**INTERNATIONAL ORGANISATION FOR STANDARDISATION**
**ORGANISATION INTERNATIONALE DE NORMALISATION**
**ISO/IEC JTC1/SC29/WG11**
**CODING OF MOVING PICTURES AND AUDIO**

**ISO/IEC JTC1/SC29/WG11 MPEG2015/M37233**
**October 2015, Geneva, Switzerland**

**Source** **Poznań University of Technology,**
**Chair of Multimedia Telecommunications and Microelectronics, Poznań, Poland**
**Status** **Input Document (MPEG-FTV)**
**Title** **[FTV] Depth estimation with enhanced temporal consistency**
**Author** **Olgierd Stankiewicz** (ostank@multimedia.edu.pl),
Krzysztof Wegner, Marek Domański

# 1 Abstract

This document presents an analysis of depth estimation algorithm from the perspective of attained temporal consistency. Also, the new motion-compensated depth estimation is proposed.

# 2 Introduction

The straight-forward method for depth estimation in video sequences is to estimate depth map for each frame independently. Such an approach is simple and also allows for parallel generation of depth map in in consecutive frames. Unfortunately, independent estimation of depth in each frame in video results in depth maps which are not temporally consistent. This manifests as random fluctuation of depth values, even for objects that are still. Such fluctuations are equivalent to chaotic movements of the pixels. Desired temporal consistency of depth map means that depth changes are correlated with actual physical motion of the objects, and do not vary from frame to frame in a random way. Therefore, one of the most significant challenges in this research area is how to provide depth maps that are consistent in time.

In the past, there were various approaches to improve temporal consistency of estimated depth by making modifications of the algorithm implemented in MPEG Depth Estimation Reference Software (DERS) [1], like in example in [2] and [3].

On the other hand, in work [4] we have proposed to improve temporal consistency of the estimated depth "outside" the depth estimation itself. It was proposed to perform a noise reduction

on the input video prior to the depth estimation, so that later, the depth estimation algorithm works on denoised data. A very simple noise removal technique was used, in which not-moving regions are low-pass filtered in time. Therefore this is called "**Still Background Noise Reduction" (SBNR**) here.

The above-mentioned techniques do not use motion estimation and compensation, which lowers their performance in the case of sequences with big amount of motion, e.g. with moving camera. In this paper we want to overcome this limitation. In particular, we provide results of depth estimation with improved temporal consistency based on noise reduction in the input video, but (despite the approach in SBNR) with the use of more advanced denoising technique with motion estimation. It will be called "**Motion-Compensated Noise Reduction with Refinement" (MCNRR).**

Apart from denoising of the input video, we consider a totally different approach, based on custom initialization of graph cuts algorithm. In that context we show results of depth estimation with graph-cuts initialization that is done basing on the previously estimated depth (for previous frames). Such will be called "**Graph Cuts Initialization" (GCI).** Also we consider motion-compensated variant of such algorithm, called here "**Motion-Compensated Graph Cuts Initialization" (MCGCI).**

Moreover, in order to evaluate the amount of temporal consistency, we show some proposals of objective temporal consistency measures: based on correlation coefficient and compression ratio.

## 3 Conditions of experiments

The evaluation of the depth estimation schemes with noise reduction has been done indirectly, through assessment of quality of synthesized virtual views (Fig. 1). For view synthesis we have used commonly known MPEG View Synthesis Reference Software (VSRS). It has been configured so that is uses depth maps from two side-views (left and right) and synthesizes the center view. Therefore, depth estimation is performed for both of the side-views.

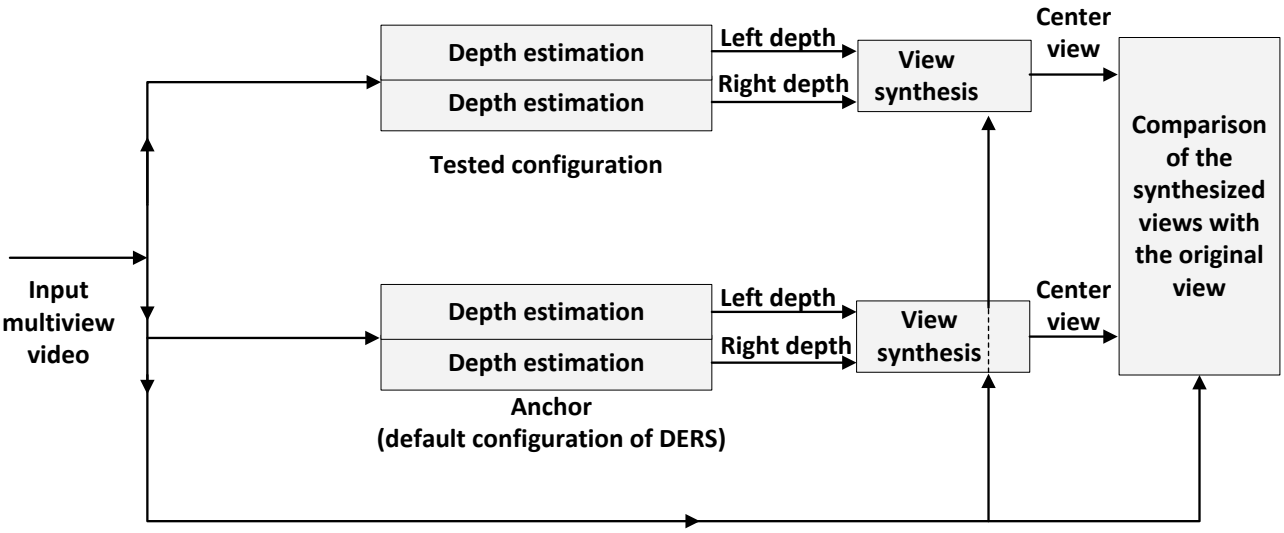


Fig 1. The scheme of quality evaluation in the work.

Table 1. Set of sequences used in the experiments.

| Sequence | Resolution | Left view | Right view | Center (original/ /synthesized view) |
|---|---|---|---|---|
| Kendo | 1024x768 | 3 | 5 | 4 |
| Balloons | 1024x768 | 3 | 5 | 4 |
| Newspaper | 1024x768 | 4 | 6 | 5 |
| Poznan Carpark | 1920x1080 | 3 | 5 | 4 |
| Poznan Hall 2 | 1920x1080 | 5 | 7 | 6 |
| Poznan Street | 1920x1080 | 3 | 5 | 4 |

The used sequences are summarized in Table 1. For all of the them, the depth estimation has been performed, always with the same common configuration of DERS: DepthEstimationMode=0. MatchingMethod=0, MatchingBlock=1, Precision=1, SearchLevel=-1, TemporalEnhancement=0, ImageSegmentation=0.

Various "Smoothing coefficient" values (in range from 1.0 to 4.0) were tested, and always the best results for the best smoothing coefficient are presented.

# 4 Measures of temporal consistency

The methodology described in point 3 is used to estimate the quality of the depth maps indirectly, through measurement of the quality of the synthesized views. However, it does not provide measurement of improvement of temporal consistency, which cannot be measured if frames are treated independently in time from each other (like in the case of PSNR).

## 4.1 Pearson Correlation Coefficient

The first approach for formulation of temporal consistency measure, which we have tested, is based on Pearson Correlation Coefficient (PCC). For datasets $a_i$ and $b_i$, PCC is given by equation:

$$\mathrm{PCC}_{a,b} = \frac{\sum_i (a_i - \bar{a}) \cdot (b_i - \bar{b})}{\sqrt{\sum_i (a_i - \bar{a})^2} \sqrt{\sum_i (b_i - \bar{b})^2}}$$

Where $\bar{a}$ and $\bar{b}$ denote expected values of a and b, respectively.

In our case, we calculate PCC between depth values at the same collocated positions in the image $(x, y)$, in successive frames k and $k-1$. Those can be denoted as $d_i(x,y)$ and $d_{i-1}(x,y)$, which we can substitute for a and b in the equation above, respectively. For all such pairs of such frames, this can be simplified as:

$$\mathrm{PCC}_{d_k(x,y),d_{k-1}(x,y)} = \frac{\sum_i \left(d_i(x,y) - \overline{d(x,y)}\right) \cdot \left(d_{i-1}(x,y) - \overline{d(x,y)}\right)}{\sum_i \left(d(x,y) - \overline{d(x,y)}\right)^2}$$

Such PCC value is then averaged over all pixel locations $(x, y)$. Therefore we get:

$$\mathrm{PCC} = \frac{1}{W \cdot H} \sum_x \sum_y \frac{\sum_i \left(d_i(x,y) - \overline{d(x,y)}\right) \cdot \left(d_{i-1}(x,y) - \overline{d(x,y)}\right)}{\sum_i \left(d(x,y) - \overline{d(x,y)}\right)^2}$$

Where $W$ and $H$ are width and height of the depth image, respectively.

Such averaged Perrson Correlation Coefficient PCC, attained in various depth estimation experiments can be compared, in order to give a measure of temporal consistency: the more correlated are collocated depth values in subsequent frames, the more temporally consistent is the analyzed depth video.

## 4.2 Video coding of the depth data

The second approach, which we have used for objective measurement of temporal consistency enhancement, employs video coding of the depth data. The estimated depth maps, resulting from experiments described above, have been coded with the use of MVC video codec. We have chosen MVC because we wanted to use a codec as simple as possible, having at the same time the ability to compress multiview video with the use of motion compensation.

In order to measure the gains/losses in compression performance, we have decided to use Bjøntegaard deltas [7], bitrate savings in particular. For those we needed a set of at least 4 coding rate-points for each tested case. We have decided to use Common Test Conditions (CTC) [8] which were used during core experiments in development of AVC-based 3D extensions. Therefore, the QP values were: 30, 35, 40, 45 (for full-resolution of depth).

# 5 Noise reduction in the input video

In this experiment three scenarios have been considered:

- The input video is denoised with the use of SBNR (Still Background Noise Reduction") technique.
- The input video is denoised with the use of MCNRR (Motion Compensated Noise Reduction with Refinement) technique.
- No noise reduction on the input video. This is anchor for the experiment.

The rough idea of MCNRR technique is to perform noise reduction in the input video with the use of motion estimation and compensation.

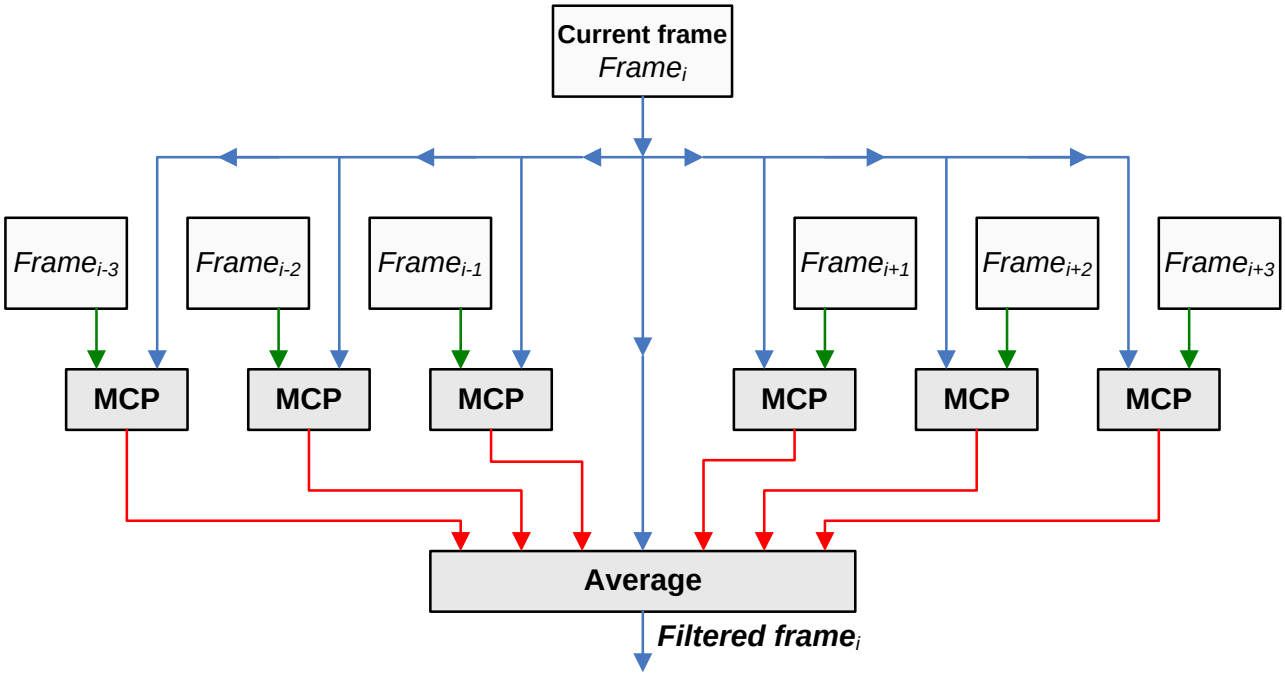


Figure 2. The core of Motion-Compensated Noise Reduction (MCNRR) algorithm.

The image is processed in overlapping blocks of size 4×4 pixels. For each processed block in the current frame, motion vectors are sought for 3 previous and 3 following frames, independently in each view. In implementation, for that purpose we have used "mv-tools" library [5]. The motion-compensated blocks from the neighboring frames (calculated in MCP blocks in Fig. 2) are then compared with the processed block in the current frame. The blocks that are classified to be similar enough (using the Sum of Squared Differences criterion) are averaged in order to generate denoised (low-pass-filtered) block. Therefore, the average may be calculated from as few as 1 block (only from the current frame) and from as many as 7 blocks (the current frame, 3 previous and 3 following frames). The detailed description of MCNRR algorithm can be found in [6].

The tested scenario is depicted in Fig. 3. For the he comparison, methodology shown in from Fig. 1 has been used. The results are presented in Table 2.

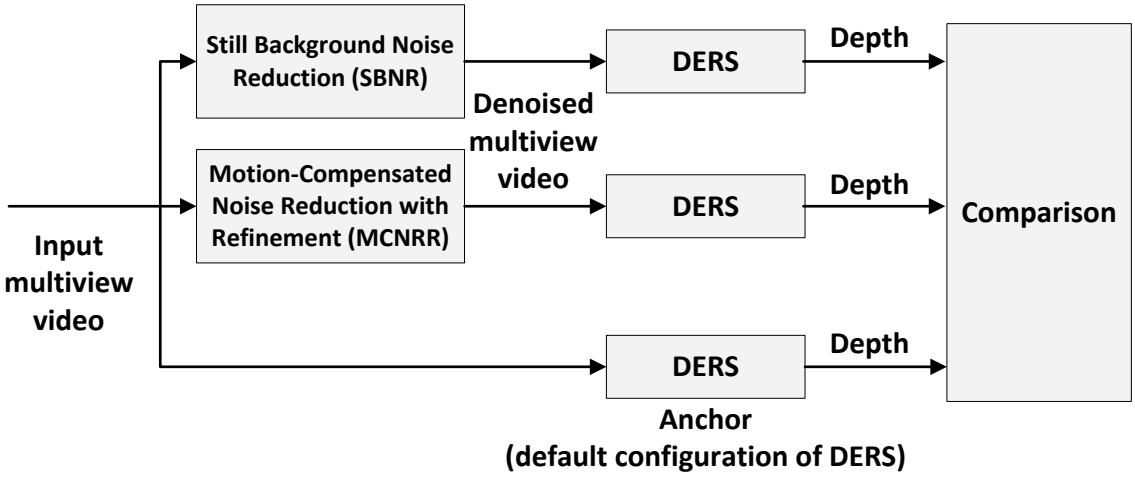


Figure 3. Comparison of depth estimation with the use of SBNR and MCNRR denoising techniques, versus original, unmodified DERS.
For the he comparison, methodology shown in from Fig. 1 has been used.

Table 2. Comparison of quality of the considered techniques of depth estimation with noise reduction, related to the original (unmodified) DERS technique, based on PSNR of view synthesis.

| Sequence Name | a) PSNR [dB] (vs. the original view ) of the virtual view synthesized with use of depth maps estimated basing on: | | | b) Relative change of correlation coefficient, related to the PCC of the anchor (the original, unmodified DERS) [%] | |
|---|---|---|---|---|---|
| | Views denoised with: | | Anchor | SBNR | Proposed MCNRR |
| | SBNR | MCNRR | | | |
| Poznan Street | 31.93 | 31.92 | 31.98 | +0.06 | +0.10 |
| Poznan Carpark | 30.74 | 30.79 | 30.71 | +0.99 | +1.64 |
| Poznan Hall 2 | 32.78 | 32.83 | 32.85 | +0.35 | +1.02 |
| Newspaper | 31.90 | 31.91 | 31.91 | +0.23 | +0.26 |
| Balloons | 32.91 | 32.93 | 32.94 | +1.74 | +1.81 |
| Kendo | 35.41 | 35.39 | 35.46 | +1.12 | +0.17 |
| Average | 32.61 | 32.63 | 32.64 | +0.75 | +0.83 |

As it can be seen, the PSNR ratios are not much changed by usage of noise reduction in the input video (Table 2a). This is not surprising, because PSNR measure is not designed to assess quality of temporal consistency.

The results presented in Table 2b show that application of the proposed noise removal techniques for depth estimation provide gains in a form of increase of correlation between subsequent depth frames in given view. The Pearson Correlation Coefficient, averaged over all

frames and views, has been compared in the cases of depth estimation: the anchor (no noise reduction, not modified original DERS), the usage of SBNR on the input video and the usage of MCNRR. For the sake of brevity, only percentage changes, related to the anchor case, are presented in Table 2b.

It can be seen that although the gains in linear correlation coefficient increase are small (up to 1,81%, about 0.06% - 1.99% in average) it must be taken into perspective that the improved regions are mostly edges of the objects that cover only a small portion of the whole scene and sometimes, differences even between the ground truth are very small – e.g. Newspaper sequence which is already highly correlated (the most the scene consists in still background).

The more interesting results are presented in Table 3. It can be seen that application of the considered noise reduction techniques on the input video have seriously influenced the estimated depth maps, because their compression ratio has vastly changed. The coding performance of such (compared to the original depth maps estimated with modified DERS basing on the original multiview video) on average is 27.02% higher in the case of the prior or on average 28.34% higher in the case of the proposed MCNRR algorithm.

Table 3. Bjøntegaard bitrate savings - results of MVC compression of depth maps estimated with use of DERS basing on denoised test sequences, related to compression of depth maps estimated with use of DERS basing on the original test sequences (anchor).

| Sequence name | Bit-rate savings [%] | |
|---|---|---|
| | SBNR technique | MCNRR technique |
| Poznan Street | 31.47 | 35.14 |
| Poznan Carpark | 46.57 | 45.19 |
| Poznan Hall 2 | 26.44 | 29.01 |
| Newspaper | 33.64 | 33.42 |
| Balloons | 23.96 | 21.99 |
| Kendo | 0.02 | 5.26 |
| Average | 27.02 | 28.34 |

In general it can be said that the average compression performance gain over the tested set is about 28% of depth bitrate reduction, while providing the same quality of synthesized views (the bitrate reduction has been measured with Bjøntegaard metric over PSNR of synthesized views). This provides a strong indication that the temporal consistency of the estimated depth has been vastly improved, because one of the main compression tools in coding technology implemented in

MVC is temporal prediction. The higher the correlation is between the subsequent frames, the higher compression performance can be attained.

## 6 Graph Cuts Initialization (GCI) and Motion Compensated Graph Cuts Initialization (MCGCI)

Normally, depth estimation is DERS is performed by multiple iterations of the graph cuts algorithm. In each iteration, another depth value (depth label) is tested by setting it as a "source" of the graph being solved. After performing graph-cuts, some pixels are assigned this new depth label, and the remaining ones stay with the depth label from previous iteration. In such a way, after iterating though all depth labels in set (from minimal to maximal disparity value), the final disparity map can contain all possible disparity values. This whole process is performed multiple times, called "cycles". For example, depth estimation can be done in 2 cycles, for 32 disparity labels, which results in 2x32=64 different graphs creations and graph-cuts solvings.

In the abovementioned depth estimation process in DERS, the initial depth map is initialized with zero values. This enforces the algorithm to change most of the labels in every iteration. In this experiment, we have tested, whether estimated depth can be improved if the graph-cuts optimization algorithm is initialized with labels attained from estimated depth. Two scenarios were considered:

- Graph Cuts Initialization **(GCI)**: The graph cuts is initialized directly with depth labels estimated for the previously processed frames (and of course 0 for the first frame)
- Motion Compensated Graph Cuts Initialization **(MCGCI):** The graph cuts is initialized with depth labels attained from motion-compensation of labels estimated for the previously processed frames (and of course 0 for the first frame).

The tested scenario is depicted in Fig. 4. For the he comparison, methodology shown in from Fig. 1 has been used. The expected outcomes were: improved temporal consistency or improved speed of estimation.

The results are presented in the Table 4.

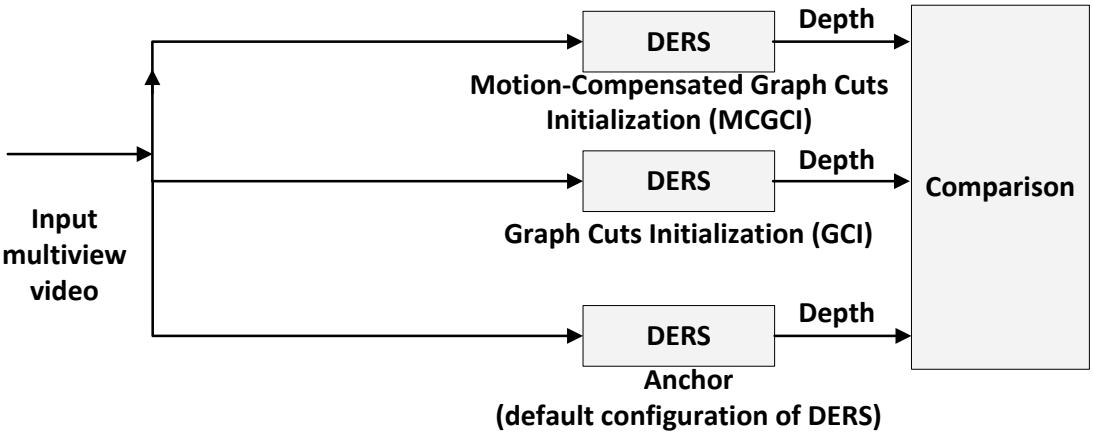


Figure 3. Comparison of depth estimation with the use of Graph Cuts Initialization (GCI) and with use of Motion-Compensated Graph Cuts Initialization (MCGCI), versus original, unmodified DERS. For the he comparison, methodology shown in from Fig. 1 has been used.

Table 4. Comparison of quality of the considered techniques of graph cuts initialization in depth estimation, related to the original (unmodified) DERS technique, based on PSNR of view synthesis.

| Sequence Name | a) PSNR of views synthesized with the following depth: | | | b) Relative change of correlation coefficient, related to the PCC of the original (unmodified) DERS [%] | |
|---|---|---|---|---|---|
| | default depth (anchor) | Depth estimated with the of use of GCI | Depth estimated with the of use of GCI MCGCI | Depth estimated with the of use of GCI | Depth estimated with the of use of GCI MCGCI |
| Poznan Street | 31.98 | 31.88 | 31.91 | +0.02 | -0.01 |
| Poznan Carpark | 30.71 | 30.70 | 30.73 | +0.07 | +0.08 |
| Poznan Hall 2 | 32.85 | 32.86 | 33.01 | +0.21 | +0.31 |
| Newspaper | 31.91 | 31.91 | 31.93 | +0.03 | -0.02 |
| Balloons | 32.94 | 32.90 | 32.92 | +0.15 | +0.17 |
| Kendo | 35.46 | 35.46 | 35.47 | +0.11 | +0.31 |
| Average | 32.64 | 32.62 | 32.66 | +0.10 | +0.14 |

Again, it can be seen that the proposed techniques (GCI and MCGCI ) do not improve the PSNR quality of the synthesized view. Also, the relative change of the correlation coefficient is practically not improved at all. Unfortunately, the same holds for the compression performance improvements presented in Table 5. This means that custom initialization of graph cuts does not improve temporal consistency of the estimated depth.

Table 5. Bjøntegaard bitrate savings - results of MVC compression of depth maps estimated with use of DERS with GC and GCI, respectively, related to compression of depth maps estimated with use of DERS basing on the original test sequences (anchor).

| Sequence name | Bit-rate savings [%] | |
|---|---|---|
| | MCGC | MCGCI |
| Poznan Street | -0.04 | 0.05 |
| Poznan Carpark | 0.23 | 0.21 |
| Poznan Hall 2 | 0.28 | 0.8 |
| Newspaper | -0.05 | 0.12 |
| Balloons | 0.33 | 0.53 |
| Kendo | 0.11 | 0.23 |
| Average | 0.14 | 0.32 |

We have also measured time of execution of depth estimation. The results are presented in Table 6. It can be seen that the time of execution of depth estimation is reduced to about 87% of the original execution time, in both cases of GCI and MCGCI, which yields in about 13% gain.

Therefore it can be concluded that although custom initialization of graph-cuts does not improve temporal consistency, it can bring interesting computational complexity reductions.

Table 6. Time of execution of depth estimation with graph cuts initialization techniques (GCI and MCGCI) versus unmodified original DERS

| Resolution | Sequence name | Time of execution per frame [s] (average over smoothing coefficient values) | | | Relative time of execution [%] | |
|---|---|---|---|---|---|---|
| | | Anchor (original, unmodified DERS) | GCI | MCGCI | GCI | MCGCI |
| HD | Poznan Street | 410.4 | 350.9 | 346.0 | 85.5% | 84.3% |
| | Poznan Carpark | 339.2 | 279.2 | 275.4 | 82.3% | 81.2% |
| | Poznan Hall 2 | 570.6 | 510.7 | 488.4 | 89.5% | 85.6% |
| XGA | Newspaper | 44.4 | 39.2 | 39.9 | 88.2% | 89.9% |
| | Balloons | 46.6 | 40.8 | 39.8 | 87.5% | 85.4% |
| | Kendo | 38.3 | 35.4 | 35.6 | 92.5% | 93.0% |
| Average relative time of execution (versus anchor): | | | | | 87.58% | 86.57% |

## 7 Conclusions

Basing on the presented results it can be concluded that usage of more advanced noise reduction technique with motion compensation (MCNRR technique) provides minor gains compared to a simple noise reduction technique operating only on still regions (SBNR technique). On the other hand, the share of any of the two considered noise reduction techniques in total depth estimation process is negligible (0.3 - 1% of time of execution of the whole process at most) so it is beneficial to use motion compensation-based technique as it will work best on any type of sequences. In any case, the attained improvement of temporal consistency of depth has been measured by about 28% bitrate reduction for considered depth, representing the same scene. Therefore, we recommend to use the proposed scheme in depth estimation for future test sequences.

We have shown that custom initialization of graph cuts algorithm does not improve temporal consistency of the estimated depth. This holds for all considered measures of quality and temporal consistency: PSNR, Pearson Correlation Coefficient and compression performance. It has however been shown that the proposed graph-cuts initialization schemes, GCI and MCGCI, respectively, bring about 13% reduction of time of execution of depth estimation. Therefore we recommend to include the proposed graph cuts initialization scheme in DERS.

As for the measures of temporal consistency, it has been shown that inter-sample correlation between depth values is not a very good indicator. It is instead proposed that enhancement of temporal consistency can be measured by improvement of compression performance of the analyzed depth maps. It is recommended that the future improvements of DERS related to temporal consistency should be demonstrated basing on that basis.

## Acknowledgement

This work was supported by the funds of National Science Centre, Poland, according to the decision DEC-2012/07/N/ST6/02267.

## References

[1] M. Tanimoto, T. Fujii, K. Suzuki, N. Fukushima, Y. Mori, “Reference Softwares for Depth Estimation and View Synthesis”, ISO/IEC JTC1/SC29/WG11 (MPEG) Doc. M15377, Archamps, France, 2008.

[2] Sang-Beom Lee and Yo-Sung Ho “Enhancement of Temporal Consistency for Multi-view Depth Map Estimation”, ISO/IECJTC1/SC29/WG11, M15594, July 2008, Hannover, Germany.

[3] Hui Yuan, Yilin Chang, Haitao Yang, Xiaoxian Liu, Sixin Lin, Lianhuan Xiong, "Depth Estimation Improvement for Depth Discontinuity Areas and Temporal Consistency Preserving" ISO/IEC JTC1/SC29/WG11, MPEG2009/M16048, Lausanne, CH, February 2008

[4] Olgierd Stankiewicz, Krzysztof Wegner, Marek Domański „Estimation of temporally consistent depth maps using noise removal from video Video", ISO/IEC JTC1/SC29/WG11 MPEG/M17612, Dresden, Germany, April 2010

[5] M. Fizick, etal. "Mv-tools web-page", http://avisynth.org.ru/mvtools/mvtools.html - online 2015.

[6] Olgierd Stankiewicz, Marek Domański, Krzysztof Wegner, "Estimation of temporally-consistent depth maps from video with reduced noise ", 3DTV-con 2015, Lisboa, Portugal, July 2015,

[7] G. Bjontegaard, "Calculation of Average PSNR Differences between RD-curves", ITU-T SG16, Doc. VCEG-M33, April 2001.

[8] Dmytro Rusanovskyy, Karsten Müller, Anthony Vetro, "Common Test Conditions of 3DV Core Experiments", Joint Collaborative Team on 3D Video Coding Extension Development of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11 JCT3V-E1100, 5th Meeting: Vienna, Austria, August 2013.