Novel View Synthesis: a comparative analysis study

A. Habed and B. Boufama

SCHOOL OF COMPUTER SCIENCE
University of Windsor
Windsor, Ontario
Canada N9B 3P4
E-mail : habed@hoggar.lams.uwindsor.ca, boufama@cs.uwindsor.ca

Abstract

The problem of synthesizing novel views consists of generating new views of a scene using at least two reference views. The new generated views, called also novel views, correspond to virtual viewpoints where no real camera has taken snapshots. In the last few year, view synthesis has been a thriving field of research in both computer vision and computer graphics. Applications of novel view synthesis range from generating photorealistic images to the modeling of real scenes. Several approaches have been proposed in the literature for novel view synthesis. However, no comparison has been done on these approaches to evaluate their performances and qualities. This paper, aims at providing a quick overview and a comparative study of all known methods for view synthesis. The results of this study are of great interest to any work involving view synthesis. Experiments with both simulated and real images were carried out to support our conclusions.

1 Introduction

The problem of image synthesis consists of generating new views of a scene using at least two reference views. The new generated views, called also novel views, correspond to virtual viewpoints where no real camera has taken snapshots. Although novel view synthesis requires dense matching, this paper focuses on the geometric aspect of generating a third image, assuming matched point in two images.

In the last few years, view synthesis has been a thriving field of research in both computer vision and computer graphics [9][1]. Applications of view synthesis range from generating photorealistic images to real scene modeling.

Several approaches for novel view synthesis have been proposed in the literature. These approaches fall into 2 categories:

- 3D-structure based novel view synthesis: using at least two reference images, the scene is explicitly reconstructed in the three-dimensional space then novel views are rendered. The reconstruction problem, has been investigated intensively in the literature [10][4]. For instance, in [8] 10 calibrated cameras were used and in [13] 5 calibrated cameras were used, to reconstruct then synthesize novel views. Note that the reconstruction does not have to be Euclidean, a projective reconstruction can be used as well.
- image based view synthesis: novel views are generated using reference images without explicit reconstruction. This category of approaches seems more attractive since the reconstruction process is avoided. Several methods, falling in this category, have been published. For instance, in [5] the synthesis of points in the third image was done using the epipolar geometry, whereas in [16], it was done using trilinear tensors.

Although it is widely accepted that synthesizing novel views directly from reference images is more efficient, it has never been demonstrated. This is one of the questions this paper tries to answer.

Section 2 describes the geometrical background used in the remaining of the paper; and gives a quick review on the epipolar geometry and the 3D reconstruction. Section 3 describes the different approaches for view synthesis and provides a comparative analysis on these approaches using both simulated and real data.

2 Background

Novel view synthesis is based on geometry and uses several computer vision algorithms. This section provides a review of the geometry and algorithms needed for view synthesis.

2.1 Camera model

This paper assumes a pinhole model for a real camera. This is a common assumption in the computer vision community.

We provide in this subsection a quick review of this model.

The pinhole model is a good approximation of a real camera and is by far the most used for modeling CCD cameras. In this model, a pixel p on the image plane is the pure perspective projection of the point P of the scene. Using homogeneous coordinates, the pure perspective projection of the scene point $P = (X, Y, Z, T)$ on the image point $p = (x, y, t)$ can be represented by the relation

$$\begin{pmatrix} x \\ y \\ t \end{pmatrix} = \lambda \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \\ T \end{pmatrix} \quad (1)$$

where λ is a scale factor.

Because the scene coordinate system and the camera coordinate system are not the same, the scene points must undergo a rigid displacement (a rotation and a translation) before the perspective projection. Furthermore, the image coordinate system uses pixels as units and uses the top-left corner of the image plane as origin. Hence, after projection, the points must undergo a scaling and a 2D translation. The above relation becomes

$$p = \lambda AIDP = \lambda MP \qquad (M = AID) \quad (2)$$

where M, $3X4$ matrix, is called the projection matrix, A the intrinsic parameter matrix, I the pure perspective projection, and D the displacement matrix.

2.2 Epipolar geometry

Epipolar geometry plays a key role in many computer vision problems. In particular, it is heavily used in matching, self-calibration, 3D reconstruction, and view synthesis. A quick review of epipolar geometry is provided here.

Two images of the same rigid scene, taken from two different viewpoints, are related by the so called *epipolar geometry* [7, 10]. When images of a rigid scene are obtained with uncalibrated cameras, the epipolar geometry is the only information we can get from point matches.

Let two images be taken by two cameras by linear projection, as shown in Figure 1. Let O and O' be the two optical centers of the two images which will be called, first and second image respectively in the following. The point O projects to the point e' in the second image, and the point O' projects to the point e in the first image. The two points e and e' are the epipoles and the lines through e and e' are the epipolar lines. Let a space point P be projected on p and p' respectively in the first and the second image. The plane defined by the three points P, O and O' is the epipolar plane, it contains the two points p and p'. The projections of this plane into the first and the second image are respectively the epipolar lines (ep) and $(e'p')$ (see Figure 1).

The most common way to describe the epipolar geometry is by means of a 3×3 matrix called the *fundamental matrix*,

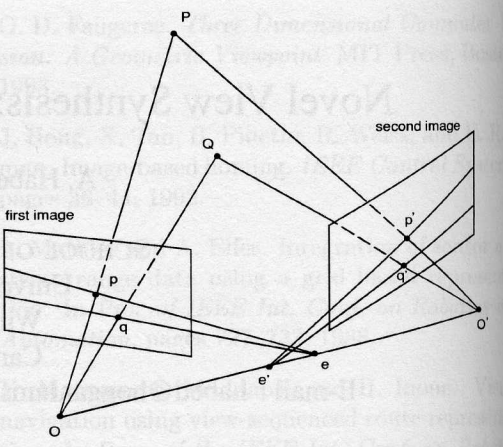

Figure 1: *A space point P defines the epipolar plane OPO' that cuts the 2 images in the 2 epipolar lines (ep) and $(e'p')$.*

usually noted F. In particular, the epipoles are implicit to F, and they can be extracted from it. For a point p given by its homogeneous coordinates in the first image, the corresponding epipolar line l_p in the second image is given by $l_p = Fp$, where, l_p is a 3-vector containing the coefficients of the line.

If in the second image p' is the corresponding point to p then p' must belong to l_p, this can be written as

$$p'^T F p = 0 \quad (3)$$

Several methods exists for computing the fundamental matrix. The most recent ones [2][6][14] provide excellent results that can be used in practice.

2.3 3D reconstruction in the projective space

It is possible to recover the projective three-dimensional structure of a scene using images taken with uncalibrated cameras and pixel correspondences between these images[3][7][11]. Although this reconstruction is defined in a projective space, it can be used for many applications. In particular, it can be used for novel view synthesis.

Several methods exist for recovering the projective 3D structure. For a comparison study, the intrested reader can consult the paper of C. Rothwell *et al*[15].

We have used in this paper a method, proposed by Rothwell[15], for projective 3D reconstruction that is based on the fundamental matrix. This method has the advantage of simplicity and is very straightforward to implement. Furthermore, it is based on the fundamental matrix which can be reliably computed using one of the most recent methods.

Given the fundamental matrix F and the the epipole e' of a pair of images, the projection matrices M_1 and M_2 of the two cameras are given by :

$$M_1 = [\ I \quad 0\ ]\, G, \quad M_2' = [\ [e']_\times F \mid e'\ ] \begin{bmatrix} I & 0 \\ \alpha^T & \alpha_4 \end{bmatrix} G$$

where $[e']_\times$ represents the asymmetric matrix derived from the vector e'. $[e']_\times F$ represents the matrix product ($[e']_\times F$). G is an arbitrary nonsingular 4x4 matrix representing a projective deformation and can be set to the 4×4 identity matrix. I is the 3×3 identity matrix. The α_i, $i \in \{1,\ldots,4\}$ can be in theory chosen with the only constraint of a nonzero α_4. However, as mentioned by Rothwell in [15], in practice these coefficients are chosen so that the projection matrices are numerically well conditioned.

Once the projection matrices have been computed, the projective 3D coordinates of the points can be calculated by triangulation.

2.4 Trilinear tensors

The trilinear tensors is an algebraic entity that links point correspondences across three images. These multilinear matching constraints were first introduced in [16]. Because they link three images through linear constraints, novel view synthesis is one of their main applications[1]. A summary of the trilinearity principles follows.

Three views of a rigid scene satisfy an algebraic relation of the form $T(\psi, \psi\prime, \psi\prime\prime)$, where ψ, $\psi\prime$ and $\psi\prime\prime$ are three arbitrary views of the object, and T has a special trilinear form. The coefficient of T can be recovered linearly using only point correspondences across the three images. Given a 3-image match $p = (x, y, 1)$, $p\prime = (x\prime, y\prime, 1)$ and $p\prime\prime = (x\prime\prime, y\prime\prime, 1)$, 9 trilinear constraints can be derived, of which 4 are linearly independent.

$$\begin{aligned} x\prime\prime x\prime p^T\alpha_1 - x\prime\prime p^T\alpha_2 - x\prime p^T\alpha_3 + p^T\alpha_4 &= 0 \\ y\prime\prime y\prime p^T\alpha_1 - y\prime\prime p^T\alpha_5 - y\prime p^T\alpha_7 + p^T\alpha_6 &= 0 \\ x\prime\prime y\prime p^T\alpha_1 - x\prime\prime p^T\alpha_5 - y\prime p^T\alpha_3 + p^T\alpha_8 &= 0 \\ y\prime\prime x\prime p^T\alpha_1 - y\prime\prime p^T\alpha_2 - x\prime p^T\alpha_7 + p^T\alpha_9 &= 0 \end{aligned} \tag{4}$$

Having a total of 27 unknown parameters, we need at least 7 matches in the three images, that is, 7 triplets of the form $(p, p\prime, p\prime\prime)$.

3 Novel view synthesis

We describe in this section all the different known approaches for view synthesis. Our goal here is to analyze and compare these different methods.

Novel view synthesis usually means synthesizing a third new view given two reference views. This third view can be defined either by its viewpoint and camera parameters or by a few points which locations are known in this view. In this paper, we use the second approach, that is, we assume that at least 8 points are matched in two views and that their locations are known in the third view. Then, any other point that appears in two views can be synthesized in the third one.

Experiments were carried out on both simulated and real images. The simulated scene we have used consists of 64 space points located on 5 parallel planes. However, only 15 points were assumed to be matched across all the three images. Uniform noise was progressively added to all pixels, and the process was repeated for each noise level. In addition, we have used 2 different geometric configurations. In the first one, the center of projection of the synthesized image was far from a close-collinear situation with the two other centers of projection. In the second configuration, the center of projection of the synthesized image was in a close-collinear situation with the two other centers of projection.

Two real scenes were used in these experiments. The images of the first real were obtained from the CMU web site (see figure 2). The images of the second real scene were made in-house (see figure 3). In both cases, we have used 3 images, the third image was used as a ground truth. The total number of points used was 60, where only 15 points were assumed to be matched across all the three images. These points are interest points, such as corners, extracted and matched using correlation. Note that in all cases, we did not assume that our cameras were calibrated. The only assumption was that 15 points are matched across 3 images.

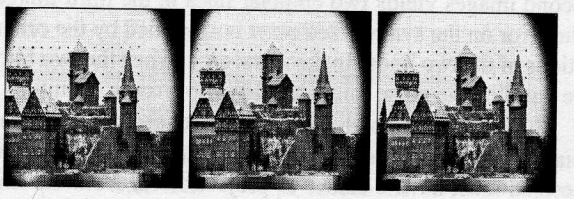

Figure 2: *Three images of the first real scene*

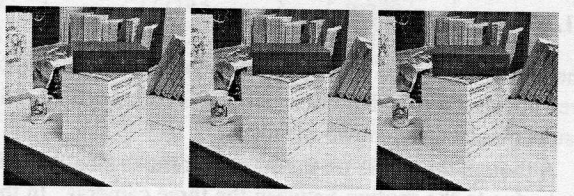

Figure 3: *Three images of the second real scene*

3.1 View synthesis using epipolar geometry

The fundamental matrices linking 3 images can be computed when at least 8 matches in the 3 images are available. Let F_{12}, F_{23}, and F_{13} be the fundamental matrices of image 1 and image 2, image 2 and image 3, and image 1 and image 3 respectively. Let p and p' be two matched points in image

1 and image 2. Their correspondence point in image 3, p'', satisfies two constraints:

- p'' belongs to the line in image 3 defined by $F_{13}p$, and
- p'' belongs to the line in image 3 defined by $F_{23}p'$.

Therefore, p'' belongs to the intersection of two lines and is given by

$$p'' = F_{13}p \times F_{23}p' \tag{5}$$

where $\times$ is the cross-product.

The above equation is the basic equation for view synthesis based on epipolar geometry.

3.1.1 Tests on simulated images

The fundamental matrices were calculated using only 15 points. The projection (synthesis) of all 64 points of the simulated scene on the third image is obtained using equation (5).

Table (1) summarizes our results: Table (1) clearly shows that errors on the synthesized pixels are higher in configuration 2. This is because of the geometry of the 3 images in the second configuration where, the 3 centers of projection were in close-collinear situation. A pixel appearing in the first and second images yields two epipolar lines in the third image. The error on the synthesized pixel is amplified by the orientations of the two lines. In the worst case, when the two lines are parallel, the noise might amplify the error to infinity.

This is a major drawback of the epipolar geometry based image synthesis method. In other words, the view to be synthesized must have a center of projection far from the line defined by the centers of projection of the two reference images.

3.1.2 Tests on real images

The scenario was the same as in the simulated case. However, no noise was added since real data is already noisy. The results obtained are shown in table (2).

Although the errors on the second scene is smaller, it is likely due to the motion between the three cameras. In the second scene case, the motion was slightly larger.

3.2 View synthesis using trilinear tensors

When at least 7 matches in 3 images are available, the trilinearity coefficients, the α_i, $i \in \{1, \ldots, 9\}$, vectors can be computed. Given two matched points, p and p', in image 1 and image 2 and the tensors, their correspondence point in image 3, p'', can be calculated using the trilinearity equations (4).

3.2.1 Tests on simulated images

The same simulated scene was used here. Similarly, the tensors were calculated using only 15 points. The projection (synthesis) of all 64 points of the simulated scene on the third image is obtained using equation (4). As in previous experiments, we have used the same 2 different geometric configurations.

The errors shown on Table (3) are similar to the ones obtained when using the epipolar geometry. In particular, this method has the same problem with the second configuration, when the 3 centers of projection are in close-collinear situation. Although this is not apparent from the trilinear tensors, the sensitivity is likely due to the geometric configuration.

3.2.2 Tests on real images

As when using epipolar geometry, the errors on the second scene are smaller (Table (4)). However, one can easily notice that the errors here are larger than in the case of epipolar geometry. One highly likely reason for such a result is that the new recent methods for calculating the epipolar geometry are numerically well conditioned yielding better results. In particular, we have used standard SVD routine for calculating the trilinear tensors.

3.3 View synthesis using projection matrices

In the projective cases, the projection matrices can be calculated using different methods. There are as many methods for calculating the projection matrices as there are methods for calculating the projective structure[15]. In this paper we have used the method described in Paragraph 2.3.

Let $M^{(1)}$, $M^{(2)}$ and $M^{(3)}$ be the camera matrices for the first, second and third image respectively. Without loss of generality, let's assume that $M^{(1)} = [I|0]$, M^2 has entries m^2_{ij} and M^3 has entries m^3_{ij}. The projections of a space point $P = (X, Y, Z, T)^T$ on the three image planes, ψ_1, ψ_2, and ψ_3, are p_1, p_2, and p_3 respectively. Using the pinhole model for our cameras, these projections yield the following equations:

$$\begin{aligned} x_k &= \frac{m^k_{11}X + m^k_{12}Y + m^k_{13}Z + m^k_{14}}{m^k_{31}X + m^k_{32}Y + m^k_{33}Z + m^k_{34}T} \\ y_k &= \frac{m^k_{21}X + m^k_{22}Y + m^k_{23}Z + m^k_{24}T}{m^k_{31}X + m^k_{32}Y + m^k_{33}Z + m^k_{34}T} \end{aligned} \tag{6}$$

where $P = (X, Y, Z, T)^T$, $p_k = (x_k, y_k, 1)^T$, and $k = \{1, 2, 3\}$.

When $k = 1$, the projection matrix $M^{(1)}$ has an identity transformation that simplifies the above equations to:

$$\begin{aligned} x_1 &= \frac{X}{Z} \\ y_1 &= \frac{Y}{Z} \end{aligned} \tag{7}$$

Noise(pixel)	Configuration 1		Configuration 2	
	Average error(pixel)	maximal error	Average error(pixel)	maximal error
0.0	0.000492	0.001122	0.000484	0.001212
0.1	0.384855	1.167554	0.525779	1.463568
0.2	0.757739	2.221504	1.089199	4.290025
0.3	1.621610	5.365299	1.703688	5.262962
0.4	1.736501	4.937816	2.273489	7.331515
0.5	2.196342	6.945092	2.476629	8.693965
0.75	3.549879	9.852100	3.487870	12.484853
1.0	3.813195	15.861182	4.038800	13.613498
2.0	7.253263	28.766391	8.431733	40.446371

Table 1: *Errors on the synthesized pixels using epipolar geometry on simulated data*

First real scene		Second real scene	
Average error(pixel)	maximal error	Average error(pixel)	maximal error
3.56	26.47	2.94	12.16

Table 2: *Errors on the synthesized pixels using epipolar geometry on real images*

Noise(pixel)	Configuration 1		Configuration 2	
	Average error(pixel)	maximal error	Average error(pixel)	maximal error
0.0	0.000498	0.001140	0.002602	0.004097
0.1	0.239174	0.684851	0.534527	1.665161
0.2	0.578762	1.580720	1.128229	3.229560
0.3	0.651290	2.210597	1.468576	5.399820
0.4	0.880714	2.859214	1.885103	5.604861
0.5	1.070942	3.617597	2.277143	7.765862
0.75	2.221716	5.302301	4.723322	11.966697
1.0	3.217710	11.444995	7.223663	26.157444
2.0	5.656304	17.884097	9.720329	36.497226

Table 3: *Errors on the synthesized pixels using trilinear tensors on simulated data*

First real scene		Second real scene	
Average error(pixel)	maximal error	Average error(pixel)	maximal error
5.61	36.85	4.41	24.76

Table 4: *Errors on the synthesized pixels using trilinear tensors on real images*

Using equations from (7) and from (6), the Z coordinate of the space point P will be given by the following 4 expressions :

$$\begin{aligned} Z &= \frac{x_2 m^2_{34} - m^2_{14}}{x_1 m^2_{11} + y_1 m^2_{12} + m^2_{13} - x_2(x_1 m^2_{31} + y_1 m^2_{32} + m^2_{33})} \\ Z &= \frac{y_2 m^2_{34} - m^2_{24}}{x_1 m^2_{21} + y_1 m^2_{22} + m^2_{23} - y_2(x_1 m^2_{31} + y_1 m^2_{32} + m^2_{33})} \\ Z &= \frac{x_3 m^3_{34} - m^3_{14}}{x_1 m^3_{11} + y_1 m^3_{12} + m^3_{13} - x_3(x_1 m^3_{31} + y_1 m^3_{32} + m^3_{33})} \\ Z &= \frac{y_3 m^3_{34} - m^3_{24}}{x_1 m^3_{21} + y_1 m^3_{22} + m^3_{23} - y_3(x_1 m^3_{31} + y_1 m^3_{32} + m^3_{33})} \end{aligned} \quad (8)$$

The above 4 equations yield 4 trilinear constraints linking the image coordinates of p_1, p_2, and p_3. These equations depend on the entries of the three projection matrices and the pixel coordinates of the image points.

$$\begin{aligned} x_3 x_2 p_1^T T_3^3 - x_3 p_1^T T_1^3 - x_2 p_1^T T_3^1 + p_1^T T_1^1 = 0 \\ y_3 y_2 p_1^T T_3^3 - y_3 p_1^T T_2^3 - y_2 p_1^T T_3^2 + p_1^T T_2^2 = 0 \\ x_3 y_2 p_1^T T_3^3 - x_3 p_1^T T_2^3 - y_2 p_1^T T_3^1 + p_1^T T_2^1 = 0 \\ y_3 x_2 p_1^T T_3^3 - y_3 p_1^T T_1^3 - x_2 p_1^T T_3^2 + p_1^T T_1^2 = 0 \end{aligned} \quad (9)$$

where, T_i^k denotes the vector $(T_i^{1k}, T_i^{2k}, T_i^{3k})^T$, and $T_i^{jk} = m^2_{ij} m^3_{k4} - m^2_{i4} m^3_{kj}$.

By comparing these four trilinear equations to the ones given in (4), one can deduce that they describe the same thing, that is, the trilinear tensors between three images. However, in that case, the tensors are obtained directly from the projection matrices of the three cameras. This is particularly of interest when synthesizing a new view given two images and the parameters of a third virtual camera.

3.3.1 Tests on simulated images

As with the previous experiments on simulated images, the matrices were calculated using only 15 points. The projection (synthesis) of all 64 points of the simulated scene on the third image is obtained using equation (9). As in previous experiments, we have used the same 2 different geometric configurations. As shown in Table (5), errors in the second configuration are smaller here. The use of projection matrices show no sensitivity to the fact that the three camera centers are in a close-collinear situation.

3.3.2 Tests on real images

Note that the error using the second scene is high (Table (6)). This seems in contradiction with the results using simulated images. However, after investigation, we found one possible explanation: images of the second scene where taken with a poor quality digital camera with a zoom. In that case, the pinhole model did not fit well and the calculated projection matrices were of a poor quality.

3.4 View synthesis using 3D projective reconstruction

It is possible to recover the projective three-dimensional structure of a scene using only pixel correspondences between at least 2 images. Once this 3D structure is calculated, it can be projected on a third image. We have assumed for all methods that at least 8 points matched across three images are available. The calculation of the projection matrix of the third image is straightforward. In fact, it can be calculated using a minimum of 6 points in the third image.

3.4.1 Tests on simulated images

As in the previous experiments, only 15 points were used to calculate the projection matrices in the projective space. Points to be synthesized are first reconstructed from two images, then projected on the third image. Table (7) summarizes the results.

Because this approach involves an extra step, the 3D reconstruction, the errors on synthesized view seems to be slightly higher on average. However, there is no apparent sensitivity to the geometric configuration. Unlike the approach using epipolar geometry, the center of projection of the image to be synthesized can be collinear with the other two centers.

3.4.2 Tests on real images

We got the same problem here (see Table (8)) where the error using the second scene is high. The same explanation can be given here: images of the second scene where taken with a poor quality digital camera with a zoom. In that case, the pinhole model did not fit well.

4 Discussion

This comparative analysis suggests the following:

- When the center of projection of the view to be synthesized is not collinear with the centers of projection of the reference views, epipolar geometry and the trilinear tensors have comparable performance. However, because the fundamental matrices where calculated using normalized pixels in the images and the tensors were calculated using a standard SVD method, the results have shown better accuracy with the use of epipolar geometry. Before drawing a final conclusion, we believe that better numerical methods should be used for calculating the trilinear tensors. For instance, the one proposed by Faugeras [12] has shown improved results. We did not use that method because it is a nonlinear method.

Noise(pixel)	Configuration 1		Configuration 2	
	Average error(pixel)	maximal error	Average error(pixel)	maximal error
0.0	0.239424	0.368459	0.169346	0.276857
0.1	0.722834	1.928672	0.410602	0.807273
0.2	1.283713	4.084165	0.587616	1.821946
0.3	1.818731	4.836686	0.780097	2.237783
0.4	2.015791	5.935629	0.992986	2.713951
0.5	2.307910	7.831862	1.194906	3.620521
0.75	4.442688	12.627440	2.100858	5.370815
1.0	6.526484	34.391938	3.737762	12.532149
2.0	8.209053	24.535249	4.803840	18.036824

Table 5: *Errors on the synthesized pixels using projection matrices on simulated images.*

First real scene		Second real scene	
Average error(pixel)	maximal error	Average error(pixel)	maximal error
2.79	8.98	24.75	48.57

Table 6: *Errors on the synthesized pixels using projection matrices on real images.*

Noise(pixel)	Configuration 1		Configuration 2	
	Average error(pixel)	maximal error	Average error(pixel)	maximal error
0.0	0.315411	0.746644	0.195996	0.479370
0.1	0.577417	1.320187	0.369445	0.830718
0.2	0.841729	2.761954	0.436365	1.185088
0.3	1.506497	4.162186	0.717207	2.353235
0.4	1.573408	3.680879	0.750951	2.262745
0.5	2.178871	6.095864	1.155766	3.089189
0.75	3.490345	9.595356	1.671494	4.133488
1.0	4.137170	11.879855	2.317133	6.482530
2.0	6.934911	21.679595	4.218296	12.908194

Table 7: *Errors on the synthesized pixels using projective reconstruction on simulated images*

First real scene		Second real scene	
Average error(pixel)	maximal error	Average error(pixel)	maximal error
2.73	7.25	6.36	18.98

Table 8: *Errors on the synthesized pixels using projective reconstruction on real images.*

- The method using the projection matrices to find the trilinear tensors seems to be more stable although on average errors are slightly high. It did not show sensitivity when the three centers of the cameras were in a close-collinear situation. The results of this approach are comparable to the results obtained with explicit 3D reconstruction.

- Using explicit 3D reconstruction involves a cost, an extra computational step. However, this step was linear and straightforward. The result here were more stable, although less accurate on average. This method has the advantage to be applicable in all geometric configurations.

5 Conclusion

This paper presented a quick overview and a comparative study of all known methods for view synthesis. The results of this study are of great importance to any work involving view synthesis.

As the study suggests, using epipolar geometry yielded more accurate results than using the trilinear tensors. However, new numerically well conditioned methods for calculating the tensors might change our conclusion. One major drawback with the use of epipolar geometry is that it does not work when the the center of projection of the view to be synthesized is collinear with the centers of projection of the reference images. Using the projection matrices yielded a less accurate results but the method was more stable. In particular, this method does not suffer from the collinearity of the centers of projection. The use of the explicit 3D reconstruction has shown more stability but it involved one more computational step. The latter, is not a costly one since it is a linear and straightforward step. Depending on the context, one of these approaches can be used for novel view synthesis.

References

[1] S. Avidan and A. Shashua. Novel view synthesis by cascading trilinear tensors. IEEE *Transactions on Visualization and Computer Graphics*, 4(4):293–306, 1998.

[2] B. Boufama and R. Mohr. A stable and accurate algorithm for computing epipolar geometry. *International Journal of Pattern Recognition and Artificial Intelligence*, 12(6):817–840, 1998.

[3] O. Faugeras. What can be seen in three dimensions with an uncalibrated stereo rig? In G. Sandini, editor, *Proceedings of the 2nd European Conference on Computer Vision, Santa Margherita Ligure, Italy*, pages 563–578. Springer-Verlag, May 1992.

[4] O. Faugeras. *Three-Dimensional Computer Vision - A Geometric Viewpoint*. Artificial intelligence. M.I.T. Press, Cambridge, MA, 1993.

[5] O. Faugeras and L. Robert. What can two images tell us about a third one? In J.O. Eklundh, editor, *Proceedings of the 3rd European Conference on Computer Vision, Stockholm, Sweden*, pages 485–492. Springer-Verlag, 1994.

[6] R. Hartley. In defence of the eight-point algorithm. IEEE *Transactions on Pattern Analysis and Machine Intelligence*, 19(6):580–593, 1997.

[7] R. Hartley, R. Gupta, and T. Chang. Stereo from uncalibrated cameras. In *Proceedings of the Conference on Computer Vision and Pattern Recognition, Urbana-Champaign, Illinois, USA*, pages 761–764, 1992.

[8] T. Kanade, P.J. Narayanan, and P.W. Rander. Virtualized reality: Concepts and early results. In *Workshop on Representation of Visual Scenes, Cambridge, Massachusetts, USA*, pages 69–76, June 1995.

[9] J. Lengyel. The convergence of graphics and vision. *IEEE Computer*, 31(7):46–53, 1998.

[10] S. Maybank. *Theory of Reconstruction from Image Motion*. Springer-Verlag, 1993.

[11] R. Mohr, B. Boufama, and P. Brand. Understanding positioning from multiple images. *Artificial Intelligence*, (78):213–238, 1995.

[12] T. Papdopoulo and O. Faugeras. A new characterization of the trifocal tensor. In Sprin ger Verlag, editor, *Proceedings of The Fifth European Conference on Computer Vision, Freiburg, Germany*, volume 1, pages 109–123, June 1998.

[13] J. Park and S. Inoue. Arbitrary view generation from multiple cameras. In *Proceedings of the* IEEE *International Conference on Image Processing*, volume 1, pages 149–152, 1997.

[14] J. Ponce and Y. Genc. Epipolar geometry and linear subspace methods: A new approach to weak calibration. *International Journal of Computer Vision*, 28(3):223–243, 1998.

[15] C. Rothwell, O. Faugeras, and G. Csurka. A comparison of projective reconstruction methods for pairs of views. *Computer Vision and Image Understanding*, 1997. to appear.

[16] A. Shashua. Algebraic functions for recognition. IEEE *Transactions on Pattern Analysis and Machine Intelligence*, 17(8):779–789, 1994.