ISSN(Online): 2319-8753
ISSN (Print): 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor & UGC Approved Journal)

Website: www.ijirset.com

Vol. 6, Issue 8, August 2017

Multi-view Object Extraction by using the Fractional Boundaries

Monica Srilakshmi, Leela Santhoshi

M.Tech, Department of ECE, Sri Padmavathi Mahila Visva Vidyalayam, Tirupathi, Andhra Pradesh, India [1]

Professor, Department of ECE, Sri Padmavathi Mahila Visva Vidyalayam, Tirupathi, Andhra Pradesh, India [2]

Abstract: This paper presents an automatic method to extract a multi-view object in a natural environment. We assume that the target object is bounded by the convex volume of interest defined by the overlapping space of camera viewing frustums. There are two key contributions of our approach. First, we present an automatic method to identify a target object across different images for multi-view binary co-segmentation. The extracted target object shares the same geometric representation in space with a distinctive color and texture model from the background. Second, we present an algorithm to detect color ambiguous regions along the object boundary for matting refinement. Our matting region detection algorithm is based on information theory, which measures the Kullback-Leibler (KL) divergence of local color distribution of different pixel-bands. The local pixel-band with the largest entropy is selected for matte refinement, subject to the multi-view consistent constraint. Our results are high quality alpha mattes consistent across all different viewpoints.

I. INTRODUCTION

In this paper, we present some reliable solutions based on the previous approaches which utilizes the features of bounding-volume prior from camera poses, appearance models under geometric constraints and iterative Markov random field optimization. So, we present a multi-view matte estimation method on the top of previous approaches which not only estimates the binary masks but also soft alpha mattes of a foreground object. In contrast to previous paper of the two phase approach for multi-view object extraction, in this we propose a method to estimate the unknown regions more accurately and also improve the quality estimated alpha mattes by providing color and texture information along with the geometric constraints for more accurate foreground estimation and providing a dynamic approach for estimating unknown regions. Here, we evaluate our algorithm to claim the advantages of automatically estimating a tight bound of foreground object by the convex hull of visible SFM points.

In addition it is based on simple geometric representation that defines 3D reference points in space and links between super pixels in images. Without requiring the 3D structure refinement stage, our approach samples regular grids of the initial convex hull and only keeps physically meaningful surface samples through the visibility computation at every iteration. Secondly our appearance model of the foreground object is more robust because it can seek the texture patterns of an object so that our MRF optimization overcomes possible local minimum. It can also handle high resolutions images with minimum user interventions because the optimization and matting procedure only performed on uncertainty regions of the trimaps.

The outlook for the process is given below: Initially the user defines which is the foreground and background regions assuming the target object is bounded by the convex hull of camera viewing frustums and develop an initial contour for the foreground region. Based on the initial contour, estimate the appearance models of foreground regions which is comprised of color and texture model. We define MRF optimization for getting binary segmentations and refine those contours so that they are accurately located at the object boundaries. To evaluate the color mixing of the foreground and background regions along with the unknown regions we utilize the process of evaluating the distribution of colors within the local pixel bands. Lastly by using the matting laplacian we solve for the fractional boundaries of the target foreground object. The local pixel band with largest entropy is selected for matting refinement, subject to multi-view consistent constraint.

ISSN(Online): 2319-8753
ISSN (Print): 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor & UGC Approved Journal)

Website: www.ijirset.com

Vol. 6, Issue 8, August 2017

II. LITERATURE SURVEY

[1] A. Laurentini: Many algorithms for both identifying and reconstructing a 3-D object are based on the 2-D silhouettes of the object. In general, identifying a non convex object using a silhouette-based approach implies neglecting some features of its surface as identification clues. The same features cannot be reconstructed by volume intersection techniques using multiple silhouettes of the object. This paper addresses the problem of finding which parts of a non convex object are relevant for silhouette-based image understanding. For this purpose, the geometric concept of visual hull of a 3-D object is introduced. This is the closest approximation of object S that can be obtained with the volume intersection approach; it is the maximal object silhouette-equivalent to S, i.e., which can be substituted for S without affecting any silhouette. Only the parts of the surface of S that also lie on the surface of the visual hull can be reconstructed or identified using silhouette-based algorithms. The visual hull depends not only on the object but also on the region allowed to the viewpoint. Two main viewing regions result in the external and internal visual hull. In the former case the viewing region is related to the convex hull of S, in the latter it is bounded by S. The internal visual hull also admits an interpretation not related to silhouettes. Algorithms for computing visual hulls are presented and their complexity analyzed. In general, the visual hull of a 3-D planar face object turns out to be bounded by planar and curved patches.

[2] K. Kutulakos and S.M. Seitz: In this paper we consider the problem of computing the 3D shape of an unknown, arbitrarily-shaped scene from multiple photographs taken at known but arbitrarily-distributed viewpoints. By studying the equivalence class of all 3D shapes that reproduce the input photographs, we prove the existence of a special member of this class, the photo hull, that (1) can be computed directly from photographs of the scene, and (2) subsumes all other members of this class. We then give a provably-correct algorithm, called Space Carving, for computing this shape and present experimental results on complex real-world scenes. The approach is designed to (1) capture photorealistic shapes that accurately model scene appearance from a wide range of viewpoints, and (2) account for the complex interactions between occlusion, parallax, shading, and their view-dependent effects on scene-appearance.

[3] G. Vogiatzis, P.H.S. Torr and R. Cipolla: This paper presents a volumetric formulation for the multi-view stereo problem which is amenable to a computationally tractable global optimization using Graph-cuts. Our approach is to seek the optimal partitioning of 3D space into two regions labeled as "object" and "empty" under a cost functional consisting of the following two terms: 1) A term that forces the boundary between the two regions to pass through photo-consistent locations; and 2) a ballooning term that inflates the "object" region. To take account of the effect of occlusion on the first term, we use an occlusion robust photo-consistency metric based on normalized cross correlation, which does not assume any geometric knowledge about the reconstructed object. The globally optimal 3D partitioning can be obtained as the minimum cut solution of a weighted graph.

III. PROPOSED METHOD

I. MULTI-VIEW OBJECT SEGMENTATION

In the first phase, our goal is to estimate binary masks X = {X1, X2. . . XN} of the target object in multi-view images, I = {I1, I2, . . . , IN}, where N denotes the number of input images. We use superscripts to represent image indexes and subscripts to represent pixel or super pixel indexes with capital symbols. We denote S = {S1, S2, . . . , SN} as superpixel sets and M = {M 1, M 2, . . . , M N} as the initial masks of the bounding volume. We utilize the iterative graphcut optimization to achieve our goal.

A. MRF formulation

The binary segmentation in the first phase is formulated as a single energy function in the MRF frame work. Our objective function consists of the data term Ed and the neighborhood term En as follows. The data term is designed based on the appearance models Ea and the geometric model E g. Similarly, the MRF allows the regularization terms, such as E nc for similar colors, textures and Eng for geometric linkages across viewpoints.

$$E_d = \rho . E_a + (1 - \rho) . E_g \qquad (1)$$
$$E_n = \lambda_{nc} . E_{nc} + \lambda_{ng} . E_{ng}$$

ISSN(Online): 2319-8753
ISSN (Print): 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor & UGC Approved Journal)

Website: www.ijirset.com

Vol. 6, Issue 8, August 2017

The parameter ρ determines which data term is more reliable for the energy assigning of a node:

$$\rho_k = \frac{|Pr_c(I_k|g_f) - Pr_c(I_k|g_b)|}{|Pr_c(I_k|g_f) + Pr_c(I_k|g_f)|} \qquad (2)$$

Where $Pr_c(I_k|g_f)$ is the probability of the pixel I_k having the foreground labelg_f. We use subscript c on the probability Pr because it is a color-based probability term. When the colors of a pixel or a super pixel have similar metrics for both the foreground and background models, we give more weight to the geometric consistency term. The neighborhood terms are weighted by λ ng for geometrically linking nodes across the related viewpoints and λ nc for considering color and texture linkages in each image. Detailed explanation of the terms in Eq. 1 and 2 are given in the next subsections.

B. Geometric representation

In our approach, the geometric coherence of the binary segmentations is evaluated for every MRF iteration. In this procedure, a foreground label in one image becomes the true foreground when a warped pixel position (correspondence point using the camera projection matrix) in the other images also belongs to the foreground regions. For this idea, we define coherency score $v_k^n$for pixel k. For super pixels, all the pixels in a super pixel share the same score. The coherency score for the superpixels is denoted as v_k^n. The scores are defined as the sum of binary labels x or X:

$$v_k^n = \frac{1}{N}\sum_{n'=1}^{N} x^{n'}_{w_k^{n\to n'}}, \qquad v_k^n = \frac{1}{N}\sum_{n'=1}^{N} X^{n'}_{w_k^{n\to n'}} \qquad (3)$$

In Eq. 3, the warping from image n to n for pixel k is denoted as wn→n k . The coherency score is normalized to have a [0, 1] range. For the score evaluation, we define anchor points to connect superpixels across input images. The anchor points are uniformly sampled 3D points in the bounding volume. We perform a visibility check for the anchor points as illustrated in Fig. 4 to remove occluded samples.

The sampling rate is chosen such that one superpixel observes at least one 3D anchor in the 3D space. Thus, the geometric consistency can be guaranteed by enforcing the same binary label assigned to superpixels which observe the same anchor point. In our experiment, this approximate representation demonstrates good results to impose a penalty for consistent foreground mask estimation.

We model the geometric energy term of background using a sigmoid function:

$$E_g(X_k = B) = \frac{en_b}{1 + \exp[-\lambda_v(V_k - V_{th})]} \qquad (4)$$

The parameters of this function include a trade-off relationship between the recall and precision rates of predicting segmentations. In practice, we empirically find the parameter values, where the parameter λv = 20 controls the shape of energy functions, andV_{th} is usually set to 0.9 for the tolerance of thin foreground segments with possible calibration errors. Each energy connected to the background is bounded en_b= 10 as its maximum value. Similarly, the geometric energy for the foreground, $E_g(X_k = F)$, is defined as $en_b - E_g(X_k = B)$

To consider the neighborhood terms, we construct inter-view links between superpixel nodes according to the 3D anchors. Our energy term for Eng is defined as follows.

$$E_{ng}(X_p^i, X_q^j) = \sum_{i,j,p,q} |X_p^i - X_q^j| \,.\, en_b \qquad (5)$$

The neighboring term in the constructed graph is solved by considering all superpixels related to one another as sharing the common geometric model.

In comparison to SfM, our system does not refine 3D structures; rather, it concentrates on the 2D segmentation task. Even if the initial reconstruction is roughly given, we are only interested in the tight bounding volume and sparse, accurate anchor positions in it. We do not additionally compute the photo-consistency of these points as seeking a

ISSN(Online): 2319-8753
ISSN (Print): 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor & UGC Approved Journal)

Website: www.ijirset.com

Vol. 6, Issue 8, August 2017

geometrically simple representation. For some missing foregrounds in several images, these connections are also extremely useful for propagating user strokes when they are available.

C. Appearance models

Our appearance models capture colors and texture patterns in superpixels. The appearance energy terms consist of the Gaussian mixture model (GMM) of per-pixel distributions Ec and Fisher kernel (FK) representations of each superpixel E_t:

$$E_a = E_c + \lambda_t . E_t \qquad (6)$$

We build color GMMs for each image and classify the foreground, background texture patterns collected from all images. By controlling the weight λt for the texture term over the color term, we can control the influence of their cooperative effects.

In modeling the color term, we take the negative log of a probability, which converts an arg max MAP problem into an arg min energy minimization in the MRF framework. Also, we assign the texture term according to the margins of the respective super pixel descriptor from the hyperplane of the support vector machine (SVM).

Color: In the color consistency measurements, we build a color model using GMMs [6], [5] for both foreground and background color distribution. In our case, we vectorize a color pixel in an image Ik n as a nine-dimension vector, stacking the lab colors space and the RGB color space for two different Gaussian blurs.

Suppose we have the current binary segmentation of the foreground and the background, we can collect samples to build the GMM color model:

$$Pr_c(I_k|g_f) = \sum_{c=1}^{c_{max}} w_f^c . N(I_k|\mu_f^c, \sigma_f^c)$$

$$Pr_c(I_k|g_b) = \sum_{c=1}^{c_{max}} w_b^c . N(I_k|\mu_b^c, \sigma_b^c) \qquad (7)$$

Where $w_f^c \cdot N(Ik|\mu c\ f, \sigma f\ c)$ indicates the weighted Gaussian component having mean μc f and variance σf c on the foreground label. We build the GMM color model in a normalized lab color space ranging between 0 and 1. This normalized lab color space gives more weight to the chromatic channels and less weight to the luminance channel to reduce the effects of shadings or shadows. We also apply the Gaussian smoothing to each RGB channel before building the GMMs to remove image noises and subtle image details to avoid over-fitting of the color distribution.

When we measure the distance of a pixel between the foreground and the background, instead of using all Gaussian functions N, we use the 5 nearest Gaussian functions by definingw_f^c, w_f^cas the minimum metric in closer Mahalanobis distances. This is slightly different from the convention considering the number of samples covered by each Gaussian.

Consider the example shown in Figure The proportion of the hand pixel is much smaller than the other foreground pixels. Therefore, the convention GMM will disregard the hand region because it has fewer samples. In our task, we prefer to take all the details of an object because other false alarms can be effectively suppressed by iterative optimization. The color consistency measurements are normalized to satisfy $Pr_c(I_k|g_f)$+ $Pr_c(I_k|g_b)$= 1 for the proper energy terms, and the average of these energies in a group of pixels is assigned to their superpixel node in MRF.

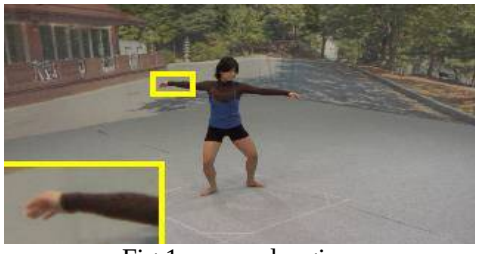


Fig.1. cropped region

ISSN(Online): 2319-8753
ISSN (Print): 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor & UGC Approved Journal)

Website: www.ijirset.com

Vol. 6, Issue 8, August 2017

Texture: Our key idea for encoding texture information is building common texture that is comprehensive foreground/background texture prior regardless of the viewpoints. To encode texture information, we take the luminance channel l in the lab space, and compute the x, y, xy, and yx directional derivatives of Gaussians at two different sigma scales, and Laplacians of Gaussians at three sigma scales. Then those 11-dimensional vectors in one viewpoint are added to the original 9-dimensional vectors of color components. The new 20-dimensional vectors for each pixel in all input images are clustered to create 64 GMMs, followed by description of all superpixel to Fisher vectors with respect to the global GMMs. In this manner, we get a descriptor Sk for superpixel k. After the normalization of superpixel descriptors as in [5], we train a linear SVM either using all positive and negative vectors and their labels X k = 2 · (Xk − 1/2) across viewpoints, or build multiple SVMs by having the samples only in the respective view. This is illustrated as Figure.

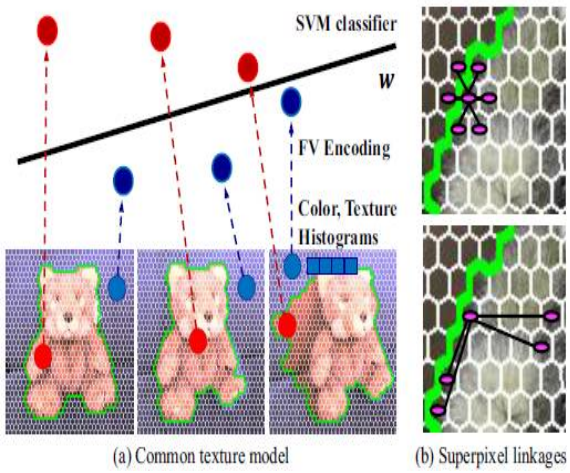


Fig. 2. Texture appearance model. We encode color and texture components into the super pixel descriptors. At every iteration, they train the linear SVM in the Fisher vector space. In MRF, one super pixel is connected to the nearest nodes, and the non-local linkages are formed by the similarity between these super pixel descriptors.

The scores from the trained w give the texture-driven energies Et to each super pixel

$$\min_{w,b} \sum_{k=1}^{k} l(X_k' . w^T S_k) + \frac{1}{2}\|w\|^2 \qquad (8)$$

In building an initial graph with neighborhood connections, the super pixel descriptors are compressively represented as S_k by the standard LDA projections. Linear Discriminant Analysis (LDA) is most commonly used as dimensionality reduction technique in the pre-processing step for pattern-classification and machine learning applications. The goal is to project a dataset onto a lower-dimensional space with good class-separability in order avoid overfitting ("curse of dimensionality") and also reduce computational costs. to finding the component axes that maximize the variance of our data (PCA), we are additionally interested in the axes that maximize the separation between multiple classes (LDA). LDA is "supervised" and computes the directions ("linear discriminants") that will represent the axes that that maximize the separation between multiple classes. Although it might sound intuitive that LDA is superior to PCA for a multi-class classification task where the class labels are known, this might not always the case.

For greater computational efficiency, mean pooling on color vectors is another good approximate option. The pair wise term is implemented by the Potts model in the super pixel segmentation context, where one node is connected to k-nearest neighbors in the X^2metric between the superpixel descriptors:

$$E_{nt}(X_p, X_q) = \sum_{p,q} |X_p - X_q| . exp\left(-\beta\left[X^2(S_p', S_q')\right]^2\right) \qquad (9)$$

we link super pixel descriptors of one image in a similar way as in [1]. However, we develop the way to have a discriminative appearance model so that the gradient magnitudes do not lose directional information. This texture

ISSN(Online): 2319-8753
ISSN (Print): 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor & UGC Approved Journal)

Website: www.ijirset.com

Vol. 6, Issue 8, August 2017

model is sufficiently representative, especially when we have similar texture parts in foregrounds with blurry backgrounds.

D. Energy optimization

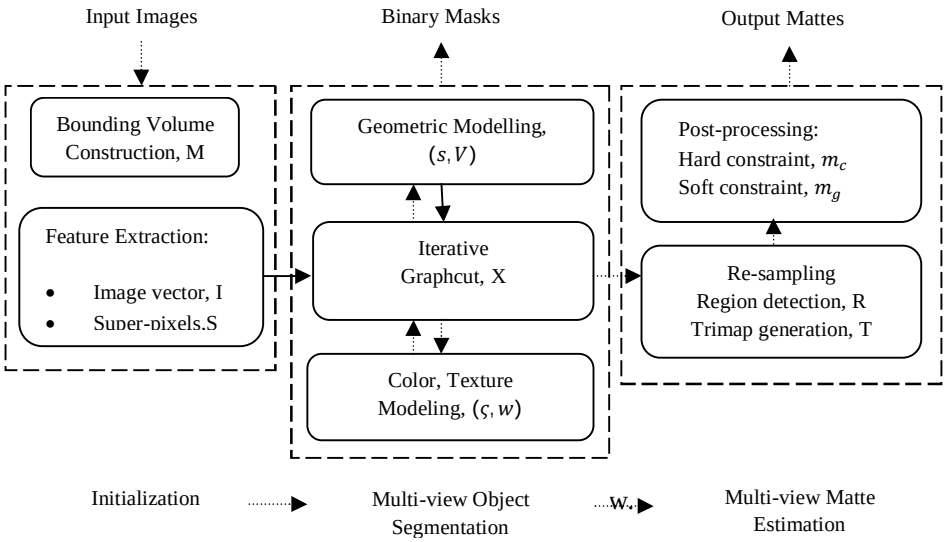
In practice, labeling all pixels at the original image resolution is a time-consuming task. Instead, we take one or two additional coarse resolutions for the rough super pixel segmentations before we start the pixel-level optimization using the initial energy. During iterative refinement of the multi-scale segmentations, we increase the weight of λng. This is because we believe that the 3D hypothesis is not reliable in the beginning, but gradually our 3D surface samples become accurate and dense enough to cover pixel-level score maps.

In the pixel segmentation, we use the typical contrast term for eight connected grids and add view-to-view linkages coming from the final 3D surface structure. We define the neighborhood energy for two adjacent nodes X_p and X_q using color energy Enc and a geometric energy Eng:

$$E_{nc}(x_p, x_q) = \sum_{p,q} |X_p - X_q| . exp\left[-\beta(\mathrm{I_p} - \mathrm{I_q})^2\right]$$

$$\mathrm{E_{ng}}(\mathrm{x_p^i}, \mathrm{x_q^j}) = \sum_{\mathrm{i,j,p,q}} |\mathrm{x_p^i} - \mathrm{x_q^j}| . \mathrm{en_b} \qquad (10)$$

Where β is the same value used for Eq. 9. In our experiment, most cases required less than three iterations to reach stable labeling results. When we could not solve the MRF with more than eight inputs at a high resolution due to memory limitation, we divided them into eight-viewpoint subsets to be respectively optimized after re-computation of the visibility maps.



However, we also found that the 3D structures derived from eight views are good enough to project to the other calibrated views. Hence, we were able to perform per-view graph-cuts using its appearance, geometric models with λng = 0. It is still useful to have such inter-view linkages coming from 3D samples [1] or SIFT-based correspondences [2] in inferring correct labels. However, once we had a good 3D model of interest in the space, we observed that additional regularization was no longer required because the simple projection scores v had already achieved geometrically consistent segmentations.

ISSN(Online): 2319-8753
ISSN (Print): 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor & UGC Approved Journal)

Website: www.ijirset.com

Vol. 6, Issue 8, August 2017

IV. MULTI-VIEW MATTE ESTIMATION

Given the estimated binary labels x and the scores maps v, our next step is to generate trimaps T and estimate alpha mattes α of the foreground object.

A. Supporting region detection

We use the KL divergence to measure the amount of local color spread for matte region detection. The use of the KL divergence is inspired by the two-color line model in [3], which indicated that, for any local color distribution, the effects of color mixing can be approximated by a linear combination of two different colors. If the distribution of color samples is concentrated in the middle of the line model, we can judge that the mixing effect of two colors is strong. In contrast, if the color samples are concentrated at the two end points of the line model, we expect a sharp boundary between two regions. A similar approach of using KL divergence to evaluate the reliability of estimated mattes has been used in. The equation for measuring the KL divergence is given by

$$f_{KL}(r_k, s_k) = \sum_{r'_k \epsilon s_k} \left[d\left(I'_{r'_k}, C_0\right) . \log \frac{d\left(I'_{r'_k}, C_0\right)}{d\left(I'_{r'_k}, C_1\right)} \right] \qquad (11)$$

where r_k is the uniformly sampled seed point along the boundaries, C0 and C1 are the RGB colors of the two end points on the line, respectively, and $d(\cdot, \cdot)$ is the Euclidean distance operator. In our approach, we measure the KL divergence of the local color distribution using the estimated line model. Instead of a regular grid to measure the divergence, we use the various shapes of local windows to adapt the window according to the boundary shape. The KL divergence is measured with various window sizes, ranging from 7 ×7 to 81 × 81 pixels, in three possible shapes (square, thin, thick) having 2, 2, and 9 types of offsets respectively. The windows s_k are illustrated in Figure

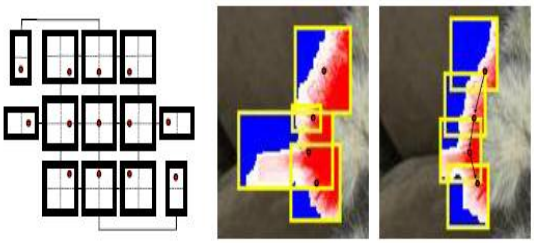


Fig. 4. Parameter optimization. (a) At a fixed size, we have different shapes with shifting offsets of a window depending on the content of its supporting region. (b) We determine window parameters that maximize the cumulative sum of KL divergences along a constrained path. After regularization, the window shapes and sizes along the path change smoothly.

For a region with a sharp boundary, increasing the local window reduces the value of the KL divergence. On the other hand, if a region requires matting, the value of the KL divergence increases with the window size until it reaches the optimum region in the sense of the maximum entropy. Therefore, the matting region is obtained by selecting the window shapes, sizes s_k with maximum KL divergence of the local color distribution.

B. Parameter optimization

After we evaluate various KL divergences with various window shapes, we increase the local window size to effectively separate the foreground and background regions. Specifically, at the positions r_k^*, the window size increases in 13 different directions. Here, the control point $r_k^*$is a better localized position of s_k using the maximum response of the Gaussian (DoG) filter difference. For this purpose, we modify the level set implementations as in [8]. With this procedure, all sampled positions have the KL divergences of the possible window shapes and sizes. The measured divergences are recorded in form of cost volume, and we utilize dynamic programming to maximize the sum of KL divergences along all object boundaries. By indexing the optimum size and shape of a detecting window s_k^* at the best contour point r_k^*, we can easily classify the foreground and background colors as well as the mixed colors.

ISSN(Online): 2319-8753
ISSN (Print): 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor & UGC Approved Journal)

Website: www.ijirset.com

Vol. 6, Issue 8, August 2017

C. Trimap optimization

In the previous section, we described the detection of the optimal local windows of trimaps. We further refine the trimap by MRF optimization. This is done by α-expansion [9] with an objective function similar to that of the first phase. All pixels in the estimated regions are assigned to new labels $T_k \in \{F, B, U\}$ before all the windows are combined with the remaining labels x to constrain the whole image. Each α- expansion iteration can be identically performed by a series of single graph-cuts, and the reasonable local optimum of the objective function in each window is shown. The MRF equation for trimap optimization is defined as follow:

$$E(x') = E_c(x') + \lambda_g . E_g(x') + \lambda_{nc} . E_{nc}(x', y') \qquad (12)$$

For modeling geometric terms, three values are simply taken for energy functions, with λ v =0.9 for x k ∈ F, 0.7 for x k ∈ U, and 0.5 for x k ∈ B, and they make $E_g(X_k = F)(X_k = B)E_g(X_k = U) = en_b$ to give normalized, geometric energies to the graph. Thereby most ambiguities are handled in color models.

The main difference of color models compared to the first phase is that locality of the color samples is given by normalized x, y coordinate information. We also find the pixels using the central colors of a local line model to consider the uncertainty areas, and we build the third GMM to infer the label U. After MRF optimization, the combined trimap can be further refined with simple morphology filters to enlarge the unknown regions. Another way to build GMM for the matting regions is to blend all combinations of foreground and background GMMs as in [10]. In comparison, our method selects blended samples mostly existing on a transition between two strong mean points in a local patch. This selection process captures the better mixed colors at sharp edges and foreground colors in thin structures, while it shows similar performance at blurry edges. In many real world examples, these procedures are successful for our trimap conversions.

D. Matting refinement

Finally, given the trimap and foreground masks, we solve for the fractional boundaries only within the uncertainty regions. This efficient approach is possible because we have the estimated trimap. Our matting framework is based on standard Laplacian matting with additional constraints [3], [11]:

$$\arg\min_{\alpha^*} \alpha^T L\alpha + \lambda_c(\alpha - m_c)^T W(\alpha - m_c) + \lambda_g(\alpha - m_g)^T(\alpha - m_g) \qquad (13)$$

Where m_c is the trimap label acquired from Eq. 12 as a hard constraint. Before the Laplacian is solved, the hard constraints are post-processed by eroding and dilating the estimated foreground/background regions. Here, W is a diagonal matrix with its entries 1 if a pixel belongs to mixed pixels and 0 otherwise; m_g =xn is a soft constraint that utilizes the result of the first phase segmentation, L is the matting Laplacian matrix, and {λc, λg} balance the weights between the two constraints. In Eq. 13, mg guarantees that the estimated alpha matte resembles the sharpness of the first phase segmentation. Therefore, the geometric consistency of the boundaries is implicitly preserved. In Eq. 13, the optimal alpha matte, α, is given by computing the smallest eigenvectors of the composite matting matrix. To recover the true foreground colors using the matte, we use the method proposed in [3], which exploits smoothness prior [5]. The smoothness prior is used to smoothly generate foreground/background color layers by minimizing the x, y directional derivatives of the two layers. In the presence of two simple color distributions, we observe that this assumption is particularly correct along the boundary pixels.

ISSN(Online): 2319-8753
ISSN (Print): 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor & UGC Approved Journal)

Website: www.ijirset.com

Vol. 6, Issue 8, August 2017

IV. SIMULATIONRESULTS

Replacing PCA by LDA, the results are shown

method	Name	Size	Bytes class	Attributes
PCA	Yy	200x4	6400	double
LDA	Xx	3x3	72	double

Fig.5 input Image

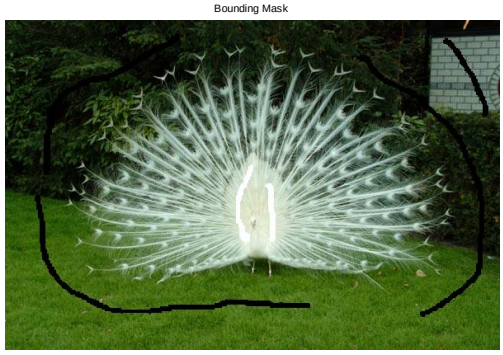


Fig.6 boundary mask

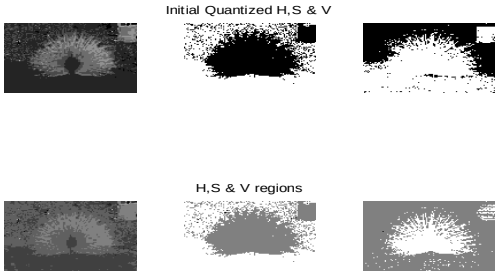


Fig.7 initial quantized H,S &V

ISSN(Online): 2319-8753
ISSN (Print): 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor & UGC Approved Journal)

Website: www.ijirset.com

Vol. 6, Issue 8, August 2017

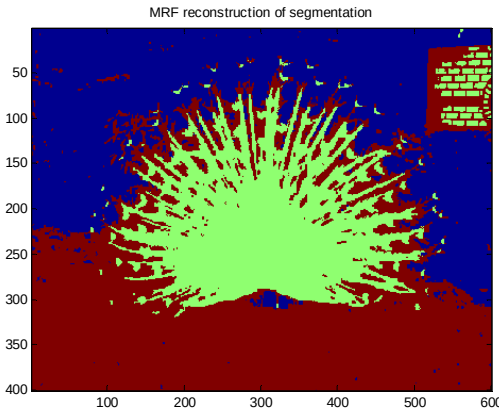


Fig.8 MRF reconstruction of segmentation

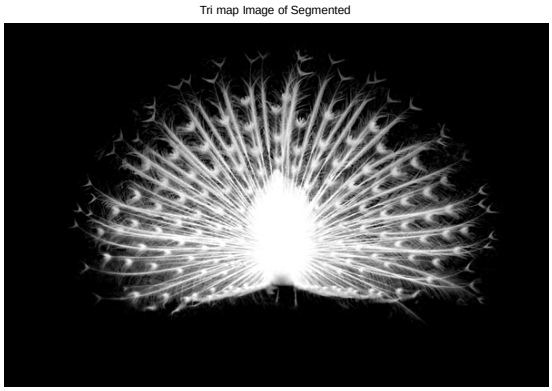


Fig.9 trimap of image

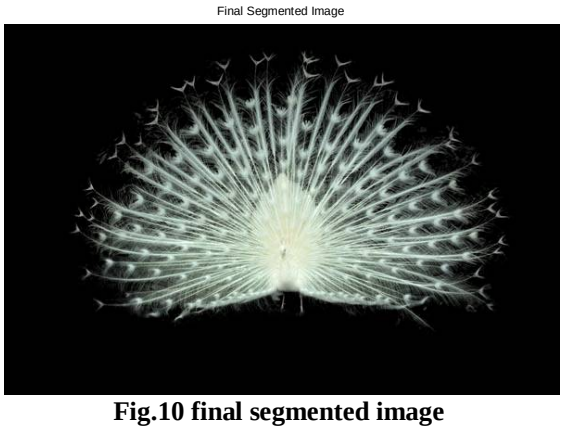


Fig.10 final segmented image

V. CONCLUSION

In this paper, we introduced a framework to extract the soft boundaries of a target object from multi-view images. We utilize coarse 3D reconstruction to define an initial volume bounding the foreground object. Sequentially, we seek geometrically consistent regions having similar appearances across all input images. The Fisher vector encoding adopted in the system allows us to model high-fidelity appearances in images. The consistent regions are cross-

ISSN(Online): 2319-8753
ISSN (Print): 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor & UGC Approved Journal)

Website: www.ijirset.com

Vol. 6, Issue 8, August 2017

validated with one another by referring to their anchor points in space. To detect the optimal matte regions, we optimize the cumulative sum of KL divergences to smoothly take matte regions according to the contexts of object boundaries. Our Laplacian matting equation considers geometrically consistent segmentations in enforcing the multi-view constraint for the final results.

The results of our method were qualitatively and quantitatively compared with state-of-the-art approaches and the key factors in the algorithm were analyzed in detail. Despite its best performance, a few user interventions might be required for some datasets. A few user-given boxes and strokes are effectively shared and propagated across viewpoints for our multi-view segmentation and fractional boundary refinement.

REFERENCES

[1] A. Djelouah and J.-S. Franco and E. Boyer and F.L. Clerc and P. Perez. Multi-view object segmentation in space and time. Proceedings of International Conference on Computer Vision (ICCV), 2013.
[2] J.-Y. Guillemaut and A. Hilton. Joint multi-layer segmentation and reconstruction for free-viewpoint video applications. International Journal on Computer Vision (IJCV), 93(1):73–100, 2011
[3] A. Levin and D. Lischinski and Y. Weiss. A closed form solution to natural image matting. IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI), 30(2):228–242, 2008
[4] C. Rhemann, C. Rother, A. Rav-Acha and T. Sharp. High resolution matting via interactive trimap segmentation. Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2008
[5] A.R. Smith and J.F. Blinn. Blue screen matting. Proceedings of the 23rd annual conference on computer graphics and interactive techniques, 1996
[6] A. Vedaldi and B. Fulkerson. VLFeat: An open and portable library of computer vision algorithms. Proceedings of the 18th ACM International Conference on Multimedia, 2010
[7] J. Shotton, J. Winn, C. Rother and A. Criminisi. TextonBoost for image understanding: multi-class object recognition and segmentation by jointly modeling texture, layout, and context. International Journal on Computer Vision (IJCV), 81(1):2–23, 2009
[8] C. Li and C. Xu and C. Gui and M. D. Fox Distance regularized level set evolution and its application to image segmentation. IEEE Transactions Image Processing (TIP), 19(12):3243–3254, 2010
[9] Y. Boykov and O. Veksler and R. Zabih Fast approximate energy minimization via graph cuts. IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI), 23(11):1222–1239, 2001
[10] O. Juan and R. Keriven. Trimap segmentation for fast and userfriendly alpha matting. Variational, Geometric, and Level Set Methods in Computer Vision (VLSM), 2005