

# Image Matting Using Linear Optimization

Shifeng Chen<sup>1</sup>, Zhenguo Li<sup>1</sup>, Jianzhuang Liu<sup>1</sup>, Xiaoou Tang<sup>1,2</sup>

<sup>1</sup>Dept. of IE, The Chinese University of Hong Kong, Hong Kong

{sfchen5, zgli5, jzliu}@ie.cuhk.edu.hk

<sup>2</sup>Microsoft Research Asia, Beijing, China

xitang@microsoft.com

## ABSTRACT

An image can be assumed to be a composite of the foreground and the background. The foreground and the background of each pixel are linearly combined in terms of this pixel's foreground opacity (called alpha). Image matting is the process of estimating the foreground, the background and the alpha for each pixel. In this paper, we transform the ill-posed image matting problem into two over-determined linear optimization problems by introducing two medium variables and imposing smoothness constraints. Closed form solutions can be obtained from the two problems. Extensive experimental results indicate that our algorithm can generate high-quality matting results.

## Categories and Subject Descriptors

I.4 [Image Processing and Computer Vision]: Miscellaneous

## General Terms

Algorithms, Experimentation

## Keywords

Image Matting, Closed Form Solution, Linear Optimization

## 1. INTRODUCTION

Image matting is an important technique in image and video editing, which was originally developed for film and video production [3]. The task of image matting is to extract the foreground object from an image by estimating a color and an opacity for each pixel of the foreground object. The opacity of a pixel  $i$  is called the *alpha* ( $\alpha_i$ ) and the whole opacity map is referred to as the *alpha matte* ( $\alpha$ ).

Formally, a color image  $\mathbf{I}$  can be modeled as the composite of a foreground image  $\mathbf{F}$  and a background image  $\mathbf{B}$ , where  $\mathbf{I}$ ,  $\mathbf{F}$ , and  $\mathbf{B}$  all have three RGB color channels, denoted by  $(I_r, I_g, I_b)$ ,  $(F_r, F_g, F_b)$ , and  $(B_r, B_g, B_b)$ , respectively. Based on the *alpha channel* presented by Porter and

Duff [8], the most common compositing equation for the  $i$ th pixel in the image is expressed as:

$$\mathbf{I}_i = \alpha_i \mathbf{F}_i + (1 - \alpha_i) \mathbf{B}_i, \quad (1)$$

where  $\mathbf{I}_i = (I_{ri}, I_{gi}, I_{bi})$ ,  $\mathbf{F}_i = (F_{ri}, F_{gi}, F_{bi})$ , and  $\mathbf{B}_i = (B_{ri}, B_{gi}, B_{bi})$  are the  $i$ th pixel's composite, foreground color, and background color, respectively, and  $\alpha_i$  is the opacity of the  $i$ th pixel. The value of  $\alpha_i$  is within the interval  $[0, 1]$ . The task of image matting is to find  $\mathbf{F}_i$ ,  $\mathbf{B}_i$ , and  $\alpha_i$  for given  $\mathbf{I}_i$ . Obviously, for natural image matting, this problem is under-constrained since at each pixel we have 3 equations from the RGB channels with 7 unknowns. To pull a matte accurately, additional constraints are necessary.

Usually, an input image is labeled manually as three parts before matte pulling, including the definite foreground, the definite background, and the unknown region. These three parts combined is called the *trimap*. For a given trimap, the matte pulling is carried out in the unknown region using information taken from the definite foreground and background. Some recent matting approaches [13], [6] start from a few scribbles indicating a small number of foreground and background pixels, and then estimate the 7 unknowns at every pixel in the unknown region.

The early matting approaches are based on known background. The *blue screen matting* [11], [7] approaches try to simplify the matting problem by constructing  $\mathbf{F}$  and  $\alpha$  with  $\mathbf{B}$  known (typically blue and green). The *difference matting* [9] is similar to the blue screen matting, which requires pre-recording a background image which may be a complex image without any foreground object. Then  $\alpha$  is determined by taking the difference between  $\mathbf{B}$  and  $\mathbf{I}$ , and comparing the difference to a threshold. Although the blue screen matting and difference matting are successful in many applications, one limitation is the need of the background from a user-controlled environment.

Recently, many natural image matting approaches are presented, which pull a matte from an image with arbitrary background. The Knockout [1] algorithm starts from the known foreground and background of the trimap and extrapolates the known foreground and background colors into the unknown region to estimate  $\alpha$ . Ruzon and Tomasi's method [10] analyzes the color samples of the foreground and background by the mixture Gaussian distribution and uses them to estimate  $\alpha$ . Hillman et al. [5] use the principal components analysis (PCA) to analyze the foreground and background samples. The Bayesian matting [2] improves Ruzon and Tomasi's method, in which the color samples of  $\mathbf{F}$  and  $\mathbf{B}$  are clustered and modeled as mixture Gaussians. A maxi-

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

MM'07, September 23–28, 2007, Augsburg, Bavaria, Germany.

Copyright 2007 ACM 978-1-59593-701-8/07/0009 ...\$5.00.

Maximum a posteriori (MAP) estimation is then used to calculate  $\mathbf{F}$ ,  $\mathbf{B}$ , and  $\alpha$  simultaneously for each pair of the foreground and background in a Bayesian framework. The final  $\alpha$  is chosen from the pair of the background and foreground giving the maximum likelihood. The Poisson matting proposed by Sun et al. [12] formulates the natural image matting problem as one that solves the Poisson equation with the matte gradient field and the Dirichlet boundary conditions.

The above natural image matting approaches need a known trimap. The methods proposed by Wang and Cohen [13] and Levin et al. [6] use a scribble-based interface. Both start from a few scribbles indicating a small amount of background and foreground pixels. The former uses belief propagation to iteratively estimate the unknowns at every pixel, while the latter proposes a spectral analysis method to do the work.

In this paper, we present a novel matting approach that converts the image matting problem into a simple linear optimization problem and obtains a closed form solution. Like some existing algorithms [1], [2], our approach also handles the problem in two steps. First,  $\mathbf{F}$  and  $\mathbf{B}$  are estimated, and then  $\alpha$  is calculated using  $\mathbf{F}$  and  $\mathbf{B}$ . In our method, with two sets of medium variables  $\mathbf{X}$  and  $\mathbf{X}'$  introduced,  $\mathbf{X}$ ,  $\mathbf{X}'$ ,  $\mathbf{F}$ , and  $\mathbf{B}$  are estimated simultaneously by solving a system of linear equations using the least square error (LSE) estimation, and then  $\alpha$  is obtained with these estimated  $\mathbf{X}$ ,  $\mathbf{X}'$ ,  $\mathbf{F}$ , and  $\mathbf{B}$  by solving another system of linear equations also using the LSE estimation. In both steps, closed form solutions are obtained.

Extensive experimental results have demonstrated that our algorithm can obtain high-quality image matting. We also give comparative experimental results obtained by our algorithm and other four recent algorithms. In most cases, the five algorithms present comparable results, but in many cases, our algorithm outperforms the others.

## 2. FORMULATION

Our approach to image matting is to formulate it as a linear optimization problem, from which a closed form solution can be obtained. From (1), obviously we cannot obtain  $\mathbf{F}$ ,  $\mathbf{B}$ , and  $\alpha$  using a linear optimization. Define two sets of medium variables

$$\mathbf{X}_i = \alpha_i(\mathbf{F}_i - \mathbf{B}_i), \quad \mathbf{X}'_i = (1 - \alpha_i)(\mathbf{B}_i - \mathbf{F}_i). \quad (2)$$

Then the compositing equation becomes

$$\mathbf{I}_i = \mathbf{X}_i + \mathbf{B}_i, \quad \mathbf{I}_i = \mathbf{X}'_i + \mathbf{F}_i. \quad (3)$$

We can see that if  $\mathbf{X}$ ,  $\mathbf{X}'$ ,  $\mathbf{F}$ , and  $\mathbf{B}$  are available, from (2),  $\alpha$  can be obtained immediately. Therefore, the matting problem is divided into two steps: to find  $\mathbf{X}$ ,  $\mathbf{X}'$ ,  $\mathbf{F}$ , and  $\mathbf{B}$  by solving (3) with a smoothness constraint, and to calculate  $\alpha$  using (2). Here one of the equations in (2) seems redundant if the other one is defined. Why both are used will be explained in Section 2.2.

### 2.1 Calculating $\mathbf{X}$ , $\mathbf{X}'$ , $\mathbf{F}$ , and $\mathbf{B}$

From (3) we cannot obtain  $\mathbf{X}_i$ ,  $\mathbf{X}'_i$ ,  $\mathbf{F}_i$ , and  $\mathbf{B}_i$  since there are 12 unknowns in the 6 equations for a color image. Therefore, we enforce a smoothness constraint on  $\mathbf{F}$  and  $\mathbf{B}$  like some other methods [2], [12]. Let  $P$  denote the set of the pixels of the unknown region and  $W_i$  be a window of size  $n \times n$  centered at pixel  $i$ . By enforcing the smoothness assumption on  $\mathbf{F}$  and  $\mathbf{B}$ , for each color channel  $c = r, g, \text{ or } b$ ,

our goal is to minimize the following cost function:

$$J(X_c, X'_c, B_c, F_c) = J_1(X_c, B_c) + J_2(X'_c, F_c), \quad (4)$$

where

$$J_1 = \sum_{i \in P} \left( (I_{ci} - (X_{ci} + B_{ci}))^2 + \sum_{j \in W_i} (B_{ci} - B_{cj})^2 \right), \quad (5)$$

$$J_2 = \sum_{i \in P} \left( (I_{ci} - (X'_{ci} + F_{ci}))^2 + \sum_{j \in W_i} (F_{ci} - F_{cj})^2 \right). \quad (6)$$

The first terms on the right hand sides of  $J_1$  and  $J_2$  are from (3), and the second terms are the smoothness constraints, where  $I_{ci}$  is known and for  $j \notin P$ ,  $B_{cj}$  and  $F_{cj}$  are also known.

It is clear that every term in (5),  $(I_{ci} - (X_{ci} + B_{ci}))^2$  or  $(B_{ci} - B_{cj})^2$ , can be rewritten in this form:

$$\left( \mathbf{a}^T \mathbf{z} - y \right)^2, \quad (7)$$

where  $\mathbf{z} = [\mathbf{x}_c, \mathbf{b}_c]^T$ ,  $\mathbf{x}_c$  (or  $\mathbf{b}_c$ ) represents the values of  $X_c$  (or  $B_c$ ) in the unknown region in vector form, and  $\mathbf{a} = (a_1, a_2, \dots, a_{2|P|})^T$ . More specifically, for  $(I_{ci} - (X_{ci} + B_{ci}))^2$ , we have

$$\begin{aligned} a_i &= 1 \text{ (for } X_{ci}), \\ a_{i+|P|} &= 1 \text{ (for } B_{ci}), \\ a_t &= 0, t \in \{1, 2, \dots, 2|P|\} \setminus \{i, i + |P|\}, \\ y &= I_{ci}; \end{aligned}$$

for  $(B_{ci} - B_{cj})^2$ ,

$$\begin{aligned} a_{i+|P|} &= 1 \text{ (for } B_{ci}), \\ a_{j+|P|} &= \begin{cases} -1, & \text{if } j \in P \setminus \{i\} \\ 0, & \text{if } j \notin P \end{cases} \text{ (for } B_{cj}), \\ a_t &= 0, t \in \{1, 2, \dots, 2|P|\} \setminus \{i + |P|, j + |P|\}, \\ y &= \begin{cases} 0, & \text{if } j \in P \setminus \{i\} \\ B_{cj}, & \text{if } j \notin P \end{cases}. \end{aligned}$$

Then  $J_1(X_c, B_c)$  in (5) can be rewritten in matrix form:

$$J_1(\mathbf{z}) = \|\mathbf{A}\mathbf{z} - \mathbf{y}\|^2, \quad (8)$$

where  $\mathbf{A}$  is a matrix of size  $n^2|P| \times 2|P|$  whose elements are the  $a_i$ 's given above, and  $\mathbf{y}$  is a vector of size  $n^2|P|$  whose elements are the  $y$ 's.

To minimize  $J_1$ , we take the derivative of  $J_1$  with respect to  $\mathbf{z}$  and set the derivative to  $\mathbf{0}$ :

$$\frac{\partial J_1}{\partial \mathbf{z}} = 2\mathbf{A}^T \mathbf{A}\mathbf{z} - 2\mathbf{A}^T \mathbf{y} = \mathbf{0}.$$

Therefore, we have

$$\mathbf{z} = [\mathbf{x}_c, \mathbf{b}_c]^T = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \mathbf{y} = \mathbf{A}^+ \mathbf{y}. \quad (9)$$

If  $\mathbf{A}$  is a square matrix of full rank,  $\mathbf{A}^+ = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T = \mathbf{A}^{-1}$  is the inverse of  $\mathbf{A}$ ; otherwise,  $\mathbf{A}^+$  is the pseudo inverse of  $\mathbf{A}$ .

Similarly,  $J_2(X'_c, F_c)$  can be rewritten in matrix form:

$$J_2(\mathbf{z}') = \|\mathbf{A}'\mathbf{z}' - \mathbf{y}'\|^2, \quad (10)$$

where  $\mathbf{z}' = [\mathbf{x}'_c, \mathbf{f}_c]^T$ , and  $\mathbf{x}'_c$  (or  $\mathbf{f}_c$ ) represents the values of  $X'_c$  (or  $F_c$ ) in the unknown region in vector form. We thus have

$$\mathbf{z}' = [\mathbf{x}'_c, \mathbf{f}_c]^T = \mathbf{A}'^+ \mathbf{y}', \quad (11)$$

where  $A'^+ = (A'^T A')^{-1} A'^T$ . Since  $J_1$  and  $J_2$  are independent,  $\mathbf{x}_c$ ,  $\mathbf{x}'_c$ ,  $\mathbf{b}_c$ , and  $\mathbf{f}_c$  are also the closed form solution to minimizing  $J(X_c, X'_c, B_c, F_c)$ .

By examining  $J_1$  and  $J_2$  in (5), (6), (8), and (10), we can see that the above solution can also be obtained by solving the following system of over-determined linear equations using the LSE estimation:

$$X_{ci} + B_{ci} = I_{ci}, B_{ci} - B_{cj} = 0, \quad i \in P, j \in W_i \setminus \{i\}, \quad (12)$$

$$X'_{ci} + F_{ci} = I_{ci}, F_{ci} - F_{cj} = 0, \quad i \in P, j \in W_i \setminus \{i\}. \quad (13)$$

## 2.2 Calculating $\alpha$

From (2), given  $\mathbf{X}$ ,  $\mathbf{X}'$ ,  $\mathbf{F}$ , and  $\mathbf{B}$ , we have:

$$\tilde{\alpha}_i = \frac{\mathbf{X}_i}{\mathbf{F}_i - \mathbf{B}_i}, \quad \tilde{\alpha}_i = \mathbf{1} - \frac{\mathbf{X}'_i}{\mathbf{B}_i - \mathbf{F}_i}, \quad (14)$$

where  $\tilde{\alpha}_i$  is a vector of size 3, whose elements are computed alpha values for R, G, and B channels, and  $\mathbf{1} = (1, 1, 1)^T$ . Conventionally, the alpha values are the same for the three channels at each pixel. Here  $\tilde{\alpha}_i$  is considered as a medium variable, from which with additional constraints, the same alpha value  $\alpha_i$  for pixel  $i$  can be achieved.

Combining the two equations in (14) results in

$$\tilde{\alpha}_i = \frac{1}{2} \left( \mathbf{1} - \frac{\mathbf{X}'_i}{\mathbf{B}_i - \mathbf{F}_i} + \frac{\mathbf{X}_i}{\mathbf{F}_i - \mathbf{B}_i} \right). \quad (15)$$

Let

$$\tilde{\alpha}_{ci} = \frac{1}{2} \left( \mathbf{1} - \frac{X'_{ci}}{B_{ci} - F_{ci}} + \frac{X_{ci}}{F_{ci} - B_{ci}} \right), \quad c = r, g, b.$$

Then the mean and variance of the components of  $\tilde{\alpha}_i$  at pixel  $i$  are  $\bar{\alpha}_i = \frac{1}{3} \sum_c \tilde{\alpha}_{ci}$  and  $\sigma_i^2 = \frac{1}{3} \sum_c (\tilde{\alpha}_{ci} - \bar{\alpha}_i)^2$ , respectively. It seems that we have been able to solve the matting problem by choosing  $\alpha_i = \bar{\alpha}_i$ . However, this scheme cannot give satisfactory results. For example, suppose that  $\tilde{\alpha}_{ri} = 0.5$ ,  $\tilde{\alpha}_{gi} = 1.0$ , and  $\tilde{\alpha}_{bi} = 0.1$  at pixel  $i$ . Then  $\bar{\alpha}_i$  is unreliable because  $\sigma_i^2$  is too large. In another case, if  $\tilde{\alpha}_{ri} = 0.52$ ,  $\tilde{\alpha}_{gi} = 0.50$ , and  $\tilde{\alpha}_{bi} = 0.48$ , then  $\bar{\alpha}_i$  is reliable. Our experimental results show that in regions where  $\sigma_i^2$ 's are large, setting  $\alpha_i = \bar{\alpha}_i$  gives unnatural matting results.

Now our goal is to minimize the following criterion:

$$J'(\alpha) = \sum_{i \in P} \left( (e^{-\mu_1 \sigma_i^2} (\alpha_i - \bar{\alpha}_i))^2 + \sum_{j \in W_i} (\mu_2 (\alpha_i - \alpha_j))^2 \right), \quad (16)$$

where  $\mu_1$  and  $\mu_2$  are two factors. In the first term of  $J'(\alpha)$ , large (small)  $\sigma_i^2$  allows more (less) deviation of  $\alpha_i$  from  $\bar{\alpha}_i$ . The second term of  $J'(\alpha)$  enforces a smoothness constraint on  $\alpha$ . The two factors  $\mu_1$  and  $\mu_2$  are used to balance the two terms.

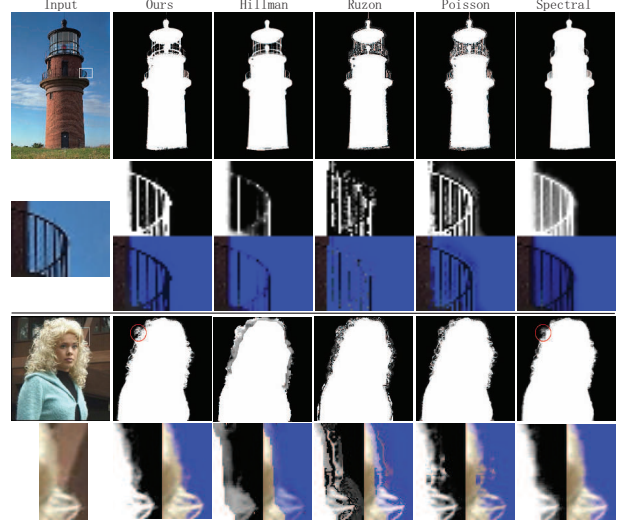
$J'(\alpha)$  in (16) is similar to  $J_1(X_c, B_c)$  in (5). It can also be rewritten in matrix form:

$$J'(\alpha) = \|D\alpha - \mathbf{m}\|^2, \quad (17)$$

where  $\alpha$  represents the values of  $\alpha$  in the unknown region in vector form,  $D$  is a matrix of size  $n^2|P| \times |P|$ , and  $\mathbf{m}$  is a vector of size  $|P|$ . The derivation of  $D$  and  $\mathbf{m}$  is similar to that of  $A$  and  $\mathbf{y}$  in Section 2.1. By minimizing  $J'(\alpha)$ , we have

$$\alpha = D^+ \mathbf{m}, \quad (18)$$

where  $D^+ = (D^T D)^{-1} D^T$ .



**Figure 1: Some comparative results by the five algorithms. The first column includes two input images and two zoomed-in regions bounded by the white boxes in the input images. The results for the two input images are the alpha mattes. The results for the zoomed-in parts include the alpha mattes and the composed images with blue background.**

Again, this result is the same as the result of the LSE estimation of the following over-determined linear equations:

$$e^{-\mu_1 \sigma_i^2} \alpha_i = e^{-\mu_1 \sigma_i^2} \bar{\alpha}_i, \quad i \in P, \quad (19)$$

$$\mu_2 \alpha_i - \mu_2 \alpha_j = 0, \quad i \in P, \quad j \in W_i \setminus \{i\}. \quad (20)$$

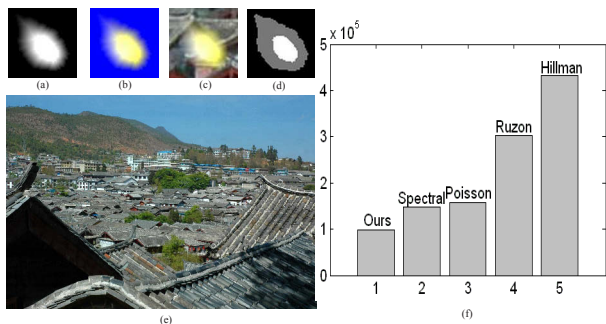
The matrices of  $A$  in Section 2.1 and  $D$  in Section 2.2 are of large size, which may cause the out-of-memory problem in implementation when  $|P|$  is large. Fortunately, these matrices are sparse and there are many techniques developed for sparse matrix computation [4]. Let us take  $A$  for example. It is not difficult to see that the density of nonzero elements in  $A$  is smaller than  $1/|P|$ .

## 3. EXPERIMENTAL RESULT

To demonstrate the performance of our algorithm, we first test it on some images frequently used in image matting literature. We also compare our algorithm to Hillman's [5], Ruzon and Tomasi's [10], Poisson [12], and the spectral analysis [6] methods. The trimaps are the same for all algorithms. In our algorithm, the parameters  $\mu_1$  and  $\mu_2$  are set to 100 and 0.01, respectively.

Figure 1 shows some results in which our algorithm outperforms the others. For the second input image, the spectral method obtains a result with quality similar to ours in the rectangle region, but it performs not as well as ours in some other regions, such as the one inside the red circle.

To provide a quantitative evaluation for the five algorithms, we test them on synthesized images. First, an alpha matte is simulated as the ground truth (Figure 2(a)). Then an foreground image is taken from an real fire image. The composed image of this foreground, the simulated alpha, and blue background is shown in Figure 2(b). In each experiment, the background is a sub-image randomly taken from a large image (Figure 2(e)). Figure 2(c) gives an example of the composed image. The summed absolute error



**Figure 2: Quantitative evaluation of the algorithms.** (a) Simulated alpha. (b) Composed image of a fire foreground, the simulated alpha, and blue background. (c) Composed image with a background taken from (e). (d) Trimap. (e) Large image to generate the background. (f) Total errors by the five algorithms.

between the computed matte and the ground truth is used to compare the performances of the algorithms. Figure 2(f) shows the total errors obtained by the five algorithms after 2000 experiments, from which we can see that our algorithm obtains the minimum error. Due to space limitation, we cannot show many comparative results. In most cases, the five algorithm give comparable results. However, in many cases, our algorithm performs best.

Figure 3 gives other results by our algorithm. Its goal is to pull the mattes from the input images and compose the foreground with new background images. From the result we can see that the new composed images look quite natural based on the high-quality mattes.

## 4. CONCLUSION

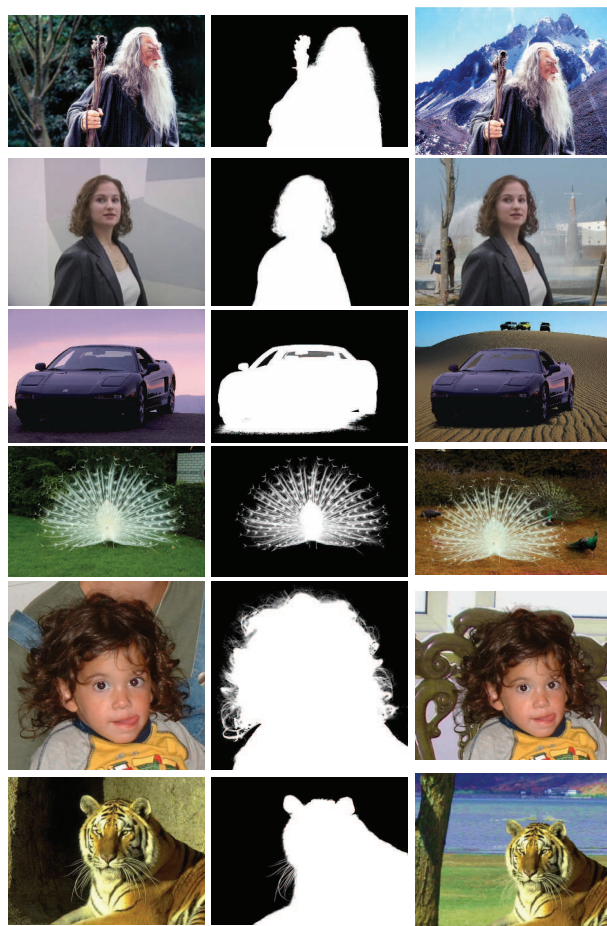
In this paper, we have proposed a novel algorithm to transform the ill-posed image matting problem into two over-determined optimization problems. From these two problems, closed-form solutions can be obtained by enforcing the smoothness constraints and introducing two medium variables. Our algorithm includes two steps. In the first step, the foreground and the background are estimated and in the second step, the matte is pulled with the estimated foreground and background. Each step can be solved using linear optimization. The experimental results demonstrate that our algorithm can generate high-quality image matting.

## Acknowledgement

This work was supported by the Research Grants Council of the Hong Kong SAR (Project No. CUHK 414306) and the CUHK Direct Grant.

## 5. REFERENCES

- [1] A. Berman, A. Dadourian, and P. Vlahos. Method for removing from an image the background surrounding a selected object. *US Patent*, 6(134):346, 2000.
- [2] Y. Chuang, B. Curless, D. Salesin, and R. Szeliski. A Bayesian approach to digital matting. *CVPR*, pages 264–271, 2001.
- [3] R. Fielding. *Techniques of Special Effects of Cinematography*. Focal Press, 1985.



**Figure 3: Some results of alpha matte obtained by our algorithm and new composed images.** From left to right: the input images, the alpha mattes, and the composed images with new background images.

- [4] J. Gilbert, C. Moler, and R. Schreiber. Sparse matrices in MATLAB: Design and implementation. *SIAM J. Matrix Anal. Appl.*, 13(1):333–356, 1992.
- [5] P. Hillman, J. Hannah, and D. Renshaw. Alpha channel estimation in high resolution images and image sequences. *CVPR*, pages 1063–1068, 2001.
- [6] A. Levin, D. Lischinski, and Y. Weiss. A closed form solution to natural image matting. *CVPR*, 2006.
- [7] Y. Mishima. Soft edge chroma-key generation based upon hexoctahedral color space. *US Patent*, 5, 1993.
- [8] T. Porter and T. Duff. Compositing digital images. *SIGGRAPH*, pages 253–259, 1984.
- [9] R. Qian and M. Sezan. Video background replacement without a blue screen. *ICIP*, 4, 1999.
- [10] M. Ruzon and C. Tomasi. Alpha estimation in natural images. *CVPR*, 1:18–25, 2000.
- [11] A. Smith and J. Blinn. Blue screen matting. *SIGGRAPH*, pages 259–268, 1996.
- [12] J. Sun, J. Jia, C. Tang, and H. Shum. Poisson matting. *SIGGRAPH*, 23(3):315–321, 2004.
- [13] J. Wang and M. Cohen. An Iterative Optimization Approach for Unified Image Segmentation and Matting. *ICCV*, 2, 2005.