DEPTH MAP MANIPULATION FOR 3D VISUALIZATION

Ianir Ideses, Leonid Yaroslavsky, Barak Fishbain

Tel-Aviv University Israel

ABSTRACT

In order to synthesize 3D content, one needs to have either a stereo pair or an image and a depth map. Many methods exist to compute depth maps, however, these methods are usually very intensive in term of computational complexity. In this paper we focus on processing of depth maps acquired by simple motion extraction methods and concentrate on manipulating these coarse depth maps to yield high quality 3D images and video.

Index Terms— Stereoscopic visualization, Anaglyphs, Depth maps, MPEG 4, 3D filtering

1. INTRODUCTION

3D visualization is a growing field that has gained much attention in recent years. Developments in this field include improved 3D visualization devices, namely autostereoscopic and multi view autostereoscopic displays, improved stereo acquisition systems, methods to compute depth maps and rendering algorithms.

One of the key aspects of 3D visualization and, in particular, 3DTV is computation of depth maps. This aspect enables synthesis of 3D video from 2D video and use of multi view displays.

Depth maps can be acquired in several ways. One method is to measure the real 3D properties of the scene objects, using, for example, a range finder, eg. a LIDAR (Light Detection and Ranging) system. Other methods rely on using 2 cameras and computing the correspondence for each stereo pair pixel. Once a depth map is acquired for every frame, it can be used to construct its artificial stereo pair.

Using depth maps for 3D visualization is superior to acquiring a stereo video stream in that that it allows transformations on the depth and manipulations on the parallax baseline.

There are many methods to compute depth maps, among them are the works of Lucas and Kanade [1], Horn and Schunck [2] Senthil Periaswamy and Hany Farid [3], Yu-Te Wu et al [4], L. Alvarez et al [5], Jochen Schmidt [6] and Brox [7]. The drawback of these methods is that they require great computation power and are not very well suited to high quality real-time 3D rendering. Another method for computing depth maps is extraction of motion vector information from standard video encoders [8].The drawback of this method is that the motion fields acquired from compression CODECS are not smooth and may yield 3D images with a high degree of artifacts. In this paper we present methods to improve the 3D visualization quality acquired from compression CODECS by spatial/temporal and logical operations and manipulations.

2. EXTRACTION OF MOTION FIELDS

MPEG 4 (H.264) is a modern compression standard that uses both temporal and spatial compression. While spatial compression is basically a form of JPEG compression, it is temporal compression that enables the high compression rates of MPEG 4. In temporal compression, each frame is divided into blocks and block search is performed between adjacent frames. In this fashion, it is necessary to store the movement of the block from one frame to the other, thus reducing the amount of information to store.

MPEG 4 enables computation of motion vectors in blocks as small as 4X4 pixels with quarter pixel accuracy. These data are very useful for depth estimation. In it's simplest form, the horizontal, X-axis, motion vectors can be used as depth data. This holds for cases where there is only lateral motion on the X-axis and no scene motion is present (a canonical stereo setup). For other motion types it is necessary to make a transformation from motion vector maps to depth maps. In our implementation, motion vector maps were extracted as a part of the MPEG 4 encoder schema. The encoder was instructed to extract motion vectors for every frame (regardless of the ultimate frame type) with the minimal block size.

In some cases, motion vector maps can be directly treated as depth maps. This approximation holds when the two images/frames are taken in parallel viewing or when they are acquired in small ranges of disparity in the case of epipolar acquisition. This is usually the case in computation of depth maps of 2D video. However, there are many cases

978-1-4244-1755-1/08/$25.00 ©2008 IEEE

Authorized licensed use limited to: Technion Israel Institute of Technology. Downloaded on December 23,2024 at 09:47:31 UTC from IEEE Xplore. Restrictions apply.

where this approximation does not hold. This happens when the motion is either too rapid in terms of camera rotation or in the case of camera zoom. Such cases can be detected by analysis of the motion vector maps and dealt with.

In the case of zoom, it is necessary to change the dynamic range of the depth values (while cropping out border pixels), the amount of dynamic range scaling has to be congruent with the zoom factor. For the purpose of visualization this has to be visually comfortable rather than true-to-life accurate. In the case of rotation around a specific object, one needs to invert the disparity values, so that the close object receives high disparity values although it appears to be static. Other depth values should remain the same. This is a non-trivial case and indeed is error prone. In our implementation, we treated motion as a sole depth cue, namely, we calculated the depth solely on the values of the X and Y motion vector values. The depth was estimated by:

$$D(i,j) = c\sqrt{MV(i,j)_x^2 + MV(i,j)_y^2} \ , \qquad (1)$$

where $D(i,j)$ is the depth value for pixel (i,j) and $MV(i,j)_x$, $MV(i,j)_y$ are the X and Y motion vectors values for that pixel respectively and ***c*** is a custom defined scale parameter. The scaling parameter ***c*** can be utilized in two ways, either automatically, or set as a user selected factor. In our implementation, both methods are supported. One may opt to scale the parameter to fit the maximal disparity over all frames – simply adding a constant gain to the depth map values, or perform automatic scaling unto some predefined parallax, keeping maximal parallax constant in all frames. In essence, this operation stretches the dynamic range of all depth maps to this level (a nominal parallax value for comfortable viewing is of the order of 20 pixels).

An example of a resulting depth map extracted from MPEG 4 is presented in Fig 1. As can be seen in the figure, the resulting depth maps retains the appropriate shape information. However it is noisy and requires filtering for good visualization. In this implementation, low pass filtering was applied to the synthesized image after rendering with adequate results.

3. DEPTH MAP MANIPULATIONS

In order to improve the visual quality of 3D images rendered from these depth maps, we propose several methods. Among these are spatial and temporal rank filtering, local 3D-DCT and other morphological/logical operations (registration, depth map repetitions, etc').

Analysis of the resulting depth map (sequences can be found in [9]) has shown that these images suffer from erratic behavior in areas of no motion and in problems associated with global motion and sign inversion when objects change their movement direction.

The first step in improving these depth maps was to find the global motion value and adjust all other motion vectors to it. This was achieved by computing the histogram of the depth map and locating its highest peak. Sequences that contain global motion usually exhibit its value in this peak (robust estimates of this value can be found using other methods, these include averaging, alpha trimmed mean or median). In sequences without global motion this value was useful for determining the value for direction inversion, under the assumption that in stationary images, the majority of the motion pixels should be zero. Once this value has

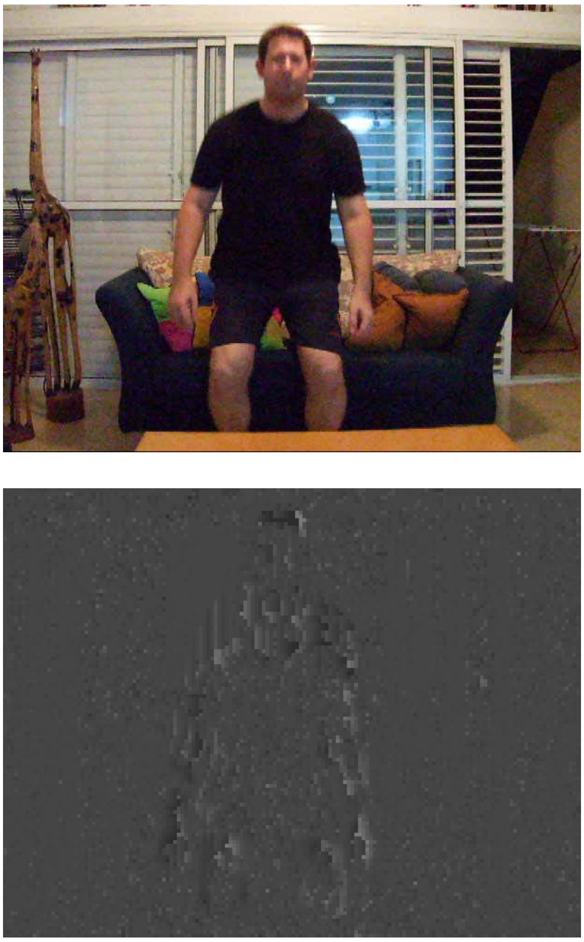

Figure 1. A video frame (top image) and the corresponding depth map (bottom image). Although the resulting depth map does not necessarily show the exact metrics of the stereo pair, it is sufficient for the purpose of visualization. Note that the depth map is noisy, especially in zero motion areas.

Authorized licensed use limited to: Technion Israel Institute of Technology. Downloaded on December 23,2024 at 09:47:31 UTC from IEEE Xplore. Restrictions apply.

been computed, it is subtracted from the entire depth maps image and the absolute value is taken.

Once the depth map has been normalized it is subjected to spatial-temporal filtering. We used 2 types of filtering, low pass filtering using local 3D-DCT and rank filtering [10]. Of these 2 methods, rank filtering proved more useful for the general case. Spatial-temporal filtering was achieved in 2 stages, computing the median for each pixel along the time domain, for global motion scenarios this required prior registration of the images. Computation of the registration parameters was not necessary as it was extracted from the motion histogram. Once images were registered and each pixel's value was computed through a median filter, temporal filtering along a spatial neighborhood was performed. In principle the median operation can be computed over the spatial-temporal cube, but for the sake of reducing computational complexity it was realized separably.

Selecting the neighborhood for temporal filter was based on evaluation of the motion field. We chose a value that enabled a reasonable amount of pixels to be evaluated (this becomes important mostly for global motion scenarios). The parameter that we chose was 7 frames. This amount to about ¼ of a second for normal video standards and enables good filtering without artifacts due to major scene changes.

Selecting the spatial neighborhood relies on previous results attained for depth map resolution and quantization [11-12]. These results indicate that the depth map resolution can be at least one order of magnitude lower than that of the image itself. Therefore we chose a filter with large support that corresponds to 1/8 of the image size on each axis.

Due to the nature of the method of depth map computation (shape from motion), it may happen that the depth maps be completely zero. This can happen when no motion is present, even though the depth was previously extracted. In these cases we use the previously obtained depth map again until motion is redetected. This is referred to as depth map repeating.

3D local DCT is an extension of 2D local DCT to the temporal domain [13]. In this case one can design filters that would act as noise suppressors/smoothers in both the spatial and temporal domains. These filters exhibit superior performance to standard 2D filters [10]. In our implementation, we used 3D-DCT filtering as the last stage, using a 3D LPF filter. In most cases this filtering was not necessary due to the good noise suppression capabilities of the rank filtering.

A flow chart of the process is shown in Fig. 2.

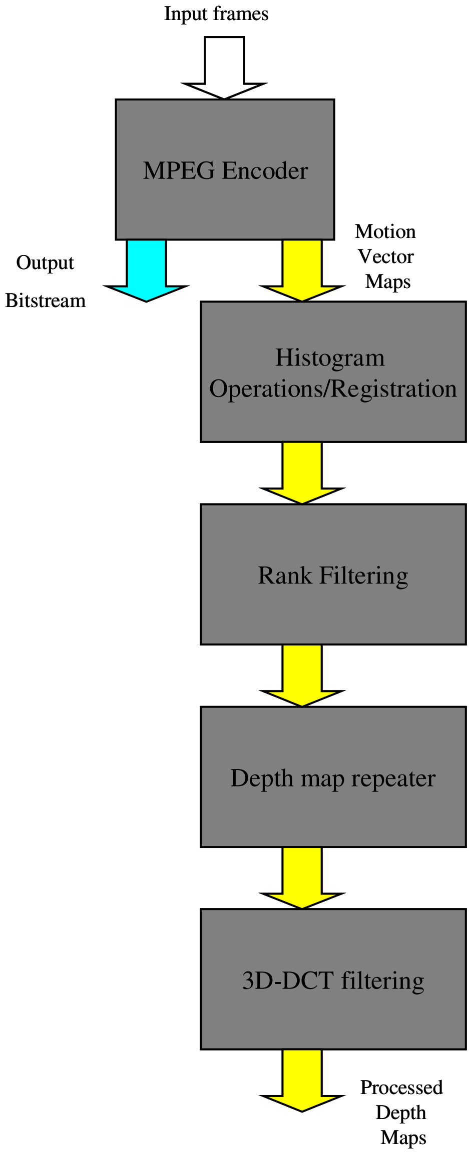

Figure 2
Flowchart of the depth map processing algorithm.

4. Rendering

Once the depth maps are computed for each frame, a stereo pair can be rendered. Rendering is performed by resampling the incoming frame from the bitstream on a grid that is governed by the depth map. This process is described in equation 1.

$$P_R(i,j) = I\{P_L(i,j), D(i,j)\}, \qquad (2)$$

Authorized licensed use limited to: Technion Israel Institute of Technology. Downloaded on December 23,2024 at 09:47:31 UTC from IEEE Xplore. Restrictions apply.

where $P_R(i,j)$ is the pixel value of the rendered right image in location (i,j), $P_L(i,j)$ is the initial left image value at (i,j), I is the resampling operator and $D(i,j)$ is the depth value at (i,j).

5. RESULTS

Once we rendered a stereo pair for each frame, we could use standard 3D visualization techniques. The most suited to standard display hardware is anaglyphs. A resulting depth map and anaglyph can be seen in figure 3.

As can be seen, we were able to filter out the noise that is present in this type of motion field based depth maps and attain very good visualization. In addition, further processing of the stereo pair (smoothing and dynamic range manipulation are also possible [13]). Additional frames and video sequences are available online [9]

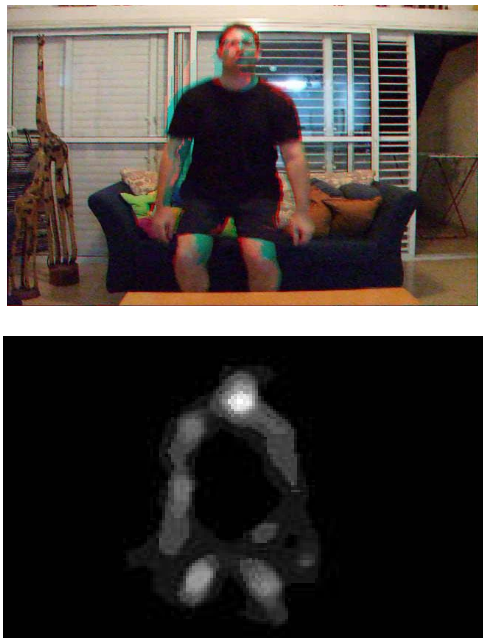

Figure 3. Top image - an example of a 3D image (image viewed with anaglyph glasses, red filter for right eye, blue filter for left eye).

6. CONCLUSION

In this paper we have described methods to manipulate depth maps for the purpose of 3D visualization. We have shown that even noisy maps that can be extracted from motion compensation based COEDCS can be filtered to achieve good 3D visualization. This processing can be performed in parallel to the depth map extraction and can be realized with very low computational complexity.

7. REFERENCES

[1] Lucas, B., and Kanade, T. “An Iterative Image Registration Technique with an Application to Stereo Vision”, in *Proceedings of 7th International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 674-679, 1981.

[2] B. Horn and B. Schunck, “Determining Optical Flow”, *Artificial Intelligence*, 17:185-203, 1981.

[3] Senthil Periaswamy, Hany Farid ", Elastic Registration in the Presence of Intensity Variations", *IEEE Transactions on Medical Imaging*, Volume 22, Number 7, July 2003.

[4] Yu-Te Wu, Takeo Kanade, Ching-Chung Li and Jeffrey Cohn, "Image Registration Using Wavelet-Based Motion Model", *International Journal of Computer Vision*, July 2000.

[5] L. Alvarez, R. Deriche, J. Sanchez, and J. Weickert. "Dense Disparity Map Estimation Respecting Image Discontinuities: A PDE and Scalespace Based Approach.", Technical Report RR-3874, *INRIA*, January 2000.

[6] Jochen Schmidt, Heinrich Niemann, and Sebastian Vote., "Dense disparity maps in real-time with an application to augmented reality". IEEE Computer Society, Orlando, FL USA, December 3-4 2002.

[7] T. Brox, A. Bruhn, J. Weickert, “Variational Motion Segmentation with Level Sets”, *ECCV*, pp. 471:483, 2006

[8] Ianir Ideses, Leonid P. Yaroslavsky and Barak Fishbain, "Real-Time 2D to 3D Video Conversion", *Journal of Real-Time Image Processing*, Volume 2, Number 1 / October, 2007.

[9] http://www.eng.tau.ac.il/~ianir

[10] L. Yaroslavsky, "Space Variant and Adaptive Transform Domain Image and Video Restoration Methods", *Advances in Signal Transforms: Theory and Applications, J. Astola, L. Yaroslavsky, Eds. , EURASIP Book Series on Signal Processing and Communications*, Hindawi, 2007.

[11] I.Ideses, L.P.Yaroslavsky, B.Fishbain, "How Sharp Must Depth Maps Be for Good 3-D Video Synthesis: Experimental Evaluation and Applications", *ICO Topical Meeting on Digital Holography and Three-Dimensional Imaging*, Vancouver Canada, 2007.

[12] I.Ideses, L.P.Yaroslavsky, I.Amit and B.Fishbain, "Depth Map Quantization – How Much Is Sufficient ?", *3DTV CON*, Kos, Greece 2007.

[13] I. Ideses and L. P. Yaroslavsky, "Three Methods that improve the visual quality of colour anaglyphs", *J. Opt. A: Pure Appl. Opt.* 7, pp. 755-762, 2005.

Authorized licensed use limited to: Technion Israel Institute of Technology. Downloaded on December 23,2024 at 09:47:31 UTC from IEEE Xplore. Restrictions apply.