An Uncalibrated View-Synthesis Pipeline

A. Fusiello and L. Irsara,
Dipartimento di Informatica, Università di Verona,
Strada Le Grazie, 15 - 37134 Verona, Italy

Abstract

This paper deals with the process of view synthesis based on the relative affine structure. It describes a complete pipeline that, starting with uncalibrated images, produces a virtual sequence with viewpoint control. Experiments illustrate the approach.

1 Introduction

Given some reference images, view synthesis consists in rendering new images of a scene as if they were taken from a virtual viewpoint which is different from all the viewpoints of the real images. This approach is called *Image Based Rendering* (IBR), and its techniques can be classified in three categories: IBR with explicit geometry, IBR without geometry, and IBR with implicit geometry.

In the first category the whole 3D structure of the scene is reconstructed. Since the complexity of a novel-view synthesis is dependent on the complexity of the 3D structure of the scene and also from the requested level of detail, these techniques may not be always the best choice, especially when the structure of the scene is very complicated. In this class we can find techniques based on view-dependent texture maps [8], 3D warping [18], layered depth images [21].

Approaches that belong to the IBR without geometry, on the other hand, do not require any information about the 3D structure of the scene: The *plenoptic function* [19] is sampled by analyzing a certain number of views of a scene. Techniques that belong to this category are, for example, concentric mosaics [24] and light field [16, 11]. These approaches yield very photorealistic results but typically require a very large number of reference images.

IBR with implicit geometry lies at the borderline between the first two: even if an explicit reconstruction of the 3D structure of the scene is not pursued, some information about the scene geometry (e.g. the disparity) it is required anyway. These approaches assume the advantages of the first two categories: photorealistic results, low space requirements and time complexity independent from the scene complexity. In this category fall view interpolation [6], view morphing [20], point transfer based on the fundamental matrix [15] and the trifocal tensor [5].

The technique we are dealing with in this paper belongs to this class and it is based on the relative affine structure [22], [23] as an implicit geometry descriptor. Some aspects of this approach have already been presented in [2] and [1]. The aim of this paper is to give a complete overview of the whole view synthesis pipeline, with references to the relevant articles that deal with topics in more detail.

2 System overview

The aim of view synthesis is to render new images as if they were taken from different viewpoints, basing all the computation on the reference views only. The process we are describing is well represented by the pipeline sketched in Fig. 1. In this section we shall give an overview of all the tasks involved. The more relevant ones will be detailed later in the rest of the paper.

For the sake of simplicity, we will consider synthesis based on two reference images I_1, I_2. The first image, I_1, will be referred to as the *source image* because the synthetic views are built by transferring (warping) the pixels of I_1. The output is a sequence of synthetic images.

According to [1], the synthesis of new images requires the following information:

- the *uncalibrated rigid transformation* D_{12} between the two reference views, which has the form

$$D_{12} \triangleq \begin{bmatrix} H_{\infty 12} & \mathbf{e}_{21} \\ \mathbf{0} & 1 \end{bmatrix} \qquad (1)$$

  where $\mathbf{e}_{21}$ is the epipole in the second image with respect to the first one and $H_{\infty 12}$ is the infinite plane homography that maps from the first image to the second one;

- the *relative affine structure* $\gamma_1^k, \quad k = 1, ..., m$ of all points in the source image.

- the *uncalibrated rigid transformation* D_{1v} between the source image and the virtual one;

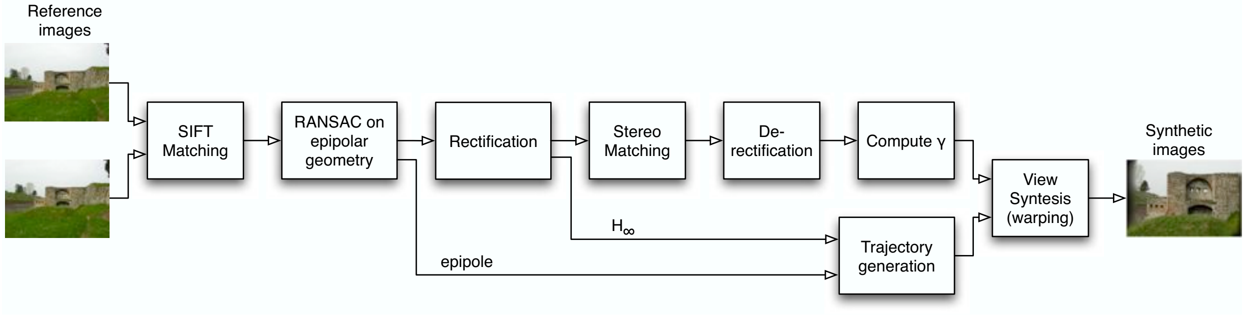

Figure 1. The view synthesis pipeline

The *relative affine structure* is a close relative of the disparity: its computation (Sec. 4) requires to establish dense correspondences between I_1 and I_2. To this end we need to rectify the input images, run a stereo matching algorithm on the rectified pair and de-rectify the results to obtain dense correspondences referred to the original images.

Since intrinsic parameters are unknown, an uncalibrated rectification procedure is employed (Sec. 3) which relies on sparse correspondences. To this end we extract SIFT features in both images and match the descriptors, as in [17]. Then we run a RANSAC estimation of the fundamental matrix in order to discard bad matches (outliers). The surviving matches (inliers) are used as input to the rectification.

As a by-product of the rectification we also obtain the infinite plane homography $H_{\infty 12}$. The epipole $\mathbf{e}_{21}$ is extracted from the fundamental matrix.

Once the images are rectified, dense matches can be obtained using any stereo matching algorithm, for example [9] or [4]. The correspondences are transferred back to the original reference images by applying the inverse of the rectifying transformation (de-rectification).

In order to be able to render a sequence of synthetic images I_v, we need to specify the trajectory of the virtual camera at the uncalibrated level, i.e. to specify a family of uncalibrated rigid transformation D_{1v}. This will be achieved thanks to the scalar multiple, commutative composition and linear combination of uncalibrated rigid transformations, as described in Sec. 5.

Finally, all the points of the source image can be transferred in the synthetic view using Eq. (9). Care must be taken to deal with occlusions and holes, as detailed in Sec. 6.

3 Rectification

Epipolar rectification is an important stage in dense stereo matching, as almost any stereo algorithm requires rectified images, i.e., images where epipolar lines are parallel and horizontal and corresponding points have the same vertical coordinates. If the camera parameters are known, rectification can easily be accomplished with [10]. In this case, the rectifying homographies are conjugated to a rotation, i.e., they are induced by the plane at infinity [13]. Otherwise, when internal parameters are unknown, as in this paper, we assume that a number of corresponding points $\mathbf{m}_1^j \leftrightarrow \mathbf{m}_2^j$ are available and – on the same line as in [14] – we seeks the rectifying homographies that make the original points satisfy the epipolar geometry of a rectified image pair.

The fundamental matrix of a rectified pair has a very specific form, namely it is the skew-symmetric matrix associated with the cross-product by the vector $\mathbf{u}_1 = (1, 0, 0)$:

$$[\mathbf{u}_1]_\times = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix} \quad (2)$$

Let H_2 and H_1 be the unknown rectifying homographies. The transformed corresponding points must satisfy the epipolar geometry of a rectified pair, hence[1]

$$(H_2\mathbf{m}_2^j)^\top [\mathbf{u}_1]_\times (H_1\mathbf{m}_1^j) = 0, \quad (3)$$

As this equation must hold for any correspondence, one obtains a system of non-linear equations in the unknown H_2 and H_1.

The way in which H_2 and H_1 are parametrized is crucial: We force the rectifying homographies to have the same structure as in the calibrated case, i.e., to be homographies induced by the plane at infinity, namely

$$H_2 = K_{n2} R_2 K_{o2}^{-1} \quad H_1 = K_{n1} R_1 K_{o1}^{-1}. \quad (4)$$

The old intrinsic parameters (K_{o1}, K_{o2}) and the rotation matrices (R_1, R_2) are unknown, whereas the new intrinsic parameters (K_{n1}, K_{n2}) can be set arbitrarily, provided that vertical focal length and vertical coordinate of the principal point are the same.

[1] Points are expressed in homogeneous coordinates.

Each homography depends in principle on five (intrinsic) plus three (rotation) unknown parameters. The rotation of one camera along its X-axis, however, can be eliminated, as this is tantamount to rotating a rectified pair around the baseline. The number of parameters is further reduce by making an educated guess on the old intrinsic parameters: no skew, principal point in the centre of the image, aspect ratio equal to one. The only remaining unknowns are the focal lengths of both cameras. Assuming that they are identical and equal to α, we get:

$$K_{o2} = K_{o1} = \begin{bmatrix} \alpha & 0 & w/2 \\ 0 & \alpha & h/2 \\ 0 & 0 & 1 \end{bmatrix} \tag{5}$$

where w and h are width and height (in pixel) of the image.

The minimization is carried out using Levenberg-Marquardt, starting with all the unknown variables set to zero. Finally, the the rectifying homographies are computed with Eq. (4).

As a by-product of this rectification, we obtain an approximation of the homography induced by the plane at infinity between the two original cameras, which is given by

$$H_{\infty 12} = H_2^{-1} K_{2n} K_{1n}^{-1} H_1. \tag{6}$$

More details on this method can be read in [13].

4 Relative affine structure

The relative affine structure [23] or plane+parallax [12] are equivalent formulations of the two-views geometry that substitute one of the two image planes with an arbitrary reference plane. Since our view synthesis algorithm is based on the relative affine structure theory, we shall briefly outline it here.

Given two images of a scene, two conjugate points $(\mathbf{m}_1; \mathbf{m}_2)$ are related by the following equation[2]

$$\mathbf{m}_2 \simeq= H_{\infty 12}\mathbf{m}_1 + \mathbf{e}_{21}\gamma_1 = [I|\mathbf{0}]D_{12} \begin{bmatrix} \mathbf{m}_1 \\ \gamma_1 \end{bmatrix} \tag{7}$$

where $\mathbf{m}_1$ is a point in the first reference image, $\mathbf{m}_2$ is its conjugate in the second one, D_{12}, defined in Eq. (1), represents the rigid transformation at the uncalibrated level that links the images, and γ_1 is the relative affine structure of $\mathbf{m}_1$.

Let $\mathbf{m}_1^k; \mathbf{m}_2^k$ with $k = 1, ..., m$ be the (dense) set of corresponding points in the reference images. The relative affine structure for each point k in the source image is obtained by solving for γ in Eq. (7), given D_{12}:

$$\gamma_1^k = \frac{(\mathbf{m}_2^k \times \mathbf{e}_{21})^\mathsf{T}(H_{\infty 12}\mathbf{m}_1^k \times \mathbf{m}_2^k)}{||\mathbf{m}_2^k \times \mathbf{e}_{21}||^2}. \tag{8}$$

It turns out [23] that the relative affine structure *depends only on the source image*, i.e., it does not depend on the second reference view. Thanks to this property, arbitrary new views I_v can be synthesized by substituting D_{12} with the matrix D_{1v} that represents the uncalibrated rigid transformation between the source image and the virtual one:

$$\mathbf{m}_v \simeq [I|\mathbf{0}]D_{1v} \begin{bmatrix} \mathbf{m}_1 \\ \gamma_1 \end{bmatrix}. \tag{9}$$

This equation allows to transfer points from the source image I_1 to the synthetic image I_v.

Please note that if the view synthesis is based on more than two images, we can merge the relative affine structure map obtained with images 1-2 with the map obtained with images 1-3 by suitably scaling one of the two maps.

5 Trajectory generation

The definition of D_{1v} (i.e. $H_{\infty 1v}$ and $\mathbf{e}_{v1}$) is one of the most awkward issues in the view synthesis algorithm. It is exhaustively dealt with in [1], so we will summarize here only the main results.

Rigid transformations at the uncalibrated level D are closely related to the rigid transformations; in fact it is proved in [1] that the group of uncalibrated rigid transformations is isomorphic to the group of Euclidean rigid transformations $\mathrm{SE}(3, \mathbb{R})$ via the conjugacy map:

$$D = \begin{bmatrix} KRK^{-1} & K\mathbf{t} \\ \mathbf{0} & 1 \end{bmatrix} = \tilde{K} \begin{bmatrix} R & \mathbf{t} \\ \mathbf{0} & 1 \end{bmatrix} \tilde{K}^{-1} \tag{10}$$

with

$$\tilde{K} = \begin{bmatrix} K & \mathbf{0} \\ \mathbf{0} & 1 \end{bmatrix}. \tag{11}$$

Alexa in [3] defined some operators that allow to interpolate, extrapolate and combine rigid transformations in $\mathrm{SE}(3, \mathbb{R})$. Thanks to the isomorphism these operators can be mapped onto the uncalibrated displacement group. In fact, we use them in an uncalibrated setting to compute the uncalibrated rigid transformation D_{1v} that specifies position and orientation of a virtual camera.

Given an uncalibrated rigid displacement D_{12} between two cameras, its *scalar multiple* is defined as:

$$t \odot D_{12} \triangleq e^{t \log(D_{12})}, \quad t \in \mathbb{R}. \tag{12}$$

Varying the value of t we are moving the virtual camera along a path in $\mathrm{SE}(3, \mathbb{R})$ passing through the position and orientation of two reference cameras. The segment that interpolates between the two cameras is the geodesic.

Given two uncalibrated rigid displacements D_{12} and D_{13}, it is possible to compute the *commutative composition* of D_{12} and D_{13}:

$$D_{12} \oplus D_{13} \triangleq e^{\log D_{12} + \log D_{13}}. \tag{13}$$

[2] $\simeq$ is the equality sign up to a scale factor.

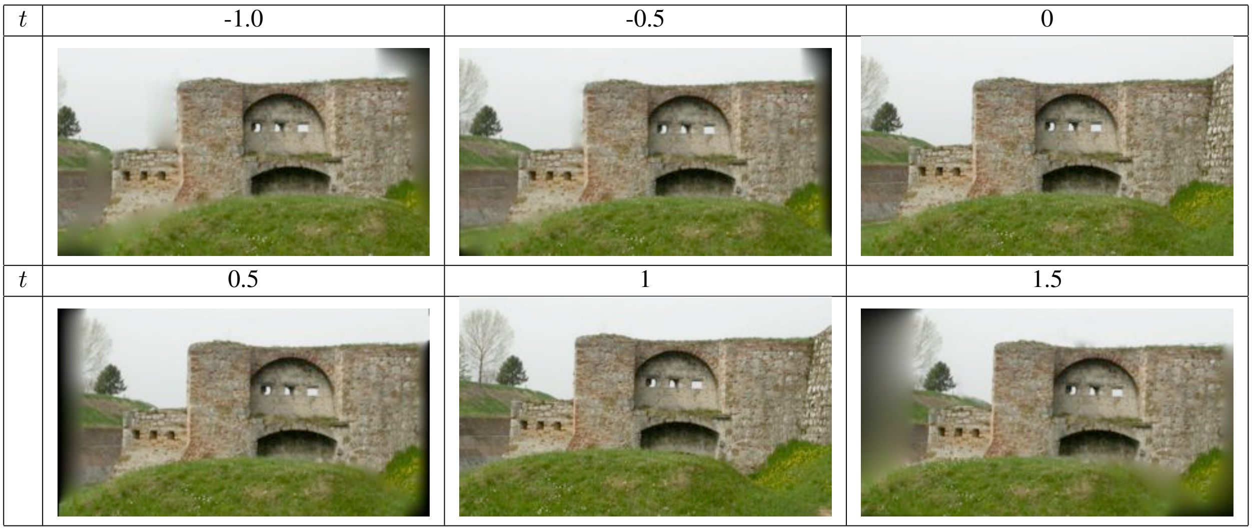

Figure 2. Some frames from the “Porta” sequence. The values $t = 0$ and $t = 1$ correspond to the reference images.

In a sense, applying $D_{12} \oplus D_{13}$ is like applying D_{12} and D_{13} simultaneously.

If three reference images are available, we can move the virtual camera on a surface of $\mathrm{SE}(3, \mathbb{R})$ by doing a weighted linear combination of D_{12} and D_{13}:

$$(u \odot D_{12}) \oplus (v \odot D_{13}) \triangleq e^{u \log D_{12} + v \log D_{13}}, \quad u, v \in \mathbb{R}. \tag{14}$$

The parameters (u, v) describe the surface. A trajectory of the virtual camera is specified by a curve in the parameters space.

6 Warping

This section deals with the last part of our view synthesis algorithm, the warping phase. Here we will explain how the pixels of the source image are transferred to build a new synthetic image.

As the point transfer mapping is not invertible (it is neither surjective nor injective), only forward mapping can be used to warp the source image. Care must be taken to preserve the coherence of surfaces and their visibility. To this end we adopted pixel splatting with back-to-front rendering. This means that points that are farther from the camera, i.e., points with a smaller relative affine structure (absolute value) are transferred first, so that points closer to the camera can overwrite them.

Please note that, in general, the relative affine structure does not depend only on the distance from the camera; however, when the reference plane is at infinity, it reduces to $\gamma = 1/\zeta$ where ζ is the depth of the point.

Pixel splatting copes only with the small holes due mainly to magnification effects. Larger holes owing to occlusions are filled by interpolation from pixel values on the boundary. Inpainting [7] could have been used for better visual quality. A more principled strategy would exploit the information coming from all the reference images, instead that from the source image only: We leave this issue for future investigation.

7 Results

Our technique allows to create an entire image sequence by continuously changing the parameters t in (12) or u, v in (14). As a result, the video seems captured by a smoothly moving virtual camera. Some examples are available on the web[3]. In this section we are only reporting sample frames from those videos.

The first example (Fig. 2) is a synthetic sequence based on two reference images. By varying the parameter t the virtual camera moves along a trajectory in $\mathrm{SE}(3, \mathbb{R})$ containing the reference cameras.

The second example (Fig. 4) shows the image sequence “Pista”, based on three reference images (Fig. 3). In this case, by varying the parameters u, v the virtual camera moves on a trajectory that belongs to a surface in $\mathrm{SE}(3, \mathbb{R})$ containing the three reference cameras.

[3] http://profs.sci.univr.it/ fusiello/demo/synth/

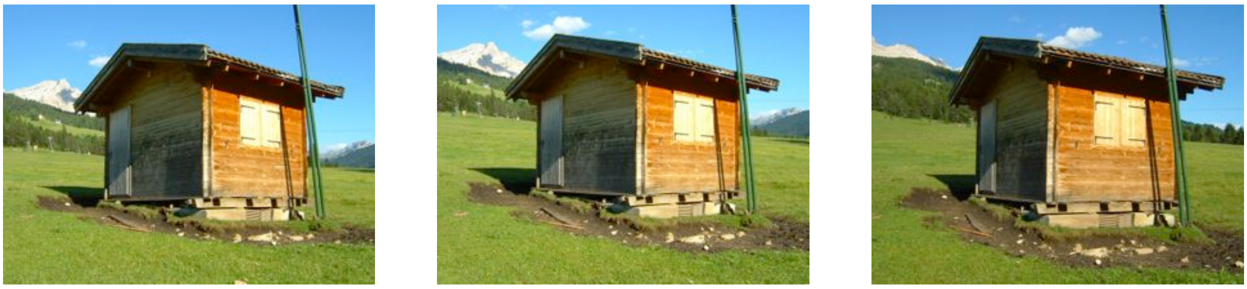

Figure 3. The three “Pista” references images.

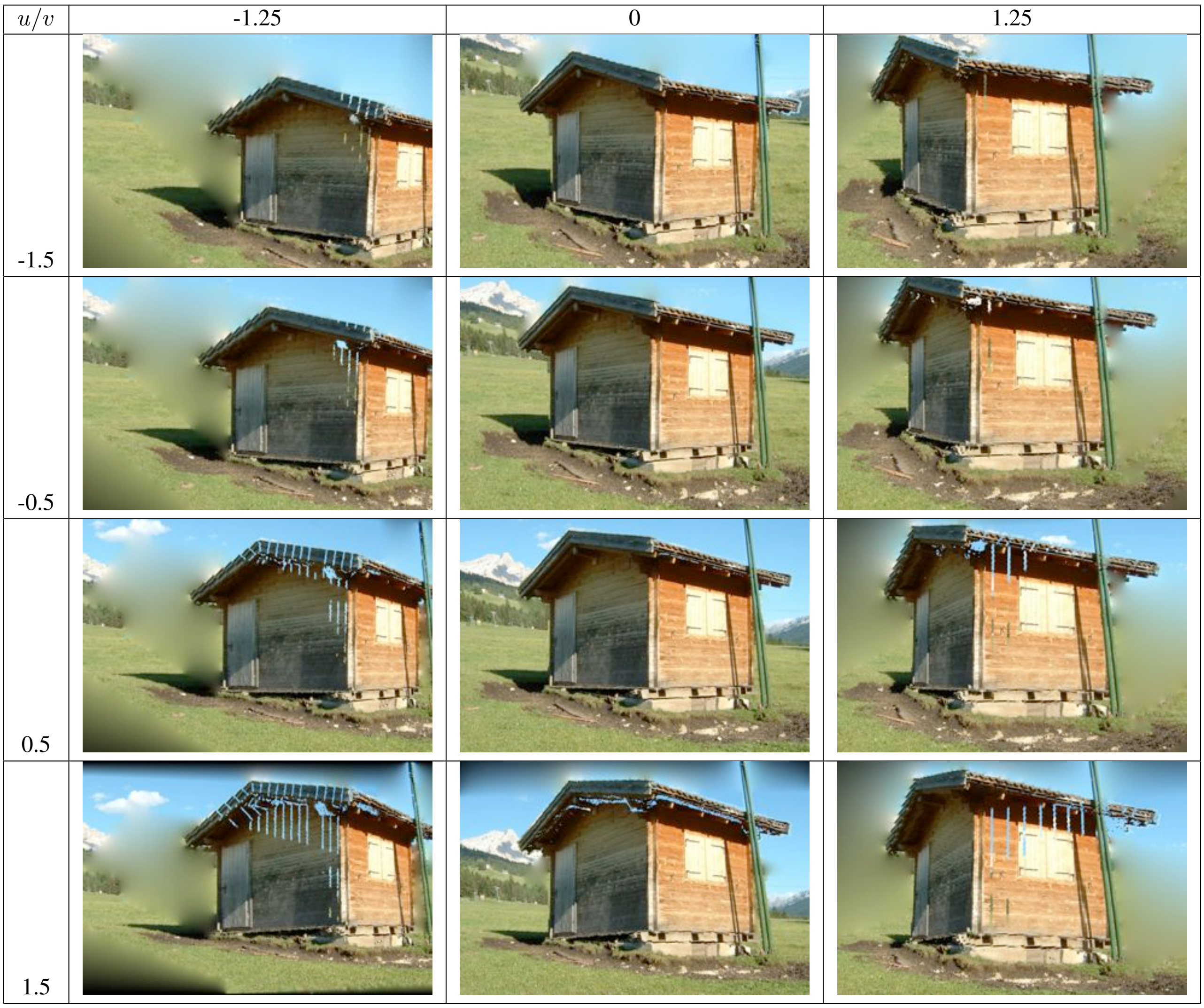

Figure 4. Some synthetic frames obtained from the “Pista” images for different values of (u, v).

8 Conclusions

We described a view synthesis pipeline that renders a synthetic sequence starting from uncalibrated images. The most salient feature of this system is the way in which virtual trajectories are specified, based on the interpolation and extrapolation of the motion among the reference views. The description of the motion at the uncalibrated level requires the homography of the infinity plane, which we estimate in conjunction with the epipolar rectification. The visual quality of the output depends critically on the disparity estimate and on the warping strategy. Future work will aim at improving these stages.

Acknowledgments

We are grateful to O. Schreer who ran his stereo matching code [4] on our images and provided us with the disparity. The use of the Matlab SIFT code by A. Vedaldi (www.cs.ucla.edu/vedaldi/sift/sift.htm) and the Matlab RANSAC code by P. Kovesi (www.csse.uwa.edu.au/~pk/research/matlabfns/) is here acknowledged.

References

[1] A. Fusiello. Specifying virtual cameras in uncalibrated view synthesis. *IEEE Transactions on Circuits and Systems for Video Technology*, 2007. In press.

[2] A. Colombari and A. Fusiello and V. Murino. Uncalibrated interpolation of rigid displacements for view synthesis. In *Proceedings of the IEEE International Conference on Image processing*, 2005.

[3] 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.

[4] N. Atzpadin, P. Kauff, and O. Schreer. Stereo analysis by hybrid recursive matching for real-time immersive video conferencing. *Circuits and Systems for Video Technology, IEEE Transactions on*, 14(3):321–334, 2004.

[5] 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.

[6] S. E. Chen and L. Williams. View interpolation for image synthesis. In *SIGGRAPH '93 Conference Proceedings*, volume 27, pages 279–288, 1993.

[7] A. Criminisi, P. Perez, and K. Toyama. Region filling and object removal by exemplar-based image inpainting. *IEEE Transactions on Image Processing*, 13(9):1200–1212, 2004.

[8] P. E. Debevec, C. J. Taylor, and J. Malik. Modeling and rendering architecture from photographs: A hybrid geometry- and image-based approach. In *SIGGRAPH 96 Conference Proceedings*, pages 11–20, 1996.

[9] A. Fusiello, U. Castellani, and V. Murino. Relaxing symmetric multiple windows stereo using markov random fields. In *Energy Minimization Methods in Computer Vision and Pattern Recognition*, Lecture Notes in Computer Science, pages 91–104, 2001.

[10] A. Fusiello, E. Trucco, and A. Verri. A compact algorithm for rectification of stereo pairs. *Machine Vision and Applications*, 12(1):16–22, 2000.

[11] S. J. Gortler, R. Grzeszczuk, R. Szeliski, and M. F. Cohen. The lumigraph. In *SIGGRAPH 96 Conference Proceedings*, pages 43–54, 1996.

[12] 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.

[13] L. Irsara and A. Fusiello. Quasi-euclidean uncalibrated epipolar rectification. Research Report RR 43/2006, Dipartimento di Informatica - Università di Verona, 2006.

[14] F. Isgrò and E. Trucco. Projective rectification without epipolar geometry. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages I:94–99, 1999.

[15] S. Laveau and O. Faugeras. 3-D scene representation as a collection of images. In *Proceedings of the International Conference on Pattern Recognition*, volume 1, pages 689–691, 1994.

[16] M. Levoy and P. Hanrahan. Light field rendering. In *SIGGRAPH 96 Conference Proceedings*, pages 31–42, 1996.

[17] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[18] 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, 1995.

[19] L. McMillan and G. Bishop. Plenoptic modeling: An image-based rendering system. In *SIGGRAPH 95 Conference Proceedings*, pages 39–46, 1995.

[20] S. M. Seitz and C. R. Dyer. View morphing: Synthesizing 3D metamorphoses using image transforms. In *SIGGRAPH 96 Conference Proceedings*, pages 21–30, 1996.

[21] J. Shade, S. Gortler, L. He, and R. Szeliski. Layered depth images. In *SIGGRAPH 98 Conference Proceedings*, pages 231–242, 1998.

[22] A. Shashua and N. Navab. Relative affine structure: Theory and application to 3-D reconstruction from perspective views. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 483–489, 1994.

[23] 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, 1996.

[24] H.-Y. Shum and L.-W. He. Rendering with concentric mosaics. In *SIGGRAPH 99 Conference Proceedings*, volume 33, pages 299–306, 1999.