by learning good image regions for deblurring.

4.1 Learning Framework

To determine the good image regions for deblurring, we analyze the image structure by small subwindows. The subwindows within an image, in the context of kernel estimation and recovered image, are spatially dependent. Two closely overlapping subwindows (e.g., shifted by a few pixels in either directions) share similar image structures. Thus, it is reasonable to expect that other subwindows, nearby a potential good subwindow for kernel estimation, contain useful image structures for deblurring. Consequently, the deblurred results based on these subwindows should be similar, which is also observed empirically in image deblurring results. Figure 5 shows one example where we estimate a blur kernel from each subwindow of size 200×200 and apply it to recover the whole image. With all the estimated kernels from subwindows and recovered images, we construct an image reconstruction error map (Fig. 5b) and a kernel similarity map (Fig. 5c) by comparing them with the ground truth image and blur kernel. The value at each pixel of these maps is computed by averaging the reconstruction errors or kernel similarity values from all the subwindows containing it. The image reconstruction map illustrates that deblurred results using subwindows for kernel estimation are spatially correlated. At the same time, high kernel similarity values match

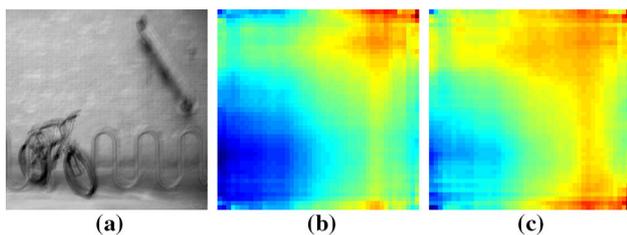


Fig. 5 Spatial correlation. **a** Input blurry image. **b** Image reconstruction error map built upon the estimated kernels from shifting sub-windows (blue to red pixels indicate low to high reconstruction errors compared with the ground truth image). **c** Kernel similarity map (blue to red pixels indicate high to low kernel similarity compared with the ground truth kernel) (Color figure online)

low image reconstruction errors well which demonstrates the effectiveness of kernel similarity as a metric for evaluating deblurred results. We pose the problem of learning good regions within the CRF framework (Lafferty et al. 2001) as it encourages spatial correlation, and label the training data using the proposed kernel similarity metric.

Let S and i represent the set of nodes and node index. Given the labels $\mathbf{y} = \{y_i\}_{i \in S}$ and the observations \mathbf{x} , the conditional distribution $P(\mathbf{y}|\mathbf{x})$ is

$$P(\mathbf{y}|\mathbf{x}) = \frac{1}{Z} \exp(E(\mathbf{y}|\mathbf{x})), \quad (3)$$

where Z is a normalization term also known as the partition function. The energy E is

$$E(\mathbf{y}|\mathbf{x}) = \sum_{i \in S} A_i(y_i, \mathbf{x}) + \sum_{i \in S} \sum_{j \in \mathcal{N}_i} I_{ij}(y_i, y_j, \mathbf{x}), \quad (4)$$

where A_i and I_{ij} denote the association (unary) and interaction (binary) potentials, respectively. The association potential $A_i(y_i, \mathbf{x})$ measures how likely the node of index i is labeled as y_i given the observation \mathbf{x} . Meanwhile, the interaction potential $I_{ij}(y_i, y_j, \mathbf{x})$ determines how the label y_j at node j affects y_i at node i . Here \mathcal{N}_i represents the neighborhood of node i .

In this paper, the sub-windows in the input blurred image are considered as nodes in the CRF model similar to the formulation of a discriminative random field (Kumar and Hebert 2006). The label y_i is set to be 1 to indicate a good sub-window for deblurring or -1 otherwise. Hence, we formulate the association potential as the log likelihood of a local discriminative model with the logistic function,

$$A_i(y_i, \mathbf{x}) = \log P_1(y_i|h_i(\mathbf{x})), \quad (5)$$

where $h_i(\cdot)$ denotes the feature vector of the local region at node i and the first element is set to 1 to accommodate the bias term. The conditional probability $P_1(y_i|h_i(\mathbf{x}))$ of class y_i at node i is defined based on the logistic function:

$$P_1(y_i|h_i(\mathbf{x})) = \sigma(y_i \mathbf{w}^\top h_i(\mathbf{x})), \quad (6)$$

where \mathbf{w} are parameters of the logistic function $\sigma(\cdot)$.

Similar to the association potential, the interaction potential is formulated by

$$I(y_i, y_j, \mathbf{x}) = \log P_2(y_i, y_j|\mu_{ij}(\mathbf{x})), \quad (7)$$

with

$$P_2(y_i, y_j|\mu_{ij}(\mathbf{x})) = \sigma(y_i y_j \mathbf{v}^\top \mu_{ij}(\mathbf{x})), \quad (8)$$

where \mathbf{v} are the parameters of the logistic function and μ_{ij} denotes the feature vector for the pair (i, j) . We use the difference of feature vector h between node i and j , with 1 as the first element, to express $\mu_{ij}(\mathbf{x}) = [1, |h_i(\mathbf{x}) - h_j(\mathbf{x})|]^\top$. Since we do not encourage negative interaction for two nodes of different appearance or at image discontinuities, the term $\mathbf{v}^\top \mu_{ij}(\mathbf{x})$ is set to be non-negative. That is, we use the value $\max(0, \mathbf{v}^\top \mu_{ij}(\mathbf{x}))$ to substitute $\mathbf{v}^\top \mu_{ij}(\mathbf{x})$.

4.2 Image Features

Numerous methods have shown that smooth image regions do not provide sufficient information for kernel estimation, and instead textured regions are often selected. Nevertheless, the estimation results may still be poor even when textured regions are used (Cho et al. 2010). Indeed, regions full of repetitive edges sometimes make no contributions to the solution of the problem when the blur movement occurs in a similar direction to that of the edges. We provide an example in Fig. 6 in which the blur movement (trajectory of camera movement) occurs in a similar direction to the edge orientations. As shown in Fig. 6, the stripes on the table cloth and scarf, which have the same direction as the blur motion, remain almost the same before and after blurring. In this case, the monotonous textures do not contribute to the kernel estimation.

For blur kernel estimation, recent algorithms focus on the use of sharp edges or edge distribution (Joshi et al. 2008; Cho et al. 2010). Analogous to the problems with textured regions, sharp edges can be of great importance for image deblurring under proper assumptions. The underlying assumption for effective use of sharp edges is that regions with high contrast in the original image retain informative structure after having been blurred. However, not all the sharp edges are effective for kernel estimation. Recently, it has been shown in Xu and Jia (2010) that edges of smaller size than the blur kernel may have adverse effects on kernel estimation and should be excluded for kernel estimation.

Taking all these factors into consideration, we present a method to extract features from regions for image deblurring. We use the responses of a Gabor filter bank $f(\mathbf{x}) =$

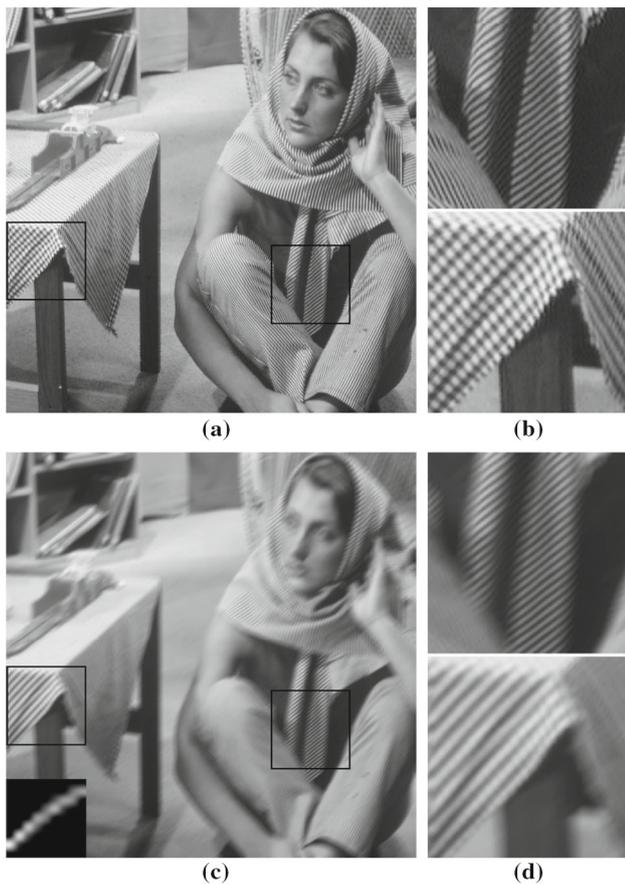


Fig. 6 Textured regions are not always effective for kernel estimation. **a** The latent image. **b** The close-up view on the two marked regions in the latent image. **c** The blurred image and corresponding blur kernel. **d** The close-up view on the two marked regions in the blurred image

$[f_1(\mathbf{x}), f_2(\mathbf{x}), \dots, f_n(\mathbf{x})]$ to represent the oriented textures of an image region \mathbf{x} . Here n denotes the number of applied Gabor filters and $f_i(\mathbf{x})$ represents the proportion that the i th orientation is the dominant direction within the observation \mathbf{x} . The image gradient histogram $g(\mathbf{x})$ is used to capture the distribution of edges along vertical and horizontal directions. We note that both $f(\mathbf{x})$ and $g(\mathbf{x})$ represent the proportion of the edge distribution. Therefore, the features of the same content x in different scales appear similarly as the edge proportion is maintained when scaling up/down.

To rule out potential negative effects from small edges, we use the mask $M(\mathbf{x}) = H(r(\mathbf{x}) - \tau)$ as suggested in [Xu and Jia \(2010\)](#), where $H(\cdot)$ is the Heaviside step function whose value is zero for negative arguments and one otherwise, and τ is the threshold. For each pixel $p \in \mathbf{x}$, $r(p)$ measures the usefulness of gradients by

$$r(p) = \frac{\| \sum_{q \in \mathcal{N}_s(p)} \nabla \mathbf{x}(q) \|}{\sum_{q \in \mathcal{N}_s(p)} \| \nabla \mathbf{x}(q) \| + 0.5}, \tag{9}$$

where $\mathcal{N}_s(p)$ is a $s \times s$ window centered at pixel p .

The feature vector $h(\mathbf{x})$ is then formed by concatenating the above-mentioned local image features,

$$h(\mathbf{x}) = [f(\mathbf{x}), g(\mathbf{x}), f(M(\mathbf{x})), g(M(\mathbf{x}))], \tag{10}$$

with varying parameters n , τ and s . We compare the proposed feature vectors with some alternatives in [Sect. 5.2.2](#).

4.3 Parameter Learning and Inference

Let θ denote the set of parameters in the CRF model, $\theta = \{\mathbf{w}, \mathbf{v}\}$. The maximum-likelihood estimates of model parameters are computed with the pseudo-likelihood $\hat{\theta}$ ([Besag 2006](#)) to approximate the partition function Z ,

$$\hat{\theta} = \arg \max_{\theta} \prod_m \prod_{i \in S_m} P(y_i^m | \mathbf{x}^m, \mathbf{y}_{\mathcal{N}_i^m}^m, \theta), \tag{11}$$

where m represents the index of the training image and S_m is the set of vertex indices in the graph generated from the m th image. Based on this formulation, we have

$$P(y_i | \mathbf{x}, \mathbf{y}_{\mathcal{N}_i}, \theta) = \frac{1}{z_i} \exp(A_i(y_i, \mathbf{x}) + \sum_{j \in \mathcal{N}_i} I(y_i, y_j, \mathbf{x})), \tag{12}$$

and the partition function can be written as

$$z_i = \sum_{y_i \in \{-1, 1\}} \exp(A_i(y_i, \mathbf{x}) + \sum_{j \in \mathcal{N}_i} I(y_i, y_j, \mathbf{x})). \tag{13}$$

To balance the effect between association and interaction potentials, we add in the penalty term $\frac{1}{2\phi^2} \mathbf{v}^\top \mathbf{v}$, where the variable ϕ is pre-defined in this work. We solve the optimization problem in the log pseudo-likelihood form,

$$\hat{\theta} = \arg \max_{\theta} \sum_m \sum_{i \in S_m} \left[\log \sigma(y_i \mathbf{w}^\top h_i(\mathbf{x})) + \log \sum_{j \in \mathcal{N}_i} \sigma(y_i y_j \mathbf{v}^\top \mu_{ij}(\mathbf{x})) - \log z_i \right] - \frac{1}{2\phi^2} \mathbf{v}^\top \mathbf{v}. \tag{14}$$

We use the Broyden–Fletcher–Goldfarb–Shanno (BFGS) method ([Fletcher 1970](#)) to solve the optimization problem and obtain parameter $\hat{\theta}$. The loopy belief propagation (LBP) algorithm ([Yedidia et al. 2003](#)) of sum-product form is utilized for inference.

4.4 Good Regions to Deblur

Given a high resolution blurred image (e.g., 4000×3000 pixels), it is rather time consuming to apply deblurring algorithms on the whole image. Even with the fast deblurring

algorithm (Cho and Lee 2009), it may not be the best choice to use the whole image for kernel estimation for the reasons discussed in Sect. 4.2. We demonstrate in Sect. 5 that kernel estimation using a region of blurred image renders better deblurred results rather than using the whole image. Clearly, one immediate solution for these problems is to select a region within the input image to estimate the blur kernel, and then apply a non-blind deconvolution algorithm to the whole image.

In this work, we use principal component analysis to reduce the dimensionality of the feature vectors for learning the model parameters, and LBP to infer the sub-windows as good regions or not for blur kernel estimation. The label of each node (sub-window) in a training image is determined by comparing the estimated kernel and the ground truth kernel. The learning process is summarized in the supplementary material. The reason we use LBP for inference instead of graph cuts here is to obtain labels with confidence values which facilitates region selection. We select the top ranked sub-window to estimate the kernel for simplicity although other weighted approaches may be used. Given the window size, the proposed method of selecting good sub-window to deblur does not require manual selection which can be problematic and time-consuming. We show that the proposed method can effectively select the optimal sub-window for deblurring in Sect. 5.

This window selection method can also be applied to patch-based non-uniform (spatially variant) image deblurring which requires large memory space and heavy computation if the whole image is used to estimate the camera motion (e.g., Whyte et al. 2010). For instance, the non-uniform deblur algorithm (Gupta et al. 2010) employs a RANSAC-based scheme to select a set of patches to estimate camera motion. Another deblurring algorithm for spatially invariant blur (Hu and Yang 2012) partitions the blurred image into patches and initializes the camera motion with the estimated kernels from the patches. With our method, it is easy and effective to choose a set of good patches as presented in Sect. 6.

5 Uniform Deblurring

We evaluate the inferred sub-windows using three state-of-the-art deblurring algorithms (Fergus et al. 2006; Shan et al. 2008; Cho and Lee 2009) for kernel estimation and compare the performance using the error metric introduced in Levin et al. (2009). This metric computes the difference between a recovered image I_r and the known ground-truth sharp image I_g , over the difference between the deblurred image I_{k_g} with the ground truth kernel k_g and the ground-truth sharp image as $\|I_r - I_g\|^2 / \|I_{k_g} - I_g\|^2$. In other words, the metric

describes how well the estimated kernel could reconstruct the image compared to the ground truth kernel. The success rate of reconstructed error ratio is then used to evaluate the efficiency of an algorithm. Since the dataset from Levin et al. (2009) has limited variability of image structure and blur kernel, we use the same technique to collect 960 blurred image using 15 sharp images (with image sizes around 450×450 pixels) and 64 blur kernels for training. These kernels are generated similarly to Levin et al. (2009) covering a wide range of camera shake and then expanded with different rotations. Furthermore, we construct another set of 120 challenging blurred images using 10 sharp images and 12 blur kernels for tests. Our experiments are carried out on a machine with a 3.40 GHz quad-core CPU and 16 GB RAM. To infer an image of 450×450 pixels, it takes around 5 seconds to process with our MATLAB implementation. The source code and more results are available at <http://eng.ucmerced.edu/people/zhu>.

5.1 Deblurring with Good Regions

With the CRF model described in Sect. 4, we learn the parameters using all the 960 training images for inferring good regions to deblur. For each image, we construct a graph model with overlapping sub-windows of 200×200 pixels as nodes and shifts of 20 pixels. The size of the sub-windows is determined empirically so that it is large enough to estimate the kernel of size smaller than 29×29 pixels and all the blur kernels in this data set are of the similar size. Given these parameters, we have at least 484 sub-windows per image as the nodes for training. As the variational Bayesian method has been shown to be more robust for kernel estimation than other approaches (Levin et al. 2009), we estimate the blur kernel from each sub-window using Fergus et al. (2006) in the training phase. With the estimated and the corresponding ground truth kernels, we label node i to be 1 or -1 using the proposed kernel similarity $S(K_i, K)$ between an estimated kernel K_i at node i and the ground truth kernel K with the threshold λ of 0.6. We determine the value of the threshold λ empirically, and details on the corresponding experiments can be found in the supplementary material. The feature parameters n , τ and s in this work are set to be 8, 0.6 to 0.9 with increment of 0.1, and 9 to 25 with increment of 4, respectively.

During the inference process, the size of sub-windows is set proportional to the user-defined kernel size. Both $f(\mathbf{x})$ and $g(\mathbf{x})$ in the proposed feature vector (10) represent the edge distribution in form of a normalized histogram. Thus, the feature vectors are scaled consistently to different region sizes. We verify this by inferring the good regions on an image of 450×450 pixels along with its upsampled image of 600×600 pixels. The sub-window size for the image is set to be 150×150 pixels, while that of its upsampled image is set to be 200×200 pixels correspondingly. The inferred regions

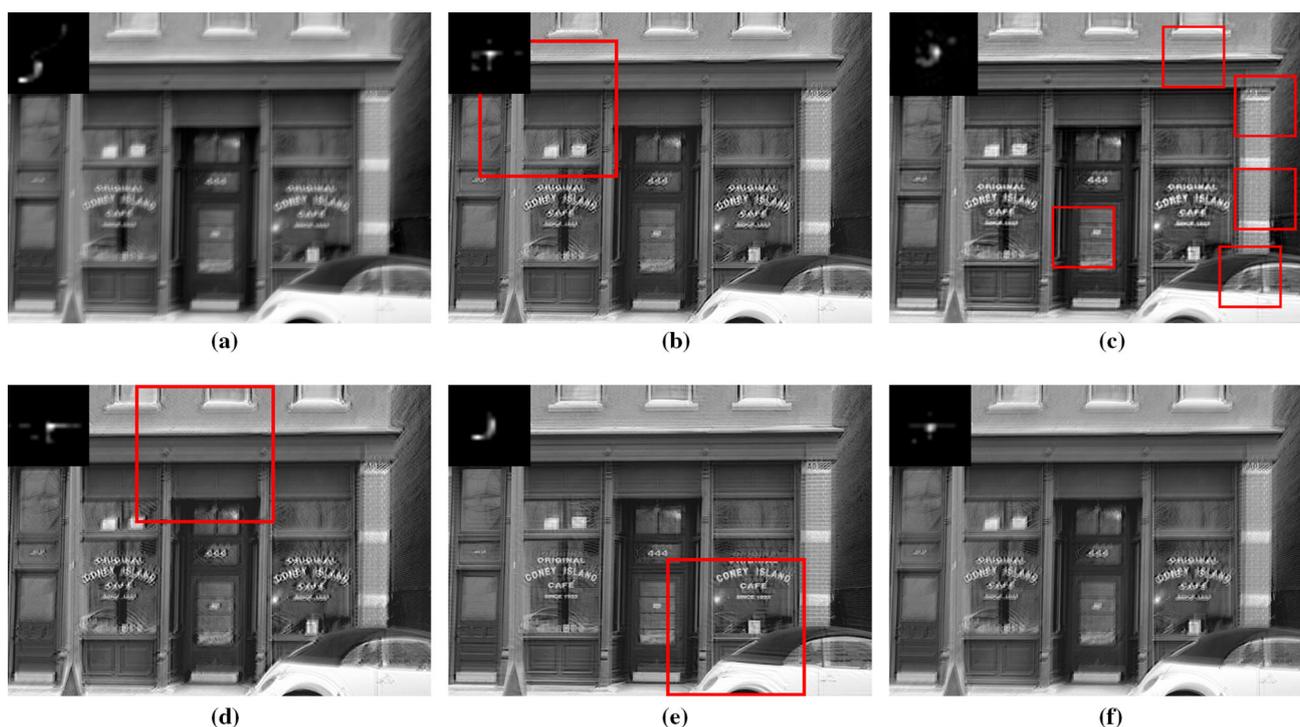


Fig. 7 Comparison of different region selection algorithms for deblurring using [Fergus et al. \(2006\)](#). **a** Input blurry image and corresponding kernel. **b** User-selected sub-window (*red box*) and the deblurred results. **c** Sub-window selected using [Bae et al. \(2012\)](#) and the deblurred results.

d Sub-window selected using [Fergus et al. \(2006\)](#) and the deblurred result. **e** Sub-window selected by our algorithm and the deblurred result. **f** Estimated kernel and deblurred result using the whole image (best viewed on a high-resolution display) (Color figure online)

of these two images by the trained model are at the corresponding locations, respectively. The reason for this is that the feature vectors of these two inferred regions are similar although there are slight differences due to the interpolation process. Thus, the choice of sub-window size does not significantly affect the efficiency of the trained model as long as the sub-window size is sufficiently large for kernel estimation.

5.1.1 Deblurring with User-Selected Regions

To demonstrate that manual selection can be replaced by the proposed algorithm, we compare the top inferred sub-windows with user-selected regions. Figures 7b and 8b show that users tend to choose regions with the most salient edges and variances. The manual selection method performs well in some images but usually requires several trials to obtain one good result. On the contrary, the proposed method does not require user guidance and the deblurred results using the inferred sub-windows of our algorithm outperform those that use user-selected regions.

5.1.2 Deblurring with Automatically Selected Regions

We compare the proposed algorithm with other region selection methods for deblurring ([Fergus et al. 2006](#); [Bae et al.](#)

[2012](#)). The automatic sub-window selector by [Fergus et al. \(2006\)](#) searches for a few regions with high variance and low saturation for kernel estimation. The sub-window selector by [Bae et al. \(2012\)](#) ranks each pixel of the downsampled image in terms of non-straightness and edge size, and searches for the regions to cover strong edges of homogeneous orientations. In this experiment, we select the top ranked sub-window from our inference algorithm for blur kernel estimation although other alternatives may be used. Once a sub-window is selected, we apply two state-of-the-art deblurring algorithms ([Fergus et al. 2006](#); [Shan et al. 2008](#)) to recover the latent image.² Figures 7 and 8 illustrate the comparison using the deblurring algorithm of [Fergus et al. \(2006\)](#) and [Shan et al. \(2008\)](#) respectively. We note that although some inferred windows (e.g., Fig. 8d, e) appear to be similar, their locations are different (60 pixels apart). The deblurring results on sub-windows inferred by our algorithm are better than those from the sub-window selection method ([Fergus et al. 2006](#)). The method ([Bae et al. 2012](#)) selects sub-windows based on the measurement of the central pixel, thus

² Since the method ([Bae et al. 2012](#)) adopts a patch-mosaic framework to combine several patches and estimate the kernel from them, we directly use the results from their released code. We note that their deblurring formulation is similar to the algorithm ([Shan et al. 2008](#)).

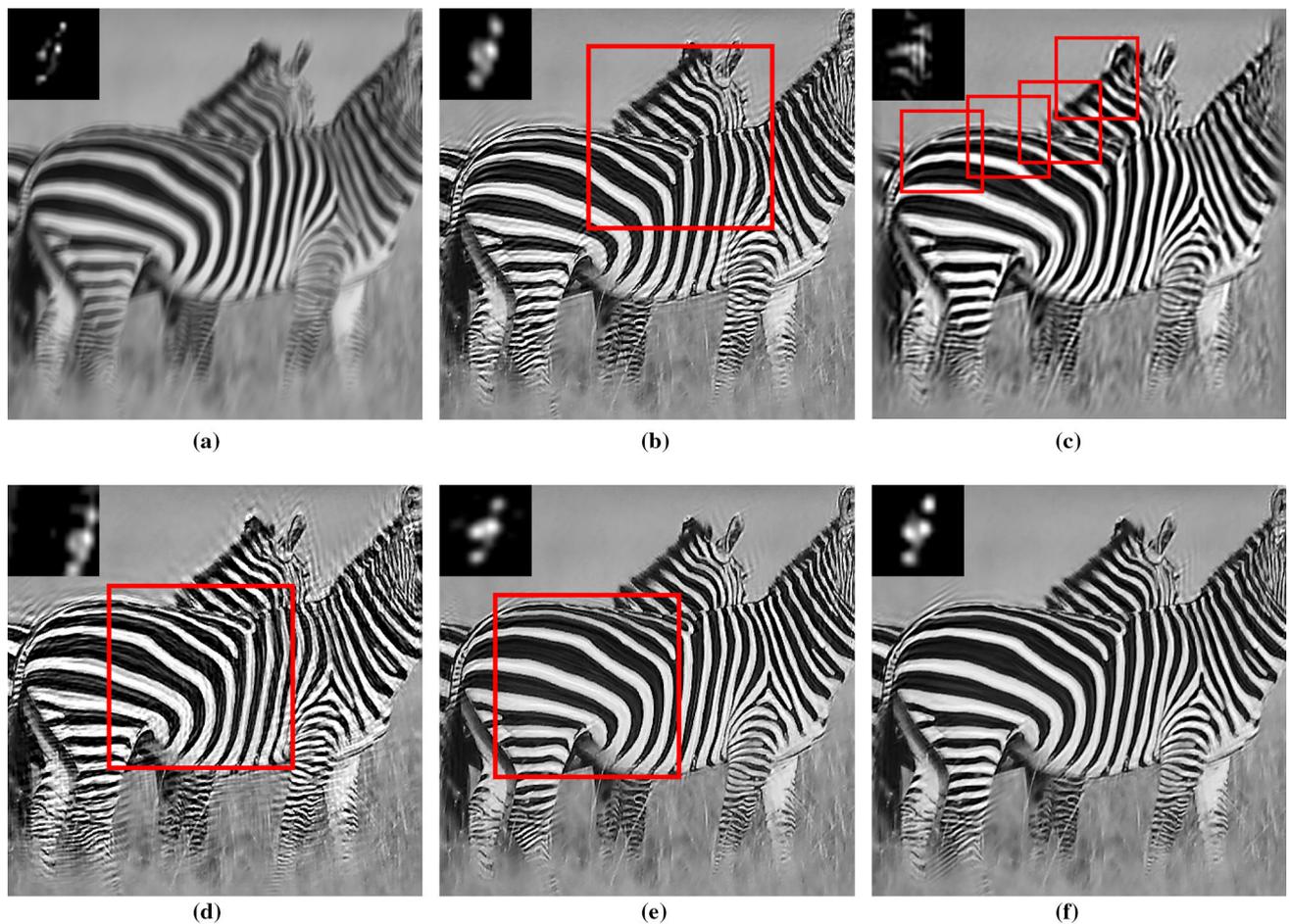


Fig. 8 Comparison on different region selection algorithms for deblurring using [Shan et al. \(2008\)](#). **a** Input blurry image and corresponding kernel. **b** User-selected sub-window (*red box*) and the deblurred results. **c** Sub-window selected using [Bae et al. \(2012\)](#) and the deblurred results.

d Sub-window selected by [Fergus et al. \(2006\)](#) and the deblurred result. **e** Sub-window selected by our algorithm and the deblurred result. **f** Estimated kernel and deblurred result using the whole image (best viewed on a high-resolution display) (Color figure online)

the selected sub-windows may cover smooth regions which are not informative for kernel estimation (Fig. 9).

5.1.3 Deblurring with Whole Images

We also compare with the deblurred results by [Cho and Lee \(2009\)](#) which uses the whole image for kernel estimation. In this experiment, we evaluate two region selection methods based on the inference results. One is to select the top ranked sub-window to estimate the blur kernel, and the other is to combine the top ten good sub-windows. To combine the top ten sub-windows, we choose the smallest rectangle which covers all the sub-windows as the region for simplicity. We note that the top ranked sub-windows are usually clustered due to the proposed CRF model which encourages spatial correlation, and thus the rectangular region is still of small size compared with the whole image as shown in Fig. 10. Figure 10 shows the deblurring results using synthesized images with known blur kernels. Figure 11 shows

the deblurring results using blurry images from [Fergus et al. \(2006\)](#). Compared to the deblurred results obtained from the whole images, the proposed algorithm generates comparable or superior kernel estimation and reconstructed images.

5.1.4 Quantitative Comparison

We conduct extensive comparisons using 120 challenging test images and present the success rate of reconstruction error ratio ([Levin et al. 2009](#)). Given a region inferred by the proposed algorithm, we use the fast algorithm ([Cho and Lee 2009](#)) to estimate a blur kernel, and the non-blind deconvolution algorithm ([Shan et al. 2008](#)) to recover the latent image (similar to [Cho and Lee 2009](#)). For thorough evaluations, we compare the results using the above-mentioned region selection methods ([Fergus et al. 2006](#)) and the whole image ([Cho and Lee 2009](#)). In addition, we compare with [Xu and Jia \(2010\)](#), [Goldstein and Fattal \(2012\)](#) using the whole image for kernel estimation. As shown in Fig. 14, the deblurring

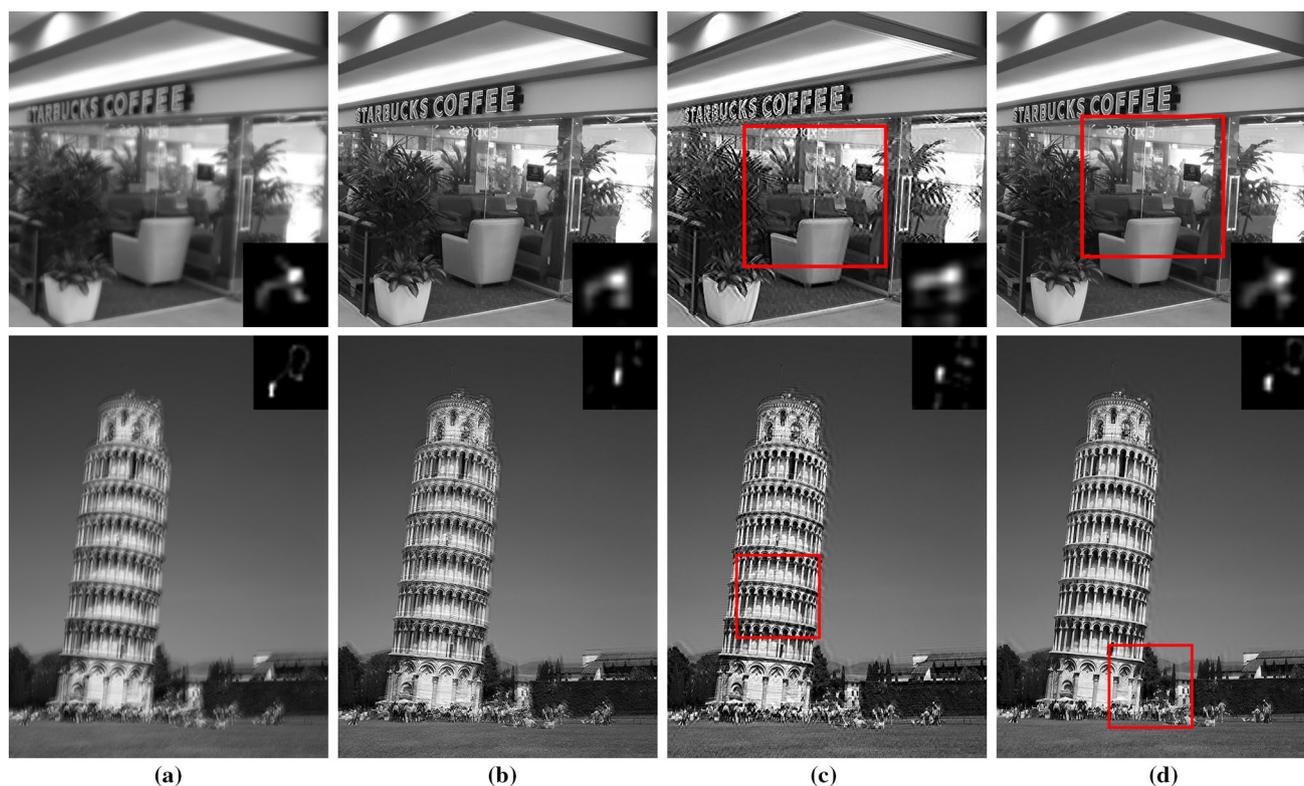


Fig. 9 Inferred top sub-windows for deblurring. We use the algorithm in [Cho and Lee \(2009\)](#) for kernel estimation and the algorithm in [Shan et al. \(2008\)](#) for non-blind deconvolution. **a** Input image. **b** The kernel estimation and deblurred result using the whole image. **c** The sub-

window (*red box*) selected by window selector ([Fergus et al. 2006](#)) and the deblurred result. **d** The sub-window selected by our algorithm and the deblurred result (best viewed on a high-resolution display) (Color figure online)

results using our top ten sub-windows are generally better than those of the other algorithms ([Cho and Lee 2009](#); [Xu and Jia 2010](#); [Goldstein and Fattal 2012](#)) using the whole image. The results also show that in a blurred image, not all the visual information is useful and using the whole image for kernel estimation may not lead to the best deblurring results.

We note that the reconstructed results are visually plausible even when the error ratio is around 5 (See examples in the supplementary material), which is different from the observations in [Levin et al. \(2009\)](#). The reason is that we employ a different non-blind deconvolution algorithm and larger test images.

5.1.5 Analysis

The favorable performance of the selected region over the whole image can be explained by the problem formulations for image deblurring. Most of the state-of-art deblurring algorithms adopt the two-step approach, which optimizes the blur kernel and the latent image in an alternating way by fixing another. With this formulation, if the blurry image contains many useless regions, it is easy to find a local minimum solution in kernel estimation when the whole image is used.

Taking the image in [Fig. 10](#) as an example, the majority of the image consists of smooth regions which remain almost the same before and after deblurring. When estimating the blur kernel by fixing the estimated latent image at one iteration, it is easy to reach a local minimum with low reconstruction errors due to dominant smooth regions. However, with deliberate selection of good regions for deblurring, this problem can be alleviated.

Our experimental results also provide insight regarding the types of image structures and features that are favored by different deblurring algorithms. In the *zebra* image, the selected region by our algorithm contains relatively strong edges as shown in [Fig. 8e](#), which is consistent to what is suggested in [Fergus et al. \(2006\)](#), [Levin et al. \(2009\)](#). However, small or detailed edges may not help kernel estimation as shown in [Fig. 8b–d](#). Furthermore, the sub-window with various oriented edges is selected rather than the regions with repetitive textures. The experimental results and observations bear out our intuition and analysis in [Sect. 4.2](#).

To provide further insight on which image patches constitute good regions for deblurring, we provide more results of inferred good regions in the same scene but different blur kernels in [Figs. 12 and 13](#). The inferred regions in the same scene are different as a result of different blur kernels.

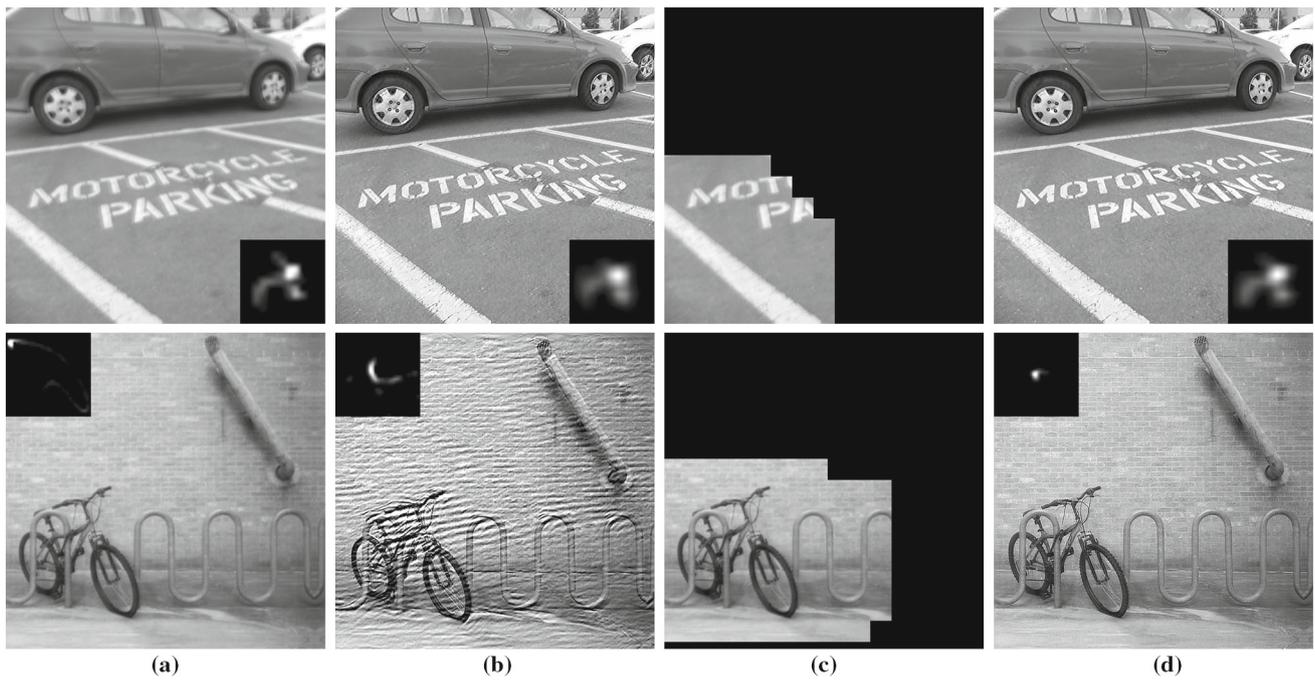


Fig. 10 Synthetic blurry images and deblurred results. **a** Input blurred image and the corresponding blur kernel. **b** The kernel estimation and deblurred result using the whole image. **c** The mask image indicating

the inferred good region. **d** The deblurred result using the good region to deblur (best viewed on a high-resolution display)

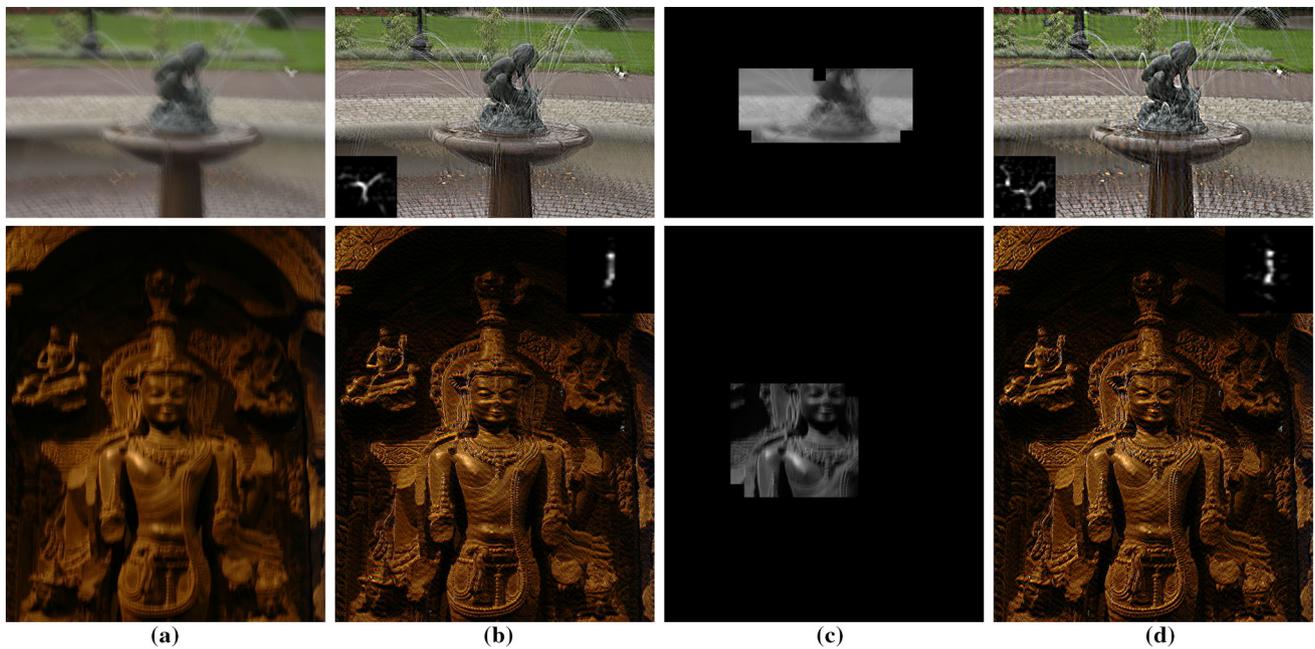


Fig. 11 Real blurry images and deblurred results. **a** Input blurred image and the corresponding blur kernel. **b** The kernel estimation and deblurred result using the whole image. **c** The mask image indicating

the inferred good region **d** the deblurred result using the good region to deblur (best viewed on a high-resolution display)

The primary drawback of the proposed method is that, for each blurred image, we need to specify a suitable subwindow size that is large enough for kernel estimation (usually proportional to the blur kernel size). However, this issue can be

addressed by employing methods to estimate the blur kernel size as suggested in [Jia \(2007\)](#), [Liu et al. \(2014\)](#). Furthermore, the proposed algorithm does not perform well for saturated images as the trained model might favor such regions. The

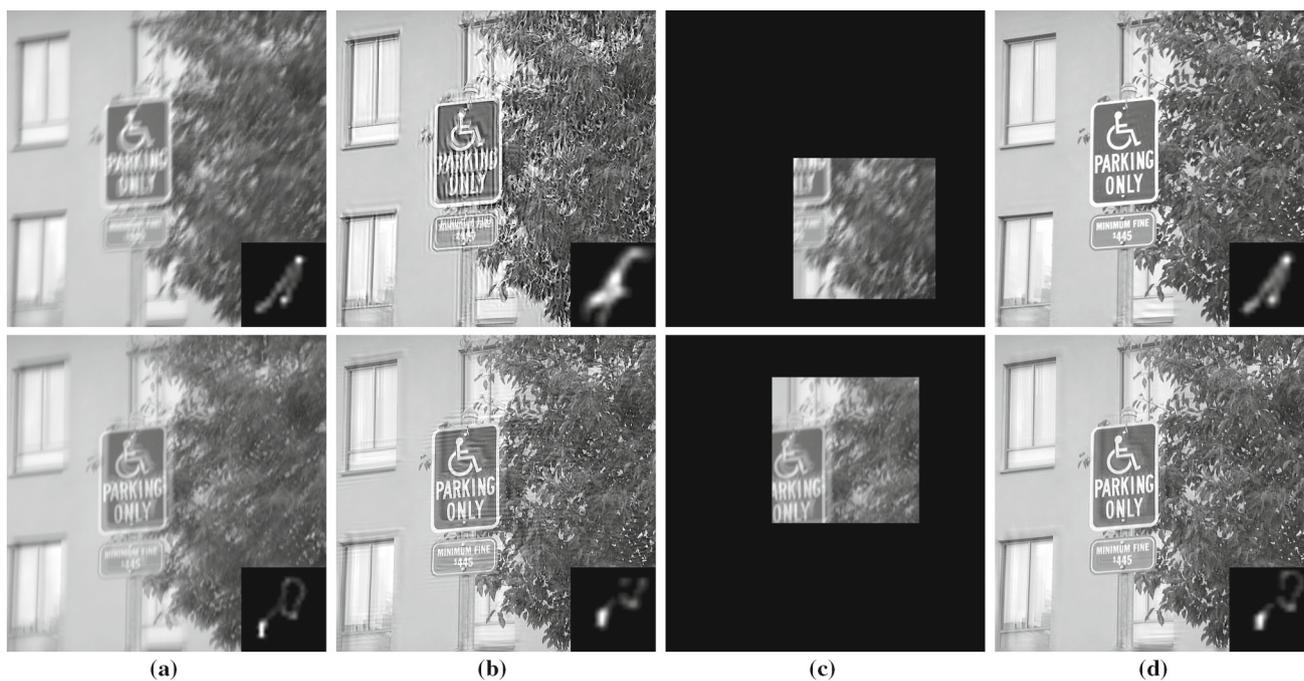


Fig. 12 Inferred top sub-windows from our algorithm are different for the images on the same scene but different blur kernels. **a** Input blurred image and corresponding blur kernel. **b** Estimated kernel and deblurred

result using the whole image. **c** Mask image showing the inferred top sub-window. **d** Deblurred result using the inferred region by the proposed algorithm (best viewed on a high-resolution display)

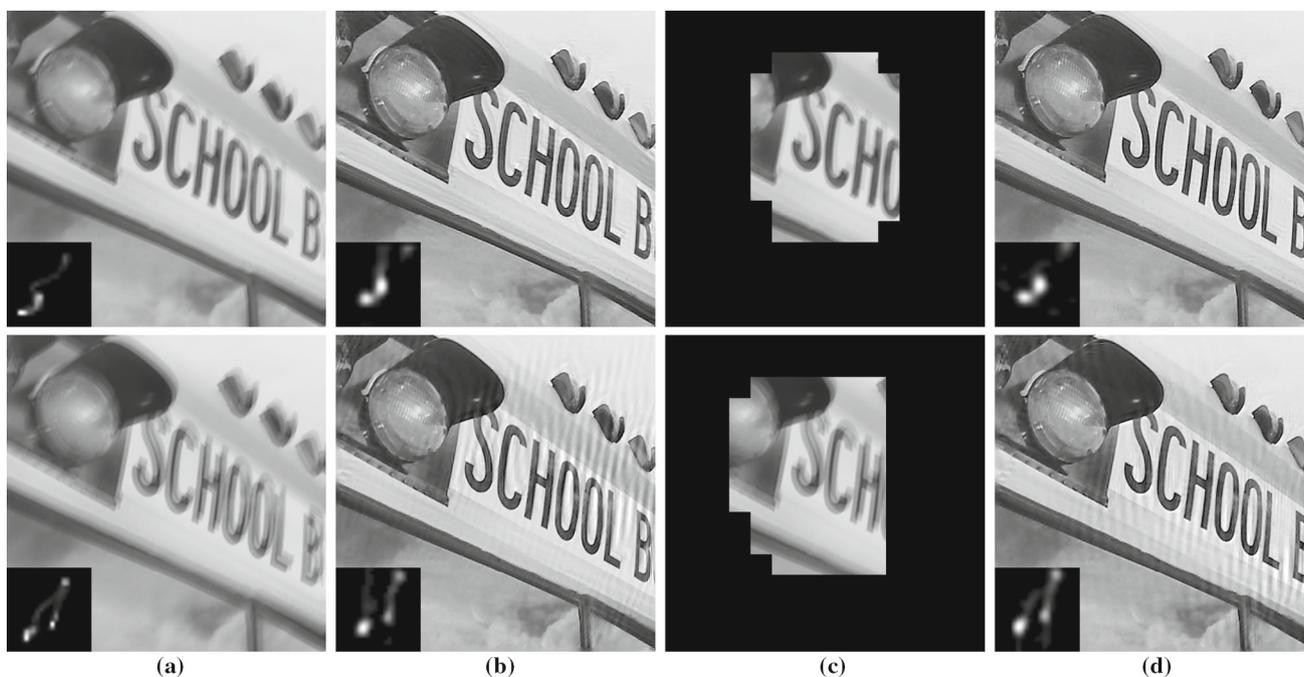


Fig. 13 Inferred top ten sub-windows from our algorithm are different for the images on the same scene but different blur kernels. **a** Input blurred image and corresponding blur kernel. **b** Estimated kernel and deblurred result using the whole image. **c** Mask image showing the

inferred good region consisting of top ten sub-windows. **d** Deblurred result using the inferred region by the proposed algorithm (best viewed on a high-resolution display)

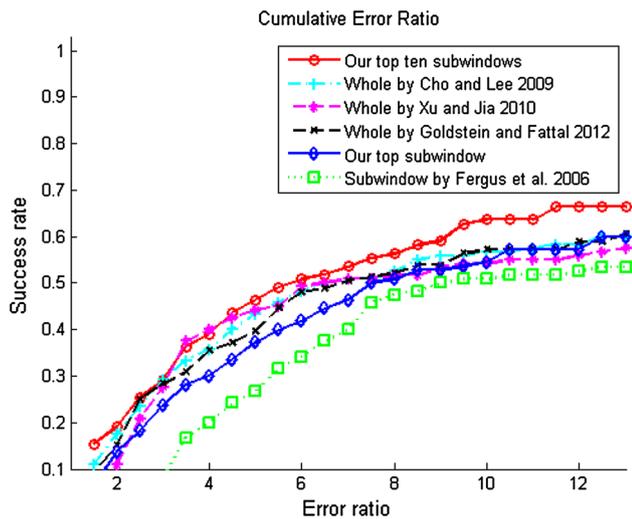


Fig. 14 Success rate of reconstructed error ratio (Levin et al. 2009) using different window selection methods including ours (Fergus et al. 2006; Xu and Jia 2010). We use the algorithm in Cho and Lee (2009) for kernel estimation and the algorithm in Shan et al. (2008) for non-blind deconvolution

reason is that saturated images contain a large number of salient edges caused by the contrast between the saturated and unsaturated regions. Those edges are significantly different from the training examples for a trained model to infer good regions. From the training perspective, the estimated results from the deblurring methods are used to label the data, the negative examples would affect the model. From the inference perspective, we could partially address the issue by excluding the saturated pixels in our feature extraction. However, we still need a deblurring method that handles saturated regions as most state-of-the-art methods do not estimate kernels well in such cases (Fig. 14).

5.2 Discussions

In this section, we discuss and address three factors in the training procedure: the method used to generate examples for training, the image features extracted from each image patch and the noise level of the data.

5.2.1 Deblurring Methods to Generate Examples

As we pose the problem in a learning framework, one question that arises is whether the trained model using the examples generated by one method can still infer good regions for other deblurring algorithms. To validate this, we use three representative methods based on MAP (Shan et al. 2008), variational Bayesian (Fergus et al. 2006), and edge prediction (Cho and Lee 2009) approaches. One model for each method is trained and used to infer regions on test images. We estimate kernels on these regions with all these

methods and evaluate the results by plotting the success rate of reconstruction error ratio (Levin et al. 2009) in Fig. 15. For fair comparisons, the non-blind deblurring method (Shan et al. 2008) is used to generate the final deconvolution results. As expected, the best results can be achieved when the same method is used in training and test stages, as shown in Fig. 15. The regions inferred by the model trained using the examples from one deblurring method produces similar results using different methods for kernel estimation. That is, the proposed region detection algorithm can be applied to other deblurring methods. Furthermore, the model trained using Fergus et al. (2006) is able to infer better regions comparing with other methods as the cumulative histograms in Fig. 15a are slightly higher than those in in Fig. 15b, c. We note that the MAP and edge prediction based methods (Shan et al. 2008; Cho and Lee 2009) sometimes do not estimate kernels well due to inconsistent and sudden change of estimated kernels for nearby sub-windows, which affects the reported performance of the trained models.

5.2.2 Image Features

We evaluate other features, e.g., the gradient histogram and the histogram of Gabor filter responses, that can be used to describe local image structures for kernel estimation. We follow the same process as we describe in Sect. 5.1.2, but replace the proposed features with the alternative features. Figure 16 shows the success rate of reconstructed error ratio using different features, and the deblurring results using the proposed features (Sect. 4.2) are better than those with different features. In addition, the proposed algorithm performs better than the deblurring method (Cho and Lee 2009) using the entire image for kernel estimation.

5.2.3 Noise Analysis

As the blurry image usually comes with signal noise, we evaluate the proposed method in terms of noise robustness. The experiments are conducted on training data and test data with Gaussian noise of different levels. That is, we learn our models from training data with Gaussian noise of different levels and then test the trained models on noisy test data. To exclude the aspect of imperfect deconvolution, we consider the accuracy of the estimated blur kernels in terms of KS and use the state-of-the-art deblurring method with special noise handling (Zhong et al. 2013) for kernel estimation. Table 1 shows the average KS from training and test data of different noise levels. We note that high noise level on test data would destroy the image structure which is used for kernel estimation in test phase, thus affects the average KS. Considering this fact, our trained models perform robustly on test data of noise level no larger than 0.8% in general. From the training data of noise level no larger than 1.0%, which is

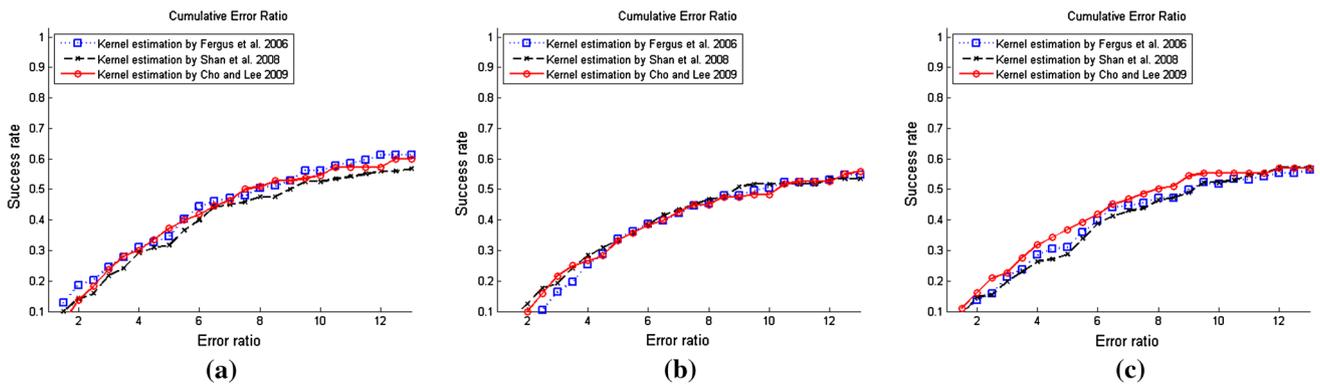


Fig. 15 Success rate of reconstruction error ratio (Levin et al. 2009) on the results of the models trained from different methods. **a** A model trained using examples generated by Fergus et al. (2006). **b** A model

trained using examples generated by Shan et al. (2008). **c** A model trained using examples generated by Cho and Lee (2009)

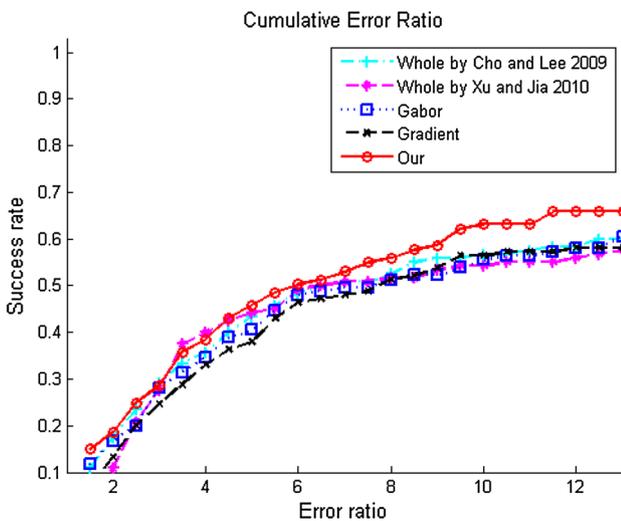


Fig. 16 Success rate of reconstruction error ratio (Levin et al. 2009) on deblurred results with different feature extraction approaches. We use the algorithm in Cho and Lee (2009) for kernel estimation and the algorithm in Shan et al. (2008) for non-blind deconvolution

not significantly noisy, the trained models perform similarly. While there is more noise on training data, the performance decreases fast. One reason is that noisy training data jeopardizes the kernel estimation and thus the labeling process and trained model in the training phase. When the noise level is below 1.0%, the deblurring method could estimate the kernel well and no significant problem is raised.

6 Non-uniform Deblurring

In this section, we show that the proposed region selection method can also be used in non-uniform deblurring problems. Recently, a geometric model has been proposed to represent the observed blurry image as the integration of all

intermediate images captured by the camera along a motion trajectory (Whyte et al. 2010; Gupta et al. 2010; Tai et al. 2011),

$$B = \sum_{\theta \in S} f(H_{\theta}, L)w_{\theta} + n = \sum_{\theta \in S} w_{\theta}(K_{\theta}L) + n, \quad (15)$$

where θ denotes the camera pose in a discrete space S , H_{θ} is the homography induced by camera pose θ , $f(H_{\theta}, L)$ represents the transformed image, and n is the observation noise. In addition, K_{θ} is the matrix that warps the latent image L to the one captured at the sampled pose θ , and w_{θ} is the corresponding weight. In this formulation, K_{θ} can also be viewed as a blur kernel basis (Hirsch et al. 2011) where row i is the weight vector representing how pixel i of the latent image L contributes to the blurry image. To solve the problem, a typical approach seeks to minimize the energy function over the variables L and W with pre-computed kernel basis K_{θ} ,

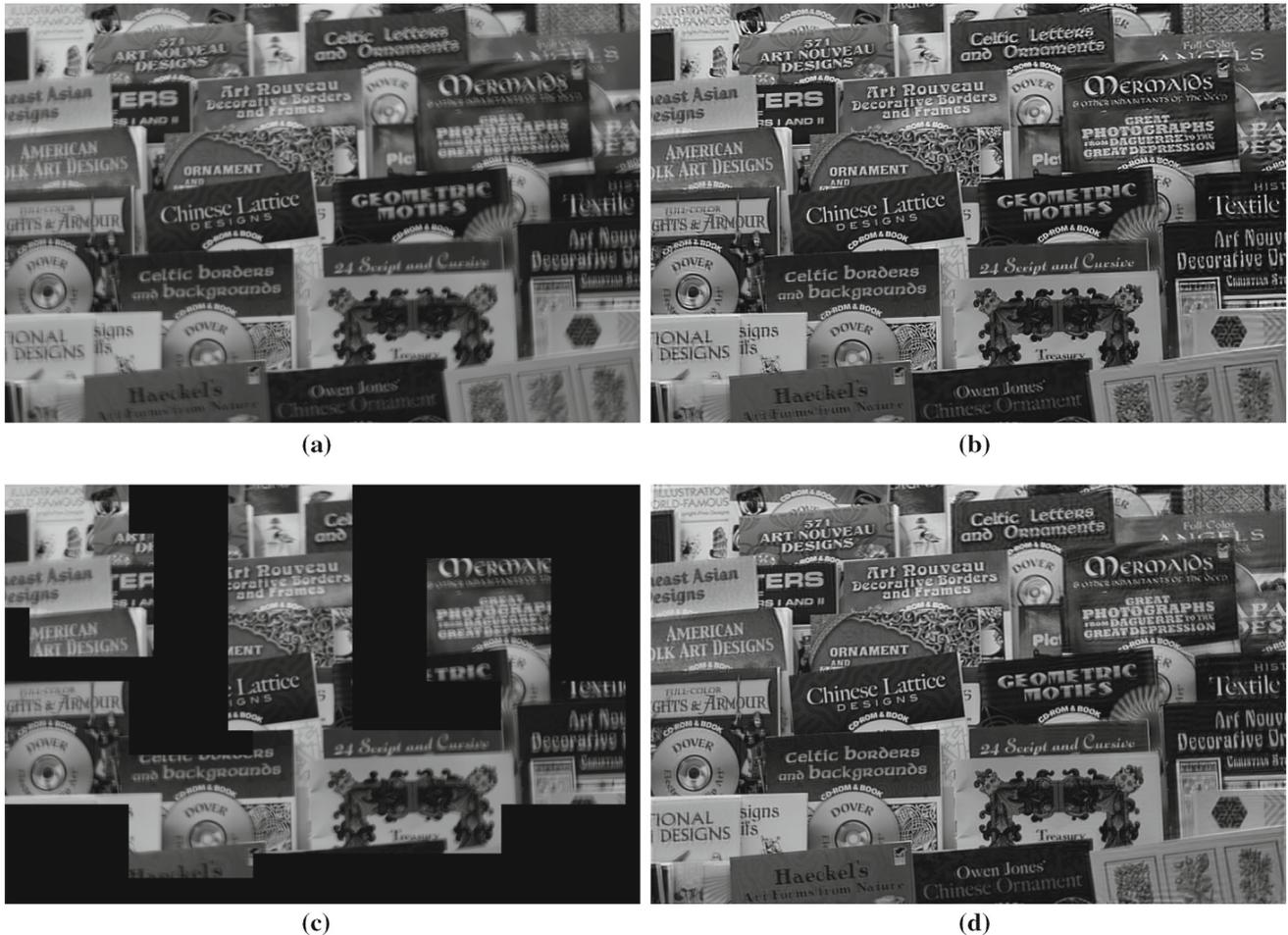
$$\min_{(L, W)} \left\| \left(\sum_{\theta \in S} w_{\theta} K_{\theta} \right) L - B \right\|^2 + \Phi_1(L) + \Phi_2(W), \quad (16)$$

where Φ_1 and Φ_2 represent the regularization terms on the latent image L and weights W respectively.

In order to solve the ill-posed problem, a good deconvolution method needs to deal with local minimums and multiple feasible solutions (i.e., pairs of sharp image and camera motion) for a given blurry image. Existing algorithms (Whyte et al. 2010; Hirsch et al. 2011) employ coarse-to-fine approaches to alleviate these problems, but require computing the kernel basis and estimating pose weights at each scale with heavy computational load. In Gupta et al. (2010), a uniform deblurring method is applied to small image patches to compute a good initial estimation of the latent image. In practice, not all the deblurred patches in this step give good results, since the kernel estimation on different patches vary

Table 1 Average kernel similarity (KS) on test data containing different Gaussian noise levels

Noise level	Test 0	Test 0.2 %	Test 0.4 %	Test 0.6 %	Test 0.8 %	Test 1.0 %
Training 0	0.7139	0.7195	0.7152	0.7034	0.6943	0.6560
Training 1.0 %	0.7197	0.7108	0.7130	0.6909	0.6675	0.6377
Training 2.0 %	0.6553	0.6617	0.6476	0.6332	0.6208	0.5811

**Fig. 17** Inferred patches for non-uniform deblurring based on Gupta et al. (2010). **a** Spatially variant blurry image. **b** Deblurred result using RANSAC based on Gupta et al. (2010). **c** Inferred good regions by

our algorithm. **d** Deblurred result using the inferred good patches (best viewed on a high-resolution display)

significantly as explained in Sect. 4. To avoid the problem of selecting best image patches for fitting the camera motion, a RANSAC-based scheme is utilized to choose a set of “good” patches for filtering the camera motion in Gupta et al. (2010).

To demonstrate the effectiveness of the proposed region selection approach for non-uniform deblurring, we first compare the results based on Gupta et al. (2010) with the results using the selected region. That is, we follow the same process as described except replacing the RANSAC scheme with the good region selected by the proposed algorithm. We select the regions such that the intersection of any two good regions is less than one half. Figure 17 shows the deblurred results of both approaches. The deblurred results using the

selected patches by the proposed algorithm are sharper than those based on the RANSAC scheme, which suggests the effectiveness of the proposed region selection approach for non-uniform image deblurring.

We also apply the proposed region selection algorithm to a fast non-uniform algorithm (Hu and Yang 2012), which initializes the camera motion based on the kernels computed from image patches. Using the same setup as described in Hu and Yang (2012), we compare the deblurring results using three different patch selection methods: all partitioned image patches, random selection of a set of patches, and the proposed region selection algorithm. The selected patches and deblurring results of these methods are presented in Fig. 18.



Fig. 18 Non-uniform deblurring results with different patch selection methods based on [Hu and Yang \(2012\)](#). **a, e** Spatially variant blurry image. **b, f** Deblurred result with image-partition patches. **c, g** Deblurred

result with randomly selected patches. **d, h** Deblurred result using the inferred good patches (best viewed on a high-resolution display)

The results using the proposed region selection method are sharper than the others, which indicates that more accurate blur kernels are estimated from the regions selected by the proposed algorithm and lead to better initialization of camera motion.

7 Conclusions

Recent methods focus on introducing priors or additional information to facilitate deblurring, and considerably less efforts are made to study structural information of the input blurred image itself. We address this issue and exploit informative image structure for effective and efficient image deblurring. In this paper, we propose a learning-based approach for selecting good features and good patches for blur kernel estimation. By detecting good image features for blur kernel estimation, an accurately estimated blur kernel can then be used to recover a latent clear image with high visual quality. Towards this, we first introduce a kernel similarity metric for effective comparison between kernels, which facilitates the process of labeling good estimated kernels. We formulate the learning problem within the Conditional Random Field framework in order to exploit contextual constraints among image regions. We construct a dataset which covers a large variety of image structures and blur kernels for evaluation, and apply the learned models to select good image regions for deblurring. The proposed algorithm selects good regions automatically for deblurring with efficient and favorable results, thereby relieving users from tedious trials for selecting image patches.

References

- Bae, H., Fowlkes, C. C., & Chou, P. H. (2012). Patch mosaic for fast motion deblurring. In *Proceedings of Asian conference on computer vision* (pp. 322–335).
- Bardsley, J., Jefferies, S., Nagy, J., & Plemmons, R. (2006). A computational method for the restoration of images with an unknown, spatially-varying blur. *Optics Express*, *14*(5), 1767–1782.
- Ben-Ezra, M., & Nayar, S. (2003). Motion deblurring using hybrid imaging. In *Proceedings of IEEE conference on computer vision and pattern recognition* (pp. 657–664).
- Besag, J. (2006). Statistical analysis of non-lattice data. *Journal of the Royal Statistical Society: Series D*, *24*(3), 179–195.
- Cai, J., Ji, H., Liu, C., & Shen, Z. (2009). Blind motion deblurring from a single image using sparse approximation. In *Proceedings of IEEE conference on computer vision and pattern recognition* (pp. 104–111).
- Cho, S., & Lee, S. (2009). Fast motion deblurring. In *Proceedings of ACM SIGGRAPH Asia*.
- Cho, S., Matsushita, Y., & Lee, S. (2007). Removing non-uniform motion blur from images. In *Proceedings of IEEE international conference on computer vision*.
- Cho, T. S., Joshi, N., Zitnick, C. L., Kang, S. B., Szeliski, R., & Freeman, W. T. (2010). A content-aware image prior. In *Proceedings of IEEE conference on computer vision and pattern recognition* (pp. 169–176).
- Cho, T. S., Paris, S., Horn, B. K. P., & Freeman, W. T. (2011). Blur kernel estimation using the radon transform. In *Proceedings of IEEE conference on computer vision and pattern recognition* (pp. 241–248).
- Fergus, R., Singh, B., Hertzmann, A., Roweis, S. T., & Freeman, W. T. (2006). Removing camera shake from a single photograph. In *Proceedings of ACM SIGGRAPH* (pp. 787–794).
- Fletcher, R. (1970). A new approach to variable metric algorithms. *The Computer Journal*, *13*(3), 317–322.
- Goldstein, A., & Fattal, R. (2012). Blur-kernel estimation from spectral irregularities. In *Proceedings of European conference on computer vision* (pp. 622–635).
- Gupta, A., Joshi, N., Zitnick, L., Cohen, M., & Curless, B. (2010). Single image deblurring using motion density functions. In *Proceedings of European conference on computer vision* (pp. 171–184).
- Hirsch, M., Schuler, C. J., Harmeling, S., & Schölkopf, B. (2011). Fast removal of non-uniform camera shake. In *Proceedings of IEEE international conference on computer vision* (pp. 463–470).
- Hu, W., Xue, J., & Zheng, N. (2012). Psf estimation via gradient domain correlation. *IEEE Transactions on Image Processing*, *21*(1), 386–392.
- Hu, Z., & Yang, M. H. (2012). Fast non-uniform deblurring using constrained camera pose subspace. In *Proceedings of British machine vision conference*.
- Hu, Z., & Yang, M. H. (2012). Good regions to deblur. In *Proceedings of European conference on computer vision* (pp. 59–72).
- Jia, J. (2007). Single image motion deblurring using transparency. In *Proceedings of IEEE conference on computer vision and pattern recognition*.
- Joshi, N., Kang, S. B., Zitnick, C. L., & Szeliski, R. (2010). Image deblurring using inertial measurement sensors. In *Proceedings of ACM SIGGRAPH* (p. 30).
- Joshi, N., Szeliski, R., & Kriegman, D. J. (2008). PSF estimation using sharp edge prediction. In *Proceedings of IEEE conference on computer vision and pattern recognition*.
- Köhler, R., Hirsch, M., Mohler, B., Schölkopf, B., & Harmeling, S. (2012). Recording and playback of camera shake: Benchmarking blind deconvolution with a real-world database. In *Proceedings of European conference on computer vision* (pp. 27–40).
- Krishnan, D., Tay, T., & Fergus, R. (2011). Blind deconvolution using a normalized sparsity measure. In *Proceedings of IEEE conference on computer vision and pattern recognition* (pp. 233–240).
- Kumar, S., & Hebert, M. (2006). Discriminative random field. *International Journal Computer Vision*, *68*(2), 179–201.
- Lafferty, J., McCallum, A., & Pereira, F. (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of international conference on machine learning* (pp. 282–289).
- Levin, A. (2006). Blind motion deblurring using image statistics. In *Neural information processing systems* (pp. 841–848).
- Levin, A., Weiss, Y., Durand, F., & Freeman, W. T. (2009). Understanding and evaluating blind deconvolution algorithms. In *Proceedings of IEEE conference on computer vision and pattern recognition* (pp. 1964–1971).
- Liu, S., Wang, H., Wang, J., Cho, S., & Pan, C. (2014). Automatic blur-kernel-size estimation for motion deblurring. *The visual computer*.
- Raskar, R., Agrawal, A., & Tumblin, J. (2006). Coded exposure photography: Motion deblurring using fluttered shutter. In *Proceedings of ACM SIGGRAPH* (pp. 795–804).
- Shan, Q., Jia, J., & Agarwala, A. (2008). High-quality motion deblurring from a single image. In *Proceedings of ACM SIGGRAPH* (pp. 73:1–73:10).

- Shan, Q., Xiong, W., & Jia, J. (2007). Rotational motion deblurring of a rigid object from a single image. In *Proceedings of IEEE international conference on computer vision*.
- Tai, Y., Tan, P., & Brown, M. S. (2011). Richardson-Lucy deblurring for scenes under projective motion path. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(8), 1603–1618.
- Tai, Y. W., Du, H., Brown, M. S., & Lin, S. (2008). Image/video deblurring using a hybrid camera. In *Proceedings of IEEE conference on computer vision and pattern recognition*.
- Whyte, O., Sivic, J., Zisserman, A., & Ponce, J. (2010). Non-uniform deblurring for shaken images. In *Proceedings of IEEE conference on computer vision and pattern recognition* (pp. 491–498).
- Xu, L., & Jia, J. (2010). Two-phase kernel estimation for robust motion deblurring. In *Proceedings of European conference on computer vision* (pp. 157–170).
- Xu, L., Zheng, S., & Jia, J. (2013). Unnatural l0 sparse representation for natural image deblurring. In *Proceedings of IEEE conference on computer vision and pattern recognition* (pp. 1107–1114).
- Yedidia, J. S., Freeman, W. T., & Weiss, Y. (2003). Understanding belief propagation and its generalizations. In *Exploring artificial intelligence in the new millennium* (pp. 236–239). Morgan Kaufmann Publishers Inc.
- Yitzhaky, Y., Mor, I., Lantzman, A., & Kopeika, N. (1998). Direct method for restoration of motion-blurred images. *Journal of the Optical Society of America*, 15(6), 1512–1519.
- Yousaf, S., & Qin, S. Y. (2013). Blur kernel optimization: A new approach to patch selection with adaptive kernel estimation. *Applied Mechanics and Materials*, 436, 531–538.
- Yuan, L., Sun, J., Quan, L., & Shum, H. (2007). Image deblurring with blurred/noisy image pairs. In *Proceedings of ACM SIGGRAPH*.
- Zhong, L., Cho, T. S., Metaxas, D., Paris, S., & Wang, J. (2013). Handling noise in single image deblurring using directional filters. In *Proceedings of IEEE conference on computer vision and pattern recognition*.