

CONTINUOUS PARALLAX ADJUSTMENT FOR 3D-TV

A. Colombari, A. Fusiello and V. Murino

Università degli studi di Verona

Dipartimento di Informatica

Strada Le Grazie, 15 - 37134 Verona, Italy

colombari@sci.univr.it, {andrea.fusiello,vittorio.murino}@univr.it

Keywords: 3D-TV, image-based rendering, uncalibrated view synthesis, plane+parallax, relative affine structure.

Abstract

This paper presents a method to continuously adjust the parallax in 3D-TV visualization. It is based on a generic framework for novel view synthesis from two uncalibrated reference views that allows to move a virtual camera along a path that is obtained starting from the epipolar geometry of the reference views. The scene is described by its relative affine structure from which novel views are extrapolated and interpolated. The main contribution of this paper is an automatic method for specifying virtual camera locations in an uncalibrated setting. Experiments with synthetic and real images illustrate the approach.

1 Introduction

Three-dimensional television (3D-TV) is expected to be the next revolution in the history of television [12]. In the last years, glasses-free 3D-TV has been brought onto the market. 3D-TV is also getting more and more popular in the research community, as it starts to appear in the list of topics of many conferences. For example, SIGGRAPH 2004 [19] dedicated a panel discussion to this emerging technology.

Stereoscopic visualization in 3D-TV is based on producing two separate video streams, one for each eye. The display have small lenses in front of each pixel, allowing different images to be seen depending on the point of view. In this way, software can calculate different images to be sent to the viewer's left and right eyes. In order to avoid viewer's discomfort, the amount of parallax encoded in the stereo pair must be adapted to the viewing condition. The idea is that the viewer might use a "3D-ness" knob [9] to continuously adjust the stereoscopic separation. This entails the ability to synthesize novel views of the scene as taken from an arbitrary virtual point of view. If the depth of the scene points is known, 3D image warping can be used as in [13, 3]. In general, one would like to be able to synthesize novel views in an uncalibrated setting, i.e., without knowing the depth of the points, or equivalently, without knowing the intrinsic parameters of the camera.

Uncalibrated view-synthesis [10, 18, 7, 4] offers a solution that does not require the reconstruction of the full scene structure, but only the estimation of disparities. The contribution of this

paper is an automatic method for specifying the virtual viewpoint in an uncalibrated setting, based on the interpolation and extrapolation of the epipolar geometry linking the reference views. The virtual cameras are positioned on a path as if the real camera continued with the same motion as between the two reference views (see Figure 1).

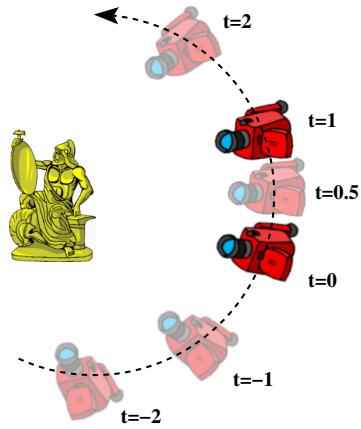


Figure 1: Parallax adjustment. The position of the virtual camera along a path is specified by a parameter t , which corresponds to the "3D-ness" knob. Camera locations at $t = 0$ and $t = 1$ correspond to the reference frames.

In the case of calibrated cameras, view synthesis algorithms based on image interpolation yield satisfactory results [14, 16, 3]. Where no knowledge on the imaging device can be assumed, uncalibrated point transfer techniques utilize image-to-image constraints, such as the Fundamental matrices [10], trilinear tensors [2], plane+parallax [6], to re-project pixels from a small number of reference images to a given view. Another way of linking corresponding points is the *relative affine structure* [18], a close relative of the plane+parallax. This is the framework in which our technique is embedded.

Although uncalibrated point transfer algorithms are well understood, what prevent them to be applied in real-world applications, is the lack of a "natural" way of specifying the position of the virtual camera in the familiar Euclidean frame, because it is not accessible. Everything is represented in a projective frame that is linked to the Euclidean one by an *unknown* projective transformation. All the view-synthesis algorithms requires either to manually input the position of points in the synthetic view, or to specify some projective elements.

In this work, we will consider the case of interpolation and extrapolation from two uncalibrated reference views. We propose a solution to the specification of the new viewpoints, based on the exploitation of the epipolar geometry that links the reference views, represented by the homography of the plane at infinity and the epipole. Thanks to the Lie group structure of these *uncalibrated rigid transformations*, interpolation and extrapolation is possible using matrix exponential and logarithm. The proposed technique allows to synthesize physically-valid views, and in this sense it can be seen as a generalization to the uncalibrated case of [16]. The framework for interpolation of Euclidean transformations was set forth in [1], whereas the idea of manipulating rigid displacements at the uncalibrated level is outlined in [15], where it is applied to rotations only.

The rest of the paper is structured as follows. In Section 2, we review the theory necessary to make the paper self-consistent. Section 3 describes our approach for specifying virtual viewpoints in an uncalibrated setting. Experimental results concerning synthetic and real scenes are shown and commented in Section 4, and conclusions are drawn in Section 5.

2 Background

We start by giving some background notions needed to understand our method. The geometry of multiple view is dealt with exhaustively in [8]. A complete discussion on the relative affine structure theory can be found in [18].

As shown in Figure 2, two points \mathbf{m}_1 and \mathbf{m}_2 that are the projection of the same 3-D point M onto the first and the second camera, respectively, are said to be conjugate points.

Given a plane Π , with equation $\mathbf{n}^T M = d$, two conjugate points \mathbf{m}_1 and \mathbf{m}_2 are related by

$$\mathbf{m}_2 \simeq H_{12}\mathbf{m}_1 + \mathbf{e}_{21}\gamma \quad (1)$$

where H_{12} is the collineation induced by the plane Π and \mathbf{e}_{21} is the epipole in the second view. The symbol \simeq means equality up to a scale factor. If the 3D point M belongs to Π , then \mathbf{m}_1 and $H_{12}\mathbf{m}_1$ are a conjugate pair. Otherwise, there is a residual displacement, called *parallax*. This quantity is proportional to the *relative affine structure* of M [18]

$$\gamma \triangleq \frac{a}{d\zeta_1}$$

where a is the orthogonal distance of the 3-D point M to the plane Π and ζ_1 is the distance of M from the focal plane of the first camera. Points \mathbf{m}_2 , $H_{12}\mathbf{m}_1$ and \mathbf{e}_{21} are collinear. The parallax field is a radial field centered on the epipole.

Since the relative affine structure is independent of the second camera, arbitrary “second views” can be synthesized, by giving a plane homography and an epipole, which specify the position and orientation of the virtual camera in a projective framework. The view synthesis algorithm that we employ, inspired by [18], is the following:

- A. given a set of conjugate pairs

$$(\mathbf{m}_1^k, \mathbf{m}_2^k) \quad k = 1, \dots, m$$

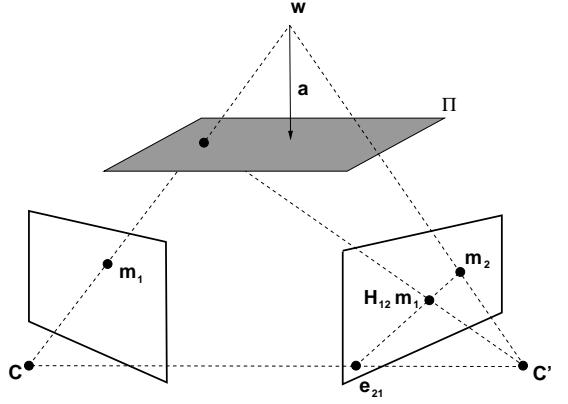


Figure 2: Parallax is the segment connecting $H_{12}\mathbf{m}_1$ with \mathbf{m}_2 .

- B. recover the epipole \mathbf{e}_{21} and the homography H_{12} up to a scale factor;
- C. choose a point \mathbf{m}_1^0 and scale H_{12} to satisfy

$$\mathbf{m}_2^0 \simeq H_{12}\mathbf{m}_1^0 + \mathbf{e}_{21}$$

- D. compute the relative affine structure γ^k from (1):

$$\gamma^k = \frac{(\mathbf{m}_2^k \times \mathbf{e}_{21})^T (H_{12}\mathbf{m}_1^k \times \mathbf{m}_2^k)}{\|\mathbf{m}_2^k \times \mathbf{e}_{21}\|^2}. \quad (2)$$

- E. specify a new epipole \mathbf{e}_{31} and a new homography H_{13} (properly scaled);
- F. transfer points in the synthetic view with

$$\mathbf{m}_3^k \simeq H_{13}\mathbf{m}_1^k + \mathbf{e}_{31}\gamma^k \quad (3)$$

The problem that makes this technique difficult to use in practice (and for this reason it has been overlooked for view synthesis) is point E, namely that one has to specify a new epipole \mathbf{e}_{31} and a new (scaled) homography H_{13} . In Section 3 we will present an automatic solution to this problem.

3 Specifying the virtual camera position

Our idea is based on the replication of the unknown rigid displacement G_{12} that links the reference views, I_1 and I_2 . The method described in this section will allow us to render a view I_3 from a pose $G_{13} = G_{12}G_{12} = (G_{12})^2$. More in general, thanks to the group structure, this will extends to to any scalar multiple of G_{12} ,

3.1 The group of uncalibrated rigid displacements

Let us consider Eq. (1), which express the epipolar geometry with reference to a plane, in the case of view pair 1-2:

$$\frac{\zeta_2}{\zeta_1} \mathbf{m}_2 = H_{12}\mathbf{m}_1 + \mathbf{e}_{21}\gamma_1 \quad (4)$$

and view pair 2-3:

$$\frac{\zeta_3}{\zeta_2} \mathbf{m}_3 = H_{23} \mathbf{m}_2 + \mathbf{e}_{32} \gamma_2. \quad (5)$$

In order to obtain an equation relating view 1 and 3, let us substitute the first into the second, obtaining:

$$\frac{\zeta_3}{\zeta_1} \mathbf{m}_3 = H_{23} H_{12} \mathbf{m}_1 + (H_{23} \mathbf{e}_{21} + \mathbf{e}_{32} \frac{d_1}{d_2}) \gamma_1 \quad (6)$$

By comparing this equation to Eq. (1), we obtain:

$$\mathbf{e}_{31} = H_{23} \mathbf{e}_{21} + \mathbf{e}_{32} \frac{d_1}{d_2} \quad (7)$$

The ratio $\frac{d_1}{d_2}$ in general is unknown, but if Π is the plane at infinity then $\frac{d_1}{d_2} = 1$ (please note that this is approximately true for planes distant from the camera). Therefore, taking the plane at infinity as Π , from Eq. (6) we obtain:

$$\begin{aligned} H_{\infty 13} &= H_{\infty 23} H_{\infty 12} \\ \mathbf{e}_{31} &= H_{\infty 23} \mathbf{e}_{21} + \mathbf{e}_{32} \end{aligned} \quad (8)$$

In matrix form Eq. (8) writes:

$$D_{13} = D_{23} D_{12} \quad (9)$$

where

$$D_{ij} \triangleq \begin{bmatrix} H_{\infty ij} & \mathbf{e}_{ji} \\ \mathbf{0} & 1 \end{bmatrix} \quad (10)$$

represents a *rigid displacement at the uncalibrated level*¹.

We then plug D_{13} as defined above in the transfer equation (Eq. (3)) that re-writes:

$$\mathbf{m}_3^k \simeq D_{13} \begin{bmatrix} \mathbf{m}_1^k \\ \gamma_1^k \end{bmatrix} \quad (11)$$

We will now prove that the virtual view I_3 obtained from the above equation is rendered from a pose $G_{13} = G_{23} G_{12}$. Let

$$G_{ij} \triangleq \begin{bmatrix} R_{ij} & \mathbf{t}_{ij} \\ \mathbf{0} & 1 \end{bmatrix} \quad (12)$$

be a matrix that represents a rigid displacement, where R is a rotation matrix and \mathbf{t} is a vector representing a translation. Rigid displacements form a group, known as the special Euclidean group of rigid displacements in 3D, denoted by $SE(3)$. Each uncalibrated displacement D_{ij} is the conjugate of an element $G_{ij} \in SE(3)$ by the matrix $\tilde{K} = \begin{bmatrix} K & \mathbf{0} \\ \mathbf{0} & 1 \end{bmatrix}$:

$$\begin{aligned} D_{ij} &= \begin{bmatrix} KR_{ij}K^{-1} & K\mathbf{t}_{ij} \\ \mathbf{0} & 1 \end{bmatrix} \\ &= \begin{bmatrix} K & \mathbf{0} \\ \mathbf{0} & 1 \end{bmatrix} \begin{bmatrix} R_{ij} & \mathbf{t}_{ij} \\ \mathbf{0} & 1 \end{bmatrix} \begin{bmatrix} K^{-1} & \mathbf{0} \\ \mathbf{0} & 1 \end{bmatrix} \end{aligned}$$

¹Technically, since we assume to know the plane at infinity, this corresponds to the affine calibration stratum [11].

The conjugacy (or similarity) mapping is an homomorphism of $SE(3)$, for it preserves the product:

$$\begin{aligned} D_{13} &= D_{23} D_{12} = \tilde{K} G_{23} \tilde{K}^{-1} \tilde{K} G_{12} \tilde{K}^{-1} \\ &= \tilde{K} G_{23} G_{12} \tilde{K}^{-1} = \tilde{K} G_{13} \tilde{K}^{-1}. \end{aligned} \quad (13)$$

This proves our thesis and also points out the conjugacy relationship between $SE(3)$ and the group of uncalibrated displacement.

3.2 Extrapolation and interpolation

Let us focus on the problem of specifying the virtual camera's viewpoint. Please note that if intrinsic parameters are constant, the scale factor of $H_{\infty 12}$ is fixed, since $\det(H_{\infty 12}) = 1$ (see [11]). So, point C in the general view synthesis procedure must be replaced with

C. scale $H_{\infty 12}$ such that $\det(H_{\infty 12}) = 1$.

As for point E., please note that formulae in (8) hold with the equality sign, hence there are no free scale factors to fix.

In the case of synthesis from two views, we know only D_{12} and want to specify D_{13} to be used in the transfer equation to synthesize the 3rd view. The replication trick consists in setting $D_{23} = D_{12}$, i.e., $D_{13} = (D_{12})^2$ thereby obtaining a novel view from a virtual camera placed at $(G_{12})^2$ with respect to the first camera. Likewise, $(D_{12})^n \forall n \in \mathbb{Z}$ corresponds to the rigid displacement $(G_{12})^n$.

Integer exponents provide us with an extrapolation scheme by discrete steps. However, $SE(3)$ is also a differentiable manifold (being a Lie group), in which we can make sense of the interpolation between two elements as drawing the geodesic path between them. Let us consider, without loss of generality, the problem of interpolating between the element G and the identity I . The geodesic path leaving the identity can be obtained as the projection of a straight path in the tangent space, and the logarithm map precisely projects a neighborhood of I into the tangent space to $SE(3)$ at I . A straight path in the tangent space emanating from 0 is mapped onto a geodesic in $SE(3)$ emanating from I by the exponential map. Hence, the geodesic path in $SE(3)$ joining I and G is given by

$$G^t \triangleq \exp(t \log(G)), \quad t \in [0, 1]. \quad (14)$$

More in general, we can define a *scalar multiple of rigid transformations* [1]:

$$t \odot G \triangleq G^t = \exp(t \log(G)), \quad t \in \mathbb{R}. \quad (15)$$

Mimicking the definition that we have done for rigid transformations, let us define

$$t \odot D \triangleq D^t = \exp(t \log(D)), \quad t \in \mathbb{R}. \quad (16)$$

If we use $D_{1i}(t) = t \odot D_{12}$ in the synthesis, as t varies we obtain a continuous path that interpolates between the two real views for $t < 1$, and extrapolates the seed displacement for $t > 1$. In this way we are able to move the uncalibrated virtual

camera continuously on a curve that passes through both camera centres. The parameter t is the ‘3D-ness’ knob that we mentioned in the Introduction.

At a calibrated level, this is equivalent to moving the camera along the trajectory $t \odot G$. Indeed,

$$\begin{aligned} D^t &= (\tilde{K}G\tilde{K}^{-1})^t = e^{t \log(\tilde{K}G\tilde{K}^{-1})} \\ &= e^{\tilde{K}(t \log G)\tilde{K}^{-1}} = \tilde{K}e^{(t \log G)}\tilde{K}^{-1} \\ &= \tilde{K}G^t\tilde{K}^{-1}. \end{aligned} \quad (17)$$

A very special case is when the reference views are rectified. Given that no rotation between the two cameras is present, the virtual camera can only be translated along the line containing the centres of the cameras (baseline).

Finally, in order for our method to make sense, we must make sure that the real logarithm of D exists. A sufficient condition for a real invertible matrix K to have a real logarithm is that K has no eigenvalues on the closed negative real axis of the complex plane [5]. G satisfy the condition because its eigenvalues are $\{1, 1, e^{\pm i\theta}\}$ and so does D because it is conjugate to G .

3.3 Is H_∞ necessary?

Can we work out a solution that does not require H_∞ but only a generic collineation induced by a plane Π ? The answer is no. The replication trick cannot be applied to a generic homography induced by a plane Π , essentially because the equation of the plane is view-dependent. More specifically, if view pair 1-2 and view pair 2-3 are related by the same rigid displacement, if $H_{\Pi12}$ transfer points of Π from view 1 to view 2, the same homography will *not* transfer correctly points from view 2 to view 3. In other words, $G_{12} = G_{23}$ does not imply that $H_{\Pi23} = H_{\Pi12}$ because the equation of plane Π in the reference frame of view 1 is different – in general – from the equation of the same plane in the reference frame of view 2. It is easy to construct a counterexample confirming this remark.

In a previous paper [4] we missed this point and replicated a general plane homography instead of H_∞ . Experiments validated the approach possibly because the background plane was sufficiently far away.

4 Results

Tests with both synthetic and real images were performed. The synthetic experiment was used to compare the extrapolated view produced by the algorithm against a ground-truth image. The real experiments illustrate what is to be expected from our technique in a real, general situation.

Assuming that the background area in the images is larger than the foreground area, the homography of the background plane is the one that explains the *dominant motion*. We are here implicitly assuming that the background is approximately planar, or that its depth variation is much smaller than its average distance from the camera. We also assume that the background is sufficiently far away so that its homography approximates well the homography of the plane at infinity [20].

After aligning input images with respect to the background plane, the residual parallax allows to segment off-plane points (foreground). From this segmentation we are able to compute the epipoles and to recover the relative affine structure for a sparse set of foreground points, which is then interpolated on the pixel grid. All these steps are better explained in [4].

Then the foreground is warped using the transfer equation, i.e. Eq. (3), and pixel “splatting” [17]. Pixels are transferred in order of increasing parallax, so that points closer to the camera overwrite farther points. The planar background is warped using the background homography with destination scan and bilinear interpolation. By warping the background of the second view onto the first one, a mosaic representing all the available information about the background plane is built. Since the foreground could occlude a background area in *both* the input images, holes could remain in the mosaic. These holes are filled by interpolating from the pixel values on the boundary².

Figure 3 shows results with images generated using OpenGL. The first two are used as reference images, and the third as ground-truth. Looking at the difference image, we can see that the error is limited to few pixels, imputable to approximations introduced in the computation of the relative affine structures.

In Figure 4 the middle view is obtained by interpolation from the other two reference images taken in “Piazza dei Signori”, Verona. As the reader can notice, the location of the statue changes with respect to the window behind it due to the parallax effect.

In Figure 5 two novel snapshots synthesized from a stereo pair of images taken in “Piazza delle Erbe,” Verona, are shown. This is an example of extrapolated views obtained by replicating the epipolar geometry in two opposite directions.

Our technique makes possible to create an entire sequence as taken by a smoothly moving virtual camera, by continuously changing parameter t in Eq. (16), as illustrated in Figure 6.

In the top row, the starting stereo pair and the corresponding relative affine structure are shown. Below, sixteen synthesised images with increasing parameter t are shown. As the reader can notice, in the beginning the virtual camera is placed in front of the statue whereas at the end the camera location is such that only the profile of the statue is visible. Albeit parallax adjustment required by 3D-TV is not so considerable, this example is shown to illustrate the geometrical behaviour of the method.

More examples and movies can be found on the WWW at <http://www.sci.univr.it/~fusielo/demo/synth>.

5 Conclusion

We presented an uncalibrated view-synthesis technique that can be used to continuously adjust the parallax in 3D-TV visualization. It is based on relative affine structure for describing the scene’s geometry and on extrapolation and interpolation of the epipolar geometry linking the reference views.

²We use the `roifill` MATLAB function, which smoothly interpolates inward from the pixel values on the boundary of the polygon by solving Laplace’s equation, but any inpainting technique could be used.

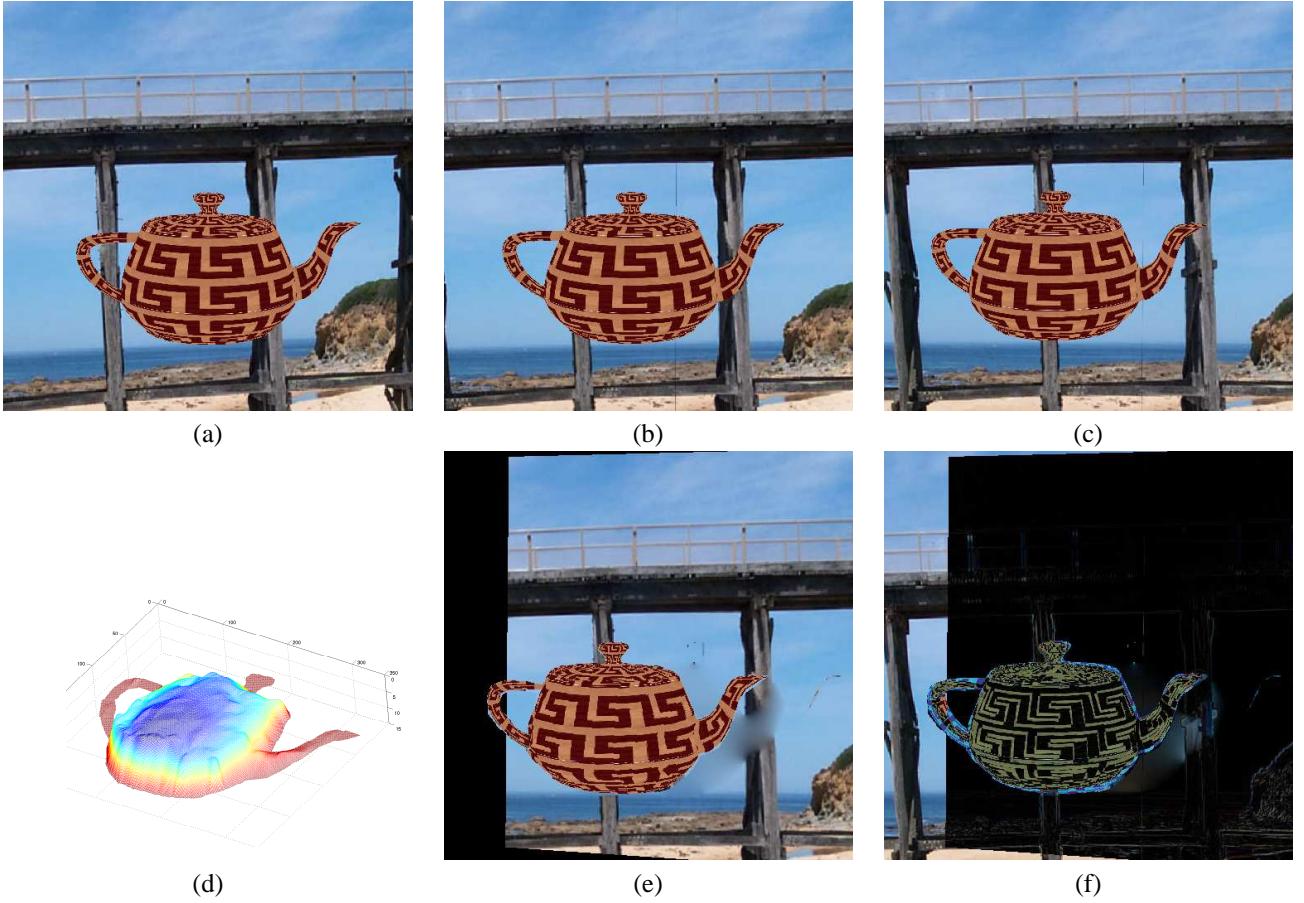


Figure 3: The three images in the first row have been generated using OpenGL: (a) and (b) are used as reference images and (c) is the ground-truth. In (d) the recovered relative affine structure is shown. The image (e) has been extrapolated from (a) and (b) with our algorithm and the image (f) is the difference between the extrapolated image and the ground-truth.



Figure 4: An example of interpolation. The first and the last images are the reference ones and the central view is interpolated ($t=0.5$).



Figure 5: An example of extrapolation. The second and the third images are the reference ones; the first and the last are extrapolated views ($t=\pm 2$).



Figure 6: In the first row the two reference images and the corresponding relative affine structures are shown. The other sixteen images are obtained with t varying from -3 to 3 with step 0.4.

After aligning input images using dominant motion estimation, a segmentation based on residual parallax is performed. From this we recover the relative affine structure and, finally, we synthesize novel views along a 1-D path, thereby changing the amount of parallax with respect to the first reference view. In another paper in preparation, we extend this method in order to be able to move the virtual camera onto a 2-manifold, starting from three reference views.

Acknowledgments

This work has been supported by the LIMA3D project (Progetto di Ricerca di Interesse Nazionale, 2003). Giandomenico Orlandi contributed with inspiring discussions. The use of the implementation of the KLT tracker by S. Birchfield is here acknowledged.

References

- [1] M. Alexa. Linear combination of transformations. In *Proceedings of the 29th annual conference on Computer graphics and interactive techniques*, pages 380–387. ACM Press, 2002.
- [2] S. Avidan and A. Shashua. Novel view synthesis in tensor space. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1034–1040, 1997.
- [3] C. Fehn. 3D-TV Using Depth-Image-Based Rendering (DIBR). In *Proceedings of Picture Coding Symposium*, San Francisco, CA, USA, December 2004.
- [4] A. Fusiello, S. Calderer, S. Ceglie, N. Mattern, and V. Murino. View synthesis from uncalibrated images using parallax. In *12th International Conference on Image Analysis and Processing*, pages 146–151, Mantova, Italy, September 2003. IAPR, IEEE Computer Society.
- [5] R.A. Horn and C.R. Johnson. *Topics in Matrix Analysis*. Cambridge University Press, London, 1991.
- [6] M. Irani and P. Anandan. Parallax geometry of pairs of points for 3D scene analysis. In *Proceedings of the European Conference on Computer Vision*, pages 17–30, 1996.
- [7] M. Irani, T. Hassner, , and P. Anandan. What does the scene look like from a scene point? In *Proceedings of the European Conference on Computer Vision*, pages 883–897, Copenhagen (Denmark), 2002.
- [8] S. Iivekovic, A. Fusiello, and E. Trucco. Fundamentals of multiple view geometry. In O. Schreer and T. Sikora P. Kauff, editors, *3D Videocommunication. Algorithms, concepts and real-time systems in human centered communication*, chapter 6. John Wiley & Sons, 2005. ISBN: 0-470-02271-X.
- [9] J. Konrad. Enhancement of viewer comfort in stereoscopic viewing: parallax adjustment. In *SPIE Symposium on Electronic Imaging Stereoscopic Displays and Virtual Reality Systems*, pages 179–190, San Jose, CA, January 1999.
- [10] S. Laveau and O. Faugeras. 3-D scene representation as a collection of images and fundamental matrices. Technical Report 2205, INRIA, Institut National de Recherche en Informatique et en Automatique, February 1994.
- [11] Q.-T. Luong and T. Viéville. Canonical representations for the geometries of multiple projective views. *Computer Vision and Image Understanding*, 64(2):193–229, 1996.
- [12] M. Magnor. 3D-TV - The Future of Visual Entertainment. In *Proceedings of Multimedia Databases and Image Communications (MDIC'04)*, pages 105–112, Salerno, Italy, June 2004. invited paper.
- [13] L. McMillan and G. Bishop. Head-tracked stereo display using image warping. In *Stereoscopic Displays and Virtual Reality Systems II*, number 2409 in SPIE Proceedings, pages 21–30, San Jose, CA, 1995.
- [14] L. McMillan and G. Bishop. Plenoptic modeling: An image-based rendering system. In *SIGGRAPH 95 Conference Proceedings*, pages 39–46, August 1995.
- [15] A. Ruf and R. Horaud. Projective rotations applied to a pan-tilt stereo head. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 144–150, Fort Collins, Colorado, June 1999. IEEE Computer Society Press.
- [16] S. M. Seitz and C. R. Dyer. View morphing: Synthesizing 3D metamorphoses using image transforms. In *SIGGRAPH 96 Conference Proceedings*, pages 21–30, August 1996.
- [17] J. Shade, S. Gortler, L. He, and R. Szeliski. Layered depth images. In *SIGGRAPH 98 Conference Proceedings*, 1998.
- [18] A. Shashua and N. Navab. Relative affine structure: Canonical model for 3D from 2D geometry and applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(9):873–883, September 1996.
- [19] SIGGRAPH 2004, Panel discussion on the 3D-TV emerging technology.
URL <http://www.siggraph.org/s2004/conference/etech>.
- [20] T. Viéville, O. Faugeras, and Q.-T. Luong. Motion of Points and Lines in the Uncalibrated Case. *International Journal of Computer Vision*, 17(1):7–42, January 1996.