Soft Segmentation of Foreground : Kernel Density Estimation and Geodesic Distances

Aditya Ramesh and S.Nagalakshmi
B.Tech, Dept of Electrical and Electronics Engineering, NITK Surathkal,Mangalore
Assoc.Prof, Dept Of Information Science and Engineering ,Dr AIT, Bangalore
aditya_2806@hotmail.com lakshmi0424@rediffmail.com

Abstract— **Segmentation of unadulterated images and videos though fundamental is a challenging problem in the area of image processing. Image segmentation is associated with the problem of localizing regions of an image relative to content (e.g., image homogeneity). Segmentation is the first essential and important step of low-level vision. The process of extracting foreground objects from either still images or from video sequences has an important role in many image and video editing applications. Accurately separating a foreground object from the background means we have establish both full and partial pixel coverage, also known as pulling a matte, or foreground matting. In this work, we present a technique to create an alpha matte guided by user scribbles. These scribbles serve as a basis to estimate the RGB distributions of the foreground/background and also for the computation a distance function to each unknown pixel in the image. The foreground and background color distributions are estimated using kernel density estimation following which local smoothness is maintained by the geodesic distance function which generates the soft segmented alpha matte. When we constrain the number of sets to be two in number (for background and foreground) and let the sets be fuzzy, the problem statement evolves to one of soft segmentation or alpha matting.**

Keywords—Segmentation, matte, alpha matte

I. Introduction

IMAGE segmentation is characterized as the problem of localizing regions of an image relative to content (e.g., image homogeneity). The segmentation of natural images and videos is one of the most fundamental and challenging problems in image processing. Segmentation is the first essential and important step of low-level vision. There are many applications of segmentation. Segmentation followed by recognition is required. Applications vary from detection of cancerous cells to the identification of an airport from remote sensing data. In all these areas, the quality of the final output depends on the quality of the segmented output [1].

Segmentation is the process of partitioning the image into non-intersecting regions such that each region is homogenous and the union of no two adjacent regions is homogenous. Formally it can be defined as follows:

Let P be the set of all pixels, segmentation is the partitioning of P into (S1, S2, S3…,Sn) such that

$$\bigcup_{i=1}^{n} SI = P \text{ and } Si \cap Sj = \varphi$$

There are three main segmentation categories: fully automatic methods, semi-automatic methods, and (almost) completely manual ones. This work deals with the semi-automatic kind. When we constrain the number of sets to be two in number (for background and foreground) and let the sets be fuzzy, the problem statement evolves to one of soft segmentation or alpha matting.

II. Alpha Matting

To extract foreground objects from images that are still or from video sequences plays an important role in some of the image and video editing applications. An accurate separation of the foreground objects from the background means to determine full and partial pixel coverage which is known as foreground matting. This problem was mathematically established by Porter and Duff in 1984 [2]. They introduced the alpha channel as the means to control the linear interpolation of foreground and background colours for anti-aliasing purposes when rendering a foreground over an arbitrary background. Mathematically, the observed image Iz (z = (x,y)) is modelled as a convex combination of a foreground image Fz and a background image Bz by using the alpha matte αz(known as the matting equation):

$$Iz = \alpha zFz + (1-\alpha z)Bz \qquad (1)$$

Matting Problem – Given an image, extracting a foreground element from a background image by estimating an opacity for each pixel of the foreground element (estimating α for all pixels)

Foreground element is extracted from a background image by estimating a colour and opacity for the foreground element at each pixel. The opacity value at each pixel is typically called its alpha, and the opacity image, taken as a whole, is referred to as the alpha matte in digital matting. Fractional opacities (between 0 and 1) are a necessity for transparency and motion, blurring of the foreground element and for the partial coverage of a background pixel around the foreground object's boundary. Matting is used in order to composite the foreground element into a new scene.

It is an inherently under constrained problem, for every pixel p only the image pixel intensity is known, Fz and Bz and αz are all unknown.

$$\begin{matrix} Ir \\ Ig \\ Ib \end{matrix} = \alpha \ * \begin{matrix} Fr \\ Fg \\ Fb \end{matrix} + (1-\alpha) * \begin{matrix} Br \\ Bg \\ Bb \end{matrix}$$

The set of equations above has 3 known parameters and 7 unknowns at every pixel. At all pixels only the RGB values of I are known. Fr, Fg, Fb, Br, Bg, Bb and α are not known and needs to be estimated.

a. Trimap

If there are no added restrictions or constraints, it is evident that the total number of valid solutions to the matting equation is infinite. To extract semantically, meaningful foreground objects, almost all matting approaches start by having the user segment the input image into three regions: definitely foreground Kf, definitely background Kb, and unknown U. This three-level pixel map is often referred to as a trimap. The matting problem is thus reduced to estimating F, B, and α for pixels in the unknown region based on known foreground and background regions.

Instead of requiring a carefully specified trimap, some proposed matting approaches allow the user to specify a few foreground and background scribbles as user input to extract a matte. This intrinsically defines a very coarse trimap by marking the majority pixels (pixels have not been touched by the user) as unknowns. Figure 1. Shows an input image and its corresponding trimap. The three level trimap represents the known foreground (white), known background (black) and the unknown region (grey).

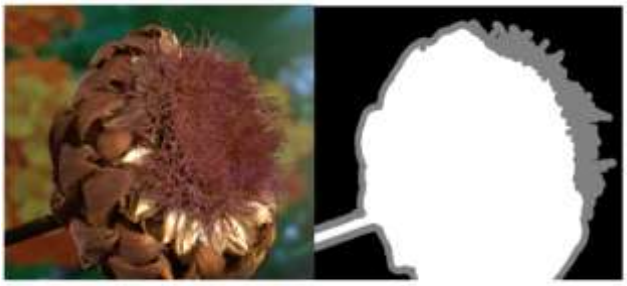

Figure 1: Input image and its trimap

The trimap is what makes alpha matting a supervised soft segmentation by utilizing user input, after the trimap is obtained it is possible to build global/local colour models. In matting, a straightforward way to use the local correlation is to sample nearby known foreground and background colours for each unknown pixel, Iz. According to the local smoothness assumption on the image statistics, it can be assumed that the colours of these samples are “close” to the true foreground and background colours (Fz and Bz) of Iz, thus these colour samples can be further processed to get a good estimation of Fz and Bz. Once Fz and Bz are determined, αz can be easily calculated from the matting equation.

b. *Binary Segmentation v. Soft Segmentation*

If we constrain the alpha values to be only 0 or 1 in equation (1), the matting problem then degrades to another classic problem: binary image/video segmentation, where each pixel fully belongs to either foreground or background.

This problem has been extensively studied since early 1960s, resulting in a large volume of related literature. Although matting is modelled as a more general problem than binary segmentation, which is theoretically harder to solve, most existing matting algorithms avoid the segmentation problem by having a trimap as another input in addition to the original image. The smoothness of the alpha matte helps capture the wispy nature of animal fur, human hair etc. which cannot be captured by binary segmentation.

III. METHODOLOGY

The proposed algorithm is illustrated in Figure 2 as a flowchart.

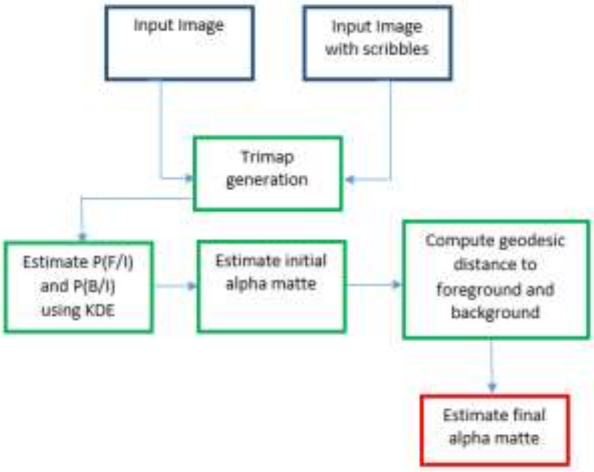
Input Image
Input Image with scribbles
Trimap generation
Estimate P(F/I) and P(B/I) using KDE
Estimate initial alpha matte
Compute geodesic distance to foreground and background
Estimate final alpha matte

Figure 2: Overview of the matting process implemented.

The data used in this work is freely available at www.alphamatting.com for benchmarking matting algorithms and comparing the performance with other available algorithms [11].

3.1 Creation of Trimap from user scribbles

One of the important factors effecting the performance of a matting algorithm is how accurate the trimap is. Ideally, the unknown region in the trimap should only cover truly mixed pixels. In other words, the unknown region around the foreground boundary should be as thinas possible to achieve the best possible matting results. However, accurately specifying a trimap requires significant amounts of user effort and is often undesirable in practice, especially for objects with large semi-transparent regions or holes.

In this work, scribbles are processed to generate a Trimap in a manner similar to GrabCut [12]. The green scribble is always closed and demarcates the boundary for the background, every pixel outside the green outline is considered to be a part of Kb (definite background).

The pixels coming under blue scribbles are taken to be a part of Kf (definite foreground) and the rest of the pixels are those which are unknown. Figure 5 shows how the scribbles are required in this algorithm.

The pixels belonging to Kb would have an α value of zero and the pixels belonging to Kf have an α = 1.

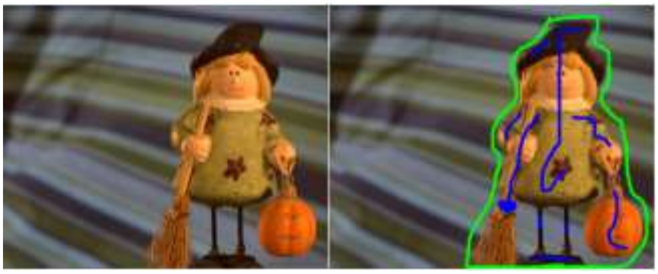

Figure 3: Given user scribbles

Figure 3. Illustrates how user scribbles are expected to be given, the green scribble completely encloses the object of interest and the blue scribble give the definite foreground information. Once the scribbles have been provided, the definite foreground pixels can be found by subtracting the B channel of RGB of the original and scribbled image.

Similarly, the definite background can be obtained by filling the area inside the green scribble and then taking an overall not operation. The trimap generated from the scribbles and original image is shown in Figure 4.

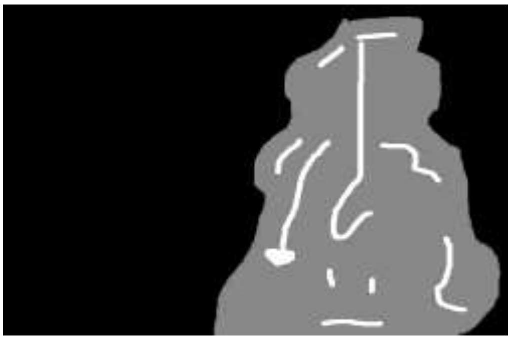

Figure 4. Trimap generated from user scribbles.

3.2 Colour Models for Foreground and Background.

The matting equation is given by –

$$Iz = \alpha z Fz + (1 - \alpha z)Bz$$

Rearranging, we get,

$$\alpha z = (Iz – Bz)/ (Fz– Bz)$$

If a reasonable estimate of Fz and Bz is possible for all each unknown pixel, then it is possible to compute each alpha value.

Another possible approach is to fit a probability distribution to the colour space of definite foreground pixels and to the colour space of definite background pixels and then evaluate the conditional probability of the pixel being a foreground element given its colour. A probability distribution needs to be fit to the RGB values of known foreground pixels to get the foreground colour distribution. A similar process needs to be performed for background pixel intensities.

Each pixel is a 3-vector of RGB values, but the R,G and B can be taken to be independent variables

$$P(I) = P(R) * P(G) * P(B)$$

The probability mass function for each channel is estimated using Kernel Density Estimation (KDE) [13]. Kernel density estimation is a non-parametric way to estimate the probability density/mass function of a random variable. KDE involves convolution of a suitable kernel with the histogram of the data.

$$f(x) = \frac{1}{nh}\sum_{j=1}^{n} K\{\frac{x-Xj}{h}\}$$

Where K is a kernel function- a non-negative function that integrates to one and has mean zero. and h > 0 is a smoothing parameter called the bandwidth. Intuitively one wants to choose h as small as the data allow, however there is always a trade-off between the bias of the estimator and its variance; more on the choice of bandwidth below.

Figure 5 shows several possible kernel functions which can be used for KDE, this work uses a Gaussian kernel.

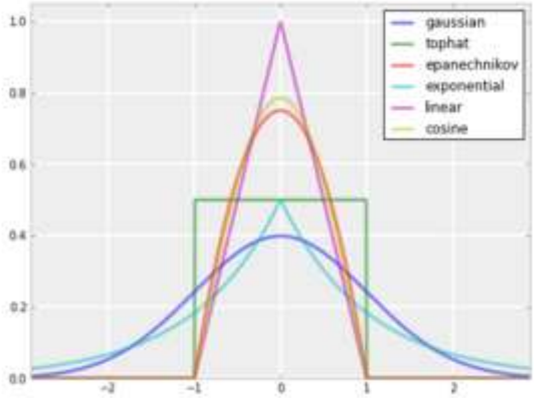

Figure 5. Range of possible kernels for KDE.

KDE can be used to estimate the probability that pixel with colour CX is a foreground pixel PF(CX) and the probability that it is a background pixel PB(CX). Once these probabilities are known it is possible to estimate an initial alpha matte which can further be refined in later steps.

$$\alpha z= PF(CX) / (PF(CX) + PB(CX))$$

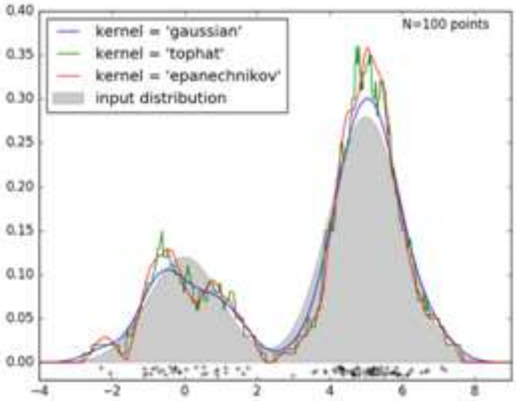

Figure 6. Kernel Density Estimation

Figure 6 shows the Kernel density estimation from samples using various kernels. Grey curve is the true distribution.

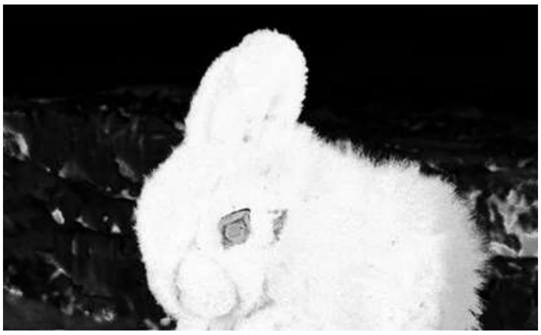

Figure 7: Belief for alpha values

Figure 7. shows the initial Belief for alpha values at each pixel only based on KDE colour models.

3.3 Geodesic Distances for accurate matting

The geodesic distance d(x) [15] is simply the smallest integral of a weight function over all possible paths from the scribbles to x (in contrast with the "average" distance as used in random walks or diffusion/Laplace based frameworks). Specifically, the weighted distance (geodesic) from each of the two scribbles for every pixel x is:

$$D_L(x) = \min d(s,x),\ L\in \{F,B\}\ s \in \Omega_L$$

and $$d(s1,s2) = \min \int_{s1}^{s2} |W(x).C_{s1,s2}(x)|\, dx$$

where $C_{s1,s2}(x)$ is a path connecting the pixels s1, s2

For each unknown pixel we find the shortest weighted path to any foreground scribble and any background pixel. The weights W here selected as the gradient of the likelihood that a pixel belongs to the foreground. That is, the gradient of the initial alpha belief obtained in the previous step. Note how in this case we are exploiting from the user-provided scribbles both their actual position and the statistics of the pixel colours marked by these scribbles.

The discrete geodesic distance can now be approximated as the minimum sum of W values along a path connecting s1 and s2.

$$d(s1,s2) = \min \sum_{x,y} W_{xy}$$

and $W_{xy} = |\ P^F(C_X) - P^F(C_Y)|\ x,y\in C_{s1,s2}$

Based on this concept of geodesic distances, a pixel is close in this metric to a scribble in the sense that there exists a path along which the likelihood function does not change much.

We can efficiently compute the distances, in optimal linear time [14]. It involves creation of a region adjacency graph where each pixel is assumed to have a 4-neighbour connection. Figure 10. Shows the visualization of a graph where edges have weights from neighbourhood.

The matrix Wxy can be estimated by taking the gradient of the PF image. The gradient can be taken using one of several edge operators, canny, Laplcacian, Sobel, Prewitt etc.

IV. Results

4.1 Estimating final alpha matte

We now combine the DL(x) (geodesic distance) with the initial probability of foreground estimate to obtain the alpha matte.

We proceed as follows :

Step 1: compute $\omega_L(x) = D_L(x)^{-r} . P^L(x)$ $L \in \{F,B\}$

Step 2: $\alpha(x) = \omega_F(x) / (\omega_F(x)+ \omega_B(x))$
When $r = 0$, $\alpha(x) = PF(x)$, when $r \rightarrow\infty$, $\alpha(x)$ becomes hard segmentation (typically $0 \leq r \leq 2$ in our case).

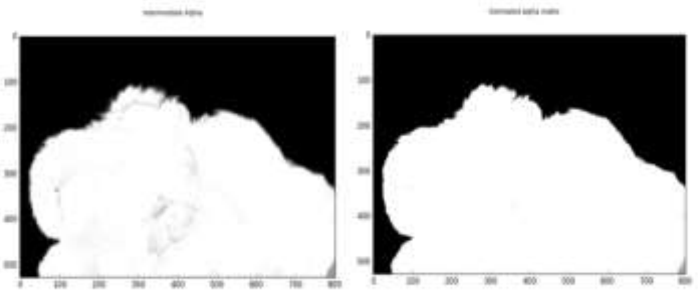

Figure 8. Shows conversion of DL(x) to α(x).

In Figure 9 the Results are shown. The form is of an image montage of the original image, scribbled image, estimated alpha and a composite image.

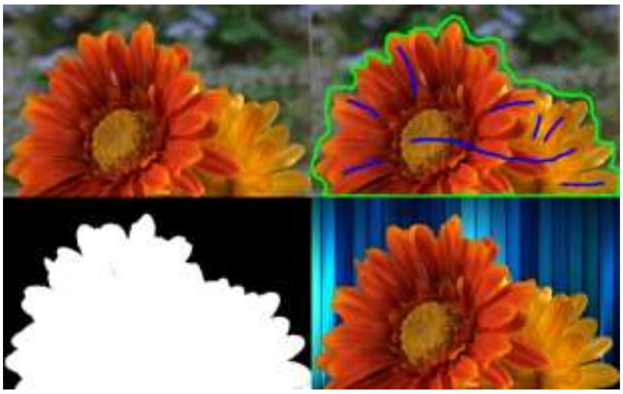

Figure 9: Resultant images

V. Conclusion

An algorithm was presented which could generate alpha mattes for images with complex backgrounds and foregrounds with minimal user input in the form of scribbles.

Although the proposed framework is general, it mainly exploited weights in the geodesic computation that depend on the pixel value distributions. As such, in this form the algorithm works best when these distributions do not significantly overlap. In principle, this can be solved with enough user interactions, but could be tedious, and would be better to solve this by enhancing the features used in deriving the weights. Efforts could be made in enhancing the features currently used for weighting the geodesic.

REFERENCES

[1] N. R. Pal and S. K. Pal. A review on image segmentation techniques. Pattern Recognition. Vol 26. No. 9, pp. 1277-1294, 1993.

[2] T. Porter and T. Duff. Compositing digital images. Proceedings of ACM SIGGRAPH, pp. 253–259, July 1984.

[3] A. Berman, P. Vlahos, and A. Dadourian, Comprehensive method for removing from an image the background surroundinga selected object. US Patent no. 6,135,345, 2000.

[4] Y. Chuang, B. Curless, D. Salesin, and R. Szeliski. A Bayesian approach to digital matting. In Proc. CVPR, 2001

[5] M. Ruzon and C. Tomasi. Alpha estimation in natural images. In Proc. CVPR, 2000

[6] J. Sun, J. Jia, C.-K. Tang, and H.-Y. Shum. Poisson matting. ACM Trans. Graph., 23(3):315–321, 2004

[7] F.Wang, J.Wang, C. Zhang, and H. C. Shen. Semi-supervised classification using linear neighborhood propagation. In Proc. IEEE CVPR,New York, 2006, pp. 160–167

[8] S. Roweis and L. Saul. Nonlinear dimensionality reduction by locally linear embedding. Science, vol. 290, pp. 2323–2326, 2000.

[9] R. Zass and A. Shashua. A unifying approach to hard and probabilistic clustering. Int. Conf. Computer Vision, Beijing, China, Oct. 2005

[10] L. Grady and G. Funka-Lea. Multi-label image segmentation for medical applications based on graph-theoretic electrical potentials. In Proc. Computer Vision and Mathematical Methods in Medical and Biomedical Image Analysis Workshop, 2004, pp. 230–245

[11] ChristophRhemann, Carsten Rother, Jue Wang, MargritGelautz, PushmeetKohli, Pamela Rott. A Perceptually Motivated Online Benchmark for Image Matting. Conference on Computer Vision and Pattern Recognition (CVPR), June 2009.

[12] Rother C., Kolmogorov V., & Blake A. (2004). Grabcut: Interactive foreground extraction using iterated graph cuts. In SIGGRAPH'04.

[13] B. W. Silverman. Kernel Density Estimation Using the Fast Fourier Transform. Journal of the Royal Statistical Society. Series C (Applied Statistics) Vol. 31, No. 1 (1982), pp. 93-99

[14] Yatziv, L., Bartesaghi, A., &Sapiro, G.O(n) implementation of the fast marching algorithm. Journal of Computational Physics,212, 393–399.

[15] XueBai, Guillermo Sapiro. Geodesic Matting: A Framework for Fast Interactive Image and Video Segmentation and Matting. Intl. Journal of Computer Vision. 2006