NATURAL IMAGE MATTING VIA ADAPTIVE LOCAL AND NONLOCAL SAMPLE CLUSTERING

Haiyan Yang, Oscar C. Au, Yuan Yuan, Wenxiu Sun, Yonggen Ling, Jiahao Pang

Department of Electronic and Computer Engineering
The Hong Kong University of Science and Technology
{hyangac, eeau, yyuanad, eeshine, ylingaa, jpang}@ust.hk

ABSTRACT

Digital image matting is the determination of foreground color, background color, and an opacity value of each pixel for an input image. Inherently, matting is a highly ill-posed and under-constrained problem. Thus, some assumptions need to be made to resolve it. Inspired by closed-form matting and color clustering matting, in this work, we first develop an adaptive sample clustering criterion to automatically assign either local or nonlocal neighborhood to each pixel. After that, in order to enhance matting accuracy, we improve the nonlocal clustering performance by introducing a new feature selection parameter to choose preferred feature space for different images in a fully automatic way. And finally we solve the problem using a closed form solution. Experimental results show that our algorithm achieves equal or even better performance among many *state-of-the-art* matting techniques.

Index Terms— image matting, sample clustering, local smoothness, nonlocal principle

1. INTRODUCTION

Natural image matting has gained an increasing attention in image and video processing societies. It refers to the problem of accurately estimating the foreground object from an image. Formally, matting problem can be expressed as the convex combination as shown below:

$$I_i = \alpha_i F_i + (1 - \alpha_i) B_i \tag{1}$$

where I_i is the pixel value in each pixel location i for an given image. F_i and B_i stands for the foreground color and background color respectively, and α_i is the so-called "alpha matte". (1) is also known as the *compositing equation*. The fact that for each pixel, we have three unknowns with only one equality constraint makes the problem severely under-constrained. To resolve the ambiguities, existing methods apply an extra input, which is known as the user-specified *trimap* as shown in Fig. 1(b), where white means purely foreground, black means purely background and gray means unknown.

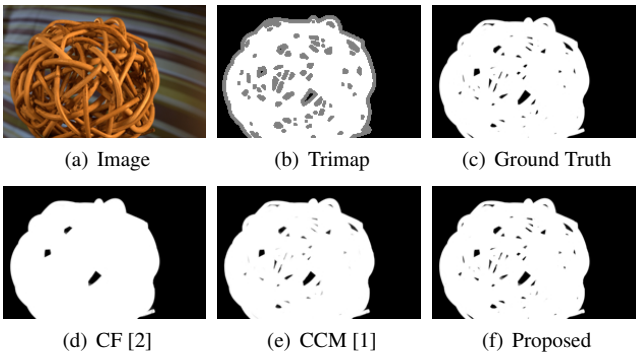
Traditionally most of the matting techniques can be categorized into: learning-based matting, sampling-based matting and propagation-based matting according to different matting techniques they employed[1]. To ease the discussion in this paper, however, we introduce another categorization according to different matting assumptions they made: local neighbor based matting, nonlocal neighbor based matting and a combination of both local and nonlocal neighbor based matting.

This work is supported in part by Hong Kong Research Grants Council (GRF), Innovation and Technology Fund, and the State Key Laboratory on Advanced Displays and Optoelectronics Technologies (Project No: ITC-PSKL12EG02).

One representative local neighbor based matting method is closed form matting (CF) [2] proposed by Levin *et al*.. They made a local smoothness assumption that in a small window, each of the foreground and background pixel value lies on a single line in RGB color space, which is known as the *color line model*. However, as discussed in [3], its matting performance is highly related to the size of the local window since large window is more likely to violate the color line model, He *et al*. [3] thus improved it by proposing an adaptive method to set appropriate window size for different image regions. Since closed form matting achieves quite satisfactory results for images with large smooth regions but does not perform well in images with complex textures, such as images with lots of holes as illustrated in Fig. 1, some nonlocal principles have been proposed in [1, 4, 5]. For example, instead of just using the neighborhood samples in each 3×3 local window, Shi *et al*. proposed a *nonlocal color ball model* in color clustering matting (CCM) [1] by assuming that "good" neighborhood samples should be gathered with similar appearance in some feature space. This nonlocal principle is also used in nonlocal matting [5] and KNN matting [4] for dealing with highly textured images.

In order to preserve the merits of both local and nonlocal principles, one recent work LNSP matting [6] treats the *matting laplacian* [2] as a smoothness prior to capture local structures of an image, then adds a nonlocal smoothness prior presented in [7] to capture global structures. Finally the matting problem is solved by constructing a sophisticated graph model. Another newly proposed method [8] utilizes not only the local information in closed form matting, but also another nonlocal principle, which is the same as in [4].

Since matting for highly textured image is always a nontrivial task to deal with, some other methods try to tackle the problem in a distinct way. In [9], E. Shahrian *et al*. model the self-defined *texture feature* via wavelet decomposition of the image followed by a two-step dimension reduction, then solve the problem by adaptively selecting good samples in either color space or texture feature space. Different from previous methods, in this paper, we address the matting problem by employing an adaptive local and nonlocal sample clustering scheme. The contributions of our method are twofold. Firstly, to effectively preserve both advantages of local smoothness assumption [2] and nonlocal principle [1], our work automatically decides which pixel should use the local principle and which one should use the nonlocal principle by developing an adaptive local and nonlocal sample clustering scheme. Secondly, compared with other two local and nonlocal combination methods [6] and [8], our method is much simpler than [6] in terms of the algorithm and more effective than [8] in terms of the performance. Experimental results tested on the benchmark datasets [10] show that our algorithm outperforms either CF or CCM in more than half of the cases. The re-



(a) Image (b) Trimap (c) Ground Truth

(d) CF [2] (e) CCM [1] (f) Proposed

Fig. 1. Performance illustrated for closed-form matting [2], color clustering matting [1] and our method on benchmark datasets [10]. Compared with the ground truth (c), CF (with PSNR = 21.47dB) fails to capture most of the holes inside foreground region and CCM (with PSNR = 26.67dB) is also unable to detect some small holes and details while our method (with PSNR = 31.04dB) can do the best among these three.

mainder of the paper is organized as follows. Two related works CF and CCM will be briefly reviewed in Section 2. Section 3 presents the proposed algorithm in detail. Experimental results will be given in Section 4 followed by the conclusion in Section 5.

2. RELATED WORKS

Recent matting methods tend to pay more attention in either exploring novel sampling techniques [11] or developing new models to combine the advantages in both local and nonlocal principles such as [8]. Since our method belongs to the latter one and is closely related with CF and CCM, these two methods are discussed briefly below.

2.1. Closed-form matting

In [2], Levin *et al*. assume that each foreground color F and background color B in a small patch can be modeled by a single line on the RGB color space respectively,

$$\begin{aligned} F_j = &\quad \beta_j^F F_i^1 + (1-\beta_j^F) F_i^2, \\ B_j = &\quad \beta_j^B B_i^1 + (1-\beta_j^B) B_i^2, \end{aligned} \tag{2}$$

where i indicates the pixel index and F_1, F_2, B_1 and B_2 are constant colors over the predefined small window (normally, they choose the window size as 3×3). After substituting these two equations into (1), it is possible to get rid of F and B and thus the alpha matte is only related with pixel values (detailed proof can be found in [2]).

$$\alpha_i = \sum_c a^c I_i^c + b \tag{3}$$

where c denotes the color channel. The goal of [2] is to minimize the error between the real alpha matte, which is to be found, and its linear approximation (3). Therefore, we want to minimize the following cost function:

$$J(\alpha, a, b) = \sum_{j \in I} \Big(\sum_{i \in w_j} (\alpha_i - \sum_c a_j^c I_i^c - b_j)^2 + \epsilon \sum_c a_j^{c2} \Big) \tag{4}$$

where w_j is the local window around pixel j, and term $\sum_c a_j^{c2}$ is the smoothness constraint on α. Since we can rewrite (4) as a *L_2-norm* minimization problem by adopting matrix notation, the optimal solution of a and b is given by a^* and b^*. This finally yields a quadratic cost function only related to α.

$$J(\alpha) = \min_{a,b} J(\alpha, a, b) = J(\alpha, a^*, b^*) = \alpha^T L \alpha \tag{5}$$

where L is called the *matting laplacian* [2, 12, 6]. By modelling the user guided input as a diagonal matrix D_S with diagonal elements being one for user constrained pixels and zero otherwise, and a vector b_S containing user labeled alpha values, the closed form solution can be achieved by solving the following linear system,

$$(L + \lambda D_S)\alpha = \lambda b_S \tag{6}$$

2.2. Color clustering matting

Instead of making use of the *color line model* in CF, Shi *et al*. [1] assume that for each pixel j, the potential candidates in its neighborhood set $\mathcal{N}(j)$ should be well clustered by using a ball model in the feature space, i.e. for $i \in \mathcal{N}(j)$,

$$\begin{aligned} F_j = &\quad F_i + r_{F_i}\mathbf{u}_F, \quad \|\mathbf{u}_F\| \le 1 \\ B_j = &\quad B_i + r_{B_i}\mathbf{u}_B, \quad \|\mathbf{u}_B\| \le 1 \end{aligned} \tag{7}$$

where r_{F_j} and r_{B_j} denote the radius of the balls in foreground and background respectively. $\mathbf{u}_F$ and $\mathbf{u}_B$ are two vectors indicating that the neighbors are all inside the balls. Substitute these two equations into (1), it is also possible to get rid of the ball model parameters and finally obtain a similar linear equation as in [2] besides the constant term b in (3),

$$\alpha_i \approx \sum_c a^c I_i^c, \qquad \forall i \in \mathcal{N}(j) \tag{8}$$

where c is the color channel. Thus, a similar cost function as (5) and linear equation as (6) can be achieved at last.

3. PROPOSED ALGORITHM

The *color line model* proposed by Levin *et al*. assumed that each pixel is only related to its local neighbors in a small window according to the smoothness assumption, which may face challenge when the image has complex intensity variations. On the other hand, the *color clustering ball model* constructed in [1] suggested that every pixel should be totally supported by the nonlocal neighbors searched throughout the whole image. However, it failed to collect appropriate neighbors when the image contains large amount of smooth regions. Inspired by these two matting techniques, we propose to solve the matting problem by a two-phase adaptive hybrid neighborhood clustering technique.

3.1. Phase 1: adaptive local and nonlocal sample clustering

As mentioned before that our algorithm automatically decides for each pixel whether local neighbor construction method as described in CF or nonlocal neighbor clustering scheme as in CCM should be adopted. The way to achieve this is by looking at the *contrast ratio* image of an image, denoted by ctr. Then ctr_i represents the contrast ratio of pixel i within a small window. ctr is constructed by filtering the image with a laplacian filter followed by an average filter,

$$ctr = I * f_{lap} * f_{ave} \tag{9}$$

where f_{lap} denotes 3×3 laplacian filter and f_{ave} denotes 3×3 average filter. The resulting image actually contains the edge information as shown in Fig. 2(b) and 2(d). If ctr_i is small, implying

that the local region around pixel i is smooth, then the image looks very dark as shown in Fig. 2(d) (in the extreme case, when $ctr_i = 0$, the local window around pixel i is totally smooth). In this situation, local neighbors in a small window are better choice to estimate the alpha matte for the target pixel. Otherwise, nonlocal neighborhood samples would be better. Concluded from the enlightenment of the above observation, we design an adaptive local and nonlocal sample clustering algorithm to automatically assign local neighborhoods for the relatively smooth regions and nonlocal neighborhoods for the complex textured regions of an image.

(a) Image (b) Ctr

(c) Part of (a) (d) Part of (b)

Fig. 2. Fig. (a) and (b) illustrate the original image and its ctr. Observed that (b) is black or nearly black in most of the smooth parts related in (a), which give us the heuristic information about different types of images in a very simple way. (c) and (d) are the part of the details of (a) and (b).

Noted that although *color line model* and *color clustering ball model* are two different models inherently, they lead to similar linear regression function (3) and (8), which brings the same closed form solution (6). Such an elegant fact gives us great insights for solving the matting problem. After replacing the construction of neighborhood set from local neighbor w_i [2] or nonlocal neighbor $\mathcal{N}(i)$ [1] by a new *adaptive local and nonlocal neighborhood set* $\mathcal{A}(i)$, a same cost function as (5) will be achieved after a similar reformulation procedure as in [1].

$$J(\alpha) = \alpha^T \tilde{L} \alpha \tag{10}$$

Except that a new *matting laplacian matrix* $\tilde{L}$ is defined according to $\mathcal{A}(i)$,

$$\tilde{L} = D - W \tag{11}$$

where $D_{ii} = \sum_j W_{ij}$ is a diagonal matrix and W is the associated affinity matrix with definition below,

$$\begin{aligned} W &= [k_{ij}]_{N \times N} \\ k_{ij} &= \sum_{m|(i,j) \in \mathcal{A}_m} \frac{1}{|\mathcal{A}_m|} (I_i - u_m) \Sigma_m^{-1} (I_j - u_m) \end{aligned} \tag{12}$$

Remember that we constrain the matting problem by the user-input trimap. Thus, we solve the matting problem by making it unconstrained via adopting the *Lagrangian* form (13) and finally obtain the closed form solution as (6).

$$J(\alpha) = \alpha^T \tilde{L} \alpha + \lambda (\alpha - b_S)^T D_S (\alpha - b_S) \tag{13}$$

3.2. Phase 2: feature space selection for nonlocal neighbor construction

In CCM [1], Shi *et al.* search the nonlocal neighbors for each pixel i in the feature space:

$$X_t(i) = [R, G, B, x, y, dR, dG, dB, ctr]_i^T, \tag{14}$$

where R, G, B are the pixel values in each color channel, x, y denote the spatial coordinates and dR, dG, dB stand for the color derivatives, The last term is ctr. Recall that in (9), ctr tells us the edge information of an image, that is, the mean value of ctr (denoted as $m(ctr)$) tends to be large when the image is highly textured and small when the image has large smooth areas. When we measure the difference or *distance* between two pixels in neighborhood clustering step, we penalize the pixel pairs with larger difference more for a particular item in X_t. Thus for smooth images, feature component ctr is detrimental to the determination of appropriate neighborhoods, because when searching in the normalized feature space, redundant ctr becomes comparable to other components in X_t. As a result, for different types of image, we cluster the nonlocal neighbors for each pixel not using the same feature space as that in [1], rather, to avoid the redundancy of feature space when dealing with images with large smooth subregions, we use another feature space X_s, which exclude ctr, for the clustering step.

$$X_s(i) = [R, G, B, x, y, dR, dG, dB]_i^T, \tag{15}$$

Inspired by the work in [13], the way we measure the preference for feature space is by introducing a feature selection parameter f_{sp},

$$f_{sp} = k_1 \cdot e^{-k_2 \cdot m(ctr)^2} \tag{16}$$

As shown in TABLE 1, higher $m(ctr)$ means smaller f_{sp}, then feature X_t performs better than X_s and vice versa, which is coincident with our analysis above. Test images are all selected from the benchmark datasets [10]. We set $k_1 = 10$ and $k_2 = 0.01$ in all our experiments. We summarize our adaptive local and nonlocal sample clustering algorithm in Algorithm 1.

Algorithm 1 Adaptive Local and Nonlocal Sample Clustering

```
procedure AdaClustering(f_sp, ctr, γ, δ)
    for i ← 1, N do
        if ctr_i ≤ δ then
            A(i) ← local neighbors
        else
            if f_sp ≤ γ then
                feature ← X_t
            else
                feature ← X_s
            end if
            A(i) ← nonlocal neighbors
        end if
    end for
end procedure
```

Table 1. Comparison of ctr among different types of images

Images		$m(ctr)$	PSNR via X_t (dB)	PSNR via X_s (dB)
Complex textured	GT02	2.5419	**31.04**	25.21
	GT04	3.6146	**25.16**	22.08
	GT11	3.6059	**28.53**	27.41
	GT16	2.5768	**19.62**	14.42
	GT25	5.3421	**20.51**	19.62
	GT26	4.7779	**20.94**	18.36
Relatively smooth	GT05	1.4414	32.5	**34.17**
	GT06	1.548	29.73	**35.24**
	GT07	1.5045	29.2	**31.01**
	GT12	1.7021	31.99	**34.05**
	GT14	1.5149	31.02	**35.07**
	GT15	1.4296	26.64	**30.38**

4. EXPERIMENTAL RESULTS

Among all the existing matting techniques, we compare our method with CF matting [2] and CCM [1]. In order to show a comprehensive comparison between our method and CCM (actually we want to demonstrate that our method performs better than CCM in most of the cases), we adopt the same evaluation as in [1], i.e., the peak-signal-to-noise-ratio (PSNR) to measure the matting accuracy.

$$PSNR = 10\lg(\frac{M^2}{MSE}) \tag{17}$$

where M is the maximum possible pixel value, typically 255 for a 8-bit image [1]. And MSE is the mean-squared-error measured between the ground truth and computed alpha matte. $MSE = \frac{1}{mn}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}\|I(i,j) - G(i,j)\|^2$, where $G(i,j)$ is the ground truth matte.

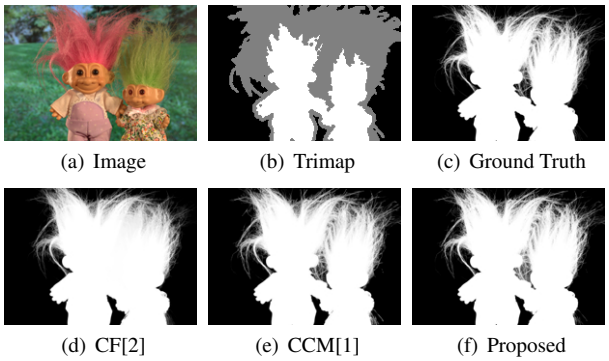
Table 2 shows the average PSNR comparison among CF, CCM and our proposed method in the same datasets. Fig. 3 to Fig. 5 shows the matting results tested on three different types of images. Specifically, the foreground object in Fig. 3 contains lots of fuzzy hair, which makes it difficult for the algorithm to capture the thin features. Fig. 4 is a natural image with many complex texture regions. Compared with the ground truth, our method detects more details than the other two methods. Fig. 5 is a relatively simpler one with less holes and more distinct edges. It turns out that our method performs the best in all these three different types of images.

5. CONCLUSION

In this paper, we present a novel matting algorithm to adaptively choose either local neighbor set or nonlocal neighbor set for each pixel in an image. There are two main steps. Firstly, we construct a feature selection parameter to automatically determine which feature space should be adopted. In the second step, either local or nonlocal neighborhood set is assigned to each pixel via our *adaptive local and nonlocal sample clustering* algorithm. Experimental results show satisfactory performance compared with other existing methods.

Table 2. Average $PSNR$ (in dB) of different matting techniques

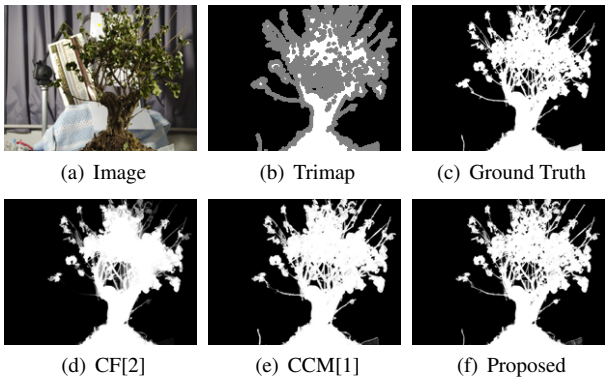
Methods	CF[2]	CCM[1]	KNN[4]	proposed
Average PSNR	27.22	27.74	26.6	**28.8**



(a) Image (b) Trimap (c) Ground Truth

(d) CF[2] (e) CCM[1] (f) Proposed

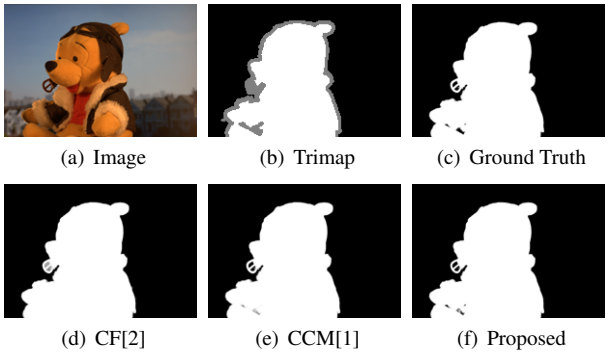
Fig. 3. Matting results on image with lots of fuzzy hair using CF (with PSNR = 19.86 dB), CCM (with PSNR = 24.02 dB) and proposed method (with PSNR = 25.16 dB)



(a) Image (b) Trimap (c) Ground Truth

(d) CF[2] (e) CCM[1] (f) Proposed

Fig. 4. Matting results on image with complex textures using CF (with PSNR = 17.34 dB), CCM (with PSNR = 18.14 dB) and proposed method (with PSNR = 20.94 dB)



(a) Image (b) Trimap (c) Ground Truth

(d) CF[2] (e) CCM[1] (f) Proposed

Fig. 5. Matting results on relatively smooth image using CF (with PSNR = 28.12 dB), CCM (with PSNR = 31.1 dB) and proposed method (with PSNR = 34.17 dB)

6. REFERENCES

[1] Y. Shi, O. C. Au, J. Pang, K. Tang, W. Sun, H. Zhang, W. Zhu, and L. Jia, "Color clustering matting," in *Multimedia and Expo (ICME)*. IEEE, 2013, pp. 1,6.

[2] A. Levin, D. Lischinski, and Y. Weiss, "A closed form solution to natural image matting," in *CVPR*. IEEE, 2006, pp. 61–68.

[3] K. He, J. Sun, and X. Tang, "Fast matting using large kernel matting laplacian matrices," in *CVPR*. IEEE, 2010, pp. 2165–2172.

[4] Q. Chen, D. Li, and Chi-Keung Tang, "Knn matting," in *CVPR*. IEEE, 2012, pp. 869–876.

[5] P. Lee and Y. Wu, "Nonlocal matting," in *CVPR*. IEEE, 2011, pp. 2193–2200.

[6] X. Chen, D. Zou, S. Z. Zhou, Q. Zhao, and P. Tan, "Image matting with local and nonlocal smooth priors," in *CVPR*. IEEE, 2013, pp. 1902–1907.

[7] X. Chen, D. Zou, Q. Zhao, and P. Tan, "Manifold preserving edit propagation," *ACM Trans. Graph.*, vol. 31, pp. 132:1–132:7, November 2012.

[8] M. Jin, B. Kim, and W. Song, "Knn-based color line model for image matting," in *ICIP*. IEEE, 2013, pp. 2480–2483.

[9] E. Shahrian and D. Rajan, "Weighted color and texture sample selection for image matting," in *CVPR*. IEEE, 2012, pp. 718–725.

[10] C. Rhemann, C. Rother, J. Wang, M. Gelautz, P. Kohli, and P. Rott, "A perceptually motivated online benchmark for image matting," in *CVPR*. IEEE, 2009, pp. 1826–1833.

[11] E. Shahrian, D. Rajan, B. Price, and S. Cohen, "Improving image matting using comprehensive sampling sets," in *CVPR*. IEEE, 2013, pp. 636–643.

[12] A. Levin, A. Rav-Acha, and D. Lischinski, "Spectral matting," in *CVPR*. IEEE, 2007, pp. 1–8.

[13] W. Sun, O. C. Au, L. Xu, and Z. Yu, "Adaptive depth assisted matting in 3d video," in *Multimedia and Expo (ICME)*. IEEE, 2011, pp. 1,6.