DIFFUSION FILTERING OF DEPTH MAPS IN STEREO VIDEO CODING

Gerhard Tech[1], *Karsten Müller*[1], *and Thomas Wiegand*[1,2]

[1]Image Processing Department
Fraunhofer Institute for Telecommunications -
Heinrich Hertz Institute
Einsteinufer 37, 10587 Berlin, Germany

[2]Image Communication Chair
Department of Telecommunication Systems
Technical University of Berlin
Einsteinufer 17, 10587 Berlin, Germany

ABSTRACT

A method for removing irrelevant information from depth maps in Video plus Depth coding is presented. The depth map is filtered in several iterations using a diffusional approach. In each iteration smoothing is carried out in local sample neighborhoods considering the distortion introduced to a rendered view. Smoothing is only applied when the rendered view is not affected. Therefore irrelevant edges and features in the depth map can be damped while the quality of the rendered view is retained. The processed depth maps can be coded at a reduced rate compared to unaltered data. Coding experiments show gains up to 0.5dB for the rendered view at the same bit rate.

Index Terms— diffusion filtering, noise reduction, stereo video coding, video plus depth coding

1. INTRODUCTION

For the transmission of stereo video several data formats and coding methods have been proposed [1]. One of the data formats is the Video plus Depth approach. To obtain a stereo pair a second view is synthesized by depth image based rendering (DIBR). For this the samples of the video are warped using disparities calculated from the depth map. The Video plus Depth approach features two advantages: One benefit is the increased compressibility of depth data. The other benefit is the flexibility to render views with variable baseline.

A drawback of the Video plus Depth approach are artifacts arising in the rendering process. This especially occurs in areas that become disoccluded in the rendering process. To overcome this problem advanced Video plus Depth approaches like Multi View plus Depth (MVD) or Layered Depth Video (LDV) have been proposed. Nevertheless, the pure Video plus Depth approach has shown its practicability in large scale subjective evaluations using small (mobile) display sizes [2].

This paper presents a method for a further improvement of the Video plus Depth approach for stereo video. The depth map is optimized regarding the synthesis of a second view in stereo distance. However, an extension of the approach to multiple output views is straight forward. The basic idea of the proposed method is that some signal parts of depth map created by a depth estimation algorithm are irrelevant for rendering. A removal or damping of these high frequency parts will increase coding efficiency and lead to an improved overall quality. Therefore the proposed algorithm applies a diffusion process to the depth data considering the distortion introduced to a rendered view. Diffusion filtering has been proposed by Perona and Malik [3] and has already been applied to depth maps ([4],[5]). In contrast to the proposed method the approaches presented in [4] and [5] use edge information from the video for depth map enhancement.

The proposed approach and its single steps are presented in section 2. An evaluation of the approach is given in section 3. Finally section 4 provides the conclusion and an outlook.

2. PROPOSED METHOD

The main concept of the proposed approach is the smoothing of the depth map in small steps and multiple iterations. In each iteration all samples of a frame are processed consecutively. The smoothing applied to a sample is controlled by the error introduced to the rendered view, as depicted in figure 1. Here x and y denote the coordinates of a sample in the frame and τ represents the iteration number.

An iteration of the proposed method starts with the calculation of a depth map with smoothed depth values candidates $\tilde{s}_D(x, y, \tau)$ from the input depth map $s_D(x, y, \tau)$ using a diffusional approach. Subsequently, all samples are processed successively to evaluate the obtained depth value candidates. The order in which the samples are processed has an influence on the filtering result. To minimize this influence the order is permuted for each iteration.

The depth map representing the current state of the processing is denoted $\hat{s}_D(x, y, \tau)$. First $\hat{s}_D(x, y, \tau)$ is initialized with $s_D(x, y, \tau)$. While processing $\hat{s}_D(x, y, \tau) = s_D(x, y, \tau + 1)$ is true for samples at positions (x, y) that already have been processed in iteration τ and $\hat{s}_D(x, y, \tau) = s_D(x, y, \tau)$ is true for samples that have not been processed. At the end of iteration τ $\hat{s}_D(x, y, \tau)$ is equal to $s_D(x, y, \tau+1)$ for all (x, y).

978-1-4244-7135-5/10/$26.00 ©2010 IEEE

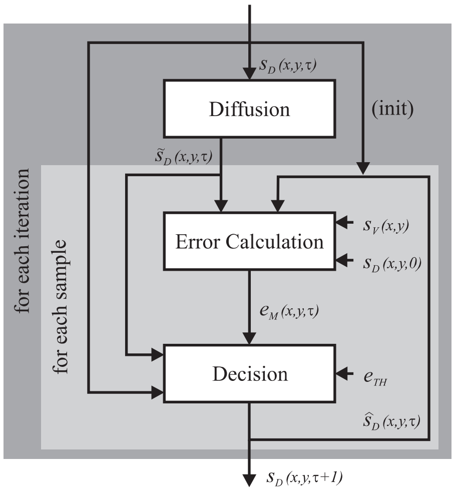

Fig. 1. Iteration steps of the proposed method; Subsequent to the calculation of smoothed candidate values $\tilde{s}$ each sample is evaluated to determine if its candidate value is used in the processed depth map $\hat{s}$.

The decision if a depth candidate at position (x_c, y_c) is accepted is based on the error $e_M(x_c, y_c, \tau)$ introduced to the view rendered from $\hat{s}_D(x, y, \tau)$ when changing $\hat{s}_D(x_c, y_c, \tau)$ from $s_D(x_c, y_c, \tau)$ to $\tilde{s}_D(x_c, y_c, \tau)$. If the introduced error is below a threshold e_{TH} the candidate value is accepted $\hat{s}_D(x_c, y_c, \tau) = \tilde{s}_D(x_c, y_c, \tau)$. Otherwise the sample remains unchanged $\hat{s}_D(x_c, y_c, \tau) = s_D(x_c, y_c, \tau)$.

The iterative filtering process can be terminated when the filter output converges. Experiments show that approx. 100 iterations are enough to obtain a good smoothing result.

2.1. Diffusion Filtering

Smoothing is carried out using an approach similar to the diffusion process proposed by Perona and Malik in [3]. For the proposed method the diffusion process is modified to

$$\begin{aligned}\tilde{s}_D(x,y,\tau) = &\quad s_D(x,y,\tau) + s_\Delta(x,y,\tau) \qquad (1)\\ s_\Delta(x,y,\tau) = &\begin{cases} +s_q & \text{if } 1/4 \cdot \Delta(s_D(x,y,\tau)) \geq s_q/2 \\ -s_q & \text{if } 1/4 \cdot \Delta(s_D(x,y,\tau)) \leq -s_q/2 \\ 0 & \text{else} \end{cases}\end{aligned}$$

with Δ denoting the 4-nearest-neighbors discrete 2D-laplacian operator and s_q denoting the quantization step size of the depth data. Thus the depth value of a sample converges to the mean of its horizontal and vertical neighbor samples with step size s_q.

2.2. Error Calculation

The error $e_M(x_c, y_c, \tau)$ introduced by a change of a depth value from $\hat{s}_D(x_c, y_c, \tau)$ to $\tilde{s}_D(x_c, y_c, \tau)$ at sample position (x_c, y_c) is estimated as follows:

- Create a depth map $\bar{s}_D(x, y, \tau)$ with

$$\bar{s}_D(x,y,\tau) = \begin{cases} \tilde{s}_D(x_c,y_c,\tau), & \text{if } (x,y) = (x_c,y_c) \\ \hat{s}_D(x,y,\tau), & \text{else} \end{cases} \qquad (2)$$

  Hence only the value of the sample under evaluation at position (x_c, y_c) is changed whilst all other depth samples retain their current values.

- Render the output view $s_R(x, y, 0)$ using $s_D(x, y, 0)$ and $s_V(x, y)$. $s_V(x, y)$ denotes the input video data. $s_R(x, y, 0)$ is the reference view. Rendering of this view must only be carried out once.

- Render the output view $\bar{s}_R(x, y, \tau)$ using $\bar{s}_D(x, y, \tau)$ and $s_V(x, y)$. Rendering using the altered depth map $\bar{s}_D(x, y, \tau)$ must be carried out for each sample and iteration. Nevertheless computational complexity is low since only image parts influenced by the sample at position (x_c, y_c) must be re-rendered.

- Set $e_M(x_c, y_c, \tau) = \max((s_R(x, y, 0) - \bar{s}_R(x, y, \tau))^2)$. $e_M(x_c, y_c, \tau)$ is the maximum squared error between the image $\bar{s}_R(x, y, \tau)$ rendered from processed depth data and the reference image $s_R(x, y, 0)$.

The proposed approach uses a simple rendering method. This method shifts the samples of the view using the disparity calculated from the depth values and interpolates the sample values at positions of the target grid. Disocclusions are filled using a straight forward line wise extrapolation of the boundary background sample value.

$s_R(x, y, 0)$ as well as $\bar{s}_R(x, y, \tau)$ are rendered using the coded video $s_V(x, y)$. This approach enables a stronger smoothing of depth data, since details and noise removed from the video data by coding are neglected when calculating the error introduced by the modified depth map.

2.3. Decision Step

In the decision step the introduced error $e_M(x_c, y_c, \tau)$ is compared to a given threshold e_{TH}. e_{TH} determines the maximal allowed error for a sample in the rendered view $\hat{s}_R(x, y, \tau)$. If $e_M(x_c, y_c, \tau)$ is higher than e_{TH} the diffusion step is rejected. This is summarized in equation (3).

$$\hat{s}_D(x_c,y_c,\tau) = \begin{cases} \tilde{s}_D(x_c,y_c,\tau), & \text{if } e_M(x_c,y_c,\tau) \leq e_{TH} \\ s_D(x_c,y_c,\tau), & \text{else} \end{cases} \qquad (3)$$

In this scope only the removal of irrelevant information from the depth data is targeted. Hence the threshold e_{TH} is set to 0. A higher threshold enables stronger smoothing but also leads to an impaired rendered view.

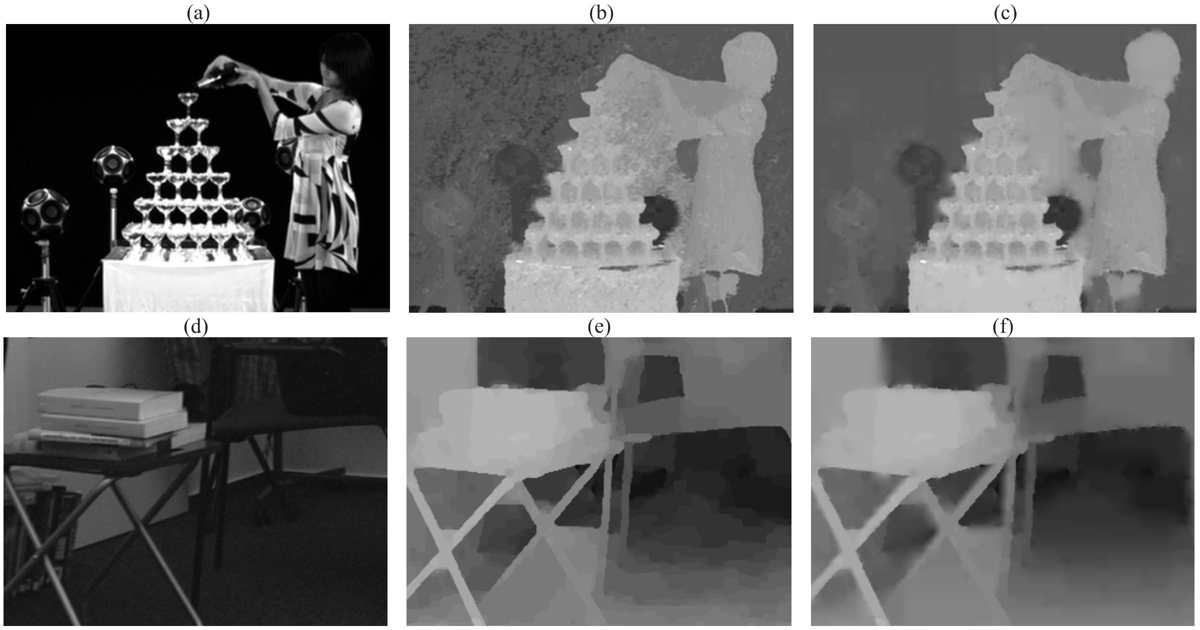

Fig. 2. Sequences *Champagne Tower* (top) and *Book Arrival* (bottom); (a), (d): Video Data; (b), (e) unprocessed depth; (c), (f): processed depth; Note that rendering using the unprocessed and processed depth provides an identical result, since only irrelevant signal parts have been reduced.

2.4. Diffusion in occluded regions

When rendering towards only one direction to generate a second view for stereo video, the value of depth samples belonging to occluded regions are not important as long as the samples stay in the background. Hence a change of these samples will result in an error $e_M(x, y, \tau) = 0$ and a strong smoothing of depth data next to a foreground objects edge occurs.

Although this smoothed area does not impair edges in the rendered view for the uncoded case, it was found that impairments can occur after coding. This is caused by the block-partitioning applied in rate-distortion optimization process carried out by the encoder. For an edge smoothed to one side usually a large block size is chosen, while for sharp edges a large block is further subdivided. In some cases the depth value of important foreground samples is better preserved in a small block in the subsequent transform and quantization steps. To avoid impairment by the changed block partitioning smoothed sample values can be rejected for all occluded samples in the decision step.

3. EVALUATION OF RESULTS

Figure 2 shows results of the proposed diffusion filter. A frame from the sequence *Champagne Tower* is depicted in Figure 2 (a). The sequence is downscaled to a size of 320x240 samples what is typical for e.g. mobile 3D TV. The corresponding unprocessed depth map is shown in figure 2 (b). It can be seen that the depth in the background is very noisy. Although this noise is irrelevant for rendering and does not affect the rendering process, it leads to higher data rates when it is compressed with a conventional encoder. The depth map processed with the proposed algorithm is presented in 2 (c). Here an error threshold of $e_{TH} = 0$ has been used, thus rendering with the processed and unprocessed depth data results in the same synthesized view. Nevertheless the noise in the background and also on the table in the foreground is removed, whilst edges in depth map that are important for correct rendering are retained.

A region clipped from the sequence *Book Arrival* can be seen in figure 2 (d). The full-sized sequence has a resolution of 1024x768 pixels. The unprocessed depth data shown in figure 2 (e) is currently used in MPEG exploration experiments [6]. For processing the threshold has been set to $e_{TH} = 0$. It can be seen that irrelevant edges are removed by the proposed method. In contrast to the results shown for *Champagne Tower* diffusion in occluded regions has been enabled. This can be observed on edges of the left side of foreground objects, where diffusion to the left side has been carried out.

To evaluate the impact of the proposed method on compression efficiency coding experiments have been carried out. The video and depth data of sequences *Champagne Tower* and *Book Arrival* have been coded using the H.264/AVC Reference Software JM. The encoder has been configured to use main profile with hierarchical B-pictures, a GOP size of 8 and an intra period of 16. The depth data has been filtered with the proposed approach using the video data coded with a QP of 30 for generation of the rendered reference view. For *Cham-*

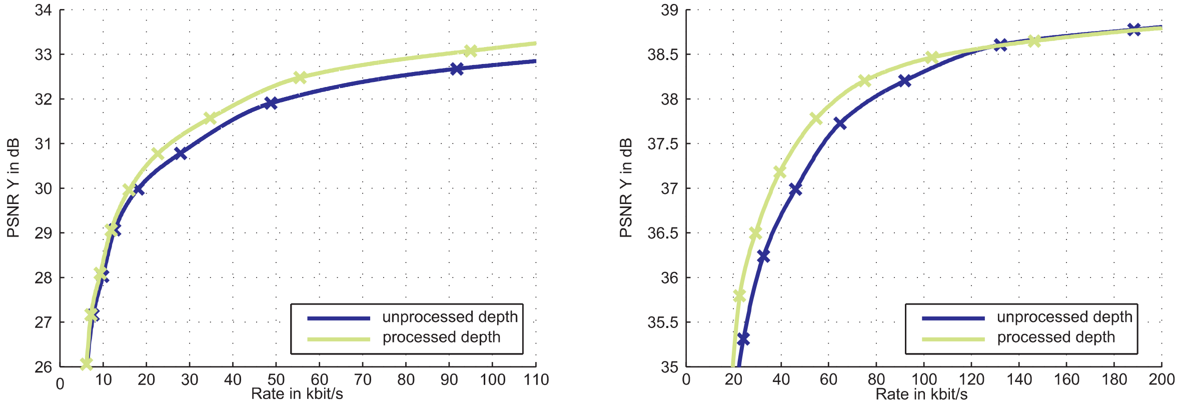

Fig. 3. Coding results for sequences *Champagne Tower* (a) and *Book Arrival* (b); PSNR Y of the rendered view vs. bit rate of the depth map. The view rendered from uncoded and unprocessed data is used as reference.

pagne Tower diffusion in occluded regions was disabled for *Book Arrival* not.

Then the processed and unprocessed depth maps have been coded. The views rendered from the coded video and coded unprocessed and coded processed depth have been compared to the view rendered from uncoded and unprocessed video and depth data. The results are depicted in figure 3. Here the PSNR obtained by this comparison is plotted versus the bit rate used for depth data. Note that the maximal PSNR is also limited by the impairment caused by the coded texture. It can be seen that gains up to 0.5dB can be achieved with the proposed method.

4. CONCLUSION AND OUTLOOK

A diffusion algorithm for the enhancement of depth maps in Video plus Depth coding has been presented. The diffusion process is controlled by the distortion introduced in the rendered view regarding the rendering algorithm and the coded video data. Hence only irrelevant high frequency parts are damped. Resulting depth maps can be coded at lower bit rates while providing the same quality in the rendering process. The applicability of the approach has been demonstrated for two sequences. PSNR gains up to 0.5dB have been shown using a view rendered from uncoded and unprocessed data as reference.

The proposed approach can be advanced in several ways: An optimization and evaluation using original views instead of rendered views as reference promises higher coding gains as presented here, since not only signal parts irrelevant for rendering but also signal parts introducing noise to the rendered view can be reduced.

Possible extensions regarding the diffusion process are anisotropic diffusion filtering and diffusion in temporal direction. Moreover an adaptation to Multi View plus Depth data (MVD) is imaginable.

5. ACKNOWLEDGEMENTS

This work was supported by EC within FP7 (Grant 216503 with the acronym MOBILE3DTV). We would like to thank Nagoya University, Tanimoto Lab. for providing the *Champagne Tower* sequence and LG Electronics for providing depth data for the *Book Arrival* sequence.

6. REFERENCES

[1] A. Smolic, K. Mueller, P. Merkle, P. Kauff, and T. Wiegand, “An overview of available and emerging 3D video formats and depth enhanced stereo as efficient generic solution,” in *Proceedings of the 27th conference on PCS*, Piscataway, NJ, USA, 2009, pp. 389–392.

[2] D. Strohmeier and G. Tech, “Sharp, bright, three-dimensional: open profiling of quality for mobile 3DTV coding methods,” in *SPIE Conference Series*, Feb. 2010, vol. 7542.

[3] P. Perona and J. Malik, “Scale-space and edge detection using anisotropic diffusion,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12, pp. 629–639, 1990.

[4] A. Banno and K. Ikeuchi, “Disparity map refinement and 3D surface smoothing via directed anisotropic diffusion,” in *3-D Digital Imaging and Modeling*, 2009, pp. 1870–1877.

[5] M. Mabaar and J.P. Siebert, “Smoothing disparity maps using intensity-edge guided anisotropic diffusion,” in *Medical Image Understanding and Analysis 2008, University of Dundee, Dundee, Scotland.*, 2008, pp. 49–53.

[6] “Description of Exploration Experiments in 3D video coding (MPEG/N11274),” *ISO/IEC JTC1/SC29/WG11*, 2010.