Semi-automated Binary Segmentation with 3D MRFs

Patel Akshkumar Rajesh
Trinity College Dublin

***Abstract*—This research provides Bayes theorem for tackling the matting issue, which involves extracting a foreground piece from a background picture by predicting an opacity for each foreground and background pixel. To achieve the final display, we implemented Maximum a posteriori (MAP) estimation in which maximum-likelihood criterion as likelihood function and Markov random field as prior probability function to calculate the ideal foreground, and background outcomes all at once.**

***Index Terms*—Image Matting, Markov random field, Motion Estimation Maximum likelihood estimation,Maximum a posteriori estimation, Segmentation**

I. Introduction

Image compositing is commonly used in the image and video editing process. Image compositing is the process of combining areas from many images to create a new image. Background is a frequent use case. Professionals often undertake many editing stages such as segmentation, matting, and foreground color decontamination manually to generate high-quality composites, which takes a long time even with advanced picture editing software. In the early days of film and photography, compositing was done by manually cutting and pasting photographs or video prints together.The over operation [1] is the most common compositing operation, and it is summed up by the compositing equation:

$$C = \alpha F + (1 - \alpha)B \tag{1}$$

Where C, F, and B are the pixel's composite, foreground, and background colours, respectively, and alpha is the pixel's opacity component used to linearly blend between foreground and background. Here, composting technique is used for colour keying in video.The opacity image as a whole is referred to as the alpha matte or key. Color key compositing, often known as chroma keying, is a visual-effects and post-production method that combines (layers) two pictures or video streams depending on chroma range. The method has been used to eliminate a backdrop from the subject of a photo or video in numerous disciplines, most notably in the post production, motion picture, and video game industries. In color keying process, binary matting also called as digital matting is used to extract foreground and background from the image matting or video matting. Compositing techniques are a relatively simple way of pulling a matte - the foreground from a green screen scene could be imposed on an arbitrary background scene. Bayesian matting is one of the approaches for digital matting.The Bayesian matting technique [1] models both the foreground and background colour distributions with spatially varying sets of Gaussian distributions, and the final output is a fractional blending of the foreground and background colours. It then uses a maximum-likelihood criteria to calculate the alpha, foreground, and background together at the same time. From the compositing equation, the value of foreground and background for unknown pixel is determined. Another approach to find foreground and background [2] is using Pairwise. The Markov random field (MRF) model is a relatively simple, yet effective, tool to encompass prior knowledge in the segmentation process, and in fact the interest on MRFs has been steadily growing in recent years. When image segmentation is formulated as a Bayesian, and more specifically a maximum a posterior probability (MAP) estimation problem, all prior information available on the image to be segmented.

II. VIDEO SEGMENTATION

Video segmentation/color keying is all about discovering alpha(x) based primarily on colour alone.The implicit algorithm here is that we are making decisions on a pixel basis. For the pixel I(x) we have to choose value of alpha. For binary matting x is the position of the pixel with value I(x). Let alpha be the binary matte value at the site. Foreground is indicated by alpha = 1 and the background is indicated by alpha = 0.

$$p(\alpha = 1|I) > p(\alpha = 0|I)\alpha = 1 \tag{2}$$

$$p(\alpha = 0|I) > p(\alpha = 1|I)\alpha = 0 \tag{3}$$

For given pixel of the image (I) we find alpha using Bayes theorem which is represent as equation 4:

$$P(\alpha(x)|I(x)) = \frac{p(I(x)|\alpha(x))p(\alpha(x))}{p(I(\cdot)} \tag{4}$$

According to the Bayes theorem, p(I(x)—alpha (x)) is likelihood function and p(alpha (x)) is prior probability distribution. We have to find the likelihood and prior to find the whether the pixel is foreground or background.

A. Maximum likelihood estimation

The goal of this algorithm is to find probability of pixel I(x) when it is foreground and probability of the pixel when it is background the pixel. Gaussian distribution helps to find the probability distribution of foreground ad background. For the gray-scale image the Gaussian distribution is given below.

$$pf(I|\alpha = 1) = \frac{1}{\sqrt{2\Pi\sigma^2}} exp - \left[\left(I - \overline{F}\right)^2 /2\sigma^2 \right] \tag{5}$$

For colour, the image pixel is actually 3 channel component vector R,G,B which is represent as vector I(x) = [Ir , Ig, Ib]. The Gaussian distribution for the 3 channels of foreground is:

$$= \frac{1}{\sqrt{2\Pi\sigma^2}} exp - \left[\frac{(Ir - \overline{Fr})^2}{2\sigma^2} + \frac{(Ig - \overline{Fg})^2}{2\sigma^2} + \frac{(Ib - \overline{Fb})^2}{2\sigma^2} \right] \tag{6}$$

Here F represents average color of foreground and B is for background. In video processing, there are co-relation between channels in RGB color space.So,YUV color space is used instead of RGB channel. The equation for YUV color space is:

$$= \frac{1}{\sqrt{2\Pi\sigma^2}} exp - \left[\frac{(Iy - \overline{Fy})^2}{2\sigma^2} + \frac{(Iu - \overline{Fu})^2}{2\sigma^2} + \frac{(Iv - \overline{Fv})^2}{2\sigma^2} \right] \tag{7}$$

For background pixels the Gaussian distribution of background for 3 channels is given as:

$$= \frac{1}{\sqrt{2\Pi\sigma^2}} exp - \left[\frac{(Iy - \overline{By})^2}{2\sigma^2} + \frac{(Iu - \overline{Bu})^2}{2\sigma^2} + \frac{(Iv - \overline{Bv})^2}{2\sigma^2} \right] \tag{8}$$

Taking a patch in the background we can find the mean value of the background pixel and variance for calculate equation 8.

A threshold energy E(k) is establish to detect whether a pixel is in the foreground or background.Foreground pixels are not consider to inspect because they all are different colour pixels. For matting, the pixel value in the background is the same as the pixel value in the foreground. As a consequence, we determined the value of the background threshold. If the background energy exceeds the threshold, the pixel is classified as background; otherwise, it is classified as foreground.

$$= exp - \left[\frac{(Ir - \overline{Br})^2}{2\sigma^2} + \frac{(Ig - \overline{Bg})^2}{2\sigma^2} + \frac{(Ib - \overline{Bb})^2}{2\sigma^2} \right] > k \tag{9}$$

Taking exponent is unstable in computer so, instead of exponent the logarithmic value is taken on the both site. So, the new equation is:

$$= \left[\frac{(Ir - \overline{Br})^2}{2\sigma^2} + \frac{(Ig - \overline{Bg})^2}{2\sigma^2} + \frac{(Ib - \overline{Bb})^2}{2\sigma^2} \right] < ln(k) \tag{10}$$

The above equation is similar to euclidean distance between pixel value and the Gaussian distribution of three channels. If the above given condition is matched then pixel must be background otherwise foreground. These shows maximum likelihood estimation for pixel.

B. Markov Random Field

Obtaining the prior distribution is one of the challenges of implementing Bayesian estimation. In other words, before using the prior probability density function to determine the a posteriori distribution, it must be estimated. In the framework of Bayesian probability theory, Markov random field theory holds the potential of giving a systematic method to picture analysis. Markov random fields (MRFs) are [2] used to simulate the statistical features of pictures.MRF describes the statistical dependence of the process at a pixel in an image on the neighboring pixels.As a result, it is considered to describe the spatial interaction of the pixels in the image. MRFs are characterized by the fact that label interaction is restricted to a local region. This area is known as the pixels' neighborhood. The conditional distribution of a pixel expresses the likelihood of probable labels at that pixel based on the labels at neighboring pixels. Because of the relatively tight consistency constraints, it is impossible to specify a Markov random field by its conditional probability structure . Fortunately, Gibbs distributions allow you to specify a Markov random field.Gibbs energy function is a widely used for computing maximum a posterior (MAP) estimation.

$$pf(\alpha(x)|N(x)) = \frac{1}{z} exp - \Lambda \left[\sum_{k=1}^{4} \lambda \left| \alpha(x) \neq \alpha(x + qk) \right| \right] \tag{11}$$

In the equation 11, For every pixel site x we measure the likelihood and energy of Es(0) and Es(1).

$$Es(0) = \sum_{k=1}^{4} V(0, (\alpha(x + qk)) \tag{12}$$

$$Es(1) = \sum_{k=1}^{4} V(1, (\alpha(x + qk)) \tag{13}$$

In equation 12 and 13, for the energy Es the V represents the lambda which represents smoothness weight, qk represents cliques and alpha which is 0 or 1 depends on the neighbourhood pixel values.

C. Maximum a posteriori (MAP) estimation

The a posteriori distribution is a combination of the probability and a priori distributions. The probability distribution connects the observed data to the optimal solution. The a priori distribution includes prior knowledge about probable solutions. This priori distribution is represented by a Gibbs distribution or a Markov random field and probability distribution is represented by the Maximum likelihood estimation.

$$E(0) = ln(z) + [((I - I_\alpha)^T) R_\alpha^{-1} (I - I_\alpha)] + \sum_{k=1}^{4} V(0, (\alpha(x + qk)) \tag{14}$$

$$E(1) = ln(z) + E(t) + \sum_{k=1}^{4} V(0, (\alpha(x + qk)) \tag{15}$$

Here, the above given both equation is considered as the Energy function. Is the E(0) is less than E(1).

III. OPTIMIZATION

This given algorithm will describe the steps of implementing Maximum a posteriori.

Algorithm 1 Maximum a posteriori (MAP)

```
Take one patch in the background
Measure Gaussian parameters for background
Get mean and variance of background
Set threshold Et for alpha = 1 at a pixel
for iteration = 1 : N do
    Measure El, Es(0) and Es(1)
    Set $E(0) = El + Es(0)$
    and $E(1) = El + Es(1)$
    if E(0) Less Than E(1) then
        alpha = 0
    else
        alpha =1
```

In the above given algorithm El is the energy function of maximum likelihood which calculated by equation 8, Es(0) and Es(1) is the energy function of Markov random field which is given in the equation 12 and 13 respectively.

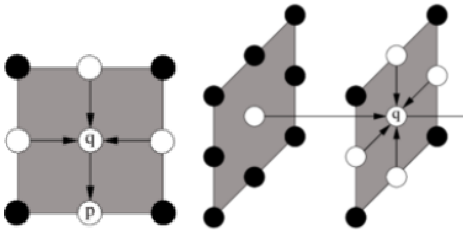

Figure 1: Neighborhood pixels in 2D Markov random field and 3D Markov Random field

There are 2 different approaches for Markov random field estimation. One is the 2D-Markov random field and second is the 3D-Markov random field. The neighborhood pixel is the key difference between 2D and 3D segmentation. We commonly employ the 4-neighborhood pixel or the 8-neighborhood pixels from the current frame or the image. In, 3D-Markov random filed the current frame neighbourhood pixels are depending on the previous frame neighbourhood pixels. Figure 1 depicts the difference between 2D-MRF approach vs 3D-MRF approach.Here, 3D-Markov random field the main pixel is depends on previous frame neighbourhood pixel.As frame changes to one from another we can see exploiting temporal redundancy.To get more smooth output we used motion compensation. Motion compensation is an algorithmic approach that is used to anticipate a frame in a video given previous or next frames with accounting for the motion of the objects in the video. For motion compensation we need input as an image vector which will get by the motion estimation. Motion estimation is the process of estimating motion vectors that characterize the transformation from one 2D image to another, often from nearby frames in a video sequence. Using motion estimation, we get backward frames that will be utilized as inputs for motion compensation, and the output of motion compensation will be used as the previous frame for 3D-MRF.

IV. IMPLEMENTATION

This section describes how we implemented Maximum likelihood estimation, 2D-Markov random field, 3D-Markov random field with and without motion estimation using NUKE software. We begin by changing the color space to linear in order to perform maximum likelihood iteration. Then, using a prepackaged tool, we determined the mean values of the background. We constructed an expression node based on the concept of equation (8) that would return the energy value of maximum likelihood. Another node was built to provide a threshold value. We built a blink script for the 2D-Markov random field, which is attached in the appendix for a reference. In the 2D-MRF blink script, we used 8 neighborhood pixels to calculate prior probability, as shown in equation (11). We wrote a blink-script for the 3D Markov random field, which you can see in the appendix. A previous frame neighbourhood and a threshold value will be used as input to the 3D-Markov random field without motion estimation.In, 3D-MRF blink script we have used only one main pixel from the previous frame.Motion vector generator node is used in 3D-Markov Random field with motion.For motion estimation, we used a vector generator node. The motion vector generator's output will be sent to the IDistort node, which conducted the motion compensation IDistort utilized to calculate the distortion for the inputs motion and an image. To refer the previous frame we used the time-offset node with offset to the previous frame. Also, we use filter node for spatial smoothness.

V. DATA

This section will describe the input frames and ground-truth. We have 5 different input frames and 3 ground-truth which is cut manually by hands.The motion is present between the two frames. Using motion estimation we generated motion vector using MATLAB which is shown in the figure 3.

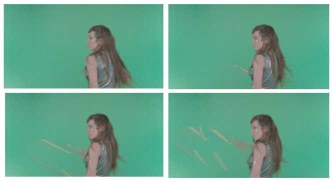

Figure 2: Input frames

VI. RESULT

This section describes about the results of maximum likelihood, 2D-Markov random field, 3D-Markov random field and their comparison.There are 3 subsection for each algorithm.

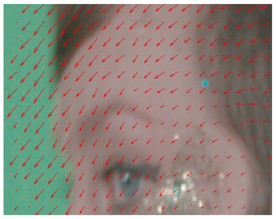

Figure 3: motion estimation vector

A. *Maximum likelihood estimation*

As we can see in the result, some error pixels are in the background and others are in the foreground. When we compare our maximum likelihood image to the original frame, we can identify certain inaccuracies. We can also see the green background on the foreground's edges.

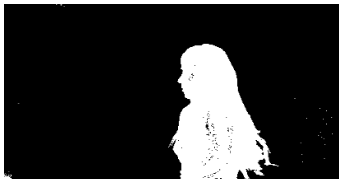

Figure 4: Output of maximum likelihood estimation

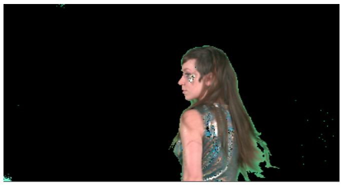

Figure 5: Output of maximum likelihood estimation

B. *2D-Markov random field*

In, output of 2D-Markov random field there is no background pixels in the foreground region. Also, we can see in the figure that as we increase the number of iteration of 2D-Markov random field the foreground is getting better and output is getting near to the ground truth. Also, we can see in the figure 5 we are getting less green background edges in the foreground compare to the maximum likelihood estimation.

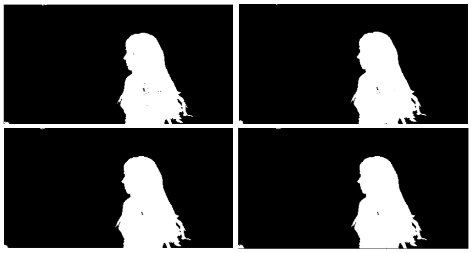

Figure 6: Output of 2D-Markov Random Field on different iteration

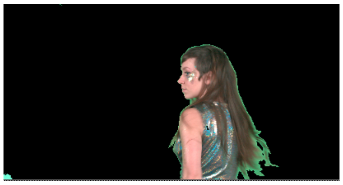

Figure 7: Output of 2D-Markov Random Field

C. *3D-Markov random field*

In 3D-Markov random field we implemented one with using motion estimation and another without using motion estimation. The figure 9 shown the output images of 3D-MRF without motion estimation.In figure 8, on increasing the number of iteration,the output gets smoother. On the 5th iteration there are no black pixels in the foreground but we can see more green background edges compare to 2D- Markov random field and maximum likelihood iteration.

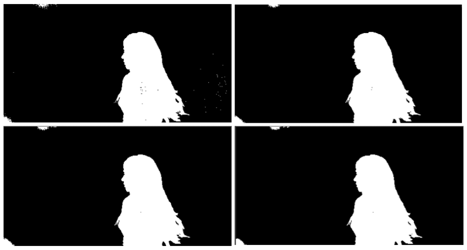

Figure 8: Output of 3D-Markov Random Field on different iteration

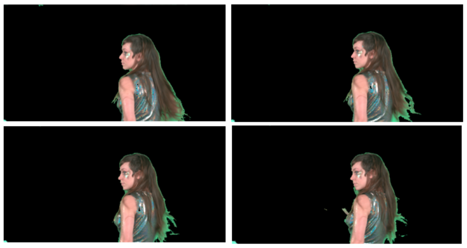

Figure 9: Output of 3D-Markov Random Field without motion estimation on different frames

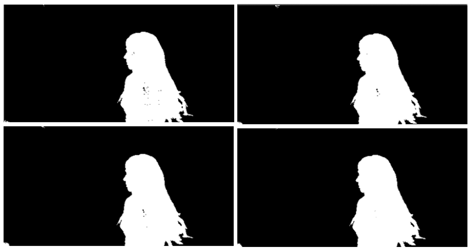

Figure 10: Output of 3D-Markov Random Field with motion estimation on diffrent iteration

When we examine the output of these four distinct algorithms, we can detect a significant variation on the edges. In

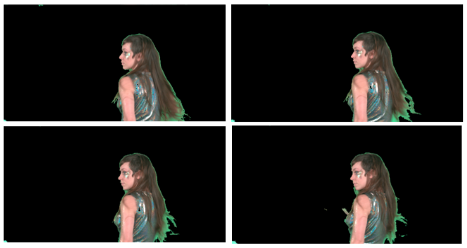

Figure 11: Output of 3D-Markov Random Field with motion estimation in different frames

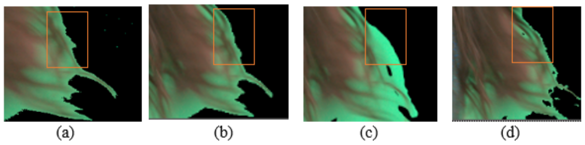

Figure 12: Comparison of edges in different algorithm (a) Maximum likelihood estimation (b) 2D-Markov Random field (c) 3D-Markov Random field (d) 3D-Markov Random field with motion estimation

the figure 12, we can observe that in 3D-MRF without motion estimation, the edges are smoother than in the other three techniques. In addition, we can see in the orange kernel that some of the parts are being cut during maximum likelihood iteration.

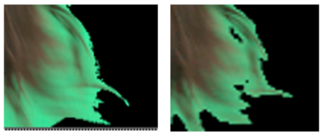

Figure 13: Difference between different approach of neighbourhood pixels from previous frames in 3D-MRF

The output of 3D-MRF is also affected by the pixels of the previous frame. We tried two approaches: one using eight neighborhood pixels from the previous frame and one with only one center pixel from the previous frame. As shown in the figure 13, the second approach delivers better results.

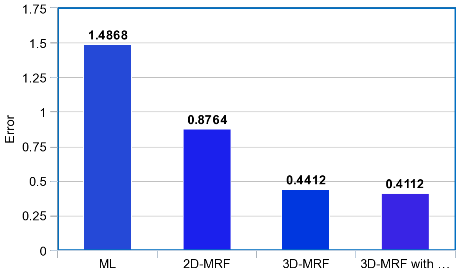

Figure 14: Mean Absolute Error

Using MATLAB we calculated Mean absolute error and PSNR of output with respect to ground truth.As we can see in the graph 14 Mean absolute error is low in 3D-MRF with motion compare to 2D-MRF and Maximum likelihood estimation.

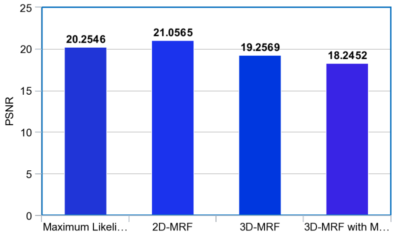

Figure 15: Peak Signal to Noise Ratio

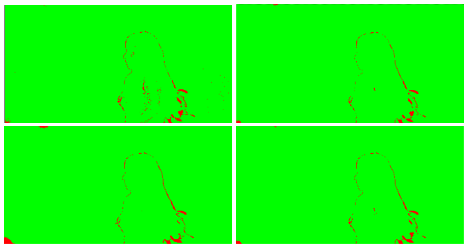

Figure 16: Difference between groundtruth and output of (a) Maximum likelihood estimation (b) 2D-Markov Random field (c) 3D-Markov Random field (d) 3D-Markov Random field with motion estimation

We visually compare the ground-truth to the output of 2D-MRF, 3D-MRF, and 3D-MRF with motion. On, figure 16 reveals the red pixels as the difference between the output and the ground truth. There are no red pixels in the foreground region of the 2D-MRF, 3D-MRF, or 3D-MRF with motion. As a result of the provided ground truth being manually cropped, we can see some red pixels on the foreground's boundaries.

Conclusion

The 3D Markov random field output is depends on the choosing the number of neighborhood pixels from the previous frame. Tests for a variety of conditions the 3D Markov Random Field showed that it performed better than 2D Markov Random Field and Maximum Likelihood Estimation for given data. Further work is planned with try different matting approaches to find solution for edge pixels of the foreground and deep learning techniques to find best neighborhood pixels values and threshold values.The limitation of 3D-MRF is that some background pixels can be seen on the foreground's edges. Also, limitation in manually cut ground truth we can use different approaches to get accurate ground truth and find more accurate comparison.

References

[1] H. Zhang, J. Zhang, F. Perazzi, Z. Lin and V. M. Patel, ”Deep Image Compositing,” 2021 IEEE Winter Conference on Applications of Computer Vision (WACV), 2021, pp. 365-374, doi: 10.1109/WACV48630.2021.00041.

[2] Li, S.Z., 2009. Markov random field modeling in image analysis. Springer Science Business Media.

[3] Zoltan Kato, Markov Random Fields in Image Segmentation , now, 2012.