

## Generalized Video Deblurring for Dynamic Scenes

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>

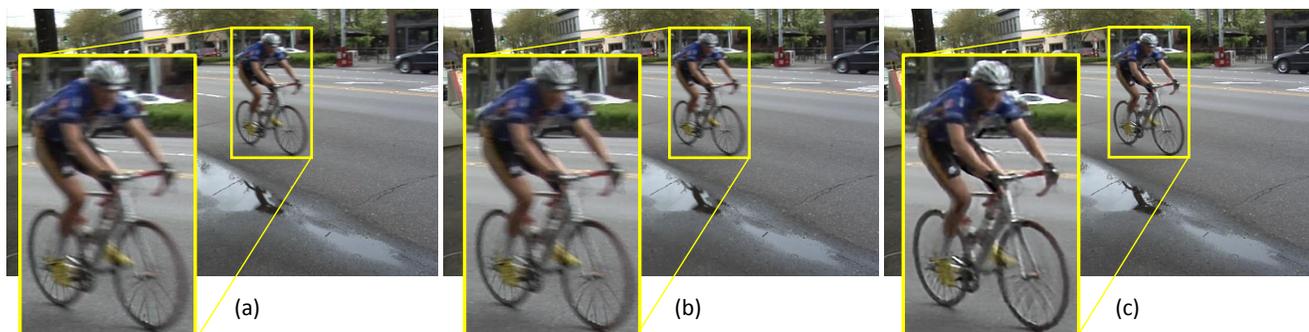


Figure 1: (a) Blurry frame of the bicycle sequence. (b) Deblurring result of Cho et al.[7]. (c) Our result.

### Abstract

Several state-of-the-art video deblurring methods are based on a strong assumption that the captured scenes are static. These methods fail to deblur blurry videos in dynamic scenes. We propose a video deblurring method to deal with general blurs inherent in dynamic scenes, contrary to other methods. To handle locally varying and general blurs caused by various sources, such as camera shake, moving objects, and depth variation in a scene, we approximate pixel-wise kernel with bidirectional optical flows. Therefore, we propose a single energy model that simultaneously estimates optical flows and latent frames to solve our deblurring problem. We also provide a framework and efficient solvers to optimize the energy model. By minimizing the proposed energy function, we achieve significant improvements in removing blurs and estimating accurate optical flows in blurry frames. Extensive experimental results demonstrate the superiority of the proposed method in real and challenging videos that state-of-the-art methods fail in either deblurring or optical flow estimation.

### 1. Introduction

Motion blurs are the most common artifacts in videos recorded using hand-held cameras. For decades, several researchers have studied deblurring algorithms to remove motion blurs. Their methodologies depend on whether the

captured scenes are static or non-static. Early works on single image deblurring usually assumed that the scene is static with constant depth [5, 9, 10, 11, 25, 27]. The successful approaches were naturally extended to video deblurring. In the work of Cai et al. [2], a multi-image deconvolution method was proposed using sparsity of blur kernels and clear image to handle registration errors. However, this method only enables two-dimensional translational camera motion, which generates uniform blur. Therefore, the proposed approach cannot handle rotational camera shake, which is the main cause of large motion blurs [27]. To overcome this limitation, Li et al. [21] used a method parameterizing spatially varying motions with 3x3 homographies, and could handle spatially varying blurs by camera rotation. In the work of Cho et al. [4], camera motion in three-dimensional space was estimated without the assistance of specialized hardware. In addition, non-uniform blurs by projective camera motion could be removed. Spatially varying blurs by depth variation in a static scene was handled recently in the works of Lee and Lee [19] and Paramanand et al. [23].

However, previous approaches, which assume that the scene is static, suffer from general blurs not only from camera shake but also from moving objects and depth variations in a dynamic scene. As parameterizing a spatially varying blur kernel in the dynamic scene is difficult with simple homography, kernel estimation to handle dynamic scene becomes more challenging. Therefore, several re-

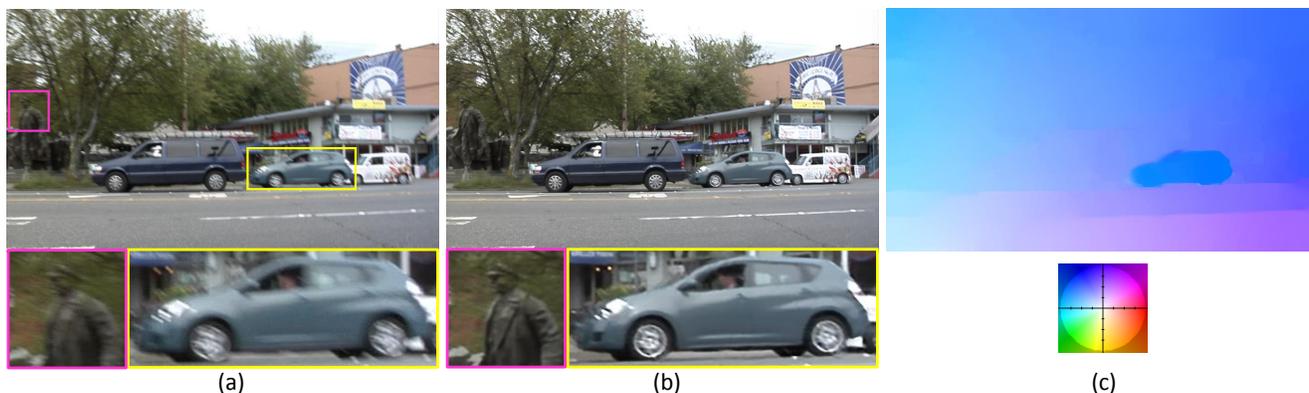


Figure 2: (a) Blurry frame of video containing moving car. (b) Our deblurring result. (c) Our color coded optical flow.

searchers have focused on restoring dynamic scenes, which is mainly grouped into two approaches: segmentation-based approach, and exemplar-based approach.

Segmentation-based deblurring approaches simultaneously estimate multiple motions, multiple kernels, and associated image segments. Cho et al. [6] proposed a method that segments images into multiple regions of homogeneous motions and estimates the corresponding blur kernel as a one-dimensional Gaussian kernel. Therefore, this method cannot handle complex motions of objects and rotational motions of cameras that generate locally varying blurs. Bar et al. [1] proposed a layered model and segmented images into two layers (foreground and background). In addition, they estimated a linear blur kernel corresponding to a foreground layer. Although this method can explicitly handle occluded regions using a layered model, the kernel is limited to a one-dimensional box filter only, and only a static camera is allowed. Wulff and Black [28] extended the previous work of Bar et al. They focused on estimating the parameters for both foreground and background motions. However, the motions within each segment are only parameterized using the affine model, and extending to multi-layered scenes is difficult because such task requires joint estimation of depth ordering of the layers. In summary, segmentation-based approaches have the advantage of handling blurs by moving objects in dynamic scenes. However, parameterizing the motions in each segment remains an issue [16]. That is, it fails to segment non-parametrically varying complex motions such as motions of people, because doing so with the simple models used in [1, 28] is difficult.

The works of Matsushita et al. [22] and Cho et al. [7] are typical exemplar-based approaches. These works estimate latent frames by interpolating sharp patches, that commonly exist in a long image sequence. Therefore, these methods disregard accurate segmentation and deconvolution, enabling the emergence of ringing artifacts. However,

the former work cannot handle blurs by moving objects. Moreover, the latter one can only treat blurs by slightly moving objects in dynamic scenes because it searches sharp patches of a blurry patch using globally parameterized kernel with homography. Therefore, handling fast-moving objects, which have distinct motions from backgrounds, is difficult. Moreover, it degrades mid-frequency textures, such as grasses and trees, because this method does not use deconvolution with spatial priors but uses interpolation to restore latent frames, which renders smooth results.

To alleviate the problems in previous works, we propose a new generalized video deblurring method that estimates latent frames without using global motion parametrization and segmentation. We estimate bidirectional optical flows and use them to estimate pixel-wise varying kernels. Therefore, we can naturally handle coexisting blurs by camera shake, moving objects with complex motions, and depth variations. However, sharp frames are required to obtain accurate optical flows because estimating flow fields is difficult between blurry images. In addition, accurate optical flows are necessary to restore sharp frames. This case is a typical chicken-and-egg problem, and thus we simultaneously estimate both variables. Therefore, we propose a new single energy model to solve our joint problem. We also provide a framework and efficient techniques to optimize the model. The result of our system is shown in Fig.2, in which the moving car is successfully restored because accurate optical flows are jointly estimated.

By minimizing the proposed energy function, we achieve significant improvements in numerous real challenging videos that other methods fail to do, as shown in Fig.1. Furthermore, we estimate more accurate optical flows compared with the state-of-the-art flow estimation method, that handles blurry images. The performances are demonstrated in our extensive experiments.

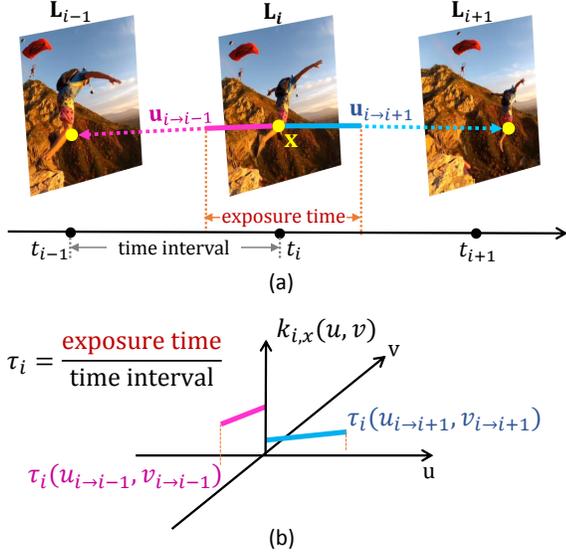


Figure 3: (a) Bidirectional optical flows. (b) Piece-wise linear blur kernel at pixel location  $\mathbf{x}$ .

## 2. Generalized Video Deblurring

Most conventional video deblurring methods suffer from the coexistence of various motion blurs from dynamic scenes because the motions cannot be parameterized using global or segment-wise parameterization. To handle general blurs, we propose a new energy model using pixel-wise kernel estimation rather than global or segment-wise parameterization. As blind deblurring is a well-known ill-posed problem, our energy model not only consists of data and spatial regularization terms but also a temporal term. The model is expressed as follows:

$$\mathbf{E} = \mathbf{E}_{data} + \mathbf{E}_{temporal} + \mathbf{E}_{spatial}, \quad (1)$$

and the details of each term in (1) are given in the following sections.

### 2.1. Data Model based on Approximated Blur

In conventional works, the motion blurs of each frame are approximated using parametric models such as homographies and affine models [1, 7, 21, 28]. However, these kernel approximations are valid when motion blurs are parameterizable within an entire frame or segment. Therefore, pixel-wise motion and kernel estimation are required to cope with general blurs. We approximate the pixel-wise blur kernel using bidirectional optical flows, in accordance with previous works [8, 16, 24].

Specifically, under an assumption that the velocity of the motion is constant between adjacent frames, our blur model is expressed as follows:

$$\mathbf{B}_i = \frac{1}{2\tau_i} \int_0^{\tau_i} H(\mathbf{L}_i, t \cdot \mathbf{u}_{i \rightarrow i+1}) + H(\mathbf{L}_i, t \cdot \mathbf{u}_{i \rightarrow i-1}) dt, \quad (2)$$

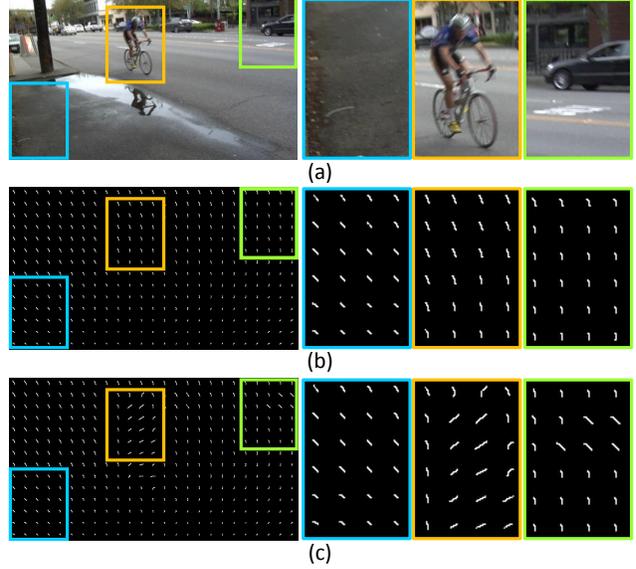


Figure 4: (a) Blurry frame of a video in dynamic scene. (b) Locally varying kernel using homography. (c) Our pixel-wise varying kernel using bidirectional optical flows.

where  $\mathbf{u}_{i \rightarrow i+1} = (u_{i \rightarrow i+1}, v_{i \rightarrow i+1})$ , and  $\mathbf{u}_{i \rightarrow i-1} = (u_{i \rightarrow i-1}, v_{i \rightarrow i-1})$  denote bidirectional optical flows at frame  $i$ . Blurry frame and latent frame are  $\mathbf{B}_i$  and  $\mathbf{L}_i$ , respectively. Camera duty cycle of the frame is  $\tau_i$  and denotes relative exposure time [21]. We define the image warping,  $H(\mathbf{L}_i, t \cdot \mathbf{u}_{i \rightarrow i+1})$ , which transforms the frame  $\mathbf{L}_i$  to  $\mathbf{L}_{i+t}$  when  $0 \leq t \leq 1$ , and  $H(\mathbf{L}_i, t \cdot \mathbf{u}_{i \rightarrow i-1})$  transforms the frame  $\mathbf{L}_i$  to  $\mathbf{L}_{i-t}$ . Our bi-directional optical flows, duty cycle, and the corresponding piece-wise linear kernel used in our blur model are illustrated in Fig. 3.

Although our blur kernel model is simple, our model can be justified because we treat video that has short exposure time to some extent. Therefore, we approximate the kernel as piece-wise linear using bidirectional optical flows:

$$k_{i,x}(u, v) = \begin{cases} \frac{\delta(uv_{i \rightarrow i+1} - v_{i \rightarrow i+1}u)}{2\tau_i \|\mathbf{u}_{i \rightarrow i+1}\|}, & \text{if } u \in [0, \tau_i u_{i \rightarrow i+1}], v \in [0, \tau_i v_{i \rightarrow i+1}] \\ \frac{\delta(uv_{i \rightarrow i-1} - v_{i \rightarrow i-1}u)}{2\tau_i \|\mathbf{u}_{i \rightarrow i-1}\|}, & \text{if } u \in (0, \tau_i u_{i \rightarrow i-1}], v \in (0, \tau_i v_{i \rightarrow i-1}] \\ 0, & \text{otherwise.} \end{cases} \quad (3)$$

where  $k_{i,x}(u, v)$  is the blur kernel using bidirectional optical flows at pixel location  $\mathbf{x}$ , and  $\delta$  denotes Kronecker delta.

Using this pixel-wise kernel approximation, we can easily manage multiple different blurs in a frame, unlike conventional methods. The superiority of our kernel model is shown in Fig. 4. Our kernel model fits blurs from differently moving objects and camera shake much better than the conventional homography-based model.

Therefore, we cast pixel-wise kernel estimation problem as an optical flows estimation problem. Discretizing the

constraint (2) gives the following data term:

$$\mathbf{E}_{data}(\mathbf{L}, \mathbf{u}, \mathbf{B}) = \lambda \sum_i \sum_{\partial_*} \|\partial_* \mathbf{K}_i(\tau_i, \mathbf{u}_{i \rightarrow i+1}, \mathbf{u}_{i \rightarrow i-1}) \mathbf{L}_i - \partial_* \mathbf{B}_i\|^2, \quad (4)$$

where the row vector of blur kernel matrix  $\mathbf{K}_i$ , corresponding to the blur kernel at pixel  $\mathbf{x}$ , is the vector form of  $k_{i,x}(\cdot)$ , and its elements are non-negative and their sum is equal to one. Linear operator  $\partial_*$  denotes the Toeplitz matrices corresponding to the partial (e.g., horizontal and vertical) derivative filters. Parameter  $\lambda$  controls the weight of the data term, and  $\mathbf{L}$ ,  $\mathbf{u}$ , and  $\mathbf{B}$  denote the set of latent frames, optical flows, and blurry frames, respectively.

## 2.2. Temporal Coherence with Optical Flow Constraint

Here, we determine that optical flows are required to estimate the pixel-wise blur kernel. However, the proposed data term does not have conventional optical flow constraints such as brightness constancy or gradient constancy in (4). In general, such constraints do not hold between two blurry frames. Thus, Portz et al. [24] proposed a method to apply flow constraints between blurry images. Based on the commutative law of shift invariance of kernels [13], the authors of [24] convolved the approximated blur of each observed image to the other image and assumed constant brightness between them at matched points. However, the commutativity property does not hold in theory when the kernel is not translation invariant. Therefore, this approach only works when the motion is smooth enough.

To address this problem, we propose a new model that finds correspondences between two latent sharp images to enable abrupt changes in motions and the corresponding kernels. In using this model, we need not restrict our blur kernels to be shift invariant. Our model is based on the conventional optical flow constraint between latent images, that is, brightness constancy. The formulation is expressed as follows:

$$\mathbf{E}_{temporal}(\mathbf{L}, \mathbf{u}) = \sum_i \sum_{n=-N}^N \mu_n |\mathbf{L}_i(\mathbf{x}) - \mathbf{L}_{i+n}(\mathbf{x} + \mathbf{u}_{i \rightarrow i+n})|, \quad (5)$$

where  $n$  denotes the index of neighboring frames at  $i$ . Constant parameter  $\mu_n$  controls the weight of each term in the summation. We apply the robust  $L^1$  norm to offer robustness against outliers and occlusions.

Notably, a major difference between the proposed model and the conventional optical flow estimation methods is that our problem is a joint problem. That is, the brightness of latent frames and optical flows need to be simultaneously estimated. Therefore, our model simultaneously enforces the temporal coherence of latent frames and estimates the correspondences.

## 2.3. Spatial Coherence

To alleviate the difficulties of highly ill-posed deblurring and optical flow estimation problems, several researchers have emphasized the importance of spatial regularization. Therefore, we also enforce spatial coherence to penalize spatial fluctuations while allowing discontinuities in both latent frames and flow fields. We assume that spatial priors for latent frames and optical flows are independent. They are expressed as follows:

$$\mathbf{E}_{spatial}(\mathbf{L}, \mathbf{u}) = \sum_i |\nabla \mathbf{L}_i| + \sum_{n=-N}^N g_i(\mathbf{x}) |\nabla \mathbf{u}_{i \rightarrow i+n}|. \quad (6)$$

The first term in (6) denotes the spatial regularization term for the latent frames. Although more sparse  $L^p$  norms (e.g.,  $p = 0.8$ ) fit the gradient statistics of natural sharp images better [17, 18, 20], we use conventional total variation (TV) based regularization [12, 14, 16], as TV is computationally less expensive. The second term denotes the spatial smoothness term for optical flows. We adopt edge-map coupled TV-based regularization [15] to preserve discontinuities in the flow fields at edges. Similar to [16], the edge-map is expressed as follows:

$$g_i(\mathbf{x}) = \nu \exp\left(-\left(\frac{|\nabla \bar{\mathbf{L}}_i|}{\sigma_I}\right)^2\right), \quad (7)$$

where  $\nu$  controls the scale of the edge-map, parameter  $\sigma_I$  controls the weight, and  $\bar{\mathbf{L}}_i$  is an initial latent image in the iterative optimization framework.

## 3. Optimization Framework

In the previous sections, we described the  $\mathbf{E}_{data}$ ,  $\mathbf{E}_{temporal}$ , and  $\mathbf{E}_{spatial}$  terms. When camera duty cycle  $\tau_i$  is known, our final objective function becomes as follows:

$$\begin{aligned} \min_{\mathbf{L}, \mathbf{u}} \lambda \sum_i \sum_{\partial_*} \|\partial_* \mathbf{K}_i(\mathbf{u}_{i \rightarrow i+1}, \mathbf{u}_{i \rightarrow i-1}) \mathbf{L}_i - \partial_* \mathbf{B}_i\|^2 + \\ \sum_i \sum_{n=-N}^N \mu_n \cdot |\mathbf{L}_i(\mathbf{x}) - \mathbf{L}_{i+n}(\mathbf{x} + \mathbf{u}_{i \rightarrow i+n})| + \\ \sum_i |\nabla \mathbf{L}_i| + \sum_{n=-N}^N g_i(\mathbf{x}) |\nabla \mathbf{u}_{i \rightarrow i+n}|. \end{aligned} \quad (8)$$

Unlike the work of Cho et al. [7], which sequentially performs multi-phase approaches, our model obtains a solution by minimizing a single objective function. However, because of its non-convexity, our model is required to adopt practical optimization methods to obtain approximated solution. Therefore, we divide the original problem into two

sub-problems and use conventional iterative and alternating optimization techniques [5, 28] to minimize the non-convex objective function. In the following sections, we introduce efficient solvers and describe how to estimate unknowns  $\mathbf{L}$  and  $\mathbf{u}$ , with one of them being fixed.

### 3.1. Sharp Video Restoration

While the optical flows  $\mathbf{u}$  are fixed, corresponding blur kernels are also fixed, and our objective function in (8) becomes convex with respect to  $\mathbf{L}$ , and is expressed as follows:

$$\begin{aligned} \min_{\mathbf{L}} \lambda \sum_i \sum_{\partial_*} \|\partial_* \mathbf{K}_i \mathbf{L}_i - \partial_* \mathbf{B}_i\|^2 + \\ \sum_i \sum_{n=-N}^N \mu_n \cdot |\mathbf{L}_i(\mathbf{x}) - \mathbf{L}_{i+n}(\mathbf{x} + \mathbf{u}_{i \rightarrow i+n})| + |\nabla \mathbf{L}_i|. \end{aligned} \quad (9)$$

To obtain  $\mathbf{L}$ , we adopt the conventional convex optimization method in [3], and derive the primal-dual update scheme as follows:

$$\begin{cases} \mathbf{s}_i^{m+1} = \frac{\mathbf{s}_i^m + \eta_L \mathbf{A} \mathbf{L}_i^m}{\max(\mathbf{1}, \text{abs}(\mathbf{s}_i^m + \eta_L \mathbf{A} \mathbf{L}_i^m))} \\ \mathbf{q}_{i,n}^{m+1} = \frac{\mathbf{q}_{i,n}^m + \eta_L \mu_n \mathbf{D}_{i,n} \begin{bmatrix} \mathbf{L}_i^m \\ \mathbf{L}_{i+n}^m \end{bmatrix}}{\max(\mathbf{1}, \text{abs}(\mathbf{q}_{i,n}^m + \eta_L \mu_n \mathbf{D}_{i,n} \begin{bmatrix} \mathbf{L}_i^m \\ \mathbf{L}_{i+n}^m \end{bmatrix}))} \\ \mathbf{L}_i^{m+1} = \arg \min_{\mathbf{L}_i} \lambda \sum_{\partial_*} (\partial_* \mathbf{K}_i \mathbf{L}_i - \partial_* \mathbf{B}_i)^2 + \\ \frac{(\mathbf{L}_i - (\mathbf{L}_i^m - \epsilon_L (\mathbf{A}^T \mathbf{s}_i^{m+1} + \sum_{n=-N}^N \mu_n \mathbf{D}_{i,n}^T \mathbf{q}_{i,n}^{m+1})))^2}{2\epsilon_L} \end{cases} \quad (10)$$

where  $m \geq 0$  indicates the iteration number, and  $\mathbf{s}_i$  and  $\mathbf{q}_{i,n}$  denote the dual variables. Parameters  $\eta_L$  and  $\epsilon_L$  denote the update steps. A linear operator  $\mathbf{A}$  calculates the spatial difference between neighboring pixels, and another operator  $\mathbf{D}_{i,n}$  calculates the temporal differences between  $\mathbf{L}_i(\mathbf{x})$  and  $\mathbf{L}_{i+n}(\mathbf{x} + \mathbf{u}_{i \rightarrow i+n})$ . To update the primal variable and obtain  $\mathbf{L}_i^{m+1}$  in (10), we apply the conjugate gradient method to optimize the quadratic function.

### 3.2. Optical Flows Estimation

While the latent frames  $\mathbf{L}$  are fixed, temporal coherence term  $\mathbf{E}_{temporal}$  becomes convex but the data term  $\mathbf{E}_{data}$  remains non-convex. Therefore, we define a non-convex fidelity function  $\rho(\cdot)$  as follows:

$$\begin{aligned} \rho(\mathbf{x}, \mathbf{u}) = \lambda \sum_i \sum_{\partial_*} \|\partial_* \mathbf{K}_i(\mathbf{u}_{i \rightarrow i+1}, \mathbf{u}_{i \rightarrow i-1}) \mathbf{L}_i - \partial_* \mathbf{B}_i\|^2 + \\ \sum_i \sum_{n=-N}^N \mu_n \cdot |\mathbf{L}_i(\mathbf{x}) - \mathbf{L}_{i+n}(\mathbf{x} + \mathbf{u}_{i \rightarrow i+n})|. \end{aligned} \quad (11)$$



Figure 5: Temporally consistent optical flows over three frames.

To find the optimized values of optical flows  $\mathbf{u}$ , we first convexify the non-convex function  $\rho(\cdot)$  by applying the first-order Taylor expansion. Similar to [16], we linearize the function near an initial  $\mathbf{u}_0$  in the iterative process as follows:

$$\rho(\mathbf{x}, \mathbf{u}) \approx \rho(\mathbf{x}, \mathbf{u}_0) + \nabla \rho(\mathbf{x}, \mathbf{u}_0)^T (\mathbf{u} - \mathbf{u}_0). \quad (12)$$

Therefore, our approximated convex function for optical flows estimation is expressed as follows:

$$\min_{\mathbf{u}} \rho(\mathbf{x}, \mathbf{u}_0) + \nabla \rho(\mathbf{x}, \mathbf{u}_0)^T (\mathbf{u} - \mathbf{u}_0) + \sum_i \sum_{n=-N}^N g_i(\mathbf{x}) |\nabla \mathbf{u}_{i \rightarrow i+n}|. \quad (13)$$

Next, we apply the convex optimization technique in [3] to the approximated convex function (13), and the primal-dual update process is expressed as follows:

$$\begin{cases} \mathbf{p}_{i,n}^{m+1} = \frac{\mathbf{p}_{i,n}^m + \eta_u (\mathbf{G}_i \mathbf{A}) \mathbf{u}_{i \rightarrow i+n}^m}{\max(\mathbf{1}, \text{abs}(\mathbf{p}_{i,n}^m + \eta_u (\mathbf{G}_i \mathbf{A}) \mathbf{u}_{i \rightarrow i+n}^m))} \\ \mathbf{u}_{i \rightarrow i+n}^{m+1} = (\mathbf{u}_{i \rightarrow i+n}^m - \epsilon_u (\mathbf{G}_i \mathbf{A})^T \mathbf{p}_{i,n}^{m+1}) - \epsilon_u \nabla_{i,n} \rho(\mathbf{x}, \mathbf{u}_0), \end{cases} \quad (14)$$

where  $\mathbf{p}_{i,n}$  denotes the dual variable of  $\mathbf{u}_{i \rightarrow i+n}$  on the vector space and the diagonal matrix  $\mathbf{G}_i$  is the weighting matrix denoted as  $\mathbf{G}_i = \text{diag}(g_i(\mathbf{x}))$ . Parameters  $\eta_u$  and  $\epsilon_u$  denote the update steps and  $\nabla_{i,n} \rho(\mathbf{x}, \mathbf{u}_0)$  means  $\frac{\partial \rho(\mathbf{x}, \mathbf{u})}{\partial \mathbf{u}_{i \rightarrow i+n}} \Big|_{\mathbf{u}_0}$ .

## 4. Implementation Details

To handle large blurs and guide fast convergence, we implement our algorithm on the traditional coarse-to-fine framework with empirically determined parameters. We use  $\lambda = 250$  for our most experiments, and other parameters are determined as  $\mu_n = \lambda$ ,  $\nu = 0.08\lambda$ ,  $\sigma_I = \frac{25}{255}$ , and  $N = 2$ . In the coarse-to-fine framework, we build image pyramid with 17 levels for a high-definition(1280x720) video, the scale factor is 0.9, and use bi-cubic interpolation to propagate both the optical flows and latent frames to the next pyramid level.

Moreover, to reduce the number of unknowns in optical flows, we only estimate  $\mathbf{u}_{i \rightarrow i+1}$  and  $\mathbf{u}_{i \rightarrow i-1}$ . We approximate  $\mathbf{u}_{i \rightarrow i+2}$  using  $\mathbf{u}_{i \rightarrow i+1}$  and  $\mathbf{u}_{i+1 \rightarrow i+2}$ . For example, it satisfies,  $\mathbf{u}_{i \rightarrow i+2} = \mathbf{u}_{i \rightarrow i+1} + \mathbf{u}_{i+1 \rightarrow i+2}$ , as illustrated in Fig. 5, and we can easily apply this for  $n \neq 1$ .

The overall process of our algorithm is in Algorithm 1. Further details on estimating the duty cycle  $\tau_i$  and post-processing step that reduces artifacts are given below.

---

**Algorithm 1** Overview of the proposed method

---

**Input:** Blurry frames  $\mathbf{B}$

**Output:** Latent frames  $\mathbf{L}$  and optical flows  $\mathbf{u}$

- 1: Initialize duty cycle  $\tau_i$  and optical flows  $\mathbf{u}$ . (Sec. 4.1)
  - 2: Build image pyramid.
  - 3: Restore *sharp video* with fixed  $\mathbf{u}$ . (Sec. 3.1)
  - 4: Estimate *optical flows* with fixed  $\mathbf{L}$ . (Sec. 3.2)
  - 5: Detect occlusion and perform post-processing. (Sec. 4.2)
  - 6: Propagate variables to the next pyramid level if exists.
  - 7: Repeat steps 3-6 from coarse to fine pyramid level.
- 

### 4.1. Duty Cycle Estimation

In this study, we assume that the camera duty cycle  $\tau_i$  is known for every frame. We can obtain the duty cycle from public SDK, when we use Kinect to capture RGB videos. However, when we conduct deblurring with conventional data sets, which do not provide exposure information, we apply the technique proposed in [7] to estimate the duty cycle. Contrary to the original method in [7], we use optical flows instead of homographies to obtain initially approximated blur kernels. Therefore, we first estimate flow fields from blurry images with [26], which runs in near real-time. We then use them as initial flows and approximate the kernels to estimate the duty cycle.

### 4.2. Occlusion Detection and Refinement

Our piece-wise linear kernel naturally results in approximation error and it causes problems such as ringing artifacts. Moreover, our data model in (4), and temporal coherence model in (5) are invalid at occluded regions.

To reduce such artifacts from kernel errors and occlusions, we use spatio-temporal filtering as a post-processing:

$$\mathbf{L}_i^{m+1}(\mathbf{x}) = \frac{1}{Z(\mathbf{x})} \sum_{n=-N}^N \sum_{\mathbf{y}} w_{i,n}(\mathbf{x}, \mathbf{y}) \cdot \mathbf{L}_{i+n}^m(\mathbf{y}), \quad (15)$$

where  $\mathbf{y}$  denotes a pixel in the 3x3 neighboring patch at location  $(\mathbf{x} + \mathbf{u}_{i \rightarrow i+n})$  and  $Z$  is the normalization factor (e.g.  $Z(\mathbf{x}) = \sum_{n=-N}^N \sum_{\mathbf{y}} w_{i,n}(\mathbf{x}, \mathbf{y})$ ). Notably, we enable  $n = 0$  in (15) for spatial filtering. Our occlusion-aware weight  $w_{i,n}$  is defined as follows:

$$w_{i,n}(\mathbf{x}, \mathbf{y}) = o_{i,n}(\mathbf{x}, \mathbf{y}) \cdot \exp\left(-\frac{\|P_i(\mathbf{x}) - P_{i+n}(\mathbf{y})\|^2}{2\sigma_w^2}\right), \quad (16)$$

where occlusion state  $o_{i,n}(\mathbf{x}, \mathbf{y}) \in \{0, 0.5, 1\}$  is determined using the method proposed in [15]. The 5x5 patch  $P_i(\mathbf{x})$  is centered at  $\mathbf{x}$  in frame  $i$ . The similarity control parameter  $\sigma_w$  is fixed as  $\sigma_w = 25/255$ .

## 5. Experimental Results

In what follows, we demonstrate the superiority of the proposed method. (For more results, see the supplementary video.)

First, we compare our deblurring results with those of the state-of-the-art exemplar based method [7] with the videos used in [7]. As shown in Fig. 6, the captured scenes are dynamic and contain multiple moving objects. The method [7] fails in restoring the moving objects, because the object motions are large and distinct from the backgrounds. By contrast, our results show better performances in deblurring moving objects and backgrounds. This exemplar-based approach also fails in handling large blurs, as shown in Fig. 7, as the initially estimated homographies in the largely blurred images are inaccurate. Moreover, this approach renders excessively smooth results for mid-frequency textures such as trees, as the method is based on interpolation without spatial prior for latent frames.

Next, we compare our method with the state-of-the-art segmentation-based approach [28]. In Fig. 8, the captured scene is a bi-layer and used in [28]. Although the bi-layer scene is a good example to verify the performance of the layered model, inaccurate segmentation near the boundaries causes serious artifacts in the restored frame. By contrast, our method does not depend on accurate segmentation and thus restores the boundaries much better than the layered model.

In Fig. 9, we quantitatively compare the optical flow accuracies with [24] on synthetic blurry images. Although [24] proposed to handle blurry images in optical flow estimation, its assumption does not hold in motion boundaries, which are very important for deblurring. Therefore, their optical flow is inaccurate in the motion boundaries of moving objects. However, our model enables abrupt changes of motions and thus performs better than the previous model.

Moreover, we show the deblurring results with and without using the temporal coherence term in (5), and verify that our temporal coherence model significantly reduces ringing artifacts near the edges in Fig. 10.

Other deblurring results from numerous real videos are shown in Fig. 11. Notably, our model successfully restores the face which has highly non-uniform blurs because the person moves rotationally (Fig. 11(e)).

## 6. Conclusions

In this study, we introduced a novel method that removes general blurs in dynamic scenes, which conventional methods fail to do. By estimating a pixel-wise kernel using optical flows, we handled general blurs. Thus, we proposed a new energy model that estimates optical flows and latent frames, jointly.

We also provided a framework and efficient solvers to

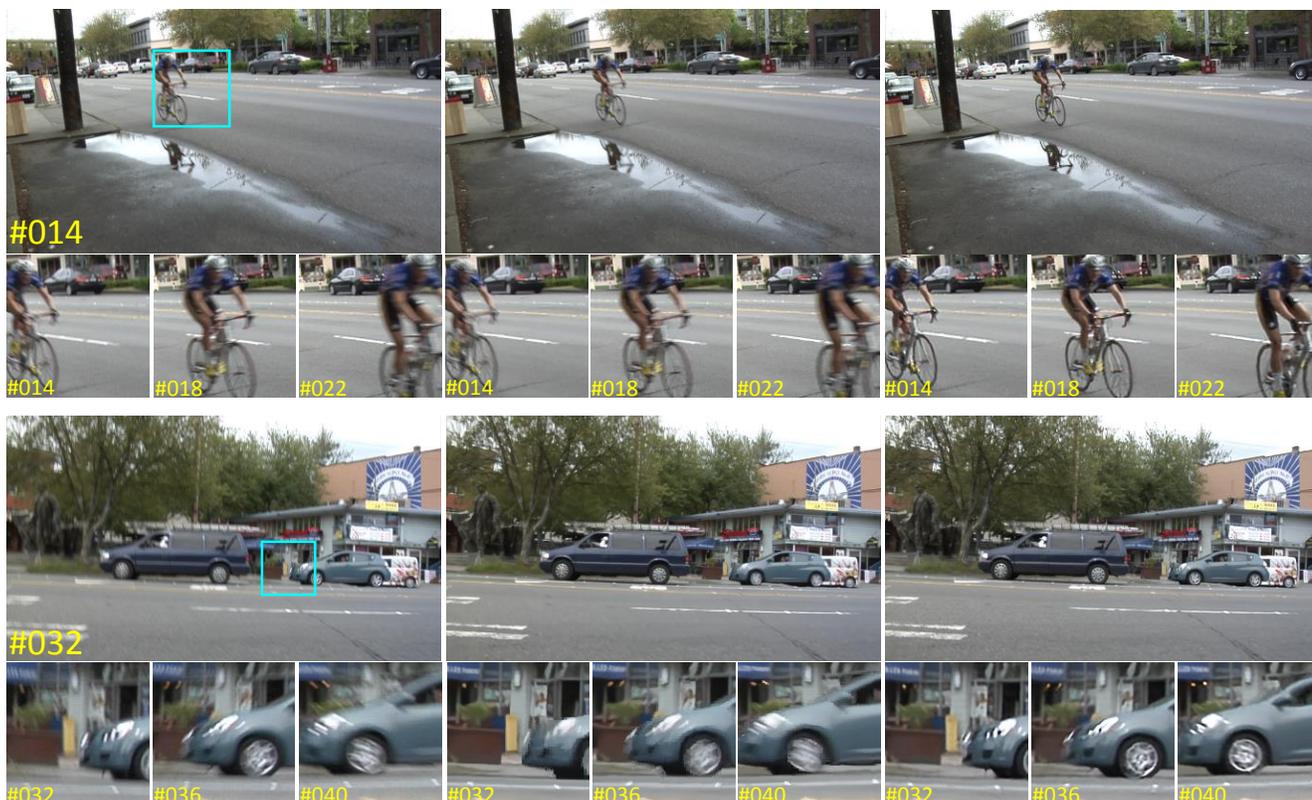


Figure 6: **Left to right:** Blurry frames of dynamic scenes, deblurring results of [7], and our results.



Figure 7: **Left to right:** Blurry frame, deblurring result of [7], and ours.



Figure 8: Comparison with segmentation-based approach. **Left to right:** Blurry frame, result of [28], and ours.

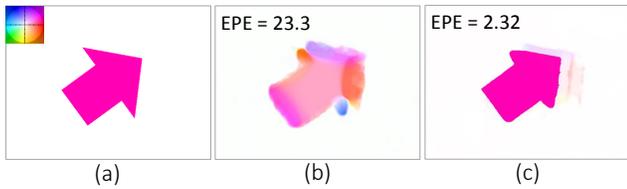


Figure 9: EPE denotes average end point error. (a) Color coded ground truth optical flow between blurry images. (b) Optical flow estimation result of [24]. (c) Our result.

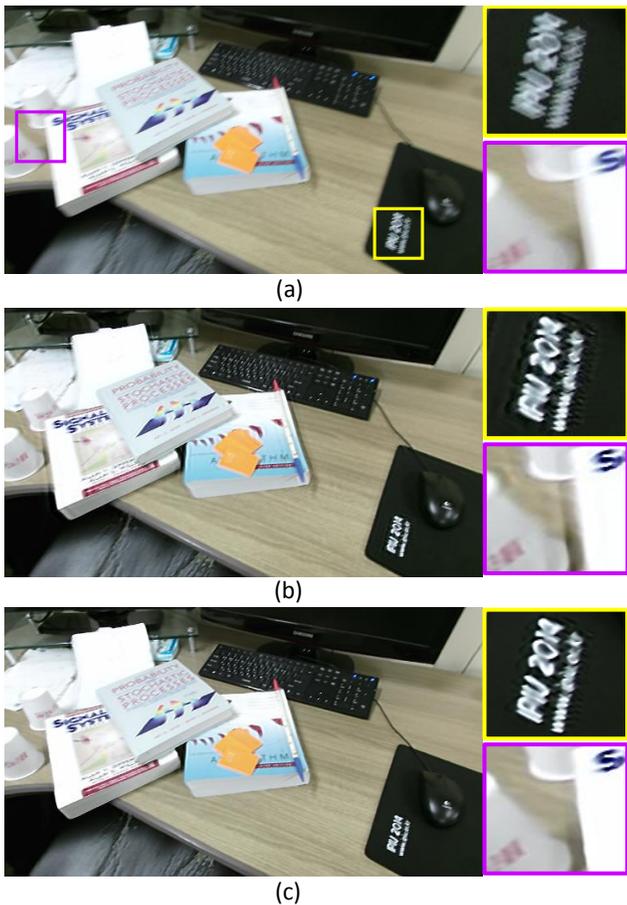


Figure 10: (a) Real blurry frame of a video. (b) Our deblurring result without using  $E_{temporal}$ . (c) Our deblurring result with  $E_{temporal}$ .

minimize the energy function and achieved significant improvements in removing general blurs in dynamic scenes.



Figure 11: **Left to right:** Numerous real blurry frames and our deblurring results. (a)-(b) Data sets used in [7]. (c)-(e) Captured RGB data sets using kinect.

## 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), and in part by the National Research Foundation of Korea (NRF) grant funded by the Ministry of Science, ICT & Future Planning (MSIP) (No. 2009-0083495)

## References

- [1] L. Bar, B. Berkels, M. Rumpf, and G. Sapiro. A variational framework for simultaneous motion estimation and restoration of motion-blurred video. In *Proc. IEEE International Conference on Computer Vision and Pattern Recognition*, 2007. 2, 3
- [2] J.-F. Cai, H. Ji, C. Liu, and Z. Shen. Blind motion deblurring using multiple images. *Journal of computational physics*, 228(14):5057–5071, 2009. 1
- [3] 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. 5
- [4] S. Cho, H. Cho, Y.-W. Tai, and S. Lee. Registration based non-uniform motion deblurring. In *Computer Graphics Forum*, volume 31, pages 2183–2192. Wiley Online Library, 2012. 1
- [5] S. Cho and S. Lee. Fast motion deblurring. In *SIGGRAPH*, 2009. 1, 5
- [6] S. Cho, Y. Matsushita, and S. Lee. Removing non-uniform motion blur from images. In *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, pages 1–8. IEEE, 2007. 2
- [7] S. Cho, J. Wang, and S. Lee. Video deblurring for hand-held cameras using patch-based synthesis. *ACM Transactions on Graphics*, 31(4):64:1–64:9, 2012. 1, 2, 3, 4, 6, 7, 8
- [8] S. Dai and Y. Wu. Motion from blur. In *Proc. IEEE International Conference on Computer Vision and Pattern Recognition*, 2008. 3
- [9] R. Fergus, B. Singh, A. Hertzmann, S. T. Roweis, and W. Freeman. Removing camera shake from a single photograph. In *SIGGRAPH*, 2006. 1
- [10] A. Gupta, N. Joshi, L. Zitnick, M. Cohen, and B. Curless. Single image deblurring using motion density functions. In *ECCV*, 2010. 1
- [11] M. Hirsch, C. J. Schuler, S. Harmeling, and B. Scholkopf. Fast removal of non-uniform camera shake. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 463–470. IEEE, 2011. 1
- [12] Z. Hu, L. Xu, and M.-H. Yang. Joint depth estimation and camera shake removal from single blurry image. In *Proc. IEEE International Conference on Computer Vision and Pattern Recognition*, 2014. 4
- [13] H. Jin, P. Favaro, and R. Cipolla. Visual tracking in the presence of motion blur. In *Proc. IEEE International Conference on Computer Vision and Pattern Recognition*, 2005. 4
- [14] T. H. Kim, B. Ahn, and K. M. Lee. Dynamic scene deblurring. In *Computer Vision (ICCV), 2013 IEEE International Conference on*, pages 3160–3167. IEEE, 2013. 4
- [15] T. H. Kim, H. S. Lee, and K. M. Lee. Optical flow via locally adaptive fusion of complementary data costs. In *Computer Vision (ICCV), 2013 IEEE International Conference on*, pages 3344–3351. IEEE, 2013. 4, 6
- [16] T. H. Kim and K. M. Lee. Segmentation-free dynamic scene deblurring. In *Proc. IEEE International Conference on Computer Vision and Pattern Recognition*, 2014. 2, 3, 4, 5
- [17] D. Krishnan and R. Fergus. Fast image deconvolution using hyper-laplacian priors. In *NIPS*, 2009. 4
- [18] D. Krishnan, T. Tay, and R. Fergus. Blind deconvolution using a normalized sparsity measure. In *Proc. IEEE International Conference on Computer Vision and Pattern Recognition*, 2009. 4
- [19] H. S. Lee and K. M. Lee. Dense 3d reconstruction from severely blurred images using a single moving camera. In *Proc. IEEE International Conference on Computer Vision and Pattern Recognition*, 2013. 1
- [20] 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. 4
- [21] Y. Li, S. B. Kang, N. Joshi, S. M. Seitz, and D. P. Huttenlocher. Generating sharp panoramas from motion-blurred videos. In *Proc. IEEE International Conference on Computer Vision and Pattern Recognition*, 2010. 1, 3
- [22] Y. Matsushita, E. Ofek, W. Ge, X. Tang, and H.-Y. Shum. Full-frame video stabilization with motion inpainting. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 28(7):1150–1163, 2006. 2
- [23] C. Paramanand and A. N. Rajagopalan. Non-uniform motion deblurring for bilayer scenes. In *Proc. IEEE International Conference on Computer Vision and Pattern Recognition*, 2013. 1
- [24] T. Portz, L. Zhang, and H. Jiang. Optical flow in the presence of spatially-varying motion blur. In *Proc. IEEE International Conference on Computer Vision and Pattern Recognition*, 2012. 3, 4, 6, 8
- [25] Q. Shan, J. Jia, and A. Agarwala. High-quality motion deblurring from a single image. In *SIGGRAPH*, 2008. 1
- [26] A. Wedel, T. Pock, C. Zach, H. Bischof, and D. Cremers. An improved algorithm for tv-l1 optical flow. In *Statistical and Geometrical Approaches to Visual Motion Analysis*, pages 23–45. Springer, 2009. 6
- [27] 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
- [28] J. Wulff and M. J. Black. Modeling blurred video with layers. In *ECCV*, 2014. 2, 3, 5, 6, 7