Integrating Color Sampling into Depth Based Bilayer Segmentation

Lorenzo Sorgi and Markus Schlosser

Technicolor R&I, 30625 Hannover, Germany
{lorenzo.sorgi,markus.schlosser}@technicolor.com
http://www.technicolor.com

Abstract. A trend in the computer vision community is observed towards the simultaneous exploitation of color images and depth maps. In this context we propose a novel approach for bi-layer segmentation, whose main contribution is given by the integration of a color classifier based on color sampling, within a depth-based segmentation framework. We have run tests on datasets available online and the outcoming results pointed out the effectiveness of this approach and its suitability for integration in automatic segmentation systems.

1 Introduction

Image segmentation is a fundamental brick of many video editing applications and despite the large volume of literature, it can be still considered an open research area. In the simplest scenario a single foreground subject needs to be extracted from the background scene. Techniques exploiting the assumption of a moving foreground and a static background are usually classified as Background Subtraction [3]. Recently active depth sensors have become available as off-the-shelf components and depth maps have been successfully integrated into the segmentation pipeline, taking advantage of their invariance to lighting conditions. However due to their reduced resolution, they are mostly used for the robust automatic initialization of the foreground mask [10]. Furthermore the misalignment with color images due to differences in viewpoints and spatial resolutions, implies an additional non trivial calibration process. Depth maps extracted from stereo or structure from motion do not suffer from these issues but provide a much lower reliability. A sort of ideal setup composed by a time-of-flight camera and a stereo camera, which combines robustness and resolution from both has been proposed in [16]. In some case depth maps have been used to prepare a trimap for a following alpha matting stage, and some matting techniques also integrate the depth information into their core matte estimation [5]. However, in our opinion their results still suffer from the broad initial trimap [11].

Binary segmentation systems using color and depth may be grouped in feature-level fusion and decision-level fusion. Approaches in the first group typically use a k-means clustering performed on feature vectors consisting of the color components and the spatial position [4,6]. The decision-level fusion systems instead, employ a graph-cuts framework, where the depth is integrated into the data term

A. Petrosino (Ed.): ICIAP 2013, Part I, LNCS 8156, pp. 31–40, 2013.

as a statistically independent variable [9,13,8,1]. Only in [15] the authors chose a voting scheme to combine the output of three classifiers based on background subtraction, color statistics and depth/motion consistency, and this approach is the closest to our work, even though we integrate different classifiers. In this paper we propose a statistical framework for video bilayer segmentation, which uses two independent classifiers based depth and color data. An objective function is defined as a weighted sum of their scores, and space-time regularization terms. All these terms are embedded in a segmentation graph as normalized probability measures. We believe that the numerical homogeneity of the different costs guaranteed by the probabilistic framework, allows for a more accurate treatment of the critical case where the pixel data cannot be explained by any of the distribution models and no useful information is provided by the classifiers. A novelty aspect is provided by the exploitation of the color data only in terms of color distance, whose statistical distribution model can be safely assumed content independent and learnt offline, providing a significant relief to the overall system. Furthermore differently from other techniques we include the color clue only in the second stage of a multiresolution framework, using color sampling. The latter has been recently proposed for alpha matting [12,7], but we are not aware of any attempt to exploitation for binary segmentation. The algorithm also is intrinsically suitable for video segmentation as the smoothness terms are homogeneously formulated both in space and time.

2 Graph Based Segmentation

Let us denote with $\mathbb{L} = \{F, B\}$ the space of binary segmentation labels and with $\alpha_i \in \mathbb{L}$ the i-th pixel label. Using this notation the segmentation of a videosequence is represented by the vector $\alpha = [\alpha_0, \ldots, \alpha_{N-1}]^T$ collecting the labels of the N unclassified pixels. The corresponding estimation problem is formulated as the minimization of an objective function comprised of three weighted terms:

$$\alpha = \min_{\alpha \in \mathbb{L}^N} \left\{ \sum_{i=0}^{N-1} e_{d,i}(\alpha_i) + w_s \sum_{(i,j)\in\Phi_s} e_{s,i,j}(\alpha_i, \alpha_j) + w_t \sum_{(i,j)\in\Phi_t} e_{t,i,j}(\alpha_i, \alpha_j) \right\}. \tag{1}$$

The first contribution in (1), denoted as data term, takes into account the coherence between the segmentation labels and the pixel data. As each pixel carries a 3D color vector and a depth measurement, the data term can be further expanded as a sum of two independent contributions, identified by the indexes c and $_z$:

$$e_{d,i}(\alpha_i) = w_c e_{c,i}(\alpha_i) + w_z e_{z,i}(\alpha_i) \qquad . \tag{2}$$

The other terms in (1), denoted as smoothness terms, provide the penalties for each pair of neighbor pixels differently labeled, within the sets Φ_s and Φ_t that collect the spatial and temporal cliques of the video sequence. The first is formed using the 4-connected neighborhood, while the temporal pairs are established using the background registration homographies. (Sec. 2.3).

The minimization of the objective function (1) can be modelled using the max-flow/min-cut optimization and solved via graph cut[18]. Video sequences normally lead to a massive graph representation and the problem must be tacked using technique ad hoc designed for this purpose [17]. We propose instead a more straightforward multiresolution framework. The sequence is initially processed at low resolution without the contribution of the color data terms. A trimap is then expanded from the contour of initialized segmentation mask and a second processing stage is performed at native resolution on the unknown area of the trimap. Furthermore this approach allows for the exploitation of color sampling in the computation color data term (Sec. 2.1).

2.1 Color Data Term

The data terms $e_{c,i}$ in (2) provide a measure of congruence between colors and segmentation labels. The color likelihood is widely used for this porpose, however we claim that this may not be the ideal way to tackle the task, as the color distribution within a video is strongly content dependent, and therefore an a-priori specified model could be inappropriate for a specific sequence. Futhermore the cumulative nature of global statistical models implies the loss of spatial accuracy, whereas it is not unrealistic that the color has an similar global distribution both in the background and foreground areas, making such models necessarily lose any classification power. On the other hand local models may suffer numerical instability which lead to a poor data representation.

For these reasons we opted for a different approach, inspired by the Alpha Matting technique proposed in [12], but implemented with a more aggressive sampling model (Fig. 2). In the second segmentation stage the unlabeled pixels are contained within the trimap stripe and for each of these two sample sets, denoted as $\mathbb{S}_{\{F,B\}}$, are randomly drawn from near foreground/background areas. We opted for a low set cardinality (3 samples). The two best samples are then selected and the color data terms are computed as a probability measure using the cumulative distribution of their distances in color space to the reference pixel:

$$e_{c,i}\left(\alpha_i\right) = P_c\left(d \leq \min_{k \in \mathbb{S}_{(F,B)}} \left\{d_{i,k}\right\} | \alpha_i = \alpha_k\right) \quad , \tag{3}$$

The remarkable aspect of the cost formulation (3) is that it does not need the inference of any specific color statistics for the video sequence.

Indeed, the color feature used in our segmentation framework is given by the pixel distance in *Lab* color space under the condition of homogeneous segmentation labeling. It is has empirically prooved that the corresponding probability density function, $f_c\left(d_{i,j} | \alpha_i = \alpha_j\right)$, which is also used as kernel for the integration of data terms, is roughly content independent and therefore the corresponding model can be selected and characterized offline. Eventually we claim that using only the color distance as classification feature we can successfully process any video sequence using a general statistical model. This approach at the same time overcomes the typical drawbacks of the color likelihood inference and is a remarkable advantage both in terms of performance and computational cost.

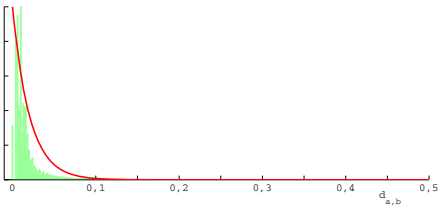

Fig. 1. Statistical distribution of the color distance in smooth areas. In green is represented the normalized histogram computed from the sample set and in red the estimated exponential model ($\lambda = 48.64$).

In our system the color distance likelihood have been estimated from 1e6 pixel pairs sampled from 50 web images with different content. The constraint on homogeneous segmentation has been approximated by two simple conditions, namely a spatial distance lower than 10 pixel and no edge across the connecting segment. These conditions do not guarantee the homogeneous labeling of the selected pixel pairs, however, we observed that the overall quality of the sample set is sufficiently high for our purpose. By visual inspection of the histogram of the color distance data obtained from the sample set we opted for an Exponential model, $f\left(d_{a,b}|\alpha_a = \alpha_b\right) = \lambda e^{-\lambda d_{a,b}}$, (Fig. 1).

The estimated model is then used to compute the data term (3) and the selected samples provide the function argument. As color sampling needs the availability of the sampling areas, in our multiresolution framework we can exploit the color data term only in the second segmentation round, when the trimap expanded from the initialized segmentation mask provides them. A second issue is related to the optimal setting for the trimap width. A narrow stripe boosts the sampling accuracy as the spatial nearness increases the color correlation between reference pixels and drawn samples. On the other hand a broad stripe allows the system to recover in case of poor initialization. Eventually it is extremely difficult to find a satisfying tradeoff. To overcome this problem, we propose an aggressive sampling which relaxes the role of the trimap, namely not only its foreground/background areas are sampled, rather the regions identified by the initial segmentation instead are considered eligible as sampling domains. In other words the samples can be drawn also inside the unlabeled stripe of the trimap (Fig. 2.a,b). The sampling model is shown in more details in Fig. 2.c. We denote with ω the half-width of the trimap and with δ the shortest distance between an unclassified pixel x and the initialized segmentation contour. The two sample set are drawn at $s_i = x + (\delta \pm dx_i) \cdot \eta_i$, where η is a random direction aiming at the contour, dx is a random variable uniformly distributed in the interval $[0, 2\omega]$ and the sign is negative if the initialized label of x and the sample label are equal, positive otherwise.

The proposed sampling strategy allows our system to cope in a natural way with a typical problem of trimap-based processing. Large portions of the foreground or background areas disappear when their linear extent is smaller than

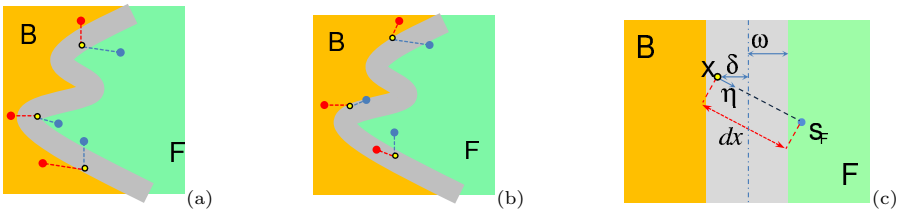

Fig. 2. *Sampling model.* The colors green, orange and grey identify the foreground, background and the unlabeled area of the trimap, and in yellow, blue and red are represented three reference pixels and the background and foreground samples. The conservative sampling draws the samples outside the grey stripe (a), in our approach instead, the initialized mask defines the sampling regions and the samples may belong to the unlabeled stripe (b). (c) Shows the sampling model.

size of the morphology operator used to expand the trimap. This happens for example when a segment has small internal holes, thin in- or out-ward lobes, or it is too close to the image border. In all these circumstances the sampling scheme shown in Fig.2.a is unfeasible as the nearby sampling areas have been absorbed into the trimap. Our approach instead, naturally tackles this problem as the trimap identifies only the unclassified pixels whereas the sampling domains are provided by the initial segmentation.

2.2 Depth Data Term

Similarly to (3) the depth data terms are computed from the depth distribution:

$$e_{z,i}(\alpha) = P_z\left(z \underset{\alpha=B}{\overset{\alpha=F}{\lessgtr}} z_i|\alpha\right) \quad . \tag{4}$$

We observed a general trend to use a Gaussian Model (GM) to capture depth likelihoods. According to our experience this representation is error prone in those areas where the segmentation inference should be easier. When the background contains multiple depth layers a GM polarizes towards one of them and the likelihood drops very low for high depth values. It is evident however, that pixels with very high depth measurements most likely belong to the background. Furthermore it is non-sporadic the case that the probability densities cross in the far depth zone making the depth classifier unreliable. Therefore we suggest a Single Side GM which better describe the natural distribution of depth data:

$$\begin{cases} f(z|\alpha = F) = C_F \cdot e^{-\frac{1}{2}\left(\frac{3z}{z_B}\right)^2} \\ f(z|\alpha = B) = C_B \cdot e^{-\frac{1}{2}\left(\frac{3(z-1)}{1-z_F}\right)^2} \end{cases} \quad , \quad 0 \le z \le 1 \tag{5}$$

The parameters C_F and C_B are estimated by constraining the probability densities intergation, and the values z_F and z_B are given by the first two peaks detected in the depth histogram. The underlying assumption is that the first two

peaks in the depth histograms provide a rough localization of the foreground and background, and accordingly we define the standard deviations to force the corresponding likelihoods near to zero for $z \geq z_B$ and $z \leq z_F$ respectively.

2.3 Smoothness Terms

The space smoothness terms in eq.(1) are related to the image contrast, in order to penalize a labeling swap inside untextured areas. Almost each graph-based segmentation technique uses the color distance between neighbor pixels as a simple approximation of the local contrast. We propose to increase the dimension of the measurement domain up to four pixels, in order to make the measure more robust against image blur. If we denote with (l, i, j, k) a quadruplet of consecutive pixels, then the (i, j) contrast is defined as $\gamma_{i,j} = \max\{d_{l,j}, d_{i,k}\}$. This is still a distance in color space, therefore we suggest to compute the space smoothness term again using the precomputed color distance likelihood:

$$e_{s,i,j}(\alpha_i, \alpha_j) = \delta_{\alpha_i,\alpha_j} \cdot P_c(d \geq \gamma_{i,j} | \alpha_i = \alpha_j) \quad , \tag{6}$$

where $\delta_{i,j}$ denotes the delta Kronecker function.

The underlying idea of the smoothness penalties is similar in the temporal or spatial domain, therefore we opted for reusing the cost expression (6) on the temporal cliques as well. The temporal smoothness terms are then computed as

$$e_{t,i,j}(\alpha_i, \alpha_j) = \delta_{\alpha_i,\alpha_j} \cdot P_c(d \geq \tau_{i,j} | \alpha_i = \alpha_j) \quad . \tag{7}$$

where $\tau_{i,j}$ is the color distance of temporal clique (i, j). These are created by means of the background registration homographies between consecutive frames. At each frame t the registration homography is estimated using the Inverse Compositional Alignment [2] to minimize the photometric registration error:

$$H_t = \min_{H \in \mathcal{H}} \left\{ \sum_{i \in \Omega} f(z_i | \alpha = B) \left\| I_{t,i} - \tilde{I}_{t+1,i} \right\|^2 \right\} \tag{8}$$

where $\mathcal{H}$ is the space of 2D projective transformations, Ω is the registration support, $I_{t,i}$ is the color of i-th pixel at time t and the notation $\tilde{I}$ means the homography warping. The depth likelihood (5) is used as weighting mask in order to polarize the homography estimation towards the background alignment, and the registration support Ω is defined over highly textured areas.

3 Results

In our test we used the weight set $\{w_c = 0.6, w_z = 0.4, w_s = 1.0, w_t = 1.0\}$ to build the objective (1). Different configurations have been also tested and the results do not change significantly. Furthermore we point out that the assignment of meaningful values to the cost weights is an easy task, as each cost term ranges

in the interval $[0, 1]$. The image resolution for the initialization round has been set to 160x120 pixels.

We performed a preliminary evaluation on a data set freely available online with the corresponding ground truth [1]. In Table 1 we present our results compared with those obtained by TofCut [14], the technique proposed by the dataset provider. In the same paper the performance of other three algorithms challenged with the same dataset have been measured. For simplicity we chose not to report these additional results also here, as TofCut is anyway able to outperform all of them. In Fig. 3 some sample frames from the dataset are shown with the corresponding segmentation mask. These results show that the proposed method can outperform TofCut.

Table 1. Comparison between TofCut and the proposed Color Sampling Segmentation with and without temporal smoothness (CSS, CSS_t), on four test video sequences. As perfomance indicator the average percentage of misclassified pixels over the whole sequence is computed.

Seq.ID	# Frames	% Err		
		TofCut	CSS	CSS_t
WL	200	1.35	0.84	0.59
MS	400	0.51	0.27	0.21
MC	300	0.15	0.14	0.12
CW	300	0.38	0.25	0.22

Although the proposed technique turned out to be very accurate, however it is worth further discussing its weak aspects, which will be object of the future refinement work. In this system we did not introduce any adaptive weighting to control the contribution of depth and color clues across the time according to their reliability. Unlike TofCut we opted for a set of constant weights as our main focus was the evaluation of the performance level when color sampling is integrated into a binary segmentation framework. Nevertheless, we consider the adaptive weighting an important feature which will be restored in the future development. The effect of this choice affects only the sequence CW, as a second moving subject is included by our algorithm in the foreground layer whereas in the ground truth it is labeled as background. The error rate provided in Table 1 for this sequence, is computed without considering the frames where this subject is present, otherwise by including the all sequence the error rate raise over %2. However we still consider this result relevant. Indeed, the moving subject is perceived by the depth sensor at the same distance as the static foreground, therefore the misclassification is mostly due to the limitations of the sensor rather than to the segmentation algorithm itself. We do not believe that it is relevant for the algorithm design to accurately model the sensor limitation, many sensors are indeed available on the market which may not suffer of this

[1] `http://vis.uky.edu/~gravity/Research/ToFMatting/ToFMatting.htm`

Fig. 3. Segmentation samples from each of the four dataset

problems. Furthermore for some applications, like video segmentation for 2D/3D conversion, distinct objects with so similar depth can be considered as part of the same foreground layer even if spatially distant. It is also interesting to notice that after the moving subject disappeared, the algorithm does not suffer from any error drift and it is able to recover correctly the segmentation mask corresponding to the main foreground subject.

Further tests have been run on two stereo-sequences at 1920x1080 HD resolution, with a moving foreground and a static but highly textured background. The sequences have been pre-processed using the graph-based matching software available online[2], to extract the depth maps, and the occluded pixels produced by the disparity estimator have been roughly filled using a median filter. In Fig. 4 a sample frame from each of the sequence is shown.

[2] `http://pub.ist.ac.at/~vnk/software.html`

Fig. 4. Segmentation samples from HD stereosequences. The color frames and the disparity maps are shown on the left and the segmented foreground on the right.

4 Conclusion

We presented a novel approach to bi-layer segmentation which integrates color sampling within a depth based segmentation framework. The results obtained from real video sequences, in different formats and with challenging content have confirmed the effectiveness of the approach. Our segmentation system does not require at runtime the estimation of any statistical model for the color data, this is a remarkable advantage as one of the weakest part of almost every Bayesian segmentation framework is therefore completely dropped. Furthermore the proposed objective function encapsulate depth, color and smoothness clues only in terms of probability measures, leading to a consistent normalized range for the edge capacities. This is also a simple but remarkable aspect since it greatly simplifies the choice of the weighting factors for non-expert users. We believe that this work has produced interesting results and it is a promising starting point for the design of a really operative video editing tool.

References

1. Arif, O., et al.: Visual tracking and segmentation using Time-of-Flight Sensor. In: IEEE Int. Conf. on Image Proc., pp. 2241–2244 (2010)
2. Baker, S., Matthews, I.: Lucas-Kanade 20 Years On: A Unifying Framework. Int. J. Comput. Vision 56(3), 221–255 (2004)
3. Benezeth, Y., Jodoin, P.M., Emile, B., Laurent, H., Rosenberger, C.: Review and evaluation of commonly-implemented background subtraction algorithms. In: 19th International Conference on Pattern Recognition, pp. 1–4 (2008)
4. Bleiweiss, A., Werman, M.: Fusing Time-of-Flight Depth and Color for Real-Time Segmentation and Tracking. In: Kolb, A., Koch, R. (eds.) Dyn3D 2009. LNCS, vol. 5742, pp. 58–69. Springer, Heidelberg (2009)
5. Cho, J., Ziegler, R., Gross, M., Lee, K.: Improving alpha matte with depth information. IEICE Electronics Express 6(22), 1602–1607 (2009)
6. Dal Mutto, C., Zanuttigh, P., Cortelazzo, G.M.: Scene Segmentation by Color and Depth Information and its Application. In: STreaming Day (2010)
7. Gastal, E., Oliveira, M.: Shared Sampling for Real-Time Alpha Matting. Computer Graphics Forum 2(29), 575–584 (2010)
8. He, H., McKinnon, et al.: Graphcut-based interactive segmentation using colour and depth cues. In: Australasian Conf. on Robotics and Automation ACRA (2010)
9. Kolmogorov, V., Criminisi, A., Blake, A., Cross, G., Rother, C.: Bi-Layer Segmentation of Binocular Stereo Video. In: Int. Conf. on Computer Vision and Pattern Recognition, vol. 2, pp. 407–414 (2005)
10. Leens, J., Piérard, S., Barnich, O., Van Droogenbroeck, M., Wagner, J.-M.: Combining Color, Depth, and Motion for Video Segmentation. In: Fritz, M., Schiele, B., Piater, J.H. (eds.) ICVS 2009. LNCS, vol. 5815, pp. 104–113. Springer, Heidelberg (2009)
11. Wang, J., Cohen, M.F.: Image and video matting: a survey. Found. Trends. Comput. Graph. Vis. 3(2), 97–175 (2007)
12. Wang, J., Cohen, M.: Optimized Color Sampling for Robust Matting. In: Int. Conf. on Computer Vision and Pattern Recognition (2007)
13. Wang, L., Gong, M., Zhang, C., Yang, R., Zhang, C., Yang, Y.: Automatic Real-Time Video Matting Using Time-of-Flight Camera and Multichannel Poisson Equations. Int. Journal of Computer Vision 97(1), 104–121 (2012)
14. Wang, L., Zhang, C., Yang, R., Zhang, C.: TofCut: Towards Robust Real-time Foreground Extraction Using a Time-of-Flight Camera. In: 3DPVT Conf. (2010)
15. Zhang, G., Jia, J., Hua, W., Bao, H.: Robust Bilayer Segmentation and Motion/Depth Estimation with a Handheld Camera. IEEE Trans. on Pattern Anal. Mach. Intell. 33(3) (2011)
16. Zhu, J., Liao, M., Yang, R., Pan, Z.: Joint depth and alpha matte optimization via fusion of stereo and time-of-flight sensor. In: Int. Conf. on Computer Vision and Pattern Recognition, pp. 453–460 (2009)
17. Lin, F., Cohen, W.: Power Iteration Clustering. In: 27th Int. Conf. on Machine Learning (2010)
18. Kolmogorov, V., Zabin, R.: What energy functions can be minimized via graph cuts? IEEE Trans. on Pattern Anal. Mach. Intell. 26(2), 147–159 (2004)