A NOVEL METHOD FOR 2D-TO-3D VIDEO CONVERSION USING BI-DIRECTIONAL MOTION ESTIMATION

Zhenyao Li *Xun Cao* *Qionghai Dai*

Department of Automation, Tsinghua University, Beijing, China

ABSTRACT

In this paper, we proposed a novel semi-automatic 2D-to-3D video conversion method. Our method requires just a few user-scribbles to generate depth maps for key frames and propagates these depth maps to non-key frames automatically. For key frames, foreground objects and the corresponding depth maps can be obtained by an interactive method. Then, both forward and backward motion vectors are estimated and compared to decide the depth propagation strategy. For pixels that failed the motion vectors comparison, a compensation process is adopted to refine their depth propagation results. Finally, stereoscopic pairs are generated by the warping method based on the original frames and associated depth maps. Our method is validated by both subjective and objective quality assessments. The experimental results show that our method outperforms several state-of-the-art 2D-to-3D video conversion methods.

Index Terms— Stereo vision, 2D-to-3D conversion, depth propagation, motion estimation

1. INTRODUCTION

3D (mostly stereoscopic) video enhances traditional viewing experience dramatically by providing an immersive perception. As 3D videos become more and more popular, 3D content seems to be quite insufficient compared with the growing number of 3D display devices. The 3D content shortage has limited the development and promotion of 3D technology. Considering the huge amount of existing 2D videos, 2D-to-3D video conversion is expected to satisfy the growing need for high quality 3D videos and attracts attention from both industrial and academic communities [1].

According to whether human-computer interactions are involved, 2D-to-3D video conversion methods can be divided into two categories: fully-automatic methods and semi-automatic methods.

Fully-automatic methods Fully-automatic methods can generate 3D videos directly from 2D inputs without any human-computer interactions. Structure from motion (SfM) methods [2] are widely studied in computer vision area and able to recover 3D structure of the scene automatically. Zhang *et al.*[3] employed the SfM method to recover consistent video depth maps via bundle optimization. In [4], Knorr *et al.*proposed a modular system that is capable of efficiently reconstructing 3D scenes from broadcasting video. However, SfM methods have certain restrictions on camera movement and scene motion, which reduces the extensive availability of these methods. Recently, Zhang *et al.*[5] integrated visual attention and occlusion analysis to calculate depth map. Compared with the SfM-based methods, [5] extracts monocular depth cues from the scene and poses no limitation on the underlying 2D video.

Semi-automatic methods By introducing human-computer interactions, semi-automatic methods can balance quality and cost more flexibly than fully-automatic methods. Stereo quality and conversion cost are determined by the key frame intervals and the accuracy of depth maps on key frames. Smaller interval and more accurate depth map will improve the stereo quality, but increase the conversion cost as well. Therefore, a tradeoff has to be made in order to obtain satisfactory quality at acceptable cost. Guttmann *et al.*[6] presented a semi-automatic system which just requires user-scribbles on the first and last frames of the video clip for the purpose of reducing manual labor. Their system employs the SVM classifier trained on the marked frames to produce disparity maps for the entire video clip through an optimization process. Yan *et al.*[7] demonstrated a depth map generation scheme based on user inputs and depth propagation. They specify the depth values of the selected pixels and locate the approximate positions of T-junctions by user inputs, and then generate depth maps by depth propagation combining user inputs, color and edge information.

Another method was proposed by Varekamp and Barenbrug [8], which used bilateral filtering algorithm to produce a per-pixel depth estimation and correct the estimation through a block-based motion compensation procedure. Recently, Cao *et al.*[9] proposed a semi-automatic conversion method that adopted a multiple objects segmentation algorithm to create disparity maps for key frames and then employed the shifted bilateral filtering algorithm to propagate disparities to non-key frames.

For the purpose of producing high-quality and cost-effective 3D videos, we propose a semi-automatic conversion method which requires only a few user instructions on key frames and propagates the depth maps to non-key frames via bi-directional motion estimation automatically. In our method, object edges in depth maps are propagated by taking both color and motion information into consideration. This operation is based on the observation that the regions where depth changes dramatically (often happens at edges) play as a key factor when viewing 3D video. The experimental results demonstrate that our approach has better performance both in subjective test and objective quality assessment algorithms.

The paper is organized as follows. In section 2, we describe the proposed method; In section 3, experimental results are presented; At last, conclusions are drawn in section 4 and future work is also discussed in that section.

2. THE PROPOSED METHOD

The proposed 2D-to-3D conversion method includes two major stages: key frame depth generation stage which requires human-computer interactions, and non-key frame depth propagation stage which is performed automatically. Key frame depth generation stage extracts foreground object and assigns depth values to both foreground object and background. When key frame depth maps are created, bi-directional motion vectors are estimated and compared to decide the appropriate depth propagation strategy, followed by a compensation process to refine the propagation results. Finally, a

stereoscopic video is synthesized through the warping method [10] using the original frames and their associated depth maps as input.

2.1. Key frame depth generation

Key frame depth generation stage is the only part that involves user instructions in our semi-automatic 2D-to-3D conversion method. This stage starts with a foreground object extraction operation and ends with a depth assignment process.

Scribble-based object extraction methods were made popular by Boykov and Jolly [11], and adopted by 2D-to-3D conversion methods recently [9][12]. In order to achieve realtime feedback on the extraction results, we perform pre-segmentation operations first. The pre-segmentation operations aim at aggregating similar pixels into regions so as to reduce computational cost and human-computer interaction lag. Existing semi-automatic conversion methods [9][12] employed the watershed algorithm to pre-segment the key frames. However, the watershed algorithm has several major drawbacks:

1. The watershed algorithm is susceptible to image noise.
2. The watershed algorithm often causes the over-segmentation problem that generates lots of tiny regions.
3. The watershed algorithm has some difficulties in finding correct object edges when the image contrast is low.

To overcome these drawbacks, we employ the K-means over-segmentation algorithm used in [7] as the pre-segmentation algorithm. The basic idea of this algorithm is to utilize the K-means algorithm to create the cluster results at first, and then convert the cluster results into connected regions through the connected domain algorithm, followed by a post-processing step to ensure its effectiveness. The advantage of the K-means over-segmentation algorithm is that there are no significant differences in size and shape between the pre-segmented regions.

When pre-segmentation operations are completed, we need to manually mark scribbles on the key frames to indicate foreground object and background. We can denote the regions that indicated to be foreground object as $\boldsymbol{F}$, regions that indicated to be background as $\boldsymbol{B}$ and unmarked regions as $\boldsymbol{U}$. Then, foreground object can be extracted by minimizing the energy function below [13].

$$E(L) = \sum_{p} D_p(L_p) + \sum_{p,q} V_{pq}(L_p, L_q) \tag{1}$$

where L_p is the label (foreground or background) of pixel p and $L = \{L_p\}$ is a labeling of the whole image. D_p is a data penalty function while V_{pq} is the interaction potential between pixel p and q.

This energy function can be optimized by the max-flow/min-cut algorithm [13], which perceives the image as a graph with foreground and background terminals and formulates the optimization problem into finding a cut with minimum cost on the graph that partitions the regions into foreground and background. The cost (*i.e.* weight) of each edge in the connected regions map is defined according to the Euclidean distance in the RGB color space.

For edges that connect adjacent regions:

$$W_{ij} = \frac{1}{\|C_i - C_j\| + \|P_i - P_j\|} \tag{2}$$

For edges that connect region i to foreground or background

$$\begin{cases} W_{iF} = 1 + \sum\limits_{j} W_{ij} \quad W_{iB} = 0 & \forall i \in \boldsymbol{F} \\ W_{iF} = 0 \quad W_{iB} = 1 + \sum\limits_{j} W_{ij} & \forall i \in \boldsymbol{B} \\ W_{iF} = 1/\min\limits_{f}(\|C_i - C_f\| + \|P_i - P_f\|) & \forall i \in \boldsymbol{U} \\ W_{iB} = 1/\min\limits_{b}(\|C_i - C_b\| + \|P_i - P_b\|) & \end{cases} \tag{3}$$

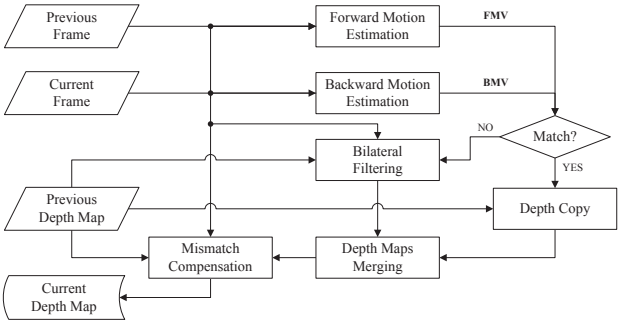


Fig. 1. The pipeline of non-key frame depth propagation stage.

In equations above, C_i and P_i are the average color and position of region i. W_{ij} is the edge weight between region i and j. C_f and P_f are the average color and position of region f that is indicated as foreground $\boldsymbol{F}$ while C_b and P_b are defined accordingly. W_{iF} and W_{iB} are the edge weights connecting region i to foreground/background.

After the edge weights are defined, foreground object can be extracted by the max-flow/min-cut algorithm. Then, key frame depth map is generated through a depth assignment process. This process is executed separably on foreground object and background. We can choose to assign depth via the stroke-based methods [6][7] or the model-based methods [9]. At last, depth maps of foreground object and background are combined to form a key frame depth map.

2.2. Non-key frame depth propagation

Non-key frame depth propagation stage is designed to create depth maps for non-key frames and does not require any user-instruction. Here, we proposed a depth propagation scheme that involves bi-directional motion estimation (*i.e.* forward and backward motion estimation) to decide the appropriate propagation strategy. Then, depth maps created by different propagation strategies are merged, and corrected by a mismatch compensation method to improve accuracy, as shown in Fig. 1.

In the bi-directional motion estimation step, we introduce the variable block-size motion estimation algorithm used in H.264 standard. Considering two successive frames t and $t+1$, the forward motion vectors are obtained by setting frame t as reference frame and calculating the motion vectors from frame t to frame $t+1$ using the algorithm mentioned above while the backward motion vectors are obtained in a reverse way.

When both forward and backward motion vectors are obtained, we need to check whether the two kinds of motion vectors match with each other. We define a mask $mask^{(t)}(x)$ to represent the match checking results at pixel x on frame t, where 1 means we find a match and 0 means no match. All masks should be set to 0 first and calculated as follows:

$$v = u + \boldsymbol{FMV}^{(t)}(u) \tag{4}$$

$$m = v + \boldsymbol{BMV}^{(t+1)}(v) \tag{5}$$

$$mask^{(t+1)}(v) = 1 \qquad \text{if } \|m - u\| \leq \xi \tag{6}$$

where u is a pixel on frame t and v is an estimation of u tracked to frame $t+1$ considering the forward motion vector $\boldsymbol{FMV}$. m is the estimation of v tracked back to frame t based on the backward motion vector $\boldsymbol{BMV}$. ξ is a pre-determined parameter to control the matching threshold.

(a) (b)

Fig. 2. Depth maps merging: (a) Depth copy result with undefined 'holes'. (b) 'Holes' filled by merging with bilateral filtering result.

The basic idea of this matching principle is like this: a pixel u is tracked to the next frame at pixel v using the forward motion vector and then tracked back to the original frame at pixel m using the backward motion vector. If the tracked back pixel m is in the neighborhood of the original pixel u, we can conclude that the forward and backward motion vectors form a good match, indicating the motion estimation results are reliable. Therefore, pixel u and pixel v are essentially the same pixels only to be presented on different frames and the estimated motion vectors constitute the connection between these two pixels. Besides, these two pixels should have the same depth value since they are actually the same. We can directly copy the depth of pixel u on the original frame to pixel v on the next frame if these pixels are 'connected' by motion vectors.

Because the pixels on two successive frames do not have a one-to-one correspondence and not all estimated motion vectors are able to form a reliable match, there may exist some undefined 'holes' in the depth map generated by the depth copy method. To fill up these 'holes', we employ the bilateral filtering algorithm proposed in [8]:

$$D^{(t+1)}(i) = \frac{\sum_j f(i,j) w^{(t+1,t)}(i,j) D^{(t)}(j)}{\sum_j f(i,j) w^{(t+1,t)}(i,j)} \quad \text{if } mask^{(t+1)}(i) = 0 \tag{7}$$

where $D^{(t+1)}(i)$ is the depth of pixel i on frame $t+1$ that needs to be calculated and $D^{(t)}(j)$ is the known depth of pixel j on frame t. The spatial weighting function $f(i,j)$ and the color difference weighting function $w^{(t+1,t)}(i,j)$ are defined as follows:

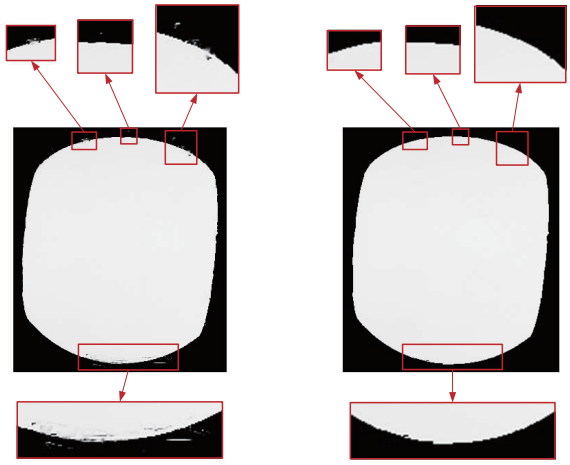
$$f(i,j) = \begin{cases} 1 & \text{if } \|i-j\| \le \Delta \\ 0 & \text{otherwise} \end{cases} \tag{8}$$

$$w^{(t+1,t)}(i,j) = e^{-\frac{\|C(i)-C(j)\|^2}{2\sigma^2}} \tag{9}$$

where Δ is the filter window size and σ is the parameter that determines color importance. $C(i)$ and $C(j)$ are the color values of pixel i on frame $t+1$ and pixel j on frame t.

It needs to be pointed out that Equation (7) is only applied to the pixels whose forward and backward motion vectors fail to match with each other. For the pixels whose motion vectors are matching, the depth copy method is sufficient to provide satisfactory depth propagation results. The reason we use the bilateral filtering algorithm is to fill up the undefined 'holes', as shown in Fig. 2. Once matching results are obtained, the depth copy method and the bilateral filtering algorithm can be executed in parallel. Then, depth maps generated by these two methods can be merged to form a complete depth map.

However, the bilateral filtering algorithm may cause several problems in depth maps [8] and a compensation procedure is needed to correct the depth errors. The compensation procedure helps to correct blended depth values and refine blurred edges in depth maps. Since these errors are mainly caused by the mismatched pixels processed by the bilateral filtering algorithm, pixels that have been handled by the depth copy method will not be compensated. Here, we propose a mismatch compensation method based on the same variable block-size motion estimation algorithm used in H.264 standard with a modification of its cost function, which takes both color information and depth values into account.



(a) (b)

Fig. 3. Mismatch compensation: (a) Depth maps merging result without compensation. (b) Mismatch compensation result.

In our proposed mismatch compensation method, we add the weighted depth differences between depth map of previous frame and depth merging result of current frame to the cost function. Now the cost function consists of color space differences and depth differences. By changing the weight of depth differences, we are able to control the compensation procedure more flexibly. Finally, depth of the pixels on the current frame that needs to be compensated can be corrected by copying the depth of their matching pixels on the previous frame, where the correct matching (*i.e.* with the smallest matching error) is decided by the new cost function. Fig. 3 illustrates the effect of mismatch compensation.

Moreover, we adopt a bi-directional depth propagation scheme considering the camera zoom in/out effect and the fact that depth of the object may vary due to the object's motion. The depth map of each non-key frame is created by combining the depth maps propagated from two adjacent key frames, as demonstrated in [12].

Once all depth maps are generated, virtual views can be synthesized through a warping process [10] according to the type of 3D display devices . After that, we integrate these views to form a stereoscopic video that is compatible with the selected 3D display devices.

3. EXPERIMENTAL RESULTS

In this section, both subjective and objective quality assessments are carried out in order to compare our method with several state-of-the-art 2D-to-3D video conversion methods, including: 1) the bilateral filtering algorithm; 2) Philip's improved depth propagation method [8]; 3) Cao's method proposed in [9][14].

The test set consists of 10 different sequences. Sequence 1–8 are collected from the Philips WowVx© project website. Sequence 9 "Interview" is published by Heinrich-Hertz-Institut and sequence 10 "InnerGate" is made by ourselves using the computer graphics method. These sequences have challenging factors such as sharp

No.	Bilateral filtering	Philip's method [8]	Cao's method [9]	Our method
1	42.59	40.91	47.46	**16.89**
2	7.76	7.55	8.51	**5.51**
3	87.13	94.83	**40.04**	41.98
4	607.15	548.94	245.50	**190.77**
5	131.40	124.75	249.98	**86.97**
6	71.18	70.01	191.53	**69.25**
7	85.70	79.78	40.58	**19.27**
8	387.67	360.47	227.29	**105.81**
9	112.32	98.93	68.23	**45.03**
10	497.47	529.96	400.77	**156.41**

Table 1. Mean Squared Error (MSE) comparison results.

No.	Bilateral filtering	Philip's method [8]	Cao's method [9]	Our method
1	36.58	38.87	35.21	**34.14**
2	40.56	38.00	17.46	**7.05**
3	35.32	41.37	42.85	**31.99**
4	28.95	39.39	38.12	**26.85**
5	45.22	41.80	43.32	**41.56**
6	44.69	**40.39**	41.84	40.46
7	39.46	32.57	35.28	**17.95**
8	**19.71**	20.08	26.15	20.83
9	31.48	21.94	22.19	**18.14**
10	31.86	36.75	33.54	**29.72**
Avg.	35.38	35.13	33.60	**26.87**

Table 2. Blind Image Quality Index (BIQI) comparison results.

edges, texture-less regions, color ambiguity and large displacement objects. The key frame interval is set to 20 to 30 frames according to different sequences. Details about the test sequences and the experimental results can be found on our website[1].

3.1. Objective assessment

Mean Squared Error (MSE) is widely used as a full-reference image quality assessment metric. In the objective assessment part, we employ the MSE metric to measure the differences between the propagated depth maps and the ground truth. Table 1 shows the MSE comparison results. We can see that our method gives the best performance in most sequences except sequence 3, in which our method finishes runner-up with a very close result.

Besides, we employ a no-reference objective assessment metric called Blind Image Quality Index (BIQI) [15] as well. BIQI generates a quality score between 0 and 100 for each depth map, where 0 represents the best quality and 100 the worst. Table 2 gives the BIQI comparison results. Similar with the MSE results, our method has the best scores in eight out of ten sequences. The average scores show that this advantage is quite obvious.

3.2. Subjective assessment

In addition to the objective assessment, a user survey is conducted on the same test set. We invited 16 viewers to watch the 3D video clips on a 3D shutter display and then rate these video clips according to their 3D performances. We put the ground truth results together with the video clips that were converted by the methods mentioned above. Table 3 presents the user survey results. We can see that the ground truth is in the lead, of course, but our method outperforms the other methods and ranks second in the survey, which verifies the effectiveness of our method.

[1]http://media.au.tsinghua.edu.cn/2Dto3D/evaluation.html

Methods	First choice	Second choice
Bilateral filtering	8.8%	13.7%
Philip's method [8]	7.5%	15.6%
Cao's method [9]	10.0%	16.3%
Our method	33.1%	28.1%
Ground Truth	40.6%	26.3%

Table 3. Subjective assessment results.

4. CONCLUSION

We have introduced a novel semi-automatic 2D-to-3D video conversion method for the purpose of converting existing 2D videos efficiently. With the help of bi-directional motion estimation, we are able to produce high-quality depth maps with limited human-machine interactions. The experimental results show that our method has better performance than several state-of-the-art 2D-to-3D conversion methods. In the future, we plan to adopt the image matting methods to generate better segmentation results and more accurate depth maps for key frames.

5. ACKNOWLEDGEMENT

This work was supported by the Key Project of NSFC (No.61035002 & 60932007) and the authors are also affiliated with Tsinghua National Laboratory for Information Science and Technology (TNList).

6. REFERENCES

[1] L. Zhang, C. Vazquez, and S. Knorr, "3D-TV content creation: Automatic 2D-to-3D video conversion," *IEEE Trans. on Broadcasting*, vol. 57, no. 2, pp. 372–383, 2011.

[2] T. Huang and A. Netravali, "Motion and structure from feature correspondences: A review," *Proc. IEEE*, vol. 82, no. 2, pp. 252–268, 1994.

[3] G. Zhang, J. Jia, T. Wong, and H. Bao, "Recovering consistent video depth maps via bundle optimization," *Proc. CVPR*, 2008.

[4] S. Knorr, E. Imre, B. Ozkalayci, A. Alatan, and T. Sikora, "A modular scheme for 2D/3D conversion of TV broadcast," *Proc. 3DPVT*, 2006.

[5] J. Zhang, Y. Yang, and Q. Dai, "A novel 2D-to-3D scheme by visual attention and occlusion analysis," *3DTV Conf.*, 2011.

[6] M. Guttmann, L. Wolf, and D. Cohen-or, "Semi-automatic stereo extraction from video footage," *Proc. ICCV*, 2009.

[7] X. Yan, Y. Yang, G. Er, and Q. Dai, "Depth map generation for 2D-to-3D conversion by limited user inputs and depth propagation," *3DTV Conf.*, 2011.

[8] C. Varekamp and B. Barenbrug, "Improved depth propagation for 2D to 3D video conversion using key-frames," *Proc. IETCVMP*, 2007.

[9] X. Cao, Z. Li, and Q. Dai, "Semi-automatic 2D-to-3D conversion using disparity propagation," *IEEE Trans. on Broadcasting*, vol. 57, no. 2, pp. 491–499, 2011.

[10] C. Fehn, "Depth-image-based rendering(DIBR), compression, and transmission for a new approach on 3DTV," *Proc. SPIE*, 2004.

[11] Y. Boykov and M. Jolly, "Interactive graph cuts for optimal boundary and region segmentation of objects in N-D images," *Proc. ICCV*, 2001.

[12] C.Wu, G. Er, X. Xie, T. Li, X. Cao and Q. Dai, "A novel method for semi-automatic 2D to 3D video conversion," *3DTV Conf.*, 2008.

[13] Y. Boykov and V. Kolmogorov, "An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision," *IEEE Trans. on PAMI*, vol. 26, no. 9, pp. 1124–1137, 2004.

[14] X. Cao, A. Bovik, Y. Wang and Q. Dai, "Converting 2D video to 3D: An efficient path to a 3D experience," *IEEE Multimedia*, vol. 18, no. 4, pp. 12–17, 2011.

[15] A. Moorthy and A. Bovik, "A two-step framework for constructing blind image quality indices," *IEEE Signal Processing Letters*, vol. 17, no. 5, pp. 513–516, 2010.