A NOVEL METHOD FOR AUTOMATIC 2D-TO-3D VIDEO CONVERSION

[1]Youwei Yan, [1]Feng Xu, [1]Qionghai Dai, [2]Xiaodong Liu

[1]TNList and Department of Automation, Tsinghua University, Beijing, China
[2]Graduate School at Shenzhen, Tsinghua University, Shenzhen, China
{yyw08, xufeng07}@mails.tsinghua.edu.cn
qhdai@tsinghua.edu.cn
xiaodong-liu@163.com

ABSTRACT

In this paper, we propose an efficient scheme to automatically convert existing 2D videos to 3D ones. The proposed method extracts motion information from two consecutive frames to estimate depth map for each of them. In the method, we first develop a region-based *Graph cut* method to fast and accurately perform motion segmentation, which is robust to large inter-frame motions. Then, a depth assigning step for the segments is conducted to obtain a smooth depth map for each frame. Experimental results on standard testing sequences demonstrate that our scheme achieves accurate motion segmentation and accordingly smooth depth map.

Index Terms **—** Over segmentation, EMD, Graph cut, Motion segmentation, 2D-to-3D

1. INTRODUCTION

With the development of relative techniques and pressing requirements of many practical applications, three dimensional (3D) visualization is becoming more and more attractive. Recently, 3D cameras such as stereo cameras and depth cameras have been invented, which aim at capturing 3D scene directly. However, they are either rather complicated to operate, or very expensive. Some economic and efficient methods were proposed to convert existing 2D videos in the last decade. Generally speaking, conversion algorithms can be divided into two categories, semi-automatic 2D-to-3D conversion and automatic 2D-to-3D conversion. As the former one needs human intersection, it can't be widely adopted in many scenarios. As a result, automatic video conversion methods have drawn more and more attention.

Recently, some methods [1][2][3][4] have been proposed to automatically convert 2D videos to 3D ones by generating depth maps associated with the original video clips. In [1], segmentation was achieved by combining color information with segments generated via optical flow using minimum discrimination information (MDI) principle. As optical flow algorithm was used to extract pixel-level motion, this method can only handle small motion. Li et al [2] proposed another 2D-to-3D conversion algorithm based on KLT tracking. In this method, KLT was used to track contour features by analyzing frame-difference map. After that, depth was assigned to each segment. However, more than two video frames were needed to fulfill the segmentation process and an additional initialization step was required. Chang et al [3] explored motion using frame difference method, and used K-Means algorithm to implement color segmentation. Generating depth map required both time and spatial information. Nevertheless, this frame difference method needed a large number of neighboring frames to achieve good performance, and K-Means algorithm required manual operation to define initial seeds. Kunter et al [4] adopted structure-from-motion method to estimate camera parameters and then background image was estimated to get the background model. Finally, independent moving objects were segmented by detecting the changed information and each object was assigned a constant depth value. This algorithm can handle even complex and challenge scene with moving camera and independent moving objects, but more than two frames are needed to construct the background model.

Since segmentation is very important in automatic 2D-to-3D conversion, we turn to survey some related researches for segmentation algorithms. Among these algorithms, there are some dealing with scenes containing large motion using only two consecutive frames. For instance, Wills et al [5] proposed a novel framework for motion segmentation which used the detected features to estimate motion layers with the same homographic motion and all pixels were grouped to each layer via graph cut algorithm [6]. However, the segmentation results were not accurate enough to obtain a smooth depth map. In addition, time complexity was rather high due to time-consumed feature detection and pixel-level *Graph cut*.

In order to overcome the disadvantages of above methods which can't handle large inter-frame motion using only two consecutive frames, we propose a novel automatic 2D-to-3D video conversion scheme. Firstly, sparse features of two consecutive frames are detected and matched. Secondly, homographic motion layers are estimated based on matched features. Moreover, graph cut is utilized to label areas which are obtained by over segmentation. Finally, depth is assigned to each semantic segment.

2. MOTION SEGMENTATION AND DEPTH ASSIGNMENT

In this part, we propose a novel method to achieve good segmentation using both color and motion information inspired by Wills' work [5]. With the segmentation results whose boundaries are accurate enough, depth is assigned to segments to generate smooth depth maps. The proposed method contains the following steps: 1. Sparse feature detection, matching and planar homograph estimation. 2. Over-segmentation using improved *Mean shift algorithm* [7]. 3. Labeling segments using modified *Graph cut algorithm* in MRF framework. 4. Depth assignment.

This work was supported in part by the National Basic Research Project of China, No.2010CB731800, in part by The Key Program of NSFC, No.60932007 and in part by The National High Technology Research and Development Program of China, No.2009AA01Z329.

978-1-4244-6379-4/10/$26.00 ©2010 IEEE

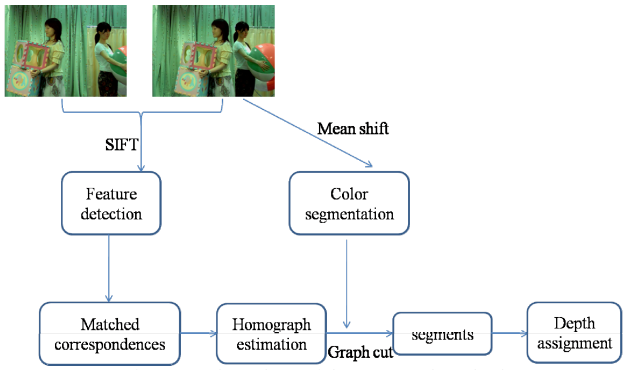

Figure 1. Flow chart of the proposed method

The main contribution of our work lies in steps 2 and 3 which are detailed in section 2.2 and 2.3. The Flow chart of the proposed method is illustrated in Figure 1.

2.1 Feature matching and Homograph estimation

Based on our previous work [8], step 1 and 2 can be performed as follows. We adopt Scale-invariant feature transform (SIFT) [9] to extract sparse features, which applies a difference of Gaussian function to identify feature points and calculates a descriptor for each feature point by using image gradients around a radius of the feature. SIFT has two appealing advantages. First, SIFT can locate the feature points with sub-pixel accuracy. Secondly, the SIFT descriptors are robust and distinctive, even if the feature points are under rotation, blurring, scale change, or illumination change. In addition, estimating homograph matrix of each motion layer only needs 4 or 6 feature points. That is to say, sparse feature points are enough. It can be done by SIFT which extracts sparse feature points in a rather short time. But Harris corners and Förstner features take a lot of time to extract dense feature points which are not indeed needed.

When feature points are detected, we developed a bidirectional feature matching method based on Lowe's feature matching method to establish sparse correspondences between two adjacent frames. Using these correspondences, we estimate the number of motion layers and homograph matrix of each layer using the method described in [8].

2.2 Over segmentation

After these procedures, the homographies of motion layers have been determined. The next step is to assign all pixels to proper motion layers, which becomes a labeling problem. It can be solved using *Graph cut*. However, if we do the energy minimization for each pixel, it only uses local information and may be sensitive to noise. In addition, it costs plenty of time. So we choose to segment the image into semantic regions because regions can supply global information and region-level *Graph cut* can save much more time. The over segmentation results are shown in Figure 2.

Our over segmentation procedure is achieved by *Mean shift* algorithm. And segmented regions should have the following properties: 1. They must contain at least 50 pixels, or the region is fused into other regions. 2. They have better not cross the object boundaries.

2.3 Region-based *Graph cut*

After over segmentation was done, we want to classify them into the estimated motion layers via *Graph cut*. In the previous work [5], the problem of assigning each pixel to proper motion layer


(a) (b)

Figure 2. (a) Over segmentation result of a frame extracted from *Akko&Kayo* sequence (published by Tanimoto Laboratory, Nagoya University) (b) Over segmentation result of a frame extracted from *Alt_moabit* sequence [14].

can be formulated as determining a function l that maps each pixel to an unique motion label from label set $L=\{1,...,m\}$, where $1,...,m$ present motion layers. The establishment of the function l for a certain frame t can be achieved by minimizing the following energy function:

$$E(l,I^t,I^{t+1}) = E_{data}(l,I^t,I^{t+1}) + \lambda E_{smooth}(l,I^t) \quad (1)$$

where I^t is intensity of frame t. The energy function has two terms with a penalized factor λ between them. In our method, pixels are substituted by regions, so we have to modify the data term and smooth term in order that the energy function can be solved by modified *region-base graph cut*. In [5], the assumption that the appearance of the object remains the same across the images is adopted. However, this assumption is not valid for occluded pixels because they only appear in one of the two frames. To handle this problem, in [8], we have added an occlusion label to the data term. It is same for regions in our method. The data term which addresses the reconstruction error can be represents as the following formulation:

$$E_{data}(l,I^t,I^{t+1}) = \sum_i \begin{cases} [I^t(i) - I^{t+1}(M(l(i),i))]^2 & \text{if } l \in \{1,...,m\} \\ d & \text{if } l = m+1 \end{cases} \quad (2)$$

where $I^t(i)$ denotes mean intensity of region i in frame t and $M(l(i),i)$ returns new label of region i in the adjacent frame under the influence of motion $l(i)$. d is a constant parameter modeling the reconstruction error for occluded regions. If $l(i)$ is ranged in $\{1,\cdots,m\}$, we see that the difference in intensity is used to model the reconstruction error. However, if it equals to $m+1$, a constant parameter d is utilized. For one region, if the reconstruction error, caused by assigning it to any real motion layer, is greater than the constant d, the region is likely to be assigned with the occlusion label. So if d is set to a too large value, some occluded regions may be wrongly assigned with motion label. Meanwhile, if d is set to a small value, visible regions may be assigned to the occlusion layer. As to smooth term, we first decide the relation of regions by judging the connection of neighboring 8 pixels. If two neighboring pixels belong to different regions, the two regions are treated as two neighboring ones. If two regions do not contain this kind of pixel pairs, we think the two regions are not connected with each other. Following [5], smooth term is as follows:

$$E_{smooth}(l,I^t) = \sum_i \sum_{j \in N(i)} s_{ij}(I^t)\left[1 - \delta_{l(i)l(j)}\right] \quad (3)$$

here, $s_{ij}(I^t)$ is the similarity between two regions i and j in frame t. δ equals to 1 when its arguments are equal, otherwise, it equals to 0. $N(i)$ represents the neighborhood of region i, which contains all regions which are connected with region i. To define the similarity between regions, the idea of bilateral filter [10] is utilized. In bilateral filter, the weight in the filtering combines the geometry closeness and the photometric similarity between two regions. To involve this property in our measure-

ment of similarity, we utilize the weight in bilateral filter as the similarity of two regions. The similarity is defined as follows:

$$s_{ij}\left(I^t\right)=\Phi_{ij}^{X}(I^t)\Phi_{ij}^{C}(I^t) \tag{4}$$

The first term $\Phi_{ij}^{X}(I^t)$ is the closeness function and the second term $\Phi_{ij}^{C}(I^t)$ is the photometric similarity function. In simple and important case, these two functions are Gaussian functions of the Euclidean distance between their arguments. More specifically, $\Phi_{ij}^{X}(I^t)$ has the following formulation:

$$\Phi_{ij}^{X}(I^t)=\exp\left[-\frac{D_{center}\left(i,j\right)^2}{\sigma_X}\right] \tag{5}$$

where $D(i,j)$ is the Euclidean distance between the centers of regions i and j and σ_X is the bandwidth. The formulation of $\Phi_{ij}^{C}(I^t)$ is perfectly analogous:

$$\Phi_{ij}^{C}(I^t)=\exp\left[-\frac{D_{EMD}(i,j)^2}{\sigma_C}\right] \tag{6}$$

where $D_{EMD}(i,j)$ is Earth Movers' Distance (EMD) [11] between regions i and j. σ_C is the bandwidth. We choose EMD as the photometric similarity metric instead of difference of two regions' mean intensities for the following reason. EMD is statistical distance between two regions. As areas of two regions are generally not equal, EMD is more accurate than distance obtained by mean intensity difference of two neighboring regions. For instance, assuming that gray values of all pixels in one region are 255/2, while in another neighboring region, gray values of all pixels in it are either 0 or 255. And the sums of the two kinds of pixels are equal. The mean intensity difference of the two regions is 0. However, the two regions indeed are totally different in aspect of gray level. So mean intensity difference is invalid or may even lead to bad results in these situations. In our method, we formulate EMD as follows. Firstly, we gather statistics of the gray values of all pixels in one region by dividing the range [0,255] into N average sub-ranges and then storing how many pixels ($N_i, i=1,\cdots,N$) fall into each sub-range. Accordingly, the probability of pixels falling into each sub-range is expressed as $\frac{N_i}{N}, i=1,\cdots,N$. Until now, two histograms for two neighboring regions are established. Finally, EMD distance can be solved as a *transportation problem*. In equation (3), $s_{ij}(I^t)$ is utilized to penalize the discontinuity assignment of motion. As motion segmentation results should be piecewise constant, this definition is reasonable.

From the formulation of energy function (1), we see that data term leads the assignment result to be consistent with the motion and occlusion in the scene, and smooth term guarantees the piecewise constant property.

The above energy function can be minimized as a metric labeling problem. Kleinberg and Tardos [12] show that this kind of problems equals to find the maximum a posteriori labeling of a class of Markov random field. We adopt Boykov's method to solve the problem which is a polynomial time algorithm. As proposed by Wills [5], we also do the intersection step so that occluded pixels are further removed.

2.4 Depth assignment

In this subsection, we assign depth to segments obtained above. Guttmann et al [13] demonstrated that depth order is the most important clue for 3D vision rather than the exact depth values. And like what we have proposed in [1] for depth assignment, we also obey the three *Rules* to assign depth for each segment. The rules are as follows:

Rule 1. We suppose the pixels at the boundaries of the image belong to the background of the scene while the segments in the center stand for the foreground objects.

Rule 2. One segment is assumed to have unique depth value.

Rule 3. Some background object like a wall or a building always has a consistent depth value and this value always implies the farthest distance in the scene.

3. EXPERIMENTAL RESULTS

The proposed method is tested on two video sequences and segmentation results are compared with Wills' method in aspects of both segmentation quality and time complexity. All experiments are performed via Matlab and C++ hybrid programming on the computer with the following configuration: Intel(R) Core(TM)2 Duo CPU E7500 2.93Hz 4.00GB Memory. Figure 3, Figure 4 and Figure 5 show segmentation results obtained by our method

Figure 3. The first row is the original first frame extracted from *Akko&Kayo* sequence. The second is obtained by proposed method. The third row is obtained by Wills' method [5].

Figure 4. The first row is the original second frame extracted from *Akko&Kayo* sequence. The second is obtained by proposed method. The third row is obtained by Wills' method [5].

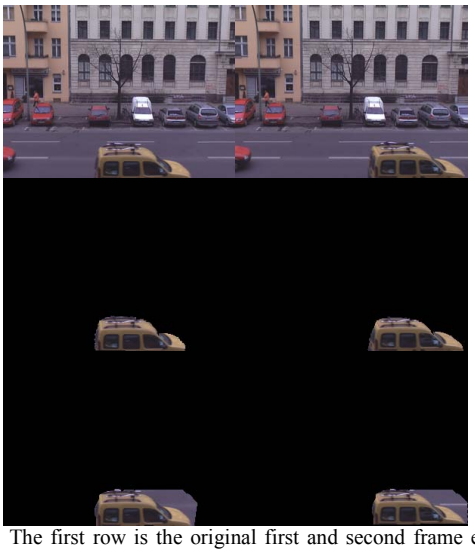

Figure 5. The first row is the original first and second frame extracted from *Alt_moabit* sequence. The second is obtained by proposed method. The third row is obtained by Wills' method [5].

TABLE I

COMPARISON OF THE COMPLEXITY PERFORMANCES BETWEEN WILLS' METHOD AND THE PROPOSED METHOD

	Akko&kayo (384×288)		*Alt_moabit (308×231)*	
	Wills' method	Our method	Wills' method	Our method
MS		11.07s		6.36s
GC	48.01s	0.10s	40.25s	0.278s
TT	174.36s	26.15s	152.16s	11.5s

MS is short for Mean Shift. *GC* is short for Graph Cut. *TT* is short for Total Time of motion assignment.

and Wills' method [5]. TABLE I shows the comparison of time complexity between proposed method and Wills' method [5].

As we can see from the experimental results, our method improves both segmentation quality and efficiency. Finally, we assign depth to all segments and smooth depth maps are illustrated in Figure 6 and Figure 7.

4. CONCLUSIONS

This paper presents a novel method to automatically convert existing 2D videos to 3D videos. We combine the color information and motion information to achieve accurate and fast segmentation. The depth map is smooth as the boundaries of objects are improved. Future works will mainly focus on two parts. The first is how to estimate motion model for non-planar scenes which has been tried by Wills' method [5]. The second is how to assign more proper depth for segments which is consistent with realistic situation.

5. REFERENCES

[1] F Xu, Q Dai, X Xie, "2D-to-3D Conversion Based on Motion and Color Mergence," *IEEE 3DTV Conference*, pp. 205-208, May.2008.

[2] T Li, Q Dai, X Xie, "An Efficient Method for Automatic Stereoscopic Conversion," *IEEE VIE Conference*, pp. 256-260, Aug.2008.

[3] Y Chang et al, "Depth Map Generation for 2D-to-3D Conversion by Short-term Motion Assisted Color Segmentation," *IEEE Conference on ME,* pp. 1958-1961, July.2007.

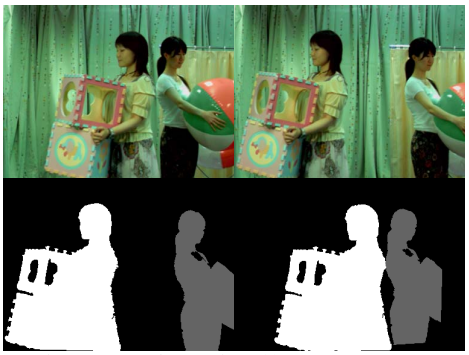

Figure 6. Depth map for *Akko&Kayo* sequence

Figure 7. Depth map for *Alt_moabit* sequence

[4] M. Kunter, S. Knorr, A. Krutz and T. Sikora, "Unsupervised Object Segmentation for 2D to 3D Conversion," *Proc. SPIE*, Vol. 7237, February. 2009.

[5] J. Wills, S. Agarwal, and S. Belongie, "A Feature-based Approach for Dense Segmentation and Estimation of Large Disparity Motion," *Int. J. Comput. Vis.*, vol. 68, no. 2, pp. 125-143, June 2006.

[6] Y. Boykov, O. Veksler, and R. Zabih, "Fast approximate energy minimization via graph cuts," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 11, pp. 1222– 1239, November 2001.

[7] D. Coman and P. Meer, "Mean Shift: A Robust Approach Toward Feature Space Analysis," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol.24, no. 5, pp. 603-619, May 2002.

[8] F Xu and Q Dai, "Occlusion-Aware Motion Layer Extraction under Large Inter-Frame Motions," *Submitted to IEEE Transactions on Image Processing*, 2009.

[9] DG. Lowe. "Object recognition from local scale-invariant features,". *International Conference on Computer Vision*, pages 1150–1157, Corfu Greece, Sept. 1999.

[10] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," in *Proc. of IEEE Int. Conf. on Computer Vision.*, Jan. 1998, pp. 836–846.

[11] Y. Rubner, C. Tomasi, L.J. Guibas, "The earth mover's distance as a metric for image retrieval," *Int. J. Comput. Vis*, vol. 40, no. 2,pp. 99-121, November 2000.

[12] J. Kleinberg and E. Tardos, "Approximation Algorithms for Classification Problems with Pairwise Relationships: Metric Labeling and Markov Random Fields," J. *ACM*, vol. 49, no. 5, pp. 616-630, September 2002.

[13] M. Guttmann, L. Wolf, "Semi-automatic Stereo Extraction from Video Footage," *IEEE ICCV*, 2009

[14] Feldmann, M. Mueller, F. Zilly, R. Tanger, K. Mueller, A. Smolic, P. Kauff, T. Wiegand "HHI Test Material for 3D Video", MPEG 2008/M15413, Archamps, France, April 2008.