Constant colour matting with foreground estimation

Guillaume GALES
National University of Ireland Maynooth
Department of Computer Science
guillaume.gales@nuim.ie

John MC DONALD
National University of Ireland Maynooth
Department of Computer Science
johnmcd@cs.nuim.ie

Abstract

Constant colour matting consists of estimating for each pixel of an image the proportion α of an unknown foreground colour with a known constant background colour. The α-matte is then used to replace this background with another image. Existing approaches approximate α directly but post-processing is required to remove spill of the background colour in semi-transparent areas. Instead of estimating α directly, we propose 3 methods to estimate the unknown foreground colour, and then to deduce α. This approach leads to high quality mattes for transparent objects and allows spill-free results (see Fig. 1). We show this through an evaluation of the proposed methods based on a ground truth dataset.

Keywords: constant colour matting, foreground colour estimation, α estimation.

1 Introduction

Matting is a classic problem that consists of creating a *matte* to mask an unwanted area from video footage. This area is then replaced by content from separate footage to create a *composite*. Cinema, television and web TV use this technique extensively for visual special effects. Although the principle of this technique is simple, it is often difficult to achieve a realistic seamless result. This is particularly true where the observed colour is a blend from foreground and background. The proportion of foreground colour F, and background colour B for one pixel is called α. This typically occurs at the boundary between both areas, with semi-transparent foreground, or blur. We distinguish between two types of matting techniques: *constant colour matting* techniques, where the background colour is known, or *natural image matting*, working with an arbitrary background. The former methods are widely used in visual productions but post-processing is usually required to clean up the matte and to deal with spill (background colour reflecting in the foreground). The latter methods can give impressive results but require additional prior information and are more computationally expensive.

In this paper, our goal is to estimate the F and α, knowing B, to obtain high quality results in difficult areas exhibiting fine and semi-transparent details. Unlike others constant colour methods, we start by estimating F, inspired by natural image matting approaches, but without

Input

Figure 1: Examples of final results obtained with the proposed method.

the need to input additional information. Our contributions are three methods for the estimation of F (and consequently α) that give mattes requiring little or no cleaning, and produce spill-free results. Furthermore, our algorithm is highly parallelisable (working independently on each pixel) with a low computational complexity. We also provide a ground truth dataset which we use to demonstrate and quantitatively evaluate the performance of our technique.

After a brief description of the state of the art, we detail our approach. Then, we provide an evaluation of the three methods based on a a proposed ground truth dataset.

2 Previous work

2.1 Constant colour matting

Constant colour matting aims at estimating a matte from images where the background is assumed to be a constant colour (usually blue or green).

In [Smith and Blinn, 1996], the authors formalise the problem of constant colour matting as follows. For each pixel, we express the observed colour (O, known) as the proportion $\alpha \in [0;1]$ (unknown) of foreground (F, unknown) and background (B, known) colours. The *matting equation* is given by:

$$\underbrace{\begin{bmatrix} r & g & b \end{bmatrix}^\top}_{O} = \alpha \underbrace{\begin{bmatrix} F_r & F_g & F_b \end{bmatrix}^\top}_{F} + (1-\alpha) \underbrace{\begin{bmatrix} B_r & B_g & B_b \end{bmatrix}^\top}_{B} \tag{1}$$

There are four unknowns for three equations and therefore the problem is underdetermined, i.e. it exists no or many solutions. The authors identify three cases where a solution can be found: no blue in the observed colour, the observed colour is gray or two different shades of background are known. However, these cases do not usually occur in practice.

We can distinguish between the following types of method to estimate α with a constant background colour:

– *Colour difference technique* – α is based on differences between the red, green and blue components. This method (a.k.a *Ultimatte*®), invented by [Vlahos, 1964], is a legacy of an optical multistep process where colour filters are placed in front of an optical printer to filter out the background colour (an interesting history of matting in filmmaking is given in [Filmmaker IQ, 2013]). This process is usually summarised by $\alpha = 1 - \max(0, b - \max(r, g))$ where B is assumed to be blue. Spill is then removed by changing the blue component to $b \leftarrow \min(b, kg)$ where k is a user control parameter.

– *Colorspace segmentation* – The colorspace is partitioned into background, foreground and semi-transparent regions. In the semi-transparent region, α is based on the distance between the other two. In [Ashikhmin, 2001, Jack, 1996], they use the YC_bC_r colorspace to separate the luminance (Y) and the chrominance (C_bC_r). The segmentation is performed in the 2D chrominance space. A classic approach, called *Hue Saturation Luminance keying* and described in [Schultz, 2006], consists of segmenting the HSL colorspace to isolate the background and foreground regions. These segmentations are done manually by selecting the center and the dimensions of basic shapes (simple polyhedron or sphere) that encapsulates the different regions. The *Primatte*® algorithm by [Mishima, 1992], is based on this principle, however it automatically adjusts a 128 face polyhedron to obtain an fine segmentation of the colorspace.

2.2 Natural image matting

Natural image matting aims at estimating a matte from images with arbitrary background. These methods use optimisation techniques to estimate the best combination of α, F and B that minimises an objective function based on spatial statistical models. To build these models, they require a pre-segmentation (called *trimap*) in three classes: foreground, background and unknown. For example, in [Chuang et al., 2001], the algorithm marches inward from known to unknown regions. It uses the colour distribution in a weighted window of the known (or already computed) neighbouring regions to estimate the most likely combination of α, F and B.

In [Rother et al., 2004], an adapted iterative graph cut optimisation method is used to minimise an objective function that takes into account a fitting term (how close the solution is to what is observed) and smoothness term (to prevent abrupt changes of α between two neighbours). These methods use local sampling for the estimation of the foreground and background colours. In [He et al., 2011], the authors obtain good results with a global sampling approach for the estimation of F and B. A detailed survey of such methods is provided in [Wang and Cohen, 2007].

3 Our approach

Our approach starts by building, for each pixel, a set of candidates for the foreground colour F. These candidates are computed from a set of predefined possible colours $\mathcal{C}$ for the image. We propose three independent methods for this computation. Then, we assign to F the best candidate according to a distance measure between the candidate and the observed colour. Finally, assuming F known, we can calculate α.

First, we describe how we obtain the set of possible colours $\mathcal{C}$. Then, we give the general structure of the algorithm. Finally, we present the three methods to estimate F.

3.1 Set of possible foreground colours for the image

We reduce the solution space for the foreground colour F using the following constraints:
(i) F should be already present in the image. – As in natural image methods, the first assumption is that F, is already present in the image. To obtain a set of possible colours for F, we calculate the modes of the colour distribution of the image. To do so, we use the *mean-shift* algorithm on the image histogram, [Comaniciu and Meer, 2002] using the *Lab* colorspace as it is perceptually uniform. The output is a set $\mathcal{C}_0$ of colour clusters.
(ii) F should be distinct from B. – We need to remove from the set $\mathcal{C}_0$ the clusters C_i that are too close to B: C_i is removed from $\mathcal{C}_0$ if $\|\overrightarrow{BC_i}\| < t_1$. Let $\mathcal{C} = \mathcal{C}_0 \setminus \mathcal{C}_B$ with $\mathcal{C}_B = \left\{ C_i \middle| \|\overrightarrow{BC_i}\| < t_1 \right\}$.
(iii) If $B \neq O$, F lies on the line passing through B and O. – By definition of Eq. (1).

3.2 Algorithm

Algorithm 1 gives a high-level description of the approach used to estimate F and α from a set of possible colour clusters $\mathcal{C}$. It starts by looking at the distance between B and O. If $B = O$, the given pixel is a background pixel. On the other hand, if this distance is large enough (above a threshold th_1), one can be confident that the given pixel is a foreground pixel. For the other pixels, each colour cluster $C_i \in \mathcal{C}$ is used to estimate F_i and a cost c, as described in the Section 3.3, depending on the chosen method m. This cost evaluates how "close" C_i is to the line BO . If it is too "far away" (above a threshold th_2) from BO (not satisfying the condition (iii)), the estimated F_i is rejected. Finally, if no candidate can be found, we assume the given pixel belongs to the background. If more than one candidate are found, we chose the closest one from the observed colour O.

3.3 Estimation of F

Ideally, if we can find exactly one colour cluster C_i lying on the line BO, satisfying the condition (iii), we could assume $F = C_i$. In practice, this alignement does not occur because of noise and because the colour clusters correspond to the mode of the colour distribution for F. To deal with this issue, we propose three independent methods to estimate F_i from a cluster C_i (see Table 1):

(a) This method minimises the sum of squared residuals of the matting equation system where F is replaced by C_i. Geometrically this solution minimises $\|\overrightarrow{O_i'O}\|$ where O_i' is the orthogonal projection of the observed colour O on BC_i. Thus, F_i is given by the intersection of the line BO and the plane passing through C_i orthogonal to the line BC_i. The cost c_i is given by: $c_i = \|\|\|\overrightarrow{C_iF_i}\|\|\|$.

Algorithm 1: Algorithm for one pixel.

Data: $O, B, \mathcal{C},\ th_1, th_2, m$
Result: F, α
if $\|\overrightarrow{BO}\| = 0$ **then**
  $F \leftarrow \mathbf{0}\ ;\ \alpha \leftarrow 0$ `/* This is a background pixel. */`
else
  if $\|\overrightarrow{BO}\| > th_1$ **then**
    $F \leftarrow O\ ;\ \alpha \leftarrow 1$ `/* This is a foreground pixel. */`
  else
    $\mathcal{L} \leftarrow \varnothing$ `/* Initiate a list of candidates. */`
    for *each* $C_i \in \mathcal{C}$ **do**
      $F_i \leftarrow$ estimate F with a method m using (O,B,C_i)
      $c_i \leftarrow$ cost for this F_i
      if $c_i < th_2$ **then**
        $\mathcal{L} \leftarrow \mathcal{L} \cap F_i$ `/* $F_i$ is a candidate. */`
    if $|\mathcal{L}| = 0$ **then**
      $F \leftarrow \mathbf{0}\ ;\ \alpha \leftarrow 0$ `/* No candidate → background. */`
    else
      `/* Get the best candidate. */`
      $F \leftarrow F_i \in \mathcal{L} \mid \forall\, F_i, F_j \in \mathcal{L}^2, \|\overrightarrow{OF_i}\| \leq \|\overrightarrow{OF_j}\|\ ;\ \alpha \leftarrow \frac{\|\overrightarrow{BO}\|}{\|\overrightarrow{BF}\|}$

(b) In this method, F_i is given by the orthogonal projection of the C_i onto the line BO. The cost is given by the Euclidean distance between C_i and F_i: $c_i = \|\overrightarrow{C_iF_i}\|$.

(c) This method rotates C_i around B with an angle $\theta_i = \cos^{-1}\left(\frac{\overrightarrow{BC_i}\overrightarrow{BO}}{\|\overrightarrow{BC_i}\|\|\overrightarrow{BO}\|}\right)$ so that B, O and C_i are collinear. The cost is given by the rotation angle: $c_i = \theta_i$.

(a)	(b)	(c)
$F_i = \frac{\overrightarrow{BO}\|\overrightarrow{BC_i}\|^2}{\overrightarrow{BO}\cdot\overrightarrow{BC_i}} + B$	$F_i = \frac{\overrightarrow{BO}(\overrightarrow{BO}\cdot\overrightarrow{BC_i})}{\|\overrightarrow{BO}\|^2} + B$	$F_i = \frac{\overrightarrow{BO}}{\|\overrightarrow{BO}\|}\|\overrightarrow{BC_i}\| + B$

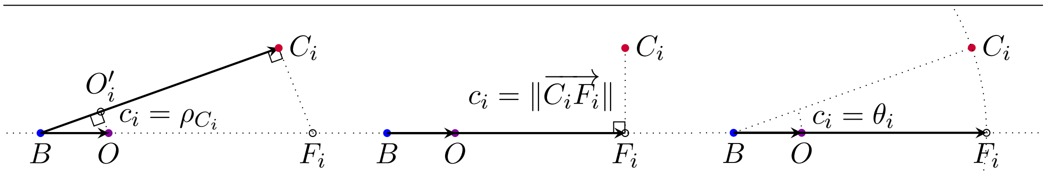

Table 1: Illustration of the 3 methods (a), (b) and (c) proposed to estimate F_i in the Algorithm 1.

4 Evaluation and results

Ground truth dataset To provide a quantitative evaluation of our three methods, we created a ground-truth dataset. As explained in [Smith and Blinn, 1996], if at least two different shades of background are known, Eq. 1 becomes overdetermined and we can estimate α and F. We took pictures of six different objects ($bath$, $bottle$, $muppet$, $pot1$, $pot2$ and $spider$, see Table 3) in front of five different backgrounds (blue, green, black, yellow and red) for further overdetermination. Then, we calculated a least squared solution for αF and α:

$$\begin{bmatrix} -B_{blue} & -B_{green} & \dots \\ \mathbf{I}_3 & \mathbf{I}_3 & \dots \end{bmatrix}^\top \begin{bmatrix} \alpha \\ \alpha F \end{bmatrix} = \begin{bmatrix} (O_{blue} - B_{blue}) & (O_{green} - B_{green}) & \dots \end{bmatrix}^\top \tag{2}$$

where $\mathbf{I}_3$ is the 3×3 identity matrix.

Parameters Each method requires two parameters for the initial clustering: the size of the bins s_b, and the size of the mean-shift window s_{ms}. It also requires th_1, the minimum $\|\overrightarrow{BF}\|$

distance, and th_2 the threshold on the cost depending on the method employed, see § 3.2. According to an initial experiment, (s_b, s_{ms}) can be fixed to $(2, 4)$ for the method (a) (giving about 60 clusters) and to $(2, 2)$ for the methods (b) and (c) (giving about 400 clusters). The best results (according to the criterion described below) are obtained with $th_1 \approx 40 \pm 5$ and (a) $th_2 \approx 2$; (b) $th_2 \approx 10$; (c) $th_2 \approx 0.4$ (see Tables 3 and 2). These values can then be fine tuned interactively.

Measure of error For each of the three methods m, we evaluate the results obtained with a different set of values for th_1 and th_2. We chose to evaluate the methods using the images having a green background. As we are interested in F and α, we compare the estimated values with the ground truth. The mean squared error for an image is given by:

$$MSE_{th_1,th_2,m} = \frac{1}{hw} \sum_{i,j=0,0}^{h-1,w-1} \|(\alpha F)_{i,j} - (\overline{\alpha F})_{i,j}\| \tag{3}$$

where (h, w) are the dimensions of the image, $(\alpha F)_{i,j}$ is the estimated value for the pixel at coordinates (i, j) and $(\overline{\alpha F})_{i,j}$ is the ground truth value for the pixel of same coordinates.

Tables 2 and 3 show the results. With the appropriate set of parameter values, the three methods can achieve results with a low error score. The best average values for each method are $MSE_{45,3,(a)} = 3.339$, $MSE_{50,5,(b)} = 2.814$ and $MSE_{50,0.4,(c)} = 2.811$ showing that (b) and (c) perform slightly better than (a). In some scenes, (b) and (c) clearly outperform (a).

(a)	(b)	(c)

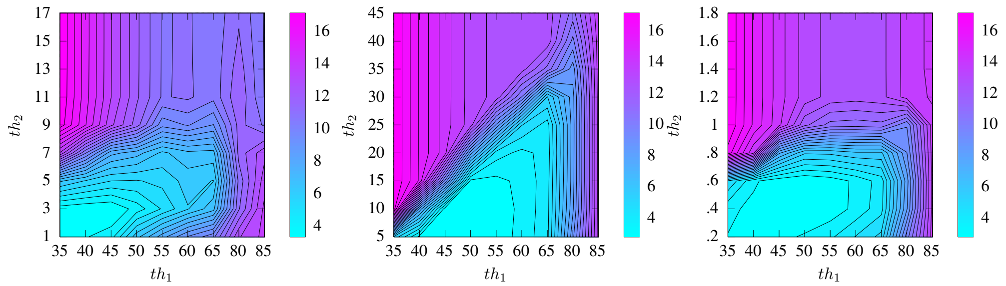

Table 2: Contour plots of the averaged $MSE_{th_1,th_2,m}$ over the 6 images for each of the three methods in function of th_1 and th_2.

5 Discussion and conclusion

We proposed three methods to estimate F and α in a constant colour background matting problem. According to our evaluation, these methods give very encouraging results.

A comparison with commercial methods would be interesting. However, source code is not available, executables are not free and they may include extra post processing to give a visually appealing, but not mathematically accurate, result. These issues make a fair and rigorous comparison difficult. But, to permit a qualitative evaluation of how our algorithm may compete, we visually compared our result (left) with one obtained using *Apple Motion HSL Keyer* (right). The latter one has more spill. We also noticed one inconsistent ground truth data in *Pot1* where the opaque fluorescent marker is considered to have some transparency. Although the variances in the observed values are relatively small (due to noise and background indirect illumination), the residual of Eq. 2 is quite large. We propose to re-estimate the ground truth using a robust

(a) *Our method*

(b) *Apple Motion HSL Keyer*

Figure 2: Visual comparison with a commercial solution.

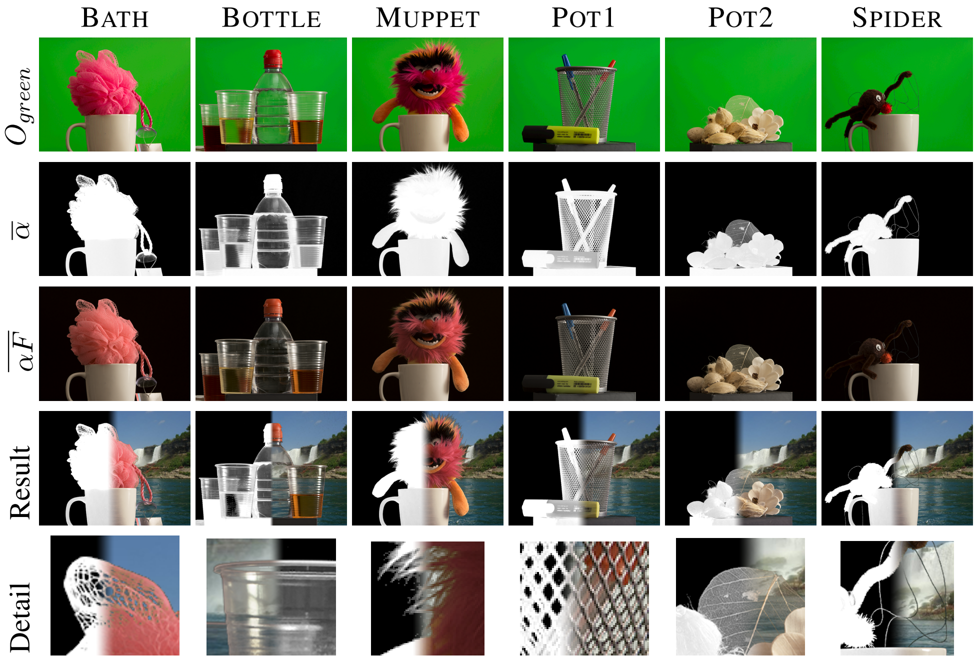

Table 3: Input (green background), ground truth dataset ($\overline{\alpha F}$, $\overline{\alpha}$) and best result with our method. Also available `www.cs.nuim.ie/research/vision/data/imvip2014/`

estimation method (e.g. LTS). We propose in a future work to refine our evaluation with more detailed criteria (examining errors in difficult areas only, sensitivity to noise), cross-validation for the choice of parameter values and provide a GPU-based implementation.

References

[Ashikhmin, 2001] Ashikhmin, M. (2001). Hight quality chroma key http://www.cs.utah.edu/ michael/chroma/. Technical report.

[Chuang et al., 2001] Chuang, Y.-Y., Curless, B., Salesin, D. H., and Szeliski, R. (2001). A bayesian approach to digital matting. In *CVPR*, volume 2, pages 264–271. IEEE Computer Society.

[Comaniciu and Meer, 2002] Comaniciu, D. and Meer, P. (2002). Mean shift: a robust approach toward feature space analysis. *PAMI*, 24(5):603–619.

[Filmmaker IQ, 2013] Filmmaker IQ (2013). Hollywood history of faking it. the evolution of greenscreen compositing http://filmmakeriq.com/lessons/hollywoods-history-of-faking-it-the-evolution-of-greenscreen-compositing/.

[He et al., 2011] He, K., Rhemann, C., Rother, C., Tang, X., and Sun, J. (2011). A global sampling method for alpha matting. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*.

[Jack, 1996] Jack, K. (1996). *Video Demystified*. Newnes, Newton, MA, USA, 5th edition.

[Mishima, 1992] Mishima, Y. (1992). A software chromakeyer using polyhedric slice. In *NICOGRAPH*.

[Rother et al., 2004] Rother, C., Kolmogorov, V., and Blake, A. (2004). "grabcut": Interactive foreground extraction using iterated graph cuts. *ACM Trans. Graph.*, 23(3):309–314.

[Schultz, 2006] Schultz, C. (2006). Digital keying methods. Technical report, University of Bremen Center for computing Technologies TZI.

[Smith and Blinn, 1996] Smith, A. R. and Blinn, J. F. (1996). Blue screen matting. In *SIGGRAPH*, pages 259–268, New York, NY, USA. ACM.

[Vlahos, 1964] Vlahos, P. (1964). Composite color photography. US Patent 3,158,477.

[Wang and Cohen, 2007] Wang, J. and Cohen, M. F. (2007). Image and video matting: A survey. *Found. Trends. Comput. Graph. Vis.*, 3(2):97–175.