Fractional Stereo Matching Using Expectation-Maximization

Wei Xiong, Hin Shun Chung, *Student Member*, *IEEE*, and Jiaya Jia, *Member*, *IEEE*

Abstract—In our fractional stereo matching problem, a foreground object with a fractional boundary is blended with a background scene using unknown transparencies. Due to the spatially varying disparities in different layers, one foreground pixel may be blended with different background pixels in stereo images, making the color constancy commonly assumed in traditional stereo matching not hold any more. To tackle this problem, in this paper, we introduce a probabilistic framework constraining the matching of pixel colors, disparities, and alpha values in different layers, and propose an automatic optimization method to solve a Maximizing a Posterior (MAP) problem using Expectation-Maximization (EM), given only a short-baseline stereo input image pair. Our method encodes the effect of background occlusion by layer blending without requiring a special detection process. The alpha computation process in our unified framework can be regarded as a new approach by natural image matting, which handles appropriately the situation when the background color is similar to that of the foreground object. We demonstrate the efficacy of our method by experimenting with challenging stereo images and making comparisons with state-of-the-art methods.

Index Terms—Stereo matching, digital matting, Expectation-Maximization, alpha matte.

1 Introduction

Stereo matching has long been an essential research topic in computer vision and has made rapid and significant progress in recent years [16], [21], [25]. Most conventional two-frame stereo matching approaches compute disparities and detect occlusions assuming that each pixel in the input image corresponds to a unique depth value.

However, this model has limitations in representing fractional-boundary objects since the color of each boundary pixel possibly blends with the background. Directly applying conventional stereo matching methods to computing the disparities for these pixels may produce erroneous results. One example is shown in Fig. 1, where the input images in Figs. 1a and 1e contain a fan in front of a background. Without considering color blending, applying the stereo matching method in [16] generates a problematic disparity result along the fan's boundary, as shown in Fig. 1b.

Recently, stereo matching methods have been developed that partially generalize the above assumption using transparency. Szeliski et al. [19] proposed solving the stereo matching problem using multiple input images, where the color and the transparency refinement are formulated as a nonlinear minimization problem. However, this method has difficulties in dealing with objects with thin and long hairs or with complex alpha values given a small number of input images. Later on, assuming a binary reflection map model, Tsin et al. [21] proposed estimating the front translucent and rear background layers using Graph Cuts. The pixel colors are estimated by iteratively reducing an energy function in a multiframe configuration. This method is not applicable to objects with general fractional boundaries. Both of these methods require multiple input images in order to estimate a satisfactory disparity map.

In this paper, given only a pair of short-baseline stereo images containing hairy objects in front of a background, we estimate alpha values, disparities, and pixel colors in a probabilistic framework and solve it using Expectation-Maximization (EM). Different from conventional stereo matching methods, which estimate only a single disparity for each pixel, our method assumes a dual-layer model (for the foreground and background) and establishes the color correspondences on the layers. The stereo constraints on different disparity layers [16] are used in our probabilistic framework to define likelihoods, whereas the sampling technique in natural image matting [2] provides useful information to define color and alpha priors. In the "fan" example shown in Fig. 1, we illustrate the disparity result by our method in Fig. 1c. A comparison of the disparities in the magnified local regions is illustrated in Figs. 1f and 1g.

Our method assumes that the fractional pixels in the input images are not primarily overlapped. This makes our algorithm a new natural image matting method with a stereopsis configuration. Fig. 1d shows the alpha matte result computed in our unified framework. Although the foreground and background colors are similar in many pixels, the hairy structures in the alpha matte are successfully preserved. A more detailed comparison with other natural image matting methods will be given later in this paper. We shall show that the stereopsis information greatly helps in producing visually satisfying matting results for many challenging examples.

- *The authors are with the Department of Computer Science and Engineering, The Chinese University of Hong Kong, Shatin, N.T. Hong Kong.*
  E-mail: wayne.xiong@hotmail.com, {hschung, leojia}@cse.cuhk.edu.hk.

Manuscript received 31 July 2007; revised 11 Jan. 2008; accepted 31 Mar. 2008; published online 11 Apr. 2008.
Recommended for acceptance by S.B. Kang.
For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number TPAMI-2007-07-0466.
Digital Object Identifier no. 10.1109/TPAMI.2008.98

0162-8828/09/$25.00 © 2009 IEEE Published by the IEEE Computer Society

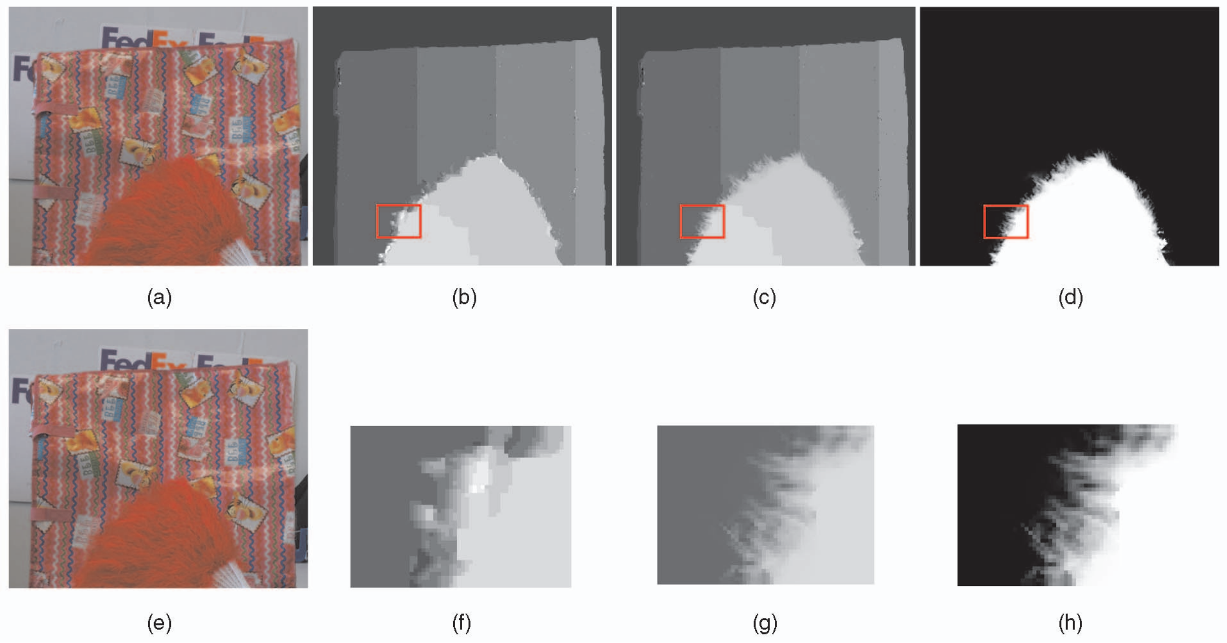


Fig. 1. A stereo image pair containing a hairy boundary object. (a) and (e) show the input stereo images containing a fan. The background texture and the fan include similar colors. (b) Stereo matching result in [16]. Because of color blending, the color constancy assumption is violated along the boundary of the fan, making the result problematic. (c) The stereo matching result obtained from our approach. The hairy structure is well preserved. (d) The computed alpha matte of the fan using our method. (f), (g), and (h) Magnified regions of the results.

Our method also contributes a nice implicit formulation of pixel occlusion. In conventional stereo matching, since each pixel has at most one disparity value, the occlusion is modeled separately for pixels having no correspondences [16]. In our approach, any pixel in the background layer behind the fractional-boundary object can be partially occluded, entirely occluded, or nonoccluded according to various degrees of transparency, which is naturally encoded using alpha values without special treatment.

The rest of the paper is organized as follows: Section 2 reviews previous work on stereo matching and digital image matting. We define our model and give notations in Section 3. The initialization of our system is depicted in Section 4. The detailed optimization process is described in Section 5. In Section 6, we illustrate and compare the experimental results. We conclude and discuss our paper in Section 7.

2 Related Work

We review dense stereo matching and digital image matting methods in this section.

2.1 Stereo Matching and Multiview Representation

Quite a few methods have been developed to solve the conventional stereo matching problem. A survey of two-frame stereo matching can be found in [13].

Markov Random Field (MRF) is widely used in stereo matching, where the corresponding Gibbs energy is defined considering labeling smoothness [7], [9], [16], [18]. Loopy Belief Propagation (LBP) [5] and Graph Cuts [1] are two popular methods for minimizing the Gibbs energy. Sun et al. [18] introduced a MAP estimation of disparity values. This method is further improved in [16] by adding the symmetry constraint and integrating color segmentation and plane fitting in the problem formulation. The disparity estimation is improved for the occluded pixels that are separately detected. In [25], a hierarchical LBP algorithm was proposed to refine the results on occluded and low-texture regions. Zhang and Seitz [26] proposed estimating both the disparity map and the corresponding MRF parameters in order to achieve an optimal parameter configuration in the EM framework. In [4], the input images are segmented into small patches to handle occlusions. All of these methods aim at improving the disparity estimation, assuming opaque or solid object boundary. Obviously, they cannot achieve similarly optimal disparity estimation for objects with hairy boundaries. The explicit occlusion detection and modeling also have no ability to handle partially occluded pixels.

Szeliski and Golland [19] first proposed solving the stereo matching problem with boundary opacity or blending. Visibility is computed through reprojection using multiple images, where color and transparency refinement is formulated as a nonlinear minimization problem. This method is insufficient to handle objects containing thin and long hairs given only two input images. Wexler et al. [23] computed alpha mattes and estimate layers from multiple images with known background information. The foreground objects are assumed to be planar so that they can be easily registered. However, in our problem definition, these conditions cannot be satisfied. In [21], the depth is estimated by considering layer overlapping. Our problem definition uses a nested plane sweep with refinement by Graph Cuts. The attenuation factors for color blending at reflecting areas are assumed to be constant values.

The objectives of the methods proposed in [3], [6], and [28] are basically to synthesize visually natural novel views using input images. One important step is boundary refinement. In [28], boundary matting on pixels with depth discontinuities is performed after the initial disparity computation. However, disparity errors due to color blending are not corrected in this method. Hasinoff et al. [6] computed a 3D boundary curve using multiple input images. Superresolution matting on a boundary can be achieved. This method requires more than two images to gather the necessary information, and it cannot be used to solve our problem with a different image configuration. Criminisi and Blake [3] proposed synthesizing a virtual view with visually pleasing object boundaries from two input images. The artifact patch is detected in the initial virtual view, and a split-patch-search method is used to find foreground and background registered patches. These patches are used to replace the artifact ones in the virtual view. This method also does not refine disparity since the objective is to make the novel view look natural. Zitnick et al. [27] computed the alpha contribution in the overlapped pixels among segments. The incorporation of segmentation and matting makes the optical flow estimation more accurate.

2.2 Digital Image Matting

Natural image matting estimates the foreground color, the background color, and the alpha value from each color-blended pixel given a natural input image. Using a *trimap*, which contains "foreground," "background," and "unknown" regions, Bayesian matting [2] and Poisson matting [15] estimate the foreground and background colors by collecting samples. Wang and Cohen [22] introduced an optimization process based on Belief Propagation (BP) to estimate the alpha matte without trimaps. In [10], Levin et al. proposed a closed form solution to solve the matting problem given the user input of a few strokes. This method is further improved in [11], where spectral analysis is performed to separate the image into components. Unsupervised matting results can be obtained by selecting the component group with the smallest matting cost.

There has been research on multiimage matting. To enhance the performance of video matting, Joshi et al. [8] used an autofocus system to first determine the pixel correlation among multiple images. Based on the variation of the related pixels, the researchers formed trimap and compute alpha mattes in real time. Sun et al. [17] used a pair of flash/no-flash images to extract mattes. All these methods cannot be directly applied to stereo matching without considering the correspondence of colors and alpha values in the input images.

3 Model Definition

In this paper, we denote the reference image by C^r and the matching image by C^m. They are taken from different viewing positions and are assumed rectified [20]. Conventionally, for a pixel (x, y) in C^r and its corresponding pixel (x', y') in C^m with disparity d, we have

$$x' = x + d, y' = y. \tag{1}$$

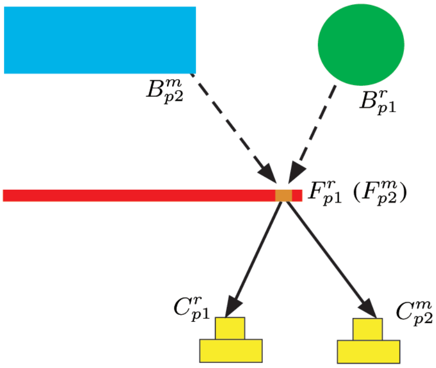


Fig. 2. Illustration of layers. The red bar represents the foreground object. Two light rays pass through the same foreground boundary point. In this case, the background points are different in the two images because of the depth differences.

Assuming Lambertian reflectance, the stereo matching problem is estimating disparity d using color similarity of the matched pixels in the two images:

$$C^r(x, y) = C^m(x + d, y). \tag{2}$$

In our problem definition, to model the color blending, we assume that each input image contains a foreground object F in front of a background scene B, both having Lambertian reflectance. The pixels in the background can be nonoccluded, partially occluded, or entirely occluded by the foreground object. According to the standard formulation of alpha blending, the mixed color in each pixel is expressed as

$$C^k(x, y) = \alpha^k(x, y)F^k(x, y) + \left[1 - \alpha^k(x, y)\right]B^k(x, y), \tag{3}$$

where $k \in \{r, m\}$. In our stereo model, instead of defining a single disparity d for each input image pixel, we introduce disparities d^f and d^b for latent pixels in foreground F and background B, respectively. This definition largely increases the flexibility of our method to model occlusions. Hence, for each latent foreground pixel $F^r(x, y)$ (or the background pixel $B^r(x, y)$) in C^r, we can obtain a matched pixel $F^m(x, y)$ (or $B^m(x, y)$) in C^m with the help of d^f (or d^b), where

$$\begin{aligned} F^r(x, y) &= F^m(x + d^f, y), \\ B^r(x, y) &= B^m(x + d^b, y). \end{aligned} \tag{4}$$

Moreover, the occlusion of the background scene can be nicely formulated using (3) according to the corresponding alpha values without requiring another explicit occlusion detection process:

$$\begin{cases} \alpha^k(x, y) = 1 & B^k(x, y) \text{ is entirely occluded,} \\ 0 < \alpha^k(x, y) < 1 & B^k(x, y) \text{ is partially occluded,} \\ \alpha^k(x, y) = 0 & B^k(x, y) \text{ is not occluded,} \end{cases} \tag{5}$$

where $k \in \{r, m\}$. Using a short-baseline camera setup, we assume that the matched latent foreground pixels should have similar transparencies in the stereo images. One illustration is given in Fig. 2. The red bar represents the foreground object. If the object boundary transparency is viewing-direction independent, the amount of background colors that can be seen through the same foreground point

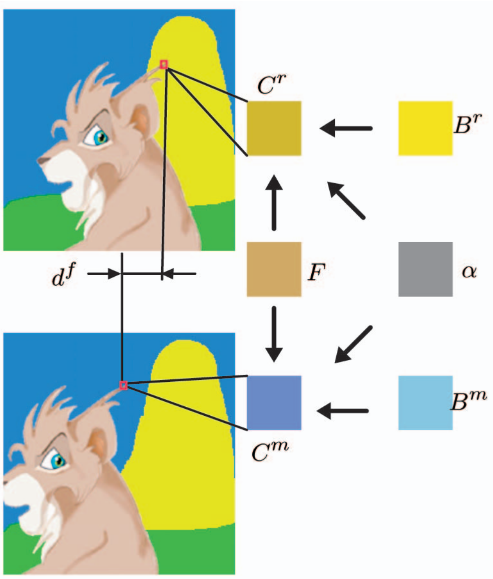


Fig. 3. Color constancy in blended pixels. Given the input stereo image pair, the semitransparent pixel colors C^r and C^m in the lion's mane are a mixture of foreground and background colors. Since C^r and C^m are matched in the foreground layer with disparity d^f, they are assumed to have similar foreground colors and alpha values. However, the corresponding background pixels are different, as shown in B^r and B^m.

should be similar from different views. In Fig. 2, two light rays hit two camera sensors in C^r_{p1} and C^m_{p2}, respectively. Since the two rays pass through the same foreground boundary point, the foreground pixel colors F^r_{p1} and F^m_{p2} and transparencies α^r_{p1} and α^m_{p2} in the two images can be assumed to be similar. In our method, we do not impose a hard constraint that corresponding foreground pixels have exactly the same alpha value. Instead, we introduce soft constraints by adopting distinct alpha variables in input images, allowing possible disagreement of alpha values due to noise or other factors. Specifically, if a foreground pixel (x, y) in C^r is matched to $(x + d^f, y)$ in C^m, we have

$$\alpha^r(x, y) \approx \alpha^m(x + d^f, y). \qquad (6)$$

This constraint will be used in our model and will be discussed in Section 5.2. It is validated by our experimental results shown in Section 6.

In the rest of the paper, for simplicity, we use subscripts p, $p + f$, and $p + b$ to denote pixels with coordinates (x, y) in image r and coordinates $(x + d^f, y)$ and $(x + d^b, y)$ in image m, respectively. Substituting (2) and (4) into (3), we obtain the following equations for each corresponding pixel pair in the input images:

$$\begin{cases} C^r_p &= \alpha^r_p F^r_p + \left(1 - \alpha^r_p\right) B^r_p, \\ C^m_{p+f} &= \alpha^m_{p+f} F^m_{p+f} + \left(1 - \alpha^m_{p+f}\right) B^m_{p+f}. \end{cases} \qquad (7)$$

We show an example in Fig. 3, where two corresponding foreground pixels are blended with different background pixels due to the disparity differences. In (7), there are unknowns F^r, F^m, B^r, B^m, α^r, and α^m to be estimated given input C^r and C^m. F^r and F^m are corresponding foreground pixels. We optimize F^r in our method. Then, F^m can be readily obtained by mapping the foreground pixels in C^r to C^m using the computed disparities. We estimate α^r, α^m, B^r, and B^m separately. It guarantees that the unmatched background pixels due to occlusion are appropriately handled, which in turn improves the estimation of disparities and foreground colors.

In what follows, without special annotation, we use F to denote F^r. Thus, by taking (4) into (7), C^m_{p+f} can be rewritten as

$$\begin{aligned} C^m_{p+f} &= \alpha^m_{p+f} F^m_{p+f} + \left(1 - \alpha^m_{p+f}\right) B^m_{p+f} \\ &= \alpha^m_{p+f} F_p + \left(1 - \alpha^m_{p+f}\right) B^m_{p+f}. \end{aligned} \qquad (8)$$

4 System Initialization

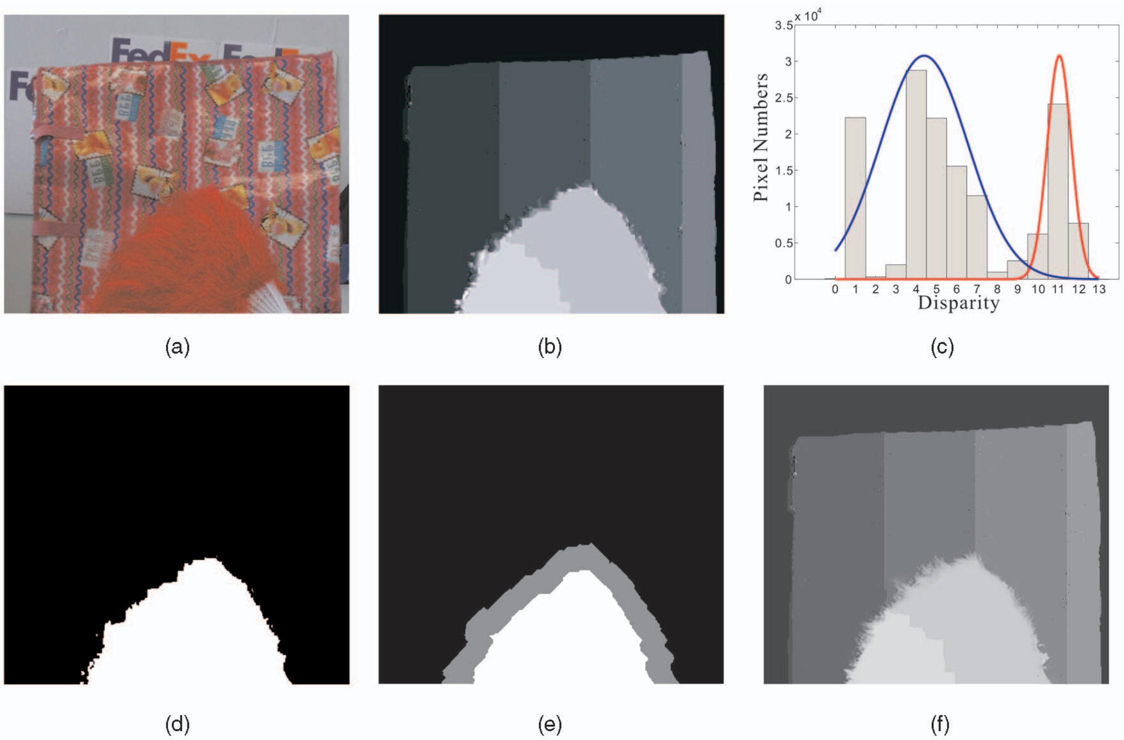
We first initialize all unknown variables. The detailed disparity and alpha matte estimation process is described in the following sections. Note that our algorithm is not restricted to only using the following alpha and disparity initialization. More sophisticated methods, such as the one described in [24], can also be employed in order to handle thin and complex structures.

Initializing disparity. We initialize a single disparity d_p for each pixel p in image C^r and C^m using the stereo matching method [16]. The disparity errors produced in this step are inevitable due to the lack of consideration of color blending. Since our method estimates two disparity values for each pixel representing the foreground and background, respectively, we take the following steps to initialize the values. We first compute the disparity histogram. By assuming that there is a depth gap between the background and the foreground objects, we propose to partition the histogram into two disjointed segments. In particular, we first fit the histogram by a two-component Gaussian mixture model. The parameters of the two Gaussians for the foreground and background are denoted by $\{\overline{d^f}, \sigma_{d^f}\}$ and $\{\overline{d^b}, \sigma_{d^b}\}$, respectively. They will also be used later to define likelihoods for foreground and background disparities. One example of the constructed histogram is shown in Fig. 4c. Then, we use the Bayes classifier to partition the histogram and assign disparity d to the foreground if $N(h(d); \overline{d^f}, \sigma_{d^f}) \geq N(h(d); \overline{d^b}, \sigma_{d^b})$, where $h(d)$ denotes the value of the dth bin in the histogram. Otherwise, d is assigned to the background.

Depending on whether setting d is d^f or d^b, we set the value of the remaining disparity as d^b_p or d^f_p as the mean of background disparities $\overline{d^b}$ or the mean of foreground disparities $\overline{d^f}$, respectively, to complete the disparity initialization.

Initializing alpha matte. In each image, we generate a trimap for alpha matte estimation. The trimap indicates whether one pixel in the input images is definitely in the foreground ($\alpha = 1$), definitely in the background ($\alpha = 0$), or unknown. Our method estimates the disparities and the alpha values within the "unknown" region in the following optimization steps.

The trimap construction is described as follows: We have produced a binary segmentation in input images according



(a) (b) (c)

(d) (e) (f)

Fig. 4. Workflow illustration. (a) The input reference image. (b) The initialized disparity map computed by [16]. (c) The initial disparity histogram. The two fitted Gaussians are also shown. (d) Binary segmentation on disparity according to the binary classifier. (e) Initial trimap created by dilating the segmentation boundary by 15 pixels. (f) The final disparity map computed by our method.

to whether $d_p = d_p^f$ or $d_p = d_p^b$, as shown in Fig. 4d. The disparity values around the segmentation boundaries are unreliable since these pixel colors are most likely mixtures of the foreground and background. We then select all these boundary pixels and dilate them by 2 to 15 pixels to form the "unknown" region. The range of dilation is determined using the following considerations. If the foreground object has long hairs, the "unknown" region should be wide enough to include all boundary pixels. Otherwise, for a near-solid boundary object, we need to dilate only one or two pixels. Note that a narrower "unknown" region contains fewer pixels, making the overall computation faster and easier.

All other pixels are then marked as "definitely foreground" ($\alpha = 1$) or "definitely background" ($\alpha = 0$) according to their initial disparities. The initial trimaps on C^r and C^m are thus created. One illustration is shown in Fig. 4e.

The initial foreground color $F^{(0)}$ and background color $B^{(0)}$ for the pixels in the "unknown" region are all directly set as the pixel color C. The alpha matte $\alpha^{(0)}$ is initialized as

$$\alpha_p^{(0)} = \begin{cases} 1 & p \text{ is in the ``foreground'' region,} \\ 0.5 & p \text{ is in the ``unknown'' region,} \\ 0 & p \text{ is in the ``background'' region.} \end{cases} \quad (9)$$

The values of F, B, and α will be refined in our following optimization.

5 Optimization

In this section, we describe our algorithm for solving the fractional stereo matching problem. Given the observation $U = \{C^r, C^m\}$, we separate the unknowns into a parameter set $\Theta = \{F, B^r, B^m, \alpha^r, \alpha^m\}$ and hidden data $J = \{d^f, d^b\}$. In this section, we aim at estimating the parameters using EM:

$$\begin{aligned} \Theta^* &= \arg\max_{\Theta} \log P(U, \Theta) \\ &= \arg\max_{\Theta} \log \sum_{J \in \psi} P(U, J, \Theta), \end{aligned} \quad (10)$$

where ψ is the space containing all J of size n. After we obtain the optimized parameter set Θ^*, we shall further refine d^f and d^b using BP.

5.1 Expectation Step

In the expectation step, we assign the probabilities at each pixel to the depths of the foreground and background layers (the "hidden variables"), given images that represent both layers and the alpha matte in the reference view (the "parameter set"). The expectation of the depth random variables is then computed.

In iteration $n + 1$, we compute the expectation of $P_p(d^f = d_1, d^b = d_2 | \Theta^{(n)}, U)$ for each pixel p given the estimated $\Theta^{(n)}$, where $d_1, d_2 \in \{0, 1, \dots, L_d\}$. L_d is the maximum disparity level. Since d^f and d^b are statistically independent, we have

$$\begin{aligned} & E\Big(P_p\Big(d^f = d_1, d^b = d_2 \mid \Theta^{(n)}, U\Big)\Big) \\ &= E\Big(P_p\Big(d^f = d_1 \mid \Theta^{(n)}, U\Big)\, P_p\Big(d^b = d_2 \mid \Theta^{(n)}, U\Big)\Big) \\ &= E\Big(P_p\Big(d^f = d_1 \mid \Theta^{(n)}, U\Big)\Big)\, E\Big(P_p\Big(d^b = d_2 \mid \Theta^{(n)}, U\Big)\Big). \end{aligned} \quad (11)$$

In what follows, we describe the expectation computations for foreground depth d^f and background depth d^b.

5.1.1 Computing Expected Foreground Depth

Using Bayes' theorem, we have

$$\begin{aligned} &P_p\Big(d^f \mid \Theta^{(n)}, U\Big) \\ &\propto P_p\Big(d^f | U, B^{r(n)}, B^{m(n)}, \alpha^{r(n)}, \alpha^{m(n)}\Big) \\ &\propto P_p\Big(B^{r(n)}, B^{m(n)}, \alpha^{r(n)}, \alpha^{m(n)} \mid d^f, U\Big) \cdot P_p\big(d^f | U\big), \end{aligned} \quad (12)$$

where $P_p(\cdot | d^f, U)$ represents the likelihood given d^f, and $P_p(d^f|U)$ represents the prior of foreground disparity d^f. In the following, we describe their specific constructions.

Using (7) and (8), ideally, the corresponding foreground pixels in the two images should have the same foreground color:

$$\begin{aligned} \alpha^r_p F_p &= C^r_p - \Big(1 - \alpha^r_p\Big) B^r_p, \\ \alpha^m_{p+f} F_p &= C^m_{p+f} - \Big(1 - \alpha^m_{p+f}\Big) B^m_{p+f}. \end{aligned}$$

Therefore, the following equation should hold for the matched latent foreground pixels:

$$\alpha^m_{p+f}\Big(C^r_p - \Big(1 - \alpha^r_p\Big) B^r_p\Big) = \alpha^r_p\Big(C^m_{p+f} - \Big(1 - \alpha^m_{p+f}\Big) B^m_{p+f}\Big). \quad (13)$$

Using (13), we define probability

$$P_p(B^{r(n)}, B^{m(n)}, \alpha^{r(n)}, \alpha^{m(n)} | d^f, U)$$

as the color similarity of the corresponding foreground pixels in the input images:

$$P_p\Big(B^{r(n)}, B^{m(n)}, \alpha^{r(n)}, \alpha^{m(n)} | d^f, U\Big) = N\big(\|\kappa\|; 0, \sigma_c^2\big), \quad (14)$$

where

$$\begin{aligned} \kappa = {} & \alpha^{m(n)}_{p+f}\Big(C^r_p - \Big(1 - \alpha^{r(n)}_p\Big) B^{r(n)}_p\Big) \\ & - \alpha^{r(n)}_p\Big(C^m_{p+f} - \Big(1 - \alpha^{m(n)}_{p+f}\Big) B^{m(n)}_{p+f}\Big), \end{aligned}$$

and $N(\cdot; 0, \sigma_c^2)$ denotes a Gaussian distribution with mean 0 and variance σ_c^2, modeling possible errors and noise. σ_c is set to 10 in our experiments.

Prior $P_p(d^f|U)$ is defined using the information obtained from our initialization step. Since we have fitted the initial disparity values by two Gaussian distributions, we directly use the foreground disparity model to define $P_p(d^f|U)$:

$$P_p(d^f|U) = N\Big(d^f; \overline{d^f}, \sigma_f^2\Big), \quad (15)$$

where $\overline{d^f}$ and σ_f^2 denote the mean and the variance of the foreground disparity Gaussian distribution estimated in Section 4.

Combining (14) and (15), the expectation of the disparity variable for each foreground pixel p can be written as

$$E\Big(P_p\Big(d^f = d_1 \mid \Theta^{(n)}, U\Big)\Big) = \frac{P_p\big(d^f = d_1 \mid \Theta^{(n)}, U\big)}{\sum\limits_{d_i} P_p(d^f = d_i | \Theta^{(n)}, U)}. \quad (16)$$

Because there are only a few discrete levels for d_i, estimating $E\big(P_p(d^f = d_1 | \Theta^{(n)}, U)\big)$ is easy.

5.1.2 Computing Expected Background Depth

For the background disparity d^b, we similarly define the conditional probability as

$$\begin{aligned} &P_p\Big(d^b \mid \Theta^{(n)}, U\Big) \\ &\propto P_p\Big(d^b | U, B^{r(n)}, B^{m(n)}, \alpha^{r(n)}, \alpha^{m(n)}\Big) \\ &\propto P_p\Big(B^{r(n)}, B^{m(n)}, \alpha^{r(n)}, \alpha^{m(n)} | d^b, U\Big) \cdot P_p(d^b|U), \end{aligned} \quad (17)$$

where the matching probability

$$P_p(B^{r(n)}, B^{m(n)}, \alpha^{r(n)}, \alpha^{m(n)} | d^b, U)$$

represents the likelihood in terms of background color similarity given the background disparity, and $P_p(d^b|U)$ represents the background disparity prior.

Unlike foreground pixels, background pixels can be possibly occluded; therefore, the likelihood $P_p(\cdot | d^b, U)$ should be defined differently from the foreground counterpart in (14). We consider all possible occlusion situations and summarize them in the following cases:

- If both $B^{r(n)}$ and $B^{m(n)}$ are not primarily occluded, i.e., the corresponding $\alpha^{r(n)} \approx 0$ and $\alpha^{m(n)} \approx 0$, the background color similarity $\|B^{r(n)}_p - B^{m(n)}_{p+b}\|$ basically measures the confidence of pixel matching. The smaller the color difference, the more likely the background pixels match.
- If $B^{r(n)}$ in the reference image is partially occluded, i.e., $\alpha^{r(n)} \neq 0$, when computing $B^{m(n)}$ to match $B^{r(n)}$ using color similarly $\|B^{r(n)}_p - B^{m(n)}_{p+b}\|$, there is always uncertainty because of the occlusion. In this case, we do not completely trust the matching result for $B^{r(n)}$, no matter what matching color $B^{m(n)}$ is.
- Otherwise, if $B^{m(n)}$ in the matching image is partially occluded, i.e., $\alpha^{m(n)} \neq 0$, the penalty of mismatching should be introduced in order to prevent $B^{r(n)}$ from always matching the occluded pixels.

We now define the likelihood probability of the background color adapting to the alpha values, seamlessly integrating the above three cases:

$$P_p\Big(B^{r(n)}, B^{m(n)}, \alpha^{r(n)}, \alpha^{m(n)} | d^b, U\Big) = N\big(\rho; 0, \sigma_b^2\big), \quad (18)$$

where

$$\rho = \Big(1 - \alpha^{r(n)}_p\Big)\Big[\Big(1 - \alpha^{m(n)}_{p+b}\Big)\Big\|B^{r(n)}_p - B^{m(n)}_{p+b}\Big\|^2 + \alpha^{m(n)}_{p+b}\tau\Big],$$

and $N(\cdot; 0, \sigma_b^2)$ denotes the Gaussian distribution similar to that in (14). σ_b is set to 10 in our experiments. α^m_{p+b} in C^m is the alpha value corresponding to α^r_p in C^r for the same background pixel, and τ is a constant value (set to 150) for penalizing the case that the value of α^m_{p+b} is close to 1, that is, the background pixel $p + b$ (with coordinate $(x + d^b, y)$) is largely occluded in image C^m.

To analyze, in (18), if both $\alpha^{r(n)}$ and $\alpha^{m(n)}$ are close to 0, $\|B_p^{r(n)} - B_{p+b}^{m(n)}\|^2$ basically measures the matching cost. Otherwise, when α_p^r or α_{p+b}^m is close to 1, the background is almost or entirely occluded by the foreground object. Then, the above simple measurement using background color correspondence is untrustworthy. Therefore, we introduce the adaptive weights $(1 - \alpha_p^r)$ and $(1 - \alpha_{p+b}^m)$ to reduce possible errors in background color matching.

When computing the corresponding background matching pixel $p + b$ in C^m for each pixel p in C^r, a matching term without including the penalty τ will make the matching pixel in C^m whose alpha value is close to 1 much preferred since the matching cost will approach zero. This obviously biases the background matching process to favoring only the largely occluded pixels in order to minimize the cost (which we call trivial matching). By adding penalty τ, we increase the trivial matching cost and make occluded background pixels less preferred in the matching process.

Prior $P_p(d^b|U)$ is defined in a way similar to (15) using the initial background disparity Gaussian distribution (described in Section 4):

$$P_p(d^b|U) = N\left(d^b; \overline{d^b}, \sigma_b^2\right) = \frac{1}{\sqrt{2}\sigma_b} \exp\left(-\frac{(d^b - \overline{d^b})^2}{2\sigma_b^2}\right). \quad (19)$$

Integrating the two probability definitions, the expectation of the background disparity variable for each pixel p can be computed by

$$E\left(P_p\left(d^b = d_2 \mid \Theta^{(n)}, U\right)\right) = \frac{P_p\left(d^b = d_2 \mid \Theta^{(n)}, U\right)}{\sum_{d_i} P_p(d^b = d_i|\Theta^{(n)}, U)}. \quad (20)$$

5.2 Maximization Step

After the expectation computation, we maximize the expected complete-data log-likelihood with respect to parameter set Θ given the observation U (i.e., the input stereo images):

$$\begin{aligned}\Theta^{(n+1)} &= \arg\max_{\Theta} \sum_{J \in \psi} P\left(J|\Theta^{(n)}, U\right) \log P(\Theta, J, U) \\ &= \arg\max_{\Theta} \sum_{J \in \psi} P\left(J|\Theta^{(n)}, U\right) \log\left(P(J, U|\Theta) P(\Theta)\right) \\ &= \arg\max_{\Theta} \sum_{J \in \psi} P\left(J|\Theta^{(n)}, U\right) \left(L(J, U|\Theta) + L(\Theta)\right),\end{aligned} \quad (21)$$

where $L(\cdot)$ is the log likelihood $L(\cdot) = logP(\cdot)$. $P(J|\Theta^{(n)}, U)$ has already been computed in the expectation step. The above definition is similar to that in [12].

Integrating (4), (7), and (8), we get

$$\begin{aligned}L(J, U|\Theta) = -\sum_{p \in C^r} \Bigg[&\left(\left\|\alpha_p^r F_p + \left(1 - \alpha_p^r\right) B_p^r - C_p^r\right\|^2 \right.\\ &\left. + \left\|\alpha_{p+f}^m F_p + \left(1 - \alpha_{p+f}^m\right) B_{p+f}^m - C_{p+f}^m\right\|^2\right) / 2\omega_C^2 \\ &+ \left\|B_p^r - B_{p+b}^m\right\|^2 / 2\omega_B^2 + \left\|\alpha_p^r - \alpha_{p+f}^m\right\|^2 / 2\omega_\alpha^2 \Bigg],\end{aligned} \quad (22)$$

where ω_C, ω_B, and ω_α denote the standard deviations of Gaussian probability distributions for C, B, and α. Their values are set to 15, 15 and in range [0.05, 0.1], respectively. The first two terms $\|\alpha_p^r F_p + (1 - \alpha_p^r) B_p^r - C_p^r\|$ and $\|\alpha_{p+f}^m F_p + (1 - \alpha_{p+f}^m) B_{p+f}^m - C_{p+f}^m\|$ define the color blending likelihood considering each foreground pixel F_p in the stereo images. $\|B_p^r - B_{p+b}^m\|$ defines the background likelihood. $\|\alpha_p^r - \alpha_{p+f}^m\|$ defines our alpha similarity.

$L(\Theta)$ is defined in a way similar to that in [2]. We expand it to

$$L(\Theta) \propto L(\alpha^r) + L(\alpha^m) + L(F) + L(B^r) + L(B^m). \quad (23)$$

We estimate the foreground color, alpha value, and background color likelihoods for each pixel using the color sampling method proposed in [2]. Specifically, we first collect foreground/background color samples for each unknown pixel from neighboring pixels in the "definitely foreground/background" regions of the trimap. Then, we model color distributions using these samples by fitting Gaussian distributions or Gaussian mixtures for the background and foreground, respectively. In what follows, we describe our method using a single Gaussian model for simplicity. The formulation and optimization using Gaussian mixtures are similar.

Using notations similar to those in [2], for each pixel p, we represent the constructed Gaussian models for the foreground color as $\{\overline{F_p}, \Sigma_{F_p}^{-1}\}$ and get

$$L(F) = \sum_p L(F_p) = \sum_p -(F_p - \overline{F_p})^T \Sigma_{F_p}^{-1} (F_p - \overline{F_p})/2. \quad (24)$$

Similarly, we define $L(B^r)$, $L(B^m)$, $L(\alpha^r)$, and $L(\alpha^m)$ as

$$\begin{aligned}L(B^k) &= \sum_p L\left(B_p^k\right) \\ &= \sum_p -\left(B_p^k - \overline{B_p^k}\right)^T \Sigma_{B_p^k}^{-1} \left(B_p^k - \overline{B_p^k}\right) \Big/ 2,\end{aligned} \quad (25)$$

$$L(\alpha^k) = \sum_p L\left(\alpha_p^k\right) = \sum_p -\frac{\left(\alpha_p^k - \overline{\alpha_p^k}\right)^2}{2\sigma_k^2}, \quad (26)$$

where $k \in \{r, m\}$. Given all the above definitions, solving (21) is equivalent to searching for the best Θ^* that minimizes $f(\Theta)$, where

$$f(\Theta) = -\sum_{J \in \psi} P\left(J|\Theta^{(n)}, U\right) \left(L(J, U|\Theta) + L(\Theta)\right). \quad (27)$$

One important observation here is that in minimizing f, each F, B, and α are correlated only with their counterparts along the same scan line in the stereo images. Therefore, we optimize them separately in the ground of the scan line. This increases the computation efficiency and improves the robustness of our method. Our optimization process in this maximization step includes iteratively estimating α and $\{F, B\}$. The optimization details are described in the Appendix. The whole EM algorithm is outlined in step 4 in Algorithm 1.

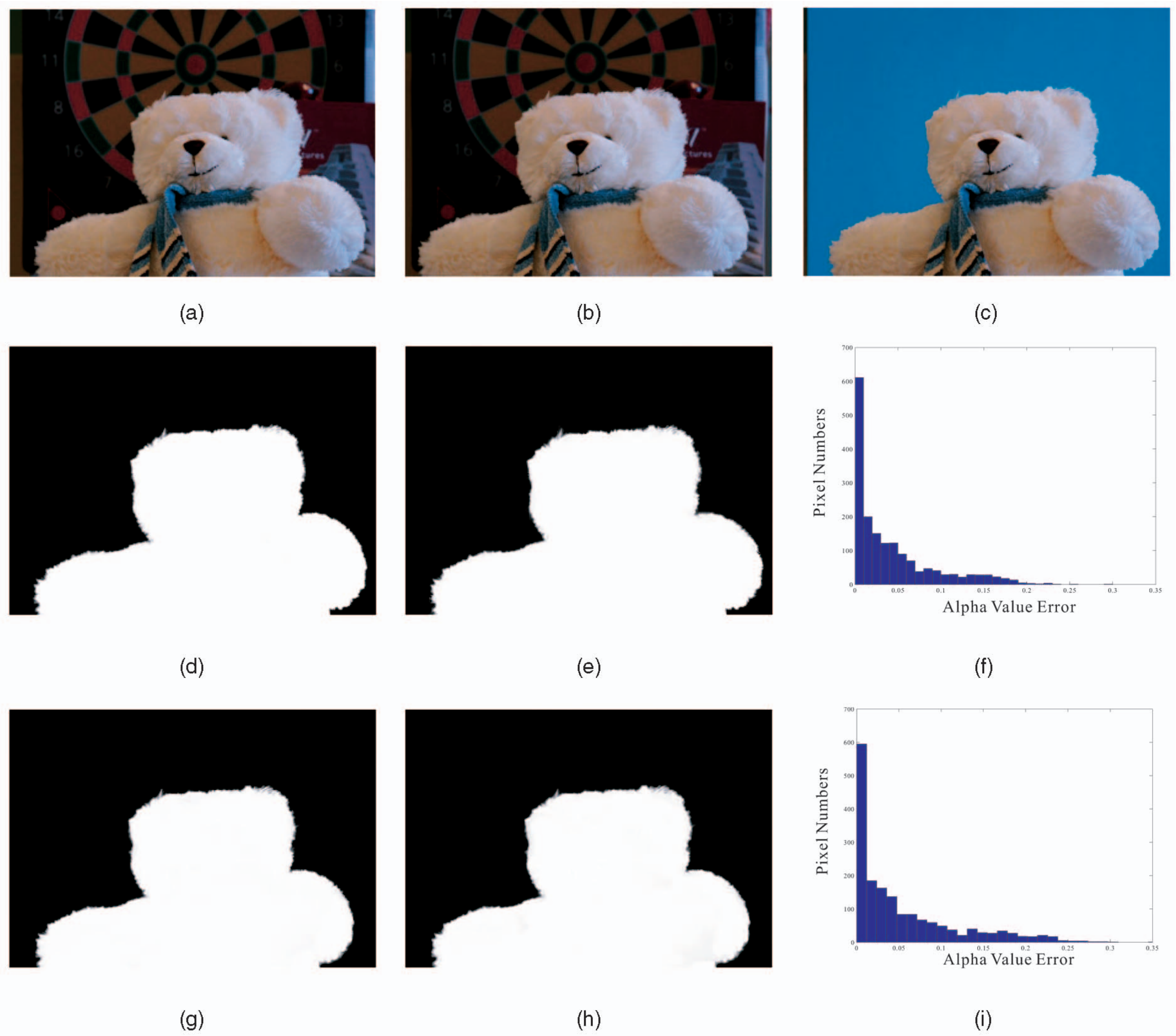


Fig. 5. The likelihood error decreases in iterations for the example shown in Fig. 1.

We plot in Fig. 5 the likelihood residual error e produced in iterations in the "fan" example shown in Fig. 4. Error e is computed by

$$e = -\sum_{J \in \psi} P\Big(J|\Theta^{(n)}, U\Big) \log \frac{P\big(\Theta^{(n+1)}|J, U\big)}{P(J|\Theta^{(n)}, U)}. \qquad (28)$$

The error monotonically decreases in iterations, and the optimization process converges rapidly.

5.3 Computing Final Disparities

The disparities are treated as hidden data in our EM framework since they have limited discrete values. As in many other stereo methods, incorporating pairwise smoothness in an MRF model makes disparities computation robust. However, it is computationally impractical to incorporate these pairwise-smoothness constraints in the EM steps. Therefore, we introduce a postprocessing step using BP to finally estimate the disparities.

In this step, we obtain a parameter set estimate Θ^* and a probability distribution of hidden data d^f and d^b. Similar to other stereo matching methods, we form a MRF on the images and define the energy, which contains a data term and a smoothness term [16] as

$$E(d^k|U, \Theta^*) = E_d(d^k|U, \Theta^*) + E_s(d^k), \qquad (29)$$

where $k \in \{f, b\}$. $E_s(d^k)$ is the smoothness term defined similar to that in [16]:

$$E_s(d^k) = \sum_{s,t \in \mathcal{N}(s)} \gamma_d \big|d_s^k - d_t^k\big|^2,$$

where $\mathcal{N}(\cdot)$ denotes the set of the pixel neighborhood, and γ_d is a weight.

$E_d(d^k|U, \Theta^*)$ is the data term:

$$E_d(d^k|U, \Theta^*) = \sum_p -\log P\Big(d_p^k = d^k \mid \Theta^*, U\Big). \qquad (30)$$

Here, data term $E_d(d^k|U, \Theta^*)$ is the output from our EM optimization. The data term faithfully represents data likelihood and can appropriately constrain our final disparity computation. We use BP to minimize the energy defined in (29) in order to compute the optimal disparities.

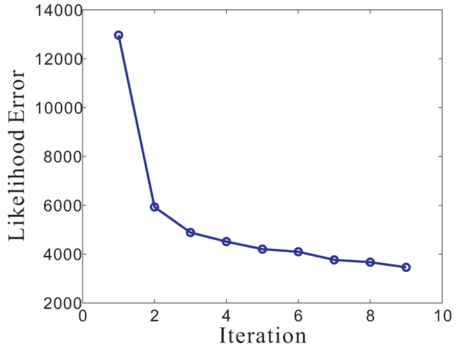


Fig. 6. Toy bear example. (a) and (b) The input "white bear" stereo images are shown. The plush toy is placed in front of a textured background scene. (c) Another image is captured by placing a blue background board behind the toy bear. (d)-(f) The ground-truth alpha matte, computed alpha matte, and alpha value error histogram, respectively, in the reference view. (g), (h), and (i) The ground-truth alpha matte, computed alpha matte, and alpha value error histogram, respectively, in the matching view. (a) Reference image. (b) Matching image. (c) Image with blue background. (d) Ground truth alpha. (e) Our alpha results. (f) Alpha error histogram (reference view). (g) Ground truth alpha (matching view). (h) Our alpha result (matching view). (i) Alpha error histogram (matching view).

Another possible approach to solving this problem is to take these refined disparities into further optimization using EM and then reiterate the EM and BP steps. However, this process is computationally expensive and does not guarantee convergence. In our method, a single iteration is sufficiently effective to produce visually and quantitatively satisfying results, as shown in the following sections.

A summary of our algorithm is given in Algorithm 1.

Algorithm 1. Work flow.

```
1) Initialize the disparity maps using the method in [16].
2) Compute the initial trimaps.
    a) Form a two-component Gaussian mixture for the foreground and background disparities and segment the disparities into two layers.
    b) Select pixels around the boundary between the two layers and form the trimaps.
3) Initialize $F^{(0)}$, $B^{(0)}$, and $\alpha^{(0)}$ using the trimap.
4) E-M optimization.
    a) E-step. Compute expectation of $d^f$ and $d^b$ given estimated $F^{(n)}$, $B^{(n)}$, and $\alpha^{(n)}$.
    b) M-step. Compute $\alpha$, $F$, and $B$ to maximize the log-likelihood given the computed expectation $J$. This step iterates between
        i) Optimization of $\alpha^r$ and $\alpha^m$ given estimated $F$ and $B$.
        ii) Optimization of $\{F, B^r, B^m\}$ given estimated $\alpha$.
5) Form a final MRF, use BP to minimize the defined energies, and compute the final disparities.
```

6 Experiment Results

In our approach, each pixel has at most two latent disparity values for the foreground and background. To visualize the hairy object boundary, we construct a *blended disparity map* similar to the alpha matte:

$$d_p^{show} = \alpha_p d_p^f + (1 - \alpha_p) d_p^b. \tag{31}$$

It has been used to illustrate the boundary details in Figs. 1c and 4f.

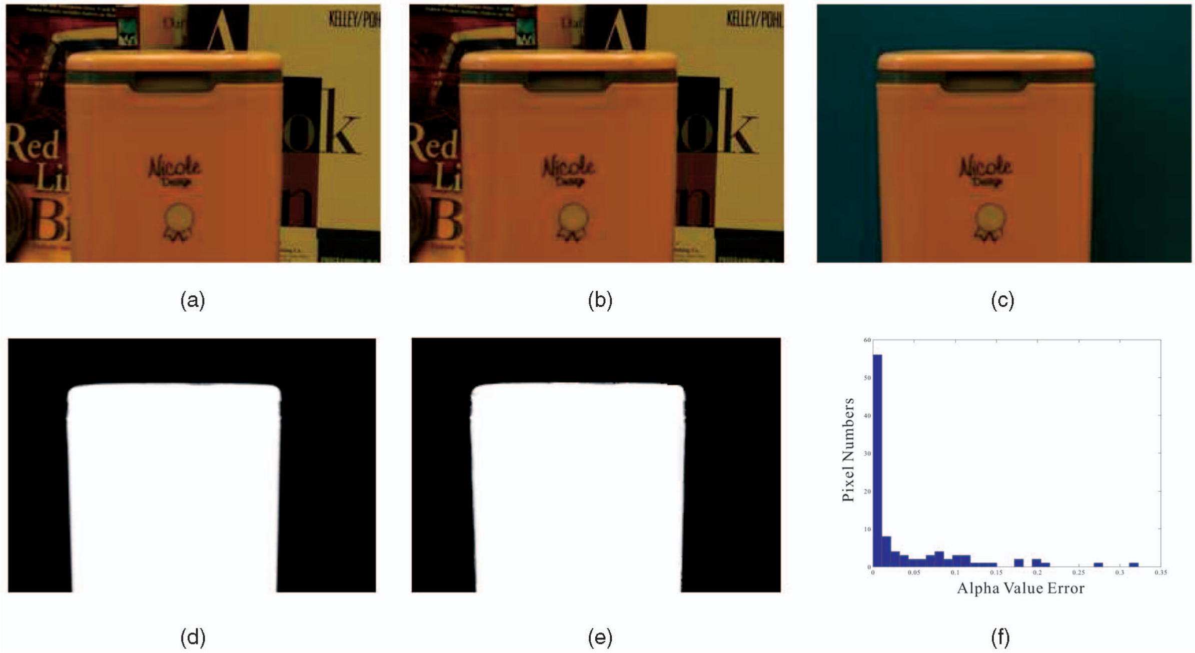


Fig. 7. Pink box example. (a) and (b) The stereo images are shown. The box has a narrow fractional boundary. (c) The same foreground object in front of a known background. (d) Ground truth of alpha matte computed from (c). Our method takes images (a) and (b) as input and estimates the foreground alpha matte, as shown in (e). It is also very similar to the ground truth (d). (f) Shows the alpha error histogram. (a) Reference image. (b) Matching image. (c) Image with constant color background. (d) Ground truth alpha. (e) Our alpha matte result. (f) Alpha error histogram.

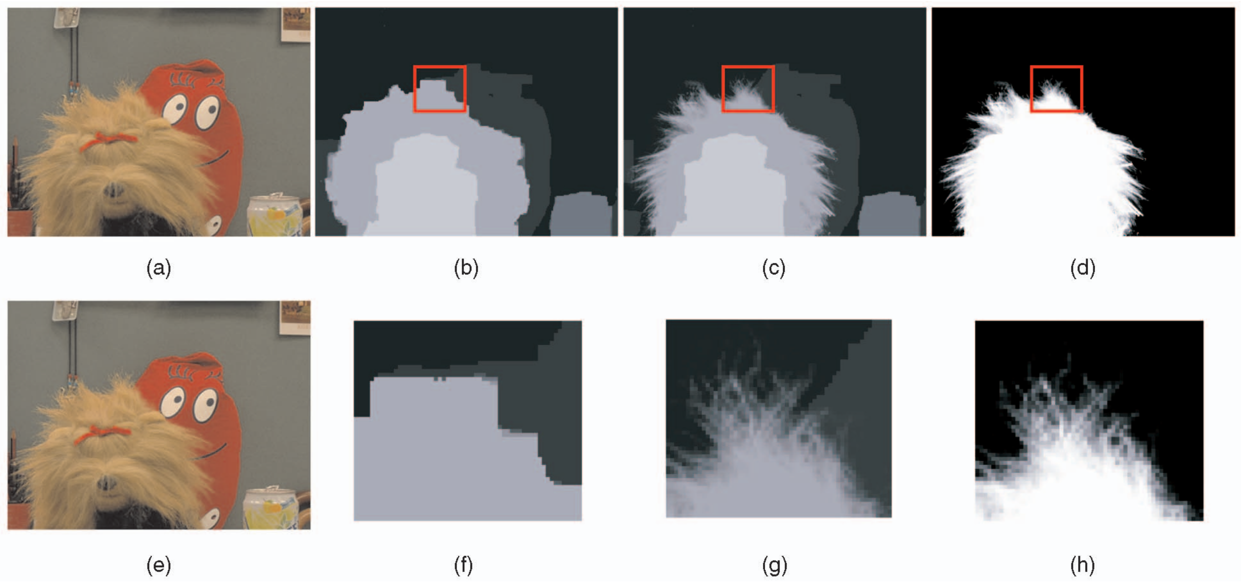


Fig. 8. Brown plush toy example. (a) and (e) The input stereo images are shown. (b) Stereo matching result in [16]. (c) The *blended disparity map* computed by our method. (d) Our estimated alpha matte. (f), (g), (h) Magnified regions. (a) Reference image. (b) Disparity result in [16]. (c) Our disparity result. (d) Our alpha matte result. (e) Matching image. (f) Magnified region of (b). (g) Magnified region of (c). (h) Magnified region of (d).

6.1 Alpha Similarity Validation

In Figs. 6 and 7, we illustrate two examples to validate our alpha similarity constraint. A toy bear example is shown in Fig. 6, where Figs. 6a and 6b contain the reference image C^r and the matching image C^m, respectively. Using the general initialization and by applying the optimization, our method estimates the alpha matte for the reference view, as shown in Fig. 6e. To compare, we show the ground-truth alpha matte computed by the blue-screen matting method [14] in Fig. 6c without modifying the foreground object and the camera. It can be observed that our alpha matte estimate is similar to the ground truth. The root mean squared (RMS) error between our alpha matte result and the ground truth is only 0.024. The alpha error histogram for the pixels in the fractional boundary region is shown in Fig. 6f.

To verify that our method is not biased toward estimating only alpha values correctly in the reference view, we also show our alpha estimate for the matching view in Fig. 6h, where the corresponding ground-truth alpha matte is shown in Fig. 6g. The RMS error of our alpha estimate is only 0.026. The alpha error histogram for the pixels in the fractional boundary regions in the matching view is shown in Fig. 6i, which is also similar to that of the reference view Fig. 6f.

In Fig. 7, we show another example, where the foreground object has a narrow fractional boundary. Figs. 7a and 7b show the input images. Our computed alpha matte for the reference view is shown in Fig. 7e. Similar to the previous example, we compute the ground-truth alpha Fig. 7d using the blue-screen matting method [14] on the background-replaced image Fig. 7c. The RMS error between our alpha result and the ground truth is only 0.013, and the alpha value error histogram for the pixels in the fractional boundary regions is shown in Fig. 7f.

These two examples empirically justify our alpha correspondences. This appropriate alpha model plays an important role in handling objects with wide or narrow fractional boundaries.

6.2 Disparities in Fractional Stereo Matching

The "fan" example with a fractional boundary has been shown in Figs. 1 and 4.

Fig. 8 shows another challenging example where two stereo images Figs. 8a and 8e contain a brown toy bear with long hair. Traditional stereo matching methods do not work well on color-blended pixels. Fig. 8b shows the disparity result from the method proposed in [16]. Because there is no consideration of fractional boundary, errors are inevitably produced. Figs. 8c and 8d show the blended disparity map and the alpha matte computed by our method. The complex fractional hair structure is faithfully retained.

6.3 Result with Standard Stereo Images

Our approach can also be applied to conventional stereo images to improve the disparity computation. We show the "Tsukuba" example in Fig. 9.

Our algorithm initializes the disparity map using the method proposed in [16], as shown in Fig. 9e. The boundary of the lamp is not smooth since only pixel matching information is used in estimating the disparities. The lamp is in the nearest layer, and a considerable distance exists between the lamp and other background objects. Therefore, we directly threshold the disparity map to separate the foreground and the background. The threshold value is set to 13. Then, the foreground boundary is expanded by 2 pixels to form the trimap, as shown in Fig. 9b.

Our method considers boundary blending between the lamp and the background. After optimization, we obtain the alpha matte of the foreground lamp, as shown in Fig. 9c. The boundary is faithfully recovered.

Our algorithm also refines the disparities in the unknown region. In order to reasonably compare our dual-layer result with the disparity results from other methods, we use the following threshold to generate a single disparity value for each pixel:

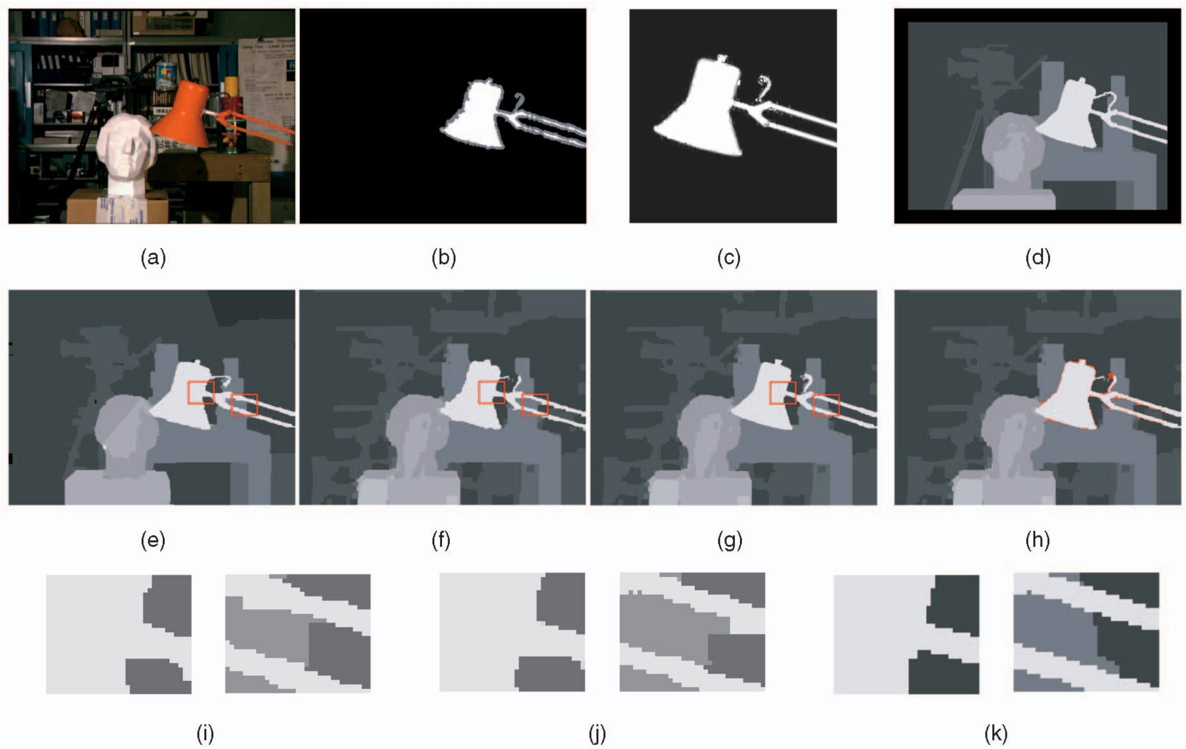


Fig. 9. "Tsukuba" example. (a) The reference image. (b) The initial trimap. (c) Zoom-in view of the alpha matte of the foreground lamp computed by our method. (d) The ground-truth disparity map. (e) Disparity result by the patch-based method [4]. (f) Disparity result by symmetric stereo matching [16]. (g) Our optimized disparity map. (h) Highlight of the improvement of our disparity estimation along the lamp boundary compared to (f). The red pixels highlight the differences. (i), (j), (k) The magnified patches extracted from (e) to (g) for comparison. (a) One of the input images. (b) Initial trimap. (c) Our alpha result (zoom-in). (d) Ground-truth disparity. (e) Disparity result in [4]. (f) Disparity result in [16]. (g) Our disparity result. (h) Disparity difference illustration. (i) Magnified regions of (e). (j) Magnified regions of (f). (k) Magnified regions of (g).

$$d_p^{refine} = \begin{cases} d_p^f & \alpha_p \geq 0.5, \\ d_p^b & \alpha_p < 0.5. \end{cases} \tag{32}$$

The above threshold classifies a pixel as part of the foreground if the pixel's alpha value is large. Otherwise, this pixel belongs to the background. The alpha value, in this process, is regarded as the estimated probability that a pixel is in the foreground or background.

Our final disparity map is shown in Fig. 9g. Two other disparity map results from the state-of-the-art stereo matching algorithms are illustrated in Figs. 9e and 9f. We visualize the difference between Figs. 9f and 9g in map Fig. 9h, where the red pixels highlight the improvement in our disparity estimate. In Fig. 10, we show the error statistics obtained from the Middlebury stereo vision web page ("http://cat.middlebury.edu/stereo/"). Our small disparity error validates the use of the dual-layer boundary refinement, especially for pixels near depth discontinuities.

6.4 Alpha Matte Results and Comparison

In this section, we show that our method can also be regarded as a new matting algorithm using stereo configuration. Our method produces better results compared to other single natural image matting methods.

In Fig. 11, a synthetic example illustrates the effectiveness of our method in handling difficult matting problems. Figs. 11a and 11b show two views of a toy in front of a two-color background. The brown background color is similar to the toy's color. The results by Bayesian matting [2], Wang and Cohen [22], and Levin et al. [10] are shown in Figs. 11g, 11h, and 11i, respectively. Because of the difficulties in distinguishing the foreground pixels from the background ones in this example, extracting the hairy toy from a single image is difficult. Artifacts can be noticed in these results, as shown in the magnified regions Figs. 11j, 11k, and 11l. This example is also difficult for a standard stereo matching method [16] to produce a satisfactory disparity map, as shown in Fig. 11d.

In our unconventional stereo matching framework, the alpha model is used. After optimization, our method outputs a fractional disparity map Fig. 11e. Our automatically extracted foreground alpha matte is shown in Fig. 11f. The mean square errors of the alpha matte obtained from different matting methods are listed in Table 1. This quantitative comparison clearly shows the advantage of our method in solving difficult digital matting problems.

In Fig. 12, we compare our matting result with those produced by the methods in [22] and [10] in the difficult "fan" example. In the input images, the background has complex patterns, and some of the colors are similar to those of the fan. This makes the foreground and the background not clearly distinguishable, largely affecting the performance of previous single natural image matting methods. In matting results Figs. 12b and 12c, produced in [22] and [10], the background patterns are mistakenly estimated as the foreground due to the color similarity. Our automatically computed alpha matte result in Fig. 12d contains fewer errors thanks to the stereo configuration and the joint optimization. The disparity visual clue is essential for disambiguating the underconstrained color blending.

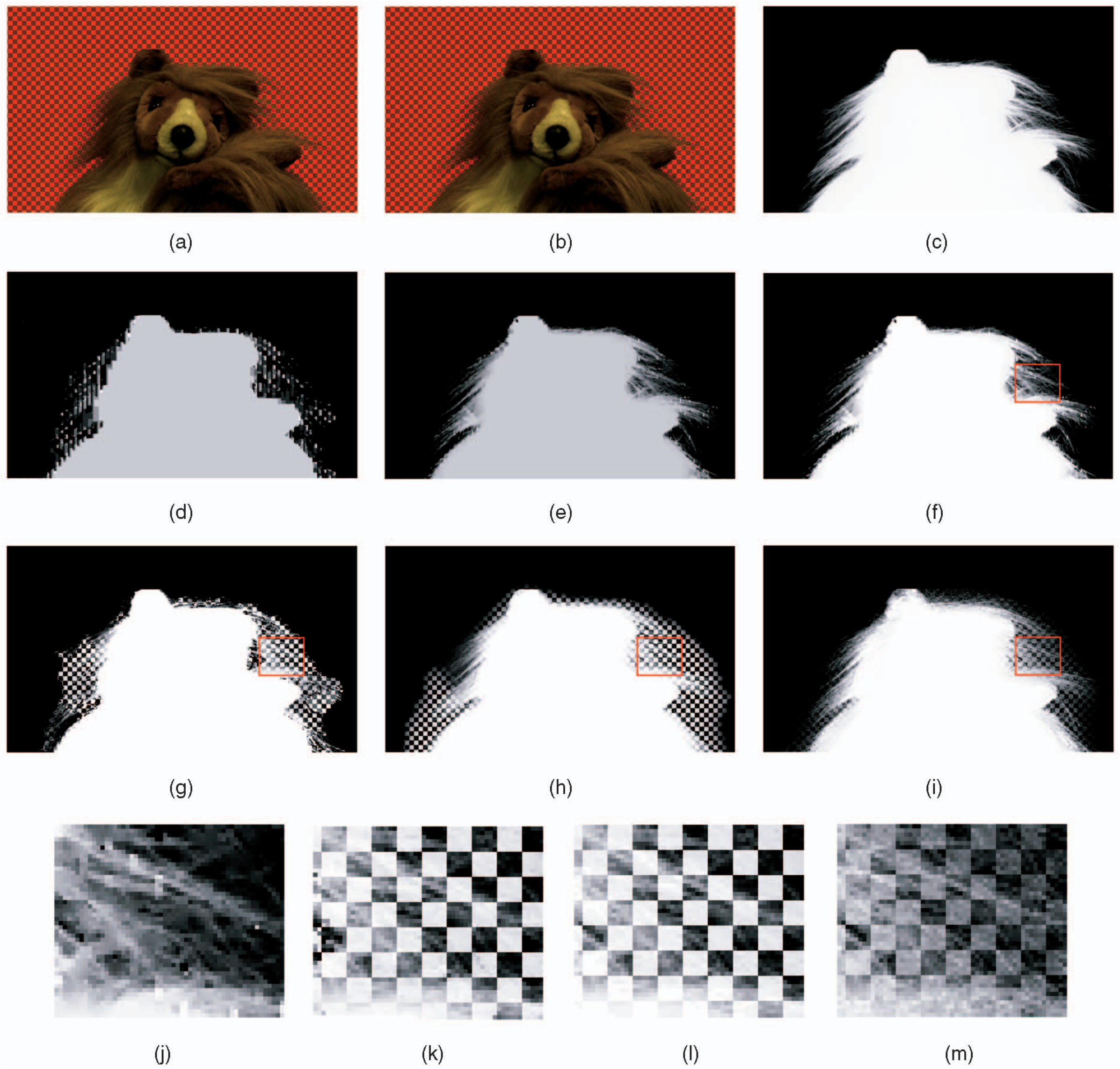
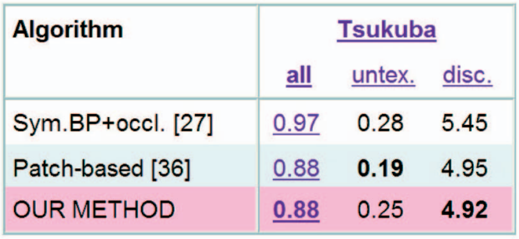


Fig. 10. A synthetic stereo example. (a) Input left view. (b) Input right view. (c) Ground truth of the alpha matte. (d) Initial disparity obtained from that in [16]. (e) Our estimated blended disparity map. The hair structures are successfully recovered. (f) Our alpha matte result. (g) Alpha matte by Bayesian matting [2]. (h) Alpha matte by Wang and Cohen's method [22]. (i) Alpha matte computed by the closed from method [10]. (j), (k), (l), and (m) Side-by-side comparison of the magnified regions of (f), (g), (h), and (i). (a) Reference image. (b) Matching image. (c) Ground-truth alpha matte. (d) Initial disparity. (e) Our final disparity result. (f) Our alpha matte result. (g) Alpha matte result in [2]. (h) Alpha matte result in [22]. (i) Alpha matte result in [10]. (j) Magnified region of (f). (k) Magnified region of (g). (l) Magnified region of (h). (m) Magnified region of (i).

7 Discussion and Conclusion

We have proposed a novel dual-layer approach to solve the stereo matching problem in objects with fractional boundaries using two-frame short-baseline stereo images. Each pixel is assumed to be blended by two latent pixels with different disparities in the estimated "unknown" region. We have defined a probabilistic model constraining the colors, disparities, and the alpha mattes on the two input images and proposed an optimization algorithm using EM to robustly estimate all parameters. Our method provides a unified framework and gives a solution to stereo matching in challenging nonstandard scenes that contain complex



Algorithm	Tsukuba		
	all	untex.	disc.
Sym.BP+occl. [27]	0.97	0.28	5.45
Patch-based [36]	0.88	**0.19**	4.95
OUR METHOD	**0.88**	0.25	**4.92**

Fig. 11. Error comparison in the example "Tsukuba."

TABLE 1
Error Comparison of the Example Shown in Fig. 10

Methods	Mean Square Errors
Levin et al. [10]	0.0050
Wang and Cohen [22]	0.0172
Bayesian [2]	0.0250
Our method	0.0033

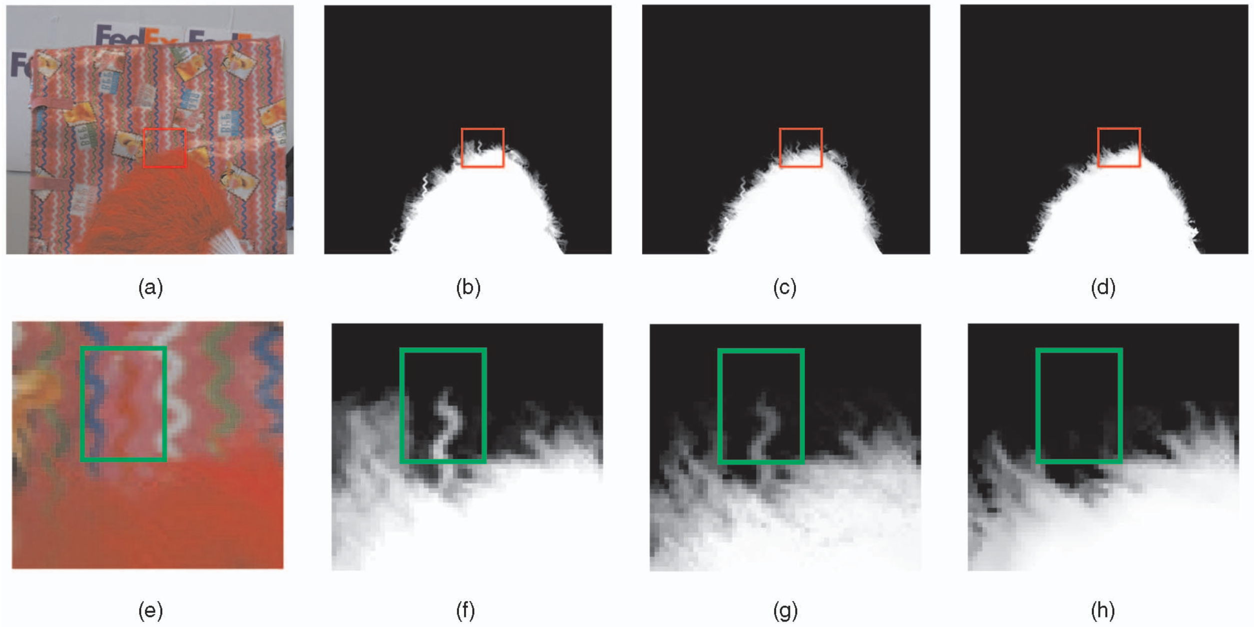


Fig. 12. Comparison of the alpha mattes in the "fan" example. (a) Input reference image. The patterns of the background are complex. (b) Result by Wang and Cohen [22]. (c) Results by Levin et al. [10]. (d) The result from our automatic method that does not require any user interaction. (e), (f), (g), and (h) The magnified regions for comparison. Within the green rectangle, results (f) and (g) mistakenly bring the background pattern into foreground, while our method produces a more satisfying alpha matte. (a) Reference image. (b) Wang and Cohen [22]. (c) Levin et al. [10]. (d) Our result. (e) Magnified region of (a). (f) Magnified region of (b). (g) Magnified region of (c). (h) Magnified region of (d).

color blending. We believe our approach is an important step toward extending the power of stereo matching to more general situations and achieving a full utilization of stereo techniques in depth estimation for a wider spectrum of input images.

The method presented in this paper improves the one proposed in [24] by relaxing the foreground alpha constancy constraint. In this paper, we assume that only the corresponding foreground pixels have similar alpha values, broadening the ability of our method to handle alpha values with sampling errors. We show one example in Fig. 13 where the stereo images are captured with a slightly wider baseline. In this case, the alpha constancy condition does not hold for many foreground boundary pixels. We show the results in [24] and our improved algorithm in Figs. 13d and 13e and compare them to the ground-truth alpha matte Fig. 13f, which is estimated using the blue-screen technique. Although both Figs. 13d and 13e look similar to Fig. 13f, a quantitative comparison of the error statistics shows that our new algorithm outperforms the previous one, validating the effectiveness of using a soft alpha constraint.

Our method can be improved in different ways and leads to a few future research directions. First, it can be extended to handling multiple depth layers. We show how our current method tackles the challenging "Cones" example in Fig. 14. In this example, a complex disparity distribution exists, and no clear foreground can be distinguished. When our method is directly applied to this example, the foreground happens to be extracted as the white region in Fig. 14e, based on which we construct trimap, as shown in Fig. 14f. Our boundary refined result is shown in Fig. 14g. In comparing Fig. 14g to the initial disparity map Fig. 14c, there is only limited improvement in disparities, as illustrated in table Fig. 14h. We expect that if more stereo images are used or other general blending models are investigated, our method can handle challenging multilayer examples.

Second, although we employ the robust EM optimization to compute the alpha and disparity values due to the large number of unknowns, our algorithm still has a chance to get stuck in the local minimum. In these cases, a good initialization of the disparity map and alpha matte would help improve the result quality. Third, when using short-baseline stereo images containing a single hairy-boundary object, the alpha similarity constraint can be generally satisfied between pixels. However, if most pixels in the input image pair indeed violate this constraint, our algorithm may not produce satisfactory results. Using more input images may be a solution.

Appendix A

We describe our optimization process here. α and $\{F, B\}$ are iteratively optimized until convergence.

A.1 Optimizing α with Fixed $\{F, B\}$

When F and B are fixed, given the image width W, the problem turns to searching:

$$\begin{aligned} X^* &= \arg\min_X f(X), \\ X &= \left[\alpha^r_{1,y}, \alpha^r_{2,y}, \ldots, \alpha^r_{W,y}, \alpha^m_{1,y}, \alpha^m_{2,y}, \ldots, \alpha^m_{W,y}\right]^T \end{aligned} \tag{33}$$

along each scan line y. Here, unknowns α^r and α^m are sparsely coupled, making it a multivariable nonlinear optimization problem. Fortunately, we show in the following that the function $f(X)$ is convex with respect to α, so the global optimum can still be obtained in this step.

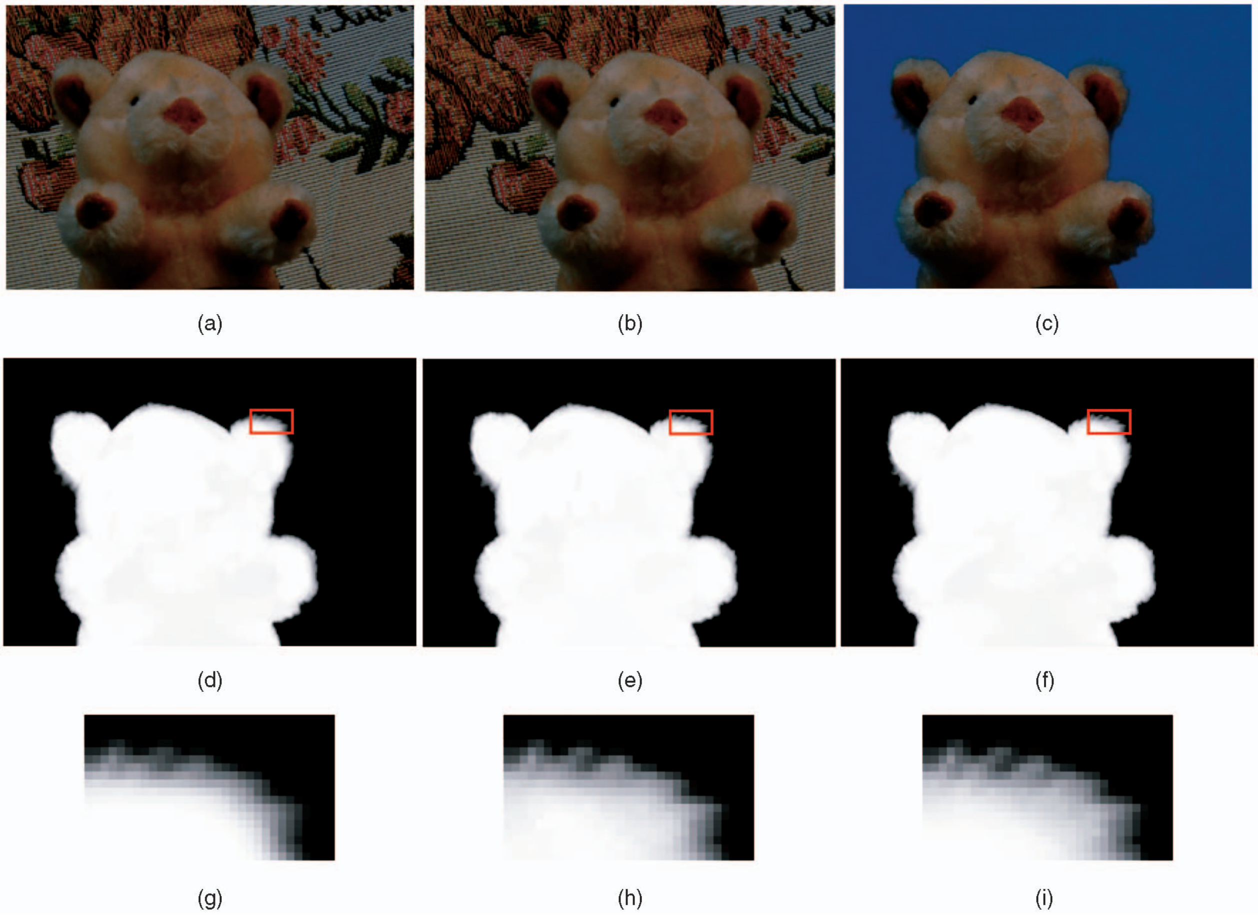
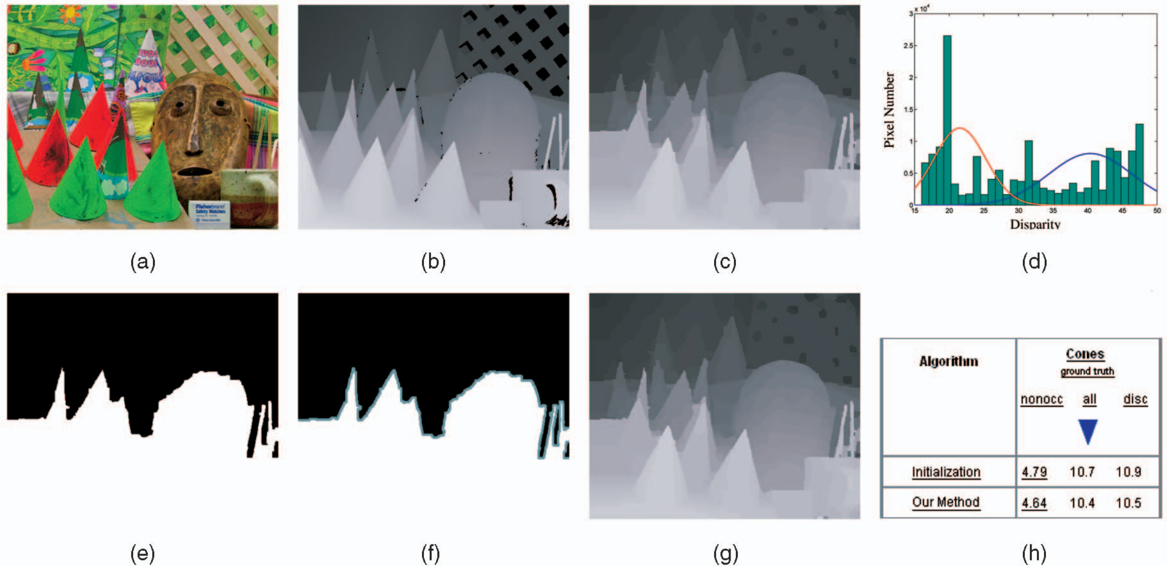


Fig. 13. Slightly wider-baseline example. (a) and (b) Two input stereo images captured with a slightly wider baseline, making the corresponding alpha values dissimilar for many pixels are shown. (c) The blue-screen configuration for computing the ground-truth alpha matte. (d) The alpha matte obtained from our original method in [24]. (e) The result computed from the method presented in this paper. (f) The ground-truth alpha matte computed from that in (c). (g), (h), and (i) The magnified regions of (d), (e), and (f). (a) Reference image. (b) Matching image. (c) Image with blue-color background. (d) Alpha matte result in [24]. (e) Our alpha matte result. (f) Ground-truth alpha. (g) Magnified region of (d). (h) Magnified region of (e). (i) Magnified region of (f).



Algorithm	Cones ground truth nonocc	 all	 disc
Initialization	4.79	10.7	10.9
Our Method	4.64	10.4	10.5

Fig. 14. “Cones” example. (a) The input image. (b) The ground-truth disparity map. (c) The initialized disparity map in [16]. (d) Initial disparity histogram. The foreground and the background layers are initialized using a two-component Gaussian mixture. (e) The binary segmentation of the foreground and the background from the disparity map. (f) The corresponding trimap constructed by dilating the foreground boundary by 4 pixels. (g) Our final disparity result. (h) Errors computed on the initialized disparity map [16] and our final result.

The Hessian Matrix of function $f(X)$ can be written as

$$\nabla^2 f(X) = Z = \begin{bmatrix} Z_{00} & Z_{01} \\ Z_{10} & Z_{11} \end{bmatrix}. \tag{34}$$

Here, Z_{ab}, a, $b \in [0,1]$, is a $W \times W$ matrix. Denoting $p^f_{p,k} = P_p(d^f = k|\Theta^{(n)}, U)$, $p^b_{p,k} = P_p(d^b = k|\Theta^{(n)}, U)$, elements in the matrix are expressed as

$$\begin{aligned} Z_{00}(i,j) = {} & \delta_{ij} \sum_{d^f} p^f_{p,d^f} \Big\{ 1/\sigma_r^2 + \Big(F_p - B^r_p\Big)^T \\ & \Big(F_p - B^r_p\Big)/\omega_C^2 + 1/\omega_\alpha^2 \Big\}, \\ Z_{11}(i,j) = {} & \delta_{ij} \sum_{d^f} p^f_{p+f,d^f} \Big\{ 1/\sigma_m^2 + \Big(F_{p-f} - B^m_p\Big)^T \\ & \Big(F_{p-f} - B^m_p\Big)/\omega_C^2 + 1/\omega_\alpha^2 \Big\}, \\ Z_{01}(i,j) = {} & -p^f_{p,d^f}/\omega_\alpha^2, \\ Z_{10}(i,j) = {} & -p^f_{p+f,d^f}/\omega_\alpha^2, \end{aligned}$$

where $p = i$ and $d^f = j - i$. δ_{ij} is the Kronecker delta. $p - f$ denotes the pixel with coordinate $(x - d^f, y)$ in image m. It can be observed that Z is a symmetric diagonally dominant real matrix with positive diagonal entries. Therefore, it is positively definite, and the function $f(X)$ is convex. We then take partial derivatives of $f(X)$ with respect to X and set them to zero to compute the optimal value of α. After some algebraic manipulations, we obtain the following linear equation system:

$$\begin{bmatrix} G_{00} & G_{01} \\ G_{10} & G_{11} \end{bmatrix} X = \begin{bmatrix} H_0 \\ H_1 \end{bmatrix}, \tag{35}$$

where G_{00}, G_{01}, G_{10}, and G_{11} are all matrices with size $W \times W$. The elements in the matrices are expressed as

$$\begin{aligned} G_{00}(i,j) = {} & \delta_{ij} \sum_{d^f} p^f_{p,d^f} \Big\{ 1/\sigma_r^2 + \Big(F_p - B^r_p\Big)^T \\ & \Big(F_p - B^r_p\Big)/\omega_C^2 + 1/\omega_\alpha^2 \Big\}, \\ G_{11}(i,j) = {} & \delta_{ij} \sum_{d^f} p^f_{p+f,d^f} \Big\{ 1/\sigma_m^2 + \Big(F_{p-f} - B^m_p\Big)^T \\ & \Big(F_{p-f} - B^m_p\Big)/\omega_C^2 + 1/\omega_\alpha^2 \Big\}, \\ G_{01}(i,j) = {} & -p^f_{p,d^f}/\omega_\alpha^2, \\ G_{10}(i,j) = {} & -p^f_{p+f,d^f}/\omega_\alpha^2, \end{aligned}$$

where $p = i$ and $d^f = j - i$. H_0 and H_1 are $W \times 1$ vectors:

$$\begin{cases} H_0(i) = \overline{\alpha^r_p}/\sigma_r^2 + \Big(F_p - B^r_p\Big)^T \Big(C^r_p - B^r_p\Big)/\omega_C^2, \\ H_1(i) = \sum_{d^f} p^f_{p+f,d^f} \Big\{ \overline{\alpha^m_p}/\sigma_m^2 \Big(F_{p-f} - B^m_p\Big)^T \\ \qquad + \Big(C^m_p - B^m_p\Big)/\omega_C^2 \Big\}. \end{cases}$$

Using the above representations, α^r and α^m can be directly computed by solving the linear equations in (35).

A.2 Optimizing $\{F, B\}$ with Fixed α

Similarly, the pixel color $\{F, B^r, B^m\}$ can also be computed along the scan line. Defining

$$\mathbf{V} = \Big[\mathbf{F}, \mathbf{B}^r_y, \mathbf{B}^m_y\Big]^T, \tag{36}$$

where

$$\begin{aligned} \mathbf{F}_y &= [F_{1,y}, F_{2,y}, \ldots, F_{W,y}], \\ \mathbf{B}^r_y &= \Big[B^r_{1,y}, B^r_{2,y}, \ldots, B^r_{W,y}\Big], \\ \mathbf{B}^m_y &= \Big[B^m_{1,y}, B^m_{2,y}, \ldots, B^m_{W,y}\Big]. \end{aligned}$$

$F_{i,j}$, $B^r_{i,j}$, and $B^m_{i,j}$ are 1×3 vectors denoting the foreground and background colors in RGB channels at pixel (i,j). The optimization problem can be expressed as

$$\arg\min_{\mathbf{V}} f(\mathbf{V}). \tag{37}$$

Since the Hessian Matrix of $f(\mathbf{V})$ can also be proved to be positively definite, we take partial derivatives of f with respect to $\mathbf{V}$ and set them to be zero to compute the optimal values of $\{F, B^r, B^m\}$. Along each scan line y, we have

$$\begin{bmatrix} A_{00} & A_{01} & A_{02} \\ A_{10} & A_{11} & A_{12} \\ A_{20} & A_{21} & A_{22} \end{bmatrix} \mathbf{V} = \begin{bmatrix} M_0 \\ M_1 \\ M_2 \end{bmatrix}, \tag{38}$$

where A_{uv} can be written as

$$A_{uv} = \begin{bmatrix} a^{1,1}_{uv} & a^{1,2}_{uv} & \cdots & a^{1,W}_{uv} \\ a^{2,1}_{uv} & a^{2,2}_{uv} & \cdots & a^{2,W}_{uv} \\ \vdots & \vdots & \vdots & \vdots \\ a^{W,1}_{uv} & a^{W,2}_{uv} & \cdots & a^{W,W}_{uv} \end{bmatrix}, u, v \in \{0, 1, 2\},$$

and $a^{i,j}_{uv}$s are 3×3 matrices with the following expressed elements:

$$\begin{aligned} a^{i,j}_{00} &= \delta_{ij} \sum_{d^f} p^f_{p,d^f} \bigg(\Big(\Big\{\alpha^r_p\Big\}^2 + \Big\{\alpha^m_{p+f}\Big\}^2 \Big) I/\omega_C^2 + \Sigma^{-1}_{F_p} \bigg), \\ a^{i,j}_{01} &= \delta_{ij} \alpha^r_p \Big(1 - \alpha^r_p\Big) I/\omega_C^2, \\ a^{i,j}_{02} &= p^f_{p,d^f} \alpha^m_{p+f} \Big(1 - \alpha^m_{p+f}\Big) I/\omega_C^2, \quad d^f = j - i, \\ a^{i,j}_{10} &= \delta_{ij} \alpha^r_p \Big(1 - \alpha^r_p\Big) I/\omega_C^2, \\ a^{i,j}_{11} &= \delta_{ij} \bigg(\Big\{ \Big(1 - \alpha^r_p\Big)^2/\omega_C^2 + 1/\omega_B^2 \Big\} I + \Sigma^{-1}_{B^r_p} \bigg), \\ a^{i,j}_{12} &= -p^b_{p,d^b} I/\omega_B^2, \quad d^b = j - i, \\ a^{i,j}_{20} &= p^f_{p+f,d^f} \alpha^m_p \Big(1 - \alpha^m_p\Big) I/\omega_C^2, \quad d^f = j - i, \\ a^{i,j}_{21} &= -p^b_{p+b,d^b} I/\omega_B^2, \quad d^b = j - i, \\ a^{i,j}_{22} &= \delta_{ij} \bigg(\Big\{ \Big(1 - \alpha^m_p\Big)^2/\omega_C^2 + 1/\omega_B^2 \Big\} I + \Sigma^{-1}_{B^m_p} \bigg). \end{aligned}$$

M_0, M_1, and M_2 can be written as

$$M_u = \big[M^1_u, M^2_u, \ldots, M^W_u\big]^T, u \in \{0, 1, 2\},$$

where M_u^is are 1×3 vectors:

$$M_0^i = \sum_{d^f} p_{p,d^f}^f \left(\left\{ \alpha_p^r C_p^r + \alpha_{p+f}^m C_{p+f}^m \right\} / \omega_C^2 + \Sigma_{F_p}^{-1} \overline{F_p} \right),$$

$$M_1^i = \left(1 - \alpha_p^r\right) C_p^r / \omega_C^2 + \Sigma_{B_p^r}^{-1} \overline{B_p^r},$$

$$M_2^i = \left(1 - \alpha_p^m\right) C_p^m / \omega_C^2 + \Sigma_{B_p^m}^{-1} \overline{B_p^m}.$$

Here, I is a 3×3 identity matrix. Given the defined linear equations in (38), F, B^r, and B^m can be computed directly.

Acknowledgments

The authors would like to thank the associate editor and all the reviewers for their constructive comments to improve the manuscript. This work was supported by a grant from the Research Grants Council of the Hong Kong Special Administrative Region, China (Project No. 412307).

References

[1] Y. Boykov, O. Veksler, and R. Zabih, "Fast Approximate Energy Minimization via Graph Cuts," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 23, no. 11, pp. 1222-1239, Nov. 2001.

[2] Y.-Y. Chuang, B. Curless, D.H. Salesin, and R. Szeliski, "A Bayesian Approach to Digital Matting," *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition (CVPR '01),* vol. 2, pp. 264-271, 2001.

[3] A. Criminisi and A. Blake, "The SPS Algorithm: Patching Figural Continuity and Transparency by Split-Patch Search," *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition (CVPR '04),* vol. 1, pp. 342-349, 2004.

[4] Y. Deng, Q. Yang, X. Lin, and X. Tang, "A Symmetric Patch-Based Correspondence Model for Occlusion Handling," *Proc. IEEE Int'l Conf. Computer Vision (ICCV '05),* pp. 1316-1322, 2005.

[5] P.F. Felzenszwalb and D.P. Huttenlocher, "Efficient Belief Propagation for Early Vision," *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition (CVPR '04),* vol. 70, no. 1, pp. 261-268, 2004.

[6] S.W. Hasinoff, S.B. Kang, and R. Szeliski, "Boundary Matting for View Synthesis," *Computer Vision and Image Understanding,* vol. 103, no. 1, pp. 22-32, 2006.

[7] L. Hong and G. Chen, "Segment-Based Stereo Matching Using Graph Cuts," *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition (CVPR '04),* vol. 1, pp. 74-81, 2004.

[8] N. Joshi, W. Matusik, and S. Avidan, "Natural Video Matting Using Camera Arrays," *Proc. ACM SIGGRAPH '06,* pp. 779-786, 2006.

[9] V. Kolmogorov and R. Zabih, "Computing Visual Correspondence with Occlusions Using Graph Cuts," *Proc. IEEE Int'l Conf. Computer Vision (ICCV '01),* vol. 2, p. 508, 2001.

[10] A. Levin, D. Lischinski, and Y. Weiss, "A Closed Form Solution to Natural Image Matting," *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition (CVPR '06),* pp. 61-68, 2006.

[11] A. Levin, A. Rav-Acha, and D. Lischinski, "Spectral Matting," *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition (CVPR),* 2007.

[12] T. Minka, *Expectation-Maximization as Lower Bound Maximization,* http://research.microsoft.com/~minka/papers/em.html, 1998.

[13] D. Scharstein and R. Szeliski, "A Taxonomy and Evaluation of Dense Two-Frame Stereo Correspondence Algorithms," *Int'l J. Computer Vision,* vol. 47, nos. 1-3, pp. 7-42, 2002.

[14] A.R. Smith and J.F. Blinn, "Blue Screen Matting," *Proc. ACM SIGGRAPH '96,* pp. 259-268, 1996.

[15] J. Sun, J. Jia, C.-K. Tang, and H.-Y. Shum, "Poisson Matting," *Proc. ACM SIGGRAPH '04,* pp. 315-321, 2004.

[16] J. Sun, Y. Li, S.B. Kang, and H.-Y. Shum, "Symmetric Stereo Matching for Occlusion Handling," *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition (CVPR '05),* vol. 2, pp. 399-406, 2005.

[17] J. Sun, Y. Li, S.B. Kang, and H.-Y. Shum, "Flash Matting," *ACM Trans. Graphics,* vol. 25, no. 3, pp. 772-778, 2006.

[18] J. Sun, N.-N. Zheng, and H.-Y. Shum, "Stereo Matching Using Belief Propagation," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 25, no. 7, pp. 787-800, July 2003.

[19] R. Szeliski and P. Golland, "Stereo Matching with Transparency and Matting," *Int'l J. Computer Vision,* vol. 32, no. 1, pp. 45-61, 1999.

[20] E. Trucco, A. Fusiello, and A. Verri, "Rectification with Unconstrained Stereo Geometry," *Proc. British Machine Vision Conf. (BMVC),* 1997.

[21] Y. Tsin, S.B. Kang, and R. Szeliski, "Stereo Matching with Reflections and Translucency," *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition (CVPR '03),* vol. 1, p. 702, 2003.

[22] J. Wang and M.F. Cohen, "An Iterative Optimization Approach for Unified Image Segmentation and Matting," *Proc. IEEE Int'l Conf. Computer Vision (ICCV '05),* pp. 936-943, 2005.

[23] Y. Wexler, A.W. Fitzgibbon, and A. Zisserman, "Bayesian Estimation of Layers from Multiple Images," *Proc. European Conf. Computer Vision (ECCV '02),* pp. 487-501, 2002.

[24] W. Xiong and J. Jia, "Stereo Matching on Objects with Fractional Boundary," *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition (CVPR),* 2007.

[25] Q. Yang, L. Wang, R. Yang, H. Stewenius, and D. Nister, "Stereo Matching with Color-Weighted Correlation, Hierarchical Belief Propagation and Occlusion Handling," *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition (CVPR '06),* pp. 2347-2354, 2006.

[26] L. Zhang and S.M. Seitz, "Parameter Estimation for MRF Stereo," *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition (CVPR '05),* vol. 2, pp. 288-295, 2005.

[27] C.L. Zitnick, N. Jojic, and S.B. Kang, "Consistent Segmentation for Optical Flow Estimation," *Proc. IEEE Int'l Conf. Computer Vision (ICCV '05),* pp. 1308-1315, 2005.

[28] C.L. Zitnick, S.B. Kang, M. Uyttendaele, S. Winder, and R. Szeliski, "High-Quality Video View Interpolation Using a Layered Representation," *ACM Trans. Graphics,* vol. 23, no. 3, pp. 600-608, 2004.

Wei Xiong received the BEng degree in electronic engineering from Tsinghua University and the MPhil degree in computer science from the Chinese University of Hong Kong. His research interests include natural image matting and stereo matching.

Hin Shun Chung received the BEng degree in computer engineering from the Chinese University of Hong Kong. He received the MPhil degree from the same department in August 2008. His research interests include photometric stereo and stereo matching. He is a student member of the IEEE.

Jiaya Jia received the PhD degree in computer science from the Hong Kong University of Science and Technology in 2004. He joined the Department of Computer Science and Engineering, Chinese University of Hong Kong, in September 2004, where he is currently an assistant professor. His research interests include vision geometry, image/video editing and enhancement, motion deblurring, and Markov Random Field analysis. He has served on the program committees of ICCV, CVPR, ECCV, and ACCV. He is a member of the IEEE.