Fast Temporal Filtering of Depth Maps

Sergey Matyunin
Moscow State University
Graphics & Media Lab
smatyunin@graphics.cs.msu.ru

Dmitriy Vatolin
Moscow State University
Graphics & Media Lab
dmitriy@graphics.cs.msu.ru

Maxim Smirnov
YUVsoft Corp.
ms@yuvsoft.com

ABSTRACT

In this paper, we propose a method of filtering of depth maps automatically generated from video sequences using optical flow, 3D reconstruction and scene analysis methods. To attain better quality, information from both source video and depth map is used. The proposed algorithm uses motion estimation to take into account temporal information, though the algorithm structure permits usage of optical flow to improve quality at the expense of greater computation time. The method can be applied as a preprocessing for result enhancement of multi-view or stereo 3D video. Further quality improvement is possible in case of joint temporal and spatial processing. A comparison with test ground truth sequences in BI-PSNR metric is presented.

Keywords

Depth map, temporal filtering, stereo, 3D, video.

1 INTRODUCTION

Depth maps are widely used for 3D video production. Their creation is a laborious process so methods of automatic generation are developed. One of the promising approaches is depth map reconstruction from object motion [KMS07]. In [SCN05] a method of spatial structure analysis based on neural network and machine learning was proposed.

A relatively simple research direction is depth from stereo reconstruction [OA05]. Nevertheless it also contains many unsolved problems. Estimation of depth from stereo is intended for parallax tuning for different types of screens and showing rooms and for nonlinear editing taking into account parallax of neighbor scenes.

The problem of definite depth reconstruction without additional information is in the general case unsolvable. For automatic depth reconstruction approaches based on local criteria minimization are applied. It leads to errors in a depth map. Such depth maps are not applicable to 3D image creation due to temporal instability and errors. A specific preprocessing is required to increase temporal and spatial stability of results. In this paper such a method of depth map processing using color and motion information is proposed.

2 RELATED WORK

Depth processing is often used to decrease noticeability of depth map errors during visualization. Modifications of gaussian blur are applied in occlusion areas. In [TZ04] authors propose asymmetric blurring: the length of the filter larger in the vertical than in the horizontal direction. They also propose to change size of the symmetric smoothing filter depending on the local values in the depth maps. An edge dependent depth filter was proposed in [CCL+05]. To increase result quality edge direction is taken into account. Artifacts are more noticeable in occlusion areas. In [LH09] a method adaptive to occlusions was proposed.

Above-listed approaches use only data from the current frame and only a part of color information from source video is used (for example, only information about object edges).

In [ZJWB09] to decrease temporal instability authors propose to solve the energy minimization problem for several consecutive frames using graph cut and belief propagation. Another approach for depth map postprocessing, proposed in [ZJWB08], is iterative refinement. For each frame the algorithm consequently refines depth maps of neighbor frames. Refinement procedure is also reduced to the energy minimization problem. Such approaches produce good results but due to computational complexity they require a lot of time to process entire video.

The proposed approach utilizes several neighbor frames for depth map refinement. Filtering is per-

formed with taking into account intensity(color) similarity of pixels and spatial distance. Information about object motion is considered by means of a motion compensation algorithm.

3 PROPOSED METHOD

The proposed algorithm uses frames of source video sequence and a depth map generated from this video. We use denotation $I_i(x)$ to represent the intensity (or color) of pixel x in frame i. It is either a 3-vector in a color image or a scalar in a grayscale image. n denotes the current frame number. $D_i(x)$ represents depth for i-th frame in position x. The proposed method consists of four steps:

1. Motion estimation (ME) between the current frame I_n and neighbor frames I_{n+d}, where $d = -m,\ldots,-1,1,\ldots,m$, $m > 0$ is a parameter. The result of this stage is a field of motion vectors $MV_{n+d}(x)$. We denote $MV_n(x) \equiv 0$.

2. Confidence metric $C_{n+d}(x)$ computation for found motion vectors $MV_{n+d}(x)$. $C_{n+d}(x) \in [0,1]$. $C_i(x)$ shows estimation quality for motion vector $MV_i(x)$.

3. Motion compensation for the depth map and source frames. We denote by D^{MC}_{n+d} motion compensated depth maps and by I^{MC}_{n+d} motion compensated source frames. We assume that $I^{MC}_n \equiv I_n$ and $D^{MC}_n \equiv D_n$.

4. Computed D^{MC}_i, $C_i(x)$ and I^{MC}_i are used for depth map filtering.

3.1 Motion Estimation

The results described in this paper were obtained using a block matching motion estimation algorithm based on one described in [SGVP08]. It operates with macroblocks of size 16×16, 8×8 and 4×4 with adaptive partitioning criteria. Motion estimation is performed with quarter pixel precision. Both luminance and chroma planes are considered.

A confidence metric is calculated for quality assessment of found motion vector field. The used metric is similar to described in [SGV08].

3.2 Depth Filtering

We use $2m+1$ consecutive frames for filtering. The first step is temporal median filtering.

$$D^{med}_n = \underset{\substack{i=n-m,\ldots,n+m\\ C_i>Th_C\\ |I^{MC}-I|<Th_{Diff}}}{med} D^{MC}_i.$$

Median is calculated using pixels of current and neighbor depth maps with sufficiently small interframe difference $|I^{MC}-I|$ and well estimated motion vectors. Median filtering is applied to eliminate sharp discontinuities in time domain.

The next step of the processing is temporal smoothing.

$$D^{smooth}_n(x) = \frac{\sum\limits_{t=n-m}^{n+m}\sum\limits_{y\in\sigma(x)} \omega(t,x,y)\cdot D^{inyut}_t(y)}{\sum\limits_{t=n-m}^{n+m}\sum\limits_{y\in\sigma(x)} \omega(t,x,y)},$$

where $\omega(t,x,y)$ is a weight function, $D^{input}_t(p)$ - is input depth at this step of processing. Source depth map D_i is used as input depth map D^{input}_i for neighbor frames and filtered D^{med}_n is used as input D^{input}_n for the current frame. It's possible to use fully processed previous depth maps D^{smooth}_{n-d} as input for previous frames processing. The latter approach gives a more smooth resulting depth map, but it is less accurate with small details. $\sigma(x)$ denotes spatial neighborhood of pixel x. Size of $\sigma(x)$ is chosen as a tradeoff between computation speed and processing quality.

Weighting function ω is given by

$$\omega(t,x,y) = f(|I^{MC}_t(y) - I_n(y)|)\cdot C_t(y)\cdot g(x,y),$$

where function f denotes the dependence on interframe difference; $C_t(y)$ is confidence of motion compensation of pixel y on frame t; g denotes dependence of weight on the spatial distance between x and y. In the simplest case g is identically constant. To achieve better quality we tested other types of dependencies between the spatial distance and weight: linear, polynomial, exponential. Function f is given by the formula

$$f(x) = max\left(0, min\left(1, \sum_{i=0}^{3}\mu_i\cdot\left(\frac{x}{\nu}\right)^i\right)\right),$$

where μ_i and ν are parameters of the algorithm.

Thereby we average depth value in the neighborhood of each pixel using information about interframe difference for the source video, the confidence metric for motion vectors and spatial proximity.

4 RESULTS

The proposed algorithm was implemented in C as a console application. The source video and its depth map is the input data for the algorithm and the filtered depth is output. Our algorithm uses one-pass processing. It can be convenient for hardware implementation. Algorithm performance is 7.6 fps on 448x372 video resolution on PC with Intel Core2Duo T6670 2.20 GHz CPU.

For objective evaluation standard sequences "cones", "venus", "sawtooth", "teddy" and "bull"

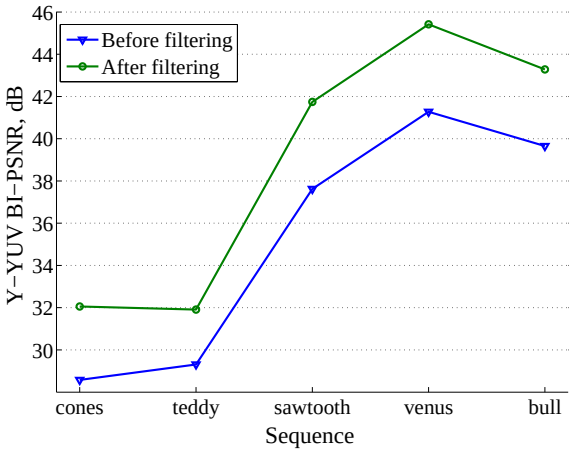


Figure 1: Results of objective quality assessment. Depth maps were compared to ground truth depth. Filtering considerably increased quality of input depth maps.

were used [SS02, SS03]. The comparison with ground truth was performed by Brightness Independent PSNR metric [VNG]. The results are presented in Fig. 1.

For subjective evaluation results of the algorithm are presented in Fig. 2.

On test sequence "road" [ZJWB09] the proposed method allowed to improve temporal stability of the depth map and recover details (see Fig. 3 and 4).

We evaluated visual quality of the rendered 3D image based on an automatically generated depth map. The rendered view is presented in Fig. 5. The proposed method allowed to reduce artifacts on the resulting rendered view and improve visual quality.

5 FURTHER WORK

In the proposed approach we don't utilize occlusion detection and processing. In occlusion areas motion estimation produces incorrect motion vectors, thus leading to artifacts. We intend to improve our confidence metric by implementation of occlusion areas processing.

It is possible to achieve better depth map quality by spatial postprocessing based on information from the source video. Such filtering can be based on the assumption that uniform areas in the source video have uniform depth.

6 CONCLUSIONS

In this paper, we described depth map filtering method. Quality evaluation was presented. Proposed algorithm improves visual quality of depth maps. It can be used to simplify manual work of 2D-to-3D video conversion. Proposed algorithm allows usage of simpler and faster methods of automatic depth map generation preserving similar quality.

ACKNOWLEDGEMENTS

This research was partially supported by the grant number 10-01-00697-a by Russian Foundation for Basic Research.

REFERENCES

[CCL+05] Wan-Yu Chen, Yu-Lin Chang, Shyh-Feng Lin, Li-Fu Ding, and Liang-Gee Chen. Efficient depth image based rendering with edge dependent depth filter and interpolation. *Multimedia and Expo, IEEE International Conference on*, 0:1314–1317, 2005.

[KMS07] D.H. Kim, D.B. Min, and K.H. Sohn. Stereoscopic video generation method using motion analysis. In *3DTV Conference*, pages 1–4, 2007.

[LH09] Sang-Beom Lee and Yo-Sung Ho. Discontinuity-adaptive depth map filtering for 3d view generation. In *IMMERSCOM '09: Proceedings of the 2nd International Conference on Immersive Telecommunications*, pages 1–6, ICST, Brussels, Belgium, Belgium, 2009. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering).

[OA05] Abhijit S. Ogale and Yiannis Aloimonos. Shape and the stereo correspondence problem. *International Journal of Computer Vision*, 65(3):147–162, 2005.

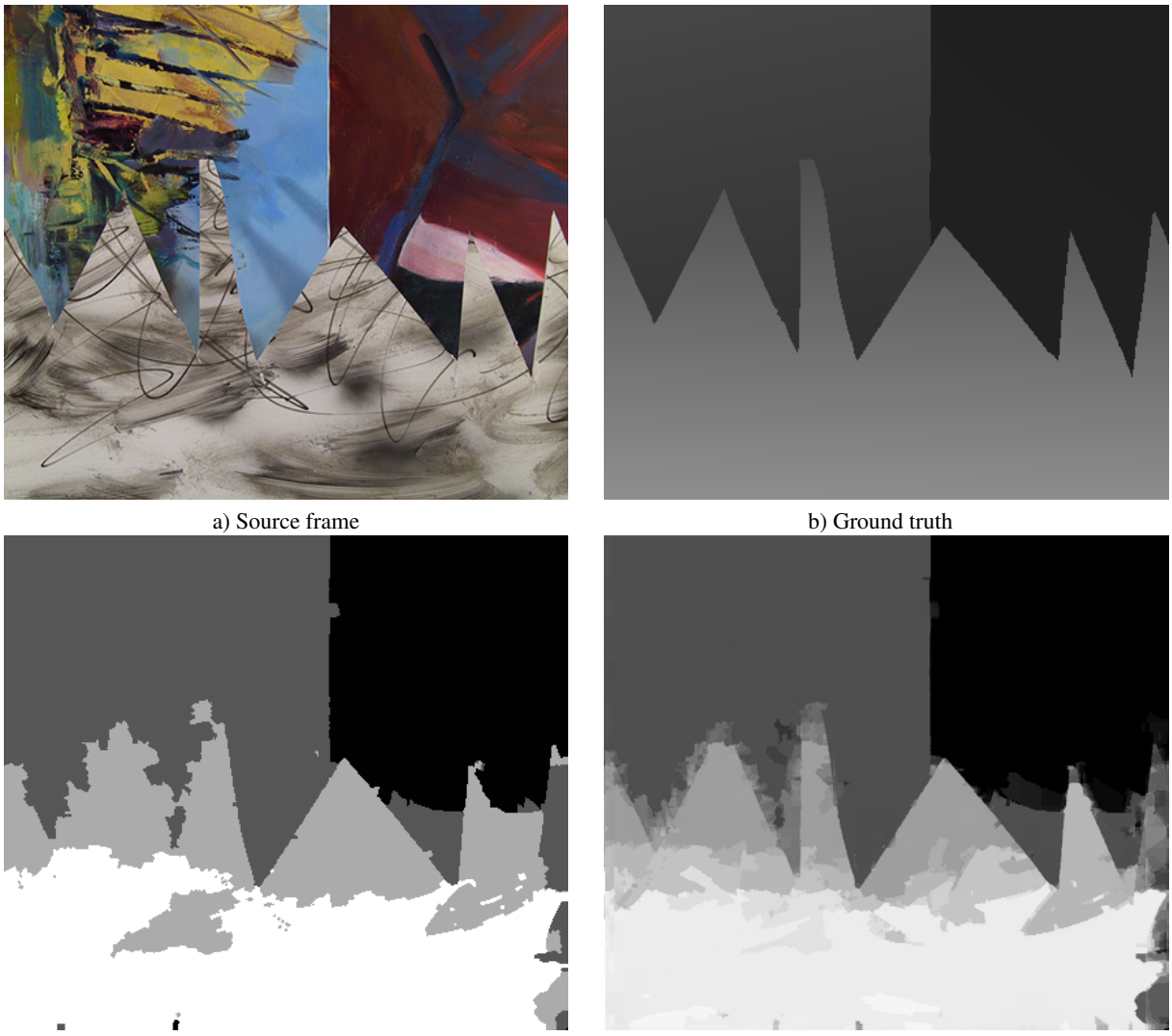
[SCN05] Ashutosh Saxena, Sung H. Chung, and Andrew Y. Ng. Learning depth from single monocular images. In *Advances in Neural Information Processing Systems 18*, pages 1161–1168. MIT Press, 2005.

[SGV08] K. Simonyan, S. Grishin, and D . Vatolin. Confidence measure for block-based motion vector field. In *GraphiCon*, pages 110–113, 2008.

[SGVP08] Karen Simonyan, Sergey Grishin, Dmitriy Vatolin, and Dmitriy Popov. Fast video super-resolution via classification. In *International Conference on Image Processing*, pages 349–352. IEEE, 2008.

[SS02] Daniel Scharstein and Richard Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *International Journal of Computer Vision*, 47(1-3):7–42, 2002.

[SS03] Daniel Scharstein and Richard Szeliski. High-accuracy stereo depth maps using structured light. *Computer Vision and*



a) Source frame

b) Ground truth

c) Estimated depth map

d) Filtered depth map

Figure 2: Sequence "Sawtooth".

Pattern Recognition, IEEE Computer Society Conference on, 1:195, 2003.

[TZ04] W. J. Tam and L. Zhang. Non-uniform smoothing of depth maps before image-based rendering. In *Proceedings of Three-Dimensional TV, Video and Display III (ITCOM'04)*, volume 5599, pages 173–183, 2004.

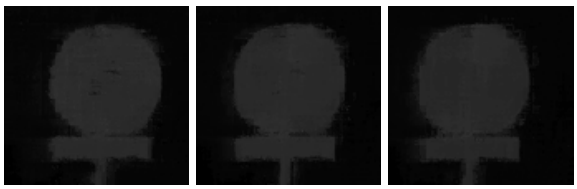
[VNG] Dmitriy Vatolin, Alexey Noskov, and Sergey Grishin. MSU Brightness Independent PSNR (BI-PSNR). `http://compression.ru/video/quality_measure/metric_plugins/bi-psnr_en.htm`.

[ZJWB08] Guofeng Zhang, Jiaya Jia, Tien-Tsin Wong, and Hujun Bao. Recovering consistent video depth maps via bundle optimization. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2008.

[ZJWB09] Guofeng Zhang, Jiaya Jia, Tien-Tsin Wong, and Hujun Bao. Consistent depth maps recovery from a video sequence. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(6):974–988, 2009.

a) Source frames

b) Depth map before filtering

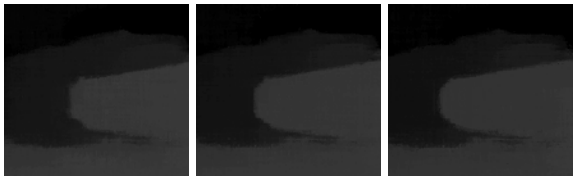


c) Depth map after filtering

Figure 3: Segment #1 of three consecutive frames of ”road” sequence.

a) Source frames

b) Depth map before filtering



c) Depth map after filtering

Figure 4: Segment #2 of three consecutive frames of ”road” sequence.

a) Original depth map

b) Filtered depth map

Figure 5: Depth based rendered view. Segment of frame 112 of ”road” sequence. View based on filtered depth seems more natural.