Stereo Matching on Objects with Fractional Boundary*

Wei Xiong and Jiaya Jia
Department of Computer Science and Engineering
The Chinese University of Hong Kong
{wxiong, leojia}@cse.cuhk.edu.hk

Abstract

Conventional stereo matching algorithms assume color constancy on the corresponding opaque pixels in the stereo images. However, when the foreground objects with fractional boundary are blended to the scene behind using unknown alpha values, due to the spatially varying disparities for different layers, the color constancy does not hold any more.

In this paper, we address the fractional stereo matching problem. A probability framework is introduced to establish the correspondences of pixel colors, disparities, and alpha values in different layers. We propose an automatic optimization method to solve a Maximum a posteriori (MAP) problem using Expectation-Maximization (EM), given the input of only a narrow-band stereo image pair. Our method naturally encodes pixel occlusion in the formulation of layer blending without a special detection process. We demonstrate the effectiveness of our method using difficult stereo images.

1. Introduction

Stereo matching has been an essential research topic in computer vision, and has been made rapid and significant progress in recent years [11, 17, 20]. Most conventional two-frame stereo matching approaches compute disparities and detect occlusions assuming that each pixel in the input image has a unique depth value.

However, this representation has limitation in faithfully modeling objects with fractional boundaries where pixels are blended to the scene behind. Directly applying previous stereo matching methods on ubiquitous hairy objects may produce problematic disparities. One example is shown in Fig. 1 where the input images (a) and (b) contain a hairy fan in front of a background with similar colors. The result generated from the stereo matching method [11] is shown in (c). The disparities are incorrect along the fan's boundary without considering color blending.

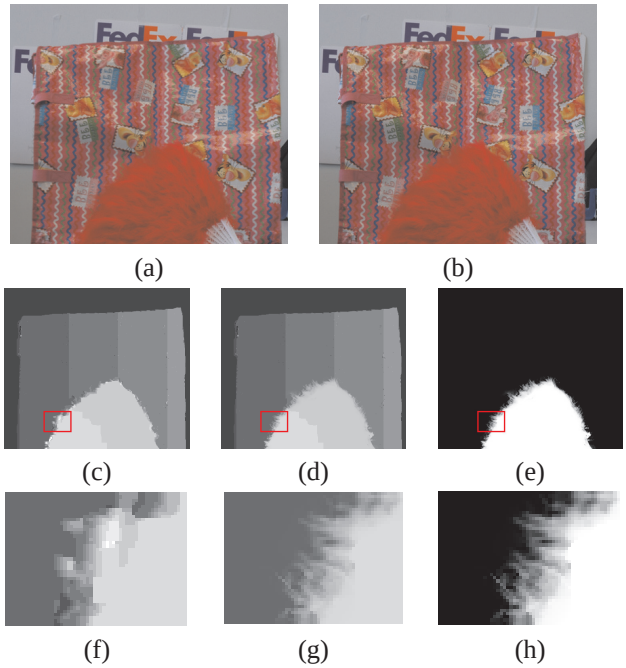

Figure 1. A stereo image pair containing a hairy object. (a) and (b) Input stereo images containing a hairy fan. Notice that the colors of the background scene and the hairy fan are similar. (c) Stereo matching result from Sun et al's method [11]. Because of the color blending, the assumption of color constancy is invalid along the boundary of the fan, making the result problematic. (d) The stereo matching result obtained from our approach. The hairy structure is successfully preserved. (e) The computed alpha matte of the fan using our method. (f)-(h) Magnified regions of the results.

Recent development on stereo matching algorithms partially generalizes the above assumption and introduces the transparency constraints. Szeliski *et al*. [13] propose to solve the stereo matching problem with opacity using multiple input images where the color and transparency refinement are formulated as a non-linear minimization problem. However, their method has difficulties to deal with objects containing thin and long hairs or with complex alpha matte given a small number of input images. Assuming a binary reflection map model, Tsin *et al*. [17] propose to estimate the front translucent and rear background layers us-

*The work described in this paper was supported by a grant from the Research Grants Council of the Hong Kong Special Administrative Region, China (Project No. 412206) and is affiliated with the Microsoft-CUHK Joint Laboratory.

ing Graph Cuts. The pixel colors are further computed by iteratively reducing an energy in multi-frame configuration. This method is not applicable to objects with general fractional boundary. Both of the above methods require multiple input images in order to obtain satisfactory disparity maps.

In this paper, taking the input of only a narrow-band stereo image pair captured in a scene where the hairy objects are in front, we neatly formulate the estimation of alpha values, disparities, and pixel colors in a probability framework. It is solved using Expectation-Maximization method. In our method, the color correspondences are established on the blended layers respectively. The two processes of transparency optimization and disparity estimation boost each other, effectively reducing the disparity errors. We show the disparity map and the alpha matte computed using our approach in Figure 1 (d) and (e) respectively. The comparison of the disparities are illustrated in (f) and (g). The estimated alpha matte of the fan shown in (e) and (h) is visually satisfying given the complex structures in the input images.

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 needs to be defined separately on pixels having no correspondences [11]. In our approach, any pixel in the layer of the scene behind the hairy objects can be partially occluded, entirely occluded, or unoccluded according to the degree of transparency. The three situations can be naturally encoded using alpha values without a special treatment.

The rest of the paper is organized as follows. Section 2 reviews previous work on stereo matching and digital image matting. We introduce our model and notations in Section 3. The detailed optimization process is described in Section 4. In Section 5, we show and compare the experimental results. We conclude our paper in Section 6.

2. Related Work

Our work is related to the research on dense stereo matching and digital image matting.

Stereo matching. There have been many methods developed to solve the conventional stereo matching problem. A two-frame stereo matching survey is in [14].

In recent years, Markov Random Field (MRF) is widely used in stereo matching [8, 12, 6, 11]. Most of these methods solve the MRF by either Belief Propagation (BP) [4] or Graph Cuts [1]. In [8], a method related to expansion move algorithm is used to find the local minimum of an energy function. Graph Cuts algorithm is applied to compute the optimal value. [3] segments the two input frames into small patches. Graph Cuts is also used to find the disparity and occlusions embedded in the patches with the symmetric mapping. [15] compares the performance of Graph Cuts and Belief Propagation on a set of images, and concludes that, in general, the results produced by the two algorithms are comparable.

The above methods are not proposed to solve the stereo matching problem with color blending because of the disparity ambiguities. Szeliski and Golland [13] first propose to solve stereo matching with boundary opacity. The visibility is computed through re-projection, where color and transparency refinement are formulated as a non-linear minimization problem. Wexler *et al*. [19] compute alpha mattes and estimate layers from multiple images with known background information. [17] estimates depth with the consideration of layer overlapping. It uses nested plane sweep with refinement from Graph Cuts. The attenuation factors for color blending at reflecting areas are constant. In [5, 22], boundary matting along depth discontinuities is applied to refine the estimation of foreground objects for a better view synthesis. Besides, [21] computes the alpha contribution on overlapped regions among segments. A more accurate optical flow estimation can thus be achieved.

Digital image matting. Natural image matting is to separate the blended pixels by computing the foreground, background and the alpha matte respectively given a natural input image. Using trimaps, Bayesian Matting [2] and Poisson Matting [10] estimate the foreground and background colors by collecting samples. Wang and Cohen [18] introduce an optimization approach based on Belief Propagation to estimate the alpha matte without trimaps. In [9], Levin *et al*. propose a closed form solution to solve the matting problem given the user input of a few strokes. Joshi *et al*. [7] use an autofocus system to determine pixel correspondences among multi-images to enhance the performance of video matting. All these methods cannot be directly applied to stereo matching without the consideration of the correspondence of colors and alpha values in input images.

3. Model Definition

Conventional two-frame dense stereo matching approaches compute depth value by estimating the correspondence of pixels in the input image pair. In this paper, we also use two images C^r and C^m in different viewing positions, and assume that the reference image C^r and the matching image C^m are rectified [16]. Conventionally, for a pixel (x, y) in C^r and its corresponding pixel (x', y') in C^m with disparity d, we have

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

The conventional stereo matching problem is formulated as the estimation of disparity d using the color constancy on the matched pixels in a scene with Lambertian reflectance:

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

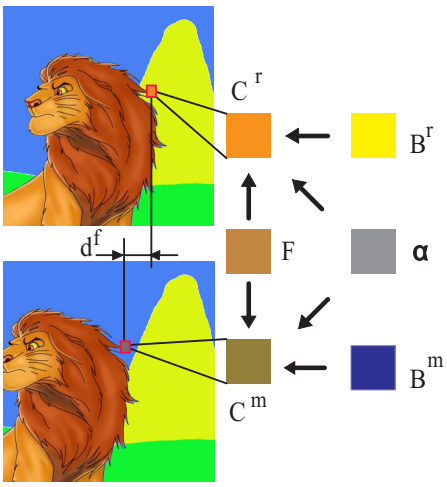

Figure 2. Color constancy on blended pixels. Given the input stereo image pair as shown, the semitransparent pixels C^r and C^m in the hair are blended by the foreground and the background. Since C^r and C^m are matched in foreground layer with disparity d^f, they have similar foreground color F and alpha value α. However, the partially occluded background pixels are different as shown in B^r and B^m.

In our problem definition, to model the color blending between the objects with hairy boundaries and the scene behind, 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 unconcluded, partially occluded, or entirely occluded by F according to the degree of transparency. Applying the equation of alpha blending, the blended color in each pixel is formulated as

$$\begin{aligned} C^k(x,y) &= \alpha^k(x,y)F^k(x,y) + \\ &\quad (1-\alpha^k(x,y))B^k(x,y), \end{aligned} \tag{3}$$

where $k \in \{r, m\}$. Accordingly, in our stereo model, instead of defining a single disparity d for each pixel in the input images, we introduce disparities d^f and d^b for latent pixels in foreground F and background B respectively. This definition provides flexibility to model occlusions. Hence, for each latent foreground pixel $F^r(x,y)$ (or the background pixel $B^r(x,y)$) in C^r, applying d^f (or d^b), we can obtain a matched pixel $F^m(x,y)$ (or $B^m(x,y)$) in C^m, 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, since there are measurable discontinuities in depth between the foreground objects and background scene. The occlusion between them can be nicely formulated using Equation 3 according to the corresponding alpha values without another explicit occlusion detection process:

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

Using a narrow-band camera setup, we can naturally assume that the transparency is an inherent property of foreground and is invariant for corresponding foreground pixels. This assumption has also been validated through our experiments on a variety of scenes. Specifically, if a foreground pixel (x,y) in C^r is matched to $(x+d^f,y)$ in C^m, we have

$$\alpha^r(x,y) = \alpha^m(x+d^f,y). \tag{6}$$

In the rest of the paper, for simplicity, we use subscripts p, $p+d^f$, and $p+d^b$ to denote pixel in (x,y), $(x+d^f,y)$, and $(x+d^b,y)$ respectively. Substituting Equation 2, 4, and 6 into Equation 3, we obtain the following two equations for each corresponding pixel pair in the input images:

$$\begin{cases} C_p^r &= \alpha_p F_p^r + (1-\alpha_p)B_p^r \\ C_{p+d^f}^m &= \alpha_{p+d^f}^m F_{p+d^f}^m + (1-\alpha_{p+d^f}^m)B_{p+d^f}^m \end{cases} \tag{7}$$

We show one example in Fig. 2 where two corresponding foreground pixels are blended by different background pixels due to the disparity differences. In Equation 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. Without loss of generality, we optimize F^r and its corresponding α^r in our method. F^m and α^m, as complements in stereo configuration, are computed by mapping the foreground pixels in C^r to C^m using the computed disparities. We estimate B^r and B^m separately in a symmetric manner. It guarantees that the unmatched background pixels due to the occlusions are appropriately handled, which in turn improves the estimation of the disparities and foreground pixels.

In what follows, without special annotation, we use α and F to denote α^r and F^r respectively. Thus, substituting Equation 4 and 6 into Equation 7, $C_{p+d^f}^m$ can be rewritten as

$$\begin{aligned} C_{p+d^f}^m &= \alpha_{p+d^f}^m F_{p+d^f}^m + (1-\alpha_{p+d^f}^m)B_{p+d^f}^m \\ &= \alpha_p F_p + (1-\alpha_p)B_{p+d^f}^m \end{aligned} \tag{8}$$

4. Our Approach

In this section, we describe our optimization method to solve the fractional stereo matching problem formulated in Equation 7 and 8.

4.1. Optimization

Given the observation $U = \{C^r, C^m\}$, we separate the unknowns into a parameter set $\Theta = \{F, B^r, B^m, \alpha\}$ and

hidden data $J = \{d^f, d^b\}$. In this section, we aim at estimating the parameters using Expectation-Maximization

$$\begin{aligned}\Theta^* &= \arg\max_{\Theta} \log P(U, \Theta) \\ &= \arg\max_{\Theta} \log \sum_{J \in J^n} P(U, J, \Theta), \end{aligned} \tag{9}$$

where J^n is the domain of J. After we have obtained the optimized parameters, we compute an optimal J combining the spatial smoothness constraint.

4.1.1 Expectation Step

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

$$\begin{aligned}&E(P_p(d^f = d_1, d^b = d_2|\Theta^{(n)}, U)) \\ &= E(P_p(d^f = d_1|\Theta^{(n)}, U) P_p(d^b = d_2|\Theta^{(n)}, U)) \\ &= E(P_p(d^f = d_1|\Theta^{(n)}, U)) E(P_p(d^b = d_2|\Theta^{(n)}, U)).\end{aligned} \tag{10}$$

In what follows, we describe the expectation computation on d^f and d^b respectively.

Computing $E((P_p(d^f = d_1|\Theta^{(n)}, U))$. The conditional probability d^f is formulated using Bayes' theorem:

$$\begin{aligned}&P_p(d^f|\Theta^{(n)}, U) \\ \propto\ &P_p(d^f|U, B^{r(n)}, B^{m(n)}, \alpha^{(n)}) \\ \propto\ &P_p(B^{r(n)}, B^{m(n)}, \alpha^{(n)}|d^f, U) \cdot P_p(d^f|U).\end{aligned} \tag{11}$$

According to Equation 7 and 8, ideally, the corresponding foreground pixels in two images should have the same pixel color:

$$C^r_p - (1 - \alpha_p) B^r_p = C^m_{p+d^f} - (1 - \alpha_p) B^m_{p+d^f}. \tag{12}$$

Thus, we define the probability $P_p(B^{r(n)}, B^{m(n)}, \alpha^{(n)}| d^f, U)$ as the color similarity of corresponding foreground pixels in both input images

$$\begin{aligned}&P_p(B^{r(n)}, B^{m(n)}, \alpha^{(n)}|d^f, U) \\ =\ &exp(-\beta_f \| (C^r_p - (1 - \alpha^{(n)}_p) B^{r(n)}_p) \\ -\ &(C^m_{p+d^f} - (1 - \alpha^{(n)}_p) B^{m(n)}_{p+d^f}) \|^2),\end{aligned} \tag{13}$$

where β_f is a weight.

$P_p(d^f|U)$ models prior probability of d_f from initial input images. In the initialization step to be discussed in Section 4.2, we model all disparities from the foreground and background pixels using two Gaussian distributions respectively. Thus we formulate the probability

$$P_p(d^f|U) = N(d^f; \overline{d^f}, \sigma_{d^f}). \tag{14}$$

where $\overline{d^f}$ and σ_{d^f} are the mean and variance of the foreground disparity Gaussian to be introduced later. The expectation for the disparity value of each foreground pixel p can be written as

$$E(P_p(d^f = d_1|\Theta^{(n)}, U)) = \frac{P_p(d^f = d_1|\Theta^{(n)}, U)}{\sum_{d_i} P_p(d^f = d_i|\Theta^{(n)}, U)} \tag{15}$$

Since we have only a few levels for d_i, the computation of the above formula is easy.

Computing $E((P_p(d^b = d_2|\Theta^{(n)}, U))$. For the background disparities, we can formulate the probability as

$$\begin{aligned}&P_p(d^b|\Theta^{(n)}, U) \\ \propto\ &P_p(d^b|U) P_p(B^{r(n)}, B^{m(n)}, \alpha^{(n)}|d^b, U)\end{aligned} \tag{16}$$

$$\begin{aligned}\propto\ &P_p(d^b|U) \sum_{p'} \{ \frac{P(d^f_{p'} = p + d^b_p - p'|\Theta^{(n)}, U)}{\sum_{p'} P(d^f_{p'} = p + d^b_p - p'|\Theta^{(n)}, U)} \cdot \\ &P_p(B^{r(n)}, B^{m(n)}, \alpha^{(n)}, \alpha^{(n)}_{p'}|d^b, U)\},\end{aligned} \tag{17}$$

where p' denote the pixels from C^r whose foreground matching pixel have the probability to be p's background matching pixel. So for pixel p' with a foreground disparity $d^f_{p'} = p + d^b_p - p'$, we may have

$$\alpha^m_{p+d^b_p} = \alpha^m_{p'+d^f_{p'}} = \alpha^r_{p'}, \tag{18}$$

where $\alpha^m_{p+d^b}$ in C^m is the corresponding alpha value to α^r_p in C^r for the same background pixel. Note here the matching probability $P_p(B^{r(n)}, B^{m(n)}, \alpha^{(n)}, \alpha^{(n)}_{p'}|d^b, U)$ is different from the foreground counterpart in Equation 13 due to the possibility of been occluded for any background pixels. Thus, we define the probability on background color matching adapting to the alpha values:

$$\begin{aligned}&P_p(B^{r(n)}, B^{m(n)}, \alpha^{(n)}, \alpha^{(n)}_{p'}|d^b, U) \\ =\ &exp(-\beta_b (1 - \alpha^{r(n)}_p)[(1 - \alpha^{m(n)}_{p+d^b}) \| B^{r(n)}_p - B^{m(n)}_{p+d^b} \|^2 \\ &+ \alpha^{m(n)}_{p+d^b} P]) \\ =\ &exp(-\beta_b (1 - \alpha^{r(n)}_p)[(1 - \alpha^{r(n)}_{p'}) \| B^{r(n)}_p - B^{m(n)}_{p+d^b} \|^2 \\ &+ \alpha^{r(n)}_{p'} P]),\end{aligned} \tag{19}$$

where β_b is a weight similar to that defined in Equation 13, and P is set to give penalty when the value of $\alpha^{m(n)}_{p+d^b}$ is close to 1. This happens when the background pixel $p + d^b$ is largely occluded by the foreground in image C^m.

To understand the definition of Equation 19, let us analyze two extreme situations. On one extreme, if α^r_p and $\alpha^m_{p+d^b}$ both approach 0, it means that both the corresponding background pixels B^r_p and $B^{m(n)}_{p+d^b}$ are not occluded. Their color differences, with a large probability, measure if

the two pixels are matched. On the other extreme, if either α_p^r or $\alpha_{p+d^b}^m$ approaches 1, one or both background pixels are occluded. Thus, the color difference $||B_p^r - B_{p+d^b}^m||$ is not reliable.

The definition of $P_p(d^b|U)$ is defined in a way similar to Equation 14 using initially estimated Gaussian distribution to be described in Section 4.2:

$$P_p(d^b|U) = N(d^b; \overline{d^b}, \sigma_{d^b}). \quad (20)$$

Integrating the above two probability definition, the expectation on d^b can be computed as

$$E(P_p(d^b = d_2|\Theta^{(n)}, U)) = \frac{P_p(d^b = d_2|\Theta^{(n)}, U)}{\sum_{d_i} P_p(d^b = d_i|\Theta^{(n)}, U)}. \quad (21)$$

4.1.2 Maximization Step

After the expectation computation, we maximize the expected complete-data log-likelihood w.r.t. J given the observation U:

$$\begin{aligned}
&\Theta^{(n+1)} \\
&= \arg\max_{\Theta} \sum_{J \in J^n} P(J|\Theta^{(n)}, U) \log P(\Theta|J, U) \\
&= \arg\max_{\Theta} \sum_{J \in J^n} P(J|\Theta^{(n)}, U) \log P(J, U|\Theta) P(\Theta) \\
&= \arg\max_{\Theta} \sum_{J \in J^n} P(J|\Theta^{(n)}, U)\{L(J, U|\Theta) + L(\Theta)\}, \quad (22)
\end{aligned}$$

where $L(\cdot) = logP(\cdot)$, the log likelihood. $P(J|\Theta^{(n)}, U)$ is computed in the Expectation step. Using Equations 7 and 8, we define

$$\begin{aligned}
&L(J, U|\Theta) \\
&= -\textstyle\sum_{p \in C^r}(\|\alpha_p F_p + (1-\alpha_p)B_p^r - C_p^r\|^2 \\
&+\|\alpha_p F_p + (1-\alpha_p)B_{p+d^f}^m - C_{p+d^f}^m\|^2)/2\sigma_C^2, \quad (23)
\end{aligned}$$

where σ_C is the standard deviation of a Gaussian probability distribution [2]. $L(\Theta)$ is expanded to

$$L(\Theta) \propto L(\alpha) + L(F) + L(B^r) + L(B^m). \quad (24)$$

Similar to the methods proposed to solve the natural image matting problem [2, 18], we estimate the foreground color, alpha value, and background color likelihoods for each pixel by first collecting samples from the neighboring pixels. Then we model these samples using single Gaussian or Gaussian mixtures for background and foreground respectively. In what follows, for simplicity, we describe our method using a single Gaussian model. The formulation and optimization using Gaussian mixtures are similar.

Denoting the constructed Gaussian mean and covariance matrix for foreground color in each pixel p as $\overline{F_p}$ and $\Sigma_{F_p}^{-1}$, we obtain

$$L(F) = \sum_p L(F_p) = \sum_p -(F_p - \overline{F_p})^T \Sigma_{F_p}^{-1}(F_p - \overline{F_p})/2 \quad (25)$$

The definitions of $L(B^r)$, $L(B^m)$ and $L(\alpha)$ are similar.

Given all above definitions of probability, to optimize, we first take partial derivatives on Equation 22 with respect to α for each pixel p, and set them to zero to compute α:

$$\alpha_p^{(n+1)} = \frac{\sum_{d^f} G(\Theta, d^f) P(d^f|\Theta^{(n)}, U)}{\sum_{d^f} H(\Theta, d^f) P(d^f|\Theta^{(n)}, U)}, \quad (26)$$

where

$$\begin{aligned}
G(\Theta, d^f) = \;& \overline{\alpha}/\sigma_{\alpha_p}^2 + (F_p - B_p^r)^T(C_p^r - B_p^r)/\sigma_C^2 \\
&+(F_p - B_{p+d^f}^m)^T(C_{p+d^f}^m - B_{p+d^f}^m)/\sigma_C^2, \\
H(\Theta, d^f) = \;& 1/\sigma_{\alpha_p}^2 + (F_p - B_p^r)^T(F_p - B_p^r)/\sigma_C^2 \\
&+(F_p - B_{p+d^f}^m)^T(F_p - B_{p+d^f}^m)/\sigma_C^2.
\end{aligned}$$

Then we take partial derivatives on Equation 22 with respect to $\{F, B^r, B^m\}$ for each pixel p, and also set them to zero to compute $\{F, B^r, B^m\}$. Denote $p_J = P(J|\Theta^{(n)}, U)$, we get

$$\begin{bmatrix} A_{00} & A_{01} & A_{02} \\ A_{10} & A_{11} & A_{12} \\ A_{20} & A_{21} & A_{22} \end{bmatrix} \begin{bmatrix} F_p \\ B_p^r \\ B_{p+d^f}^m \end{bmatrix} = \begin{bmatrix} M_0 \\ M_1 \\ M_2 \end{bmatrix}, \quad (27)$$

where

$$\begin{aligned}
A_{00} &= \sum_J p_J(2\alpha_p^2 I/\sigma_C^2 + \Sigma_{F_p}^{-1}), \\
A_{11} &= \sum_J p_J((1-\alpha_p)^2 I/\sigma_C^2 + \Sigma_{B_p^r}^{-1}), \\
A_{22} &= \sum_J p_J((1-\alpha_p)^2 I/\sigma_C^2 + \Sigma_{B_{p+d^f}^m}^{-1}), \\
A_{01} &= A_{10} = A_{02} = A_{20} = \sum_J p_J(\alpha_p(1-\alpha_p)I/\sigma_C^2), \\
A_{12} &= A_{21} = \mathbf{0},
\end{aligned}$$

and

$$\begin{aligned}
M_0 &= \sum_J p_J(\alpha_p C_p^r/\sigma_C^2 + \alpha_p C_{p+d^f}^m/\sigma_C^2 + \Sigma_{F_p}^{-1}\overline{F_p}), \\
M_1 &= \sum_J p_J((1-\alpha_p)C_p^r/\sigma_C^2 + \Sigma_{B_p^r}^{-1}\overline{B_p^r}), \\
M_2 &= \sum_J p_J((1-\alpha_p)C_{p+d^f}^m/\sigma_C^2 + \Sigma_{B_{p+d^f}^m}^{-1}\overline{B_{p+d^f}^m}).
\end{aligned}$$

Here I is a 3×3 identity matrix and $\mathbf{0}$ represents a 3×3 matrix containing all zeros. Using the estimated $\alpha^{(n)}$,$F^{(n)}$ and $B^{(n)}$ as an initialization, the above optimization processes on α and $\{F, B^r, B^m\}$ are iteratively performed until convergence.

4.1.3 Computing Final Disparities

After the optimization using the EM described, we obtain the estimated parameters Θ^*. We then form a MRF on images based on Θ^* and compute the final disparities integrating the neighboring smoothness in pixels:

$$E(d^k|U, \Theta^*) = E_d(d^k|U, \Theta^*) + E_s(d^k), \quad (28)$$

(a) (b) (c) (d) (e)

Figure 3. Initialization. (a) The input reference image. (b) The initially computed disparity histogram. The two fitted Gaussians are also shown. (c) Initial trimap computed. (d) Initial alpha values computed using the trimap in (c). (e) Final computed disparity map using our iterative optimization method.

where $k \in \{f, b\}$. $E_s(d^k)$ is the smoothness term defined similar to that in [11], and $E_d(d^k|U, \Theta^*)$ is the data term

$$E_d(d^k|U, \Theta^*) = \sum_p -\log P(d_p^k = d^k|\Theta^*, U). \quad (29)$$

We use Belief Propagation to minimize the energy and compute the final optimal disparities.

4.2. Initialization

Initializing disparity. We initialize a single disparity d_p for each pixel p in images C^r and C^m using a conventional stereo matching method [11]. However, we require two initial disparity values in each pixel for foreground and background respectively. So we first compute the histogram of disparities. Since we assume that there's a distance gap between the background and the foreground, it is possible to partition the histogram into two disjoint segments. For robustness, we fit the histogram into a two-component Gaussian mixture model. The parameters of the two Gaussians for foreground and background are denoted as $\{\overline{d^f}, \sigma_{d^f}\}$ and $\{\overline{d^b}, \sigma_{d^b}\}$ respectively which are also used in Equation 14 and 20. One example is shown in Figure 3 (b). Then we use the Bayes classifier to partition the histogram

$$\begin{cases} d \text{ is in foreground} & N(h(d); \overline{d^f}, \sigma_{d^f}) \geq N(h(d); \overline{d^b}, \sigma_{d^b}) \\ d \text{ is in background} & N(h(d); \overline{d^f}, \sigma_{d^f}) < N(h(d); \overline{d^b}, \sigma_{d^b}) \end{cases}$$

where $h(d)$ the value of the dth bin in the histogram. For each pixel p, if the initialized d_p is classified as the foreground disparity d_p^f, then we set d_p^b to be the background disparity Gaussian mean $\overline{d^b}$. Otherwise, we set d_p^f to $\overline{d^f}$.

$$d_p^f = \begin{cases} d_p & d_p \text{ is in foreground} \\ \overline{d^f} & d_p \text{ is in background} \end{cases} \quad (30)$$

$$d_p^b = \begin{cases} \overline{d^b} & d_p \text{ is in foreground} \\ d_p & d_p \text{ is in background} \end{cases} \quad (31)$$

Initializing alpha matte. We use Bayesian Matting [2] method to solve the matting problem initially on both images. However, this method requires a trimap to indicate whether one pixel in the input images is definitely foreground ($\alpha = 1$), definitely background ($\alpha = 0$), or unknown.

Equation 30 and 31 produce a binary segmentation in input images according to whether $d_p = d_p^f$ or $d_p = d_p^b$. The disparity of the pixels around the segmentation boundaries are obviously unreliable since these pixels are more likely to be mixtures of foreground and background. We then automatically select all these boundary pixels, and dilate them by 2 to 15 pixels to form the final 'unknown' region in the trimap. All other pixels are automatically marked as 'known' in the trimap, as shown in Figure 3(c). Two initial trimaps on C^r and C^m are, thus, created. Based on the trimaps, the foreground $F^{(0)}$, background $B^{(0)}$, and alpha matte $\alpha^{(0)}$ are automatically computed using Bayesian Matting in the two input images. Of course, since the initial matting is performed separately in two images, there are inevitable alpha errors, as shown in Figure 3(d).

5. Experiment Results

We have shown one difficult example in Figure 1. Since each pixel has at most two disparities in our results, only for visualizing the hairy object boundary, we construct the *blended disparity map* similar to the color blending

$$d_p^{show} = \alpha_p d_p^f + (1 - \alpha_p) d_p^b, \quad (32)$$

which has already been used in Figure 1 (d) and 3 (e).

Figure 4 shows another difficult example where two stereo images contain a toy bear with long hair. (a) and (b) are two input images. (c) is the disparity result using the method in [11], which obviously causes errors around the object boundary. (d) and (e) are our *blended disparity map* and alpha matte through optimization. The complex alpha structure is preserved.

Our approach can also be applied to the traditional stereo image pairs to improve the object details. We show the "Tsukuba" example in figure 5. In our experiments, the lamp is automatically segmented as the foreground objects since it has largest disparities. We show our optimized alpha matte and the extracted foreground in (b) and (c) respectively. Note that the boundary of the extracted lamp is smooth and natural. Using the optimized alpha matte, we compute the disparities and compare them with those gen-

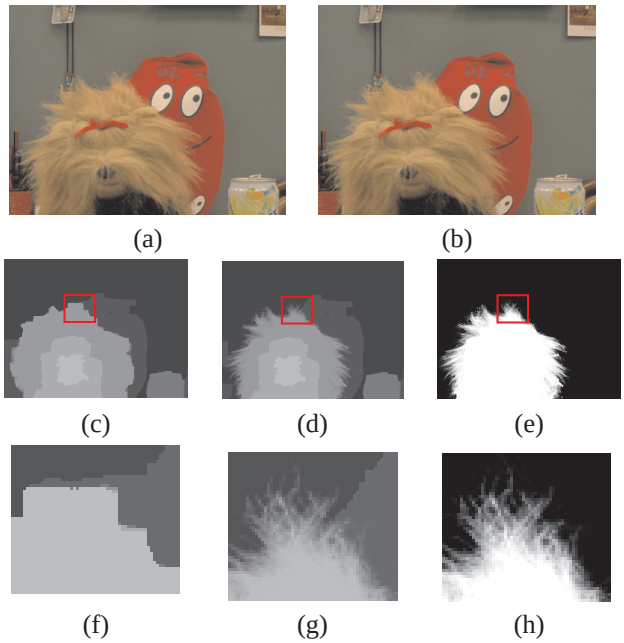

Figure 4. Bear example. (a) and (b) The input stereo images. (c) Stereo matching result using method in [11]. (d) The *blended disparity map* computed from our method. The structures are well preserved. (e) The alpha matte computed in our approach. (f)-(h) Magnified regions in (c), (d), and (e).

erated in [3] and [11] in 5 (d-f) using the following formula to produce a single disparity for each pixel:

$$d_p^{refine} = \begin{cases} d_p^f & \alpha_p \geq 0.5 \\ d_p^b & \alpha_p < 0.5 \end{cases} \quad (33)$$

Obviously, our result has clearer boundary of the lamp.

Besides, our method can also produce better matting results comparing to previous single natural image matting methods. In figure 6, we compare our result with two state-of-art natural image matting methods [18, 9] on the difficult "fan" example. The background has complex patterns and similar colors as the foreground, which make the foreground and background color estimation unstable. In (b) and (c), it is observable that the background patterns are mistakenly estimated as the foreground. Our result in (d) has less errors in the alpha matte thanks to the stereo configuration and the joint optimization.

6. Conclusion and Discussion

In this paper, we have proposed a novel approach to solve the stereo matching problem on objects with fractional boundary using two-frame narrow-band stereo images. Each pixel, with the definition of the layer blending, is assumed to be blended by two latent pixels with different disparities. We have defined a probabilistic model constraining the colors, disparities, as well as the alpha mattes on the two input images, and designed an optimization method using Expectation-Maximization to robustly estimate all parameters.

In discussion, our method has achieved large improvement in handling general boundary transparencies in stereo matching using an image pair. Our method currently can separate two layers, i.e., background and foreground. We expect that if more stereo images or other image information are given, our model can be extended to handle more depth layers. This will be our future work.

References

[1] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *TPAMI*, 23(11):1222–1239, 2001. 2

[2] Y. Y. Chuang, B. Curless, D. H. Salesin, and R. Szeliski. A bayesian approach to digital matting. *CVPR*, 2001. 2, 5, 6

[3] Y. Deng, Q. Yang, X. Lin, and X. Tang. A symmetric patch-based correspondence model for occlusion handling. *ICCV*, 2005. 2, 7, 8

[4] P. F. Felzenszwalb and D. P. Huttenlocher. Efficient belief propagation for early vision. *CVPR*, 1:261–268, 2004. 2

[5] S. W. Hasinoff, S. B. Kang, and R. Szeliski. Boundary matting for view synthesis. *IEEE Workshop on Image and Video Registration*, 2004. 2

[6] L. Hong and G. Chen. Segment-based stereo matching using graph cuts. *CVPR*, 1:74–81, 2004. 2

[7] N. Joshi, W. Matusik, and S. Avidan. Natural video matting using camera arrays. *SIGGRAPH*, 2006. 2

[8] V. Kolmogorov and R. Zabih. Computing visual correspondence with occlusions using graph cuts. *ICCV*, 2001. 2

[9] A. Levin, D. Lischinski, and Y. Weiss. A closed form solution to natural image matting. *CVPR*, 2006. 2, 7, 8

[10] J. Sun, J. Jia, C. Tang, and H. Shum. Poisson matting. *SIGGRAPH*, 2004. 2

[11] J. Sun, Y. Li, S. B. Kang, and H.-Y. Shum. Symmetric stereo matching for occlusion handling. *CVPR*, 2005. 1, 2, 6, 7, 8

[12] J. Sun, N. N. Zheng, and H. Y. Shum. Stereo matching using belief propagation. *TPAMI*, 25(7):787–800, 2003. 2

[13] R. Szeliski and P. Golland. Stereo matching with transparency and matting. *IJCV*, 32(1):45–61, 1999. 1, 2

[14] R. Szeliski, D. Scharstein, and R. Zabih. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *IEEE Workshop on Stereo and Multi-Baseline Vision*, 2001. 2

[15] F. Tappen and W. T. Freeman. Comparison of graph cuts with belief propagation for stereo, using identical mrf parameters. *ICCV*, 2:900–906, 2003. 2

[16] E. Trucco, A. Fusiello, and A. Verri. Rectification with unconstrained stereo geometry. *BMVC*, 1997. 2

[17] Y. Tsin, S. B. Kang, and R. Szeliski. Stereo matching with reflections and translucency. *CVPR*, 2003. 1, 2

[18] J. Wang and M. Cohen. An iterative optimization approach for unified image segmentation and matting. *ICCV*, 2005. 2, 5, 7, 8

[19] Y. Wexler, A. Fitzgibbon, and A. Zisserman. Bayesian estimation of layers from multiple images. *ECCV*, 2002. 2

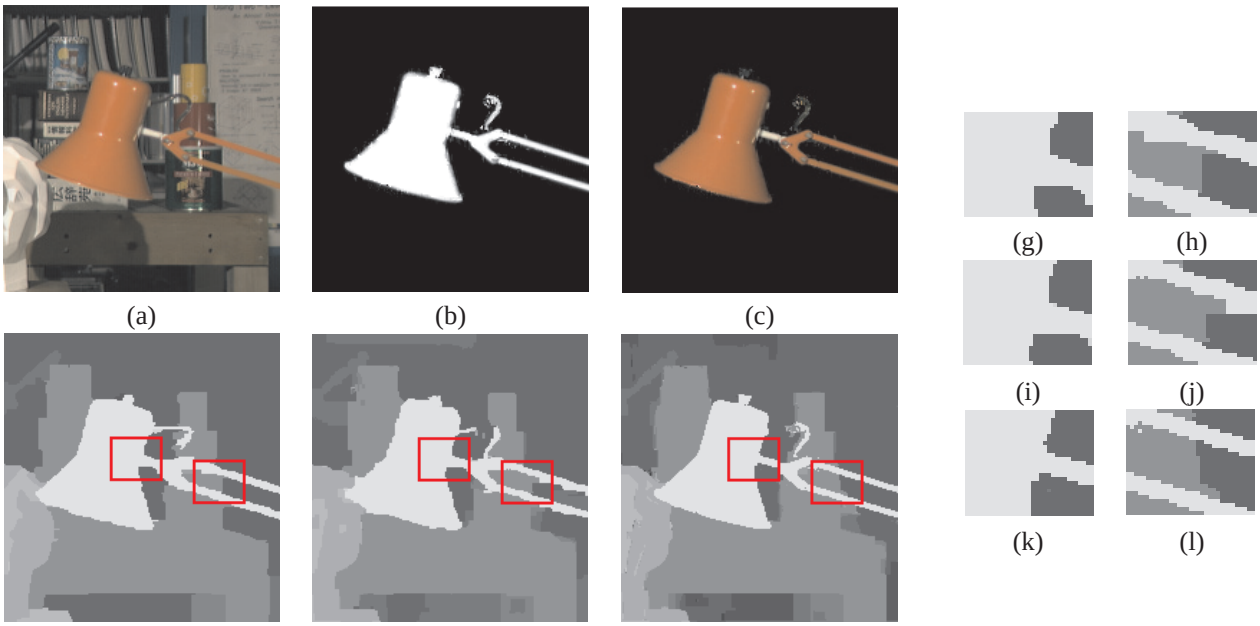

Figure 5. The lamp from the stereo image pair "Tsukuba". (a) Input reference image. (b) The alpha matte of the foreground lamp computed from our method. The boundary is natural and smooth. (c) The extracted foreground. (d) Result from the patch-based method [3]. (e) Result of symmetric stereo matching [11]. (f) Our optimized disparity map. The lamp boundary has large improvement comparing to (d) and (e). (g)-(l) Side-by-side comparison on the magnified regions.

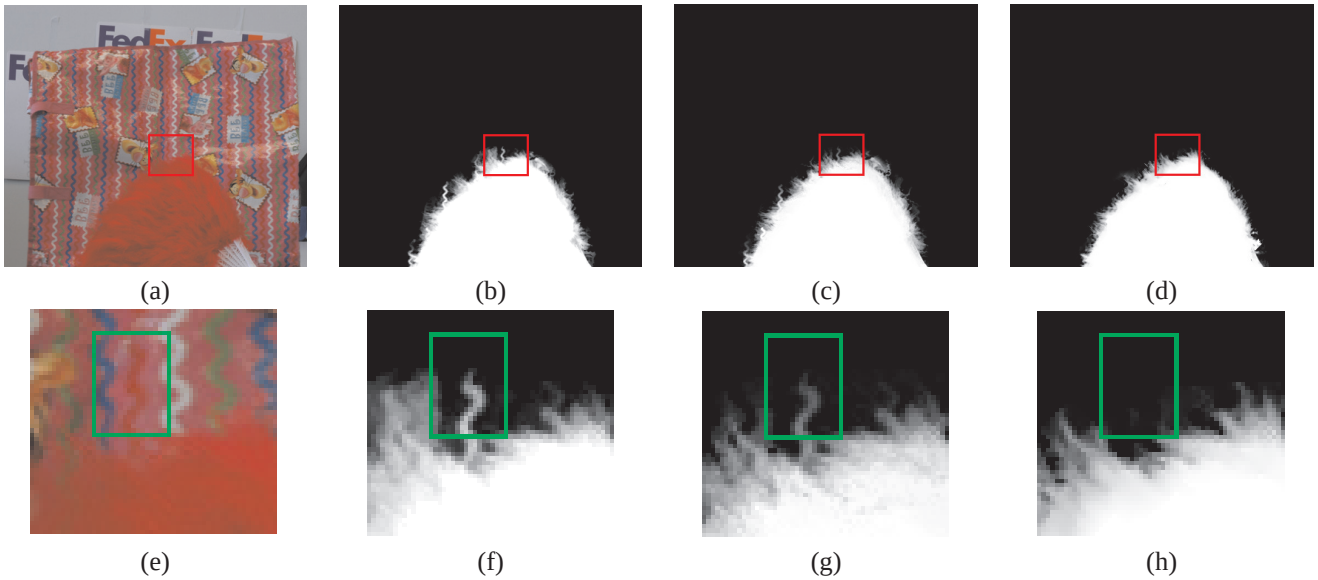

Figure 6. Comparison of the alpha matte. (a) Input reference image. The background and foreground have similar colors. The patterns of the background are also complex. (b) Result from the method in [18]. (c) Results from the method in [9]. (d) Our method is automatic, and does not require any user input. (e)-(h) The magnified regions for comparison. Notice that, within the green rectangles, results (f) and (g) mistakenly take the background pattern into foreground while our method produces a satisfactory alpha matte.

[20] Q. Yang, L. Wang, R. Yang, H. Stewenius, and D. Nister. Stereo matching with color-weighted correlation, hierarchical belief propagation and occlusion handling. *CVPR*, 2006. 1

[21] C. L. Zitnick, N. Jojic, and S. B. Kang. Consistent segmentation for optical flow estimation. *ICCV*, 2005. 2

[22] C. L. Zitnick, S. B. Kang, M. Uyttendaele, S. Winder, and R. Szeliski. High-quality video view interpolation using a layered representation. *SIGGRAPH*, 2004. 2