# Foreground Depth Estimation for Semi-automatic 2D-to-3D Video Conversion

*Wen-Nung Lie, Yi-Kai Chen, and Jui-Chiu Chiang*

Department of Electrical Engineering and Center for Innovative Research on Aging Society (CIRAS)
National Chung Cheng University, Chia-Yi, Taiwan, ROC.
Email: ieewnl@ccu.edu.tw

***Abstract*** **– Depth map estimation is important in 2D to 3D video conversion. Normally, the background part is static or changes slowly, while the foreground part might change substantially between consecutive frames. A good strategy is that depths for the foreground and the background parts are estimated separately and then combined together to form the final depth map. In this paper, we propose, for non-key frames, an algorithm of automatic foreground depth propagation from key frames where the foreground part is segmented and depth-assigned manually with some supporting computer tools. For each non-key frame, the foreground region is segmented independently based on the graph-cut and GMM (Gaussian Mixture Model) algorithms. The superpixel algorithm is then applied to the foreground area only for partitioning it into homogeneous patches. To propagate/compensate the foreground depths from key frames, superpixel matching (based on color component and foreground labels) is performed between each non-key frame and its reference frame, with the background parts removed. We then refine the foreground depths by using bilateral filtering. Experiments show that compared to conventional algorithms of block matching, optical flow, and superpixel, our method is advantageous of resisting large foreground motion and erroneous matching caused by background interference (similar colors). In overall, our algorithm improves the resulting foreground depth map significantly.**

***Index Terms*** — Semi-automatic, Superpixel, 2D-to-3D stereo video conversion, sprite, key frame, GMM.

## I. INTRODUCTION

In view of the vigorous development of 3DTV technology since 2010, many researches focus on the 2D-to-3D video conversion, which is capable of solving the problem of the lack of 3D video content in a more efficient way. 2D to 3D video conversion relies on accurate depth generation/estimation for each frame. It has been known that depth quality of "semi-automatic" methods will make a good tradeoff between quality and efficiency and seem to be prevailing in the future [3,5,12]. For semi-automatic processing of a series of image sequence, a small set of key-frames is chosen for manual assignment of depths, which are then automatically propagated to other non-key frames for reducing the overall production cost. Taking a set of *N* frames for example, if frame #1 and #N stand for two consecutive key frames (front and rear), the intermediate ones (#2 ~ #(N-1)) will be considered as the non-key frames.

Depth propagation can be divided into three categories: block-based [1-5], contour-based [6-7], and superpixel-based [8-9]. Block-based methods divide a full frame into non-overlapping blocks, for each of which a motion vector (MV) is estimated and the corresponding depth information is compensated from the reference frame. Quality of depth propagation is limited by the accuracy of MVs. Contour-based methods relies on the tracking of the foreground object contours assigned or extracted in the preceding key or non-key frame. They often assume uniform or constant depths within the foreground object area and do not concern about the variation of background depth profiles between consecutive frames. On the other hand, superpixel-based methods segment frames into groups of superpixels and compensate depth information from the reference frame by matching superpixels. They seem to have better performance than block-matching methods. All of the above three kinds of methods suffer from error matching caused by foregrounds' large motion and color similarity near the foreground/background boundaries, as shown in Fig.1.

Traditionally, both the foreground and background depths for key frames are manually assigned [1-4], which are then propagated to non-key frames. However, most of the background part changes slowly, while the foreground part might change substantially between consecutive frames. Recognizing this fact, foreground and background depths are treated separately in our prior works [5,12]. We adopted a strategy that foreground depths are manually assigned for key frames, while background depths are drew for a background sprite model (BSM) constructed to integrate all the background parts from all frames (key and non-key). This has the advantages of reducing human efforts in drawing background depths and eliminating mutual interferences from each other (background or foreground) in depth estimation or propagation.

In this paper, we continue the same strategy to treat the foreground and background depths separately. We will focus on the foreground depth estimation/propagation only, leaving the background depths set with fixed patterns of profile (e.g., constant or top-bottom gradient depths that were commonly used) or treated based on the BSM method proposed in [5,12]. Compared to [5,12], we adopt a modified superpixel matching algorithm for replacement of the traditional block matching. The results show that much better performance can be achieved.

## II. PROPOSED METHOD

Our foreground depth propagation algorithm is divided into two parts: (1) foreground segmentation for key frames, (2) foreground segmentation for non-key frames, and (3) foreground depth propagation from key to non-key

frames. The detailed flowchart of our algorithm is shown in Fig. 2.

(a) (b) (c)

**Fig. 1** Erroneous depth propagation caused by large motion (a) reference frame, (b) current frame (c) result of depth compensation by block matching.

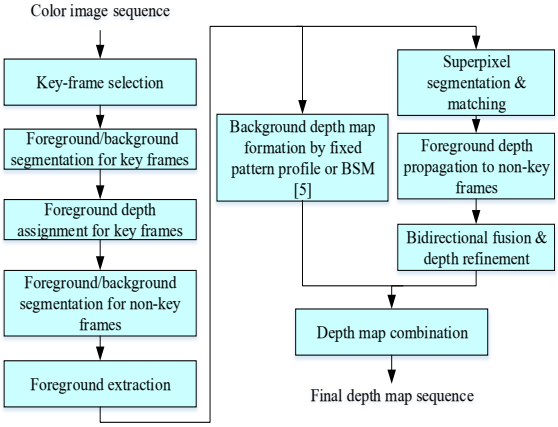
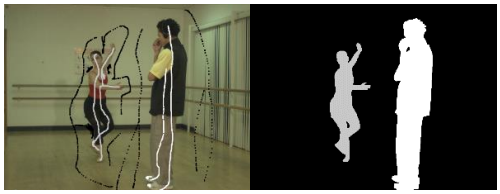



**Fig. 2** Flowchart of the proposed algorithm

### A. Foreground/background segmentation for key frames

The first step is interactively achieved by user with the aid of mouse strokes (Fig. 3(a)), followed by graph-cut algorithms [5,12] to segment out foregrounds from backgrounds. Computer tools (e.g., Photoshop) are then used to label each foreground object with a different label number (the background is with a label "0", and foreground objects are with label "1", "2",… "*K*", as shown in Fig. 3(b)).



(a) (b)

**Fig. 3** Foreground/background segmentation for key frames (a) mouse strokes of foreground (white) and background (black), (b) assigned label map.

### B. Foreground/background segmentation for non-key frames

In principle, the graph-cut algorithm can be applied similarly to extract the foreground regions for non-key frames. However, we suffer from the prohibition of user intervention (by aiding strokes) during depth estimation process for non-key frames. To cope with this problem, "kernel labels" (pixels with confident labels) [12] for both the foreground and background regions are detected in non-key frames to function similarly to user's strokes, leaving the unconfident pixels with "UND" (undetermined) label. The accuracy or correctness of the object kernels should be guaranteed so as to make the following foreground segmentation successful.

Traditional motion compensation is adopted as a step to identify the kernels: those pixels whose motion compensated residues are very small. To be more robust for identifying the kernel parts, a bi-directional (forward and backward) process from two bounding key frames is conducted, whose results are then fused together (only those consistent are identified) [12].

On the other hand, after completing key frames' label maps (Fig. 3(b)), a foreground GMM (Gaussian Mixture Model) and a background GMM can be constructed separately by using the key frames as the training data [13]. Combining the foreground object kernels and the foreground/background GMMs, the graph-cut algorithm [5] is used to segment the foreground areas for each non-key frame.

### C. Superpixel segmentation and matching

Motion estimation based on superpixel matching has been shown to be superior to that based on block matching in certain application fields. It is modified here to meet our application in foreground depth propagation. Here we adopt the state-of-the-art SLIC superpixel segmentation algorithm [10], which clusters the pixels in 5-D feature domain ($L$,*$a$, *$b$, $x$, $y$, including location and color information). Based on the foreground masks after graph-cut segmentation for non-key frames, we extract the foreground areas for superpixel segmentation, as shown in Fig. 4. By removing the background part based on the resulting masks, it is possible to filter out interferences from the similar or dis-occluded background area in the process of motion estimation and compensation.

After superpixel segmentation, superpixel matching is performed between the reference and the current frames. To be robust, the matching cost is defined as below:

(1) Costs from all superpixels between reference and current frames are calculated,

(2) Based on an extra domain from label, the matching cost is defined in a 6-D space (containing $L$,*$a$, *$b$, $x$, $y$, $d$, where $d$ is the label number) as:

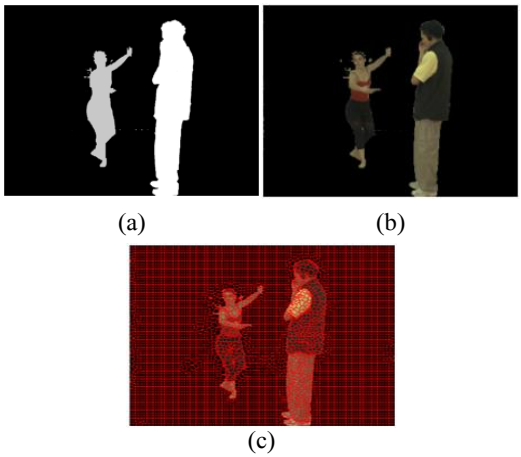
$$Cost(T_i, R_j) = \alpha * e^{Cdiff} + \beta * e^{Ediff} + Gmmdiff$$
$$1 \le i \le N, 1 \le j \le M$$

$$where\ Cdiff = \left|r_{T_i} - r_{R_j}\right| + \left|g_{T_i} - g_{R_j}\right| + \left|b_{T_i} - b_{R_j}\right|$$
$$Ediff = \left|x_{T_i} - x_{R_j}\right| + \left|y_{T_i} - y_{R_j}\right|$$
$$Gmmdiff = \begin{cases} 0 & if\ l_{T_i} = l_{R_j} \\ \infty & otherwise \end{cases}$$

$Cost(T_i, R_j)$ denotes the matching cost between the $i_{th}$ superpixel in current frame $T$ and the $j_{th}$ superpixel in

reference frame *R*. *N* is the number of superpixels in frame *T* and *M* is the number of superpixels in frame *R*; $(r_{T_i}, g_{T_i}, b_{T_i})$ and $(r_{R_j}, g_{R_j}, b_{R_j})$ stand for average RGB intensities for superpixels $T_i$ and $R_j$, respectively. $(x_{T_i}, y_{T_i})$ and $(x_{R_j}, y_{R_j})$ are coordinates of the centers of $T_i$ and $R_j$, respectively. *Gmmdiff* is used to against interference from other areas of different labels ($l_{T_i}$ *and* $l_{R_j}$ represent the labels of the region $T_i$ and $R_j$, respectively). $\alpha$ and $\beta$ are weights for normalizing importance between location, color, and label domains.

Depth compensation can be conducted after superpixel matching. Due to different sizes and shapes between the matched $T_i$ and $R_j$, the average depth value of superpixel $R_j$ is used for depth compensation of superpixel $T_i$. As the label map for foregrounds and backgrounds is considered in the matching cost, it will prevent the influence of similar-color superpixels from other foreground objects.



(a) (b)

(c)

**Fig. 4** Foreground extraction and superpixel segmentation by SLIC algorithm, (a) foreground mask, (b) foreground extraction, (c) foreground superpixel segmentation.

### D. Bidirectional depth fusion and refinement

To make the results more robust, bidirectional (forward and backward) depth propagations from the front and rear key frame, respectively, are conducted, as shown in Fig.5. The two resulting foreground depths are then fused via a temporal weighting strategy. As shown in the following equation, the weight is inversely proportional to the temporal distance between the current frame and the front and rear key frames.

$$D_t(l) = \frac{k2-t}{k2-k1} D_{k1}(l) + \frac{t-k1}{k2-k1} D_{k2}(l)$$

where *k1*, *k2*, and *t* stand for the indices of the front and rear key frame, and the current non-key frame, respectively, $D_t(l)$ is the foreground depth map of the current non-key frame *t* at location *l*. As shown in Fig.5, a bilateral filtering is applied to refine and obtain the final foreground depth map so as to suppress discontinuous depth boundaries between superpixels and make human perception comfort.

## III. EXPERIMENTS

Our test sequences include: (1) Undo Dancer (1920× 1088 pixels, 71 frames), (2) Ballet (1024×768 pixels, 61 frames), and (3) Mobile (720×540 pixels, 30 frames). Among them, "Ballet" and "Mobile" contain static backgrounds, but fast foreground motion, while "Undo Dancer" contains both foreground and camera motions. Experiments are conducted on a platform of Intel(R) Core(TM) i7-3770 3.40GHz and 8GB RAM.

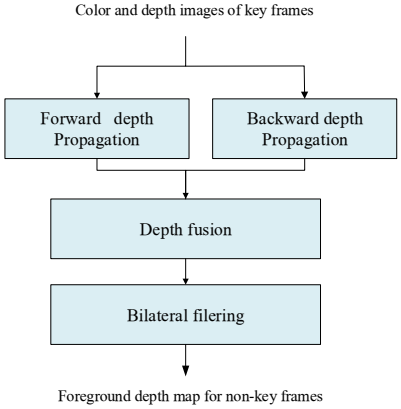



**Fig. 5** Bidirectional foreground depth propagation, fusion, and filtering.

Performance is evaluated in terms of foreground PSNR. We compare the performances of foreground depth propagation between our algorithm and the conventional ones based on block matching and optical flow. As shown in Fig. 6, conventional block matching method in [5] and [3] ((g)&(f)) cannot faithfully compensate depths from the reference frame, especially when the motion is large and colors around the foreground and background boundaries are similar. For optical flow method (Fig. 6(e)), the estimation of motion still suffers from the interferences from similar background areas. As for our proposed modified superpixel method (Fig.6(h)), owing to the spatial pre-processing to separate the foreground and background based on graph-cut and GMM, better performances can be achieved. Our modified algorithm for superpixel matching, in contrast to the traditional ones in [8][9], is capable of distinguishing multiple foreground objects of similar colors. An example in Fig. 7 shows that depth of the left hand of the dancer is influenced by the depth of the standing male due to the similarity in their skin colors. Thanks to our label domain in matching cost so as to exclude the interferences from other objects.

Another example is in Fig. 8, where the test sequence is "Undo Dancer". Due to fast motion of the dancer, the traditional method [5] cannot preserve the shape of the dancer's hands in depth map. This will be very harmful in 3D perception.

Table I shows quantitative comparison for three test sequences between [5] and our method. Our proposed method only achieves slight improvement for test sequences of less or non-deformed foreground objects (like "Mobile"). Much better performance can be achieved for

test sequences whose foreground objects have large motion or deformations (such as "Ballet" and "Undo dancer").

## IV. CONCLUSION

In this paper, we propose a modified superpixel matching algorithm, in combination with the graph-cut method for foreground segmentation, to achieve foreground depth propagation in semi-automatic 2D to 3D video conversion. With spatial segmentation between foregrounds and backgrounds, traditional superpixel matching can be enhanced to faithfully retrieve better depths from the reference frames. Experiments show that our result outperforms those traditional ones based on block matching and optical flow.

## ACKNOWLEDGEMENT

This work was supported by the Center for Innovative Research on Aging Society (CIRAS) from The Featured Areas Research Center Program within the framework of the Higher Education Sprout Project by Ministry of Education (MOE) in Taiwan.

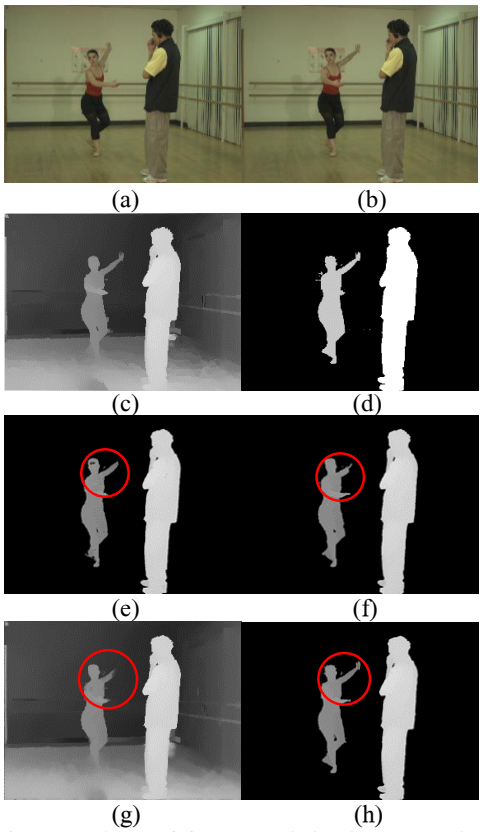



**Fig. 6** Comparison of foreground depth propagation: (a) color frame 1, (b) color frame 2, (c) depth ground truth of frame 2, (d) foreground mask for frame 2, (e) foreground depths by optical flow, (f) by [5], (g) by [3], and (h) by proposed method.


**Fig. 7** Comparison of superpixel matching (a) color image, (b) without label matching, (c) with label matching.

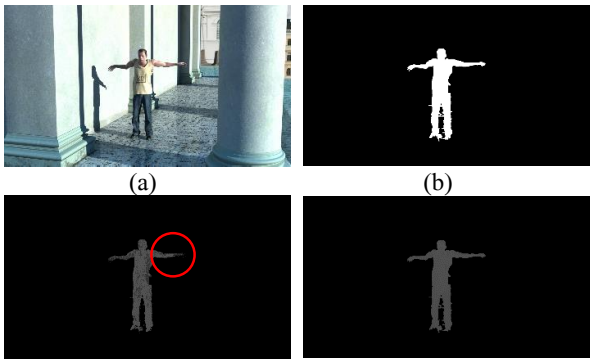



**Fig. 8** Foreground depth propagation result for "Undo Dancer". (a) Color frame 24, (b) foreground mask, (c) result by [5], (d) result by our proposed method.

**Table Ⅰ**. Comparison of results between [5] and Proposed Method

| Method | [5] | Proposed Method |
|---|---|---|
| Mobile | 26dB | 26.9dB |
| Ballet | 17dB | 22.9dB |
| Undo dancer | 20.9dB | 29dB |

## REFERENCES

[1] Xun Cao, Zheng Li, and Qionghai Dai, "Semi-automatic 2D-to-3D Conversion Using Disparity Propagation," *IEEE Trans. on Broadcasting*, vol. 57, pp.491-499, 2011.

[2] Haoqian Wang, Yushi Tian, Yongbing Zhang, "A Novel Depth Propagation Algorithm With Color Guided Motion Estimation," *Proc. of IEEE Int'l Conf. on Visual Communications and Image Processing*, pp.1-5, 2013.

[3] Guo-Shiang Lin, Jian-Fa Huang, and Wen-Nung Lie, "Key-frame-based Depth Propagation for Semi-automatic Stereoscopic Video Conversion," *Journal of Visual Communication and Image Representation*, Volume 43, pp. 127-137, February 2017.

[4] Wen-Nung Lie, Chun-Yu Chen, and Wei-Chih Chen, "2D to 3D Video Conversion with Key-frame Depth Propagation and Trilateral Filtering," *IET Electronics Letters*, Vol.47, No.5, pp.319-321, March 2011.

[5] Wen-Nung Lie, Chih-Hao Hu, Yi-Kai Chen, Jui-Chiu Chiang, "Multi-layer background sprite model for 2D-to-3D video conversion," *Proc. of Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA*

*ASC)*, pp. 270-274, Dec. 2017.

[6] C. Wu, Er Guihua, X. Xie, Tao. Li, X. Cao, and Q. Dai, "A Novel Method for Semi-automatic 2D to 3D Video Conversion," *Proc. of IEEE 3DTVCON*, pp. 65-68, May. 2008.

[7] Z. Li, X. Xie, and X. Liu, "An Efficient 2D to 3D Video Conversion Method Based on Skeleton Line Tracking," *Proc. of IEEE 3DTVCON*, pp. 1-4, Potsdam, Germany, May 2009.

[8] Cheolkon Jung; Jiji Cai, "Superpixel matching-based depth propagation for 2D-to-3D conversion with joint bilateral filtering," *Proc. of IEEE Int'l Conf. on Image Processing (ICIP)*, pp. 3515-3519, 2015.

[9] Cheolkon Jung; Jiji Cai, "Image-guided depth propagation using superpixel matching and adaptive autoregressive model," *Proc. of IEEE Int'l Conf. on Visual Communications and Image Processing (VCIP)*, pp. 1-4, 2015.

[10] A. Radhakrishna, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Susstrunk, "Slic superpixels," Technical Report 149300, EPFL, 2010.

[11] Y. Boykov, and M. P. Jolly, "Interactive graph cuts for optimal boundary and region segmentation of objects in N-D images," *Proc. of IEEE Int'l Conf. on Computer Vision*, Vol.1, Vancouver, BC, pp.105-112, July 2001.

[12] Wen-Nung Lie, Shao-Ting Chiu, Jui-Chiu Chiang, "Semi-automatic 2D-to-3D Video Conversion Based on Background Sprite Generation", *Proc. of Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, Dec. 2016.

[13] Wen-Nung Lie, Shao-Ting Chiu, Yi-Kai Chen, and Jui-Chiu Chiang, "Semi-automatic 2D-to-3D Video Conversion Based on Background Sprite Generation", revised in *IEEE Trans. on Circuits and Systems for Video Technology*, 2018.