

# Segmentation-Free Dynamic Scene Deblurring

Tae Hyun Kim and Kyoung Mu Lee

Department of ECE, ASRI, Seoul National University, 151-742, Seoul, Korea

{lliger9, kyoungmu}@snu.ac.kr, <http://cv.snu.ac.kr>

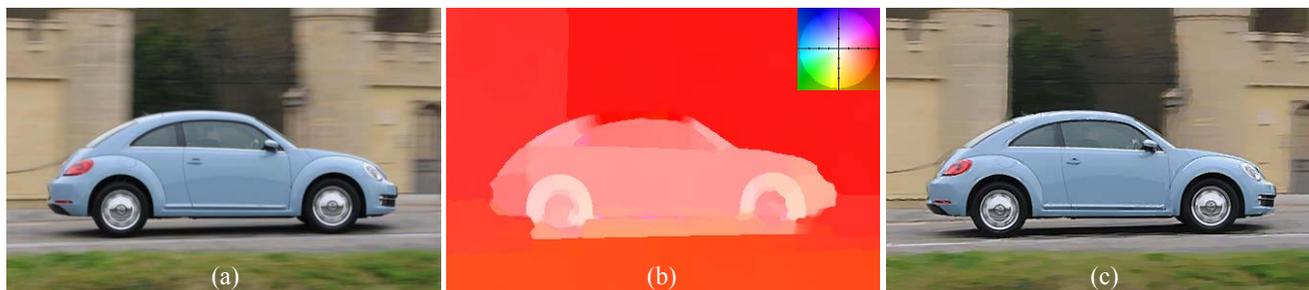


Figure 1: (a) Blurry image (b) Color coded motion flow (c) Our deblurring result

## Abstract

Most state-of-the-art dynamic scene deblurring methods based on accurate motion segmentation assume that motion blur is small or that the specific type of motion causing the blur is known. In this paper, we study a motion segmentation-free dynamic scene deblurring method, which is unlike other conventional methods. When the motion can be approximated to linear motion that is locally (pixel-wise) varying, we can handle various types of blur caused by camera shake, including out-of-plane motion, depth variation, radial distortion, and so on. Thus, we propose a new energy model simultaneously estimating motion flow and the latent image based on robust total variation (TV)-L1 model. This approach is necessary to handle abrupt changes in motion without segmentation. Furthermore, we address the problem of the traditional coarse-to-fine deblurring framework, which gives rise to artifacts when restoring small structures with distinct motion. We thus propose a novel kernel re-initialization method which reduces the error of motion flow propagated from a coarser level. Moreover, a highly effective convex optimization-based solution mitigating the computational difficulties of the TV-L1 model is established. Comparative experimental results on challenging real blurry images demonstrate the efficiency of the proposed method.

## 1. Introduction

Blind single-image deblurring is a method used to restore a sharp image from an image blurred by camera shake or object motion under low light conditions. This approach has become an active research topic in computer vision because of the recent demand for clear images. However, deblurring is difficult to solve because the problem is highly ill-posed. To solve this problem, many researchers have studied the joint estimation of latent image and blur kernel, thus recasting the deblurring problem as an energy minimization problem based on the general constraint:

$$\mathbf{B} = \mathbf{K}\mathbf{L} + \mathbf{N}, \quad (1)$$

where  $\mathbf{L}$ ,  $\mathbf{B}$ , and  $\mathbf{N}$  denote the vector forms of the latent image, blurred image, and noise, respectively. The matrix  $\mathbf{K}$  denotes the blur kernel with a row vector corresponding to the blur kernel placed at each pixel location. There are various kernel estimation methods depending on the types of motion blur. The 2D translational camera shake is the most comprehensively studied motion [3, 6, 24]. Such camera motion creates a spatially invariant blur kernel, such that the matrix operation in (1) can be expressed in a convolution form with the aid of linearity. Therefore, using fast Fourier transform makes fast kernel estimation from 2D translational camera shake possible. Moreover, camera shake, including rotational movement, generalizes the 2D translational camera motion and it enables spatially varying blur by camera rotation [8, 10, 23]. Although these efforts to model real camera motion have yielded promising results



Figure 2: (a) Partially blurred image by forward motion. (b)-(c) Segmentation and deblurring results of [13] (d) Our deblurring result. Notably, the taillight and rear fender of the taxi are restored to a significantly better degree than those in (c).

in the restoration of blurry images through the motion of a camera, applying these approaches to cases where the blur does not come from the global camera motion is difficult.

By contrast, without knowing the specific type of motion blur, traditional deterministic filter-based approaches can handle various types of blur [1, 7], but these filters are incapable of handling large motion blur and are sensitive to noise. Additionally, edge statistics can be also used to estimate locally varying blur kernels [12], but, the method is also restricted to small motion blur. Therefore, some researchers have studied methods that can be used to deal with large blur without knowing the specific type of motion blur. Harmeling et al. [9] proposed a method based on the assumption that local blur kernel is expressed as a weighted sum of the neighboring blur kernels and Ji et al. [11] interpolated initially given kernels robustly. Although these approaches can handle smoothly varying blur kernel, these methods cannot handle abrupt changes in the blur kernel, which commonly occur in a dynamic scene containing multiple moving objects.

To address this problem, some researches have focused on deblurring dynamic scenes and established approaches that commonly require accurate motion segmentation. In [16], Levin segmented blur motions by comparing likelihoods with a set of given 1D box filters. However, limited number of 1D box filters were used. Thus, the poor segmentation could cause undesirable artifacts. Couzinie-Devy et al. extended the work of Levin in [4] by casting the problem as a multi-label segmentation problem and estimating locally varying blur. However, the method could not handle large blur because of the exponential increases in the the number of candidate labels. This condition restricts the blur kernels to small 2D Gaussian or linear. The recent work of Kim et al. [13] proposed a method to estimate blur kernels, latent image, and motion segments jointly. The existence of camera shake, including rotation and 2D translational motion of objects, was assumed, but the method also fails in both segmentation and deblurring when the captured image is blurred by unexpected blur effects, such as forward motion, depth variation, or radial distortion. For example, the taxi is blurred by its forward motion, and such an unexpected motion causes a failure of both segmentation and deblurring as shown in Figs. 2(b) and 2(c).

In sum, state-of-the-art dynamic scene deblurring methods require accurate motion segmentation for specific types of motion blur. To mitigate this restriction, we propose a method to deblur dynamic scenes without segmentation and without restricting the types of motion blur, when the locally varying blur kernels can be approximated to 2D motion vectors. Although this restriction excludes non-linear motion, numerous types of motions can be linearized in practical situations [4, 5, 16]. We observed that this assumption holds for many real blurry images and that this approach works even when small rotational camera shake exists, as shown in our experiments.

In the previous work of Dai et al. [5], the authors estimated motion flow via the alpha channel [17] of the blurry image. However, they used a constraint that is different from ours and applied implicit segmentation based on RANSAC. In addition, the result depends on the accuracy of the given alpha channel. By contrast, we propose a new framework that enables abrupt changes in motion without segmentation based on the robust TV-L1 model. As illustrated in Fig.1, the proposed approach estimates the motion flow and latent image. Our method does not require any complex model to handle specific types of motion blur [10, 23, 27] and does not depend on the accurate motion segmentation. In addition, the proposed method is embedded into the traditional coarse-to-fine framework to handle large blur. Our finding is that small structures with distinct motion blur give rise to serious artifacts in the coarse-to-fine framework. Thus, we also propose a novel method to re-initialize the motion flow and reduce the error propagated in the coarse to fine framework.

In this paper, we introduce a new deblurring framework that jointly estimates the spatially varying motion flow and the latent image. We also provide a highly practical solver optimizing TV-L1 model based on convex optimization. As shown in Fig. 2(d), we achieve better results with the aid of accurately estimated motion flow even when the state-of-the-art methods fail. We demonstrate the effectiveness and practicality of our new deblurring framework with test results on challenging real images on which conventional techniques fail.

## 2. Dynamic Scene Deblurring Model

To solve the ill-posed blind deblurring problem, various energy models that are composed of the regularization and data terms have been proposed to estimate the sharp image and the blur kernel in the following form

$$E = E_{data}(\mathbf{L}, \mathbf{K}, \mathbf{B}) + E_{reg}(\mathbf{L}, \mathbf{K}), \quad (2)$$

where the data term  $E_{data}$  measures the data fidelity, and the regularization term  $E_{reg}$  enforces the smoothness constraint to the latent image as well as to the blur kernel. In [13], the authors argued that conventional energy model in (2) is invalid for dynamic scene deblurring because it principally requires motion segmentation. However, in this paper, we propose a new energy model based on (2) for dynamic scene deblurring. The proposed model does not require explicit motion segmentation. Further details on the model are described in the following sections.

### 2.1. L1-based Robust Blur Model

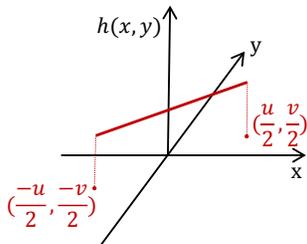


Figure 3: Blur kernel corresponding to motion flow  $\mathbf{u}$

In our study, we assumed that the locally varying blur kernel can be linearized in terms of a 2D motion vector,  $\mathbf{u} = (u, v)^T$ , which can be expressed as

$$\mathbf{h}(x, y) = \begin{cases} \frac{1}{\|\mathbf{u}\|} \delta(vx - uy), & \text{if } x \leq \frac{\|u\|}{2}, y \leq \frac{\|v\|}{2} \\ 0, & \text{otherwise.} \end{cases} \quad (3)$$

where  $\mathbf{h}(x, y)$  denotes the blur kernel corresponding to motion flow  $\mathbf{u}$  as illustrated in Fig. 3, and  $\delta$  is the Dirac delta function. Thus, we can cast the kernel estimation problem as a motion estimation problem. The data term of our new energy model is given by

$$E_{data}(\mathbf{L}, \mathbf{u}, \mathbf{B}) = \lambda \sum_{\mathbf{x}} \sum_{\partial_*} |(\mathbf{k}(\mathbf{x}, \mathbf{u}))^T \partial_* \mathbf{L} - \partial_* \mathbf{B}(\mathbf{x})|, \quad (4)$$

where  $\mathbf{x} \in \mathbb{R}^2$  is the index of the discrete locations in the image domain, and  $(\mathbf{k}(\mathbf{x}, \mathbf{u}))^T$  denotes the discretized vector form of  $\mathbf{h}$  with a sum of elements equal to one. This vector also corresponds to a row vector of the kernel matrix,  $\mathbf{K}$ , where  $\mathbf{x}$  indicates the index of the row.

The operator  $\partial_* \in \{\partial_x, \partial_y\}$  denotes the partial derivatives in the horizontal and vertical directions [3, 13]. The parameter  $\lambda$  controls the importance of the data term.

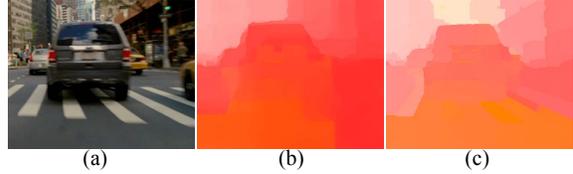


Figure 4: Comparison of motion flow with and without edge-map. (a) Blurry image (b) Color coded motion flow without edge-map (c) Color coded motion flow with edge-map

As mentioned in [13], we use only gradient maps of the latent and blurry images in the data model to reduce ringing artifacts. In addition, the proposed model must enable abrupt changes in the blur kernel near motion boundary because the goal is deblurring a dynamic scene without segmentation. Therefore, we propose an L1-based model in the data term that allows discontinuities in the flow field.

Although the robust L1 model was proposed before to estimate the latent image in a non-blind deblurring process [24], this model has not been used in the estimation of blur kernel, to the best of our knowledge, because of its computational difficulties. Despite its difficulties, we adopt the L1 model for estimating not only the latent image but also the blur kernel.

### 2.2. TV-based Robust Regularization

The smoothness of variables must be enforced to obtain a reliable solution when the problem is ill-posed. In our energy model, two primal unknown variables are the latent image  $\mathbf{L}$  and the motion flow  $\mathbf{u}$ , and each has different kinds of regularization, as given by

$$E_{reg}(\mathbf{L}, \mathbf{u}) = E_{reg}(\mathbf{L}) + E_{reg}(\mathbf{u}). \quad (5)$$

More details on the regularization of each variable are described in the following sections.

#### 2.2.1 Regularization on Latent Image

In (1), noise is added to our blurry image. Thus, we should suppress the noise in the latent image while preserving edges. To address this problem, many researchers studied the prior of latent image [14, 15, 18]. The sparse gradient prior is known to describe the edge statistics of natural images well. In the same vein, we adopt the TV model to regularize the latent image, as in [13, 24]. The formulation is given by,

$$E_{reg}(\mathbf{L}) = \sum_{\mathbf{x}} |\nabla \mathbf{L}|. \quad (6)$$

#### 2.2.2 Edge-aware Regularization on Motion Flow

To have similar blur kernels among neighboring pixels, we regularize the motion flow. Similar to the proposed data term in the preceding section, the key to regularizing the motion flow is how well the motion boundary is preserved.

To this end, we propose an edge-map coupled TV model to regularize conditionally, which yields

$$E_{reg}(\mathbf{u}) = \mu \sum_{\mathbf{x}} g(\mathbf{x}) |\nabla \mathbf{u}|, \quad (7)$$

where the parameter  $\mu$  controls the strength of regularization, and the edge-map,  $g(\mathbf{x})$ , measuring the similarity between neighboring pixels is defined as

$$g(\mathbf{x}) = \exp\left(-\left(\frac{\max(|\nabla \mathbf{B}|, |\nabla \mathbf{L}_0|)}{\sigma_I}\right)^2\right). \quad (8)$$

The parameter  $\sigma_I$  denotes the decay exponent and  $\mathbf{L}_0$  is an intermediate latent image propagated from a coarser level in the coarse-to-fine framework. In particular, the blurred image itself is also used to obtain the edge-map, because an unblurred region may exist in a partially blurred image, and it works when  $\mathbf{L}_0$  is smoothed out.

The outstanding performance of edge-aware regularization is demonstrated in Fig. 4. As expected, the proposed regularizer preserves motion discontinuities much better than the model without edge-map.

### 3. Optimization Framework

The proposed single-image blind deblurring model introduced in the previous sections includes the data and regularization terms, and the objective function is given by

$$\min_{\mathbf{L}, \mathbf{u}} \sum_{\mathbf{x}} |\nabla \mathbf{L}| + \mu \sum_{\mathbf{x}} g(\mathbf{x}) |\nabla \mathbf{u}| + \lambda \sum_{\partial_*} |(\mathbf{k}(\mathbf{x}, \mathbf{u}))^T \partial_* \mathbf{L} - \partial_* \mathbf{B}(\mathbf{x})|. \quad (9)$$

Our final objective in (9) is highly complex, that is, to obtain a global optimum. Thus we divide the original problem into two easier subproblems for  $\mathbf{L}$  and  $\mathbf{u}$ . We then apply alternating optimization procedure, which was introduced in [3, 21, 26]. The overall algorithm of the proposed method is given in Algorithm 1.

Moreover, our approach is embedded into the traditional coarse-to-fine framework [6, 19] to handle large motion blur. However, we found that it gives rise to severe artifacts when applied to a dynamic scene, which has small structures with distinct motion blurs. Thus, we propose a novel re-initialization method to solve this problem. Further detail is given in the following section.

#### 3.1. Kernel Re-initialization

The conventional coarse-to-fine approach is widely used in various fields and also has been shown to yield successful results in the deblurring of static scene. This method can be used in dynamic scene deblurring, but it has limitations that have not been observed in the restoration of a static scene.

For example, a small structure, such as a thin line, that has distinctive motion blur, cannot be seen at the coarser level. Therefore, the motion flow of such structure is estimated from its neighbors, which exhibit different motion,

---

#### Algorithm 1 Overview of the proposed method

---

**Input:** A blurry image  $\mathbf{B}$

**Output:** Latent image  $\mathbf{L}$  and motion flow  $\mathbf{u}$

- 1: Build an image pyramid with 10 levels and a scale factor of 0.8
  - 2: Kernel re-initialization. (Sec. 3.1)
  - 3: **for**  $t = 1$  to 3 **do**
  - 4:   Continuous optimization of motion flow  $\mathbf{u}$  with fixed  $\mathbf{L}$ . (Sec. 3.2)
  - 5:   Continuous optimization of latent image  $\mathbf{L}$  with fixed  $\mathbf{u}$ . (Sec. 3.3)
  - 6: **end for**
  - 7: Propagate variables to the next pyramid level if exists.
  - 8: Repeat steps 2-7 from coarse to fine pyramid level.
- 

with the aid of regularization. However, this process generates reconstruction error toward the finer level when the small structure appears suddenly in the blurry image. Notably, a similar problem has been reported in other vision applications [22, 25].

Local and sparse reconstruction error may not raise severe artifacts in the estimation of global camera motion, which is more serious in a local approach, such as the method proposed in this study.

To mitigate this problem, we propose a novel kernel re-initialization method for both motion flow and the latent image. The key idea is to detect the erroneous region and deblur it using a deterministic filter [20], after which accurate motion flow is estimated. Hence, we cast the problem of detecting the erroneous region as a labeling problem, that is,

$$\min_{\mathbf{e}} \sum_{\mathbf{x}} \mathbf{e}(\mathbf{x}) (c - |(\mathbf{k}(\mathbf{x}, \mathbf{u}))^T \mathbf{L} - \mathbf{B}(\mathbf{x})|) + \frac{1}{\nu} |\mathbf{e}(\mathbf{x})|, \quad (10)$$

where the vector variable  $\mathbf{e}$  is pixel-wise binary indicator variable,  $\mathbf{e}(\mathbf{x}) \in \{0, 1\}$ , and the constant  $c$  is a positive threshold value.

The first term introduced in (10) is the likelihood term that causes  $\mathbf{e}(\mathbf{x})$  to be one when the reconstruction error,  $|(\mathbf{k}(\mathbf{x}, \mathbf{u}))^T \mathbf{L} - \mathbf{B}(\mathbf{x})|$ , is higher than  $c$ . The second term is a prior giving the sparsity on the variable  $\mathbf{e}$ , as we assume high reconstruction error is sparsely distributed, and the parameter  $\nu$  controls the importance of the prior.

Through the continuous relaxation of  $\mathbf{e}$ , we can obtain the approximated solution of (10) quickly by adapting the first-order primal dual algorithm [2], that is,

$$\begin{cases} \mathbf{r}(\mathbf{x})^{n+1} = \frac{\mathbf{r}(\mathbf{x})^n + \sigma \mathbf{e}(\mathbf{x})^n}{\max(1, \mathbf{r}(\mathbf{x})^n + \sigma \mathbf{e}(\mathbf{x})^n)} \\ \mathbf{e}(\mathbf{x})^{n+1} = \arg \min_{\mathbf{e}} \frac{(\mathbf{e} - (\mathbf{e}(\mathbf{x})^n - \tau \mathbf{r}(\mathbf{x})^{n+1}))^2}{2\tau} + \\ \quad \nu \cdot \mathbf{e}(\mathbf{x}) (c - |(\mathbf{k}(\mathbf{x}, \mathbf{u}))^T \mathbf{L} - \mathbf{B}(\mathbf{x})|) \\ \mathbf{e}(\mathbf{x})^{n+1} = \max(0, \min(1, \mathbf{e}(\mathbf{x})^{n+1})), \end{cases} \quad (11)$$

where  $n \geq 0$  indicates the iteration number, vector  $\mathbf{r}$  denotes the dual variable of  $\mathbf{e}$  and the positive update steps  $\sigma$  and  $\tau$  control the convergence rate [2].

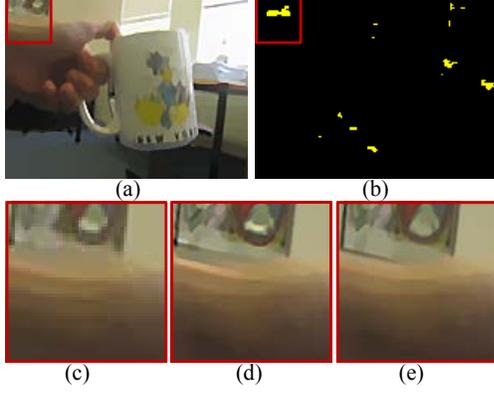


Figure 5: (a) Estimated latent image in a coarse level (b) Yellow color denotes the detected erroneous region. (c) Cropped result in the coarse level (d) Cropped result in the finest level without the use of re-initialization (e) Cropped result in the finest level with the use of re-initialization

After detection of the erroneous region, we re-initialize the propagated motion of this region to be zero, which denotes the impulse blur kernel. By applying sharp image restoration in section 3.3, we restore the small structure with distinct blur motion. However, the newly restored small structure remains blurry because it is estimated from the impulse blur kernel. Thus, we apply a deterministic filter [20] to deblur the structure and to facilitate the fast convergence of motion estimation. For this process, we use the prediction step introduced in [3]. The overall process is shown in Algorithm 2.

The necessity of our re-initialization step is illustrated in Fig. 5. Notably an unseen thin line in the coarse level in Fig. 5(c) is successfully restored using our proposed method, as compared in Figs. 5(d) to (e).

---

#### Algorithm 2 Kernel re-initialization algorithm

---

**Input:** A blurry image  $\mathbf{B}$ , intermediate latent image  $\mathbf{L}$  and motion flow  $\mathbf{u}$  propagated from coarser pyramid level.

**Output:** Re-initialized motion flow  $\mathbf{u}$  and edge-enhanced  $\mathbf{L}$

- 1: **for**  $n = 1$  to 50 **do**
  - 2:   Compute  $\mathbf{e}$  using (11)
  - 3: **end for**
  - 4:  $\mathbf{u}(\mathbf{x}) \leftarrow 0$ , if  $\mathbf{e}(\mathbf{x}) > 0.5$
  - 5: Continuous optimization of  $\mathbf{L}$  with re-initialized  $\mathbf{u}$  (Sec. 3.3)
  - 6: Enhance edge by applying prediction step in [3]
- 

### 3.2. Motion Flow Estimation

For the latent image  $\mathbf{L}$  being fixed, the proposed energy model in (9) is simplified, but the data term  $\rho(\mathbf{x}, \mathbf{u})$  is non-convex in the argument  $\mathbf{u}$ , which makes the optimization intractable

$$\rho(\mathbf{x}, \mathbf{u}) = \sum_{\partial_*} |(\mathbf{k}(\mathbf{x}, \mathbf{u}))^T \partial_* \mathbf{L} - \partial_* \mathbf{B}(\mathbf{x})|. \quad (12)$$

To make the optimization tractable, we linearize the data function via the Taylor approximation to obtain

$$\rho(\mathbf{x}, \mathbf{u}) \approx \rho(\mathbf{x}, \mathbf{u}_0) + \nabla \rho(\mathbf{x}, \mathbf{u}_0)^T (\mathbf{u} - \mathbf{u}_0), \quad (13)$$

where  $\mathbf{u}_0$  is an initial motion flow of  $\mathbf{u}$ , and  $\nabla \rho(\mathbf{x}, \mathbf{u}_0)$  denotes the first-order derivative. Through approximation, the proposed energy model becomes convex near  $\mathbf{u}_0$ , which results in

$$\mathbf{u} = \arg \min_{\mathbf{u}} \sum_{\mathbf{x}} \mu \cdot g(\mathbf{x}) |\nabla \mathbf{u}| + \lambda (\rho(\mathbf{x}, \mathbf{u}_0) + \nabla \rho(\mathbf{x}, \mathbf{u}_0)^T (\mathbf{u} - \mathbf{u}_0)), \quad (14)$$

To solve this, we adapted the convex optimization algorithm in [2], and the primal dual update process is given by

$$\begin{cases} \mathbf{p}^{n+1} = \frac{\mathbf{p}^n + \sigma(\mathbf{GA})\mathbf{u}^n}{\max(1, \mathbf{p}^n + \sigma(\mathbf{GA})\mathbf{u}^n)} \\ \mathbf{u}^{n+1} = (\mathbf{u}^n - \tau(\mathbf{GA})^T \mathbf{p}^{n+1}) - \tau \left( \frac{\lambda}{\mu} \right) \nabla \rho(\mathbf{x}, \mathbf{u}_0), \end{cases} \quad (15)$$

where  $\mathbf{p}$  denotes the dual variable of  $\mathbf{u}$  on the vector space. The continuous linear operator  $\mathbf{A}$  calculates the difference between neighboring pixels, and the diagonal matrix  $\mathbf{G}$  is the weighting matrix denoted as  $\mathbf{G} = \text{diag}(g(\mathbf{x}))$ . This update process is easy to implement and converges quickly ( $n=100$ ).

### 3.3. Sharp Image Estimation

For the motion flow  $\mathbf{u}$  and corresponding kernel matrix  $\mathbf{K}$  being fixed, the proposed energy model becomes a well-known non-blind deblurring model. However, as we use the L1 model in the proposed data term, which requires high computations. To address this problem, we adopt the quadratic relaxation method [22] to facilitate the solution. Thus, we introduce auxiliary variables  $\mathbf{f}_*$ , which yields

$$\min_{\mathbf{L}, \mathbf{f}} \sum_{\mathbf{x}} |\nabla \mathbf{L}| + \lambda \sum_{\partial_*} |\mathbf{f}_*(\mathbf{x})| + \frac{1}{2\theta} \sum_{\partial_*} ((\mathbf{k}(\mathbf{x}, \mathbf{u}))^T \partial_* \mathbf{L} - \partial_* \mathbf{B}(\mathbf{x}) + \mathbf{f}_*(\mathbf{x}))^2. \quad (16)$$

If the fixed parameter  $\theta$  is set to a very small value, then the minimization of (16) is close to that of its original TV-L1 model. Using decomposition, both the function of  $\mathbf{L}$  and functions of  $\mathbf{f}_*$  become a thousand times easier to solve, and each variable is solved by [2] in the alternating optimization process. The update process for  $\mathbf{f}_*$  while  $\mathbf{L}$  being fixed is expressed as

$$\begin{cases} \mathbf{q}_*(\mathbf{x})^{n+1} = \frac{\mathbf{q}_*(\mathbf{x})^n + \sigma \mathbf{f}_*(\mathbf{x})^n}{\max(1, \mathbf{q}_*(\mathbf{x})^n + \sigma \mathbf{f}_*(\mathbf{x})^n)} \\ \mathbf{f}_*(\mathbf{x})^{n+1} = \arg \min_{\mathbf{f}} \frac{(\mathbf{f} - (\mathbf{f}_*(\mathbf{x})^n - \tau \mathbf{q}_*(\mathbf{x})^{n+1}))^2}{2\tau} + \\ \frac{1}{2\theta\lambda} \sum_{\partial_*} ((\mathbf{k}(\mathbf{x}, \mathbf{u}))^T \partial_* \mathbf{L} - \partial_* \mathbf{B}(\mathbf{x}) + \mathbf{f}_*(\mathbf{x}))^2, \end{cases} \quad (17)$$

where the vector  $\mathbf{q}_*$  is the dual variable of  $\mathbf{f}_*$ .

Similarly, the update process for  $\mathbf{L}$  while  $\mathbf{f}_*$  being fixed

is given by

$$\begin{cases} \mathbf{s}^{m+1} = \frac{\mathbf{s}^m + \sigma_L \mathbf{A} \mathbf{L}^m}{\max(\mathbf{1}, \mathbf{s}^m + \sigma_L \mathbf{A} \mathbf{L}^m)} \\ \mathbf{L}^{m+1} = \arg \min_{\mathbf{L}} \frac{(\mathbf{L} - (\mathbf{L}^m - \tau_L \mathbf{A}^T \mathbf{s}^{m+1}))^2}{2\tau_L} + \\ \frac{1}{2\theta\lambda} \sum_{\partial_*} (\mathbf{K} \partial_* \mathbf{L} - \partial_* \mathbf{B} + \mathbf{f}_*)^2, \end{cases} \quad (18)$$

where  $m \geq 0$  indicates iteration number, the parameters  $\sigma_L$  and  $\tau_L$  denote the update steps and the vector  $\mathbf{s}$  is the dual variable of  $\mathbf{L}$ . In addition, we adopt conjugate gradient (CG) method to update primal variable,  $\mathbf{L}$ , in (18).

The overall procedure of our sharp image estimation is shown in Algorithm 3.

---

### Algorithm 3 Sharp Image Estimation

---

**Input:** A blurry image  $\mathbf{B}$  and motion flow  $\mathbf{u}$

**Output:** Sharp image  $\mathbf{L}$

- 1: **for**  $n = 1$  to 3 **do**
  - 2:   Update  $\mathbf{f}_*$  using (17)
  - 3:   **for**  $m = 1$  to 30 **do**
  - 4:     Update  $\mathbf{L}$  using (18)
  - 5:   **end for**
  - 6: **end for**
- 

## 3.4. Implementation

In implementation, we use fixed parameters for most experiments except for parameter  $\lambda$ . Because each blurry image has a different amount of noise and a different blur kernel size, the parameter  $\lambda$  adjusting the smoothness of motion flow and latent image can be changed. We set the parameters as  $\mu = 1$ ,  $\sigma_I = \frac{15}{255}$ ,  $c = 0.05$ ,  $\nu = 0.02$ ,  $\sigma = \frac{1}{2\sqrt{(2)}}$ ,  $\tau = \frac{1}{2\sqrt{(2)}}$ ,  $\sigma_L = 10$ ,  $\tau_L = 0.0125$ , and  $\theta = \frac{0.1}{\lambda}$ . The value of  $\lambda$  ranges from 10 to 50.

In the coarse-to-fine framework, we build an image pyramids with 10 levels and a scale factor of 0.8, and use bi-cubic interpolation for both motion flow and the latent image in propagation. Initially, all dual variables are set as zeros. In addition, the blurry image itself is used for initial latent image  $\mathbf{L}$ , and we use small random values for initial motion flow  $\mathbf{u}$ . However, as our blur kernel is symmetric (i.e.  $\mathbf{k}(\mathbf{x}, \mathbf{u}) = \mathbf{k}(\mathbf{x}, -\mathbf{u})$ ), we constrain the motion flow,  $(u, v)$ , to be on a set,  $\mathbb{B} = \{(u, v) \in \mathbb{R}^2 | u \geq 0\} - \{(u, v) \in \mathbb{R}^2 | u = 0, v < 0\}$ .

Our current Matlab implementation requires approximately 25 minutes to deblur a VGA image. The running time can be greatly reduced using optimized C or CUDA on a modern GPGPU.

## 4. Experimental Results

In this section, we show the outstanding performance and efficiency of the proposed work.

In Fig. 6, deblurring results and estimated motion flow are illustrated for real challenging dynamic scenes. In qual-

itative analysis, the edges in deblurred images are restored keenly without segmentation and without restricting the type of motion blur. The color codes of estimated motion flows in textured regions are as accurate as we expected except homogeneous regions that do not raise severe artifacts. Notably, the motion flow in the the bottom row is corresponding to the depth map of the latent image, and this could be another application of our work that is estimating depth map from a blurry single-image.

In Fig. 7, deblurring results for camera shakes which includes rotational camera movements are illustrated. Because the proposed method is based on the approximation of locally linear blur kernel, the results may not accurate than the results from methods estimating the global camera motion in less-textured regions. However, the restored strong edges from both synthetic image in the top row and the real image in the bottom row demonstrate that our approximation is valid for small rotational camera shake to some degree.

In addition, we compared our results with the state-of-the-art deblurring methods in various challenging situations. First, in Fig. 8(a), the blurry image is degraded by severe radial distortion and it causes serious problems in conventional methods but ours works well. Second, in Fig. 8(b), because the cause of blur is specific forward motion which can not be handled in conventional methods, ours outperforms and restores the edges of characters and arrow more sharply. Third, in Fig. 8(c), [26] fails in deblurring as depth discontinuity exist in the blurry image. Although it is possible to deblur with segmentation based method, but [13] also fails, because the background is too narrow to be segmented. Thus, both conventional methods provide unsatisfactory results. Finally, in Fig. 8(d), as the synthetic image, which has depth discontinuity, is blurred by rotational camera shake, and thus, both the conventional methods fail but our method shows successful result.

## 5. Discussion and Limitations

In this study, we have presented an efficient dynamic scene deblurring method that does not require accurate motion segmentation with the aid of robust TV-L1 based model. In addition, the proposed method can handle various types of blur kernels when the kernels can be approximated to linear motions. We also provided an efficient and practical solution optimizing TV-L1 model, and demonstrated that the proposed method outperforms the state-of-the-art methods in various challenging situations.

On the other hand, the remaining problem is that, in case of knowing the specific types of motion blurs, conventional deblurring methods estimating the global camera motion work well and the results in less-textured region are better than ours. Therefore our future work will combine such a global and our local approach in a unified framework.

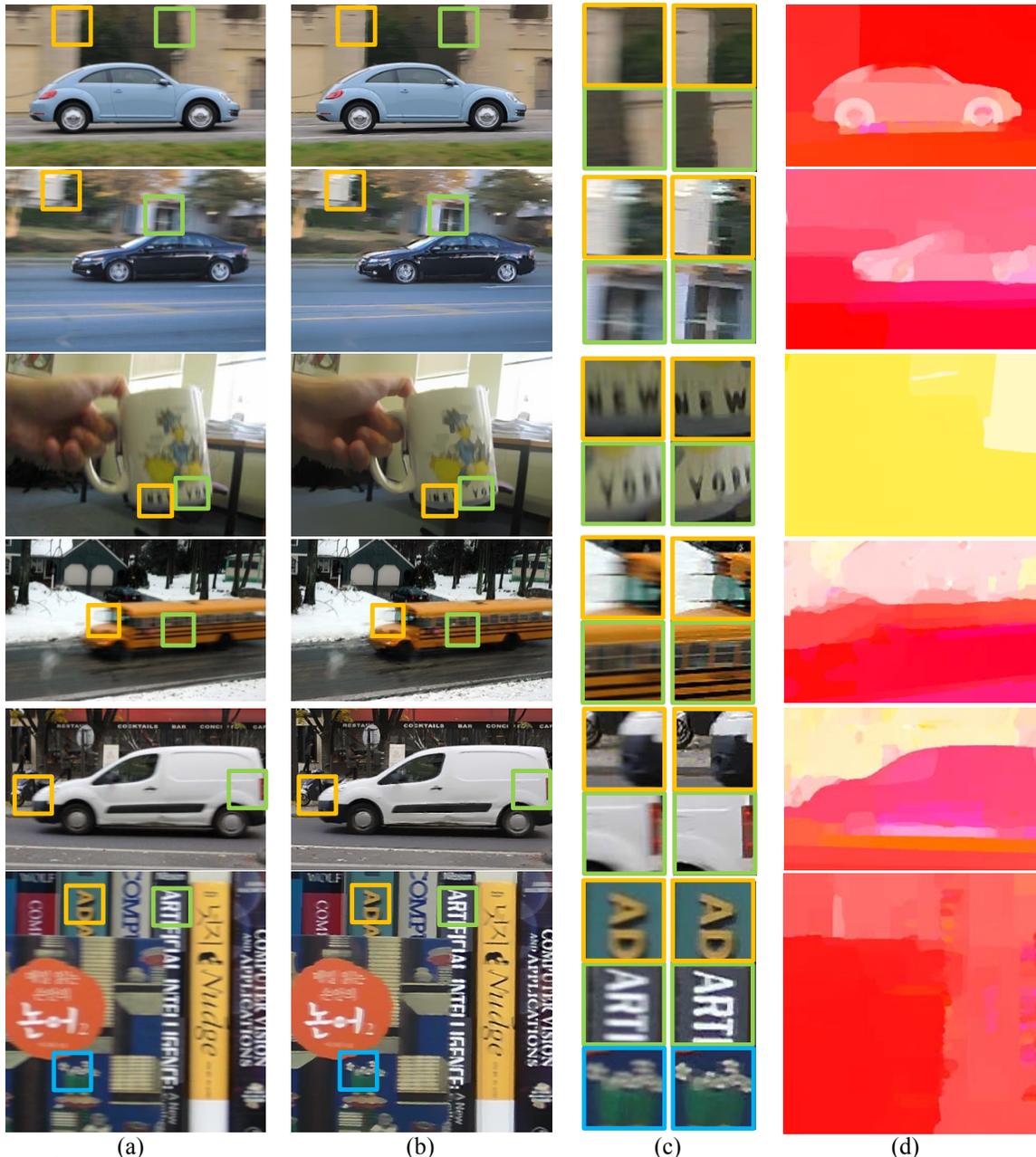


Figure 6: (a) Real blurry dynamic scenes (b) Deblurring results (c) Comparisons of cropped results (d) Estimated motion flows

## Acknowledgments

This research was supported in part by the MKE(The Ministry of Knowledge Economy), Korea and Microsoft Research, under IT/SW Creative research program supervised by the NIPA(National IT Industry Promotion Agency) (NIPA-2013-H0503-13-1041).

## References

- [1] P. Bhat, C. L. Zitnick, M. Cohen, and B. Curless. Gradientshop: A gradient-domain optimization framework for image and video filtering. *ACM Transactions on Graphics (TOG)*, 29(10), 2010. 2
- [2] A. Chambolle and T. Pock. A first-order primal-dual algorithm for convex problems with applications to imaging. *Journal of Mathematical Imaging and Vision*, 40(1):120–145, May 2011. 4, 5
- [3] S. Cho and S. Lee. Fast motion deblurring. In *SIGGRAPH*, 2009. 1, 3, 4, 5
- [4] F. Couzinie-Devy, J. Sun, K. Alahari, and J. Ponce. Learning to estimate and remove non-uniform image blur. In *CVPR*, 2013. 2
- [5] S. Dai and Y. Wu. Motion from blur. In *CVPR*, 2008. 2
- [6] R. Fergus, B. Singh, A. Hertzmann, S. T. Roweis, and W. Freeman. Removing camera shake from a single photograph. In *SIGGRAPH*, 2006. 1, 4
- [7] G. Gilboa, N. Sochen, and Y. Y. Zeevi. Image enhancement and denoising by complex diffusion processes. *IEEE Trans. Pattern Analysis Machine Intelligence*, 26(8):1020–1036, 2004. 2
- [8] A. Gupta, N. Joshi, L. Zitnick, M. Cohen, and B. Curless. Single image deblurring using motion density functions. In *ECCV*, 2010. 1

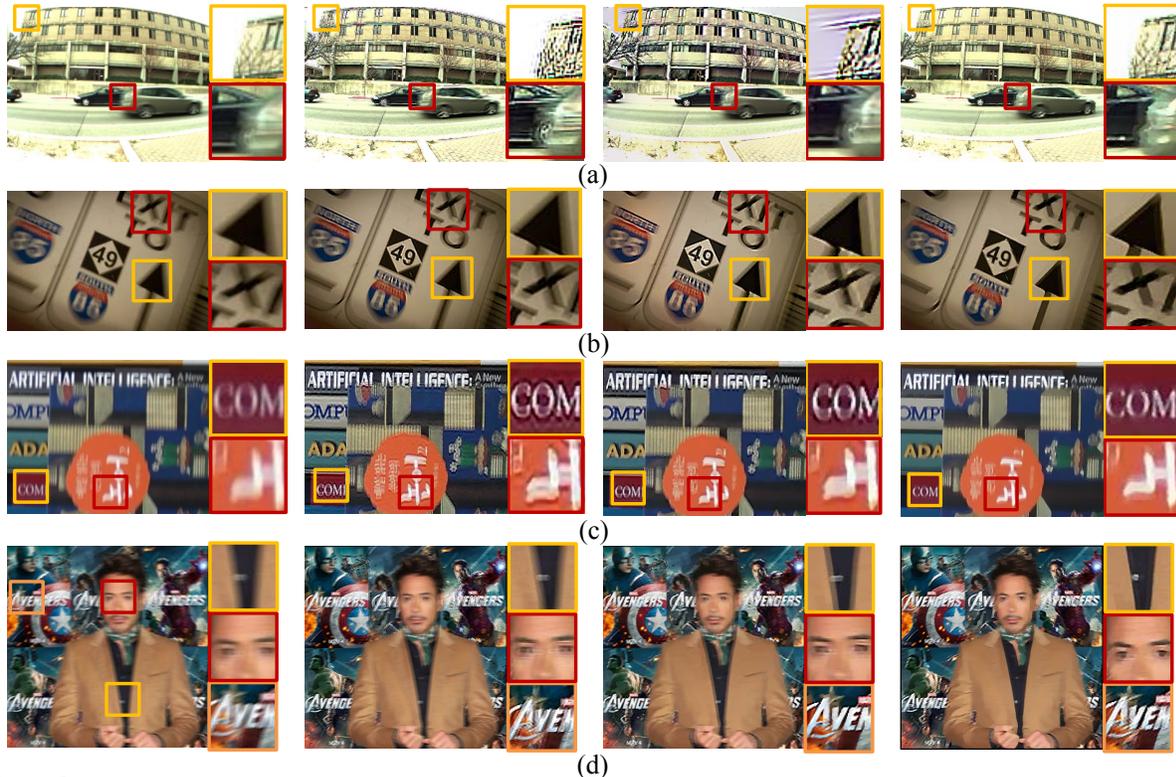


Figure 8: From left to right: Blurry images, results of Xu et al [26], results of Kim et al [13], and results of the proposed method

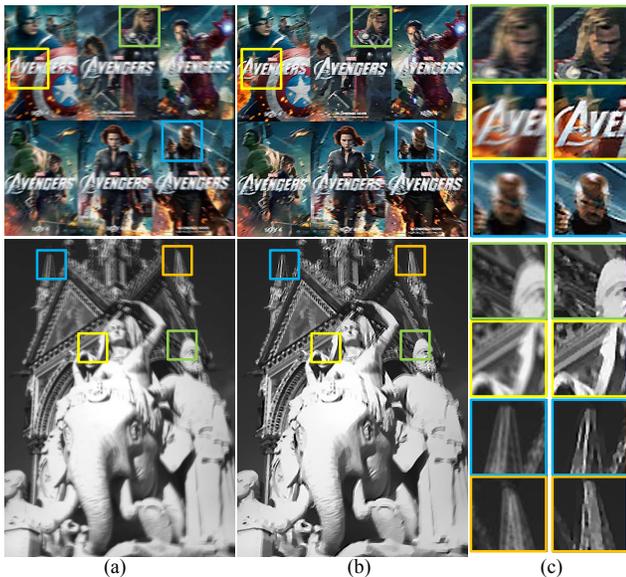


Figure 7: (a) Blurry images from rotational camera shakes (b) Deblurring results (c) Comparisons of cropped results

[9] S. Harmeling, H. Michael, and B. Schoelkopf. Space-variant single-image blind deconvolution for removing camera shake. In *NIPS*, 2010. 2

[10] M. Hirsch, C. J. Schuler, S. Harmeling, and B. Scholkopf. Fast removal of non-uniform camera shake. In *ICCV*, 2011. 1, 2

[11] H. Ji and K. Wang. A two-stage approach to blind spatially-varying motion deblurring. In *CVPR*, 2012. 2

[12] N. Joshi, R. Szeliski, and D. Kriegman. Psf estimation using sharp edge prediction. In *CVPR*, 2008. 2

[13] T. H. Kim, B. Ahn, and K. M. Lee. Dynamic scene deblurring. In *ICCV*, 2013. 2, 3, 6, 8

[14] D. Krishnan and R. Fergus. Fast image deconvolution using hyper-laplacian priors. In *NIPS*, 2009. 3

[15] D. Krishnan, T. Tay, and R. Fergus. Blind deconvolution using a normalized sparsity measure. In *CVPR*, 2009. 3

[16] A. Levin. Blind motion deblurring using image statistics. In *NIPS*, 2006. 2

[17] A. Levin, D. Lischinski, and Y. Weiss. A closed form solution to natural image matting. *IEEE Trans. Pattern Analysis Machine Intelligence*, 30(2):228–242, 2008. 2

[18] A. Levin and Y. Weiss. User assisted separation of reflections from a single image using a sparsity prior. *IEEE Trans. Pattern Analysis Machine Intelligence*, 29(9):1647–1654, 2007. 3

[19] A. Levin, Y. Weiss, F. Durand, and W. T. Freeman. Understanding and evaluating blind deconvolution algorithms. In *CVPR*, 2009. 4

[20] S. Osher and L. I. Rudin. Feature-oriented image enhancement using shock filters. *SIAM Journal on Numerical Analysis*, 27(4):919–940, 1990. 4, 5

[21] Q. Shan, J. Jia, and A. Agarwala. High-quality motion deblurring from a single image. In *SIGGRAPH*, 2008. 4

[22] F. Steinbruecker, T. Pock, and D. Cremers. Large displacement optical flow computation without warping. In *ICCV*, 2009. 4, 5

[23] O. Whyte, J. Sivic, A. Zisserman, and J. Ponce. Non-uniform deblurring for shaken images. *International Journal of Computer Vision*, 98(2):168–186, 2012. 1, 2

[24] L. Xu and J. Jia. Two-phase kernel estimation for robust motion deblurring. In *ECCV*, 2010. 1, 3

[25] L. Xu, J. Jia, and Y. Matsushita. Motion detail preserving optical flow estimation. *IEEE Trans. Pattern Analysis Machine Intelligence*, 34(9):1744–1757, 2012. 4

[26] L. Xu, S. Zheng, and J. Jia. Unnatural 10 sparse representation for natural image deblurring. In *CVPR*, 2013. 4, 6, 8

[27] S. Zheng, L. Xu, , and J. Jia. Forward motion deblurring. In *ICCV*, 2013. 2