Chroma Keying Based on Stereo Images

by

Mengdie Chu

Thesis submitted to the
Faculty of Graduate and Postdoctoral Studies
In partial fulfillment of the requirements
For the M.A.Sc. degree in
Electrical and Computer Engineering

School of Electrical Engineering and Computer Science
Faculty of Engineering
University of Ottawa

Abstract

This thesis proposes a novel chroma keying method based on stereo images, which can be applied to post-process the alpha matte generated by any existing matting approach. Given a pair of stereo images, a new matting Laplacian matrix is constructed based on the affinities between matching pixels and their neighbors from two frames. Based on the new matting Laplacian matrix, a new cost function is also formulated to estimate alpha values of the reference image through the propagation between stereo images.

To avoid over-smoothing during propagation, a united window with an adaptive size is used in the Laplacian matrix to overcome the dilemma between the inadequate propagation caused by small size windows and the over-smoothness triggered by large size windows. Moreover, an optional pre-processing is presented. When it comes to the foreground object with strong reflection, based on the histogram statistics, the initial alpha values of the particular unknown pixels can be refined before the proposed post-processing method. In addition, a ground truth generation method is developed for chroma keying.

Visual quality comparisons between the original mattes produced by existing approaches and the results post-processed by the proposed method show that our method can effectively enhance the quality of alpha mattes. Also, objective quality comparisons demonstrate that our proposed algorithm can significantly reduce the errors of alpha mattes.

Acknowledgements

I would like to thank my supervisor, Professor Jiying Zhao, for his guidance and providing me the chance to conduct the research on chroma keying. I greatly appreciate his patience and confidence in my research work.

I am grateful to Wenyi Wang, Lei Chen and Dan Zhou, my lab colleagues, for their suggestions and sharing of knowledge and experience with me, as well as the encouragements they give. I would also like to thank my friends Xiaoli He and Danyan Chen for their helps and suggestions on writing.

Dedication

This is dedicated to my dear parents.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Image matting and chroma keying

Image matting, which has been studied for more than thirty years, is commonly considered as a procedure that accurately extracts foreground targets from static images or video sequences. In practice, it often comes with the inverse operation of image compositing, as shown in Fig. 1.1. The origin of study on image matting could trace back to the film industry at the beginning of last century. The blue screen method, instead of the black one used before, was first used to create a traveling matte background in the movie "The Thief of Bagdad", which won the Academy Award for Best Special Effects in 1940. Later, researchers began to develop the blue screen methods relying on optical techniques.

In 1984, the matting problem, which is referred to as *digital/alpha matting* now, was first mathematically established by Porter and Duff. In [1], they introduced a model that the observed image I_z could be thought as a convex combination of foreground image F_z and background image B_z. The compositing equation is given by

$$I_z = \alpha_z F_z + (1 - \alpha_z) B_z \tag{1.1}$$

where $z = (x, y)$ is image coordinate, the opacity value α_z takes any values in the range [0, 1] and it linearly controls the interpolation of F_z and B_z for each pixel. Pixels with

$\alpha_z = 1$ are definite foreground and those with $\alpha_z = 0$ are definite background, otherwise they are referred to as mixed pixels. The goal of matting problem is to solve F_z, B_z and α_z for the given color value I_z, especially through estimating alpha values for mixed pixels to separate foreground from background.

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

Figure 1.1: A matting example from [4]. (a) Original image. (b) Extracted matte. (c) Estimated foreground colors. (d) A new compositing.

With the great progress of computer and digital camera technologies, alpha matting has been playing an increasingly vital part in multimedia area, especially bringing growing interests in numerous image and video processing applications today. Consequently, both academia and industries have proposed a variety of matting algorithms and systems to separate high-quality mattes. According to the image characteristics, we can classify matting techniques into two groups: blue screen matting [2] and natural image matting [3].

Blue screen matting, which is also known as chroma keying, is a special case of alpha matting that sets the foreground object to be shot in front of the background with one or multiple known constant colors. In the early 1970s, American and British television networks started to make use of green backdrops instead of blue color, and now green or blue backgrounds are more commonly used in professional studios (see Fig. 1.2). Now, this kind of techniques has been developed to remove a background and composite extracted foreground objects onto a virtual set. Accordingly, fruitful applications have been widely found in many fields such as movie, news casting and 3D game industries. Nonetheless,

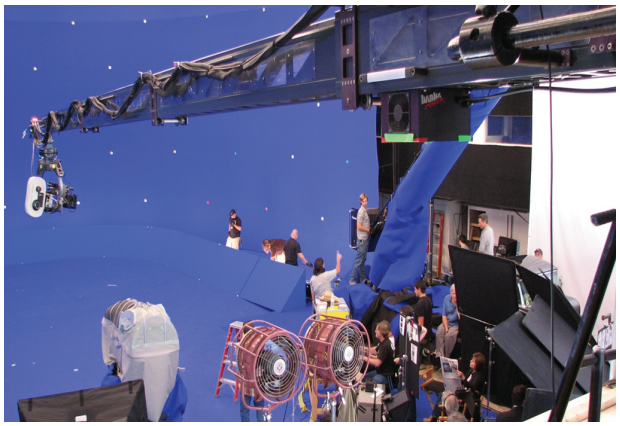


Figure 1.2: The blue background environment for chroma keying from [11].

the influence on the matting procedure caused by reflective regions, transparent objects and shadow casting is still a hard problem to handle.

Compared with chroma keying, more academic works focus on generalized alpha matting, which are classified into natural image matting. These algorithms deal with the images where backgrounds are arbitrary. Generally, the academic alpha matting methods also perform well when they process the images with constant colored backgrounds that are usually processed by chroma keying methods. Natural image matting has been extended to some specialized issues, such as video matting [5] [6], shadow matting [7] [8] and environment matting [9] [10]. The typical approaches with respect to chroma keying and natural image matting are reviewed in Chapter 2.

1.2 Trimap and scribbles

According to the *matting equation* as presented in Eq. (1.1), considering a given pixel in a 3 color channel image, we can rewrite three equations in a matrix expression:

$$\begin{pmatrix} I_r \\ I_g \\ I_b \end{pmatrix}_z = \alpha_z \begin{pmatrix} F_r \\ F_g \\ F_b \end{pmatrix}_z + (1 - \alpha_z) \begin{pmatrix} B_r \\ B_g \\ B_b \end{pmatrix}_z \tag{1.2}$$

As shown in Eq. (1.2), there are seven unknown variables in total: the scalar alpha value α_z and the three dimensional color vectors F_z and B_z. However, the information we know is only the three dimensional color vector I_z. Apparently, it is an under-constrained ill-posed problem. Thus, most algorithms need prior assumptions and side information to constrain variables to get good solutions. Trimap [12] and scribbles [13] are two typical side information modes from users.

An image is segmented into three regions in its trimap (see Fig. 1.3(b)): the white, the black and the gray, which denote absolute foreground, absolute background, and unknown regions respectively. The alpha values of pixels in absolute foreground and absolute background are set to be 1 or 0, respectively. For the pixels in unknown region, the situation becomes complicated since foreground and background colors are mixed. Hence, their alpha values are estimated according to prior assumptions including color, location, texture, etc. Different from trimap demanding to label all regions, the scribble-based methods require less user-supplied information. In this case, the user only needs to mark a few image regions as definite foreground or definite background by the yellow or blue scribbles, as illustrated in Fig. 1.3(c). Thus, scribbles-based algorithms require fewer user interactions and operations than trimap-based ones.

(a) (b) (c)

Figure 1.3: An example of trimap and scribbles used to label the image from [14]. (a) Original image. (b) Trimap mode. (c) Scribbles mode.

Generally, the accuracy of the obtained image matte crucially relies on the precision of side information. In terms of the trimap, the unknown regions around the edge of

foreground objects are expected to be marked as exactly as possible, particularly for images with complicated scenes, for instance, hairs. Obviously, it is time-consuming to obtain a trimap supplying more known detailed information, but such a precise trimap can help to improve the matting accuracy and generate a satisfactory result. On the other hand, although scribbles-based methods save stringent user operations, they face challenges about scribbles locations. For example, dealing with fuzzy regions in images, the matting quality is directly affected by the predefined location of the known labels, and unsuitable locations could result in a degraded matte.

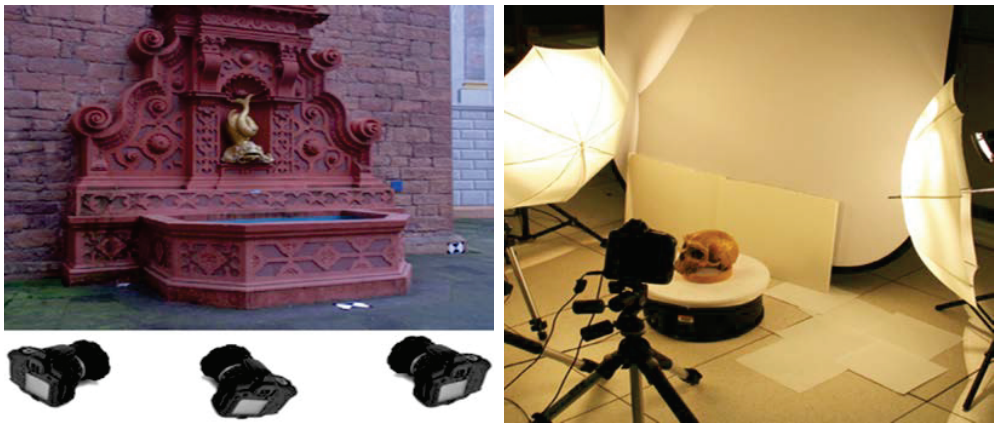
Meanwhile, scribbles-based approaches usually do not perform as well as trimap-based methods because they have larger unknown regions to estimate. Consequently, the scribble is acceptable for the user who prefers fewer interactions and a low quality matte. In contrast, the trimap is appropriate for occasions where a high-quality matte is needed. In addition to being specified by users, trimaps could be given automatically [15] or semi-automatically [16] as well.

1.3 Multiple view photographs and stereo images

Similarly to chroma keying, 3D imaging, reconstruction and printing techniques have attracted researchers' attention for more than 30 years and been becoming a widespread concern in recent decades. One of the key parts among them is to collect a set of multi-view photographs of an object and estimate the corresponding camera parameters under the assumptions of known viewpoints and lighting conditions (see Fig. 1.4).

The multi-view stereo algorithms [18] [19] [20] use more than two images and capitalize on the stereo correspondence as their main cue to reconstruct a 3D object. For instance, Google StreetView [21] is a well-known application in real life. Usually, this kind of algorithms comprises of four steps:

- Collect images,



(a) (b)

Figure 1.4: Multi-view images capture setups from [17]. (a) Outdoor capture of small-scale scenes. (b) A controlled capture using diffuse lights and a turn table.

- Estimate and calibrate camera parameters for each image,
- Reconstruct the 3D geometry of the scene from the images through corresponding parameters,
- Optionally recover the details of the scene.

In terms of photo-consistency, many concepts of the multi-view stereo are shared with two-view stereo systems. This is because a pair of stereo images, where two photographs of the same object are captured at slightly different angles, is a basic cell of multiple view photographs. The relation and difference between two frames can be explained by the concept of binocular disparity, which is first introduced by Marr in [22] to define the difference in location of corresponding features captured by the left and right eyes. As presented in Fig. 1.5, the horopter is the set of points with zero binocular disparity by the red dotted curve. It is a circle that includes the fixation points and the optical centers (lenses) of the two eyes. A crossed object that is shown by yellow dot is nearer to the observer than the horopter line. It has crossed disparities since the observer has to cross (converge) eyes to fixate on it. As a result, it lies further to the left from the right viewpoint than from the left viewpoint. In contrast, the point farther away from the observer than the

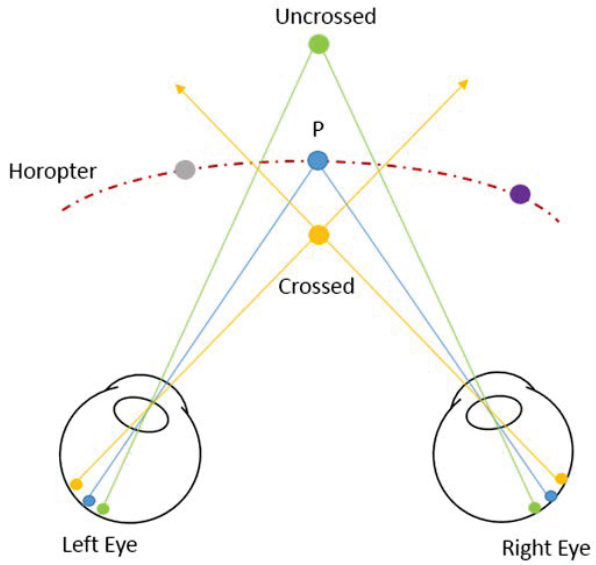


Figure 1.5: The horopter and disparities.

horopter is an uncrossed object, which is shown by green dot. It has uncrossed disparities because the observer has to uncross (diverge) eyes to fixate on it. It lies further to the right from the right perspective.

Stereo images are usually applied to create an impression of depth and solidity in stereo matching or stereoscopy techniques. They can provide viewers with increased information about the binocular disparity and more perceivable details of the object taken in images than common single 2D image. Fig. 1.6 gives an example of stereo images and a simulation of 3D image by coloring the two pictures, one in blue and one in red.

Since Marr defined the disparity mathematically in 1982, a variety of algorithms on stereo correspondence have been developed. These kind of approaches aim to estimate a disparity at each pixel and generate a disparity map. Meanwhile, a disparity map is often regarded as a depth map. Through this map, it could be easier to obtain the relevance of matching parts and deal with stereograph applications such as image-based rendering and view synthesis.

Figure 1.6: A simulation of stereoscopic image from [23]. Top left: The left eye view. Top right: The right eye view. Bottom: The 3D stereoscopic image.

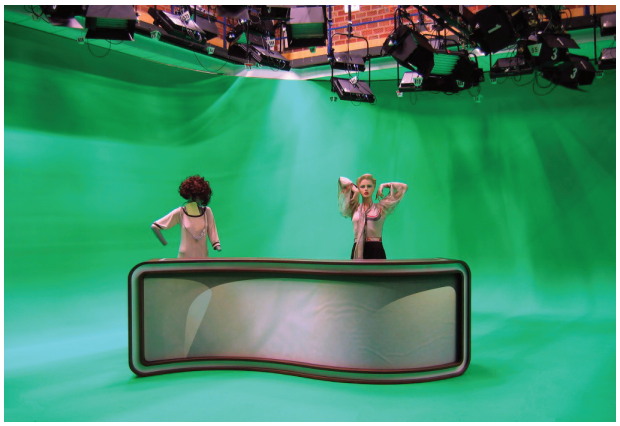


Figure 1.7: The lighting influence for chroma keying from [11].

1.4 Problem statement

The images for chroma keying are commonly shot in an indoor studio, and the existing matting techniques face challenges from the lighting conditions in practice. Although the background is lightened and monochromatic, the color cannot be ensured to be constant ideally for the whole background. As illustrated in Fig. 1.7, different parts of the green screen present a strong contrast. Besides, being shot in front of a constant-colored screen, the foreground objects are usually tinted with the same color of the background. It is known as the spill effect. For example, the green reflections and shadows can be observed on the desk in Fig. 1.7. Both of these phenomenons are results of light reflection and scattering, and accordingly increase the matting difficulty. In addition, the luminance changes on the transparent/semi-transparent object also make matting hard to be solved.

In academia, a number of existing algorithms which deal with natural image matting can also be used for the chroma keying if a fine hand drawn trimap is supplied by users. However, the state-of-the-art chroma keying techniques tend to automatically implement the whole processing procedures including specifying trimaps. In this case, the matting quality would be probably degraded with an auto-generated trimap if the matting algorithm was originally designed with manually specified trimap. For instance, transparent regions and the surfaces with low brightness or high reflection often bring the puzzles to label

judgment. As a result, those regions are marked as unknown regions since their color (e.g., Hue, Saturation, Lightness) ranges are different from that of known regions, and the quality of final matte will be greatly affected.

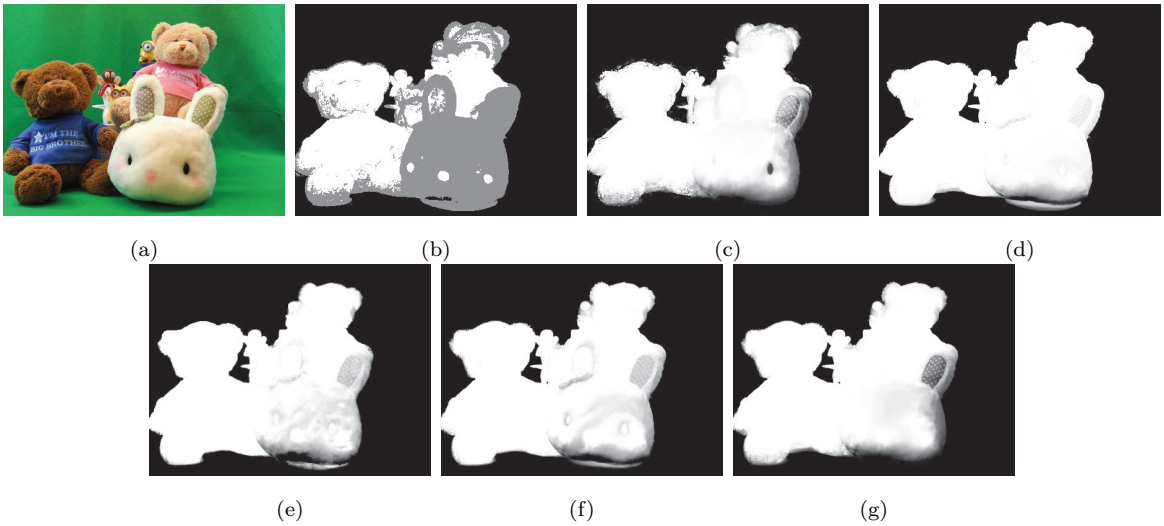
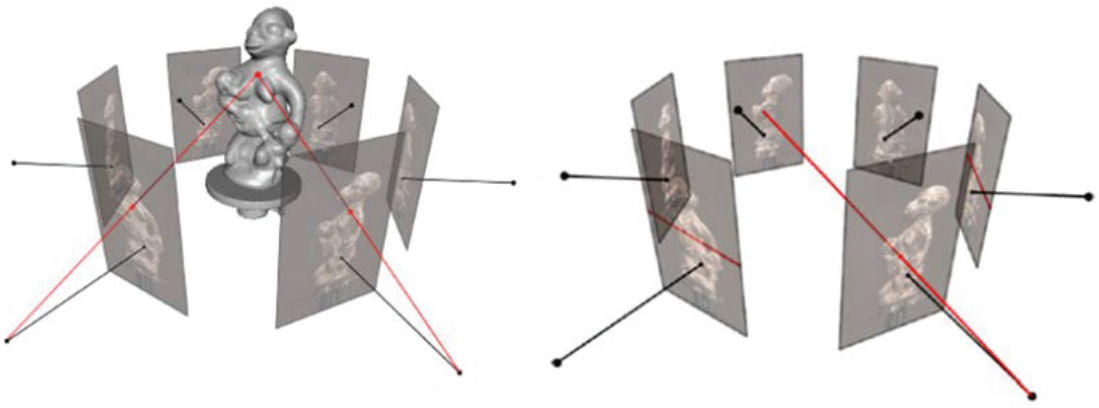


Figure 1.8: Visual quality comparison for the mattes from different methods with an auto-generated trimap. (a) Input image. (b) Trimap produced by the quadmap method [24]. (c) Closed-form [25]. (d) KNN [26]. (e) Weighted color [27]. (f) Comprehensive sampling [28]. (g) Learning based [29].

Given the input colored image and its trimap which is automatically generated by the quadmap method [24], Fig. 1.8 shows a group of matting results obtained by five typical matting algorithms—closed-form [25], KNN [26], weighted color [27], comprehensive sampling [28], and learning based [29] matting methods. Although these methods are the state-of-the-art algorithms that solve the matting problem from different aspects, the results presented here are far from perfect.

Take the bunny doll in Fig. 1.8 as an example, we can see that these methods estimate the alpha values improperly in this region, especially for the results from closed-form, weighted color, comprehensive sampling, and learning based matting methods (see Fig. 1.8(c) (f) (g) (h)). Although KNN matting visually performs better than others in this region, its result still needs to be improved from human perspective. For example, there

are obvious errors in the right ear and the chin of the bunny doll in Fig. 1.8(d). Besides, except KNN, all other methods do not work well at the fuzzy edge of the bear doll on the right top corner.



(a) (b)

Figure 1.9: An example of multiple photographing and matching images from [17]. (a) The 3D target determines a correspondence between pixels in different images. (b) For a certain pixel in one image, searching the corresponding pixel in another image is a 1D problem only if the image are calibrated.

On the other hand, 3D model reconstruction has been conducted with great progress recently. As shown in Fig. 1.9, a group of images which focus on the same 3D target scene are shot in varied viewpoints. Through analyzing the properties (*e.g.*, chrominance and luminance) of the pixels from different surfaces, the multi-view images can provide us with the information on the homologous location of the scene in each image, as well as the relative position and depth relations between different parts in the scene. The whole analysis procedure is included into the stereo matching problem. Consequently, with the help of multi-view images, a novel recognition and reconstruction of the target is accomplished.

Inspired by stereo image processing, we propose a novel method to estimate the alpha matte using the multi-view images instead of a single image. Since multi-view images

contain more comprehensive information of the scene, by analyzing corresponding pixels, our proposed method can generate better matting results than conventional algorithms that only use one input image. Without the loss of generality, we use stereo images as the input in the remaining part of this thesis.

1.5 Thesis contributions

Most systems on alpha matting focus on generating a matte by using single image only. This thesis proposes a novel post-processing method based on stereo images for chroma keying. The main contributions include the following work:

1. For a pair of stereo images, one frame is regarded as the reference image from which the alpha matte is extracted. The other frame is treated as the matching image providing more complementary correlation information. A novel matting Laplacian matrix is constructed based on the correspondences between stereo images. Meanwhile, a specialized cost function is built. Our experimental results demonstrate that the proposed method can efficiently improve the alpha matte quality as a post-processing for any existing matting approach.

2. Compared with fixed window size used in most Laplacian-based algorithms, an adaptive window size is employed in our new matting Laplacian matrix. According to image characteristics, we choose a proper window size to overcome the dilemma between the inadequate propagation caused by small size windows and the over-smoothness triggered by large size windows.

3. As a post-processing method, the quality of the input initial alpha matte would affect our work. In order to alleviate the influence of reflections on the object, pre-processing is used to correct the initial alpha value of some pixels. It is an optional step depending on the luminance and reflection distribution of the input image.

4. Due to the lack of ground truth for test images in the thesis, a method is provided to compute a matte that approximates ground truth as closely as possible. The generated result is considered as ground truth to evaluate our work objectively.

1.6 Thesis organization

The next chapter reviews the state-of-the-art chroma keying and alpha matting techniques in the fields of both industry and academia. The stereo matching work is reviewed in the Chapter 2 as well. Chapter 3 fully introduces the proposed chroma key method based on stereo images. Then Chapter 4 illustrates the experimental results and the quality improvement of alpha mattes by the proposed method that post-processes the matting results of existing matting approaches. Finally, the conclusion of the thesis is drawn and the future work is discussed in Chapter 5.

Chapter 2

Literature Review

In this chapter, we will first review the classical algorithms on chroma keying and natural image matting. Since our work is based on stereo images, the concept and the typical approaches on stereo matching will be reviewed as well. Finally, we will review some combinations of matting and stereo matching.

2.1 Chroma keying

As introduced in Section 1.1, existing matting techniques could be categorized into two types as follows.

- Chroma keying in which the foreground is extracted from a monochromatic background.
- Natural image matting in which the foreground is separated from a compound background.

Compared to the limited amount of academic articles, more techniques on chroma keying are published by industrial patents, *e.g.*, [30] [31]. They have been extensively applied to

the multimedia industry. Before the overview of chroma keying techniques in industry, it is necessary to look back into the early matting development.

2.1.1 The early development of blue screen matting

The ancestry of matting was to settle the problem during the black and white filmmaking that transferred elements into a scene which were not present in the initial exposure. Georges Méliès first introduced the double exposure method during the production of the film "Un home de têtes" in 1898, using a black draping where a green screen would be placed today. Later, Edwin S. Porter used double exposure to add background mattes with a garbage scene to the window area.

In order to make background move along with the live foreground, Frank Williams patented a travelling matte technique [32] in 1918. The process started with photographing foreground objects in front of a black background, and then developed a negative film. The negative film was intensified to produce a black silhouette as far as possible. Once it turned into a completed black and white mark which segmented the foreground and background, the matte could travel with foreground objects. This approach was applied to many films including "The Invisible Man" in 1933.

In 1928, the colored screen first emerged in the Dunning Process [33]. With the help of a color filter, both positive and negative films were exposed by photographing a substituting background. Later, credited to Larry Butler, a blue screen matting process was developed to create a travelling matte with proper colors. It was first used in the film "The Thief of Bagdad", which achieved the Academy Award for Best Special Effects in 1940.

After Larry Bulter, Petro Vlahos also won the Academy Award in 1964 for his matting techniques used in color films. Sodium vapor process [34] was one of the well-known contributions among tremendous achievements [34] [35] [36] [37] by him. The sodium vapor illumination, whose wavelength is near 589.6 nanometers, plays an important role in this method, since this specific wavelength range does not overlap with the wavelengths of

common colors. Based on this principle, the sodium vapor separated foreground objects from scenes using a light filter.

At the same period, the similar patent of creating ultraviolet travelling mattes [38] was invented by Tondreau *et al.*. Either sodium vapor or ultraviolet light solved the blur and translucency problems resulting from the overlap of foreground and back screen colors. These techniques based on physical ideas were extensively used in the motion picture and television industries until computer-time.

2.1.2 Chroma keying in industry

When it comes to digital-time, the blue screen matting is also recognized as chroma keying. Although the term "blue" is mentioned here, some other colors can be set as background as well depending on situations and applications. In practice, the background color plays a crucial role as side information in the matting process, and now a green or blue screen is commonly used because these two colors are most distinct from human flesh pigments. In this thesis, we do not differentiate blue screen matting and chroma keying. The classical patents and techniques on them are introduced in this section.

Color difference matting

One remarkable contribution of Vlahos was introducing a color difference matting approach [39] used in the filmmaking to correct the foreground colors. It was accomplished by calculating the difference between the positive film and the negative film on the blue and green channel respectively. Slightly different from the matting equation used today, he modeled the observed object colors as follows:

$$C_0 = C_f + (1 - \alpha_0)C_b \tag{2.1}$$

where C_0, C_f and C_b represent the observed, foreground and background colors respectively.

In 1978, Vlahos refined the approach underlying the fact that most objects on the earth have the nearly same amount of blue and green components. The assumption of the relation between blue and green elements from foreground colors was:

$$B \leq kG \tag{2.2}$$

where B and G are respective values of blue and green channels for each pixel, and k is a constant with the range [0.5, 1.5]. The red channel was taken into consideration as well. Based on the initial matte produced by the key color difference, the estimated values could be corrected through remaining maximum color component pixel by pixel. The final matte control function was provided by

$$E_c = K_1[B - k(K_2 \max(K_3R, K_4G) + (1 - K_2) \min(K_3R, K_4G))] \tag{2.3}$$

where R, G, B are the red, green, blue components of each pixel on the foreground, and $K_i(i \in (1, 2, 3, 4))$ are tunable parameters. The technique was realized in a piece of digital equipment exploited by the professional video studio called Ultimatte Corporation. Furthermore, the latest patent [40] regarding this approach developed the generation function into:

$$E_c = K(B - K_1) - K_2 \max(K_3G,\ K_4R) - \max(K_5(G - R),\ K_6(R - G)) \tag{2.4}$$

where K and $K_i(i \in (1, 2, ..., 6))$ are tunable parameters.

Distinguished from the early skills introduced in Section 2.1, the color difference approach could be regarded as the beginning to model the matting problem mathematically. Not only did it accelerate the special effort development of motion pictures in last century, but also its basic idea influences a lot of chroma key techniques today. However, such equation forms (Eq.(2.3) and Eq. (2.4)) are considered as conclusions from decades of experiences and experiments, not by a convincing theoretical derivation.

Triangulation matting

Since back screens present a fixed color, the matting problem is simplified by estimating known background values. Given an unknown pixel, unfortunately, it is not that simple to separate background if the back screen has imperfect illumination and shadows. Smith and Blinn made an extension [2] touching on the problems such as blue spill, back impurities, and backing shadows. Compared to the Vlahos' work that α_0 was related to the given image only, they proposed the desired alpha channel α_0 was a function combining the known backing image C_k and the image C_f that was composited by their model. Firstly, given foreground elements C_f and background elements C_b at corresponding points, they assumed a matting equation could be expressed as follows:

$$C_f = C_0 + (1 - \alpha_0)C_k \tag{2.5}$$

where C_0 is referred to uncomposited foreground color, and C_k is known backing color. Once C_0 is determined, a composited color $C = C_0 + (1 - \alpha_0)C_b$ could be computed to obtain the final result.

Furthermore, Smith and Blinn discussed three special situations about C_0 without blue, grey or flesh, and triangulation. In the simplest triangulation case, the foreground was shot against two different shades of blue backing colors $[0,\ 0,\ B_{k_1}]$ and $[0,\ 0,\ B_{k_2}]$, where $B_{k_1} - B_{k_2} \neq 0$. When it came to a foreground color $F_0 = [F_r,\ F_g,\ F_b]$, it had two observed colors:

$$\begin{aligned} [I_{r_1},\ I_{g_1},\ I_{b_1}] &= [F_r,\ F_g,\ F_b + (1 - \alpha_0)B_{k_1}] \\ [I_{r_2},\ I_{g_2},\ I_{b_2}] &= [F_r,\ F_g,\ F_b + (1 - \alpha_0)B_{k_2}] \end{aligned} \tag{2.6}$$

Then they deduced the expression for the alpha channel from combining these equations to eliminate F_b as follows:

$$\alpha_0 = 1 - \frac{I_{b_1} - I_{b_2}}{B_{k_1} - B_{k_2}} \tag{2.7}$$

In addition, a more generalized expression is presented by Eq.(2.8) for the situation of photographing foreground objects before two distinct background colors C_{k_1} and C_{k_2}. In

this case, either background color is arbitrary and $(R_{k_1} - R_{k_2}) + (G_{k_1} - G_{k_2}) + (B_{k_1} - B_{k_2}) \neq 0$.

$$\alpha_0 = 1 - \frac{(I_{r_1} - I_{r_2}) + (I_{g_1} - I_{g_2}) + (I_{b_1} - I_{b_2})}{(R_{k_1} - R_{k_2}) + (G_{k_1} - G_{k_2}) + (B_{k_1} - B_{k_2})} \tag{2.8}$$

Based on the work above, triangulation matting is employed in a variety of matting methods being the part of computing ground truth maps [41] [42]. In practice, it is not necessary to require an image with a constant colored background, although Eq.(2.8) is built on the condition that the difference between two sums of RGB channels' values of two background colors must not be equal to zero.

Hexoctahedron matting

Founded on representative foreground and background samples, Mishima investigated a geometrical method [30] for chroma keying. As shown in Fig. 2.1, two hexoctahedrons, which are defined by combining triangular pyramids centered on a representative background color, are constructed in RGB three-dimensional space. Due to the fact that the background has only one color cluster, the small inner hexoctahedron can enclose all background pixels. In contrast, the other big hexoctahedron that lies outside the small one covers all foreground pixels. To ensure the accuracy of the final alpha values, the size of the inner one is expected to be set as small as possible, while the opposite situation shall be set for the outer one.

In this case, the alpha value of a pixel is estimated by calculating the relative distance to the two hexoctahedrons as follows:

$$\alpha = \begin{cases} 0 & (d_r \leq d_1) \\ 1 & (d_r \geq d_2) \\ (d_r - d_1)/(d_2 - d_1) & (d_1 < d_r < d_2) \end{cases} \tag{2.9}$$

where d_r is the distance from the centroid of the polyhedron to the observed color, d_1 and d_2 are the radiuses of the two polyhedrons respectively.

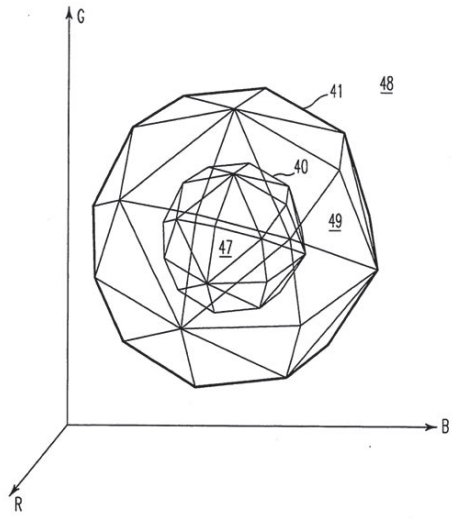
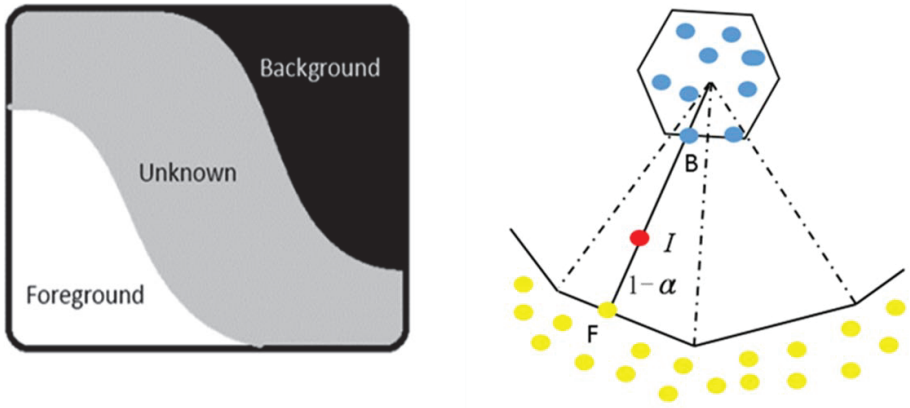


Figure 2.1: Two hexoctahedrons defined in RGB space [30]



(a) (b)

Figure 2.2: Mishima's hexoctahedron matting method. (a) Sampling. (b) Estimation of alpha values.

Fig. 2.2 provides an example that blue dots in the polyhedron sphere B are definite background pixels, and yellow dots in the polyhedron sphere F are definite foreground pixels. Moreover, the red dot located between the two known spheres is an unknown pixel I which may belong to the edge or transparent parts where $\alpha \in (0, 1)$. According to the Eq.(2.9), once a straight line passes through the sphere and the unknown pixel I, α could be easily derived from the distances d_r, d_1 and d_2.

The most important and the most difficult step in the whole process is how to control the hexoctahedrons' sizes according to group colors in sRGB space. In fact, it is hard to classify colors perfectly when they are mixed and the locations are clustered on the surfaces of the polyhedrons.

HSL based matting

A feature but also a drawback of Hexoctahedron matting method [30] is to define color volumes by the polyhedrons with many dozens of sides. Considering the complexity of the realization, Hue-Saturation-Lightness (HSL) based matting approaches tend to use luminance information for simplicity. Under ideal situations, there should be a consistency between the hue and saturation for the background, as both hue and saturation are only related to the background dye and are independent of lightness. Consequently, this consistency makes the background colors easier to be clustered into a particular region and thus to be removed. Discreet Keyer [43] and Diamond Keyer [44] are two well-known industrial patents regarding HSL based matting techniques.

Different from Mishima's method [30], the background color distribution and blending tolerances are covered by two HSL triplets in Diamond Keyer [44] (see Fig. 2.3). In particular, a background color is first received in this method. According to the known information, a color diamond in the plane of a constant luminance is constructed. A luminance range is also built simultaneously. After that, a transformation region is defined in response to known user indication. At last, a final matte is identified by processing

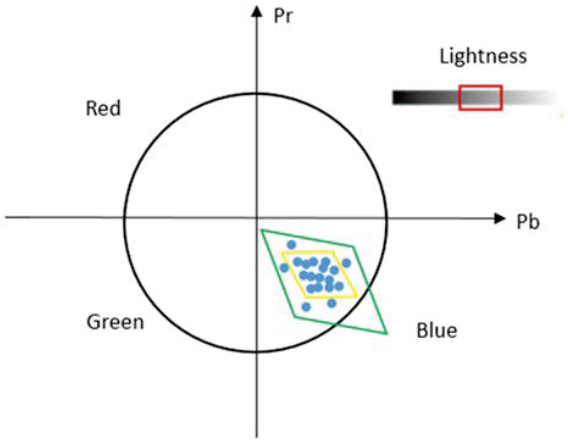


Figure 2.3: Diamond keyer's segmentation in the HSL color space: the yellow polygon covers absolute background chrominance from a user indication; the green polygon covers transformation ranges; the red box on the lightness bar constrains the background luminance.

pixels of the known foreground image with the defined transformation.

These kind of approaches perform well based on strong assumptions of the consistency of hue and saturation of the background. However, the illumination that is not pure white in practice usually has a negative effect on the consistency and hence limits the applications to complicated lighting cases.

2.2 Natural image matting in academia

A variety of industrial chroma keying patents pull a matte from a solid colored background, however, much more academic works pay attention to extract foreground objects from an arbitrary and complex background. These kind of alpha matting algorithms are referred to as natural image matting. It is a more generalized discussion since it is common to catch an unlimited scene as a background in the real world. Nevertheless, the unpredictable background requires heavier computation and more user interactions.

As described in Eq. (2.10), the matting problem could be specifically considered as a process to find the optimal solution to the alpha value for each pixel and its corresponding foreground and background colors [3] [4].

$$\begin{pmatrix} I_r \\ I_g \\ I_b \end{pmatrix}_z = \alpha_z \begin{pmatrix} F_r \\ F_g \\ F_b \end{pmatrix}_z + (1 - \alpha_z) \begin{pmatrix} B_r \\ B_g \\ B_b \end{pmatrix}_z \tag{2.10}$$

where $z = (x, y)$ is image coordinate.

According to fundamental ideas of matting systems, most exsiting natural image matting methods could be divided into four categories:

1. Color sampling-based approaches;
2. Propagation-based approaches;
3. Combination of color sampling and propagation approaches;
4. Learning-based approaches.

Although this thesis concentrates on chroma keying, the representative methods in each category are reviewed as well. This is because the classic ideas of natural image matting have been inspiring the researchers when solving issues in chroma keying.

2.2.1 Color sampling-based approaches

Depending on image matting statistics, it is believed that neighboring pixels with similar colors probably have similar alpha values. Thus, early color sampling-based approaches utilize a straightforward way to consider a local region around each unknown pixel. Through sampling neighboring known foreground and background colors, and analyzing their color distributions, the color samples are involved in estimating the true foreground and background colors (F_z and B_z) of the unknown pixel (I_z). When the local correlation

between alpha parameter and color samples is built successfully, the final matte solution is easily determined. Besides constructing the correlation, recent methods have paid more attention to better-selecting color samples.

Although the fundamental principle looks simple, researchers have been working on several challenges as follows:

- Collecting trustworthy samples, in other words, how to decide sampling regions.
- Define the reliability of the collected samples, and how many samples are needed to be collected.
- Constructing the correlation between the alpha value and the samples, and how to evaluate the reliability of the F_z and B_z estimated from the chosen samples.

Around 2000, some classical color sampling-based approaches [45] [46] tended to use low order parametric statistic models, like Gaussians, to measure the unknown pixel's distances to the known samples' foreground and background color distributions. The estimated distances were directly related to the desired alpha value. Hence, such methods including Ruzon and Tomasi's method [45] and Bayesian matting [46] are further thought as parametric sampling methods.

Ruzon and Tomasi's method

Ruzon and Tomasi [45] combined the alpha parameters with the probability distributions of foreground and background color samples. It was done by connecting the composite color to the known samples.

As shown in Fig. 2.4 (a), the unknown region is divided into several parts, and the samples enclosed by a rectangular window comprising of foreground, background, and unknown elements. Among the collected samples, the foreground and background elements are modeled via Gaussian probability functions ($P(F)$ and $P(B)$) whose symmetric axis is

parallel to the coordinate axis. According to an estimated alpha values, another probability function $P(I)$ is modeled for the unknown pixel I, which is an intermediate distribution linearly interpolating the mean and covariance values between corresponding functions $P(F)$ and $P(B)$. In this case, the optimal alpha value is the one that yields the distribution where the observed color has the maximum probability. When the alpha value of an unknown pixel I is determined, its foreground and background colors are calculated by interpolating the means of foreground and background Gaussian pairs.

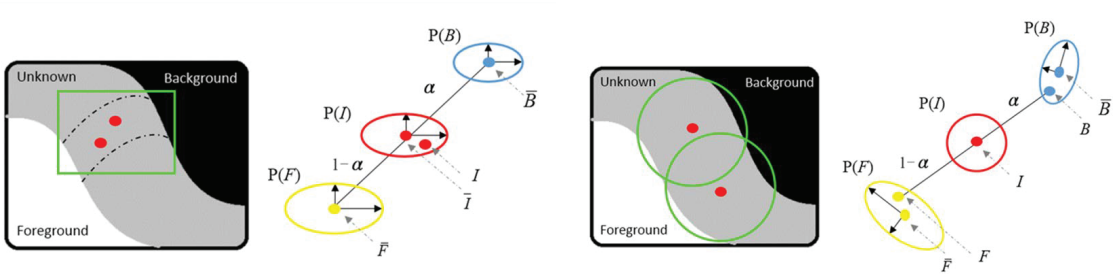


Figure 2.4: Parametric sampling methods and estimations of alpha values. (a) Ruzon and Tomasi's method [45]. (b) Bayesian matting [46].

Bayesian matting

In Bayesian matting [46], Chuang *et al.* also modeled foreground and background colors as mixtures of Gaussians. Their refinements based on Ruzon and Tomasi's work may be listed in the following aspects:

- Instead of a rectangular local window, a continuously sliding window is used to collect nearby samples, which marches inward from the boundaries.
- The model not only utilizes the chosen foreground and background samples, but also uses neighboring candidates for each unknown pixel I (F_I, B_I and α_I) to contribute to the Gaussian functions.

- Formulate the matting problem in a Bayesian framework and provide a Maximum A Posteriori (MAP) method to produce the matte.

The illustration of the color sampling and alpha estimation are presented in Fig. 2.4 (b). Mathematically, given an unknown pixel I, its alpha solution (F, B, and α) is estimated by maximizing the probability

$$\arg\max_{F,\ B,\ \alpha} P(F,\ B,\ \alpha|I) = \arg\max_{F,\ B,\ \alpha} L(I|F,\ B,\ \alpha) + L(F) + L(B) + L(\alpha) \tag{2.11}$$

where $L(\cdot)$ is the log likelihood $L(\cdot) = \log P(\cdot)$, and $L(\alpha)$ is treated as a constant. The first term is measured as

$$L(I|F,\ B,\ \alpha) = -\frac{\parallel I - \alpha F - (1-\alpha)B \parallel^2}{\sigma^2} \tag{2.12}$$

where σ is the local color variance. The rest terms $L(F)$ and $L(B)$ are treated in a similar way via corresponding samples. After the selected foreground colors being partitioned into groups, the Gaussian function in each group has the mean $\overline{F}$ and covariance Σ_F. Then they can be defined as

$$L(F) = -\frac{(F-\overline{F})^T \Sigma_F^{-1} (F-\overline{F})}{2} \tag{2.13}$$

Similarly, the selected background colors are defined as

$$L(B) = -\frac{(B-\overline{B})^T \Sigma_B^{-1} (B-\overline{B})}{2} \tag{2.14}$$

The specific steps solving the Eq. (2.11) are accomplished by iteratively estimating three items constituting the solution. Firstly, fix α to calculate F and B, and then fix F and B to calculate α. When it comes to multiple sample clusters, the iterations first process each pair of foreground and background clusters, and then choose the pair that supplies the maximum likelihood as the final result. Consequently, the solution to α is given in Eq.(2.15), which has been adopted by almost all color sampling-based methods later.

$$\alpha = \frac{(I-B)\cdot(F-B)}{\parallel F - B \parallel^2} \tag{2.15}$$

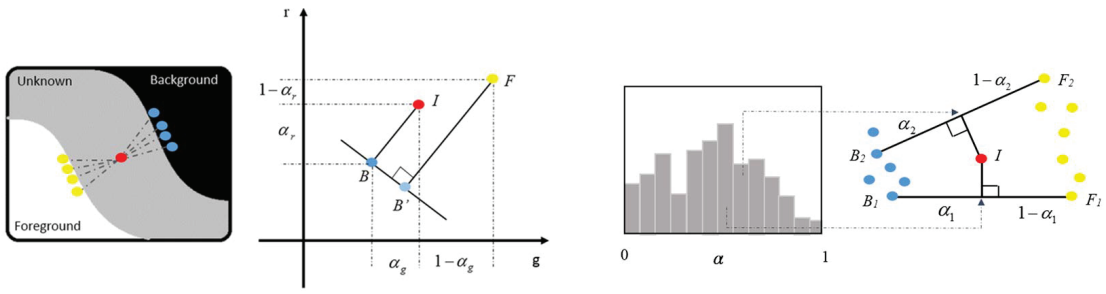
Parametric sampling methods can yield good mattes for the images with distinct foreground and background color distributions, while they usually fail in the case that color distributions are unsuitable for Gaussian Mixture Models (GMMs). Therefore, many approaches [15] [47] use nonparametric approaches to avoid this problem.

Knockout matting

Knockout method [47] [48] extrapolates foreground and background colors into the unknown region (see Fig. 2.5 (a)). Given an unknown pixel I, the corresponding foreground color F is treated as a weighted sum of the collected nearby foreground samples in the trimap. Here the weights are proportional to the spatial distances between the chosen samples and I. The background color B' is estimated by background samples B in the same way. Then, it is refined by comparing the relative positions of I and F. As a result, the alpha component with respect to each color channel is estimated by

$$\alpha_i = \frac{I_i - B'_i}{F_i - B'_i} \tag{2.16}$$

where $i \in \{R,\ G,\ B\}$ represents the three channels in RGB space. At last, final α is calculated as a weighted sum of the three components in which the weights are proportional to the differences between F and B in each channel.



(a) (b)

Figure 2.5: Non-parametric sampling methods and estimations of alpha values. (a) Knockout matting [47] [48] . (b) The iterative method [15].

The early color sampling-based approaches collect color samples through local windows. However, facing the challenges such as a rough trimap, a lack of valid scribble information, or an amount of complex patterns including foreground and background regions, the samples limited in a local window may have large color variances. Consequently, such samples within a window lead to degrading the quality of the alpha matte. Therefore, more non-parametric methods [15] [49] [50] have been developed aiming to collect the optimal samples for unknown pixels.

Iterative optimization approach

In 2005, Wang and Cohen proposed an iterative optimization approach [15] which used a histogram of discrete alpha levels to evaluate collected samples. In this approach, a histogram is generated by scanning all foreground and background sample pairs. For each sample pair, all possible values are involved in contributing to the synthetic colors. If a level of alpha values produces a color which is closest to the unknown pixel, this level is selected. Accordingly, the corresponding bin is increased by one (see Fig. 2.5 (b)). Mathematically, given k possible alpha values $\alpha_1,\ \alpha_2,\ \cdots,\ \alpha_k$ from the continuous range [0,1], the likelihood for level α_k is defined as

$$L_k(I) = \frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N} w_i^F w_j^B \ \cdot\ \exp(-d(I,\hat{I})^2/2\sigma^{k^2}) \tag{2.17}$$

where F_i and B_j are selected color samples for unknown pixel I, $\hat{I}$ is the the composited color defined as $\alpha_k F_i + (1-\alpha_k)B_j$, $d(\cdot)$ is the Euclidian distance in RGB space, and σ^k is the local variance defined as $\alpha_k\sigma_F + (1-\alpha_k)\sigma_B$, in which σ_F and σ_B are foreground and background samples' variances.

Robust color matting

Later, Wang and Cohen developed a robust color sampling method [41] to avoid the problems caused by local windows. In this method, the searching range is expanded to collect

color samples along the boundaries of the unknown region so as to improve the samples diversity (see Fig. 2.7 (a)). Furthermore, in order to boost the robustness of the samples, trustworthy samples are recognized by the property that the color of the unknown pixel could be convexly combined with the trustworthiness. In other words, for each unknown pixel, only the foreground and background sample pairs with high confidence values would be adopted. Here, given a sample pair (F, B) for an unknown pixel I, the confidence term is determined by

$$f(F,B) = \exp\big(\frac{-R(F,\ B)^2 \cdot w(F) \cdot w(B)}{\sigma^2}\big) \tag{2.18}$$

where σ is fixed to a constant with 0.1. $R(F,\ B)$ is the distance ratio which is defined as

$$R(F,B) = \frac{\parallel I - (\alpha F + (1-\alpha)B) \parallel}{\parallel F - B \parallel} \tag{2.19}$$

where α is a particular result for the unknown pixel I with the sample pair (F, B). The two weights are color similarities which are treated as

$$\begin{aligned} w(F) &= \exp\big(-\frac{\parallel F - I \parallel^2}{D_F^2}\big) \\ w(B) &= \exp\big(-\frac{\parallel B - I \parallel^2}{D_B^2}\big) \end{aligned} \tag{2.20}$$

where D_F and D_B are the minimum distance between I and the foreground and background samples respectively.

Based on the robust sampling method [41], an improved sampling method [51] employs geodesic distance in the sampling procedure. Replacing Euclidean spatial distance, the geodesic distance is defined as the shortest path on a weighted graph from the unknown pixel to the boundary of the unknown region. And its weights are set by the probability that a pixel belongs to the foreground or background region. Fig. 2.6 from paper [51] explains the difference between geodesic samples (in yellow) and spatial samples (in blue).

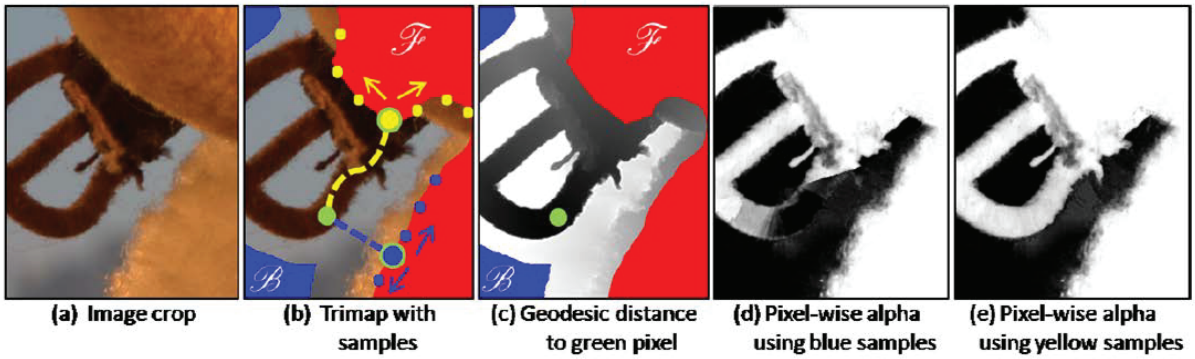


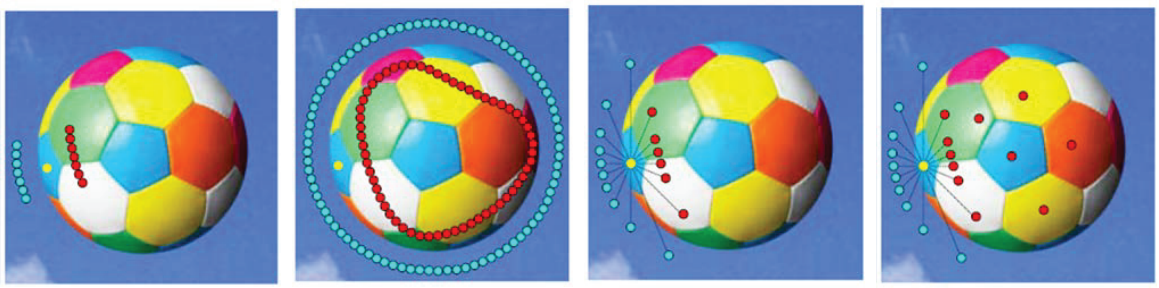
Figure 2.6: Collecting color samples using the improved sampling method [51].

Global sampling and Ray-based matting

On the other hand, [50] introduced a global sampling approach to form a comprehensive sample set. It is accomplished by collecting all pixel colors on the unknown region boundary to make color samples as sufficient as possible (see Fig. 2.7 (b)). However, it gives rise to a huge computation and time consumption as the number of samples increases.

In terms of time consumption, Gastal and Oliveira introduced a ray-based sampling for real time alpha matting in paper [49]. In this approach, a number of rays emit along with different directions from an unknown pixel. Only the samples located at the regions where rays intersect with the boundary of the known parts would be selected (see Fig. 2.7 (c)). Whatever, some true samples may be neglected since the sampling step is strongly related to the unknown regions' shapes in the trimap.

Observing the sampling approaches above, they have a common problem that the quality of the matte strongly depends on the true colors F and B for unknown pixels being collected or not. Their searching areas are usually along with the unknown region boundary, so that the shapes of the unknown regions in the trimap play a significant role in the sampling process. Moreover, due to considering color information only, they fail to pull a good matte from the regions in which both foreground and background color distributions overlap.



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

Figure 2.7: Different color sampling strategies. (a) Robust [41]. (b) Global [50]. (c) Ray-based [50]. (d) Texture [27]. The figures are from [27].

Comprehensive sampling

To overcome these disadvantages, weighted color and texture information are taken into account in more comprehensive color sampling approaches [27] [52] [28]. In [27] [52], in addition to color values, texture is provided as a feature to boost the discrimination ability for similar colors in known regions. Same as the contribution of color, the contribution of texture is automatically computed by analyzing the content of the image. In order to avoid true samples being left out of the color sample sets, a local sampling is then combined with a global sampling scheme (see Fig. 2.7 (d)). Besides, an objective function consisting of color and texture is optimized to choose the best sample pair (F, B) representing an unknown pixel's foreground and background colors.

Similar to [27] [52], a more comprehensive and representative sample set is built in [28]. To ensure samples from each color distribution being collected, this method expands the sample searching range farther from the boundary of foreground or background regions. Thus, it alleviates the color distribution overlap of foreground and background regions by keeping correct samples from being missed. To handle the overlap problem further, the best sample pair (F, B) for estimating α is the one which is produced from the minimum overlapping color distributions of foreground and background clusters. The selection is accomplished via a brute-force optimization of an objective function which takes three

terms into account. It is defined as

$$O(F_i, B_i) = K(F_i, B_i) \times S(F_i, B_i) \times C(F_i, B_i) \tag{2.21}$$

where K is chromatic distortion, S and C are spatial and color statistics of the image. i indicates the index of the sample candidates.

Sparse coding matting

Recently, [53] introduces a novel color-sampling based approach that capitalizes on a sparse coding of pixel features. Here the feature vector is a 6D vector $[R, G, B, L, a, b]$ that consists of the concatenation of the RGB and CIELAB color spaces respectively. Given an unknown pixel i and its sample dictionary $D = [F_1, \cdots, F_n, B_1, \cdots, B_n]$, the spare code for i is treated as

$$\beta = \arg\min \parallel v_i - D\beta_i \parallel_2^2 \qquad s.t \quad \parallel \beta_i \parallel \leq 1; \quad \beta_i \geq 0 \tag{2.22}$$

where v_i is the feature vector at i. The sparse codes with respect to the atoms in the foreground dictionary are summed to determine the alpha value as

$$\widehat{\alpha} = \sum_{p \in F} \beta^{(p)} \tag{2.23}$$

In addition, since it is not easy to constitute a feature vector for an unknown pixel in the overlap region having both foreground and background colors, a probabilistic segmentation is adopted to label the pixel with low or high certainty in this method.

Generally, in terms of color sampling-based system, the chosen samples can supply intuitive color information to unknown pixels and take a good effect in the images containing smooth regions. However, these kind of approaches may fail in images that have complicated color content and highly textured regions resulting from neglecting correct samples in the searching area. Therefore, how to efficiently collect and utilize color samples is the core problem for these methods. As inspired by the employments of geodesic distances or texture data, color information and Euclidean distance might not be the only way to tackle

the difficulties of color overlap and sample collection. Introducing new measurements could also lead to effective solutions. In addition, collecting and using samples is still influenced by valid information from scribbles or trimaps.

2.2.2 Propagation-based approaches

In order to solve the drawback of color sampling-based system, some approaches [54] [55] [25] [26] have explored another matting technique based on local image statistics and propagation between adjacent pixels, which are classified into propagation-based matting system. Their basic idea is defining an $affinity$ within a small window (usually 3×3) where the pixel correlations are strong and keep local smoothness. The various affinities between neighboring pixels model the *matte gradient* throughout the image lattice rather than estimating each pixel's alpha value directly.

Compared with color sampling-based approaches, propagation-based approaches require fewer user input and avoid matte discontinuities especially in the region with complex color distributions. Almost all such methods make their efforts to resolve the two issues:

- Defining the affinities between pixels.
- Modeling the matte gradient to propagate alpha values from known regions to unknown regions.

Poisson matting

In 2004, Sun *et al*. proposed Poisson matting [54] assuming that the color intensities of foreground and background change locally smooth. Founded on the assumption, given the original color, foreground and background color samples for an unknown pixel (I, F,B), the matte gradient can be modeled mathematically as:

$$\bigtriangledown \alpha_z \approx \frac{1}{F_z - B_z} \bigtriangledown I_z \tag{2.24}$$

where $z = (x, y)$ indicates the image coordinate, and $\bigtriangledown = (\frac{\partial}{\partial x}, \frac{\partial}{\partial y})$ is the gradient operator. That is to say the matte gradient is approximately proportional to the image gradient. The variational problem is further minimized by Eq.(2.25) in an unknown region Ω with the Dirichlet boundary condition that is consistent with the input trimap.

$$\alpha^* = \arg\min \int\int_{z\in\Omega} \parallel \bigtriangledown \alpha_z - \frac{1}{F_z - B_z} \bigtriangledown I_z \parallel^2 dz \tag{2.25}$$

The associated *Poisson equation* demanding the same boundary condition is defined as

$$\Delta\alpha = div(\frac{\bigtriangledown I}{F - B}) \tag{2.26}$$

where $\Delta = (\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2})$ is Laplacian operator and div is Divergence operator. In addition, the Gauss-Seidel iteration with the overrelaxation method is used to compute the optimal solution of Poisson equation, and both $(F - B)$ and $\bigtriangledown I$ are processed in the grayscale channel even for color images.

Propagation between adjacent pixels is a feature of Poisson matting, but it also is a restriction to estimate the values of F and B via the local samples enclosed in the small window only. Consequently, it leads to dealing with complex images imprecisely.

Random walk

Grady *et al.* proposed a Random Walk matting approach [55] in 2005. This approach calculates the probabilities of random walkers that start at each unknown pixel, reaching a pixel in the foreground before striking pixels in the background. The probability of first arriving walker is adopted as the alpha value of the unknown pixel. In particular, given a weighted graph where each pixel is treated as a node, there are four edges connecting each pixel and its neighbors. The probability that a random walker at i transits to j is calculated as

$$p_{ij} = \frac{w_{ij}}{d_i} \tag{2.27}$$

where d_i is the sum of four weights of the edges from pixel i, and the weight w_{ij} is defined as

$$w_{ij} = \exp\big(-\frac{\parallel I_i - I_j \parallel^2}{\sigma^2}\big) \tag{2.28}$$

where σ is a free parameter.

Note that instead of RGB channels, the color distance in Eq. (2.28) is measured in the channels created by Local Preserving Projections (LPP) algorithm [56]. The projections modified by LPP method are changed to solve a generalized eigenvector problem:

$$ZLZ^T x = \lambda ZDZ^T x \tag{2.29}$$

where Z is a $3 \times N$ matrix with each I_i vector as a column, D is a diagonal matrix with the corresponding element d_i, and L is the graph Laplacian matrix defined by

$$L_{ij} = x = \begin{cases} d_i & (i = j) \\ -w_{ij} & (i \; and \; j \; are \; ajacent \; pixels) \\ 0 & (otherwise) \end{cases} \tag{2.30}$$

Denoting the solution to the Eq. (2.29) by Q, each row of which is an eigenvector, the weight of Eq. (2.28) is resolved by

$$w_{ij} = exp\big(\frac{(I_i - I_j)^T Q^T Q (I_i - I_j)}{\sigma^2}\big) \tag{2.31}$$

In addition, the desired random walker probabilities can be treated as the solution to the combinatorial Dirichlet problem. Consequently, Grady further set the random walk problem as an inhomogeneous Dirichlet problem [57]. The final result can be efficiently implemented by solving a single system of linear equations assisted by a graphics processing unit (GPU). In [57], founded on an exact trimap, it takes about 0.5 seconds to generate the alpha matte for a 1024×1024 image. Nonetheless, over-smoothing is often observed in the output mattes using this approach.

Closed-form matting

As another classical propagation-based matting technique, closed-form matting [25] computes the optimal solution to minimize a cost function based on a local smoothness assumption on colors. The *color line model*, which is the underlying assumption in this method, assumes that both foreground and background colors rarely change over a small window around each pixel. That means the foreground values F of all pixels in a small window lie on a single line, as well as the same situation of the background values B. Mathematically, for an image with N pixels, that means each F and B can be written as a linear mixture of two colors expressed as

$$\alpha_i = \sum_c a^c I_i^c + b, \ \forall i \in w \tag{2.32}$$

where c denotes color channels in RGB color space, and a^c and b are constants in the small image window w. Then the cost function is defined as

$$J(\alpha, a, b) = \sum_{j \in I} (\sum_{i \in w_j} (\alpha_i - \sum_c a_j^c I_i^c - b_j)^2 + \epsilon \sum_c a_j^{c^2}) \tag{2.33}$$

where w_j is a small window around pixel j. It is the goal of closed-form algorithm to find α, a and b that can minimize the function. Moreover, a_j^c and b_j from Eq. (2.33) can be eliminated, and accordingly yield a quadratic cost with α only:

$$J(\alpha) = \alpha^T L \alpha \tag{2.34}$$

where α is an $N \times 1$ vector and L is an $N \times N$ matrix called *matting Laplacian*. The matting Laplacian is defined as

$$L(i,j) = \sum_{k|(i,j) \in w_k} \big(\delta_{ij} - \frac{1}{|w_k|}\big(1 + (I_i - \mu_k)^T \big(\Sigma_k + \frac{\epsilon}{|w_k|} I_3\big)^{-1} (I_j - \mu_k)\big)\big) \tag{2.35}$$

Under the user-supplied constraints like a trimap, the optimal alpha values are further solved by

$$\alpha = \arg\min \alpha^T L \alpha, \quad s.t. \ \alpha_i = 1 \ or \ 0, \quad \forall i \in \Omega \tag{2.36}$$

where Ω represents the known pixels in the trimap or scribbles.

The matting Laplacian L described in Eq. (2.35) is a vital contribution of closed-form algorithm since it has attracted a lot of attention from researchers. A number of representative approaches have been proposed based on it.

Spectral matting

Spectral matting [42] is an automatic matting approach extended from the closed-form method. In this approach, the input image is assumed as a convex combination of K layers as

$$I_z = \sum_{k=1}^{k} \alpha_z^k F_z^k \tag{2.37}$$

where the vector α_z^k is the matting component that specifies the fractional contribution of the kth image layer F_z^k at each pixel. Moreover, founded on the fact that the smallest eigenvectors of matting Laplacian L can capture image information about the fuzzy cluster of pixels, even without any user-specified constraints, this method demonstrates that the smallest eigenvectors span the individual matting components of the image. Therefore, the task of obtaining the matting components is equivalent to finding solutions to the eigenvectors. When the solution is deduced, a foreground matte is extracted even with less supervised information from users.

KNN matting

In order to make the local lattice fit the color line model, closed-form [25] and spectral matting [42] methods usually use a 3×3 small window size to achieve good performances. But it may give rise to small gaps between foreground and background especially in the regions with complicated scenes. KNN matting [26] does not assume the local color line model and capitalizes on the K nearest neighbors (KNN) in matching nonlocal neighborhoods. It assumes that pixels with similar colors and texture appearances can be expected to share similar alpha values. Hence, given a pixel i, the corresponding feature vector $X(i)$

is specified as

$$X(i) = (\cos(h), \sin(h), s, v, x, y)_i \tag{2.38}$$

where h, s, v are the respective HSV coordinates. (x, y) are the spatical coordinates of pixel i, and they are involved in the feature vector to reinforce spatial coherence. In this case, the affinity between two pixels can be constituted by a kernel function expressed as

$$k(i, j) = 1 - \frac{\parallel X(i) - X(j) \parallel}{C} \tag{2.39}$$

where C is the least upper bound of $\parallel X(i) - X(j) \parallel$ to let $k(i, j) \in [0,\ 1]$. At last, a closed-form solution which leverages on the preconditioned conjugate gradient method implements an alpha matte efficiently.

Furthermore, compared with previous algorithms using RGB color space, it is easy for KNN matting to extract $n \geq 2$ multiple layers dealing with SVBRDF or high dimensional data in non-RGB space. It performs outstandingly especially for the image with highly textured regions, or the image whose trimap has some hollow known regions, while its performance lacks of smoothness for the fuzzy regions.

Other nonlocal Laplacian-based methods

To solve the drawback caused by fixed window size, a fast matting approach with the adaptive window size is presented in [58]. In this method, a large kernel matting Laplacian is utilized to improve the matte quality and speed up the propagation. Also, an adaptive kernel size by a KD-tree trimap segmentation technique is used to save running time. However, using a large window size is more likely to go against the color line model.

On the other hand, a novel nonlocal matting technique which overcomes the local limit of the color line model is provided in [16]. Relying on the nonlocal principle which is first introduced in [59] and assuming that a denoised pixel could be treated as a weighted sum of the pixels with similar appearances, the fundamental idea is expanded to employ a weighted sum of nonlocal alpha values to image matting. Consequently, the pixel affinity

in the matting Laplacian of this approach involves nonlocal neighboring pixels, in contrast to only the spatial neighbors being taken into account in the closed-form matting.

In conclusion, the propagation-based matting approaches work more remarkably than color sampling-based approaches in the cases of input images having intricate details and textured regions with complex intensity variations. This kind of approaches models the local affinity and propagate alpha values between neighboring pixels, which can lead to satisfactory alpha mattes. But if the color smoothness assumption is not suitable for target images, their performances may be unsatisfactory due to the negative effects on affinities from the assumption.

2.2.3 Combination of color sampling and propagation

Depending on the observation that both color sampling-based and propagation-based matting approaches have respective advantages and disadvantages, some algorithms [41] [28] [60] tend to combine the two types of techniques into one energy function and solve it in a global optimization process.

Joint methods based on graph cut segmentation

Graph cut segmentation method [61] treats an image as a graph where pixels are represented by nodes. The segmentation problem is reformulated to find the smallest cut of the graph by labeling pixels. A generalized energy function taking the form of a "Gribbs" energy is expressed by two terms:

$$E(A) = \lambda U(A) + V(A) \tag{2.40}$$

where A represents how to label pixels, and λ is a weight balance. $U(A)$ is called data term that stands for region characteristics and the semantic goal of the optimization. $V(A)$ is called neighborhood term that stands for the coherence between pixels with similar colors.

Inspired by [61], Grubcut matting [60] models alpha values by soft step functions under the help of a new trimap got from the binary segmentation. The solution is further estimated by minimizing the function as

$$E = \sum_{i \in \Omega} U_i(\alpha_i) + \sum_{t=1}^{T} V(\Delta_t, \sigma_t, \Delta_{t+1}, \sigma_{t+1}) \tag{2.41}$$

where t indicates each segment, Δ and σ are the parameters of the step function of a segment. Before this energy function, similarly to some color sampling-based methods, the distributions of the foreground and background colors are modeled as GMMs. Each unknown pixel is matched with a GMM of foreground or background, and thus the data term U denotes the corresponding Gaussian probability of the alpha value. Besides, the neighborhood term V is defined as

$$V(\Delta_t, \sigma_t, \Delta_{t+1}, \sigma_{t+1}) = \lambda_1(\Delta_t - \Delta_{t+1})^2 + \lambda_2(\sigma_t - \sigma_{t+1})^2 \tag{2.42}$$

Note that though the alpha values' smoothness benefits from Eq.(2.42), its effort may be negligible when the soft step function does not guarantee the complicated boundary regions in the image.

Joint methods based on Matting Laplaican

As discussed in Section 2.2.1, for some color sampling-based methods such as the ones in [41] [50], the preliminary alpha value $\widehat{\alpha}$ of an unknown pixel can be estimated via its selected color samples (F, B) using Eq. (2.15). Here we explain that the quality of the alpha matte can be further enhanced by combined techniques.

In a specific way, $\widehat{\alpha}$ is assigned to the data term, and the matting Laplacian proposed in [25] is associated with the neighborhood term. In robust color matting [41], the energy function can be formulated as

$$E(\alpha) = \sum_{i \in I} \left(\widehat{\eta}_i (\alpha_i - \widehat{\alpha}_i)^2 + (1 - \widehat{\eta}_i)(\alpha_i - \ \omega(\widehat{\alpha}_i > 0.5))^2 \right) + \lambda \cdot J(\alpha, a, b) \tag{2.43}$$

where $J(\alpha, a, b)$ is described by Eq.(2.33), the left summation and the right product terms represent the data and neighborhood terms respectively. The trade-off between the two terms is implemented by the weight parameter λ. Besides, α and $\widehat{\eta}$ are alpha and confidence values in the color sampling process, and ω stands for the Boolean operation in which only 0 or 1 will be returned. For the data term only, this function prefers the alpha value estimated by the color sample if the confidence value $\widehat{\eta}$ is high, otherwise, the pixel is more likely to be definite foreground or background.

Following this scheme, a number of later approaches [49] [50] [51] make use of a novel objective function which is usually defined as Eq.(2.44) as a post-processing.

$$E(\alpha) = (\alpha_i - \widehat{\alpha}_i)^T D(\alpha_i - \widehat{\alpha}_i) + \lambda \cdot J(\alpha, a, b) \tag{2.44}$$

where D is an $N \times N$ diagonal matrix, whose diagonal elements are a relatively large number for known pixels and the confidence values for unknown pixels.

Furthermore, in [53], besides employing an energy function to smooth out the matte like [27] [28], a graph-based optimization is used to obtain the final matte. Particularly, it combines the initial sparse coded $\widehat{\alpha}$ (Eq. (2.23)) with K-nearest neighbors and the smoothness term involved by matting Laplacian to be the data term in a graph model. And then a sparse system of linear equations can be processed in the closed-form like before.

2.2.4 Learning-based approaches

The technique of machine learning has also cast some new sights into handling the image matting. Such approaches as [29] [62] [63] usually generate an alpha matte via training samples in a learning-based framework.

In [29], the matting problem is considered as a semi-supervised learning task. Zheng *et al*. provided a global learning based matting method with trimaps and a local learning based matting method with scribbles.

For either local or global learning procedure, its aim is to obtain a general alpha-color model which can be linear or nonlinear. In the linear case for local learning, given a data vector $\mathbf{x}' = [\mathbf{x}^T\ 1]^T$, the linear alpha-color model is written as

$$\alpha = \mathbf{x}^T\beta + \beta_0 = \mathbf{x}'^T \begin{bmatrix} \beta \\ \beta_0 \end{bmatrix} \tag{2.45}$$

where $\beta = (\beta_1, \beta_2, \cdots, \beta_d)$ and β_0 are the model coefficients. Besides, the complete set of image pixels is denoted by $\Omega = 1, 2, \cdots, n$. For any pixel $i \in \Omega$, we have its neighboring pixels denoted by $N_i = \tau_1, \cdots, \tau_m$ and $\alpha_i = (\alpha_{\tau_1}, \alpha_{\tau_2}, \cdots, \alpha_{\tau_m}^T)$, where $\tau_j \in N_i$ denotes the vector of alpha values of N_i. What is more, we denote the new notation $\mathbf{X}_i = [\mathbf{x}'_{\tau_1}, \cdots, \mathbf{x}'_{\tau_1}]^T$, which is a matrix with the size $m \times (d+1)$ and is stacked by the values of the pixels in N_i. Then the β and β_0 can be obtained by solving a quadratic optimization problem:

$$\arg\min_{\beta,\beta_0} \left\| \alpha_\mathbf{i} - \mathbf{X}_i \begin{bmatrix} \beta \\ \beta_0 \end{bmatrix} \right\|^2 + \lambda \begin{bmatrix} \beta \\ \beta_0 \end{bmatrix}^T \begin{bmatrix} \beta \\ \beta_0 \end{bmatrix} \tag{2.46}$$

where λ is a parameter. Finally, the optimal solution of Eq. (2.45) is derived as

$$\alpha = \mathbf{x}'^T \mathbf{X}_i^T (\mathbf{X}_i \mathbf{X}_i^T + \lambda \mathbf{I}_m)^{-1} \alpha_i \tag{2.47}$$

where $\mathbf{I}_m$ is a $m \times m$ identity matrix.

When it comes to the nonlinear alpha-color model, Eq. (2.45) can be reformulated to Eq. (2.48) below with the help of the kernel trick. It is done by using a feature vector $\Phi(\mathbf{x}) \in \mathbb{R}^p$ to replace $\mathbf{x} \in \mathbb{R}^d$, where $p > d$ and Φ is a nonlinear map function.

$$\alpha = \Phi(\mathbf{x})^T \beta + \beta_0 \tag{2.48}$$

where $\beta = [\beta_1, \cdots, \beta_p]^T$ and $\Phi(\mathbf{x}) = [\phi_1(\mathbf{x}, \cdots, \phi_p(\mathbf{x})]^T$.

Also, a global learning method is presented in this method. For each unknown pixel, all known pixels around it would be collected no matter foreground or background pixels. Similarly to the local learning procedure, through weighting them via the weighted ridge

regression technique [64], the global alpha-color model is obtained from training, as well as the alpha values.

Nowadays, a number of learning-based approaches have been implemented in image matting. In [62] [63], the matting problem is treated as a nonlinear regression issue. It capitalizes on Support Vector Regression (SVR) to learn the spatially shifting relations between pixel properties and alpha values. [65] resolves the matting problem into a transductive inference model. In order to avoid over-smoothness phenomenon, an iterative transductive learning approach [66] is proposed for alpha matting. This approach uses a new objective function of matting by considering the neighboring pixels' consistency and the influence of both labeled and unlabeled regions.

In general, the learning-based approaches rely on the learning parameters but prior assumptions to train samples and generate final mattes. Therefore, a number of high-quality samples and features are required for training. Actually, different features may give a quite another result. Unfortunately, it is still hard to conclude how to choose features mathematically. This is because the weights of features always vary to a large extent with the change of input images, even for different regions in the same image. That may be a reason for the fact that relatively fewer researchers pay attention to this kind of approaches instead of the previously reviewed matting algorithms.

2.3 Stereo matching

When it comes to multiple view stereo images, stereo correspondence system has been an energetic and fruitful research area. In this section, we will review typical two-frame stereo matching approaches via a taxonomy of them.

2.3.1 Basic concepts

Most existing stereo matching methods compute a univalued disparity map $d(x, y)$ as an output, and their central concept is the *disparity space* (x, y, d). The term *disparity* is first proposed in [22] to stand for the difference in location of corresponding features captured by left and right eyes. Now it is usually equivalent to the term *inverse depth* in literature on computer vision.

The stereo images used in this thesis are shot on a linear path with the optical axis, so we have a classical inverse-depth interpretation [67] here. The correspondence between a pixel (x, y) in a reference image r and a pixel $(x^{'}, y^{'})$ in a matching image m is defined as

$$x' = x + sd(x, y), \qquad y' = y \tag{2.49}$$

where $s = \pm 1$ is a sign chosen, (x, y) and (x', y') are spatial coordinates. Note that the direction of the pixels moving along with is opposite to the that of matching images. For instance, if the images are from left to right, the pixels move from right to left, and vice versa.

After that, a disparity space image (DSI) [68] is output, which is defined across a continuous or discretized version of disparity space (x, y, d). In addition, the DSI usually refers to the confidence or log likelihood of a specific match according to disparity map $d(x, y)$ actually. As shown in Fig. 2.8 (c), the dark bands indicate the regions are matched at this disparity, and the smaller darker ones are often the result of textureless regions.

Therefore, the stereo correspondence problem is treated as a synonymous process of generating a univalued function in disparity space. Depending on the function, it is accomplished to describe the shapes of the surfaces in the scene as accurately as possible. In other words, the problem of finding a surface embedded in the disparity space can be solved by the optimization function concentrating on the best smoothness and lowest cost.

(a) (b) (c)

Figure 2.8: An example for stereo images and disparity map from [69]. (a) Left view image. (b) Right view image. (c) The disparity map based on the left view.

2.3.2 Stereo matching approaches

For most stereo matching approaches, they comprise of the four generalized parts as follows [70]:

1. Computing the matching cost;
2. Aggregating the cost;
3. Computing and optimizing the disparity;
4. Refining the disparity.

The actual sequence of the parts above may be permutated and combined differently for different algorithms. For instance, some approaches estimate the disparity at a given pixel only relying on the intensity values with a finite window. Accordingly, these approaches are referred to as local-based algorithms, and they often choose implicit smoothness assumptions as an aggregating support. On the other hand, global-based approaches adopt explicit smoothness assumptions and solve an optimization problem. Skipping the aggregation, they prefer to seek a disparity assignment that minimizes a global cost function.

In terms of matching cost computation, squared intensity differences (SD) [71] [72], absolute intensity differences (AD) [73] are the most common terms of the pixel-based

matching costs. When they are extended to the video processing field, the criteria are the mean squared error (MSE) and mean absolute difference (MAD). Moreover, some robust measures such as contaminated Gaussians and truncated quadratics are introduced in [74] [75]. Besides, [76] presents another matching cost that is insensitive to image sampling. Rather than just comparing pixel values shifted by integral amounts, it compares each pixel in the reference image against a linear function of the other image. As a result, an initial disparity space image $C_0(x, y, d)$ is formed by the disparities and the matching costs across all pixels.

As discussed before, local-based approaches tend to involve the cost aggregation that is implemented by averaging or summing on a support region from the disparity space image $C(x, y, d)$. Traditionally, a support region is a square window or a Gaussian convolution region. Recently, researchers have expanded it to variable forms such as removable windows [77], adaptive size windows [78] [79], and specialized windows based on connected components of constant disparity [80]. Mathematically, the aggregation over a fixed support region can be expressed in the form of convolution as

$$C(x, y, d) = w(x, y, d) * C_0(x, y, d) \tag{2.50}$$

On the other hand, considering global-based approaches, they emphasize almost all their works on the disparity computation and global optimization. A classical energy minimization is proposed in [81]. The assignment is to find a disparity function d to minimize a global energy constituted by a data term E_{data} and a smoothness term E_{smooth}:

$$E(d) = E_{data}(d) + \lambda E_{smooth}(d) \tag{2.51}$$

$E_{data}(d)$ determines the extent of the disparity d agreeing with the input image pair. It is defined as the sum of the initial or aggregated matching cost DSI:

$$E_{data}(d) = \sum_{(x,y)} C(x, y, d(x, y)) \tag{2.52}$$

Besides, the smoothness term relies on the adopted smoothness assumptions in different algorithms respectively, so $E_{smooth}(d)$ is often constrained to only estimate the disparity

differences between neighboring pixels to make the computation tractably.

$$E_{smooth}(d) = \sum_{(x,y)} \rho(d(x,y) - d(x+1,y)) + \rho(d(x,y) - d(x,y+1)) \tag{2.53}$$

where ρ is certain monotonically increasing function of disparity difference.

Moreover, [82] proposes a discontinuity-preserving energy function based on Markov Random Fields (MRFs). [83] uses the spline which is a lower-dimensional representation to act as an alternative smoothness term. [74] explains the procedure of the line processes that is subsumed by a robust regularization framework. Also, $E_{smooth}(d)$ can be computed by Eq. (2.54). Depending on the intensity differences, it can bring the smoothness cost down at high intensity gradients.

$$\rho_d(d(x,y) - d(x+1,y)) \cdot \rho_I(\parallel I(x,y) - I(x+1,y)) \parallel) \tag{2.54}$$

where ρ_I is some monotonically decreasing function of intensity differences. A number of global optimization algorithms [84] [77] [85] applied the fundamental idea of Eq. (2.54) and encouraged disparity discontinuities to cohere with intensity edges.

Once the global energy is decided, a variety of methods such as continue MRFs [86], highest confidence first [87] and mean-field annealing [88] can be used to estimate an optimal disparity solution. Furthermore, some more efficient methods including max-flow and graph-cut are also utilized to solve the novel and special optimization problems [85] [89] [90].

On the other hand, a different category of global-based approaches are founded on dynamic programming, which focuses on independent scanlines in polynomial time to find a global minimum. The emphasises of [77] [91] [92] are extended to the dense scanline through computing the minimum-cost path across a particular matrix. Here the matrix comprises of all pairwise matching costs associated with the corresponding scanlines. Furthermore, some other cooperative methods [75] [93] iteratively compute solutions locally rather than using nonlinear operations, and give rise to an overall behavior like global optimization does.

Last but not least, most stereo correspondence methods estimate disparities in discretized space via the optimization techniques including optical flow [94], but their results appear to be quantized and imperfect. Many of them thus have raised a variety ways to refine the disparities. For instance, based on sub-pixel disparity, [78] chooses the iterative gradient descent with only a little addition of computation. [95] uses the advisability of fitting correlation curves to improve the accuracy after the initial discrete correspondence stage.

2.4 Combination of matting and stereo matching

Most conventional image matting algorithms only use a single image as an input and generate a matte. On the other hand, traditional stereo correspondence approaches seldom solve the matching problem with the help of matting idea. Recently, a few researchers started to try to combine matting and stereo matching procedures. Here we review three joint approaches.

2.4.1 Combination of stereo matching and matting for transparency

Szeliski and Golland [96] formulated and solved a stereo matching and matting problem aiming to transparent objects. Even though the problem was more difficult than the traditional stereo correspondence, Szeliski and Golland provided a principled way of recovering the depth, true color and the opacity at each pixel.

The proposed model can deal with the pixels near depth discontinuities, which mix foreground and background. Also, this approach supplies a solution to frequent phenomenons such as occlusions. By explicitly modeling a 3D (x, y, d) *disparity space* which concludes color and opacity values, a multi-frame stereo algorithm is proposed to predict the appearance of each input image. Meanwhile, with the aid of evidence aggregation and energy

minimization, this method can separate foreground and background objects without blue screen.

Since multi-frame stereo images can increase the stability of the algorithm, Szeliski and Golland formulated their multi-frame stereo problem relying on the concept of a *virtual camera* and a *generalized disparity space* (see Fig. 2.9 (a)). Concretely, the position and orientation of a virtual camera are first determined. It can be coincident with one of the input images or based on practical requirements. After that, the orientation and spacing of the constant d planes, which are called disparity planes, can be defined. Accordingly, the relation between d and 3D space is projective. In addition, a 4D (x, y, d, k) space is supplied to describe multi-frame samples along a fictitious camera dimension (see Fig. 2.9 (b)). For the kth image, the values in a given (x, y, d) cell are considered as the color distributions at a given position in disparity space.

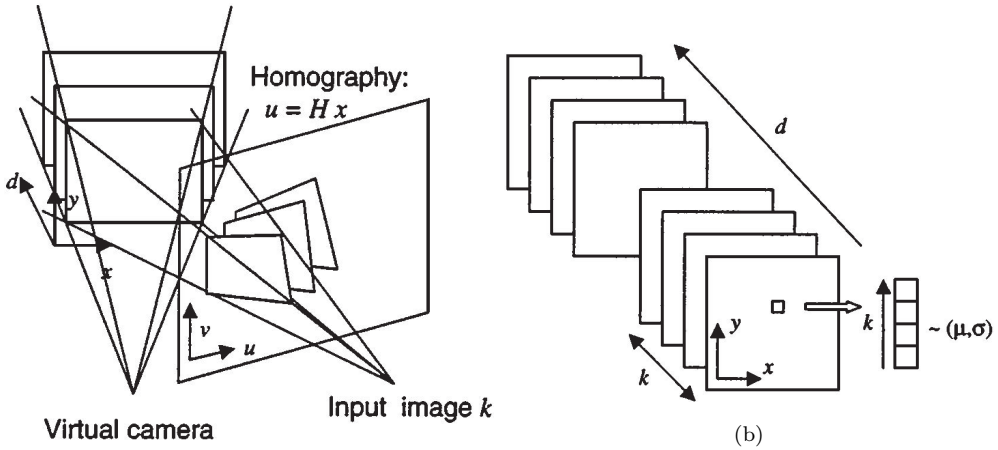


Figure 2.9: Generation of disparity space in Szeliski and Golland's method [96]. (a) 3D (x, y, d) space. (b) 4D (x, y, d, k) space.

To compute the initial evidence for the pixels locating at (x, y, d), the colors of the K input images are first resampled across the entire 4D (x, y, k, d) space as follows:

$$\mathbf{c}(x, y, d, k) = w_f(\mathbf{c}_k(u, v); \mathbf{H}_k + \mathbf{t}_k[0\ 0\ d]) \tag{2.55}$$

where $\mathbf{c}(x, y, d, k)$ represents the pixel mapped into the 4D generalized disparity space, w_f

is the forward warping operator, $\mathbf{c}_k(u, v)$ is the kth input image, and $\mathbf{H}_k + \mathbf{t}_k[0\ 0\ d]$ is the homography mapping of this image to disparity plane d.

Given a (x, y, d) cell, when the color or luminance values are collected, we can compute initial statistics over the K colors, *e.g.*, the mean μ and variance σ^2. However, the local evidence cannnot determine the correct disparities unless each pixel has a unique color. The high variance may be presented at occluded regions where pixels actually belonging to a different disparity level. Thus, local evidence aggregation is used to help disambiguate matches.

Once obtaining an initial (x, y, d) volume which comprises estimated RGBA values (color and 0/1 opacity), the visibilities of this volume are re-projected by the transformation

$$\mathbf{x}_k = \mathbf{M}_k \widehat{\mathbf{M}}_0^{-1} \mathbf{x}_0 \tag{2.56}$$

where $\mathbf{x}_k$ is the image coordinate in the kth image, $\mathbf{M}_k$ is the kth camera matrix, $\widehat{\mathbf{x}}_0$ is a homogeneous coordinate in (x, y, d) space, and $\widehat{\mathbf{M}}_0$ is the complete camera matrix corresponding to the virtual camera. In this method, the (x, y, d) volume is interpreted at different d levels as a set of potentially transparent acetate stacks. Each acetate is warped into a given input frame as follows

$$\mathbf{x}_k = \mathbf{H}_k \mathbf{x}_0 + \mathbf{t}_k d = (\mathbf{H}_k + \mathbf{t}_k[0\ 0\ d])\mathbf{x}_0 \tag{2.57}$$

where $\mathbf{x}_0 = (x, y, 1)$. Moreover, the resampling step for a given layer d into the coordinate set of camera k is defined as

$$\widetilde{\mathbf{c}}_k(u, v, d) = w_b(\widehat{\mathbf{c}}(x, y, d); \mathbf{H}_k + \mathbf{t}_k[0\ 0\ d]) \tag{2.58}$$

where $\widehat{\mathbf{c}} = [r\ g\ b\ \alpha]^T$ is the color and opacity estimated at the location (x, y, d), and w_b is a resampling operation. Furthermore, the resampled layers are used to composite a new image via the standard *over* operator [1] as follows

$$\widetilde{\mathbf{c}}_k(u, v) = \odot_{d=d_{max}}^{d_{min}} \widetilde{\mathbf{c}}_k(u, v, d) = \tilde{\mathbf{c}}(u, v, d_{max}) \odot \cdots \odot \widetilde{\mathbf{c}}(u, v, d_{min}) \tag{2.59}$$

Here the over operator defined in Eq. (2.60) actually represents the matting procedure described in the matting equation [1]

$$\mathbf{f} \odot \mathbf{b} = \mathbf{f} + (1 - \alpha_f)\mathbf{b} \tag{2.60}$$

When re-projection is done, the disparity estimates are refined depending on the visibility map that illustrates a camera k can see a voxel at the position (x, y, d) or not. The visibility value is defined as

$$V_k(u, v, d-1) = V_k(u, v, d)(1 - \widetilde{a}_k(u, v, d)) = \prod_{d'=d}^{d_{max}} (1 - \tilde{a}_k(u, v, d')) \tag{2.61}$$

where $\widetilde{a}_k$ is the opacity of $\widetilde{\mathbf{c}}_k$ in Eq. (2.58) and all the initial visibilities are set to 1, $V_k(u, v, d_{max}) = 1$. In this case, the compositing operation can be linearly expressed as

$$\widetilde{\mathbf{c}}_k(u, v) = \sum_{d=d_{min}}^{d_{max}} \widetilde{\mathbf{c}}_k(u, v, d) V_k(u, v, d) \tag{2.62}$$

Nevertheless, the above procedure cannot recover the true colors and transparencies for mixed pixels near depth discontinuities. A minimization model is formulated to solve the problem. The cost function is written as

$$C = \lambda_1 C_1 + \lambda_2 C_2 + \lambda_3 C_3 \tag{2.63}$$

Then the criterion is accomplished by an iterative gradient descent algorithm. Here C_1 is the weighted error norm on the difference between the re-projected images $\tilde{\mathbf{c}}_k(u, v)$ and the original images $\mathbf{c}_k(u, v)$

$$C_1 = \sum_{(u,v)} w_k(u, v) \rho_1(\widetilde{\mathbf{c}}_k(u, v) - \mathbf{c}_k(u, v)) \tag{2.64}$$

C_2 is a smoothness constraint on the colors and opacities,

$$C_2 = \sum_{(x,y,d)} \sum_{(x',y',d') \in N(x,y,d)} \rho_2(\widehat{\mathbf{c}}(x', y', d') - \widehat{\mathbf{c}}(x, y, d)) \tag{2.65}$$

and C_3 is a distribution on the opacities as follows.

$$C_3 = \sum_{(x,y,d)} \phi(\alpha(x, y, d)) \tag{2.66}$$

In the equations above, ρ_1 and ρ_2 are either quadratic or robust penalty functions, an ϕ is a function encouraging opacities to be 0 or 1.

This method introduces a framework to simultaneously recover disparities, colors, and opacities from multiple images. It can process many occurring problems in stereo matching, such as transparent regions and pixels which mix foreground and background colors. Furthermore, due to the estimated color and opacity values, it also can be extended to extract the foreground object from the image.

2.4.2 Combination of alpha matting and depth maps for natural 3D scenes

It is commonly observed that many artifacts of an alpha matte appear around the high discontinuities in the corresponding depth map. Thus, it is meaningful to combine depth maps with matting problem for the joint work. Compared to single image matting, the correspondence and depth information are more easily used in the video matting that deals with a series of images.

Cho *et al.* developed a fully automated video matting method in [97]. With the aid of a depth camera, the acquired depth information is used to automatically compute a highly accurate trimap, which is the emphasis of this method. According to the fuzziness of the foreground object, the method can generate wide unknown regions for fuzzy areas, while generating narrow unknown regions for the sharp edges. Moreover, the standard closed-form matting approach is extended to optimize both spatial and temporal domains for natural 3D scenes.

First of all, color and depth video sequences are captured simultaneously by a depth camera. Then grayscale depth video is converted into binary depth video. Also, boundary contours are retrieved from binary video sequences. After that, the system automatically produces a trimap for each frame. The size of a trimap's unknown region relies on the fuzziness of the centered pixels and its neighbors. Through depth information, this method

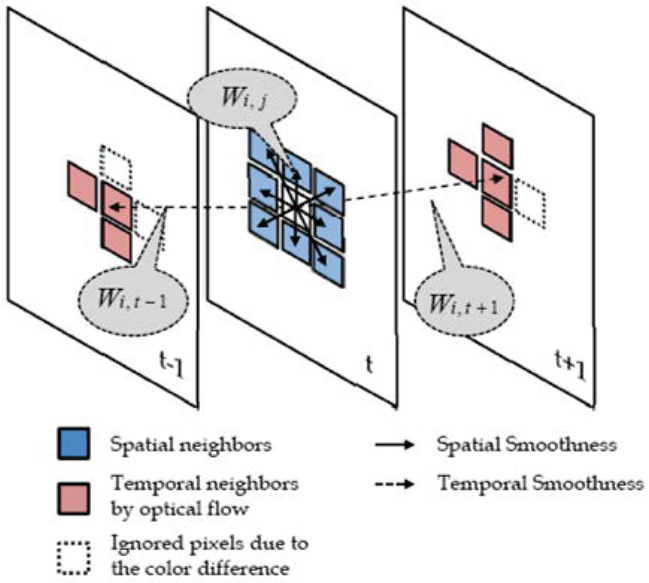


Figure 2.10: 3D graph construction for alpha matting in Cho's method [97].

can trace the exterior boundaries of foreground objects and inner boundaries of holes in objects. For each pixel in the initial boundary, its fuzziness can be treated as

$$\gamma = \frac{2}{|W_k|} \sum_{k \in W_k} \min(\alpha_k,\ 1 - \alpha_k) \tag{2.67}$$

where $|W_k|$ is the size of the area W_k, and α_k is the alpha value of the pixel k. Then, given a pixel p, the radius r_p of the structuring element SE_p is defined as

$$r_p = \gamma \cdot r_{\max} \tag{2.68}$$

As a result, the produced trimap has a wide unknown region for a hairy region and a narrow unknown region for objects with sharp edges.

Furthermore, this work optimizes the alpha matting in both spatial and temporal domains. It inserts two additional weights into the closed-form matting algorithm [25], and the 3D construction of temporal matting is presented in Fig. 2.10. Given a pixel i, it is connected to two types of neighboring pixels: spatial neighbors and temporal neighbors. Firstly, the spatial neighbors of pixel i are connected in frame t, and the edge weight

W_{ij} between two neighboring pixels i and j can be written in the expression from the closed-form matting approach [25]:

$$W_{ij} = \sum_{k}^{(i,j)\in w_k} \frac{1}{|w_k|}\big(1 + (I_i - \mu_k)(\Sigma_k + \frac{\epsilon}{|w_k|}I_3)^{-1}(I_j - \mu_k)\big) \tag{2.69}$$

where w_k is a 3×3 window containing pixel i and j, μ_k is a mean color vector in window w_k, Σ_k is a covariance matrix, I_3 is a 3×3 identity matrix, and $\epsilon = 0.00001$.

Secondly, two temporal neighbors are connected by an optical flow method. The weights depend on both Euclidean distance and color difference. For the frame t, two weights from the previous and next frames, $W_{i,t-1}$ and $W_{i,t+1}$, are defined as follows:

$$W_{i,t-1} = \gamma_{t-1} \cdot \alpha_i^{t-1} \tag{2.70}$$

$$W_{i,t+1} = \gamma_{t+1} \cdot \alpha_i^{t+1} \tag{2.71}$$

where γ is the temporal smoothness factor and α is the alpha value. In this way, the matte can be more temporally smooth as the value of γ increases. Finally, the random walk algorithm [57] is chosen to generate an expected alpha matte.

Cho's method can extract mattes well from both static and dynamic scenes, since it generates good quality trimaps automatically. The temporal information used in the video matting aims to improve the temporal consistency of the alpha mattes for image sequences. But the enhancement are highly dependent on the accuracy of the optical flow.

The other example of combining matting and depth maps is the adaptive depth map assisted matting method proposed to process 3D video in paper [98]. In this method, Sun *et al.* derived a Lagrange-multiplier-free closed-form solution. Also, they introduced an adaptive smoothness criterion based on depth map variance. The matting model can generate an improved alpha matte by automatically producing a trimap from the depth information.

Based on closed-form matting approach [25], this method treats the depth map as an additional clue to guide propagation during matting procedure. It is reasonable because

the smoothness of the corresponding depth map usually has a high consistency with the smoothness of an alpha matte. In closed-form approach, the scalar ϵ of the matting Laplacian is fixed for different pixels, but different ϵ can give rise to different smoothness level of the generated mattes. Replacing the constant ϵ, Sun's method applies a penalty coefficient based on depth variance to control the smoothness adaptively. The penalty term is defined as

$$\epsilon_i = k_1 e^{-k_2 \cdot var(d_i)} \tag{2.72}$$

where $var(d_i)$ is the variance of depth map in the small window centered at pixel i, k_1 and k_2 are constant scalars. Note that the depth values are normalized in the range [0,1]. Accordingly, the modified matting Laplacian $\tilde{L}$ is treated as

$$\widetilde{L}_{(i,j)} = \sum_{k|(i,j)\in w_k} \big(\delta_{ij} - \frac{1}{|w_k|}\big(1 + (I_i - \mu_k)^T \big(\Sigma_k + \frac{\epsilon_k}{|w_k|} I_3\big)^{-1} (I_j - \mu_k)\big)\big) \tag{2.73}$$

Dealing with the cost function proposed from closed-form approach, a Lagrange-multiplier-free solution of the function is derived to reduce computation complexity and improve matting accuracy. Firstly, the cost function is rewritten as

$$\begin{aligned} J(\alpha_u) &= \begin{bmatrix} \alpha_u^T & \alpha_s^T \end{bmatrix} \begin{bmatrix} L_u & L_c \\ L_c^T & L_s \end{bmatrix} \begin{bmatrix} \alpha_u \\ \alpha_s \end{bmatrix} \\ &= \alpha_u^T L_u \alpha_u + 2\alpha_s^T L_c^T \alpha_u + \alpha_s^T L_s \alpha_s \end{aligned} \tag{2.74}$$

where the variable α_u represents the non-user-specified alpha mattes, the known vector α_s represents the user-specified alpha mattes, and L is the corresponding matting Laplacian matrix. Then the gradient of the cost function is set to be zero to acquire the alpha solution that minimizes the cost. At last, the solution of unknown alpha matte is

$$\alpha_u = -L_u^{-1} L_c \alpha_s \tag{2.75}$$

In this way, the computation complexity is decreased by omitting Lagrangian parameter λ and the diagonal matrix D_s. The quadratic form of modified cost function can be written as

$$\alpha^T L \alpha = \frac{1}{2} \sum w_{ij} (\alpha_i - \alpha_j)^2 \tag{2.76}$$

where $-w_{ij}$ is the off-diagonal entry of $\tilde{L}_{(i,j)}$.

Moreover, Sun's work also makes use of depth information to built a trimap in an automatic way. The depth map can be segmented into foreground and background regions by k-means clustering. Considering a simple scene which comprises of two main layers, the pixel intensities are the 1D observations represented as $(x_1,\ x_2, \cdots,\ x_n)$, and these observations are clustered into $k = 2$ sets $S = \{S_1,\ S_2\}$. Minimizing the within-cluster sum of squares is treated as

$$\arg\min_{S} \sum_{i=1}^{k} \sum_{x_j \in S_i} \parallel x_j - \mu_i \parallel^2 \tag{2.77}$$

where μ is the mean value. The trimap is further built by regarding the top $t\%$ observations that are the nearest to their means as the known foreground or background regions. Since depth maps are mainly produced using stereo matching techniques, the noise and errors appearing along the object boundaries are excluded from the top $t\%$ observations. Consequently, the remaining parts are labeled as unknown.

Besides an automatically produced trimap, the depth assisted by closed-form method can efficiently generate an matte with user-specified scribbles as well. Compared to the closed-form approach [25], the method saves matting time with either scribbles or trimaps due to the reduction of computational complexity. It also decreases the errors in the boundary region of the foreground object to an extent.

Chapter 3

The Proposed Method for Chroma-keying Based on Stereo Images

As aforementioned, binocular disparities present the relation and difference between stereo images. For instance, a disparity map can describe the relative depth of the foreground objects in a particular view and make it easy to pay attention to the charateristics delivered by the single image. Besides, stereo images are capable of capturing more details to increase robustness to image noise and texture, where a single image fails. Therefore, we propose a novel post-processing method that applies affinities between matching corresponding pixels from stereo images to alpha propagation.

We propose our matting frame work that can enhance the matting quality based on the stereo input which comprises of an reference image and its matching image. Specifically, our matting framework is built as shown in Fig. 3.1. Assisted by auto-produced trimaps of the two images, the original alpha mattes are generated by an arbitrary matting algorithm. As such, the input information is expanded to two classes of input images, trimaps, confidence maps and initial mattes to post-process. Based on the correspondence between stereo

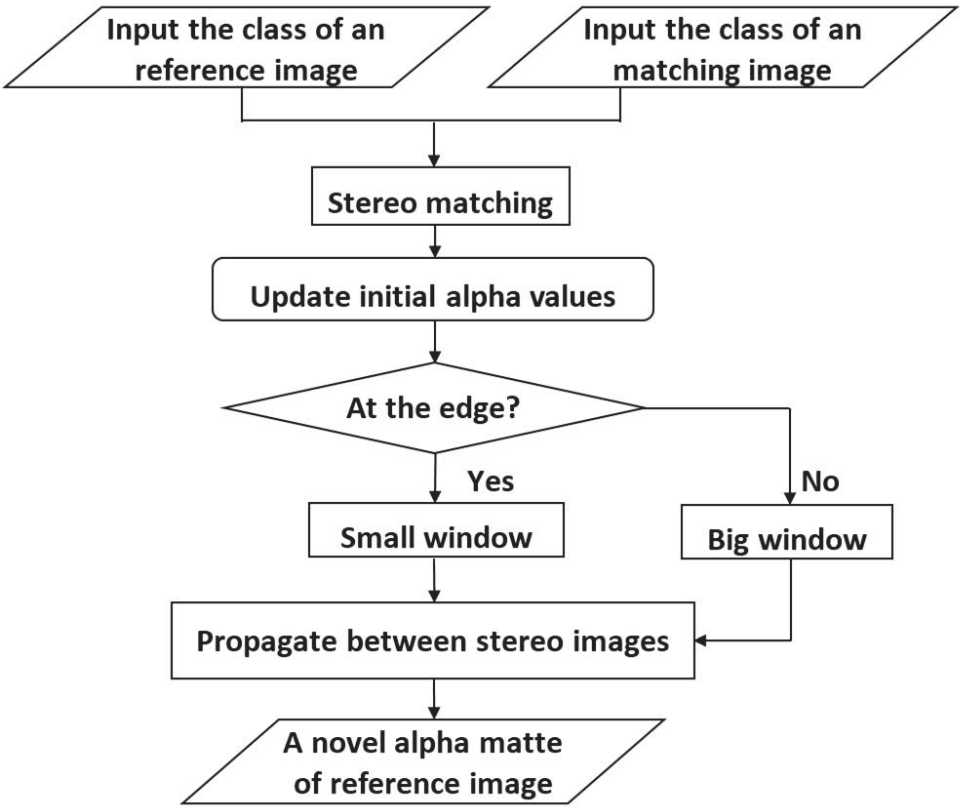


Figure 3.1: Workflow of proposed post-processing model. Each input class includes an original colored image, a trimap, an initial alpha matte and a confidence map.

images, a novel matting Laplacian and a cost function are developed to further propagation between stereo images. The window size of collecting neighboring pixels is determined adaptively during the propagation between two frames. Updating the initial alpha values is an optional step depending on the luminance distribution of the input reference image.

Since an automatically generated trimap and stereo matching are the foundation of our work, we first introduce the chosen methods for these two steps. Specifically, we will present the modified matting Laplacian-based model formulated for stereo images. After that, two additional steps used to refine the final result will be supplied. At last, we provide a method to compute a ground truth of the images for chroma keying.

3.1 Generating a quadmap automatically

As mentioned in Section 2.2.2, Spectral matting [42] is an automatic matting algorithm with little pre-specified information from users. But its model to compute trimaps is designed for natural images. It does not work acceptably for the images processed by chroma keying especially when the background has reflections and shadows. Here we adopt a novel segmentation map technique [24] which is designed for chroma-keying technically. In this section, we briefly introduce this model that uses a *quadmap* to segment the original image into background, foreground, reflective, and unknown regions.

First of all, through analyzing the global color distribution based on histograms, the color range of background is estimated in HSV space. When the color value of a pixel belongs to the estimated background color set in each channel, the pixel is labeled as background. The background region is not completely extracted until the spatial entropy in the value channel is taken into account to detect the texture-less region of the image. This is because the purity of the background is usually degraded by some practical factors such as noise, lighting condition and background setup. Under the less-than-ideal background, some tiny pixels at the fuzzy boundary of the foreground object may be detected mistakenly, as well as the transparent or semitransparent parts. Thus a complete background is

extracted from the image after the refinement by using the spatial entropy-based method.

When it comes to the foreground region, it is divided into absolute foreground and reflective foreground respectively. The absolute foreground includes the pixels with chrominance values that are different from the background color range distinctly. It is accomplished by judging the Hue distance from the pixel candidate to the background color. On the other hand, chrominance information often fails to detect the colorless regions. For example, there are some hunks of black and white in the foreground objects. Therefore, the reflective foreground labels the pixels with low saturation and low lightness. Moreover, a correction is utilized to remove the mixed background color from this labeling procedure.

Finally, the pixels with transparency or semitransparency are classified into the remained unknown region, and a quadmap is then output as the segmentation result. One contribution of this method is that it pays attention to the color spill problem when the color reflecting from the back screen results in an obvious tint on the foreground object. To an extent, it differentiates the reflective regions from the unknown part. Also, it can capture a precise shape of foreground objects with a narrow edge labeling unknown regions. Therefore, we choose the quadmap output by this method automatically as the trimap for an input image in our work.

3.2 The chosen method of stereo matching

The classical stereo matching algorithms commonly focus on horizontal disparity computation rather than vertical disparity, even though the latter phenomenon was possible if the eyes were verged. In practice, photographing the same objects at two perspectives with a slight distance usually leads to both horizontal and vertical disparities between stereo images. For instance, the disparities in these two directions could be caused by misaligned cameras or an unstable tripod. Hence, it is expected that the stereo correspondence problem can be generally resolved in both directions in the thesis.

(a) (b) (c)

Figure 3.2: An example for images and motion vector results using the optical flow method[99]. (a) The reference image. (b) The matching image. (c) A zoomed region for disparities shown by vectors.

Compared to the large computation by applying typical algorithms to the disparities along with the two directions, we capitalize on the method [99] that efficiently accomplishes the matching in both horizontal and vertical directions with less computation. It uses local motion cues to boost the generated maps, while optical flow is utilized to ensure temporal depth consistency. Moreover, it is applicable to images and videos. Different from all disparities values written in the positive form in the conventional algorithms [67] [68] [79] (Eq. (2.49)), the results output by this method can be presented intuitively by means of motion vectors. Consequently, the motion orientation can be directly determined by the sign of the number instead of user specification. Fig. 3.2 gives an example inferred by the chosen method, where graph (c) is a zoomed region as the coordinates described.

3.3 Matting Laplacian and cost function for stereo images

In order to better capitalize on the advantage of stereo images, our work is based on a new matting Laplacian matrix containing two views' pixels. Firstly, we recall the basic concept of matting Laplacian from the closed-form method of Levin *et al.* [25] assumed that either foreground color F or background color B was approximately constant within a

small window. Note that this assumption does not mean that the input image I is locally smooth, for the discontinuities in I can be reflected in the discontinued alpha values. Based on the assumption, alpha values can be expressed as a linear model of the image I in each channel individually. The linear relation of α suggests solving alpha values by minimizing the cost function:

$$J(\alpha, a, b) = \sum_{j \in I} (\sum_{i \in w_j} (\alpha_i - \sum_c a_j^c I_i^c - b_j)^2 + \epsilon \sum_c a_j^{c^2}) \tag{3.1}$$

where w_j is a small window around pixel j. According to mathematically analysis, a_j^c and b_j can be eliminated and result in a quadratic cost with α only:

$$J(\alpha) = \alpha^T L \alpha \tag{3.2}$$

For an image with N pixels, here recall the matting Laplacian L which is an $N \times N$ matrix defined as

$$L(i,j) = \sum_{k|(i,j) \in w_k} \big(\delta_{ij} - \frac{1}{|w_k|}\big(1 + (I_i - \mu_k)^T \big(\Sigma_k + \frac{\epsilon}{|w_k|} I_3\big)^{-1} (I_j - \mu_k)\big)\big) \tag{3.3}$$

where I_i is a 3×1 vector indicating three colors channel, Σ_k is a 3×3 covariance matrix, μ_k is a 3×1 mean vector of the colors in a window w_k, I_3 is the 3×3 identity matrix, δ_{ij} is the Kronecker delta, and $|w_k|$ is the number of pixels in the window. Note that the elements in each row of L sum to zero, and thus the null space of L contains the constant vector.

The matting Laplacian defined in Eq. (3.3) depends on the color line model that foreground color F and background color B are approximately constant within a local small lattice. Fortunately, it is reasonable to extend the application of the assumption to stereo images. This is because the similarities among the corresponding pixels and the neighboring pixels around them have a high consistency. Meanwhile, the pixel correlations between the matched regions are also consistent with the affinities proposed in a single image. Therefore, the affinities from stereo images can be used to propagate alpha values between the two matching regions centered at corresponding pixels.

Let I_1 be a reference image with N pixels, whose matte is needed to be refined, and I_2 be an matching image as a complementary input. As presented in Fig. 3.1, each image has a class of information as the input. The first and the most important issue is to create a new matting Laplacian L for stereo images.

Fig. 3.3 provides an example of two-frame stereo images. The left and right view are respectively divided into two parts by the yellow lines. Intuitively, the parts $\mathcal{B}$ and $\mathcal{C}$ are two homologous parts that represent the same region of the scene. Pixels belonging to one of them are likely to find their corresponding pixels located in the other one through stereo matching. While, either the left part $\mathcal{A}$ of the left view or the right part $\mathcal{D}$ of the right view stands for the unique scene that is captured by the left/right viewpoint only resulting from the binocular disparity. Consequently, for a reference image I_1, its pixels are split up into two parts, *i.e.*, one which has a corresponding pixel in the matching image I_2 and the other one fails to do that.

Apparently, our emphasis is the first part of pixels. When the matching assignment is accomplished, it is easy for the corresponding pixels to obtain two small windows that center on themselves. Based on the assumption that all pixels in the both windows satisfy the color line model, these two matched windows can be united together to a new window to calculate the matting Laplacian (see Fig. 3.4). Compared with algorithms using a single image, by creating a union containing stereo information, the advantage of the correlation between corresponding pixels and their neighbors could be further employed in the proposed approach.

Therefore, different from the window used in Eq. (3.3) for a single image, the window w_k is changed to comprise two cases now:

$$w_k = \begin{cases} w_{k_1} \cup w_{k_2} & (i,j) \in I_1,\ (i+d_i, j+d_j) \in I_2 \\ w_{k_1} & otherwise \end{cases} \tag{3.4}$$

Here, w_{k_1} is a reference window centered at the pixel (i,j) in reference image I_1, w_{k_2} is a matching window centered at the pixel $(i+d_i, j+d_j)$ in matching image I_2. d_i and d_j

Figure 3.3: An example of stereo images

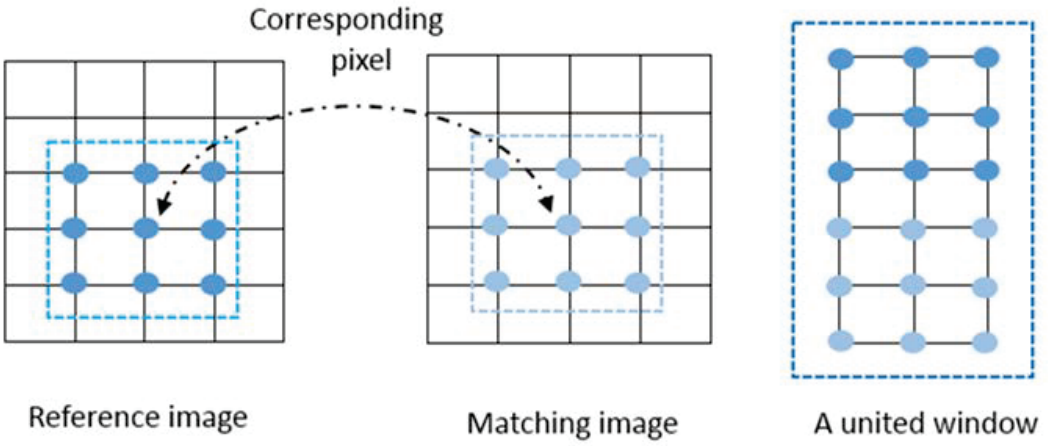


Figure 3.4: An example of creating a new window.

are disparities between the pair of corresponding pixels (i, j) and $(i + d_i, j + d_j)$ in the horizontal and vertical directions.

Although the expression of matting Laplacian for stereo images is the same as defined in Eq. (3.3), the size of L is expanded to $2N \times 2N$ which is four times as that of L before. This is because the essential idea of the new matting Laplacian is to measure weights of pixels from a $2N \times 2N$ united image, which is constituted by the stereo images I_1 and I_2. Consequently, the correlations between all pixels of two frames are applied to recreate a matting Laplacian based on stereo images, and then to propagate alpha values between them. To demonstrate the relation between pixels, the matrix L can be segmented into four $N \times N$ parts as described as

$$L = \begin{bmatrix} L_{11} & L_{12} \\ L_{21} & L_{22} \end{bmatrix} \tag{3.5}$$

where L_{11} and L_{22} represent the weighting affinities among pixels in each image respectively, while L_{12} and L_{21} show the relevance between two images' pixels.

Once the modified matting Laplacian is constructed, we formulate the new cost function which is based on stereo images. Due to the binocular disparity of two views, there may be some differences between the lightness distributions of input stereo images. Accordingly,

the differences lead two distinctive trimaps, and the generated confidence maps and alpha mattes vary as well. Thus, the confidence values are as important as matting Laplacian for stereo images processing. In terms of the cost function designed for a singe image, [49] gives a quadratic cost function consisting of a smoothness term with the matting Laplacian matrix and a data term with the alpha and confidence values. Besides the typical optimization from the closed-form [25], the function also leads alpha propagation from high confidence to low confidence pixels. Hence, we adopt the model introduced in [49] as the basic propagation model, but expand its definition from a single image to stereo images.

Specifically, $\widehat{\alpha} = [\alpha_1, \ \cdots \ \alpha_p, \ \cdots \ \alpha_{2N}]^T$ is treated as a column vector whose first N elements are the all given alpha values for I_1 and second N elements are given alpha values for I_2. The cost function is given by

$$\alpha = \arg\min \alpha^T L\alpha + \lambda(\alpha - \widehat{\alpha})^T D(\alpha - \widehat{\alpha}) + \gamma(\alpha - \widehat{\alpha})^T \widehat{\Gamma}(\alpha - \widehat{\alpha}) \tag{3.6}$$

where λ is a large weighting number compared to the estimated values in $\widehat{\alpha}$ and $\widehat{\Gamma}$, and $\lambda = 10^{-1}$ is a tiny constant that points out the relative importance of data and smoothness terms. D is regarded as a diagonal matrix with 1 for known foreground and background pixels and 0 for others, while diagonal matrix $\widehat{\Gamma}$ is defined by the confidence values for unknown pixels and 0 for others. Similarly to $\widehat{\alpha}$, the first N diagonal elements of D and $\widehat{\Gamma}$ are both aimed for I_1, and the second N diagonal elements of them are aimed for I_2. Note that the solution to minimize Eq. (3.6) is a vector with $2N \times 1$, and the first N elements are the optimal alpha values for I_1.

Combining two frames of stereo images, we not only utilize the affinities within a novel united window involving two matched lattices, but also let the alpha propagation from high confidence to low confidence pixels. In this way, the post-processing can enhance the quality of original alpha mattes generated by any matting algorithm.

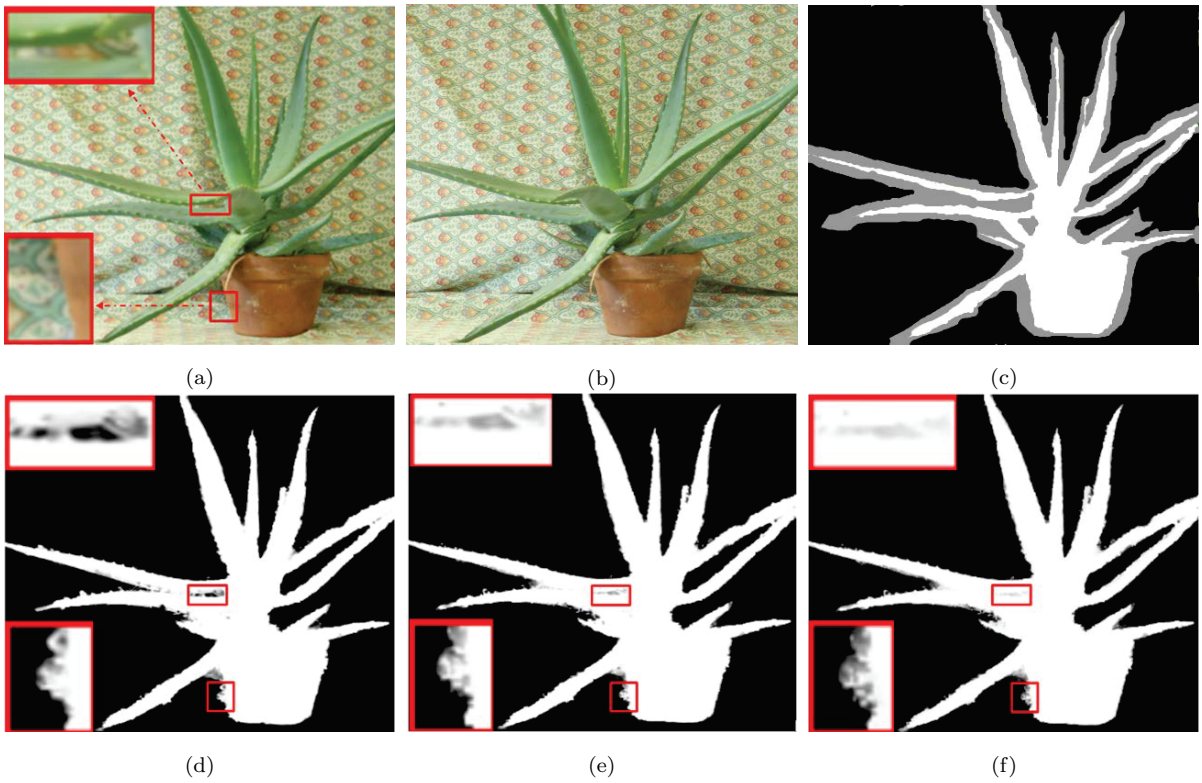
3.4 Adaptive Window Size

In order to reduce computation cost, the window in matting Laplacian (Eq. (3.3)) is usually set small (e.g., 3×3 in [25] [49]). While, [58] shows a larger window size would refine the matte because a lager window may contain more disconnected pixels of true foreground or background and deliver more affinities to be dealt with. Using a larger window is more stable when the color line model holds, but the failure probability of the local assumption also increases with the growth of the window size. How to choose a suitable window size in the propagation is a challenge.

Fig. 3.5 provides three mattes by different fixed window sizes. We can see that, using a 3×3 window, the zoomed region at the bottom contains obvious errors of estimating pixels to be background or foreground (see Fig. 3.5 (d)). Compared with a fixed small size window used in single image matting, the proposed method can improve the accuracy of the most matte regions, which is illustrated in Fig. 3.5 (e). This is because the united window constructed in our method consists of double size of the pixels involved in the window before. Even though the two windows which form a united window are both limited to the size 3×3, the initial alpha values of the corresponding pixels from I_2 have a good effect on the final result of I_1 after the propagation between stereo images. When the united window is constituted by two 5×5 windows, the errors are more alleviated, since a wide window covers more background details. Meeting the color line model, large windows help to correct the alpha values at the bottom region (see Fig. 3.5 (f)).

However, a large window may overly smooth the resulting matte when it covers the image regions with narrow gaps or complex scenes, which is demonstrated in the zoomed region on the top (see Fig. 3.5 (e) (f)). In this narrow region, a 3×3 window is small enough and never cover the gap between two edges. In contrast, a large window is more likely to cross the gap and result in errors hard to be avoided.

By analyzing the experimental results, we use an adaptive window size to solve the problem caused by fixed window sizes. For an reference image I_1, it is easy to detect the

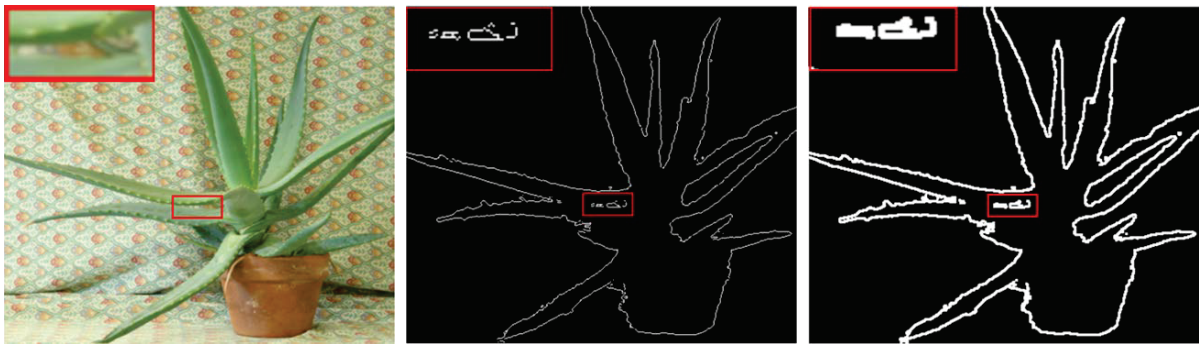


(a) (b) (c)

(d) (e) (f)

Figure 3.5: Comparison of mattes using different fixed window sizes. (a) Reference image I_1. (b) Matching image I_2. (c) A hand-drawn trimap for the reference image I_1. (d) Use a 3×3 window size in post-processing the matte of I_1 only. (d) Use a united window comprising of two 3×3 windows by the proposed method. (e) Use a united window comprising of two 5×5 windows by the proposed method. The initial alpha matte is produced by the weighted color method [27].

edge of its initial matte. Apparently, the gradient and variance change more sharply over this area than other regions. Therefore, the pixels at the edge need a small window to avoid over-smoothing, especially at the narrow gap or hollow-out regions like the red box regions shown in Fig. 3.6. To reduce the error effect of the initial matte, a small window size (3×3) is used in the edge area of I_1 after dilating (via 3 pixels in our experiments). For the matching image I_2, the matching window around the corresponding pixel is also chosen to be small (3×3). The remaining pixels in the other areas of I_1 and I_2 are dealt with a large window size (5×5) to refine the matte.



(a) (b) (c)

Figure 3.6: Detecting the edge of the initial alpha matte. (a) Reference image I_1. (b) The edge of the initial alpha matte of I_1. (c) The dilating edge area.

In other words, given a pixel r at the edge area of I_1, which has a corresponding pixel m in I_2, the size of the united window (Eq. (3.4)) in new matting Laplacian matrix is constructed by two small windows (3×3) centered at r and m respectively. While, the united window constituts two large windows (5×5) for the else pixels of I_1 which also have corresponding pixels in I_2.

Through choosing window size adaptively, those pixels needing a small window to avoid covering high texture are successfully separated from other pixels. Meanwhile, for the pixels beyond the edge area, using a large window helps to ensure the strengths of the affinity propagation between constrained pixels from stereo images. Fig. 3.7 provides an example. Compared to Fig. 3.7 (c), the pixels located in the hole area of the basket in Fig. 3.7

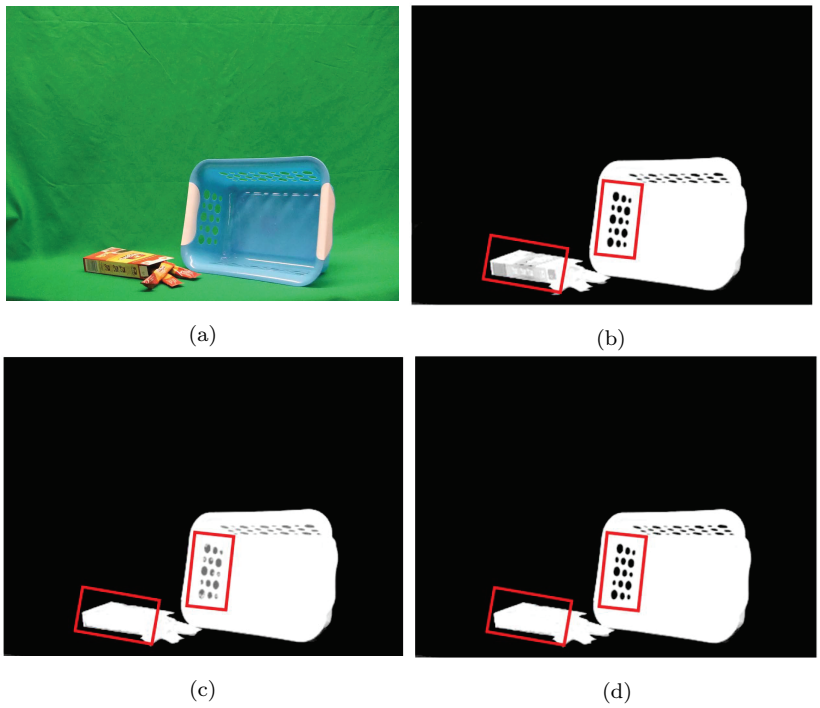


Figure 3.7: Comparision of mattes using a fixed window and an adaptive window size. (a) Reference image I_1. (b) The matte using a small united window. (c) The matte using a large united window. (d) The matte using an adaptive window. The initial alpha matte is produced by KNN [26].

(d) do not have the over-smoothness mistake caused by a large window because they are processed by a small united window. Besides, except for the pixels at the edge area, since all the remaining pixels located in the box area on the left are dealt with a large united window, their alpha values in Fig. 3.7 (d) are estimated more correctly than the values resulted from a small window as shown in Fig. 3.7 (b).

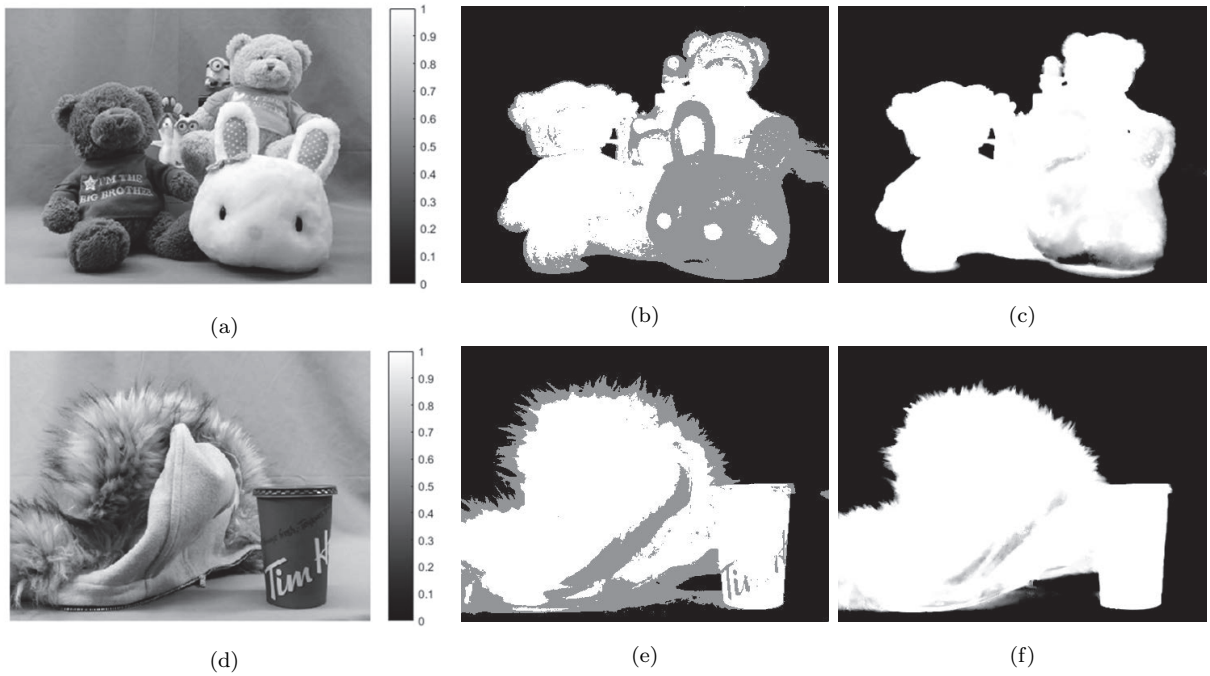
3.5 Pre-processing

Compared to natural image matting, the quality of chroma keying is more affected by luminance. As mentioned earlier, it is a common phenomenon that the foreground pixels with high/low luminance values are more likely to be set as unknown regions in the trimap automatically [24]. Consequently, those pixels are further computed with the alpha values less than 1 in the original alpha matte. The first and second rows of Fig. 3.8 respectively illustrate the high and low lightness influences on the mattes resulting in some obvious errors at the foreground regions.

Since the initial alpha values act as one of inputs in the proposed method, the final matte founded on stereo images processing is consequently affected by an inaccurate input. As such, we update the input matte of the reference image I_1 before solving the cost function Eq.(3.6).

First of all, we normalize the luminance values of all pixels of I_1 and analyze the luminance distribution of all the unknown pixels by histograms (see Fig. 3.9). From the statistics in our experiments, we find that excluding the first six bins, the rest bins in the histograms contain the most particular pixels that actually locate on the absolute foreground object but are labeled as unknown. It can be explained that, in the first six bins, most pixels with relatively lower lightness values more likely represent shadows or locate at the edge area, rather than locate at obvious foreground regions.

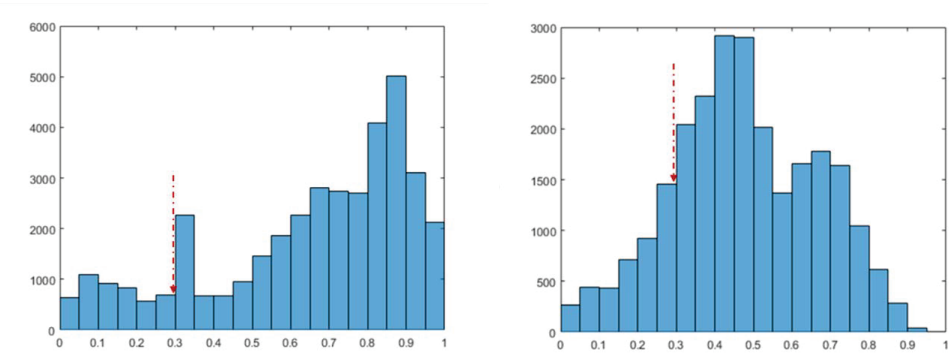
We assume that the particular unknown pixels whose luminance numerics are higher



(a) (b) (c)

(d) (e) (f)

Figure 3.8: The effects of luminance on the initial mattes. (a) The luminance map of the input image *dolls*. (b) The trimap of the image *dolls* produced by the quadmap method [24]. (c) The initial matte of the image *dolls* using the weighted color method [27]. (d) The luminance map of the input image *wool coat*. (e) The trimap of the image *wool coat* produced by the quadmap method [24]. (f) The initial matte of the image *wool coat* using the weighted color method [27].



(a) (b)

Figure 3.9: Luminance histograms of unknown pixels. (a) Histogram of the unknown pixels from the image *dolls*. (b) Histogram of the unknown pixels from the image *wool coat*.

than the threshold t_1 with a high probability of being foreground region. So we increase the initial alpha values of the selected unknown pixels by a certain extent t_2. Then the new alpha values of these pixels are substituted into the $\widehat{\alpha}$ to recalculate the cost function in Eq. (3.6). In this way, the errors occurred in obvious foreground regions may be corrected to closer to the definite foreground. By analyzing the experiment results, the value of the sixth bin's edge is set as the threshold value t_1. That means $t_1 = 0.3$ in our experiments. Also, we add $t_2 = 0.2$ to initial alpha values in our experiments to ensure effective pre-processing. Fig. 3.10 provides two examples of pre-processing.

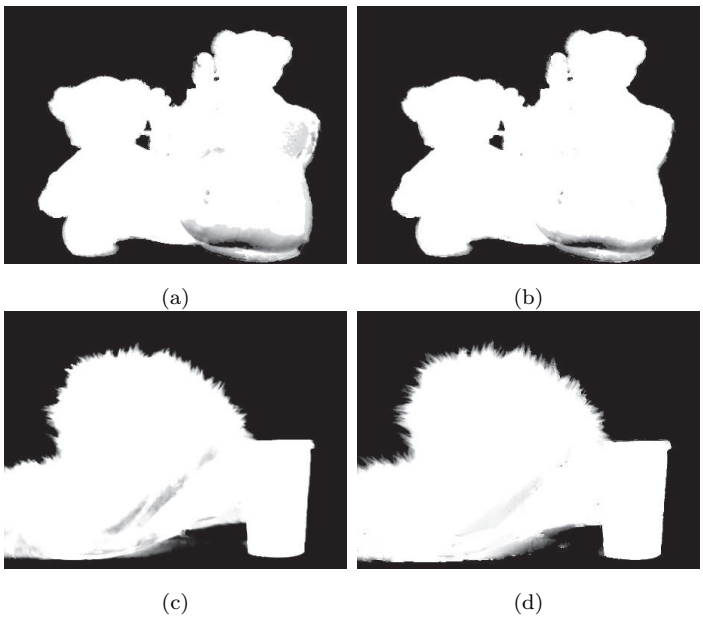


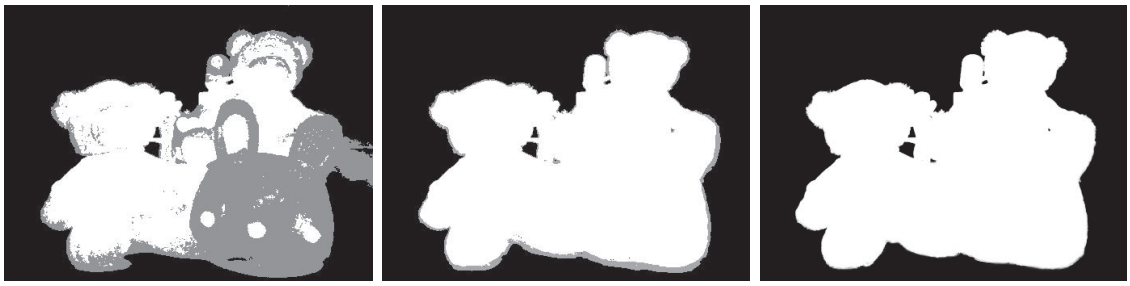
Figure 3.10: Examples of mattes with pre-processing. (a) The matte of image *dolls* by stereo images propagation without pre-processing. (b) The matte of image *dolls* by stereo images propagation with pre-processing. (c) The matte of image *wool coat* by stereo images propagation without pre-processing. (d) The matte of image *wool coat* by stereo images propagation with pre-processing.

Please note that the pre-processing step is based on the trimap generated by Wang's quadmap method [24]. Since this method can separate a precise shape of foreground objects with a narrow edge labeling unknown regions, the selective unknown pixels used to update initial alpha values is ensured to contain background pixels as few as possible.

In addition, this step is optional to improve the quality of the final alpha matte, because it is not suitable for the images with transparent regions, where alpha values of the pixels should be in the range (0,1) but close to 1.

3.6 Generating ground truth

In this section, we introduce the implementation required in the evaluation. Most current matting algorithms evaluate their natural image alpha mattes by the ground truth alpha maps supplied in the online benchmark [14]. However, the benchmark does not provide ground truth for chroma keying images with a fixed colored background. To objectively evaluate the proposed approach results, we propose a method to generate a ground truth map as close as possible.



(a) (b) (c)

Figure 3.11: Generating ground truth. (a) Auto-generated quadmap [24]. (b) Specialized trimap for ground truth. (c) Output ground truth by our method.

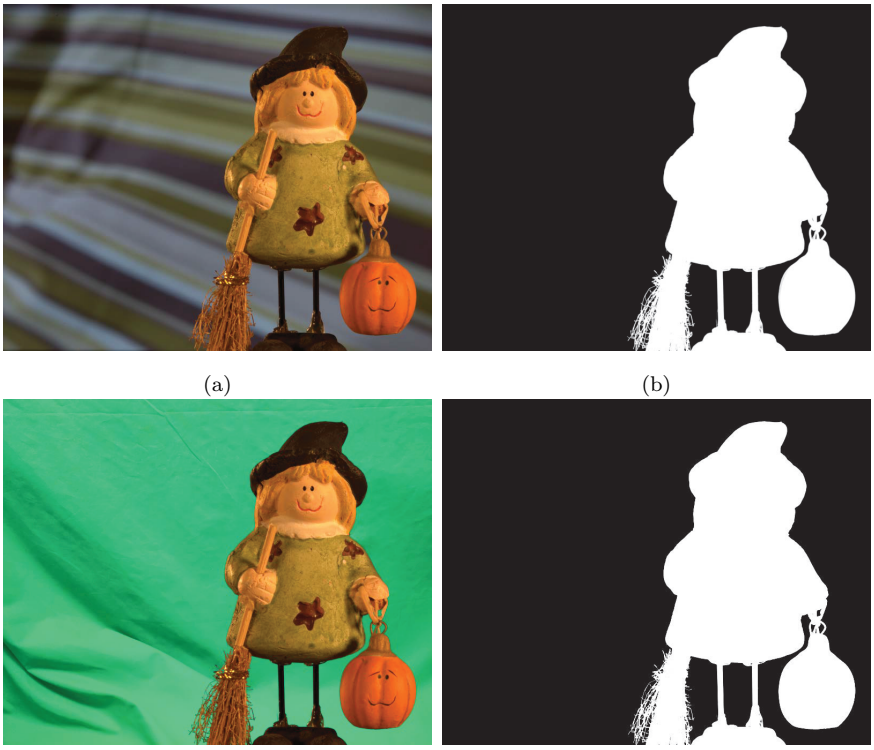
Firstly, a quadmap is automatically generated by the quadmap method [24]. The quadmap generated can not only capture the exact shape of the foreground objects, but also develop a narrow edge regions with a few unknown pixels. But as discussed before, those pixels locating in foreground/background regions may be marked as unknown in the quadmap when their luminance ranges are quite different from other pixels' luminance range. For instance, as shown in Fig. 3.11 (a), almost all the pixels of the bunny's face are marked as unknown though most of them should be set as foreground.

Based on a quadmap, definite foreground and definite background pixels are further labled by users from their visual perspectives to produce a modified quadmap that only labels the pixels at the very narrow edge area as unknown while any other pixles are considered as known pixels. In this way, the number of unknown pixels in the quadmap can be greatly reduced. After user interaction, the modified quadmap is chosen as a specialized trimap designed for generating ground truth (see Fig. 3.11 (b)).

Once we get the specialized trimap, a ground truth can be produced by the state-of-the-art matting alogrithms. To achieve this, the comprehensive sampling method [28] and KNN matting [26], which typify the color sampling-based and propagation-based approaches respectively, are utilized to obtain their own alpha mattes. The average values of the two mattes are used as the expected ground truth (see Fig. 3.11 (c)).

Fig. 3.12 presents a comparision of ground truth maps. In this experiment, the foreground object of the original image is composited to a green background. And then a ground truth map for the composited image is generated by our work. Compared to the ground truth provided by the benchmark [14], our result delivers the mean squared error (MSE) as tiny as 0.0008 only.

Please note that our method can generate excellent ground truth for most images in which there are no transparent foreground objects. But if there are transparent objects or very tiny foreground details in the image (see Fig. 3.13), this method may not generate reliable ground truth. Because the comprehensive sampling method [28] and KNN matting [26] cannot have good alpha mattes in these complex regions, our method fails to ensure the accuracy of alpha mattes.



(a) (b)

(c) (d)

Figure 3.12: Comparision of ground truth maps. (a) Original image from the online benchmark [14]. (b) Ground truth from the online benchmark [14]. (c) Composited image with a green background. (d) Output ground truth for the composited image by our method.

(a) (b)

Figure 3.13: Difficulties to generate ground truths. (a) The semi-transparent cup on the left. (b) The floating hair in the red box.

Chapter 4

Experimental Results

In this chapter, we demonstrate that the proposed method can enhance the alpha matte quality effectively as a post-processing for existing matting techniques.

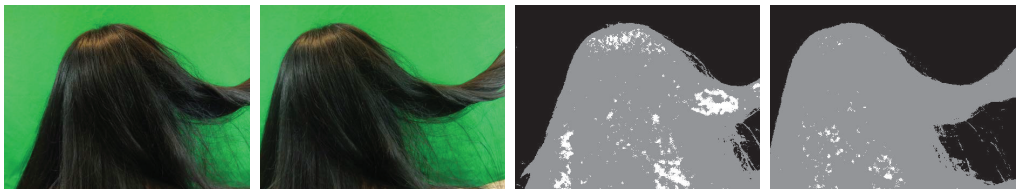
In terms of industrial chroma keying patents, we can not post-process their results because commercial softwares such as Adobe After Effects [100] and Primatte [101] do not provide us confidence maps required in our model. Therefore, our proposed algorithm is compared with five representative matting algorithms in academia—closed-form [25], KNN [26], weighted color [27], comprehensive sampling [28] and learning based [29] matting methods—which are widely accepted as a baseline for evaluating new matting method. They cover the four categories of current matting approaches. Specifically, the closed-form and KNN are from the propagation-based category. The learning based method belongs to the machine learning matting category. The weighted color and the comprehensive sampling methods are from the color sampling-based category, and they are also regarded as the combination of sampling and propagation based methods because they use an extra propagation-based cost function.

All test stereo images were taken by the binocular camera (Fujifilm 3D W3), and their trimaps were automatically generated by the quadmap method [24].

4.1 Visual quality comparison

Four groups of test stereo images and their trimaps are presented in Fig. 4.1, Fig. 4.3, Fig. 4.5 and Fig. 4.7 respectively. The stereo images contain a number of difficult problems such as complex fuzzy boundaries and highly textured regions. The conventional matting approaches do not give satisfactory alpha mattes, however, the proposed method as a post-processing can significantly improve the alpha matte quality for these methods. The visual quality difference between the original matting result and the matte refined by our method is evident enough to figure out.

Experiment 1



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

Figure 4.1: Test images and their trimaps for Experiment 1. (a) Left view. (b) Right view. (c) Trimap for left view. (d) Trimap for right view.

In the first experiment we use the right view as the reference image I_1. As shown in Fig. 4.1, there are many fuzzy regions in the stereo images, but very few known foreground pixels in the trimaps. The pixels of black hair usually have low luminance and the color spill problem which is referred to as the phenomenon of color reflection from the back screen. Apparently, the closed-form method [25] fails in this experiment (see Fig. 4.2 (a1)), resulting in a significant loss of foreground hair in the right half area in the image. The learning based method [29] works better than the closed-form, yet still delivers disappointing performance when processing the same area in the image (see Fig. 4.2 (e1)). Compared to these two approaches, our method significantly enhances the quality and accuracy of the matte (see Fig. 4.2 (a2) (f2)). It can be seen that our mattes present more foreground

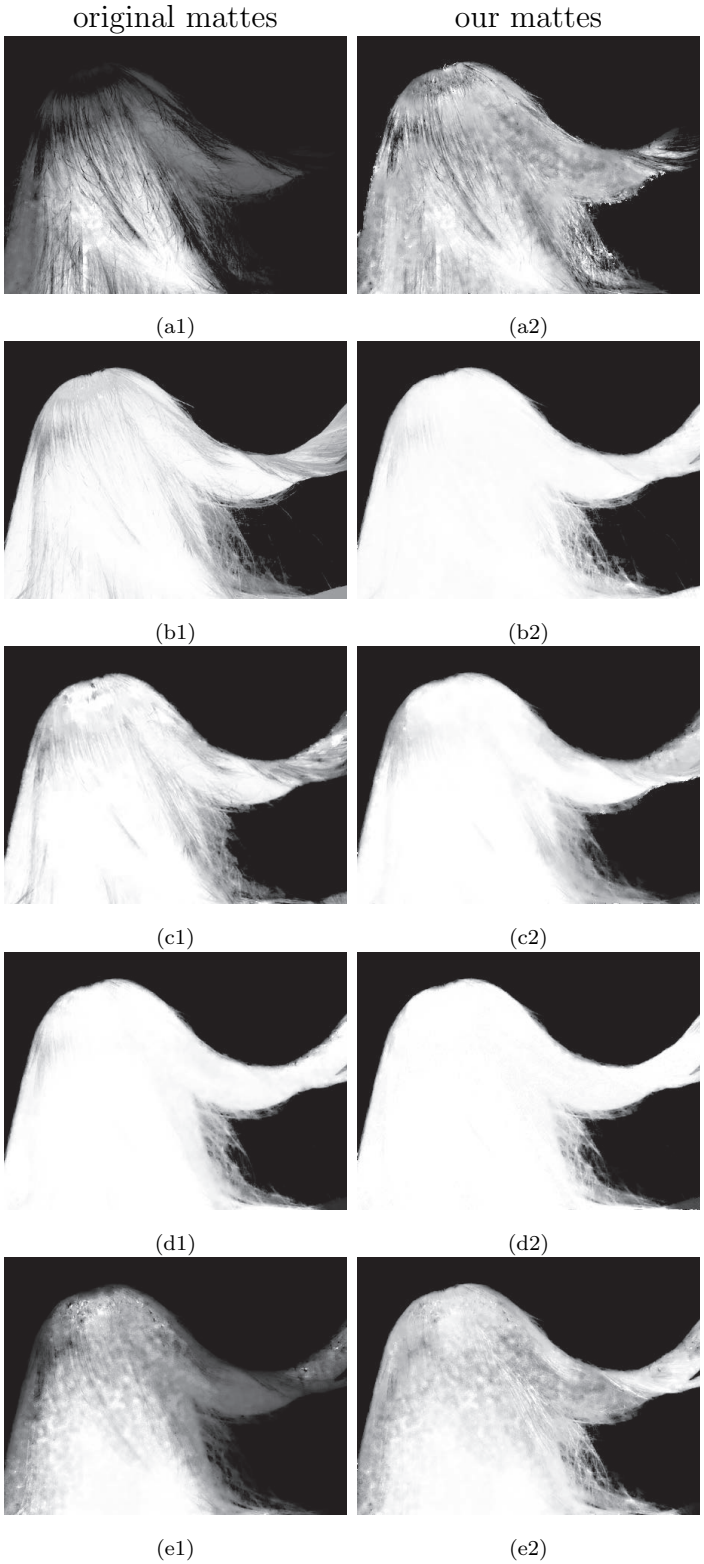
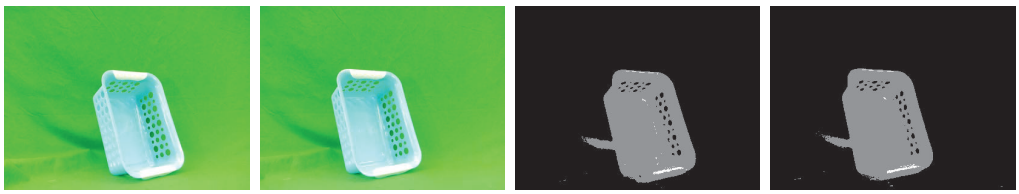


Figure 4.2: Experiment 1: Comparison of results obtained by different approaches and the proposed method. (a) Closed-form [25]. (b) KNN [26]. (c) Weighted color [27]. (d) Comprehensive samples [28]. (e) Learning based [29].

scene of hair in the right half.

On the other hand, the KNN [26], the weighted color [27] and the comprehensive sampling [28] methods all produce fine original results (see Fig. 4.2 (b1) (c1) (d1)). But the color spill problem degrades their results, so that a variety of pixels whose alpha values are smaller than 1 can be observed in the mattes of these three approaches. For instance, the comprehensive sampling performs the best in this experiment, however, there are still some artifacts on the top left part of the foreground hairs. Through the proposed work based on stereo images, we can see that for each approach, our work succeeds in removing most of the artifacts located in the foreground regions of original matte (see Fig. 4.2 (b2) (c2) (d2)). Due to the propagation between stereo images, the proposed method has an evident advantage of processing fuzzy regions than other methods.

Experiment 2



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

Figure 4.3: Test images and their trimaps for Experiment 2. (a) Left view. (b) Right view. (c) Trimap for left view. (d) Trimap for right view.

As shown in Fig. 4.3 (a) and (b), there is a strong reflection on the foreground basket with a number of holes. We choose the right view to be the reference image I_1. In both of the trimaps, as shown in Fig. 4.3 (c) and (d), there are only a few known absolute foreground pixels in them. The main body of the basket has a high luminance level, which is labeled as a unknown region. Accordingly, these trimaps fail to generate satisfactory alpha mattes.

Among all the original results, the reference image's matte produced by the closed-form

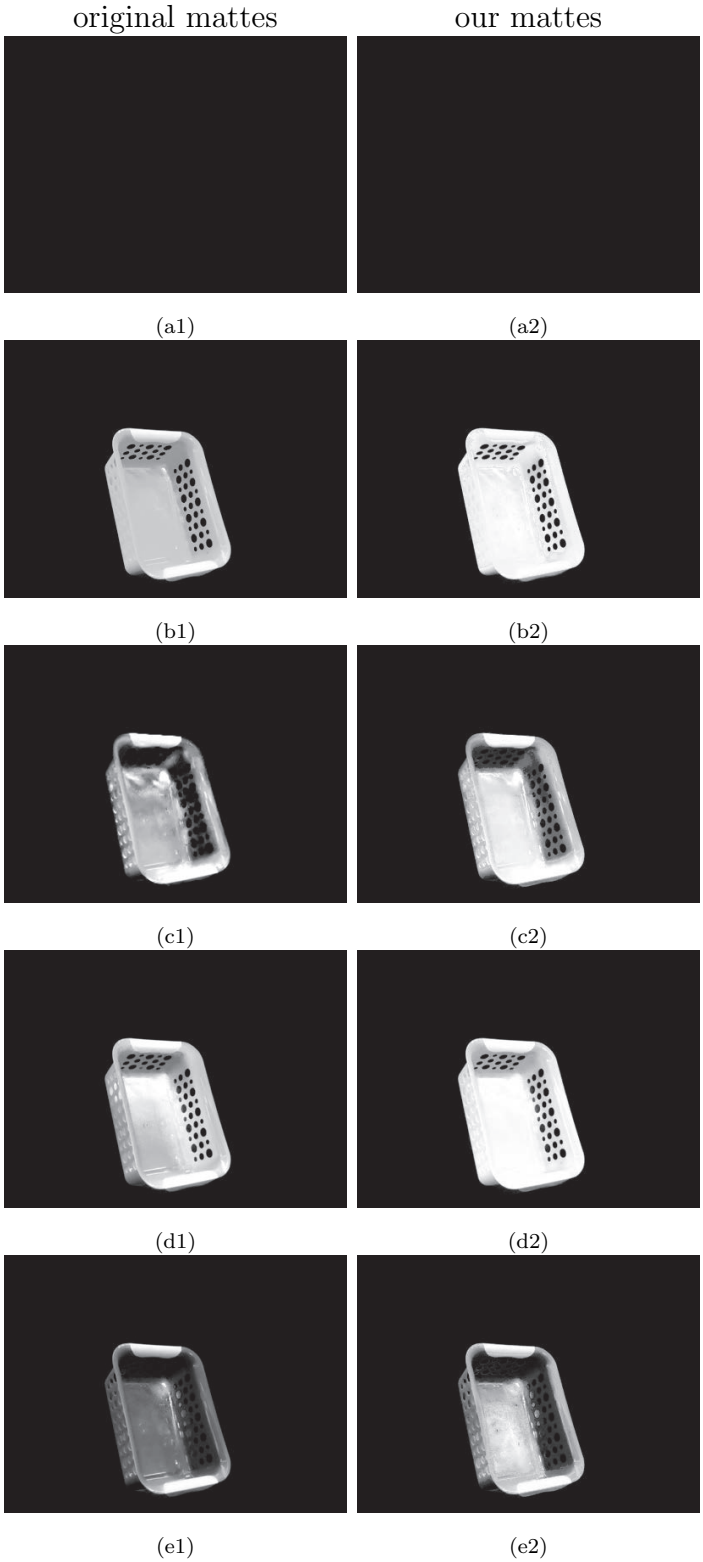
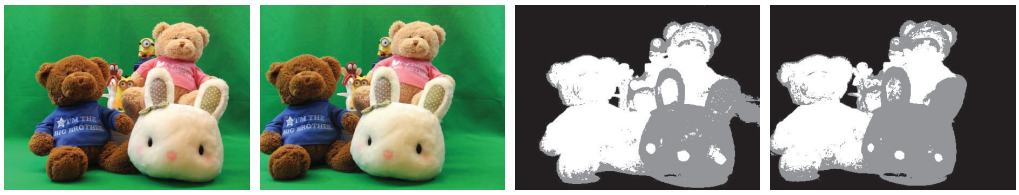


Figure 4.4: Experiment 2: Comparison of results obtained by different approaches and the proposed method. (a) Closed-form [25]. (b) KNN [26]. (c) Weighted color [27]. (d) Comprehensive samples [28]. (e) Learning based [29].

method [25] is a total failure as there is only a pure background without any foreground pixels (see Fig. 4.4 (a1)). Also, the matching image's matte presents the same failure. Based on the propagation of stereo images, our method also fails to provide a matte with foreground objects. Consequently, as demonstrated in Fig. 4.4 (a2), the proposed method cannot refine the matting result when the initial alpha matte values of stereo images are all 0.

As for the other four original mattes, it is noticeable that they all present a dark gray basket as the foreground. This illustrates that the rest four algorithms all falsely compute the alpha values of the pixels on this region, which are obviously smaller than 1 (see Fig. 4.4 (b1) (c1) (d1) (e1)). The proposed method succeeds in improving the original result quality by boosting the alpha values of the basket to a level closer to 1. Visually, when compared to the weighted color [27] and the learning based [29] approaches, the proposed method significantly relieves the gray color of the solid bottom surface of the basket (see Fig. 4.4 (c2) (e2)). For KNN [26] and the comprehensive sampling method [28], respective mattes created by our method obviously refine all the alpha values in the foreground region and present a basket color closer to white (see Fig. 4.4 (b2) (d2)).

Experiment 3



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

Figure 4.5: Test images and their trimaps for Experiment 3. (a) Left view. (b) Right view. (c) Trimap for left view. (d) Trimap for right view.

In the third experiment, the left view is defined as I_1. In Fig. 4.6, all the five approaches' original mattes show obvious errors at the areas of the bunny on the right side and the

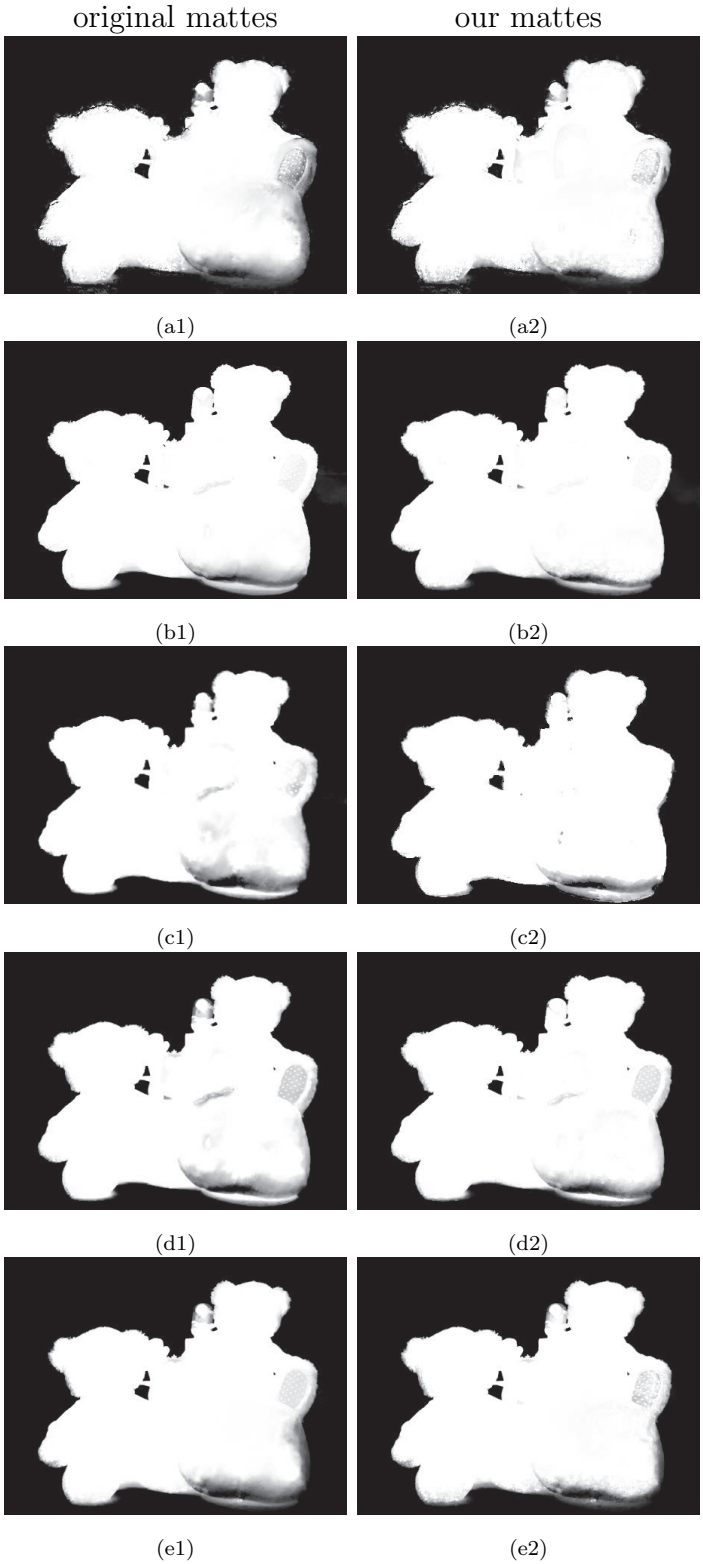


Figure 4.6: Experiment 3: Comparison of results obtained by different approaches and the proposed method. (a) Closed-form [25]. (b) KNN [26]. (c) Weighted color [27]. (d) Comprehensive samples [28]. (e) Learning based [29].

farthest minion model. Besides, except the closed-form's [25] original matte, the rest original matting results all mistakenly treat the shadow region near the bunny's chin as foreground.

Comparing with all the original mattes, our post-processed results correct the errors in the area around the bunny's eyes and reduce the level of error around the bunny's right ear and chin. In addition, compared to the initial results by the approaches in papers [26] [27] [29], the proposed method rectifies the mistakes in the shadow region to an extent (see Fig. 4.6 (b2) (c2) (e2)).

On the other hand, there are several small and even tiny holes among dolls. Such holes are computed correctly as background in our results rather than being over-propagated. This demonstrates that the adaptive window size used in the proposed method can effectively avoid over-smoothness caused by large size windows.

Experiment 4

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

Figure 4.7: Test images and their trimaps for Experiment 4. (a) Left view. (b) Right view. (c) Trimap for left view. (d) Trimap for right view.

In Fig. 4.8, the fourth experiment deals with the right view as I_1. Most pixels at the foreground region of the apple and the envelope are labeled as unknown because of the luminance effect. Consequently, all methods do not estimate the correct alpha values at these regions.

For the regions of the apple and the envelope, according to the initial and the refined results, it could be noticed that all of five initial mattes have been improved effectively.

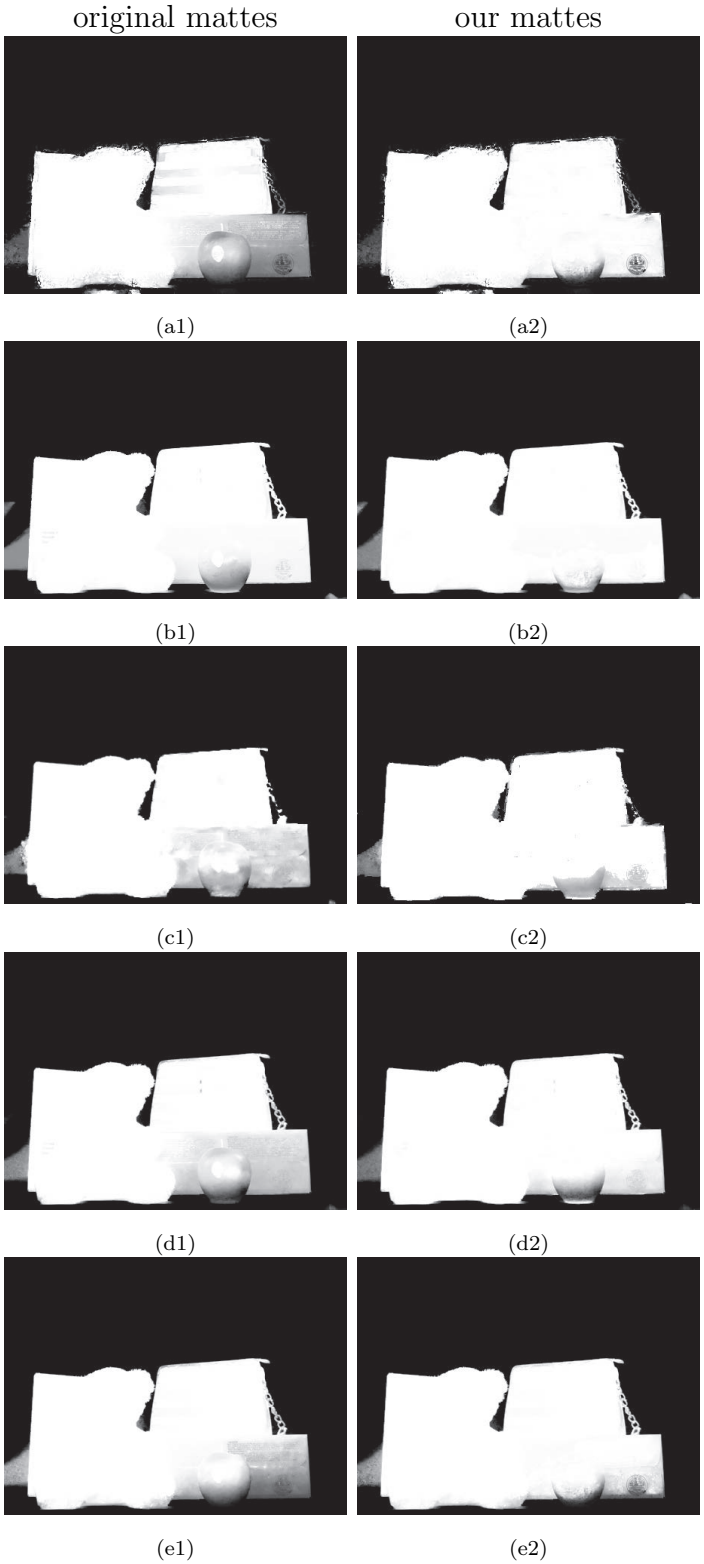


Figure 4.8: Experiment 4: Comparison of results obtained by different approaches and proposed method. (a) Closed-form [25]. (b) KNN [26]. (c) Weighted color [27]. (d) Comprehensive samples [28]. (e) Learning based [29].

Our method enables the refined mattes to have much fewer pixels with incorrect alpha values. This contribution can be attributed to the application of a big window size at these regions. On the other hand, since a small window size is used in the bag's chain region, our mattes succeed in avoiding the over-smoothing phenomenon when processing the small holes from the chain.

However, considering the shadow on the background, the proposed method fails to change the impure background region. The reason can be explained as follows. Given a pair of original alpha mattes of the stereo images, they both have mistakenly estimated alpha values in this shadow region. Depending on the inaccurate initial input, it is very difficult for our post-processing method to greatly lower the level of the alpha values.

4.2 Objective quality comparison

In this section, the proposed chroma keying method is utilized to generate alpha mattes of the test images shown in Fig. 4.9, where (a1) - (a12) are reference images, and (b1) - (b12) are the ground truth maps obtained by our method described in Section 3.7. The original mattes produced by five existing approaches are shown in Fig. 4.10, where (a1) - (a12), (b1) - (b12), (c1) - (c12), (d1) - (d12) and (e1) - (e12) are results by the closed-form method [25], the KNN matting [26], the weighted color method [27], the comprehensive sampling method [28] and the learning based method [29], respectively. The post-processed alpha mattes by our algorithm are shown in Fig. 4.11, where (a1) - (a12), (b1) - (b12), (c1) - (c12), (d1) - (d12) and (e1) - (e12) are corresponding post-processed results of the five approaches.

Compared with ground truth maps, the mean squared error (MSE) and the sum of absolute differences (SAD) of the generated alpha maps are calculated by Eq. (4.1) and Eq. (4.2) respectively. Besides, the gradient error introduced in [102] is computed by Eq.

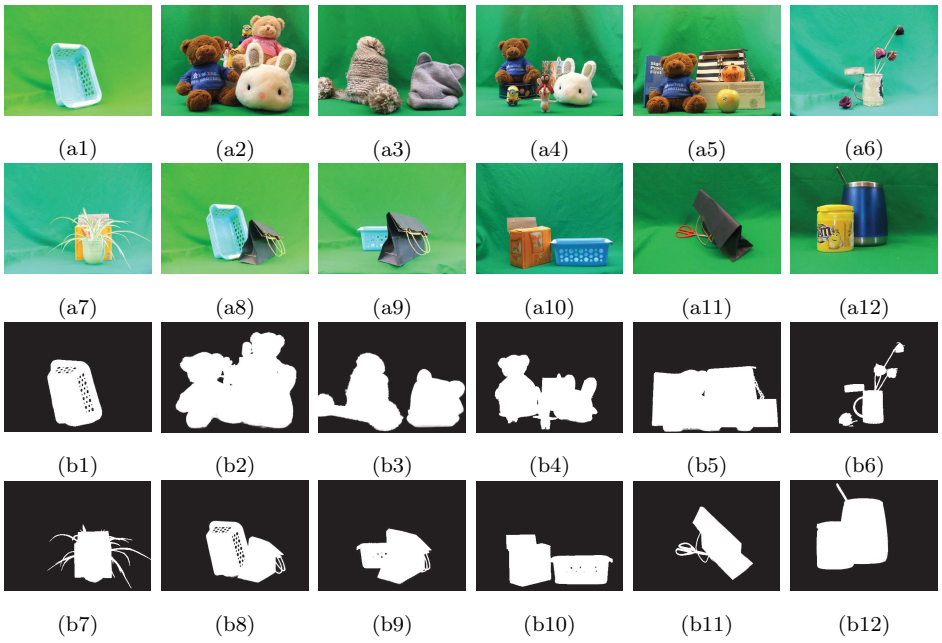


Figure 4.9: Test images for objective evaluation.

(4.3) to evaluate over-smoothness and erroneous discontinuities in the mattes.

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(\alpha_i - \alpha_i^*)^2 \tag{4.1}$$

$$SAD = \frac{1}{N}\sum_{i=1}^{N}|\alpha_i - \alpha_i^*| \tag{4.2}$$

$$gradient = \frac{1}{N}\sum_{i=1}^{N}(\bigtriangledown\alpha_i - \bigtriangledown\alpha_i^*)^2 \tag{4.3}$$

where N is the number of the image's pixels, α is the computed alpha matte, α^* is the ground truth, $\bigtriangledown\alpha_i$ and $\bigtriangledown\alpha_i^*$ are the normalized gradients of the mattes at pixel i that are computed by convolving the mattes using first-order Gaussian derivative filters with variance $\sigma = 2$.

4.2.1 Average error comparison

Intuitively, Fig. 4.12 supplies the comparisons of the average MSE, SAD and gradient of 12 groups of generated alpha mattes between five matting approaches and corresponding

Figure 4.10: Original mattes by five matting approaches.

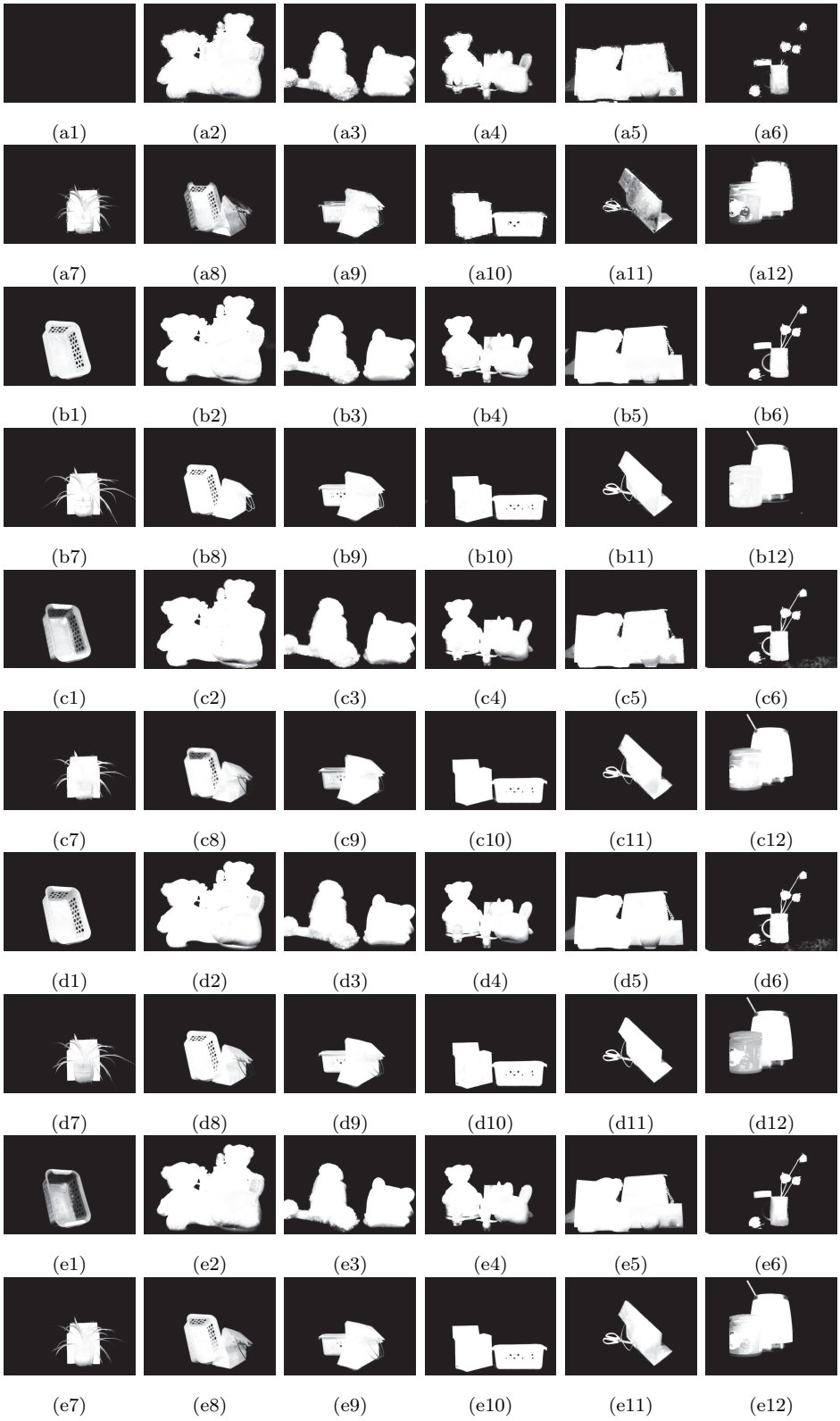
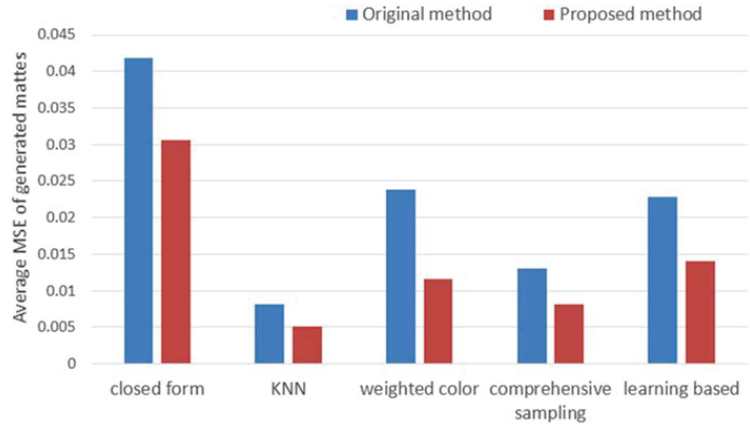
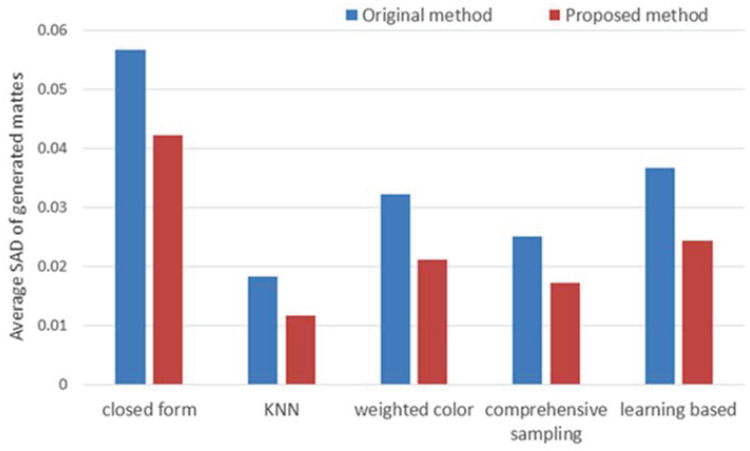


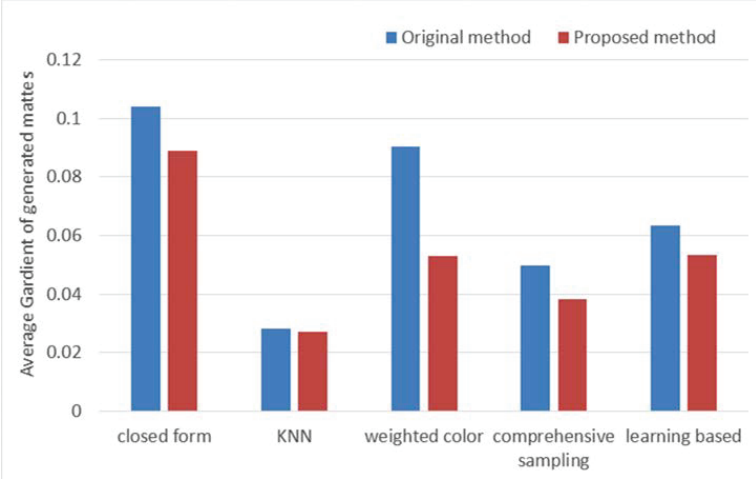
Figure 4.11: Generated mattes by proposed method.



(a)



(b)



(c)

Figure 4.12: Average value comparisons for three types of errors. (a) MSE. (b) SAD. (c) Gradient.

	Closed form [25]	KNN [26]	Weighted color [27]	Comprehensive sampling [28]	Learning based [29]
Original MSE	0.0418	0.0081	0.0239	0.0131	0.0228
Proposed MSE	0.0306	0.0052	0.0116	0.0082	0.0140
Original SAD	0.0566	0.0183	0.0322	0.0250	0.0366
Proposed SAD	0.0422	0.0116	0.0211	0.0171	0.0244
Original gradient	0.1041	0.0281	0.0902	0.0496	0.0634
Proposed gradient	0.0888	0.0270	0.0531	0.0383	0.0533

Table 4.1: Comparisons of average errors.

post-processes by the proposed method, where each bar chart displays one error type. The specific data can be found in the Table 4.1.

As shown in Fig. 4.12, the error decrease can be considered as the most satisfying performance of the proposed method. Generally speaking, there are two noticeable things in Fig. 4.12. For one thing, among the five initial approaches, KNN matting [26] gives the smallest MSE, SAD and gradient that are calculated by original mattes. Accordingly, the proposed method further reduces errors and provides the post-processed mattes that have the lowest mean value in each error type. In other words, KNN matting delivers the most outstanding performance in both original method and post-processing by our method. For another, the closed-form method [25] gives the highest level of average MSE, SAD and gradient of both original and post-processed mattes.

For MSE, the proposed method can greatly refine the original mattes produced by the closed-form method [25] (from more than 0.04 to approximately 0.03), the weighted color method [27] (from roughly 0.024 to about 0.012) and the learning based method [29] (from around 0.023 to 0.014). The result is similar to that of SAD. The proposed algorithm improves the weighted color method by decreasing the mean gradient error from 0.09 to just above 0.05.

Table 4.1 provides more specific numeric comparisons. It can be observed that, when

post-processing the five algorithms, our method succeeds in obviously decreasing the average MSE, SAD and gradient error of the 12 mattes generated by each algorithm. For the closed form matting [25] and its post-processing, the decrease of MSE, SAS and gradient are all larger than 0.01. Similarly, the difference of this level can also be noticed in the case of the weighted color approach [27].

Moreover, compared with the comprehensive sampling [28] and the learning based [29] methods, the proposed approach lowers the levels of their average gradient errors, from 0.0496 and 0.0634 to 0.0383 and 0.0533, respectively. Also, the noticeable refinements regarding to the average gradient error are all larger than 0.01, except for KNN.

On the other hand, for KNN matting [26], there is only a subtle distinction smaller than 0.01 in terms of MSE. Besides, our gradient error is only 0.0009 smaller than its original result, which is hardly noticeable. Among the five matting approaches, since the KNN algorithm performs best in original mattes, its improvement of matte quality by our method is relatively smaller than for the other four approaches.

4.2.2 Error comparison of separate matte

Moreover, from the aspect of each error type, comparisons of the separate mattes' values of 12 groups of produced alpha mattes are intuitively demonstrated from Fig. 4.13 to Fig. 4.17.

As show in Fig. 4.13, the MSE, SAD and gradient error of alpha mattes from the closed-form method [25] are at the same level of that from the proposed method for Test Image 1. The Test Image 1 is also used in the Experiment 2 of the visual quality comparison and the related discussion is given in Section 4.1.

Besides, if we only consider gradient of generated alpha matte, Fig. 4.13 (c) displays an evident enhancement of the image quality. The line of original method goes all above that of the proposed method, excluding the starting points of both lines. Specifically, the proposed method has the most significant refinement in the gradient error of the matte for

image 7, with the scale of variation at almost 0.03 (see Fig. 4.13 (c)). When it comes to MSE or SAD values, the most noticeable error decrease occurs at in image 8.

Fig. 4.14 illustrates the error comparison of each group of mattes created by KNN matting [26] and the proposed method. The largest error decline of our method can be found in image 1, varying from almost 0.02 to around 0.005 for MSE and from 0.05 to below 0.02 for SAD.

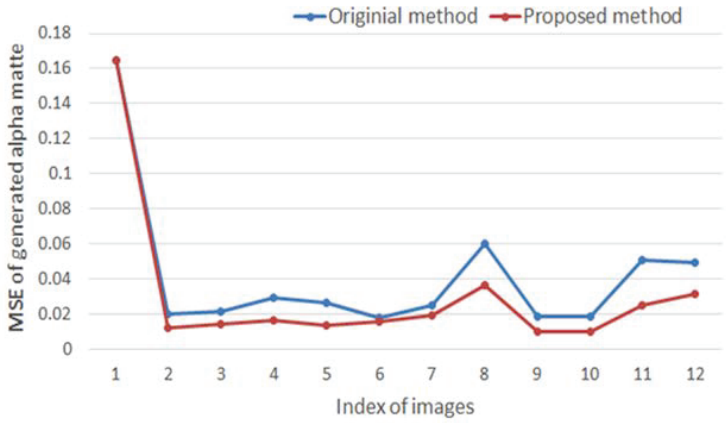
In addition, the level of gradient values for both KNN matting and the proposed one fluctuate a lot as displayed in Fig. 4.14 (c). In particular, the alpha matte of image 7 using proposed approach has a higher gradient error around 0.016. The possible reason could be explained in this way. When the image area of leaves and green flowerpot is processed, the alpha values by KNN matting vary smoothly even though those on the original matte are all lower than 1. In contrast, the post-processed matte by our method has rectified the alpha values to 1 for some pixels from that area. But it fails to do so for all pixels in the region. Therefore, the discontinuities of the alpha values in this region led to the relatively higher gradient.

Considering all three graphs in Fig. 4.15, it is observed that the respective parts of lines from image 7 to image 9 represent the most effective post-processing when dealing with the weighted color approach [27]. Also, as shown in Fig. 4.15 (c), our method efficiently lowers the gradient error of original mattes by the weighted color approach.

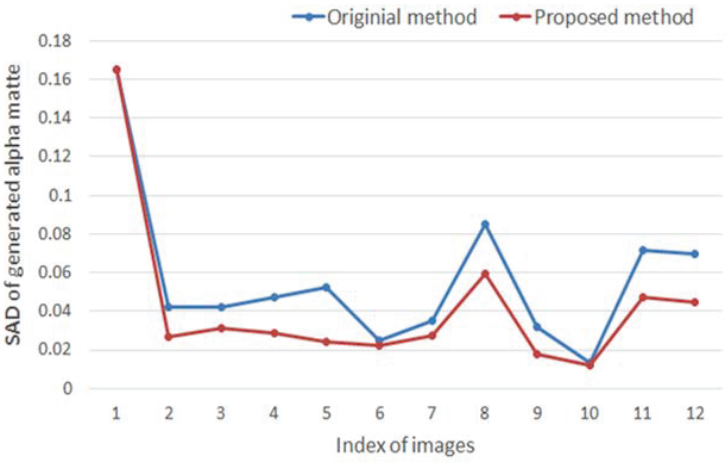
When applying the proposed method to the comprehensive sampling approach [28], Fig. 4.16 (a) and (b) have a similar trend pattern, except for the points in image 5. Another noticeable feature can be found at the values of MSE and SAD to image 11, where the value in Fig. 4.16 (a) are almost 0 and that in Fig. 4.16 (b) are just above 0. In this case, the improvement by our method may not be observed by the users visually.

The error graphs related to the learning based algorithm [29] are shown in Fig. 4.17. The fluctuation consistency of the first two graphs illustrates that the proposed method has a similar effectiveness to refine MSE and SAD of the original mattes. In addition,

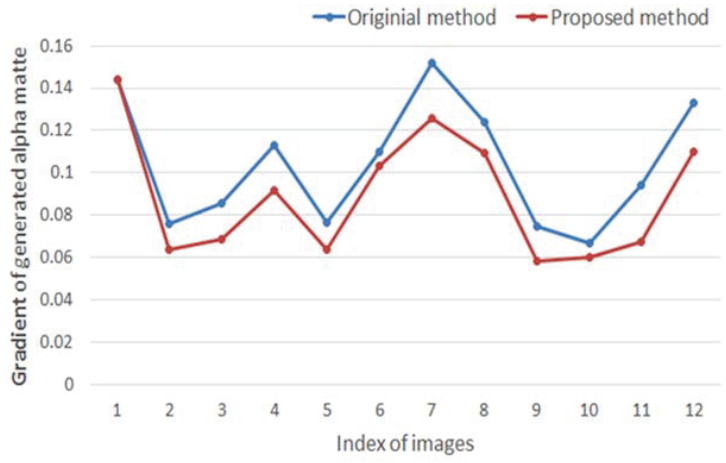
observing the gradient error in Fig. 4.17 (c), even though some parts of the blue line of learning based method almost overlap the red line of the proposed method (see the points of images 5, 6 and 10), there is still a clear gap between these two lines for most of points.



(a)

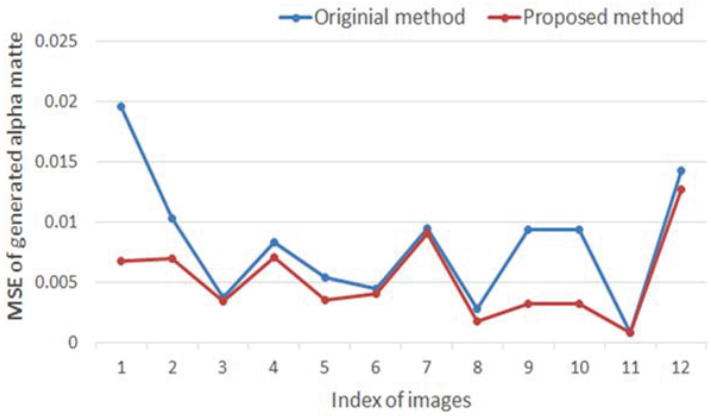


(b)

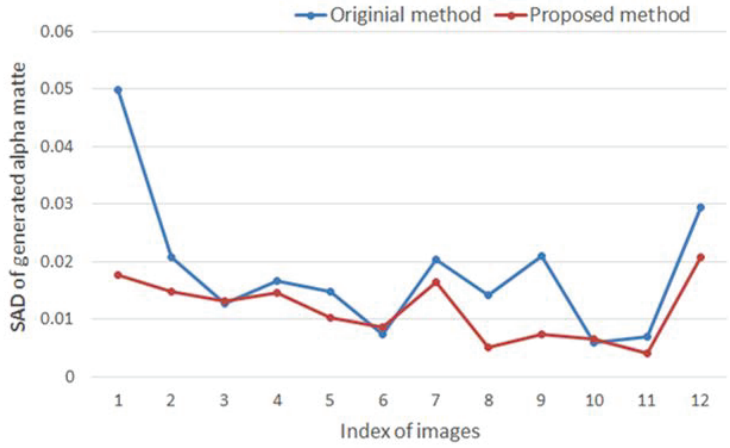


(c)

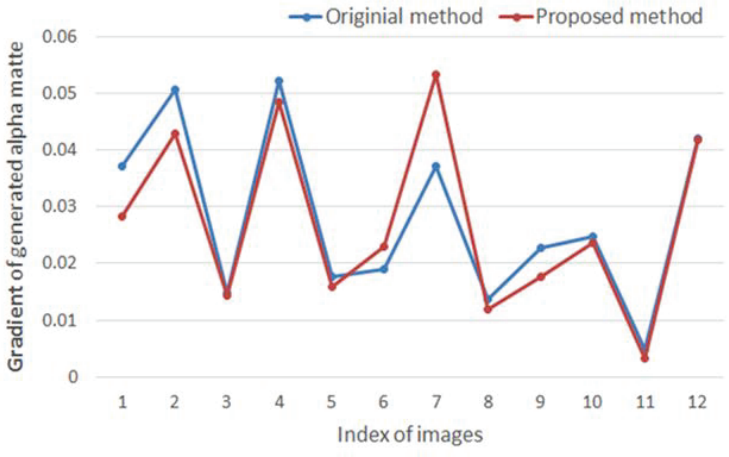
Figure 4.13: Comparison between the closed-form [25] and proposed post-processing methods. (a) MSE. (b) SAD. (c) Gradient.



(a)

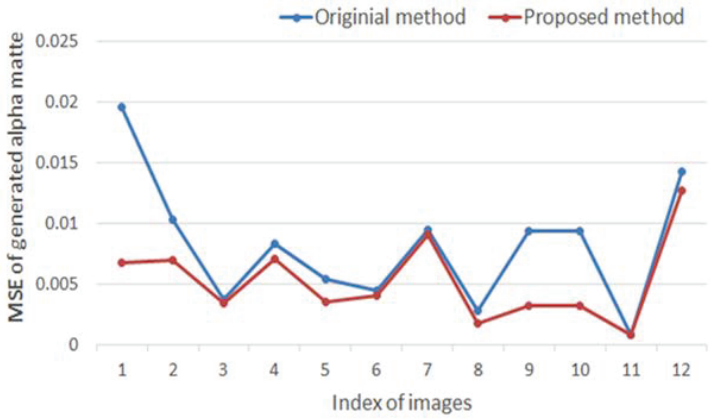


(b)

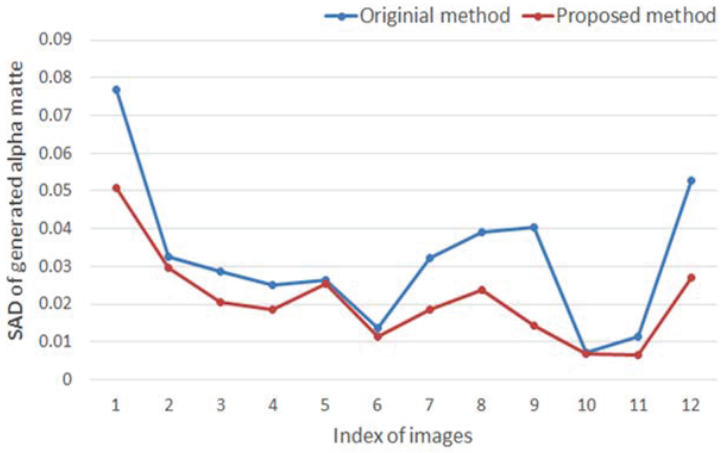


(c)

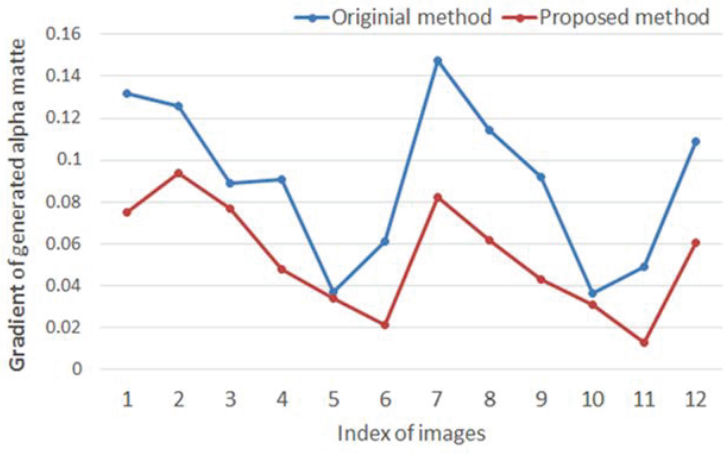
Figure 4.14: Comparison between KNN matting [26] and proposed post-processing methods. (a) MSE. (b) SAD. (c) Gradient.



(a)

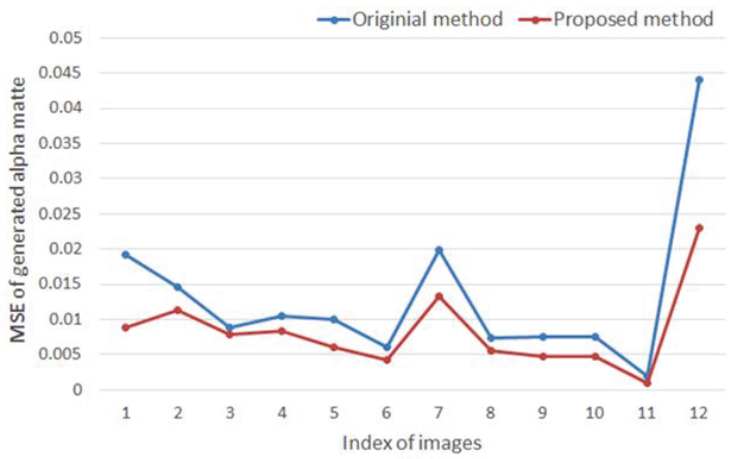


(b)

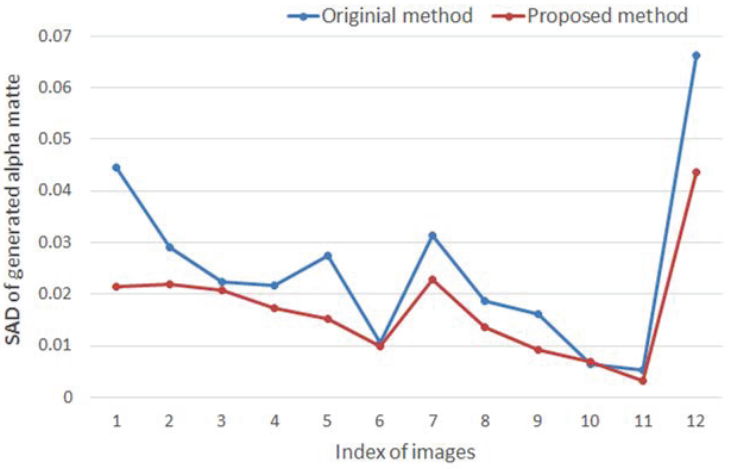


(c)

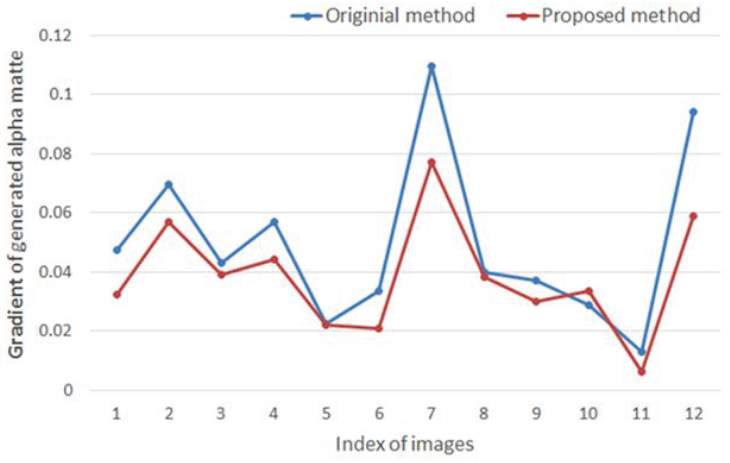
Figure 4.15: Comparison between the weighted color [27] and proposed post-processing methods. (a) MSE. (b) SAD. (c) Gradient.



(a)

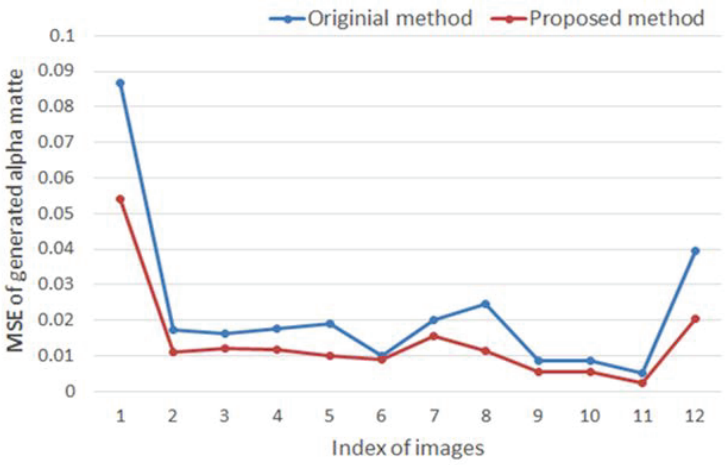


(b)

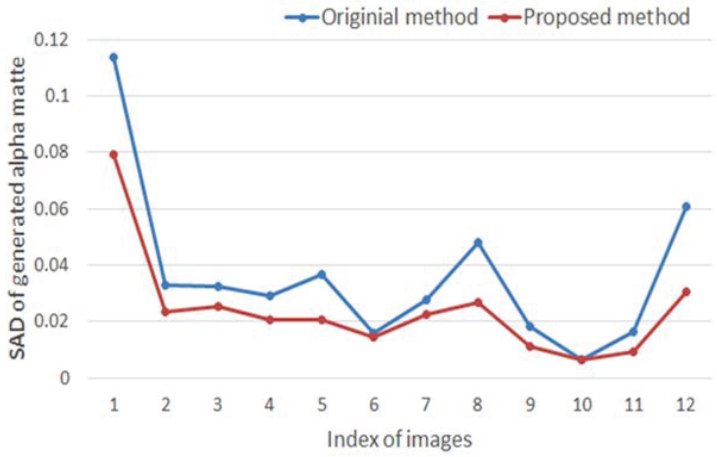


(c)

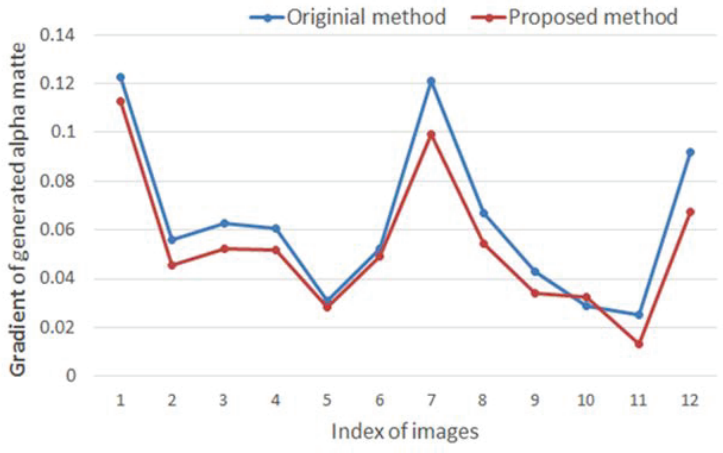
Figure 4.16: Comparison between the comprehensive sampling [28] and proposed post-processing methods. (a) MSE. (b) SAD. (c) Gradient.



(a)



(b)



(c)

Figure 4.17: Comparison between the learning based [26] and proposed post-processing methods. (a) MSE. (b) SAD. (c) Gradient.

Chapter 5

Conclusion and Future Work

5.1 Conclusion

In this thesis, a novel chroma keying method based on stereo images has been proposed. It can be applied as a post-process for the alpha matte generated by any existing matting approach. Given a pair of stereo images, a novel matting Laplacian matrix is constructed, which contains all pixels of two frames. Based on the new matting Laplacian matrix, a new cost function is also formulated to estimate alpha values of the reference image through propagation between stereo images. The proposed method is founded on the relevance between the corresponding pixels of the stereo images and the resemblance of their neighbors within a united window. In this case, it can propagate a more accurate matte than the original result. To avoid over-smoothing during propagation, a united window with adaptive size is used instead of a window with a fixed size for the Laplacian matrix. In the complex image regions such as fuzzy regions, narrow gaps or hollow holes, a small window is chosen. Otherwise, a big window is used for other regions. In this way, the proposed method can overcome the inadequate propagation and the over-smoothness due to improper sizes of windows. Moreover, an optional pre-processing is introduced. For a foreground object with strong reflection, based on the histogram statistics, the initial alpha

values of the particular unknown pixels can be refined before the proposed post-processing method. In addition, a ground truth generation method is developed to compute a matte to approximate ground truth.

As a post-processing procedure, the proposed method effectively decreases the errors of the original mattes obtained by current matting algorithms. Especially when some regions of foreground objects are missing in the original mattes, our method based on stereo images can capture and present more foreground details. Meanwhile, our method enables the alpha values of the absolute foreground pixels to be more accurate (closer to 1) when they are falsely estimated in original mattes. But there are some limitations in our work. The propagation based on stereo images may give rise to over-propagation in transparent regions. Accordingly, the pre-processing step is not suitable to deal with images with a transparent foreground object.

5.2 Future work

Reducing the computation of the proposed method will be our future work. Because stereo image brings twice as much data and information as a single image does, the computational cost of the new cost function is larger than conventional matting method. There could be some attempts, for instance, narrowing down the size of a united window. In our current work, all the pixels from the two squared windows are collected into the united window. So an attempt could be made that only the neighboring pixels with low color variance are selected into the united window to constitute the matting Laplacian matrix. By excluding samples with high color variance, the computation should be simplified, and the propagation could be more efficient and reliable.

References

[1] T. Porter and T. Duff, "Compositing digital images," in *Proceedings of ACM SIGGRAPH*, 1984, pp. 253–259.

[2] A. R. Smith and J. F. Blinn, "Blue screen matting," in *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques*, 1996, pp. 259–268.

[3] Q. Zhu, L. Shao, X. Li, and L. Wang, "Targeting accurate object extraction from an image: A comprehensive study of natural image matting," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 26, no. 2, pp. 185–207, 2015.

[4] J. Wang and M. F. Cohen, "Image and video matting: a survey," *Foundations and Trends in Computer Graphics and Vision*, vol. 3, no. 2, pp. 97–175, 2007.

[5] Y.-Y. Chuang, A. Agarwala, B. Curless, D. H. Salesin, and R. Szeliski, "Video matting of complex scenes," *ACM Transactions on Graphics*, vol. 21, no. 3, pp. 243–248, 2002.

[6] V.-Q. Pham, K. Takahashi, and T. Naemura, "Real-time video matting based on bilayer segmentation," in *Proceedings of Asian Conference on Computer Vision*, 2009, pp. 489–501.

[7] Y.-Y. Chuang, D. B. Goldman, B. Curless, D. H. Salesin, and R. Szeliski, "Shadow matting and compositing," *ACM Transactions on Graphics*, vol. 22, no. 3, pp. 494–500, 2003.

[8] T.-P. Wu, C.-K. Tang, M. S. Brown, and H.-Y. Shum, "Natural shadow matting," *ACM Transactions on Graphics*, vol. 26, no. 2, p. 8, 2007.

[9] D. E. Zongker, D. M. Werner, B. Curless, and D. H. Salesin, "Environment matting and compositing," in *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques*, 1999, pp. 205–214.

[10] Y.-Y. Chuang, D. E. Zongker, J. Hindorff, B. Curless, D. H. Salesin, and R. Szeliski, "Environment matting extensions: Towards higher accuracy and real-time capture," in *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques*, 2000, pp. 121–130.

[11] Available: https://en.wikipedia.org/wiki/Chroma_key, Accessed: 09/15/2016.

[12] C. Rhemann, C. Rother, A. Rav-Acha, and T. Sharp, "High resolution matting via interactive trimap segmentation," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2008, pp. 1–8.

[13] Y. Guan, W. Chen, X. Liang, Z. Ding, and Q. Peng, "Easy matting-a stroke based approach for continuous image matting," in *Proceedings of Computer Graphics Forum*, 2006, pp. 567–576.

[14] "Alpha matting benchmark," http://www.alphamatting.com/, Accessed: 09/10/2016.

[15] J. Wang and M. F. Cohen, "An iterative optimization approach for unified image segmentation and matting," in *Proceedings of 10th IEEE International Conference on Computer Vision*, 2005, pp. 936–943.

[16] P. Lee and Y. Wu, "Nonlocal matting," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2011, pp. 2193–2200.

[17] Y. Furukawa and C. Hernández, *Multi-view stereo: A tutorial*. Now Publishers Incorporated, 2015.

[18] A. Delaunoy and E. Prados, "Gradient flows for optimizing triangular mesh-based surfaces: Applications to 3D reconstruction problems dealing with visibility," *International Journal of computer vision*, vol. 95, no. 2, pp. 100–123, 2011.

[19] Y. Furukawa and J. Ponce, "Accurate, dense, and robust multiview stereopsis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 8, pp. 1362–1376, 2010.

[20] C. H. Esteban and F. Schmitt, "Silhouette and stereo fusion for 3D object modeling," *Computer Vision and Image Understanding*, vol. 96, no. 3, pp. 367–392, 2004.

[21] Google, "Google maps," https://www.google.com/streetview/, Accessed: 09/10/2016.

[22] D. Marr, "Vision: A computational investigation into the human representation and processing of visual information," Henry Holt & Company, 1982.

[23] Available: https://en.wikipedia.org/wiki/Stereoscopy, Accessed: 09/02/2016.

[24] W. Wang and J. Zhao, "Robust image chroma-keying: a quadmap approach based on global sampling and local affinity," *IEEE Transactions on Broadcasting*, vol. 61, no. 3, pp. 356–366, 2015.

[25] A. Levin, D. Lischinski, and Y. Weiss, "A closed-form solution to natural image matting," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 2, pp. 228–242, 2008.

[26] Q. Chen, D. Li, and C.-K. Tang, "KNN matting," *IEEE Transactions on Pattern Analysis and Mmachine Intelligence*, vol. 35, no. 9, pp. 2175–2188, 2013.

[27] E. S. Varnousfaderani and D. Rajan, "Weighted color and texture sample selection for image matting," *IEEE Transactions on Image Processing*, vol. 22, no. 11, pp. 4260–4270, 2013.

[28] E. Shahrian, D. Rajan, B. Price, and S. Cohen, "Improving image matting using comprehensive sampling sets," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 636–643.

[29] Y. Zheng and C. Kambhamettu, "Learning based digital matting," in *Proceedings of 12th IEEE International Conference on Computer Vision*, 2009, pp. 889–896.

[30] Y. Mishima, "Soft edge chroma-key generation based upon hexoctahedral color space," US Patent 5,355,174, Oct. 11, 1994.

[31] P. Vlahos, "Comprehensive electronic compositing system," US Patent 4,625,231, Nov. 25, 1986.

[32] F. D. Williams, "Method of taking motion-pictures," US Patent 1,273,435, Jul. 23, 1918.

[33] C. D. Dunning, "Composite photography," *Transactions of the Society of Motion Picture Engineers*, vol. 12, no. 36, pp. 975–979, 1928.

[34] P. Vlahos, "Composite photography utilizing sodium vapor illumination," US Patent 3,095,304, Jun. 25, 1963.

[35] P. Vlahos, "Electronic composite photography," US Patent 3,595,987, Jul. 27, 1971.

[36] P. Vlahos, "Electronic composite photography with color control," US Patent 4,007,487, Feb. 8, 1977.

[37] P. Vlahos, "Comprehensive electronic compositing system," US Patent 4,100,569, Jul. 11, 1978.

[38] A. W. Tondreau and F. W. Gage, "Process of composite photography," US Patent 2,651,233, Sep. 8, 1953.

[39] P. Vlahos, "Composite color photography," US Patent 3,158,477, Nov. 24, 1964.

[40] P. Vlahos, A. Dadourian, and G. Sauve, "Method and apparatus for adjusting parameters used by compositing devices," US Patent 5,907,315, May 25, 1999.

[41] J. Wang and M. F. Cohen, "Optimized color sampling for robust matting," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2007, pp. 1–8.

[42] A. Levin, A. Rav-Acha, and D. Lischinski, "Spectral matting," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 10, pp. 1699–1712, 2008.

[43] D. J. Chaplin, P. M. Trethewey, and P. S. Miller, "Graphical display of chroma keyer controls," US Patent 5,627,951, May 6, 1997.

[44] D. Pettigrew and F. Paquin, "Color diamond chroma keying," US Patent 6,751,347, Jun. 15, 2004.

[45] M. A. Ruzon and C. Tomasi, "Alpha estimation in natural images," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2000, pp. 18–25.

[46] Y.-Y. Chuang, B. Curless, D. H. Salesin, and R. Szeliski, "A Bayesian approach to digital matting," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2001, pp. 264–271.

[47] A. Berman, A. Dadourian, and P. Vlahos, "Method for removing from an image the background surrounding a selected object," US Patent 6,134,346, Oct. 17, 2000.

[48] A. Berman, P. Vlahos, and A. Dadourian, "Comprehensive method for removing from an image the background surrounding a selected subject," US Patent 6,134,345, Oct. 17, 2000.

[49] E. S. Gastal and M. M. Oliveira, "Shared sampling for real-time alpha matting," *Computer Graphics Forum*, vol. 29, no. 2, pp. 575–584, 2010.

[50] K. He, C. Rhemann, C. Rother, X. Tang, and J. Sun, "A global sampling method for alpha matting," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2011, pp. 2049–2056.

[51] C. Rhemann, C. Rother, and M. Gelautz, "Improving color modeling for alpha matting." in *Proceedings of British Machine Vision Conference*, 2008, p. 3.

[52] E. Shahrian and D. Rajan, "Using texture to complement color in image matting," *Image and Vision Computing*, vol. 31, no. 9, pp. 658–672, 2013.

[53] J. Johnson, E. S. Varnousfaderani, H. Cholakkal, and D. Rajan, "Sparse coding for alpha matting," *IEEE Transactions on Image Processing*, vol. 25, no. 7, pp. 3032–3043, 2016.

[54] J. Sun, J. Jia, C.-K. Tang, and H.-Y. Shum, "Poisson matting," *ACM Transactions on Graphics*, vol. 23, no. 3, pp. 315–321, 2004.

[55] L. Grady, T. Schiwietz, S. Aharon, and R. Westermann, "Random walks for interactive alpha-matting," in *Proceedings of International Conference on Visualization, Imaging, and Image Processing*, 2005, pp. 423–429.

[56] X. Niyogi, "Locality preserving projections," in *Proceedings of Neural Information Processing Systems*, 2004, p. 153.

[57] L. Grady, "Random walks for image segmentation," *IEEE Transactions on Pattern Analysis and Mmachine Intelligence*, vol. 28, no. 11, pp. 1768–1783, 2006.

[58] K. He, J. Sun, and X. Tang, "Fast matting using large kernel matting laplacian matrices," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2010, pp. 2165–2172.

[59] A. Buades, B. Coll, and J.-M. Morel, "A non-local algorithm for image denoising," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2005, pp. 60–65.

[60] C. Rother, V. Kolmogorov, and A. Blake, "Grabcut: Interactive foreground extraction using iterated graph cuts," *ACM Transactions on Graphics*, vol. 23, no. 3, pp. 309–314, 2004.

[61] Y. Y. Boykov and M.-P. Jolly, "Interactive graph cuts for optimal boundary & region segmentation of objects in nd images," in *Proceedings of 8th IEEE International Conference on Computer Vision*, 2001, pp. 105–112.

[62] Z. Zhang, Q. Zhu, and Y. Xie, "Learning based alpha matting using support vector regression," in *Proceedings of 19th IEEE International Conference on Image Processing*, 2012, pp. 2109–2112.

[63] Q. Zhu, Z. Zhang, Z. Song, Y. Xie, and L. Wang, "A novel nonlinear regression approach for efficient and accurate image matting," *IEEE Signal Processing Letters*, vol. 20, no. 11, pp. 1078–1081, 2013.

[64] P. W. Holland, "Weighted ridge regression: Combining ridge and robust regression methods," National Bureau of Economic Research Cambridge, 1973.

[65] J. Wang, "Image matting with transductive inference," in *Proceedings of International Conference on Computer Vision/Computer Graphics Collaboration Techniques and Applications*, 2011, pp. 239–250.

[66] B. He, G. Wang, C. Shi, X. Yin, B. Liu, and X. Lin, "Iterative transductive learning for alpha matting," in *Proceedings of IEEE International Conference on Image Processing*, 2013, pp. 4282–4286.

[67] M. Okutomi and T. Kanade, "A multiple-baseline stereo," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 15, no. 4, pp. 353–363, 1993.

[68] Y. Yang, A. Yuille, and J. Lu, "Local, global, and multilevel stereo matching," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 1993, pp. 274–279.

[69] H. Hirschmuller and D. Scharstein, "Evaluation of cost functions for stereo matching," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2007, pp. 1–8.

[70] D. Scharstein and R. Szeliski, "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms," *International Journal of Computer Vision*, vol. 47, no. 1-3, pp. 7–42, 2002.

[71] P. Anandan, "A computational framework and an algorithm for the measurement of visual motion," *International Journal of Computer Vision*, vol. 2, no. 3, pp. 283–310, 1989.

[72] E. P. Simoncelli, E. H. Adelson, and D. J. Heeger, "Probability distributions of optical flow," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 1991, pp. 310–315.

[73] T. Kanade, H. Kano, S. Kimura, A. Yoshida, and K. Oda, "Development of a video-rate stereo machine," in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, 1995, pp. 95–100.

[74] M. J. Black and A. Rangarajan, "On the unification of line processes, outlier rejection, and robust statistics with applications in early vision," *International Journal of Computer Vision*, vol. 19, no. 1, pp. 57–91, 1996.

[75] D. Scharstein and R. Szeliski, "Stereo matching with nonlinear diffusion," *International Journal of computer vision*, vol. 28, no. 2, pp. 155–174, 1998.

[76] S. Birchfield and C. Tomasi, "A pixel dissimilarity measure that is insensitive to image sampling," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 4, pp. 401–406, 1998.

[77] A. F. Bobick and S. S. Intille, "Large occlusion stereo," *International Journal of Computer Vision*, vol. 33, no. 3, pp. 181–200, 1999.

[78] T. Kanade and M. Okutomi, "A stereo matching algorithm with an adaptive window: Theory and experiment," *IEEE Transactions on Pattern analysis and Machine Intelligence*, vol. 16, no. 9, pp. 920–932, 1994.

[79] O. Veksler, "Stereo matching by compact windows via minimum ratio cycle," in *Proceedings of 8th IEEE International Conference on Computer Vision*, 2001, pp. 540–547.

[80] Y. Boykov, O. Veksler, and R. Zabih, "A variable window approach to early vision," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 12, pp. 1283–1294, 1998.

[81] D. Terzopoulos, "Regularization of inverse visual problems involving discontinuities," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 8, no. 4, pp. 413–424, 1986.

[82] S. Geman and D. Geman, "Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 6, no. 6, pp. 721–741, 1984.

[83] R. Szeliski and J. Coughlan, "Spline-based image registration," *International Journal of Computer Vision*, vol. 22, no. 3, pp. 199–218, 1997.

[84] P. Fua, "A parallel stereo algorithm that produces dense depth maps and preserves image features," *Machine Vision and Applications*, vol. 6, no. 1, pp. 35–49, 1993.

[85] Y. Boykov, O. Veksler, and R. Zabih, "Fast approximate energy minimization via a graph cuts," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 11, pp. 1222–1239, 2001.

[86] A. Blake and A. Zisserman, *Visual reconstruction.* MIT press, 1987.

[87] P. B. Chou and C. M. Brown, "The theory and practice of Bayesian image labeling," *International Journal of Computer Vision*, vol. 4, no. 3, pp. 185–210, 1990.

[88] D. Geiger and F. Girosi, "Parallel and deterministic algorithms from MRFs: Surface reconstruction," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 13, no. 5, pp. 401–412, 1991.

[89] O. Veksler, "Efficient graph-based energy minimization methods in computer vision," Ph.D. dissertation, Cornell University, 1999.

[90] V. Kolmogorov and R. Zabih, "Computing visual correspondence with occlusions using graph cuts," in *Proceedings of 8th IEEE International Conference on Computer Vision*, 2001, pp. 508–515.

[91] I. J. Cox, S. L. Hingorani, S. B. Rao, and B. M. Maggs, "A maximum likelihood stereo algorithm," *Computer Vision and Image Understanding*, vol. 63, no. 3, pp. 542–567, 1996.

[92] S. Birchfield and C. Tomasi, "Depth discontinuities by pixel-to-pixel stereo," *International Journal of Computer Vision*, vol. 35, no. 3, pp. 269–293, 1999.

[93] C. L. Zitnick and T. Kanade, "A cooperative algorithm for stereo matching and occlusion detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 7, pp. 675–684, 2000.

[94] J. R. Bergen, P. Anandan, K. J. Hanna, and R. Hingorani, "Hierarchical model-based motion estimation," in *Proceedings of European Conference on Computer Vision*, 1992, pp. 237–252.

[95] M. Shimizu and M. Okutomi, "Precise subpixel estimation on area-based matching," *Systems and Computers in Japan*, vol. 33, no. 7, pp. 1–10, 2002.

[96] R. Szeliski and P. Golland, "Stereo matching with transparency and matting," *International Journal of Computer Vision*, vol. 32, no. 1, pp. 45–61, 1999.

[97] J.-H. Cho, T. Yamasaki, K. Aizawa, and K. H. Lee, "Depth video camera based temporal alpha matting for natural 3D scene generation," in *Proceedings of 3DTV*

Conference: The True Vision-Capture, Transmission and Display of 3D Video, 2011, pp. 1–4.

[98] W. Sun, O. C. Au, L. Xu, and Z. Yu, "Adaptive depth map assisted matting in 3D video," in *Proceedings of IEEE International Conference on Multimedia and Expo*, 2011, pp. 1–6.

[99] K. Karsch, C. Liu, and S. B. Kang, "Depth extraction from video using non-parametric sampling," in *Proceedings of European Conference on Computer Vision*, 2012, pp. 775–788.

[100] Adobe, "After effects," http://www.adobe.com/products/aftereffects.html, Accessed: 11/10/2016.

[101] Primatte, "Whitepapers," http://www.primatte.com/content.cfm?n=whitepapers, Accessed: 11/10/2016.

[102] C. Rhemann, C. Rother, J. Wang, M. Gelautz, P. Kohli, and P. Rott, "A perceptually motivated online benchmark for image matting," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 1826–1833.