

# Extension of Incomplete 3D for Arbitrary Multi-View-Synthesis

Eddie Cooke, Oliver Schreer, Bernhard Pasewaldt, Peter Kauff

Heinrich-Hertz-Institut (HHI)

Einsteinufer 37, 10587 Berlin, Germany

Email: [cooke@hhi.de](mailto:cooke@hhi.de), [schreer@hhi.de](mailto:schreer@hhi.de), [pasew@hhi.de](mailto:pasew@hhi.de), [kauff@hhi.de](mailto:kauff@hhi.de)

## Abstract

The proposed approach is motivated by applications which allow user navigation and individual viewpoint specification in shared virtual environments with telepresence quality. In this context, we present a synthesis method for arbitrary virtual views in a multi-view camera set-up. This method generates a close to real-time, view adaptable reconstruction of a 3-dimensional, (3D), object taken from at least two cameras. In this method, we use the recently developed *Incomplete 3D*, (IC3D), technique, a disparity-based multiview representation for a weakly convergent camera setup in combination with the trilinear warping functions to describe new virtual camera positions. Previously, only virtual views inside the baseline could be described by IC3D. Hence, user movement in virtual environments (VE) was restricted. In order to create a virtual camera position and orientation outside the baseline we apply point correspondences across the two reference images to a trilinear tensor, built from the fundamental matrix between the reference views. The trilinearities provide a general warping function from the reference images to virtual view that is governed directly by the virtual camera parameters.

## 1 Introduction

Currently, one can observe an increase of R&D activities in the fields of applications

integrating seamlessly arbitrary-shaped natural video objects into VEs. This development is due to the promotion of new international standards such as MPEG-4, and VRML and the rapid growth of video communication, entertainment, and multimedia systems willing to exploit them. Against this background it would be most desirable to model natural, arbitrarily shaped, video objects as 3D objects. However former investigations have shown that full 3D modelling is too computationally expensive for a process to allow realtime. In contrast, a simple texture mapping of video on a 2-dimensional, (2D), node, as it is often used in VE applications produces inconsistencies in the view adaptation during navigation in the scene. In fact, the 3D nature of a VE implies scene navigation and therefore viewpoint adaptation of these objects is crucial. Typical examples of such applications are video conferencing systems providing motion parallax and immersive telepresence applications [1][2][3].

To overcome these shortcomings, this paper presents significant extensions of IC3D technique which is based on methods of intermediate view interpolation and which has recently been proposed by HHI [12][13][14]. Section 2 briefly discusses the pro's and con's of the conventional technique. Then, section 3 reviews the original baseline IC3D approach and section 4 explains some of its limitations with respect to novel view synthesis. Afterwards, section 5 describes the theory of trilinear warping and section 6 presents the IC3D extension towards more flexibility in terms of synthesis by combining it with trilinear warping. Finally

section 7 and 8 present experimental results and some conclusions.

## 2 3D Representation of Video Objects

The key problem of these applications is the reconstruction of the 3D shape of natural objects from multi-viewpoint video signals. Two commonly used approaches are:

- *3D Modeling*: 3D models consist of deformable surface meshes or wireframes containing a set of adjacent elementary planar patches and are commonly used to describe surfaces with a desired precision. This can be achieved in principle as long as multiple camera views are available and the respective camera positions are known [4][5]. Resulting 3D wire-frame representations of natural video objects can for example be encoded by using 3D mesh structures as foreseen for MPEG-4 [6]. The flexibility of such models, their efficiency in computing, storage and coding are the most important advantages of this triangulation-based surface approximation. However three drawbacks are associated with 3D models created from 2D views from different perspectives:

- they are often incapable of properly reflecting the physical surface characteristics of the object, because the nodes and edges of the mesh generally have no or only limited physical significance;
- around surface areas of high local curvature, a lot of small triangles are required to approximate the surface to the desired accuracy;
- most reported methods suffer from extremely large algorithmic complexity.

In most time critical applications the complexity of generating 3D models generally overloads the realtime capability of the rendering engine. To avoid these problems several authors considered model-based methods tailored for special situations, e.g., face and human body models for videoconferencing applications [7]. These methods assume *a priori* knowledge about the object to be modelled

[8][9]. Although model-based methods work well for the specified objects these techniques obviously reduces the visual realism as generic representations are used for each participant.

- *Intermediate Viewpoint Interpolation* : In this approach disparities are estimated from adjacent camera views, and an intermediate view is generated by disparity-compensated interpolation from the original views [10][11][12]. To extract objects, it is sufficient to apply a conventional 2D segmentation technique to the separate camera views. In intermediate image interpolation, it is generally necessary to encode all image views separately, or to separately employ a technique exploiting the multiview redundancy, like the MPEG-2 multiview profile. However, disparity data derived for optimum encoding are often not appropriate for viewpoint interpolation [13].

## 3 The Incomplete 3D Technique

IC3D is a disparity-based multiview data representation that was developed here at HHI in the context of MPEG-4. This incompleteness is two-fold:

- the technique does not retain the full pixel representation of all the views available, thus resulting in higher compression;
- it does not perform full 3D modeling analysis, with the advantage of a simplified complexity.

The general concept is to limit the number of pixels that have to be encoded, by analysis of the correspondences between the particular views available, such that for an object, each area that is visible within more than one camera view is encoded only once with the highest possible resolution. If the disparity correspondences are estimated from the original views, it is straightforward to reconstruct all the areas that were excluded from encoding by use of disparity compensated projection, Fig. 1. A synopsis of the theoretical background of IC3D follows, further information can be found in [14].

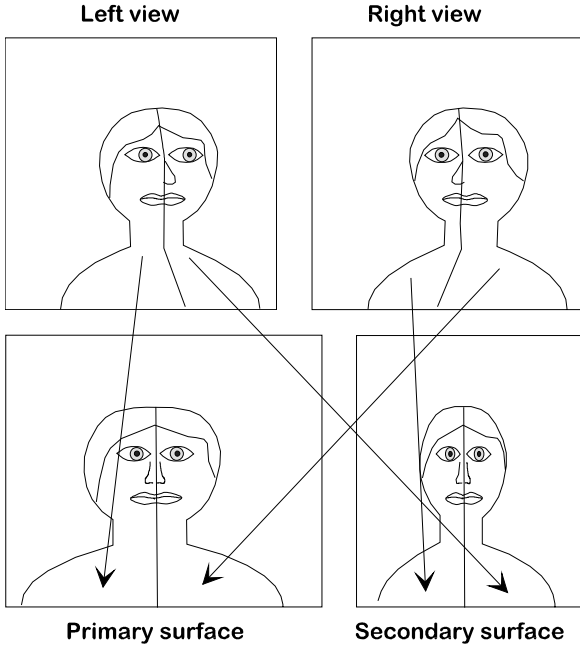


Fig. 1: Generation of primary and secondary surfaces

Using a pinhole camera model, taking object space to be 3D projective space  $\mathcal{P}^3$  and image space to be 2D  $\mathcal{P}^2$ , the projection of a point  $P=(X, Y, Z, 1) \in \mathcal{P}^3$  in a 3D world to a point  $p=(x, y, 1) \in \mathcal{P}^2$  in the camera's 2D image plane is given by the central projection equation

$$x = F \frac{X}{Z}, \quad y = F \frac{Y}{Z} \quad (1)$$

where  $F$  is the focal length of the camera, the optical center is assumed to be at world coordinate  $(0,0,0,1)$ , and the image plane  $(x,y,1)$  is parallel to the  $(X,Y)$ -plane with origin at  $(0,0,F,1)$ .

Assume that two points on the object's surface  $P_1 = (X_1, Y_1, Z_1, 1)$  and  $P_2 = (X_2, Y_2, Z_2, 1)$  become visible in the image plane of both of the cameras at positions  $x_1$  and  $x_2$ , Fig. 2. The distances between observed point positions  $x_1$  and  $x_2$  will deviate between the different camera image planes (we limit this analysis to the horizontal distances, which is reasonable if the cameras have parallel optical axis's).

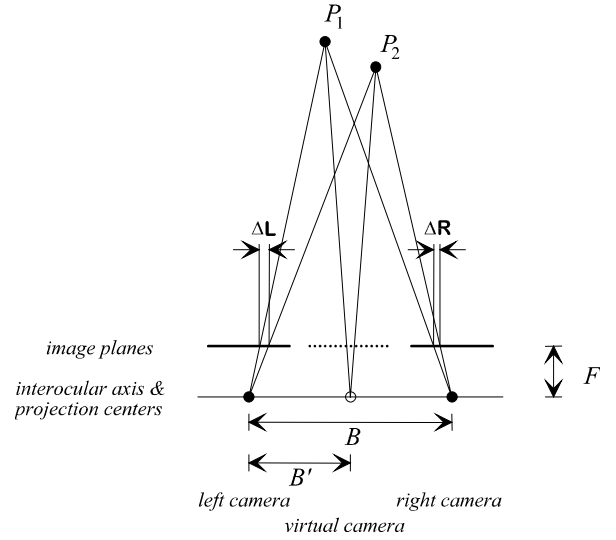


Fig. 2: Correspondences in image planes using two cameras

For example, in a stereoscopic pair of left,  $L$ , and right,  $R$ , cameras we come to the following relations:

$$\begin{aligned} \Delta L &= x_{L,2} - x_{L,1} = F \left[ \frac{X_2}{Z_2} - \frac{X_1}{Z_1} \right] \\ \Delta R &= x_{R,2} - x_{R,1} = F \left[ \frac{x_2}{Z_2} - \frac{x_1}{Z_1} \right] + FB \left[ \frac{1}{Z_1} - \frac{1}{Z_2} \right] \end{aligned} \quad (2)$$

which leads to a gradient of disparities indicating the correspondences between this image pair:

$$\Delta d = \Delta L - \Delta R = FB \left[ \frac{1}{Z_2} - \frac{1}{Z_1} \right] \quad (3)$$

where  $B$  is the baseline between the two cameras. From (3), and assuming we are working with convex only objects in the 3D world, we can draw the following conclusions:

- If both points  $P_1$  and  $P_2$  have the same depth position,  $Z$ ,  $\Delta d$  becomes zero, which means at the same time, that the area between these points is equally visible in both camera views;
- If  $\Delta d$  is positive,  $\Delta L > \Delta R$ , which means that the area is better visible in the left camera view;

- If  $\Delta d$  is negative,  $\Delta R > \Delta L$ , which means that the area is better visible in the right camera view.

Here, *better visibility* implies that more samples are available for the area between points  $P_1$  and  $P_2$  in one of the camera's image planes. It is obvious the disparity data can be used to reconstruct other views, e.g. by performing interpolation, extrapolation of the pixels from the available views. It follows from (3), that the disparity  $d$  is proportional to the baseline distance  $B$ :

$$d = \frac{FB}{Z} \quad (4)$$

This implies, that it is possible to generate an artificial camera viewpoint with an artificially changed baseline position  $B'$ , if a scaled disparity value  $d'$  is used in the projection. We obtain:

$$d' = \upsilon d, \quad B' = \upsilon B \quad (5)$$

where  $\upsilon$  is the scaling factor. The effect is a simulation of an alternative camera view, which would lie anywhere on the line that interconnects the two camera's optical centers such that:

- the original area of width  $\Delta L$  in the left camera view is mapped to an area of width  $\Delta L - \Delta d'$  in the artificial camera view at  $(0, 0, B', 1)$ ;
- the original area of width  $\Delta R$  in the right camera view is mapped to an area of width  $\Delta R + \Delta d'$  in the artificial camera view at  $(0, 0, B', 1)$ .

To generate the IC3D representation, one single unwrapped texture of a video object's 3D surface is extracted from the multiple camera views. By analysis of the disparity gradient we determine areas which are best visible from particular camera positions. These areas are called the *areas of interest* (AOI) of the individual cameras. Examples of AOI are the areas of the 'left view' and 'right view' that are combined to create the 'primary surface' in Fig. 1. The secondary surface is only an auxiliary

plane consisting of the transition between the AOI at the stitching boarder (separation lines in Fig. 1). The sizes  $\Delta L$  and  $\Delta R$  are left unchanged in the left and right AOI, respectively. In order to reconstruct different viewpoints for the IC3D texture surface, disparity-controlled projection is performed from the texture data within the particular AOI, towards a view plane with the virtual camera position at  $(0, 0, B', 1)$ .

## 4 IC3D Restrictions

Inherent to the IC3D process is a parallel camera setup. This setup allows us to express the geometric relationship between the two camera coordinate systems by a simple translation along the baseline, as indicated in (4). This relationship is based on point correspondences along the x-axis. The synthesis was restricted to positions on the baseline, resulting in fast algorithms and providing high quality synthesis results. Hence, integration of the IC3D algorithm into a VE causes user navigation restrictions, meaning arbitrary multi-view synthesis is not possible.

However convergent camera set-up's are required for advanced applications such as video conferencing scenarios due to the small distance of the object related to the camera system. This leads to a general disparity estimation method using the epipolar constraint, implying that the simple translation along the baseline can no longer be used and a new synthesis procedure is required.

In general virtual view synthesis is possible using epipolar constraints, [15], however the view synthesis is subject to epipolar singularities under certain camera motions, i.e. when the virtual camera center is collinear with the reference cameras the epipolar lines do not intersect. While singularities can be controlled using depth maps, the generation of reliable maps for non-synthetic objects is subjective. In order to overcome these shortcomings we concentrate instead on the use of concatenating trilinear warping functions to provide view synthesis from arbitrary virtual viewpoints.

## 5 Trilinear Warping

It has been shown in [16] that any three perspective views of a scene satisfy a pair of trilinear functions of image co-ordinates. Using the trilinear result one can manipulate views of an object to synthesise images that are far away from the viewing positions of the sample reference images without 3D reconstruction. The following is an explanation of how the trilinear functions of the image coordinates across three views are obtained.

We assume two image views  $\psi_1$  and  $\psi_2$  and an arbitrary point  $P \in \mathcal{P}^3$  with corresponding image points  $p = (x, y, I)$  and  $p' = (x', y', I)$ , where  $p \in \psi_1$  and  $p' \in \psi_2$ . Let  $A$  be a  $3 \times 3$  homography mapping of  $\psi_1 \rightarrow \psi_2$  due to some plane  $\pi$ .  $A$  is scaled to satisfy  $p'_o \cong A p_o + v'$ , where  $p_o \in \psi_1$  and  $p'_o \in \psi_2$ ; both points coming from an arbitrary point  $P_o$  which is not on  $\pi$ ; and  $\cong$  means equal up to a scale factor. Then, every corresponding pair  $p \in \psi_1$  and  $p' \in \psi_2$  obeys (6), where  $v'$  represents a constant vector depending on the epipole of  $\psi_1$  in  $\psi_2$  as well as the distance and orientation of plane  $\pi$ :

$$p' \cong Ap + \lambda v' \quad (6)$$

The scalar  $\lambda$  is the ‘relative affine invariant’ mentioned in [16], here we will not discuss how  $\lambda$  is recovered.

We now introduce a third view  $\psi_3$ , such that  $p'' \in \psi_3$ , and  $v'' \in \psi_3$  is a constant vector depending on plane  $\pi$  and the epipole between  $\psi_1$  and  $\psi_3$ . Since (6) holds for any plane we choose some plane  $\pi$ , and let  $A, B$  be the homographies  $\psi_1 \rightarrow \psi_2$  and  $\psi_1 \rightarrow \psi_3$  respectively. Then for every point  $p \in \psi_1$  with corresponding points  $p' \in \psi_2, p'' \in \psi_3$  there exists a scalar  $\lambda$  such that  $p' \cong Ap + \lambda v'$  and  $p'' \cong Bp + \lambda v''$ . If we solve for  $\lambda$  and equate the results we obtain the trilinear functions of the image coordinates across three views. Nine such functions exist, implying at most 27 distinct coefficients. In [16] it is shown that at most four of these trilinearities are linearly independent:

$$x'' \alpha_{13}^T p - x'' x' \alpha_{33}^T p + x' \alpha_{31}^T p - \alpha_{11}^T p = 0 \quad (7)$$

$$y'' \alpha_{13}^T p - y'' x' \alpha_{33}^T p + x' \alpha_{32}^T p - \alpha_{12}^T p = 0 \quad (8)$$

$$x'' \alpha_{23}^T p - x'' y' \alpha_{33}^T p + y' \alpha_{31}^T p - \alpha_{21}^T p = 0 \quad (9)$$

$$y'' \alpha_{23}^T p - y'' y' \alpha_{33}^T p + y' \alpha_{32}^T p - \alpha_{22}^T p = 0 \quad (10)$$

Where  $a_{ij} = v'_i b_j - v''_j a_i$  and  $b_1, b_2, b_3$  and  $a_1, a_2, a_3$  are the row vectors of  $A$  and  $B$  and  $v = (v_1, v_2, v_3)$ . These trilinearities allow us to generate a new view,  $\psi_3$ , consisting of points  $(x'', y'')$ , by using the correspondences  $p, p'$  across the two image views  $\psi_1$  and  $\psi_2$ . In [17] Hartley encapsulates these 27 coefficients into a trilinear tensor. In [18] this tensor is used as an operator that describes the transformation from a given tensor of three views to a novel tensor of a new configuration of three views.

## 6 IC3D Extension

The IC3D algorithm has point correspondences between two views and no initial third view, yet the trilinear warping functions build a relationship of points across three views. Hence, in order to obtain an initial correspondence for our setup we specify our initial virtual view to coincide with one of the reference images. We then build a trilinear relationship, the trilinear tensor, across these views. Shashua calls this tensor a *seed* tensor and explains how it can be calculated using the fundamental matrix in the case of only having two reference images; interested readers are referred to [18].

Since an IC3D extension can be applied to both strongly and weakly calibrated systems we can either compute or estimate the fundamental matrix between the two reference images. As described in [18] we use this fundamental matrix to build the seed tensor between three views, in which views two and three coincide. While the rank of this tensor is 2 in comparison to rank of 4 for a tensor of three distinct views all other properties remain the same. Further details can be found in [19].

Once this initial relationship is calculated it can be modified in order to describe a new

configuration of the three views. We only want to modify one of the three camera positions, our virtual view. By repeatedly applying virtual camera transformations on the seed tensor we obtain a chain of warping functions which can be applied to the reference images to create the desired virtual images. Since the tensor operator is based on the set of trilinearities described in the previous section we do not go into a detailed explanation of tensors, the relevant information can be found in [18].

Each of the four trilinear equations describe a matching between the point  $p$  in  $\psi_1$ , some line passing through the matching point  $p'$  in  $\psi_2$  and some line passing through  $p''$  in  $\psi_3$ . In 3D space this corresponds to an intersection between a ray and two planes. Under ideal intrinsic camera parameters in the two reference cameras this intersection would be at one point. However, due to imperfections in digital image sampling the pixel image positions of  $P$  do not always correspond to the expected position of the 3D space point. These discrepancies lead to differences in the location of the intersection. In order to obtain the most accurate pixel position of  $p''$  the choice of the position of the aforementioned lines through  $p'$  and  $p''$  is critical.

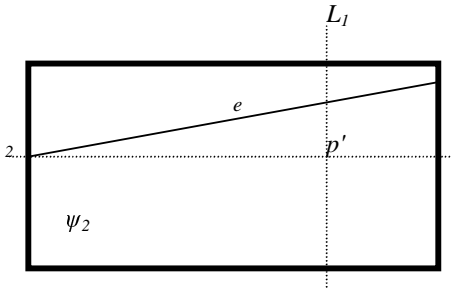


Fig. 3:  $L_1$  and  $L_2$  are the horizontal and vertical planes respectively,  $e$  is epipolar line.

We specify two coincident lines, horizontal,  $L_1$ , and vertical,  $L_2$ . By examining the slope,  $n$ , of the epipolar line,  $e$ , of  $p$  in  $\psi_2$  we decide which choice of planes to take, Fig. 3. Using the following rule:

- $n < 45^\circ$  :  $p''$  defined by vertical planes
- $n > 45^\circ$  :  $p''$  defined by horizontal planes

In Fig. 3 the point of intersection of  $L_1$  with  $e$  provides a better estimation of  $p'$  than that of  $L_2$  so vertical lines and hence planes are chosen. The problem of forward mapping in the virtual view is overcome by dividing the reference images into rectangles whose corners are mapped using perspective transformations onto quadrilaterals in the virtual images; then computing a backward map for all the pixels in this quadrilateral [20].

## 7 Experimental Results

In the following figures we present the results of our extension of IC3D on image pairs captured using both strongly and weakly convergent camera setups. The examples CLAUDE, Fig. 4, and DOLL, Fig. 5, are CCIR images captured using two cameras and a strongly calibrated setup.



Fig. 4: CLAUDE – original images.



Fig. 5: DOLL – Original images.

CLAUDE is a weakly convergent camera setup of 8 degrees with a baseline of 0.3 meters, while DOLL is a strongly convergent setup of 16 degrees and a baseline of 0.2 meters. The first step in the process is to find point correspondences through a disparity estimation process. Then, as stated in the previous section, a seed tensor, initially based on the fundamental matrix between the camera pair, is

created. This tensor assumes that the initial virtual view coincides with one of the reference cameras. Since the cameras are stationary the seed tensor need only be computed once for each pair. Using the trilinearity functions, the point correspondences, and a user specified virtual camera transformation we calculate the reprojected virtual view.

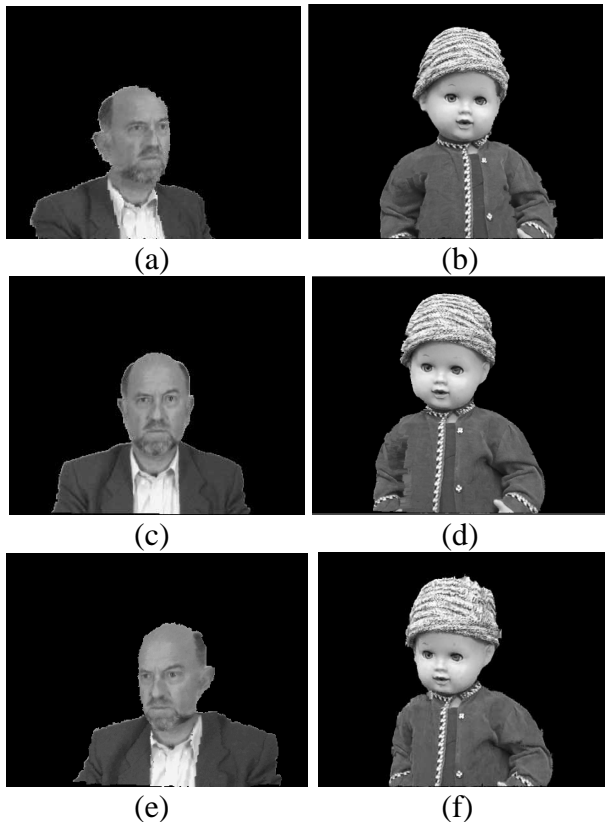


Fig. 6: CLAUDE and DOLL synthesised images.

The results in Fig. 6 show navigation of the virtual camera outside the baseline from left to right. Fig. 6 (c) and (d) show *head on* synthesised views of CLAUDE and DOLL respectively. Fig. 6 (a) and (e) show CLAUDE at a position of 18 degrees around the y-axis and 6 around the x-axis either side of the middle view; while Fig. 6 (b) and (f) show DOLL at a position of 16 degrees around the y-axis and 3 around the x-axis either side of the middle view.

The presented synthesis result demonstrates the efficiency of the proposed method.

## 8 Conclusion

We have discussed the challenges of virtual view synthesis in relation to the integration of natural, arbitrarily shaped, video objects in virtual environments. Using the IC3D approach we have shown how an efficient multiview data representation can be created; and described how the current synthesis method limits the virtual view to the baseline and hence restricts user navigation in the VE. We then identified how trilinear warping functions can be used to solve point correspondences across three arbitrarily positioned views. Recovering the fundamental matrix between the reference images we created an initial trilinear relationship across the two reference images and a virtual view which initially coincides with one of the reference images. This initial relationship is used as an operator to build new virtual views. Transformations of the virtual camera combined with this operator and the reference image point correspondences can be used to create a new virtual view. Concatenating the results of these warping functions one can manipulate views of an object to synthesise virtual views that are far outside the baseline of the sample reference images. Finally we presented results of this new extension to IC3D on images of CCIR size, for both weakly and strongly convergent camera systems, for arbitrary virtual camera positions outside the baseline.

## 9 Acknowledgements

This study is supported by the Ministry of Science and Technology of the Federal Republic of Germany, Grant-No.01BN701/1.

## References

- [1] P. Kauff et al: „The Virtual Meeting Room: A Realtime Implementation of a Shared Virtual Environment System Using Today's Consumer Technology in

- Connection with the MPEG-4 Standard“, *Presence 2000 3<sup>rd</sup> International Workshop on Presence*, Delft, Netherlands, March 2000.
- [2] E. Cooke et al: „Realtime View Adaptation of Video Objects in 3-Dimensional Virtual Environments“, *Presence 2000 3<sup>rd</sup> International Workshop on Presence*, Delft, Netherlands, March 2000.
- [3] J. Stone, M. David: „VIRTUE - High realism telepresence conferencing“, *Presence 2000 3<sup>rd</sup> International Workshop on Presence*, Delft, Netherlands, March 2000.
- [4] E. Izquierdo, X. Feng: „Image-Based 3D Modeling of Arbitrary Natural Objects“, *VLBV Workshop 1998*, Urbana, Illinois, USA, October 1998.
- [5] E. Izquierdo, X. Feng: „Modeling of Arbitrary Objects Based on Geometric Surface Conformity“, *IEEE Transactions on CSVT – Special Issue on SNHC*, 1999.
- [6] MPEG-4 Video Group: „MPEG-4 Video Verification Model“, *IEEE Transactions on CSVT – Special Issue on SNHC*, Melbourne, Australia, 1999.
- [7] *Proc. of International Workshop on SNHC and 3D Imaging*, Rhodes, Greece, 1997.
- [8] M. Kampmann, J. Ostermann: „Automatic Adaptation of a Face Model in a Layered Coder with an Object-Based Analysis Synthesis Layer and a Knowledge Based Layer“, *Signal Processing: Image Communication*, 1997.
- [9] R. Koch: „Adaptation of a 3D Facial Mask to Human Faces in Videophone Sequences using Model Based Image Analysis“, *Picture Coding Symposium*, 1991.
- [10] E. Chen, L. Williams: „View Interpolation for Image Synthesis“, *Proc. ACM SIGGRAPH’93*, 1993.
- [11] T. Werner et al: „Rendering Real-World Objects using View Interpolation“, *Proc. IEEE Int. Conf. Computer Vision*, Boston, 1995.
- [12] J.-R. Ohm, E. Izquierdo: “An Object-Based System for Stereoscopic Viewpoint Synthesis”, *IEEE Transactions on CSVT*, 1997.
- [13] J.-R. Ohm, K. Müller: “Incomplete 3D for Multiview Representation and Synthesis of Video Objects”, *ECMAST98*, 1998.
- [14] J.-R. Ohm, K. Müller, S. Ekmeci, “Incomplete 3D – A new Technique for Multiview Data Representation”, *Proc. Image and Multidimensional Digital Signal Processing (IMDSP’98)*, Alpbach, Austria, pp. 311-314, 1998.
- [15] S.Laveau and O.D. Faugeras: „3-D Scene Representation as a Collection of Images“, *Proc. Int’l Conf. Pattern Recognition*, 1994.
- [16] A. Shashua: „Algebraic Functions for Recognition“, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1995.
- [17] R. Hartley: „Lines and Points in Three Views – A Unified Approach“, *Proc. ARPA Image Understanding Workshop*, 1994.
- [18] S. Avidan, A. Shashua: „Novel View Synthesis by Cascading Trilinear Tensors“, *IEEE Transactions on Visualization and Computer Graphics*, Oct - Dec 1998.
- [19] S. Avidan, A. Shashua: „Tensorial Transfer: On the Representation of  $n > 3$  Views of a 3D Scene“, *Proc. ARPA Image Understanding Workshop*, 1996.
- [20] G. Wolberg: *Digital Image Warping*. Los Alamitos, Calif.: IEEE CS Press, 1992.