MVA '96 IAPR Workshop on Machine Vision Applications, November. 12-14, 1996, Tokyo, Japan

Low Complexity Disparity Estimation and Object-Based Image Synthesis for Multi-Viewpoint Stereoscopic Videoconferencing

Ebroul Izquierdo M., and Michael Karl *
Heinrich-Hertz-Institute for Communication Technology (HHI)

Abstract

This paper focuses the realization of a low complexity disparity estimator and the generation of arbitrary intermediate views for typical stereo videoconference sequences. The method presented has been optimized in order to achieve a very low hardware complexity keeping a good performance with regard to the addressed application. A first hardware realization of the estimator is described. Additionally, an object-based approach to synthesize intermediate views is presented. The performance of the system is verified by several computer simulations.

1 Introduction

The generation of realistic world scenarios through virtual environments becomes an important tool in recent communication technologies. Especially important to the achievement of this goal is the three dimensional impression, which can be reached via stereo analysis. The principle at the heart of this technology is to analyze the information supplied by a stereo camera with large baseline in order to synthesize intermediate stereo views. These views simulate virtual stereo cameras for the human viewer who is positioned anywhere between the two real cameras. Such a system offers telepresence illusion with continuous motion parallax and in the particular case of videoconferencing the system additionally allows eye-contact between the conference participants.

In this paper a low complexity disparity estimator, its hardware realization and an algorithm for generation of intermediate views are described. Starting-point of the research presented is the method introduced in [1]. This previously reported method has been optimized in order to develop a disparity estimator for special videoconference situations at a low hardware complexity cost. The algorithm presented has been taken as basis for the development of a device capable to perform real-time video conferencing with realistic multi-viewpoint 3D-impression. The disparity estimator consists of four modules: Preprocessing, Block-Matching for global disparity estimation, Block-Matching for local disparity estimation and dense disparity fields generation. A flowchart describing the interrelation of these modules is shown in figure 1.

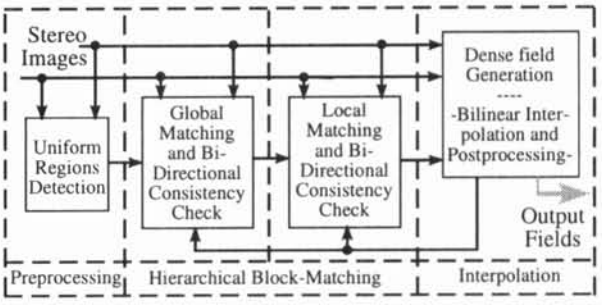

Figure 1: System Overview of Disparity Estimator

According to the application addressed in this paper an object based algorithm to synthesize intermediate views using disparity information is presented as well. This approach assumes a convex object located in the center of the scene. This assumption is fulfilled by typical videoconferencing situation, in which usually the scene consists of the head and shoulder part of a person in front of a uniform background or a previously recorded textured background. A system overview of the image synthesis method is depicted in figure 2.

2 Disparity Estimation

The most promising disparity estimation technique from a hardware realization point of view is a Block-Matching based estimator. In this case the success of correspondence estimation depends essentially on the intensity variation of the images. It is very well known that Block-Matching fails in areas of the image where the intensity variation is low. For this reason the decomposition of the image into homogeneous and textured (non-homogeneous) areas seems reasonable. The matching process is then

*Address: Einsteinufer 37, D-10587 Berlin, Germany. E-mail: ebroul@hhi.de

applied to textured regions whereas the displacements in homogeneous regions are estimated using a suitable interpolation strategy. The process of uniform regions extraction or background/foreground segmentation is performed by the first module of the disparity estimator. In order to select the image points, which can not be distinguished from their neighbors, a simple difference-based interest operator is used. The preprocessing system hardware was only optimized for head and shoulder scenes with uniform background. For scenes with static textured background, a foreground mask can be easily extracted if the foreground information has been previously recorded.

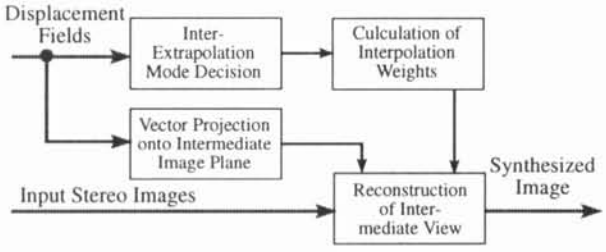

Figure 2: System Overview of Image Synthesis

The correspondence search is performed by the second and third modules. Herein a hierarchical approach in two levels has been realized. In each level the best match is selected using a cost function, which consists of an area correlation term and a smoothness term involving temporal prediction vectors. The area correlation is based on the mean absolute difference (MAD). Let us denote the left and right intensity maps at time t as I_l^t and I_r^t. For a sampling position z in the left image, the cost function is defined as:

$$F(d, d_p, \alpha) = \frac{1}{mn} \sum_m \sum_n |I_l^t(z) - I_r^t(z+d)| + \alpha |d - d_p|$$

with d the current displacement vector, d_p the temporal prediction vector and α a suitable weight coefficient. In the first level global displacements are estimated. In order to reduce noise sensitivity and simultaneously reach higher efficiency, both images are subsampled. Thereafter each image is split into rectangular blocks. In each block the sampling position, which is best distinguished from its neighbors, is chosen as the representative point for the entire block. Full search along the epipolar lines in a maximum search range is performed. Finally, a bi-directional consistency check is performed in order to reject outliers. In the second level each fourth sampling position is taken as reference point to be matched. The matching process is applied to the full resolution images using relatively small measurement windows. Ten vectors are selected as candidate vectors: Six from the global level, three from the surrounding sampling positions already calculated and one from the temporally preceding displacement field at the same spatial position. Each candidate vector is tested within a small search range. In addition, those candidates, which point into an homogeneous area of the other image are regarded as invalid. The matching process is also performed bidirectionally, in order to apply the cross-consistency check on the estimation results.
After local matching disparity vectors at sparse positions are available. In order to obtain dense disparity maps in the last module bilinear interpolation in a separable fashion is performed. We start with horizontal linear interpolation, which generates the final values within each fourth row of dense fields. After horizontal interpolation, a vertical 7-tap median filter is applied. Finally an ordering constraint check is carried out, followed by linear interpolation of those disparity values which violate this constraint. Once dense disparity fields are already calculated at each fourth row, the remaining rows are filled by a vertical linear interpolation. Constant value extension from the next position with known disparity is applied in the image borders.

3 Hardware Realization

In order to achieve a low hardware complexity, the second and the third modules have to be optimized, because the amount of hardware essentially depends on the implementation of Block-Matching. Taking the calculation of the pixel difference and its accumulation as one Block-Matching operation, the number of operations amounts to 300 Million/sec for the global level and 3.000 Million/sec for the local level. It turns from this, that a pure solution based on Digital Signal Processors (DSP) is not suitable and specific Block-Matching processors have to be applied. The Block-Matching array PAME117 [4] designed at the Heinrich-Hertz-Institut is an example for a Block-Matching processor based on the principle of shifting measure pixels, but the number of processor elements does not match the parameters of disparity estimation. Furthermore, an additional processor and peripheral circuitry have to provide the pixels to be matched as well as computational power to perform the postprocessing of MADs.
In deviation from the general application described in [2], the search for disparity estimation is carried out only in horizontal direction, so that the structure of the processor can be simplified. Furthermore, in most cases the ten start vectors for the local search can be replaced by a single search of 20 search positions. This noticeably decreases the number of Block-Matching operations and leads to a processor array of 20 processor elements, each calculating

the MAD for one search position. The complexity of present high-end Field Programmable Gate Arrays (FPGA) allows the design of a FPGA-based Block-Matching processor containing 20 processor elements for parallel MAD calculation as well as additional circuitry for pixel address generation and MAD postprocessing.
The processor is depicted in figure 4. It interacts with a DSP for provision of parameters used for address generation and MAD postprocessing and with a dual ported memory which sequentially stores both left and right image slices and blockwise outputs left and right pixels for Block-Matching. In case of left-to-right estimation a crossbar switches left pixels to the reference pixel input and right pixels to the test pixel input of the processor array. The inputs are exchanged if a right-to-left estimation occurs. The cost function is performed by an addition of the MAD and a counter register which is preloaded with the temporal prediction vector. Thus, the expression $\alpha|d - d_p|$ is built by decrementing or incrementing the counter register. The last step of finding the minimum of the cost function requires the presence of a vector mask which defines the valid positions to be taken into account for the search. The mask is provided by the DSP and results from ten start vectors for local search and the start vector transmitted to the processor.

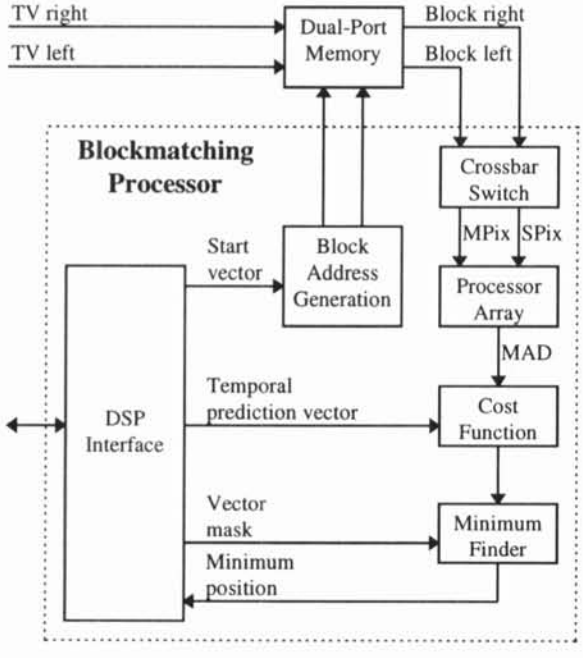

Figure 3: FPGA-based Block-Matching Processor

As depicted in figure 4, the estimation algorithm allows a parallel connection of Block-Matching modules, each calculating a separate image slice. The parameters used for local disparity estimation require four modules working in parallel to meet the real-time requirements.

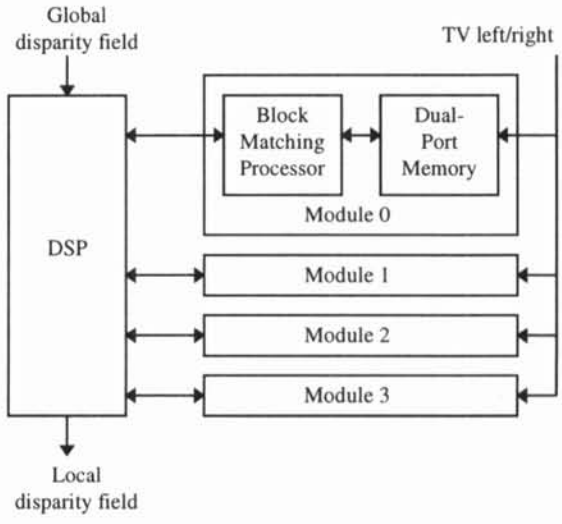

Figure 4: Block Structure for Local Disparity Estimation

The modules are supervised by a DSP which builds the Block-Matching commands for each module and performs the bidirectional consistency check. The hardware solution offers an area efficient design so that both local level and dense field generation can be placed on a single board. The whole disparity estimator will be realized by two boards.

4 Image Synthesis

To generate an arbitrary view, the calculated disparity vectors are first projected onto the intermediate image plane. In order to avoid holes each vector is also projected onto the adjacent sampling positions. Conflicts can occur if a vector is already assigned to the current sampling position. Such conflicts are solved by applying the same strategy described in [1]. After projection of displacement vectors onto the intermediate image, occluded image regions have to be detected. This step plays a crucial role because in occluded image regions the available intensity information has to be extrapolated from only one of the two stereo images, whereas in regular image regions interpolation of intensity values provides from both stereo images has to be performed. Due to the assumed object model occluded image areas are implicitly known. The left image side of the intermediate image is preferably reconstructed using only the left stereoscopic channel. The right image side of the intermediate image is preferably reconstructed using only the right stereoscopic channel. The central part of the intermediate image is recon-

structed using both channels. A detailed realization of this concept is presented in [3], where suitable weights are defined in order to perform a smooth transition between extrapolation and interpolation mode.

5 Results of Computer Simulations

The performance of the methods presented has been tested by processing the stereoscopic sequences ANNE, MAN, CLAUDE and GWEN, which represent typical videoconferencing situations. These sequences have been recorded using stereo cameras with baselines varying between 15 and 80 cm. The distance between the conferencing person and the camera varies between 1.2 and 2.5 m. Extremely large occluded image regions are present within the foreground object. The largest disparity vector can reach 230 pixels. Excellent overall image quality has been obtained in all cases. Most of the videoconferencing sequences considered were chosen to fulfill the uniform background assumption, which simplifies the correspondence estimation and fits in with the proposed object-based image synthesis method. For the sequence GWEN the background (textured) has been recorded previously. This information has been taken into account in the disparity estimation as well as in the image synthesis. Figure 5 shows synthesized central viewpoints for the tenth frame pair of the sequences ANNE and CLAUDE. In this representation the computed central viewpoint is displayed between the two original stereo images.

6 Summary and conclusions

A method for disparity estimation and image synthesis applied to 3D-videoconferencing with viewpoint adaptation is introduced. The novelty of the disparity estimator is twofold: On the one side it has been optimized in order to achieve a very low hardware complexity and on the other side it shows robustness and accuracy with regard to the addressed application. The goal of estimation of reliable displacement maps with extremely low computational costs is reached by a hierarchical Block-Matching method. A first hardware realization of the disparity estimator is described. The presented image synthesis approach exploit the a priori knowledge of scene properties in typical videoconferencing situations. Several computer simulations show, that the system reported in this paper is capable of offering telepresence illusion with continuous motion parallax and good image quality in videoconferencing, keeping implementation costs low.

Figure 5: Synthesized central viewpoint of the tenth frame pair of sequences ANNE (left) and CLAUDE (right): Original left view (top), synthesized central view (middle) and original right view (below).

7 Acknowledgments

This work was supported by the German Ministry of research and education under grant 01BK304.

References

[1] E. Izquierdo M. and M. Ernst, "Motion/Disparity Analysis and Image Synthesis for 3DTV," Proc. Signal Processing of HDTV VI, N. Ninomiya, L. Chiariglione, Editors, Part 6-B, 1995.

[2] M. Karl, M. Talmi, "TMS320C40 Applications for Real-Time Image Processing: Communication Port Interface Designs for Synchronized Data Transfer," The 5th International Conference on Signal Processing Applications and Technology, Dallas, USA, Oct. 18-21, 1994.

[3] J. R. Ohm and E. Izquierdo M., "An Object-Based System for Stereoscopic Videoconferencing with Viewpoint Adaptation," Proc. On European Symposium on Advanced Imaging and Network Technologies, Berlin, Germany, 1996.

[4] C. V. Reventlow, M. Talmi, S. Wolf, M. Ernst, K. Müller, C. Stoffers, "System Considerations and the System Level Design of a Chip Set for Real Time TV and HDTV Motion Estimation," Journal of VLSI Signal Processing, Kluver Academic Publ., April 1993.