International Journal of
Emerging Research in Management &Technology
ISSN: 2278-9359 (Volume-3, Issue-2)

Research Article | February 2014

Disparity-Map-Based Rendering for Hole Prevention

Mr. S. K. Somasundaram,
Department of Computer Science & Engineering, Associate Professor,
PSNA College of Engineering & Technology, Anna University,Chennai, India

Abstract*: A popular way to convert video is given by depth image-based rendering methods, in which a depth that is associated with an image frame is used to generate a virtual view. An adaptive image warping approach as an improvement to the regular approach is proposed. The new algorithm (DMBR) exploits the smoothness of a typical depth map to reduce the complexity of the underlying optimization problem that is necessary to find the deformation, which is required to prevent holes. This is achieved by splitting a depth map into blocks of homogeneous depth using quad trees and running the optimization on the resulting adaptive grid. So finally, we obtain the virtual view of the depth image location on the video image frame. We focus on the problem to generate the virtual videos of different viewpoints from a given video and associated depth maps at a viewpoint so that the viewers can realistically sense the depths. A spatio-temporal global optimization approach is proposed to synthesize the virtual views from a single video-plus-depth video sequence. Because of the lack of knowledge about the 3D structure of a scene and its corresponding texture, the conversion of 2D video, inevitably, however, leads to holes in the resulting 3D image as a result of newly-exposed areas. The new algorithm exploits the smoothness of a typical depth map to reduce the complexity of the underlying optimization problem that is necessary to find the deformation, which is required to prevent holes. Our proposed approach for the pre-processing of depth maps to prevent holes during the view synthesis. We propose to generate the virtual videos of different viewpoints from a given video and associated depth maps at a viewpoint so that the viewers can realistically sense the depths. We also propose a spatio-temporal global optimization approach that formulates an energy function considering depth map, image structure, texture information and patch shift. We also propose a new and efficient adaptation of the regular grid-based warping.

Key Terms—2D-to-3D-conversion, depth image-based rendering, hole-filling, image warping, optimization.

I. INTRODUCTION

Recently, stereo vision has received much attention due to its high success in feature films, visualization and interactive applications such as computer games. However, 3D vision does not come for free and often implies that two images need to be rendered instead of a single one, as for standard rendering. This can have a high impact on performance which is an issue for real-time applications. In order to satisfy the increasing demand for 3D content, several 2D-to-3D conversion methods have been devised. Technical commitment that was required for the presentation of 3D video was significant, which limited its benefits to a circle of professionals and burdened its general acceptance by a broader audience. However, as a result of a series of technical advances over the past years, 3D displaying devices are becoming increasingly popular in consumer products, for example in television sets, home projectors, or even hand-held devices like smartphones. This development has lead to a growing demand for 3D content. Despite the diversity of technologies for the generation of 3D video for stereoscopic or multi-view displays, the majority of them relies on dense depth maps. As such, the actual 3D image can be created directly on the device. The processes can be split into two steps: (1) depth estimation and (2) view synthesis. Major events were broadcast in 3-D and first commercial 3DTV channels are on air now. Global players in consumer electronics assume that in 2015 more than 30% of all high-definition (HD) panels at home will be equipped with 3-D capabilities. Gamers enjoy their favorite entertainment in a new dimension. Mobile phones, personal digital assistants (PDAs), laptops, and similar devices provide us the extended visual sensation anytime, anywhere. Three dimensional camera systems are already available for professional and private users. Unlike in previous attempts to establish 3-D video in wide markets, there are good reasons to believe that this time it will be sustainable. Stereoscopic 3-D is this time mature enough for the market in terms of quality of technology and content. There is a clear demand from user side and attractive business opportunities are visible. The content and value chains are in place for various application scenarios. Despite the high standards that stereoscopic 3-D has reached today, there is still room for improvement of technology and a number of central problems remain unsolved. Further, new display technology such as autostereoscopic multiview displays, integral imaging, or holographic displays as well as advanced functionalities such as free viewpoint navigation require further research and development. In the context of the 3-D video processing chain from capture to display, this paper is focused on postproduction and processing.

II. RELATED WORKS

More and more 3D display applications can be found in the high-tech products, including 3D LCD/LED displays, 3D laptops, 3D cameras, mobile devices and home video/games, etc. The suitable file formats to support these modern 3D devices have become one of the most important issues. The video-plus-depth format is commonly used in the 3DTV community. It consists of the color intensity and the associated per-pixel depth map at one view. Based on this format, the DIBR (Depth-Image-Based Rendering) system [4] produced virtual views based on the three steps: preprocessing of depth image, image warping and hole filling. However, the major problem in DIBR is how to fill the holes caused by the disocclusion regions in which the occluded pixels in the source view may become visible in the virtual views. Under the video-plus-depth format, some research works focused on the preprocessing of the depth image [5] to reduce the disocclusion regions. Others developed hole filling methods based on image inpainting techniques [6] to fill in the disocclusion regions. For preprocessing of the depth image, the common approach is to apply the smoothness filters, (e.g. Gaussian filter and average filter) to the depth image. After image warping with the smoothed depth image, the disocclusion regions may be split into several small holes. Then the color interpolation can be used to fill in the small hole regions. Zhang et al. extended the idea of the depth preprocessing for the hole filling from the symmetric smoothing filter to the asymmetric smoothing in order to reduce the geometric distortion. For image inpainting, Oh et al. proposed a hole filling method by using depth-based inpainting.

This method is designed by combining the depth-based hole filling and the image inpainting technique. Because hole filling is the major problem in depth-image based rendering. Image inpainting is a technique widely utilized to recover the disocclusion regions. The objective is to fill the unknown regions in a plausible way. Recent exemplar based approaches are commonly used in image inpainting. In [7], a fast algorithm was presented to propagate texture and structure in a small patch. The success of structure propagation was dependent on the order in which the filling proceeds. The confidence value in the synthesized pixel values was propagated in a manner similar to the propagation of information in inpainting. Contrary to the greedy methods, some approaches formulate the image completion as discrete global optimization problems [8] [9] [10]. In [8], image completion was automatically solved using an efficient BP algorithm. However, it did not consider structure information and thus the results may contain structure inconsistency. In [9], the image was completed with manually added structure information. Huang et al. [10] improved the hole filling method in [8] by adding the structure information into the global optimization formulation and solved the optimization problem with a two-step BP. In their method, only a single image was considered for the completion. Then, extended the method to the video completion [11] by adding the motion information to keep spatial and temporal coherence.

III. DISPARITY MAP BASED RENDERING (DMBR)

Using DMBR approaches: DMBR (Disparity Map based Rendering) Approach: The disparity maps with the original size are reconstructed from the received sub-sampled maps by using bilinear interpolation. Here, the image warping procedure is not needed for rendering because the disparity maps have been transmitted to the receivers. Virtual left and right views are created using the reconstructed disparity maps. The computational cost of the rendering procedure is also reduced because it creates two virtual views directly from the disparity maps. It is very effective for mobile devices with the limited computational power and memory. In conventional DIBR methods, the D to D conversion has been performed in the receiver part. However, in our method, the D to D conversion is performed in the transmitter part because disparity information is transmitted through transmission channels instead of depth information. This is very efficient because the transmitter part has high speed computing and mass storage systems compared to the receiver part.

Depth preprocessing and D to D conversion: As mentioned above, one major problem of DIBR is disocclusion which is commonly referred as 'hole'. The other filter is a gradient-based smoothing filter which smoothness original depth images in the horizontal direction and reduces holes. This is the depth preprocessing procedure. Above all, the additional data stream should be transmitted in bitrates below 64Kbps over T-DMB. If the disparity maps, which can be represented by only 3bits, are transmitted, the constraint of the bandwidth (64Kbps) can be satisfied even if the frame rate is 30fps without using lossy compression techniques such as MPEG-4 and H.264. That is, 6.9Mbps is needed for transmission of a disparity map in the case of 3bits, i.e. 320 240 3bit 30frame/s = 6,912,000bps.If the data is compressed using lossless compression techniques such as entropy encoding (generally, its compression rate is higher than 97%), the total transmission data size can be lower than the constraint. Moreover, depth can be effectively preserved because of the lossless compression. To adjust bit-rates to the constraint of the bandwidth, the disparity maps are subsampled to a quarter of the size. The scheme for transmitting the disparity maps over transmission channels. As can be seen, the disparity information is compressed and transmitted through transmission channels using lossless compression techniques. DMBR in the processed part: In receivers, the disparity maps with the original size are reconstructed from the received sub-sampled maps by using bilinear interpolation. Here, the image warping procedure is not needed for rendering because the disparity maps have been transmitted to the receivers. Virtual left and right views are created using the reconstructed disparity maps. Although the holes have been efficiently handled by the depth preprocessing procedure in the transmitters. Holes mainly appear around the boundary of objects, and are filled simply by bilinear interpolation of neighborhood pixels. Then, auto-stereoscopic images for 3D displays are created by interleaving the two virtual views. Consequently, it is possible to create the depth preserving auto stereoscopic images because of using lossless compression techniques in the receivers. Moreover, the computational cost of the rendering procedure is also reduced because it creates two virtual views directly from the disparity maps. It is very effective for mobile devices with the limited computational power and memory. The computational cost of the rendering procedure is also reduced because it creates two virtual views directly from the disparity maps. It is very effective for mobile devices with the limited computational power and memory.

IV. DEPTH-BASED METHODS

In DIBR, Depth-image-based rendering (DIBR) is the process of synthesizing “virtual” views of a scene from still or moving images and associated per-pixel depth information. The left and right virtual views, which create auto stereoscopic images, are rendered by reference image and its corresponding depth image in auto-stereoscopic displays. To maintain the backward compatibility with traditional 2D broadcasting, regular 2D color videos in digital TV format. Its -corresponding depth image, which stores depth information of 8-bit gray values with 0 at the furthest place and 255 at the nearest place, is just added with the same spatiotemporal resolution. Also, the high-quality DIBR technique using the shift-sensor camera setup. Information on the depth structure of a scene always plays an important role in postproduction. That is already well known from conventional TV and cinema production. For instance, together with camera calibration and tracking, depth information is frequently used for visual effect editing, depth keying, augmented reality, scene composition, mixing of computer graphics data with real content, virtual studio productions, and currently also for advanced chroma keying and blue/green screening in news or weather forecast. In 3-D postproduction, however, the role of depth information becomes even more important in a manifold manner. Much functionality in the context of stereo processing outlined in Section II require virtual view synthesis. Today, DIBR is the most important class of algorithms to perform such high-quality virtual view synthesis. Therefore, this section gives an overview of related algorithms and some examples of depth estimation and view synthesis.

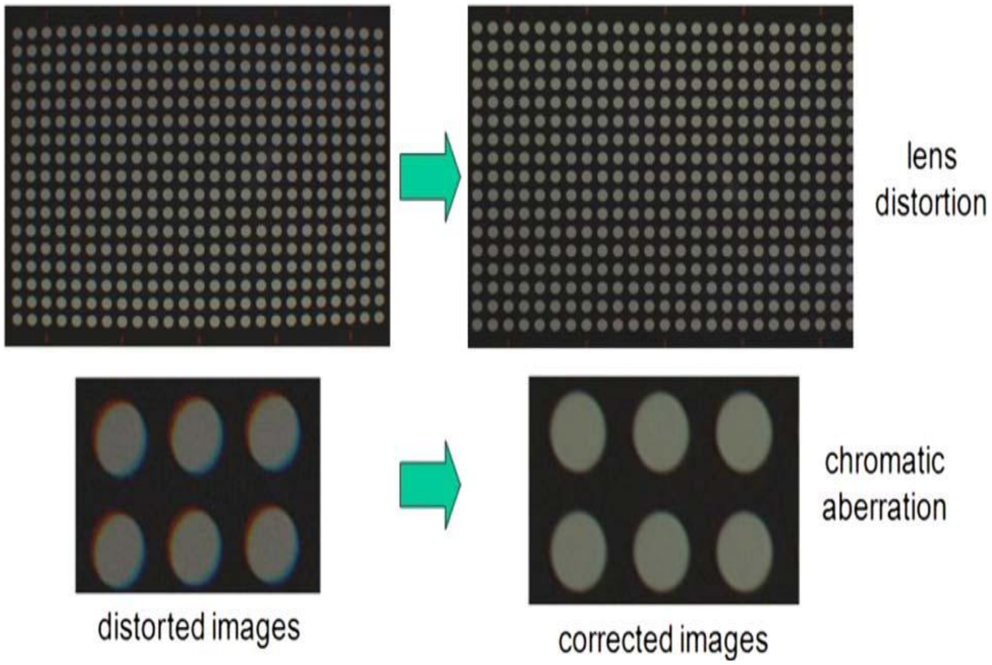
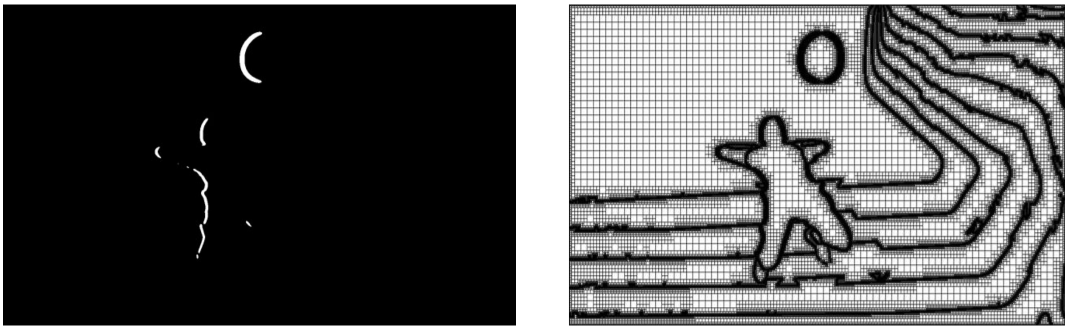


Fig. 1 Examples of measuring and correcting geometrical lens distortions and chromatic aberration.

A. Hole-Filling

Having identified those regions, in which holes will occur, the next step will be to convert the holes masks into a series Fig. 5. Depth discontinuities mask **G** is the result of an edge detection applied to disparity map **D** and indicates depth changes in the scene. In conjunction with the holes masks **H***L* and **H***R*, these mask are used to identify those edges between neighbouring pixels that may be deformed. The proposed adaptive warping algorithm exploits these masks and the depth structure in order to generate a quadtree grid over the image, which is depicted in (c). A span is defined as a sequence of consecutive hole pixels in a row, for which the location of the first pixel before and after the hole is stored as starting *a* and ending point *b*. Additionally, either the starting or the ending point will be marked as fixed, depending on which is closer to the viewer, i.e. $\varphi(a, b)$ = _a, if **C**(a) > **C**(b) b, otherwise.(7) Fixing one side of a span is necessary, as the other side will be pulled towards it, which closes the hole. The pulling of pixels is achieved by assigning the warping value of the fixed span point to all the other pixels in that span. Let (a, b) denote a hole span and assume for the moment that *a* is marked as fixed, we then define the hole-filling boundary constraint for the current span as _i = ˜_a, (8) where $a \leq i \leq b$. This means, in other words, that the warping values of pixels inside a hole span are fixed to the warping value of *a*, which guarantees that these pixels will maintain a relative pair-wise distance of one pixel after warping, which will avoid any holes. The error that is introduced by this constraint will be hidden by spreading it over neighboring non-hole pixels as will be explained later. The case, where *b* is fixed, can be constructed analogously. Hole spans of the left holes mask **H***L* define only boundary constraints for the left warping map _*L*, and spans in **H***R* only for the right warping map _*R*. These constraints will transform exactly those pixels that are marked as holes in **H***L* and **H***R*. However, shifting these marked pixels will not automatically pull along their neighboring pixels outside of the span, which are not marked as holes explicitly. As a result new holes may occur along the sides of hole spans. The next sections will address this problem by introducing a kind of cohesive attraction between pixels. This attraction is expressed as energy terms and ensures that the background is stretched to the foreground in such a way that new holes in the texture are avoided while also preserving disparity.

(a) input image (b) Depth discontinuities mask



(c) Left holes mask (d) quadtree

Fig. 2. Depth discontinuities mask G is the result of an edge detection applied to disparity map D and indicates depth changes in the scene. In conjunction with the holes masks H*L* and H*R*, these mask are used to identify those edges between neighbouring pixels that may be deformed. The proposed adaptive warping algorithm exploits these masks and the depth structure in order to generate a quadtree grid over the image, which is depicted in (d).

B. Image Warping

Image warping is usually required in depth-based view synthesis. In this problem, image warping is to map the pixel position to the corresponding location in the desired view based on associated depth map. In autostereoscopic display, image warping is generally degenerated to one-dimensional displacement along the horizontal scanline based on the assumption that the human eyes are in parallel to the screen at the same horizontal line when watching the display. Thus, we simplify the description of image warping to one-dimensional displacement along the horizontal line in this work. We warp the images and the associated depth maps to the position of the desired view in accordance with the user provided parameter setting. Figure 4 illustrated the warping
results (c)(d) of the original color image and its depth map (a)(b) without depth preprocessing. (e)(f) shows the warping results when the depth images are preprocessed. Users should set the desired view and the relation between input disparities and the desired disparity in terms of pixel unit. A higher input disparity means it is closer to viewers and a lower input disparity indicates it is farther away from viewers. In our warping method, we assume the desired disparity is proportional to the input disparity.

V. PROPOSED APPROACH OF DISPARITY MAP BASED RENDERING (DMBR)

In conventional DIBR methods, the DtoD conversion has been performed in the receiver part. However, in our method, the DtoD conversion is performed in the transmitter part because disparity information is transmitted through transmission channels instead of depth information. This is very efficient because the transmitter part has high speed computing and mass storage systems compared to the receiver part.

Depth preprocessing and DtoD conversion: As mentioned above, one major problem of DIBR is disocclusion which is commonly referred as 'hole'. As shown in Fig. 3, the holes inevitably occur because the scene area, which is occluded in the reference image, become visible in any of the virtual left and right views. The holes are caused by the disoccluded regions of Fig. 3(a) and appear in white regions of Fig. 3(b). Since there is no information to fill the holes in both the reference image and its corresponding depth image, it is not easy to handle the holes. Thus, to minimize the holes and preserve the depth information, two different smoothing filters are sequentially applied to original depth images in the previous works [1, 2, 6]. The first filter is a discontinuity-preserving smoothing filter which removes noise and preserves original depth information. The other filter is a gradient-based smoothing filter which smoothness original depth images in the horizontal direction and reduces holes. This is the depth preprocessing procedure.

Fig3: Depth map after matting process

A. *Gaussian filter*

Gaussian filters is to distributed the uniform gaussion process in all pixels (256 x 256). Gaussian filtering is used to remove noise. It is not particularly effective at removing salt and pepper noise. Gaussian filtering is more effective at smoothing images. It has its basis in the human visual perception system. This is a common first step in edge detection. The images below have been processed with a Sobel filter commonly used in edge detection applications. The image to the right has had a Gaussian filter applied prior to processing.

The Gaussian filter works by using the 2D distribution as a point-spread function. This is achieved by convolving the 2D Gaussian distribution function with the image. We need to produce a discrete approximation to the Gaussian function. This theoretically requires an infinitely large convolution kernel, as the Gaussian distribution is non-zero everywhere. Fortunately the distribution has approached very close to zero at about three standard deviations from the mean. 99% of the distribution falls within standard deviations.

B. *Quadtree filter*

A **quadtree** is a tree data structure in which each internal node has exactly four children. Quadtrees are most often used to partition a two-dimensional space by recursively subdividing it into four quadrants or regions. The regions may be square or rectangular, or may have arbitrary shapes. A similar partitioning is also known as a *Q-tree*. All forms of quadtrees share some common features: They decompose space into adaptable cells. Each cell (or bucket) has a maximum capacity. When maximum capacity is reached, the bucket splits. The tree directory follows the spatial decomposition of the quadtree. Types are given below

The region quadtree:

The region quadtree represents a partition of space in two dimensions by decomposing the region into four equal quadrants, subquadrants, and so on with each leaf node containing data corresponding to a specific subregion. Each node in the tree either has exactly four children, or has no children (a leaf node). The region quadtree is a type of trie. A region quadtree with a depth of n may be used to represent an image consisting of $2^n \times 2^n$ pixels, where each pixel value is 0 or 1. The root node represents the entire image region. If the pixels in any region are not entirely 0s or 1s, it is subdivided. In this application, each leaf node represents a block of pixels that are all 0s or all 1s. A region quadtree may also be used as a variable resolution representation of a data field. For example, the temperatures in an area may be stored as a quadtree, with each leaf node storing the average temperature over the subregion it represents. If a region quadtree is used to represent a set of point data (such as the latitude and longitude of a set of cities), regions are subdivided until each leaf contains at most a single point.

Point quadtree:

The point quadtree is an adaptation of a binary tree used to represent two dimensional point data. It shares the features of all quadtrees but is a true tree as the center of a subdivision is always on a point. The tree shape depends on the order data is processed. It is often very efficient in comparing two dimensional ordered data points, usually operating in O(log n) time.

Edge quadtree:

Edge quadtrees are specifically used to store lines rather than points. Curves are approximated by subdividing cells to a very fine resolution. This can result in extremely unbalanced trees which may defeat the purpose of indexing.

Polygonal map quadtree:

The polygonal map quadtree (or PM Quadtree) is a variation of quadtree which is used to store collections of polygons that may be degenerate (meaning that they have isolated vertices or edges). There are three main classes of PMQuadtrees which vary depending on what information they store within each black node. PM3 quadtrees can store any amount of non-intersecting edges and at most one point. PM2 quadtrees are the same as PM3 quadtrees except that all edges must share the same end point. Finally PM1 quadtrees are similar to PM2 but in this case black nodes can either contain a point and its edges or just a set of edges that share a point but you cannot have a point and a set of edges which do not contain that point.

C. ALGORITHM Balanced QuadTree(T)

Input: Quad Tree T
Output: a balanced version of T

```
L ← list of all leaves of T
while L ≠ ∅ do
        remove a leaf s from L
        if s has to split then do
            add 4 children $s_{NE}$ , $s_{NW}$ , $s_{SE}$ , $s_{SW}$ to s in T &
                update their object contents
            insert $s_{NE}$ , $s_{NW}$ , $s_{SE}$ , $s_{SW}$ into L
            check if $s_{NE}$ , $s_{NW}$ , $s_{SE}$ , $s_{SW}$ have neighbors that
                should split & add them to L
        end-if
end-while
return T
end
```

DMBR in the processed part: In receivers, the disparity maps with the original size are reconstructed from the received sub-sampled maps by using bilinear interpolation. Here, the image warping procedure is not needed for rendering because the disparity maps have been transmitted to the receivers. Virtual left and right views are created using the reconstructed disparity maps. Although the holes have been efficiently handled by the depth preprocessing procedure in the transmitters. holes mainly appear around the boundary of objects, and are filled simply by bilinear interpolation of neighborhood pixels. Then, auto-stereoscopic images for 3D displays are created by interleaving the two virtual views. Consequently, it is possible to create the depth preserving auto stereoscopic images because of using lossless compression techniques in the receivers. Moreover, the computational cost of the rendering procedure is also reduced because it creates two virtual views directly from the disparity maps. It is very effective for mobile devices with the limited computational power and memory.

VI. PERFORMANCE RESULTS

This section evaluates the performance of the proposed adaptive grid-based algorithm and compares it against the regular grid-based algorithm in terms of visual quality and computational complexity. The first column depicts the depth maps with the quadtree segmentations as an overlay in red, which are used to warp the images. In column two the anaglyph output of the DMBR without any warping is depicted. Columns three and four depict the output of regular warping and the proposed adaptive warping as a preprocessing step to DMBR

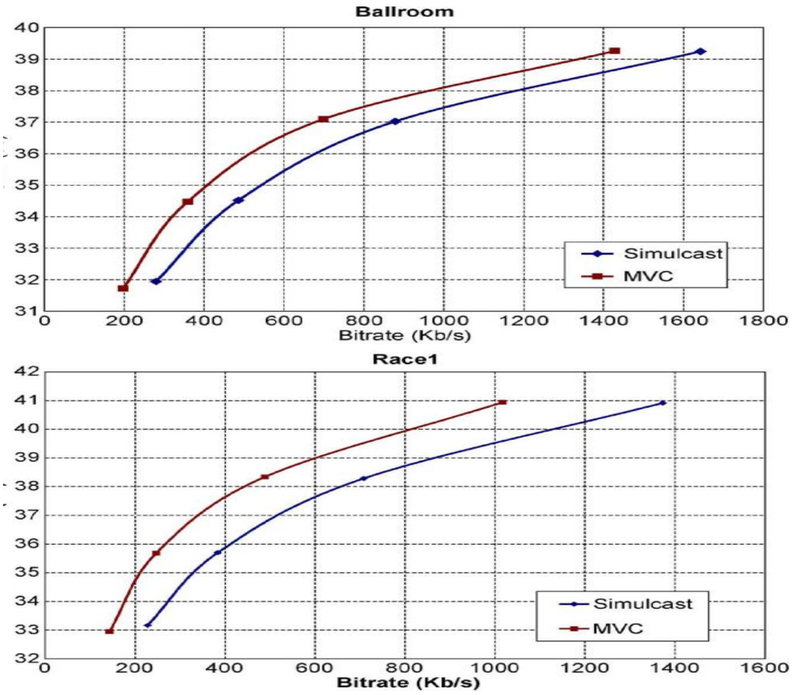


Fig. 4. Coding results for ballroom and race 1 sequences; each sequence includes eight views at high resolution.

The algorithm has no notion of objects or object structure, which may lead to visible distortions when the background consists of regular structured objects like the buildings in the background image. In order to alleviate this problem additional energy terms that preserve structures such as straight lines could be employed. The parameters used for warping are given in Table I. The values for the smoothness weight *ws* are chosen to be larger than the disparity weight *wd* and the similarity weight *wa* to balance the different value ranges of the smoothness energy and the other energies. The depth budget has been chosen such that it produces a good stereo experience on a computer display. The regular grid-based approach and the proposed approach, the image size of all test data mentioned in Table I was downsized to a width of 640 pixels and the values of the near and far shifts were scaled accordingly. This restriction is necessary because the regular grid-based approach either depletes the available computer memory, if a direct solver is used, or run-time becomes infeasibly long before an iterative solver converges.

VII. CONCLUSION

We also proposed a new algorithm based Disparity Map Based Rendering (DMBR) on an adaptive grid-structure, which facilitates an approximation of the first algorithm and increases the efficiency significantly. The effectiveness of the proposed method has been shown by a series of experiments which demonstrate that the proposed DMBR algorithm produces visually very similar results at a fraction of the computational cost that is required by the regular grid-based approach. We conclude using DMBR approach we can enhance the visual quality with less computational cost and error loss in 2D and 3D images in real time applications. The revolution from 2D display to 3D display may be as significant as the revolution from monochrome to color display in the television history. several TV manufacturing companies have successively developed their advanced 3D display technologies that allow people to experience realistic 3D scenes without wearing 3D glasses. People can watch the 3D contents from different viewpoints by changing the viewing directions. Many companies have invested a considerable amount of resources in the related technology as the 3D industry has been foreseen to develop very rapidly in recent years.

REFERENCES

[1] Fidaner. (2008). *A Survey on Variational Image Inpainting,Texture Synthesis and Image Completion* [Online]. Available: http://www.scribd.com/doc/3012627/.

[2] Y. Wang, C. Tai, O. Sorkine, and T. Lee, “Optimized scale-and-stretch for image resizing,” *ACM Trans. Graph.*, vol. 27, no. 5, pp. 118–126, Dec. 2008.

[3] P. Krähenbühl, M. Lang, A. Hornung, and M. Gross, “A system for retargeting of streaming video,” *ACM Trans. Graph.*, vol. 28, no. 5, pp. 126–136, Dec. 2009.

[4] Y. Guo, F. Liu, J. Shi, Z. Zhou, and M. Gleicher, “Image retargeting using mesh parametrization,” *IEEE Trans. Multimedia*, vol. 11, no. 5, pp. 856–867, Aug. 2009.

[5] ITU-T and ISO/IEC JTC 1, “Advanced video coding for generic audiovisual services,” ITU-T Recommendation H.264 and ISO/IEC 14496-10 (MPEG-4 AVC), 2010.

[6] J. Konrad, M. Wang, and P. Ishwar, “2D-to-3D image conversion by learning depth from examples,” in *Proc. IEEE Comput. Soc. Conf.CVPRW*, Jun. 2012, pp. 16–22.

[7] Smolic, P. Kauff, S. Knorr, A. Hornung, M. Kunter, M. Müller, and M. Lang, “Three-dimensional video postproduction and processing,” *Proc. IEEE*, vol. 99, no. 4, pp. 607–625, Apr. 2011.

[8] “Adaptive Image Warping for Hole Prevention in 3D View Synthesis”Nils Plath, Sebastian Knorr, *Member, IEEE*, Lutz Goldmann, and Thomas Sikora, *Senior Member, IEEE* Transactions, Vol. 22, NO. 9, September 2013.

[9] M. Lang, A. Hornung, O. Wang, S. Poulakos, A. Smolic, and M. Gross, “Nonlinear disparity mapping for stereoscopic 3D,” *ACM Trans. Graph.*, vol. 29, no. 4, p. 75, Jul. 2010.

[10] O. Wang, M. Lang, M. Frei, A. Hornung, A. Smolic, and M. Gross, “Stereobrush: Interactive 2D to 3D conversion using discontinuous warps,” in *Proc. 8th Eurograph. Symp. SBIM*, 2011, pp. 47–54.

[11] E. Haber, S. Heldmann, and J. Modersitzki, “Adaptive mesh refinement for nonparametric image registration,” *SIAM J. Sci. Comput.*, vol. 30, no. 6, pp. 3012–3027, 2010.

[12] Choi, D. Kim, S. Choi, and K. Sohn, “Visual fatigue modeling and analysis for stereoscopic video,” *Opt. Eng.*, vol. 51, no. 1, pp. 017206- 1–017206,-12, Feb. 2012.