A FAST APPROACH FOR NATURAL IMAGE MATTING USING STRUCTURE INFORMATION

Qianhui Ning[1] Weiqiang Wang[1] Caifeng Zhu[1] Laiyun Qing[1] Qingming Huang[1, 2]

[1]Graduate School of Chinese Academy of Sciences, Beijing 100080
[2]Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100080
{qhning, wqwang, cfzhu, lyqing, qmhuang}@jdl.ac.cn

ABSTRACT

Natural image matting plays an important role in image and video editing. It has been addressed hotly because it is inherently an under-constrained problem – we must estimate both the foreground and background colors (we call them component colors) at each pixel to calculate its opacity, according to the only known observed color. Prior assumption such as statistics and smoothness are utilized for estimation, but these methods either are simple to handle situations such as complex background and large interaction region or have high computational complexity.

This paper proposes a fast technique to estimate the component colors based on structure information. Our approach exploits a simple convolution operation to detect structure information in images. It then uses two kinds of estimation methods to propagate colors based on the structure types. Experimental results show that our method is fast and efficient to handle objects with strong structures and large part of interaction region with the background

1. INTRODUCTION

Image matting is an important technique in image and video postproduction. Formally, an observed image C is considered as a composite of a foreground image F and a background image B under an opacity matte α . For a pixel $C(i,j)$ at location (i,j), the matting equation is

$$C(i,j)=\alpha_{ij}F(i,j)+(1-\alpha_{ij})B(i,j), \qquad (1)$$

where α_{ij} denotes the opacity at location (i,j) in image C , $\alpha_{ij}\in[0,1]$. In traditional movie industry, editors generally simplify the problem through setting a single background color, and this technique is called "blue screen matting". Smith et al. [1] have given a well study on this problem.

Recently, many researchers have worked on the matting technique for natural images. It is a challenging task to extract objects from complex unknown background. General solution to this issue usually includes three steps. First, hint information of foreground and background are obtained through user interaction. As a result, the observed image is partitioned into foreground, background and uncertain regions. Then some prior constraints are exploited to estimate F and B for each pixel in uncertain regions. Finally, α is calculated according to eq.(1). Apparently, color estimation in the second step is the core technique for natural image matting.

1.1. Previous work

The existing color estimation methods can be classified into two categories: (1) methods that employ local color statistics information; and (2) methods that utilize smoothness assumptions. In first category, the component colors of a pixel are considered as derived from the color models of the local region at which it centered. The models are calculated by sampling in a local window which includes both foreground and background pixels. Berman et al. [2] estimated component color as a weighted sum of the colors of pixels on the perimeter of definite regions. Ruzon et al. [3] first introduced probability into estimation problem. They used GMM to describe color distributions in each small region. The component colors of a pixel are estimated as a weighted sum of the color pairs of F and B in the color models. Based on [3], Chuang et al. [4] presented a well-defined Bayesian framework and estimated F, B and α simultaneously by the maximum-likelihood criterion. The calculations in this category are direct and fast but they usually get poor results in complex images or large uncertain regions.

Methods in the second category add smoothness assumptions to improve the matting results. Sun et al. [5] supposed that F and B are smooth, by taking the gradient over eq.(1) and neglecting the gradients in F and B, they derived a Poisson equation to approximate the gradient field of α matte. Wang et al. [7] included the smoothness of α value in 4-nighborhood as a constraint in their object function. They employed belief propagation algorithm to

This work was partly supported by National Hi-Tech Development Program (863 Program) of China under Grant 2006AA01Z117.

iteratively pass information of colors and α values between pixels. Levin et al. [8] derived a closed form solution to matting. They assumed that both F and B are constant over a small window and then reduced the matting function into a quadratic function only with α as unknown. Generally, methods in the second category usually get better matte results, but the computational costs are also very expensive.

1.2. Structure information in images

Images have various structures, such as smooth regions with homogeneous color, one-pixel-width lines corresponding to hair-like boundaries, and edges disjoining two distinct regions, etc. These structures imply the spatial relationship of pixels in a neighborhood. Pixels located on the same structure have similar colors. These information give us clues where the right colors can be sampled for estimation.

Based on the above observation, we present a fast and efficient method to estimate the component colors. Our method first extracts the structure information in uncertain region. Then it exploits two kinds of strategies to propagate the known colors into uncertain regions based on the structure types. Compared with previous approaches, our method has two advantages: (1) the smoothness constraint is not imposed in the form of assumption. It is directly extracted from the observed images, so our calculation does not need any iterative computation; (2) the color propagation strategy gives pixels on smooth structure priority to be estimated. Therefore our method has the ability to handle objects having large interaction regions with background.

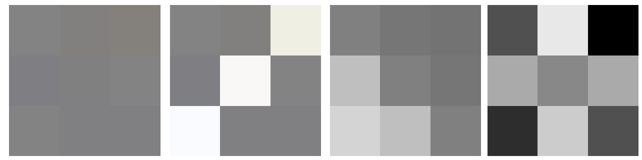
The remainder of the paper is organized as follows. Section 2 gives a detailed description of our method. Experimental results are shown in section 3 and section 4 concludes the paper

2. OUR METHOD

Our method requires user to provide a tri-map (as shown in Fig.3(c)) to pre-segment observed images into foreground, background and uncertain regions. It then detects the structure information of the 9-nighborhood of each pixel in uncertain regions. Two kinds of propagation techniques are introduced to estimate the component colors for each pixel according to the detected structure type. The propagation is performed by a well defined processing order. Finally, collinear constraints are used to improve the accuracy of the estimation.

2.1. Structure detection

For a 9-neighborhood, we classify its structure information into four categories: flat, line, edge and clutter. As shown in Fig.1, each structure type exhibits its own color smoothness. For flat type, all neighboring pixels have similar color with the center pixel. For line type, there is a one-pixel-width line crossing the center, and pixels on the line have homogenous colors while other pixels have another kind of color. For edge type, the center pixel is on the boundary of two regions which have different colors. For clutter type, the nine pixels have different colors.



(a) flat (b) line (c) edge (d) clutter

Fig. 1 sketch map of structure types in 3×3 region

We use Frei-Chen masks [9] to detect the structure types. These masks, as shown bellow, describe different features of structure: u_1 and u_2 are gradient masks; u_3 and u_4 correspond to ripples; u_5 and u_6 represent one-pixel-width line; u_7 and u_8 are similar to Laplacian masks; the last one is an average mask.

$$u_1=\frac{1}{2\sqrt{2}}\begin{bmatrix}1&\sqrt{2}&1\\0&0&0\\-1&-\sqrt{2}&-1\end{bmatrix}\quad u_2=\frac{1}{2\sqrt{2}}\begin{bmatrix}1&0&-1\\\sqrt{2}&0&-\sqrt{2}\\1&0&-1\end{bmatrix}\quad u_3=\frac{1}{2\sqrt{2}}\begin{bmatrix}0&-1&\sqrt{2}\\1&0&-1\\-\sqrt{2}&1&0\end{bmatrix}$$

$$u_4=\frac{1}{2\sqrt{2}}\begin{bmatrix}\sqrt{2}&-1&0\\-1&0&1\\0&1&-\sqrt{2}\end{bmatrix}\quad u_5=\frac{1}{2}\begin{bmatrix}0&1&0\\-1&0&-1\\0&1&0\end{bmatrix}\quad u_6=\frac{1}{2}\begin{bmatrix}-1&0&1\\0&0&0\\1&0&-1\end{bmatrix}$$

$$u_7=\frac{1}{6}\begin{bmatrix}1&-2&1\\-2&4&-2\\1&-2&1\end{bmatrix}\quad u_8=\frac{1}{6}\begin{bmatrix}-2&1&-2\\1&4&1\\-2&1&-2\end{bmatrix}\quad u_9=\frac{1}{3}\begin{bmatrix}1&1&1\\1&1&1\\1&1&1\end{bmatrix}$$

Convolving these nine masks with an image I, we get nine response images. For convenience of presentation, we first define some notations for further discussion. R_i denotes the response image of convolution between image I and mask u_i; (x,y) is the location of pixel at each image, $z(x,y)$ denotes the intensities of the 9-nighborhood centered at pixel (x,y) in I, we can write the convolution as

$$R_i(x,y)=\sum\nolimits_{j=1}^{9}u_{ij}z_j(x,y),$$

so R_i means the projection of z onto mask u_i, if the sub-image has the similar type with a certain mask, the convolution get a higher response.

To detect the structure type, we classify the nine masks into three subsets: $\{u_1 \sim u_4\}$, $\{u_5 \sim u_8\}$ and $\{u_9\}$, corresponding to the type of edge, line and flat respectively. We use the method in [10] for detection. Four values are calculated:

$$E_{edge}=\sum\nolimits_{i=1}^{4}[R_i(x,y)]^2,\ E_{line}=\sum\nolimits_{i=5}^{8}[R_i(x,y)]^2,$$

$$E_{flat}=[R_9(x,y)]^2,\ \text{ and } E=\sum\nolimits_{i=1}^{9}[z_i(x,y)]^2\ .$$

It can be proved that $\sum\nolimits_{i=1}^{9}[z_i(x,y)]^2=\sum\nolimits_{i=1}^{9}[R_i(x,y)]^2$, because these masks are orthogonal. So we can reduce the computation cost by calculating

$$E_{edge}=E-E_{line}-E_{flat}\ .$$

Type detection is described in Alg.1,

Alg.1 classify the structure types of 9-neighborhood z

if $E_{flat} / E > t1$, **then** $type(z)$ is flat;.
else
if $E_{line} / E_{edge} > t2$, **then** $type(z)$ is line;
else
if $E_{edge} / E_{line} > t3$, **then** $type(z)$ is edge;
else $type(z)$ is cluster;

where $t1, t2, t3$ are predefined thresholds.

We assign the detected structure type of the neighborhood to its center pixel. The information will guide us to choose the most similar colors from nearby pixels for color estimation during the propagation process.

2.2. Propagation strategy

Based on the detected structure types, two different propagation strategies, *smoothing propagation* and *sampling propagation,* are employed. The former, corresponding to types of flat, line and edge, utilizes the smoothness information to estimate component colors. The latter, corresponding to clutter, estimates component colors from nearby pixels through statistical method. We design a processing order to make sure that in each estimation the most available known information are used.

Color propagation starts from the definite foreground (background) regions inward to the uncertain regions separately. Suppose that p is the current pixel to be processed on the boundary, estimation is described as follows.

2.2.1 *Smoothing propagation*
If p has flat type, we search its 8-neighborhood to find a pixel q satisfying conditions that it has been processed and has the most similar observed color with p. If the similarity of the observed colors exceeds a threshold, we copy the component colors of q to p. Otherwise, the structure type of p is set to clutter.

If p has line type, we detect the direction of line crossing p. If the direction extends from the known region into the uncertain region, we propagate the known color through the line. Otherwise, the structure type of p is set to clutter.

If p has edge type, we can find the isophote line crossing p, which is perpendicular to the gradient direction. Since colors along the isophote line are similar, we propagate colors using the same method as for the line type.

2.2.2 *Sampling propagation*
For the clutter structure type, we estimate the component colors of p through a weighted sum of known pixel colors in a $(2n+1)\times(2n+1)$ sized window centered at p:

$$est\_color(p) = \sum_{q \in window(p,n)} w_q' \cdot est\_color(q) \qquad (2)$$

where w_q' is the normalized weight of w_q, which is defined as:

$$w_q = \begin{cases} \exp\left(-\dfrac{dist(Cp, Cq)}{\sigma^2}\right), & q \text{ has been processed}; \\ 0, & else. \end{cases} \qquad (3)$$

where $dist(C_p, C_q)$ is the Euclidian distance of two observed colors C_p and C_q in RGB color space.

2.2.3 *Processing orders*
The calculation starts from the boundaries of the uncertain region inward to it. The smoothing propagation executed first. When the types of pixels on the current boundaries are all clutter, sampling propagation is performed once to continue the whole processing.

We use the term "confidence" to determine the order of processing during the smoothing and sampling propagations. The confidence of a pixel is defined as the number of pixels that has already been processed in its 8-neighbor. The larger the number is, the more reliable information can be used, and the higher priority the pixel has to be handled in propagations.

2.3. Co-linear constraints

According to Equation (1), the colors C, F and B for a pixel are collinear, and the alpha value falls in [0, 1]. We utilize the following constraints to refine the estimation results.

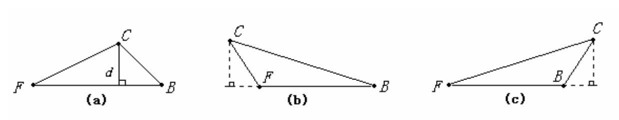


Fig. 2. Color co-linear constraints.

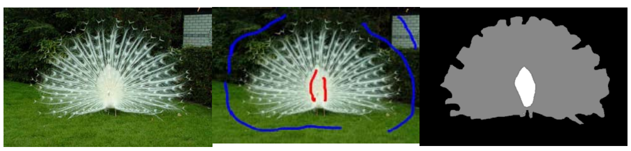
1. **Distance constraint**: As shown in Fig.2 (a), the ratio of d to the length of FB is required to be under a threshold, to prevent C from being far away from FB.
2. **Projection constraint**: as shown in Fig.2 (b) and (c), the projective point of C onto FB is required to lie on FB, to guarantee the alpha value to belong to the interval [0, 1].

Based on the above two constraints, the system checks the estimated color pairs F and B for each pixel, and for those failed points, they are estimated through sampling propagation under a bigger size of sampling window.

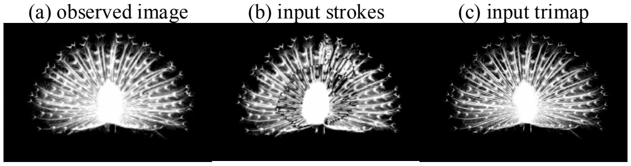
3. EXPERIMENTAL RESULTS

In our evaluation experiments, we first select two images with typical structures, *peacock* (Fig.3) and *cobweb* (Fig. 4) to check the validity of the proposed approach. We compare our algorithm with Wang's method [7] in visual effect and computation efficiency. All the experimental results are obtained on an IBM PC with P4 2.8G CPU and 256M RAM. In our experiments, the parameters in formula (2) and (3) are: $n = 3$ and $\sigma = 15$.As shown in Tab.1, our method is obviously faster than Wang[7]. Moreover, the results generated by our method are visually comparable to theirs. Fig.3 shows the results for *peacock* generated by the methods of Wang et al. [7], Bayesian [4] and ours. The run time of Wang's is longer (shown in Tab.1). Bayesian method fails because it can not get a well inference from far regions.

There mainly two reasons for the good performance of our method. Firstly, instead of taking numerous iterative computations, our method extracts the smoothness information directly from an observed image. Secondly, the processing order preserves the smoothness and gives the priority for estimation along the smooth types. Thus for estimation in large uncertain regions, the color information we used usually has high reliability.

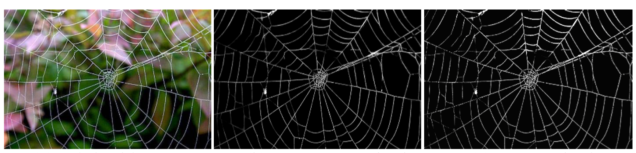


(a) observed image (b) input strokes (c) input trimap



(d) Wang 's method (e) Bayesian (f) our method

Fig. 3. Matting results of peacock. (e) is taken from [7].



(a) original image (b)Wang 's method (c) our method

Fig. 4. Matting results of the image cobweb.

Tab. 1. Comparison of computational efficiency for two images

Image	Wang's method[7]	Our method
Peacock	41.17s	16.77s
Cobweb	46.68s	9.63s

We also experiment on some hair-like images (in Fig.5) to further check the effectiveness and efficiency of our method. The results are encouraging. For the image *head*, the tiny strands of hair can be noticed. The computational time are summarized in Tab.2. It is obvious that our method has a good performance to objects with strong structures.

4. CONCLUSION

We have presented a simple and fast algorithm to solving the color estimation in natural image matting. Guided by the spatial structure information, two heterogeneous strategies are exploited to propagate color values from known regions into uncertain regions in an adaptive way. The experimental results show our method is comparable to the existing methods in accuracy, but is much more efficient in computational cost.

(a) *Gandalf* (b) *Galadriel* (c) *head*

Fig. 5. Results of hair-like objects.

Tab. 2. Computational time for the images in Fig.5

Image	*Gandalf*	*Galadriel*	*head*
Wang's [7]	2.05s	1.19s	4.17s
Our method	1.92s	1.31s	2.03s

5. ACKNOWLEDGEMENT

This work was supported in part by National Special R&D Project for 2008 Olympic Games of China under Grant 2005BA904B09 and Beijing Natural Science Foundation under Grant 4063041.

6. REFERENCES

[1] A.R. Smith and J.F. Blinn, "Blue Screen Matting", *SIGGRAPH96*, pp. 259-268.

[2] A.Berman, A.Dadourian and P.Vlahos, "Method for removing from an image the background surrounding a selected object", *U.S. Patent 6,134,346*, 2000.

[3] M.A.Ruzon and C.Tomasi, "Alpha estimation in natural images", *CVPR 2000,* pp.253-259.

[4] Y.Y.Chuang, B.Curless, D.H.Salesin and R.Szeliski, "A Bayesian Approach to Digital Matting", *CVPR 2001*, pp.264-271.

[5] J.Sun, J.Y.Jia, C.K.Tang and H.Y.Shum, "Poisson matting", *SIGGRAPH 2004*, pp.315-321.

[6] S.Y.Lin, Q.H.Zhang and J.Y.Shi, "Alpha Estimation in Perceptual Color Space", *ICASSP 2005*, pp.721-724.

[7] J.Wang and M.F.Cohen, "An interactive optimization approach for unified image segmentation and matting", *ICCV 2005*, pp.936-943.

[8] A.Levin, D.Lischinski and Y.Weiss, "A Closed Form Solution to Natural Image Matting", *CVPR 2006*, pp.61-68.

[9] W.Frei and C.Chen, "Fast Boundary Detection: A Generalization and a New Algorithm", *IEEE Trans. on Computers*, 26(2), pp.988-998, 1977.

[10] L.G.Shaprio and G.C.Stockman, *Computer Vision*, Prentice Hall, NJ, 2001