

# Motion Regularization for Matting Motion Blurred Objects

Hai Ting Lin, Yu-Wing Tai, and Michael S. Brown

**Abstract**—This paper addresses the problem of matting motion blurred objects from a single image. Existing single image matting methods are designed to extract static objects that have fractional pixel occupancy. This arises because the physical scene object has a finer resolution than the discrete image pixel and therefore only occupies a fraction of the pixel. For a motion blurred object, however, fractional pixel occupancy is attributed to the object's motion over the exposure period. While conventional matting techniques can be used to matte motion blurred objects, they are not formulated in a manner that considers the object's motion and tend to work only when the object is on a homogeneous background. We show how to obtain better alpha mattes by introducing a regularization term in the matting formulation to account for the object's motion. In addition, we outline a method for estimating local object motion based on local gradient statistics from the original image. For the sake of completeness, we also discuss how user markup can be used to denote the local direction in lieu of motion estimation. Improvements to alpha mattes computed with our regularization are demonstrated on a variety of examples.

**Index Terms**—Matting, regularization, motion direction estimation, motion blur.

## 1 INTRODUCTION

SEGMENTING a motion blurred object from a background scene is desirable for various image processing tasks such as image editing (i.e., cut and paste) and image deblurring. Such object segmentation is inherently a matting problem given that the object's motion over the exposure time results in a mixture of foreground (FG) and background colors. This mixture can be expressed as

$$I = P_s(S) + (1 - P_s(M_s))B, \quad (1)$$

where  $I$  is the observed image,  $P_s(\cdot)$  is the point spread function (PSF) of the motion blur,  $S$  is the foreground object,  $M_s$  is the binary mask of the object, and  $B$  is the background image. Here, we assume that the moving object is opaque and in sharp focus and the fractional pixel occupancy targeted by conventional matting is negligible as the significant color mixing effect is attributed to the motion. We also assume that the motion blur region does not contain any saturated pixels. Combining (1) with the conventional matting equation  $I = \alpha F + (1 - \alpha)B$  [1], where  $F$  is the foreground object and  $\alpha \in [0, 1]$ , we obtain

$$\alpha = P_s(M_s), \quad F = P_s(S)/P_s(M_s), \quad B = B. \quad (2)$$

Working from (2), one solution to compute  $\alpha$  would be to first estimate the motion blur PSF,  $P_s(\cdot)$ . Applying the inverse motion blur,  $P_s^{-1}(\alpha)$  would, in an ideal situation, result in a binary image,  $M_s$ , with hard boundaries. This suggests the possibility of designing an alternative optimization strategy to iteratively solve for  $\alpha$  and  $M_s$  using  $P_s(\cdot)$ . This approach, however, relies heavily on

obtaining a satisfactory estimation of the object's PSF from the image. State-of-the-art techniques (see [3] for a review) almost exclusively estimate PSFs that are spatially invariant and are only valid for moving objects with in-plane translational motion. In practice, however, motion blurred objects include complex motion such as rotation, zoom in/out, and even arbitrary deformations. Second, in the case where a suitable  $P_s(\cdot)$  can be found, the original binary mask of the object  $M_s$  must still be estimated through some form of motion deblurring and thresholding, which presents challenges of its own.

We propose a solution to the motion matting problem that avoids estimating the object's PSF and original hard boundary. Instead, we assume that an object's motion can be described by a series of piecewise 1D translational motions, a representation that has been shown to be a highly effective approximation for many real-world scenarios [4]. Our approach is to incorporate an estimation of the local 1D motion of the blurred object into the matting process to regularize the matte, as shown in Fig. 1. To do this, we first describe how to estimate the local object's motion based on local gradient statistics from the original image, or by simple user markup. We then describe how to incorporate a regularization term based on the estimated motion directions as a soft constraint in two existing matting techniques: closed-form matting formulation [2] and robust matting (RM) [5]. The effectiveness of our approach is demonstrated on a variety of inputs.

The remainder of this paper is organized as follows: Section 2 discusses related work, Section 3 describes local motion estimation, Section 4 presents our algorithm for matting, and Section 5 presents results and comparisons with other approaches. A discussion and summary of this work is presented in Section 6.

## 2 RELATED WORK

The focus of alpha matting is to perform a soft segmentation of a foreground object from the background scene. While prior work has used matting to extract soft segmentation of motion blurred objects, the matting techniques utilized were designed for static objects and did not consider the objects' motion. To the best of our knowledge, there is no previous image matting work that explicitly targets obtaining mattes of motion blurred objects. We therefore discuss existing conventional matting (i.e., static object) techniques first. For a more complete survey on matting, readers are referred to [6]. We also discuss recent techniques that use alpha mattes as inputs for object deblurring and motion estimation.

### 2.1 Matting Algorithms

Image matting can be classified into two types: 1) single image approaches with user-supplied markup and 2) multi-image approaches exploiting hardware or imaging manipulation. In single image approaches, the user supplies a *trimap* (typically user-supplied scribbles) that identifies definite *foreground* regions, definite *background* regions, and the *unknown* regions where the alpha values need to be estimated. Representative work includes Bayesian matting [1], Poisson matting [7], and closed-form matting [2]. The basic idea of these approaches is to use the definite foreground and background regions as hard constraints to infer the alpha values within the unknown regions. This is done by assuming that the colors within local regions are smooth [1], the color gradients are smooth [7], or the local color distribution satisfies a linear model [2]. Soft scissors [8] proposed an interactive user interface to identify foreground, background, and unknown regions along an object's boundary. This work was based on robust matting [5], which incorporates better sampling in the closed-form matting formulation. Recently, Rhemann et al. [9] proposed a PSF-based matting algorithm which included a deconvolution step into their framework. Their approach, however, targets blurring effects caused by limited resolution or out-of-focus blur.

• H.T. Lin and M.S. Brown are with the School of Computing, National University of Singapore, 13 Computing Link, Singapore 117590. E-mail: {linhait, brown}@comp.nus.edu.sg.

• Y.-W. Tai is with the Korea Advanced Institute of Science and Technology (KAIST), Rm 2425, E3-1 CS Building, 373-1 Guseong-dong, Yuseong-gu, Daejeon 305-701, South Korea. E-mail: yuwing@cs.kaist.ac.kr.

Manuscript received 1 Sept. 2010; revised 14 Mar. 2011; accepted 17 Mar. 2011; published online 28 Apr. 2011.

Recommended for acceptance by A. Fitzgibbon.

For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number TPAMI-2010-09-0677.

Digital Object Identifier no. 10.1109/TPAMI.2011.93.



Fig. 1. (a) An image with motion blurred foreground object, (b) conventional matte obtained from closed-form matting [2], (c) estimated local motion, and (d) motion matting using [2] with motion regularization incorporated.

Compared to single image approaches, multi-image approaches are fully automatic. These techniques capture several images to identify foreground, background, and unknown regions automatically. Representative work includes defocus matting [10], camera array matting [11], and flash matting [12]. In [10], a set of images with different focal settings is captured. By analyzing the amount of defocus blur, a trimap can be automatically computed. Similarly to [10], Joshi et al. [11] use a camera array to capture multiple images and estimate the trimap based on stereoview analysis. Work in [12] takes two images of the same scene with and without flash. By assuming that the foreground object is near and the background is distant (and unaffected by the flash), a trimap can be automatically computed.

## 2.2 Mattes for Blurred Images

Several deblurring approaches [13], [14], [15], [16], [17], [18] use matting to first segment a motion blurred object from the background before applying deconvolution. Alpha matting has also been used as the input to directly estimate the PSF of an object [19], [20]. In [4], local motions of a blurred object are estimated directly from an alpha matte of the blurred object. Interestingly, these approaches [19], [20], [4] work from results based on conventional matting that is not designed to matte motion blurred objects. This can lead to unsatisfactory results, as discussed in [20].

Compared to prior work, our approach is unique in its focus on matting of motion blurred objects and its use of local gradient statistics to estimate local motion. Related work in [4], [21] has addressed the inverse problem which uses alpha mattes to help estimate object motion. Because these approaches rely on conventional matting that is not designed for motion blurred objects, the examples demonstrated in this prior work target objects on homogenous backgrounds.

## 3 LOCAL MOTION DIRECTION ESTIMATION

Since our approach uses local motion estimation in the proposed regularization, we first describe how we extract and represent motion directions. The actual motion regularization term is described in Section 4. Two methods to obtain the local motions are discussed: 1) automatic estimation from local gradient statistics and 2) interactive estimation based on user markup.

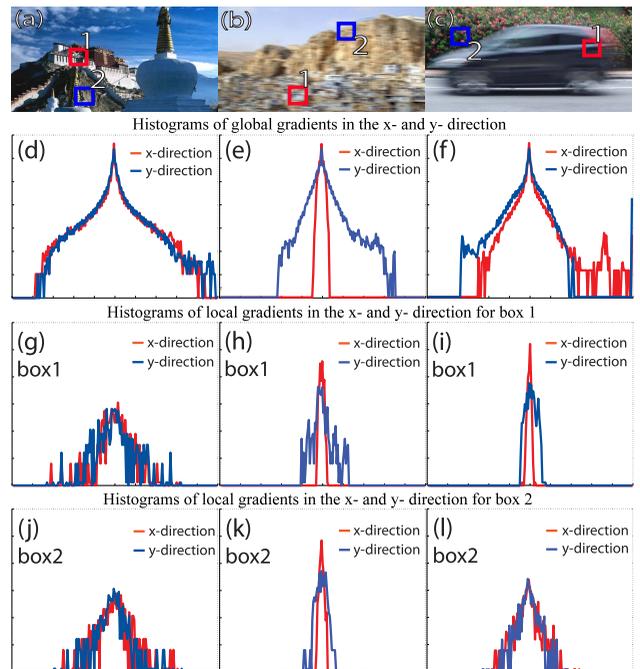


Fig. 2. Global and local distributions for gradients in the x and y-directions are shown for a natural image, a globally motion blurred image, and an image containing a motion blurred object.

## 3.1 Motion from Local Gradient Statistics

Local motion estimation is based on the observation that motion blurring smoothes gradients parallel to the motion direction but has significantly less effect on gradients perpendicular to the motion direction. This statistical property has been exploited for blur detection [22] and blur classification [23], but not for estimating local blur direction.

Fig. 2 is provided to help illustrate the idea. Shown are the gradient magnitude distributions for a natural image (Fig. 2a), a globally motion blurred image (Fig. 2b), and a natural image containing a motion blurred object (Fig. 2c). Two local regions ( $19 \times 19$ ) are selected (labeled as box 1 and box 2). The global statistics of the gradient distributions are plotted in the second row of Fig. 2. The local statistics of the gradient distributions (within the boxes) are shown in the third and fourth rows.

Fig. 2 (column 1) shows that for the natural image, a long-tailed distribution exists in the overall image as well as local regions. Fig. 2 (column 2) shows that for the image with global motion blur, the gradient distribution for the x-direction has much of its mass about zero. This is because the gradients along the x-direction are blurred by the motion. The gradient distribution in the y-direction, however, is relatively unaffected and exhibits a much wider distribution. This effect is also exhibited in the local regions in box 1 and box 2 in Figs. 2h and 2k. Fig. 2 (column 3) shows that the image of the motion blurred object has a mixture of distributions. For example, box 1 (Fig. 2h), which is selected from the moving object, shows characteristics of motion blur in its gradient distributions, while the region in box 2 (Fig. 2l), from the static backgrounds, has wider distributions in the x and y-directions.

To estimate the motion direction about a pixel  $x$ , we compute the local gradient distributions within a  $19 \times 19$  window along eight different radial directions:  $0, \pi/8, 2\pi/8, \dots, 6\pi/8, 7\pi/8$ . For each of these eight directions, we parameterize the distribution by fitting it to a Laplacian and Gaussian mixture defined as

$$\Phi_D(x) = \pi_0 L(x; \mu_0, \sigma_0) + \pi_1 G(x; \mu_1, \sigma_1), \quad (3)$$

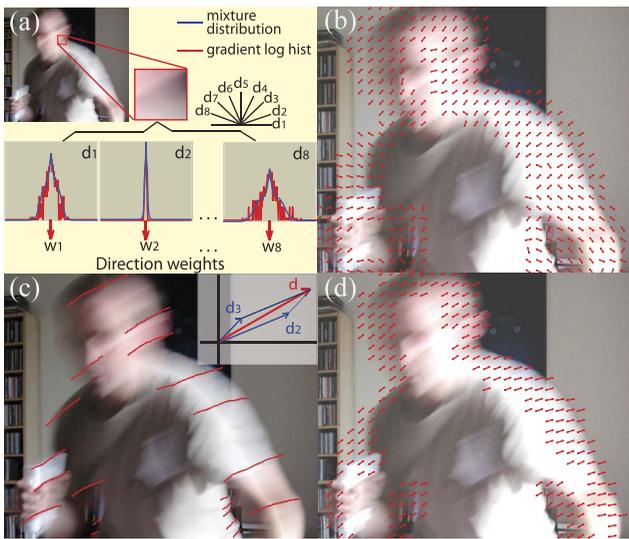


Fig. 3. This figure shows: (a) the process of estimating weights on eight directions about a pixel, (b) resulting estimated directions (direction with the largest weight is shown at each pixel), (c) example of user markup and direction decomposition, and (d) propagated directions from markup.

where  $\pi_0, \pi_1$  are the estimated weights of the two distributions,  $\mu_0 = \mu_1 = 0$  are the means of the distributions, and  $\sigma_0$  and  $\sigma_1$  are the shape parameters of the two distributions.

Each of the eight discrete directions will be assigned a weight,  $w_d$ , based on the area under the Laplacian. This process is shown in Fig. 3a. The idea is that directions with more gradients centered about 0 represent the underlying local motion (i.e., a larger area under the Laplacian). The exact computation of  $w_d$  is explained in Section 4 as it relates to our regularization. Note that the motion is not explicitly detected; instead, weights are assigned to each of the discrete motions. Therefore, regions with ambiguous motion (uniform blurring or homogenous texture) will have weights that favor no particular direction. For the needle maps shown in this paper to represent estimated motion, only the direction with the largest weight is drawn (e.g., Fig. 3b).

Local motion estimation is performed only for pixels in the definite foreground region. Pixels in the unknown region exhibit a mixture of the foreground and background motion and, hence, are not reliable. For unknown regions, the direction weights are propagated smoothly from the estimated motion based on (4), which will be described in the next section as it is also used to propagate directions from user-supplied markup.

Our method for computing the dominant direction of a motion blurred patch is based on the assumption that there is no dominant gradient direction inside the original unblurred patch. To test this assumption, we performed a simple analysis on the PASCAL VOC

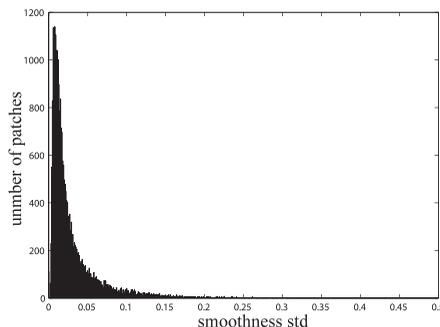


Fig. 4. This histogram plots the standard deviation of gradient directions computed from 20,000 patches from the PASCAL data set. The histogram peaks with a very low std imply that most unblurred patches have no dominant direction.

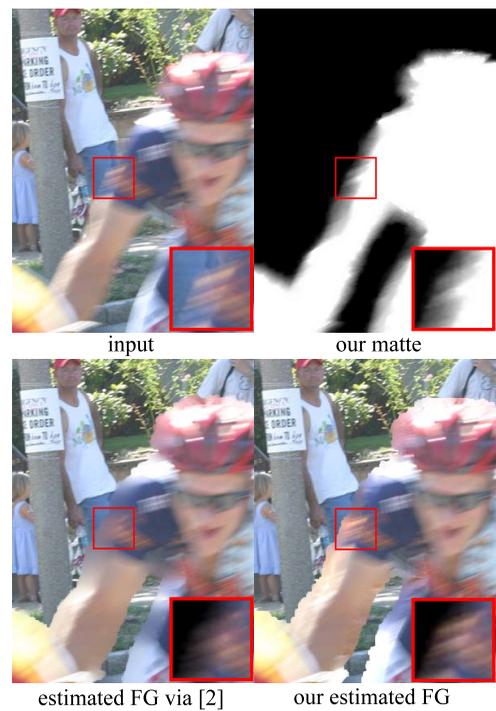


Fig. 5. A comparison of foreground estimation using closed-form matting [2] and our approach (described in Section 4.2). By adding the motion regularization into foreground estimation, we get a better foreground estimation as shown in the zoomed in composites. Note that the same matte is used to estimate both results.

2006 database which contains more than 2,000 images of different objects and scenes. We computed the standard deviation (std) of the directional weights  $w_d$  for 20,000 randomly selected patches (size  $19 \times 19$ ). Fig. 4 shows that the vast majority of patches exhibit a low std of the directional weights implying no dominate direction.

### 3.2 User-Supplied Local Motion

Since matting already requires user markup, another option for obtaining local motion is to have the user mark up the motion directly. Such markup is not difficult to perform as the blurred object typically has strong visual cues to the underlying motion.

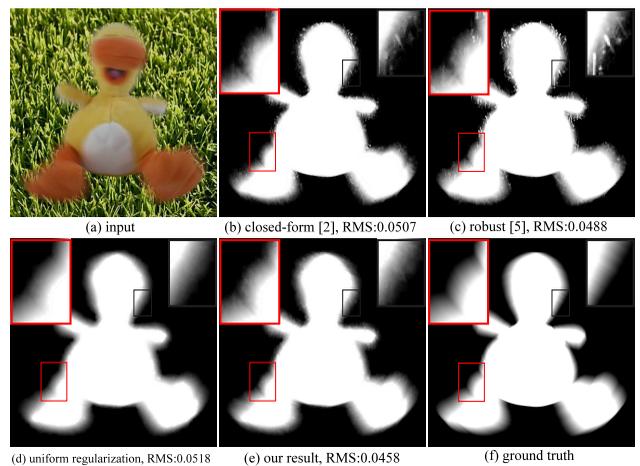


Fig. 6. This figure compares our results with other matting techniques on (a) a synthetic input (with known ground truth). Other techniques are (b) closed-form matting and (c) robust matting. (d) We also show the results of using simple uniform motion weights (i.e., all weights are equal). (e) Our result. (f) Ground-truth image. While the root mean square (RMS) error for our approach is only slightly better than other approaches, our results are the most visually similar to the ground truth.



Fig. 7. Comparison of results using estimated motion and user markup. Shown (left-to-right) are (first row) the original image, estimated direction using local gradient statistics, result using estimated directions; (second row) user-directional markup, propagated directions, result using user markup.

For this approach, the user draws scribbles on top of the image in the direction of the motion. Based on the user-provided directions, we obtain a set of sparse local motion directions along the scribbles. These sparse direction labels can be propagated to other unmarked pixels by solving the following equation:

$$\arg \min_{\vec{v}} \sum_{r \in S} (\vec{v}_r - \vec{v}_r^*)^2 + \sum_{r \in U} \sum_{s \in N(r)} (\vec{v}_r - \vec{v}_s)^2, \quad (4)$$

where  $\vec{v}_r$  are the local motion directions we want to estimate for each pixel  $r$  and  $\vec{v}_r^*$  are the sparse local motion directions obtained from user markup. The terms  $S$  and  $U$  are the scribble areas and the unknown areas, respectively, and  $N(r)$  is the first order neighborhood of a pixel  $r$ . Equation (4) can be solved using a sparse linear solver. We project the user-supplied motion to the two closest discrete directions as described in Section 3.1. To assign the weights, the user can either select a weight via a GUI (large, medium, or small) or we can use the length of the drawn stroke—longer strokes equal more weight. Similarly, if an image region does not contain any motion blur, the user can simply draw a “dot,” meaning that the regularization weight at that local region is zero. The direction weight is propagated in the same fashion as the directions using (4). Figs. 3c, 3d and 3e, 3f show an example of our estimated local motion direction and its regularization weight map.

#### 4 MOTION MATTING PROCEDURE

Here, we describe how to include the local motion information into conventional matting, in particular closed-form matting [2] and

robust matting [5]. Our approach assumes that the inputs are an image, a trimap, and local motions with regularization weights, either estimated by our gradient statistics technique or provided by the user. We also describe how to estimate the extracted foreground colors by incorporating motion information.

##### 4.1 Motion Regularization

In closed-form matting [2], for an  $N$  pixel natural image, the optimal matte is the one that minimizes the following energy:

$$E = \alpha^T \mathbf{L} \alpha + \lambda (\alpha - \tilde{\alpha})^T \mathbf{D} (\alpha - \tilde{\alpha}), \quad (5)$$

where  $\alpha$  is the solution of closed-form matting,  $\tilde{\alpha}$  is the vector containing user specified  $\alpha$  values for constrained pixels,  $\mathbf{D}$  is an  $N \times N$  diagonal matrix with its entries equal to 1 for constrained pixels and 0 for unconstrained pixels,  $\lambda$  is a large number to guarantee that  $\alpha$  is consistent with the constrained pixels. The term  $\mathbf{L}$  is the matting Laplacian matrix whose  $(i, j)$ th element is

$$\sum_{k|(i,j) \in w_k} \left( \delta_{ij} - \frac{1}{|w_k|} (1 + (I_i - \mu_k) (\Sigma_k + \frac{\varepsilon}{|w_k|} I_3)^{-1} (I_j - \mu_k)) \right), \quad (6)$$

where  $w_k$  represents the  $3 \times 3$  window which contains pixels  $i$  and  $j$ ,  $\mu_k$  and  $\Sigma_k$  are the color mean and variance in each window,  $I_3$  is a  $3 \times 3$  identical matrix, and  $\varepsilon$  is a regularization coefficient which is set to  $10^{-6}$  in our implementation. In robust matting [5], an additional data constraint is added, accounting for the alpha estimation with its confidence based on robust local sampling. The energy function is formulated as [6]

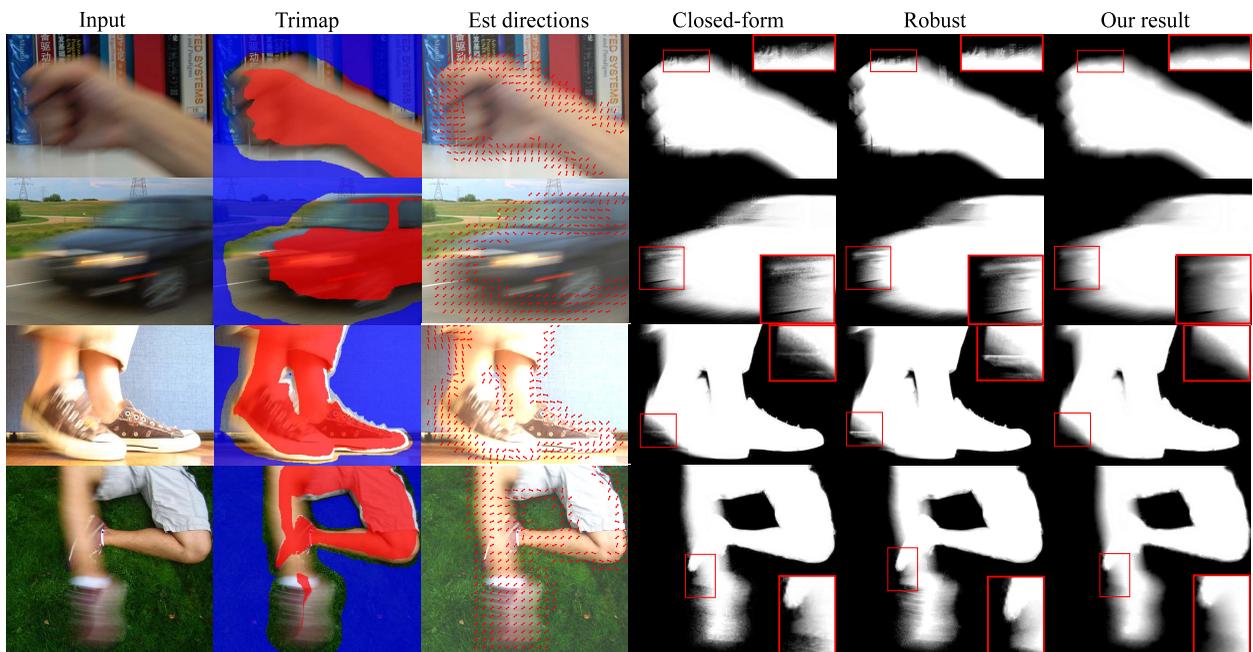


Fig. 8. Several examples comparing our results which are based on closed-form matting with closed-form matting [2] and robust matting [5]. Estimated directions are shown in column 3.

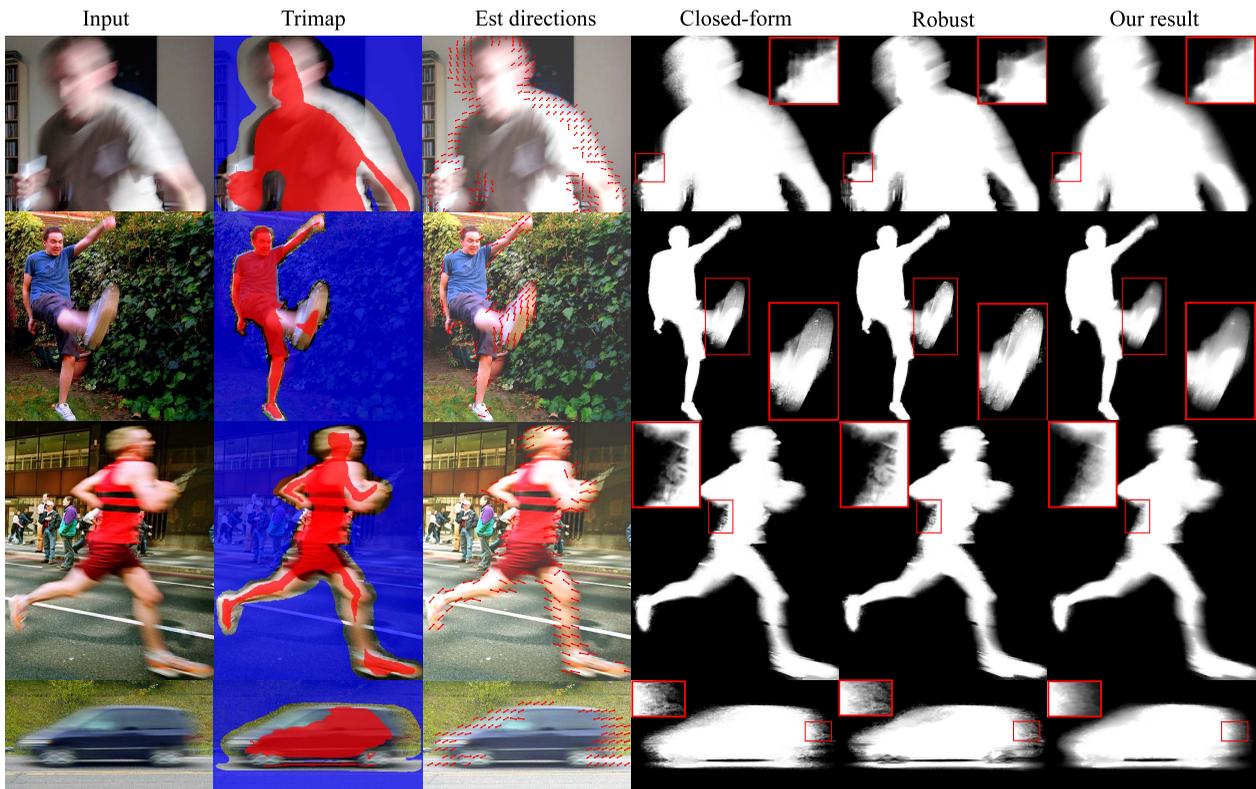


Fig. 9. Several examples comparing our results that incorporate regularization into robust matting with those of other techniques.

$$E = \sum_{i \in \psi} [\hat{f}_i(\alpha - \hat{\alpha}_i)^2 + (1 - \hat{f}_i)(\alpha - \delta(\hat{\alpha}_i > 0.5))^2] + \lambda_1 \cdot \alpha^T \mathbf{L} \alpha + \lambda_2 \cdot (\alpha - \hat{\alpha})^T \mathbf{D} (\alpha - \hat{\alpha}), \quad (7)$$

where  $\hat{\alpha}_i$  is the estimated alpha value at pixel  $i$  from sampling,  $\hat{f}_i$  is the corresponding confidence value, and  $\delta(\cdot)$  is a Boolean function which returns 0 or 1. By introducing the confidence values, reliable samples are favored while bad estimations, associated with low confidence values, are suppressed.

To include motion information into these matting techniques, we add the following regularization:

$$R_m(\alpha) = \sum_{d=1}^8 w_d \nabla_d \alpha^T \nabla_d \alpha, \quad (8)$$

where  $\nabla_d \alpha$  is the  $\alpha$ -gradient in direction  $d$ , and  $w_d$  is the weight of regularization for direction  $d$ . We set

$$w_d = \text{Area}(L_d) - \min_i (\text{Area}(L_i)), \quad (9)$$

where  $\text{Area}(L_d)$  is the area under the Laplacian curve (estimated in (3)) at direction  $d$  within the range  $[-0.05, 0.05]$ . The term  $\min_i (\text{Area}(L_i))$  is the minimal area among the eight Laplacians and is considered to correspond to the direction perpendicular to the motion direction. Thus,  $w_d$  corresponds to the strength of motion blur. Regularizing the alpha matte based on these  $w_d$  essentially suppresses the matte gradient according to local motion estimates. If an image region does not contain any motion blur, the term  $\text{Area}(L_d)$  will be similar in all eight directions. This will



Fig. 10. Compositing examples that paste images matted in previous figures onto new backgrounds. The first row shows the result using mattes from robust matting; the second row shows the result using our mattes for compositing. Our results are visual compelling and more plausible than the results from robust matting.

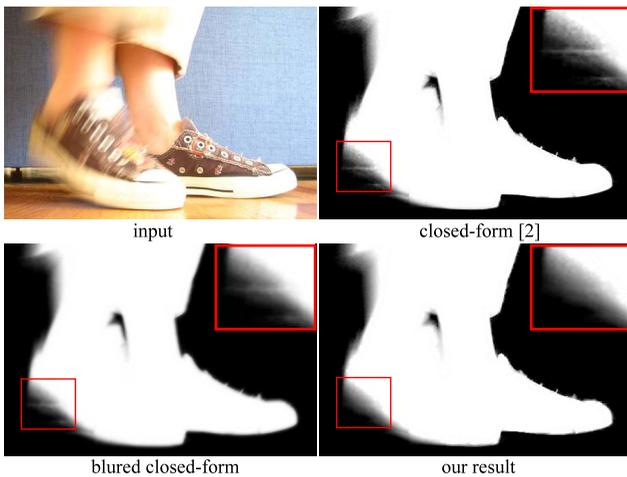


Fig. 11. This figure shows a comparison of our result and that obtained by closed-form matting that has been blurred using the estimated motion directions. Simple postprocessing of a conventional matte does not produce results similar to ours.

produce a result similar to conventional matting since  $w_d$  is small making the regularization term to have little effect on the estimated matte. For user-supplied motion, the regularization is identical. The only difference is that the user-supplied directions are decomposed into their two most dominant directions (as previously shown in Fig. 3c).

Combining our regularization term, the final matting energy function becomes

$$E_m = E + \gamma \cdot R_m(\alpha), \quad (10)$$

where  $E$  is the conventional energy function and  $\gamma = 1$  is a weighting factor. In our implementation, we removed the pixels from the linear system within the interior regions (measured by  $5 \times 5$  window) of definite foreground and definite background. The removed pixels are considered as hard constraint in the system. By removing these interior pixels, we significantly reduce the size of the linear system in (10).

## 4.2 Color Estimation

With the estimated  $\alpha$ , we can solve for  $F$  and  $B$  using least-squares minimization of the following energy:

$$E(F, B|\alpha) = \|\alpha F + (1 - \alpha)B - I\|^2 + \sum_{d=1}^8 w_d \nabla_d F^T \nabla_d F + \varpi_b \sum_{d=1}^8 \nabla_d B^T \nabla_d B, \quad (11)$$

where  $\|\alpha F + (1 - \alpha)B - I\|^2$  is a data term to minimize the estimation error according to matte compositing equation in [1],  $\sum_{d=1}^8 w_d \nabla_d F^T \nabla_d F$  and  $\sum_{d=1}^8 \nabla_d B^T \nabla_d B$  are regularization terms to enforce spatial smoothness of estimated color, and the term  $\varpi_b$  is a small number ( $\varpi_b = 0.01$ ) to enforce background color smoothness.



Fig. 13. This example shows a failure case where the motion direction in the blur region deviates the locally linear assumption.

Note that we use the same weighting scheme as in (9) for the estimation of  $F$ . Hence, our estimated foreground colors better reflect the estimated local motion directions while the approach in [2] produces oversmoothed foreground colors as demonstrated in Fig. 5.

## 5 RESULTS AND COMPARISONS

Experimental results on a variety of input images exhibiting various types of motion blur effects are shown in this section. If not specified, our results are obtained using the closed-form matting energy function. For comparisons with existing matting approaches, we chose closed-form matting [2] and robust matting [5] to serve as representative conventional matting techniques. For comparisons with [5], the primary author from [5] has vetted our parameters used to produce the results for robust matting.

### 5.1 Synthetic Example

Fig. 6 shows a synthetic example. Here, the motion blur is synthesized by rotating the object about the center and accumulating the results. We compare our result with those obtained by closed-form matting and robust matting, as well as the result obtained using uniform regularization (i.e.,  $w_d$  are the same for all directions). We can see that the result obtained by our approach more closely resembles the ground truth, while closed-form matting and robust matting produce unsatisfactory results. The result using uniform regularization produces an oversmoothed matte.

### 5.2 Real Examples

Figs. 8 and 9 show several examples of images containing motion blurred objects. Our results are compared with closed-form matting [2] and robust matting [5]. For our approach, the motions have been estimated by the technique described in Section 3.1. Some of the images contain both motion blurred regions and sharp regions. The transition of the extracted matte in the motion blurred regions to the sharp regions is smooth since the regularization weights are gradually decreased based on the amount of estimated motion blur. The trimaps are shown in the second column of the figures. Fig. 10 shows the comparisons of compositing results using our mattes and mattes from robust matting. The mattes from robust matting transfers structures from the original image that should not appear in the composited images. Fig. 14 shows more compositing results using our mattes on different backgrounds.

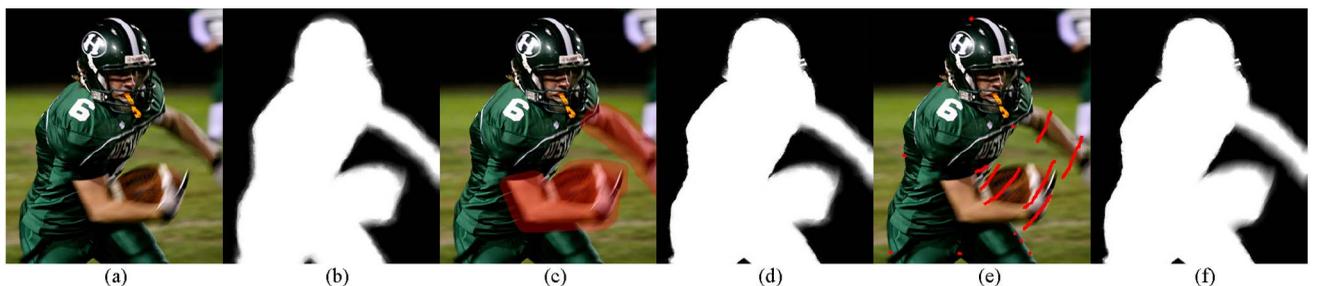


Fig. 12. In this example, the estimated motion is erroneous in nonblurred regions (a). This results in a slight blurring in the edges of the matte (b). By denoting the blurred region (c), the matte can be improved (d). For user-supplied markup (e), this could easily be avoided by ignoring these regions, producing a good matte shown in (f).

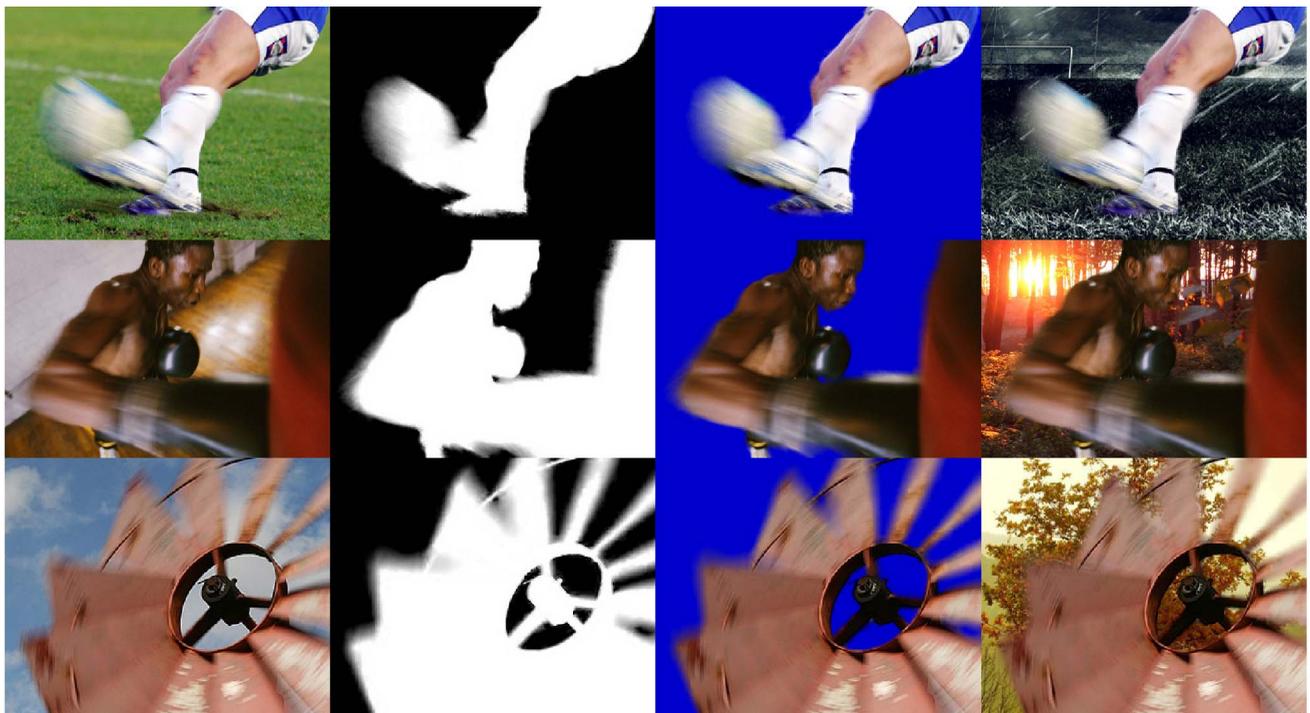


Fig. 14. Compositing examples, from left to right: input, computed matte, composite with solid background, composite on a new background.

### 5.3 Estimated Motion versus Motion Markup

Fig. 7 compares the mattes obtained using estimated motion versus user-supplied motion. We can see that the motion estimation results are better for user-supplied motion; however, the overall mattes are similar.

In some cases when automatic estimation is used and there is no blurring, our approach can produce erroneous errors due to strong local content posing as blur. Fig. 12 shows an example. In such cases, the user can simply mark up the region that is to be estimated as shown in Fig. 12c. In the case where user markup was performed, the user would have correctly marked up these regions (e.g., Fig. 12e), resulting in the matte shown in Fig. 12f.

### 5.4 Blurring a Conventional Matte

Our motion regularization is inherently a part of the alpha matte optimization; therefore, simply applying a blur to a conventional matte cannot obtain results of the same quality. Fig. 11 demonstrates this by comparing our results with those obtained by closed-form matting that have been blurred in the same direction as the estimated motion. This simple postprocessing blurring does not produce a matte similar to ours.

## 6 DISCUSSION AND SUMMARY

This paper has presented two contributions relevant to the task of matting motion blurred objects. First, we introduced a regularization term that was added to closed-form matting and robust matting to incorporate local motion constraints. Second, a method to estimate local motions by analyzing the local gradient statistics was presented. In addition, local motion estimation from simple user markup was also discussed.

As mentioned in Section 2, while there has been a great deal of work targeting matting of static objects, it is a bit surprising that there is no prior work explicitly targeting motion blurred objects. This is likely due to the fact that motion blurred objects are perceived as degraded and therefore not targeted by tasks such as cutting and pasting. However, motion blurred objects are interesting from a graphics design standpoint as they can be used to give a clear indication of energy (and not surprisedly motion). In

such situations, graphics arts often synthesize object motion blur using motion-blur filters. Our work allows them to avoid this step by facilitating cut and paste of blurred objects directly. Furthermore, in the case where the motion blur is considered an undesirable degraded artifact, matting of the blurred object is one of the first steps necessary for applying deblurring algorithms.

We note that our approach requires a reasonably tight trimap, similar to techniques like robust matting. We have found that first applying closed-form matting with no regularization and thresholding the results can provide a good initial trimap. As discussed in Section 3.1, our approach assumes that the original unblurred image patches have a uniform distribution of the image gradients. This can be violated if there is strong structure in the image content. Regions with strong image content, however, are the ones that are easiest to mark up by hand since image content under motion is easy to distinguish. This suggests that one strategy may be to combine automatic estimation with user corrections.

Another assumption of our approach is that the motion is locally linear. Our approach fails when this is violated as shown in Fig. 13. Part of future work will be to explore how more complex motions can be incorporated into our regularization scheme. In addition, we would like to explore if our regularization can also help to overcome situations where the object boundary is near highly saturated regions such as that shown in Fig. 9 (third row). Finally, while we demonstrated the ability to incorporate this regularization into closed-form matting and robust matting, our regularization approach should be applicable to other matting techniques.

## ACKNOWLEDGMENTS

This work was supported in part by FRC Grant R-252-000-423-112 and the Ministry of Culture, Sports, and Tourism (MCST) and Korea Content Agency (KOCCA) in the Culture Technology Research and Development Program 2010.

## REFERENCES

- [1] Y. Chuang, B. Curless, D.H. Salesin, and R. Szeliski, "A Bayesian Approach to Digital Matting," *Proc. IEEE CS Conf. Computer Vision and Pattern Recognition*, 2001.

- [2] A. Levin, D. Lischinski, and Y. Weiss, "A Closed-Form Solution to Natural Image Matting," *Proc. IEEE CS Conf. Computer Vision and Pattern Recognition*, 2006.
- [3] A. Levin, Y. Weiss, F. Durand, and W. Freeman, "Understanding and Evaluating Blind Deconvolution Algorithms," *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2009.
- [4] S. Dai and Y. Wu, "Motion from Blur," *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2008.
- [5] J. Wang and M. Cohen, "Optimized Color Sampling for Robust Matting," *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2007.
- [6] J. Wang and M. Cohen, "Image and Video Matting: A Survey," *Foundations and Trends in Computer Graphics and Vision*, vol. 3, no. 2, pp. 97-175, 2007.
- [7] J. Sun, J. Jia, C. Tang, and H. Shum, "Poisson Matting," *Proc. ACM SIGGRAPH*, 2004.
- [8] J. Wang, M. Agrawala, and M. Cohen, "Soft Scissors: An Interactive Tool for Realtime High Quality Matting," *Proc. ACM SIGGRAPH*, 2007.
- [9] C. Rhemann, C. Rother, P. Kohli, and M. Gelautz, "A Spatially Varying PSF-Based Prior for Alpha Matting," *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2010.
- [10] M. McGuire, W. Matusik, H. Pfister, J.F. Hughes, and F. Durand, "Defocus Video Matting," *Proc. ACM SIGGRAPH*, 2005.
- [11] N. Joshi, W. Matusik, and S. Avidan, "Natural Video Matting Using Camera Arrays," *Proc. ACM SIGGRAPH*, 2006.
- [12] J. Sun, Y. Li, S. Kang, and H. Shum, "Flash Matting," *Proc. ACM SIGGRAPH*, 2006.
- [13] A. Agrawal and Y. Xu, "Coded Exposure Deblurring: Optimized Codes for PSF Estimation and Invertibility," *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2009.
- [14] M. Ben-Ezra and S. Nayar, "Motion-Based Motion Deblurring," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 26, no. 6, pp. 689-698, June 2004.
- [15] Y.-W. Tai, H. Du, M.S. Brown, and S. Lin, "Image/Video Deblurring Using a Hybrid Camera," *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2008.
- [16] Y.-W. Tai, N. Kong, S. Lin, and S.Y. Shin, "Coded Exposure Imaging for Projective Motion Deblurring," *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2010.
- [17] Y.-W. Tai, H. Du, M.S. Brown, and S. Lin, "Correction of Spatially Varying Image and Video Motion Blur Using a Hybrid Camera," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 32, no. 6, pp. 1012-1028, June 2010.
- [18] Y.-W. Tai, P. Tan, and M.S. Brown, "Richardson-Lucy Deblurring for Scenes under Projective Motion Path," accepted.
- [19] J. Jia, "Single Image Motion Deblurring Using Transparency," *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2007.
- [20] Q. Shan, W. Xiong, and J. Jia, "Rotational Motion Deblurring of a Rigid Object from a Single Image," *Proc. IEEE Int'l Conf. Computer Vision*, 2007.
- [21] V. Caglioti and A. Giusti, "On the Apparent Transparency of a Motion Blurred Objects," *Int'l J. Computer Vision*, vol. 86, nos. 2/3, pp. 243 - 255, Jan. 2010.
- [22] A. Levin, "Blind Motion Deblurring Using Image Statistics," *Proc. Neural Information Processing Systems*, pp. 841-848, 2006.
- [23] R. Liu, Z. Li, and J. Jia, "Image Partial Blur Detection and Classification," *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2008.

► For more information on this or any other computing topic, please visit our Digital Library at [www.computer.org/publications/dlib](http://www.computer.org/publications/dlib).