

# 1D dense disparity estimation for 3D reconstruction

Lionel Oisel, Étienne Mémin, Luce Morin and Franck Galpin

## Abstract

We present a method for fully automatic 3D reconstruction from a pair of weakly calibrated images in order to deal with the modeling of complex rigid scenes. A 2D triangular mesh model of the scene is calculated using a two-step algorithm mixing sparse matching and dense motion estimation approaches. The 2D mesh is iteratively refined to fit any arbitrary 3D surface. At convergence, each triangular patch corresponds to the projection of a 3D plane. The algorithm proposed here relies first on a dense disparity field. The dense field estimation modeled within a robust framework is constrained by the epipolar geometry. The resulting field is then segmented according to homographic models using iterative Delaunay triangulation. In association with a weak calibration and camera motion estimation algorithm, this 2D planar model is used to obtain a VRML-compatible 3D model of the scene.

## I. INTRODUCTION

The evolution of techniques and hardware used in computer graphics allows more and more realistic representations of our surrounding world. These representations are of interest in many contexts:

- virtual reality such as navigation in well-known historic buildings or 3D video games;
- electronic business: the goal is here to transmit a 3D model of a desired object to potential buyers to offer many different viewpoints;
- dangerous environment simulators (nuclear or military applications for instance).

Though real time rendering techniques are already available, the modelization step is still very time consuming as it is manually performed. In this paper, we present a method for complex scene modeling from sets or sequences of images with unknown camera motion, in the context

IRISA, Campus de Beaulieu, Rennes, France. email : luce.morin@irisa.fr

of a static scene, or equivalent modeling for images acquired simultaneously from different view-points. This model is then used for generating intermediate or extrapolated views.

Many recent works attempt to deal with 3D reconstruction from set of images. Two different classes of approaches are generally proposed using different types of information: the first one includes model-based methods and the second one deals with model-free methods.

In model-based approaches, the scene information is assumed to be composed of large polygonal objects described by a limited set of 3D points characterizing the vertices of each 3D plane. This model can be computed in the 2D space without 3D information. This can be done by extracting, matching and 3D reconstructing points of interest [7] or edges [10]. One of the main limitations of these methods is that effective planarity of generated facets is assumed but not always satisfied. To enforce a global planarity, a manual intervention is even usually necessary to indicate reliable coplanar points. Another way of estimating the model is to use disparity maps (or alternatively depth maps). In [13], Koch *et al.* suggest computing differential properties from a dense disparity map. Images are then segmented according to similar surface orientation at each point of a region. The underlying strongly polyhedral assumption is indeed the major limitation of model-based techniques.

To enlarge the variety of treated scenes, model-free representations (second class of approaches) have been proposed. Such methods generally rely on a dense disparity map. This map can be combined with weak or strong calibration information to provide a depth map that can be manipulated for view synthesis [7], [12]. The major limitation consists here in the estimation of reliable dense disparity information allowing occlusion areas and spatial discontinuities to be coped with efficiently.

The main objective of our study is to propose an entirely automatic approach for the reconstruction of not necessarily polyhedral textured scenes. In addition to this non-specialized goal, we impose to have the ability of an easy and real time visualization. This latter requirement dismisses practically the use of methods based entirely on a dense depth map. On the other hand, the removal of the polyhedral scenes assumption favors such approaches. Following these two remarks and in order to comply with the previously described goals, our aim is to suggest a compromise between model-free and model-based methods. We first propose to describe the 3D scene by a triangular mesh which can be displayed by most visualization dedicated systems. Our method therefore belongs to the first class (model-based approaches) but as this triangular

mesh is automatically computed from a dense disparity field, it is also related to the second class.

The key point of our method is to segment the images into regions which are actually planar in the 3D scene and to extract the planarity property from the image data (and not from a user intervention). This is indeed equivalent to realizing motion segmentation according to an homographic model. As the homographic model describing the set of admissible transformations of planar patches is non-linear, a direct region-based segmentation method is hardly feasible. We have therefore designed a two-step method. The first step provides a geometrically constrained dense depth map and an associated discontinuity map. This dense information is then used to initialize the second step: homographic model estimation and segmentation.

The outline of the paper is the following. The first section briefly describes geometric definitions associated with perspective projection of two images. In this context, epipolar geometry is presented. This important geometric constraint is used in all the following steps of our method and has to be previously estimated.

In the second section, in order to facilitate a subsequent planar facet segmentation step, we present a geometrically constrained disparity field estimation. This technique is derived from a robust optical flow estimation approach. Unlike classical correlation methods, it provides a reliable piecewise smooth motion field [3], [19]. Moreover the disparity estimation is constrained by the associated epipolar geometry so that the estimated field is explicitly forced to be geometrically consistent with a perspective projection model and with the fixed scene assumption. This constraint also yields a substantial computational cost decrease (the 2D disparity estimation problem is reduced to a 1D problem).

The third section presents the planar facet segmentation step of our method. To ensure the effective planarity of each reconstructed triangle, an adaptive iterative triangulation based on homographic models estimation is computed from the disparity field.

By arbitrarily fixing intrinsic parameters, 3D rotation and translation parameters can be extracted from the epipolar geometry. Using this 3D information, the resulting 2D model is then re-projected in the 3D space to be visualized as a VRML representation.

This method has been validated on synthetic and real world images. Comparison with existing classical techniques are presented in the last section of the paper.

*Remark:* in the following, vectors will be represented by bold letters.

## II. EPIPOLAR GEOMETRY

### A. Definition

The characterization of the geometry associated with the two cameras is of key importance in order to build a 3D model of the scene. In our case, we deal with two uncalibrated cameras (or alternatively one moving camera shooting a rigid scene) assuming a pinhole camera model. This model characterizes the projection of a 3D point  $\mathbf{P}(X, Y, Z)$  on a point  $\mathbf{p}(x, y)$  of the image plane. In the case of two images, the projection model is defined by a system of two equations linking a 3D point  $\mathbf{P}(X, Y, Z)$  to its projections  $\mathbf{p}_1(x, y)$  in the first image and  $\mathbf{p}_2(x', y')$  in the second one. Without loss of generality, we assume that the world coordinate system coincides with the first camera coordinate system. The resulting system can be written using homogeneous coordinates as follows (where  $\tilde{\cdot}$  denotes homogeneous coordinates):

$$\begin{aligned}\tilde{\mathbf{p}}_1 &= A_1 [I \ 0] \tilde{\mathbf{P}} \\ \tilde{\mathbf{p}}_2 &= A_2 [R \ \mathbf{t}] \tilde{\mathbf{P}}\end{aligned}\tag{1}$$

$R$  is the rotation matrix and  $\mathbf{t}$  is translation vector defining the second camera location (extrinsic parameters). More precisely,  $R$  and  $\mathbf{t}$  are the orientation and position of the first camera expressed in the second camera coordinate system. Matrix  $A$  contains internal camera parameters (intrinsic parameters).

Let  $\mathbf{C}_1$  and  $\mathbf{C}_2$  be the camera optical centers as well as the centers of camera coordinate systems.  $\mathbf{p}_1$  belongs to line  $(\mathbf{C}_1, \mathbf{P})$  and  $\mathbf{p}_2$  belongs to line  $(\mathbf{C}_2, \mathbf{P})$ . Thus  $\mathbf{C}_1$ ,  $\mathbf{C}_2$ ,  $\mathbf{p}_1$ ,  $\mathbf{p}_2$  and  $\mathbf{P}$  are coplanar points. This coplanarity constraint implies:

$$\overrightarrow{\mathbf{C}_2 \mathbf{p}_2} \cdot (\overrightarrow{\mathbf{C}_2 \mathbf{C}_1} \times \overrightarrow{\mathbf{C}_1 \mathbf{p}_1}) = 0\tag{2}$$

where  $\cdot$  denotes the scalar product and  $\times$  denotes the cross product. In the second camera coordinate system,  $\overrightarrow{\mathbf{C}_2 \mathbf{p}_2} = A_2^{-1} \tilde{\mathbf{p}}_2$ ,  $\overrightarrow{\mathbf{C}_2 \mathbf{C}_1} = \mathbf{t}$  and  $\overrightarrow{\mathbf{C}_1 \mathbf{p}_1} = R A_1^{-1} \tilde{\mathbf{p}}_1$ .

Substituting these values in equation (2) leads to a relation linking the projections of a 3D point in both images:

$$\tilde{\mathbf{p}}_2^T A_2^{-T} [\mathbf{t}]_{\times} R A_1^{-1} \tilde{\mathbf{p}}_1 = 0,\tag{3}$$

where  $[\mathbf{t}]_{\times}$  denotes the cross product matrix associated with the translation vector.

This constraint called *epipolar constraint* has been first introduced by Longuet-Higgins [15]. It is entirely defined by a  $3 \times 3$  homogeneous matrix called the *fundamental matrix* formulated as

$F_{12} = A_2^{-T}[\mathbf{t}]_{\times} R A_1^{-1}$ . By construction, this matrix is of rank 2 and is defined up to a non-zero scalar factor. A fundamental matrix has therefore only seven degrees of freedom.

The fundamental matrix can be used to determine the *epipolar line*  $\mathbf{l}_2$  in the second image associated with point  $\mathbf{p}_1$ . Line  $\mathbf{l}_2$  is defined by:

$$\tilde{\mathbf{l}}_2 = F_{12} \tilde{\mathbf{p}}_1 \quad (4)$$

where  $\tilde{\mathbf{l}}_2$  denotes homogeneous coordinates of  $\mathbf{l}_2$ , i.e. all points in  $\mathbf{l}_2$  satisfy  $\tilde{\mathbf{p}}^T \tilde{\mathbf{l}}_2 = 0$ . The epipolar constraint (3) can now be written as:

$$\tilde{\mathbf{p}}_2^T F_{12} \tilde{\mathbf{p}}_1 = \tilde{\mathbf{p}}_2^T \tilde{\mathbf{l}}_2 = 0, \quad (5)$$

which states that the correspondent  $\mathbf{p}_2$  of point  $\mathbf{p}_1$  belongs to  $\mathbf{l}_2$ . The epipolar line  $\mathbf{l}_2$  thus defines the set of admissible correspondants for point  $\mathbf{p}_1$ .

### B. Case specific application

The issue we address is the recovery of 3D information from sets of 2D images. It consists in solving system (1) to obtain the 3D point  $\mathbf{P}$ . To that end, corresponding points  $\mathbf{p}_1$  and  $\mathbf{p}_2$  and calibration parameters (extrinsic and intrinsic parameters) giving  $A_1$ ,  $A_2$ ,  $R$ ,  $\mathbf{t}$  have to be recovered. The first issue can be greatly simplified by constraining the matching process with the epipolar geometry while the second one can be achieved using a decomposition of the fundamental matrix (see section V). The epipolar geometry estimation is indeed a crucial key point of our method. The next paragraph will present the method we use to recover the fundamental matrix from two uncalibrated images.

### C. Fundamental matrix estimation

We assume here that corresponding points have been extracted and matched using a Harris and Stephens detector associated with a cross correlation process. This first step is equivalent to the one developed by Zhang [24].

To take into account the nullity of the fundamental matrix determinant, we follow a method proposed by Boufama *et al.* based on the virtual parallax [5]. This method may be briefly described as follows. The fundamental matrix is first estimated from 8 matches: three of them are selected to perform a projective change of basis to constraint the matrix to be of rank 2. A

fourth arbitrary pair is also added to complete the projective change of basis [5]. The four last pairs are then used to provide a unique fundamental matrix solution which respects the rank 2 constraint (determinant of null value).

In association with the determinant nullity constraint, the change of basis provides a normalization effect on points coordinates: the coordinates of the three points selected to characterize the new basis are assigned to values between 0 and 1. This involves that coordinates of points belonging to the triangle defined by these points also belong to the range of 0 to 1. It has been shown that normalization is a critical point for getting a well-conditioned system of equations [9]. If pixel coordinates are used directly without normalization, the linear closed form solution is not reliable due to numerical instability. In order to perform normalization for all points, the pairs of points are chosen as near as possible to image corners.

Besides, to cope with erroneous matches, a robust estimation based on least median squares estimation is incorporated [21].

### III. DENSE DISPARITY FIELD ESTIMATION

#### A. Constrained optical flow expression

Let  $I_i(\mathbf{s})$  be the intensity in the  $i$ th image, where  $\mathbf{s} = (x, y)$  denotes the spatial position of pixel  $s$  on the image grid  $S$ . Assuming a constant intensity along motion trajectories, the brightness constancy assumption is expressed as:

$$DFD(\mathbf{s}, \mathbf{d}_s) = I_1(\mathbf{s}) - I_2(\mathbf{s} + \mathbf{d}_s) = 0, \quad (6)$$

where  $DFD$  stands for the Displaced Frame Difference function and  $\{\mathbf{d}_s = (d_x, d_y), s \in S\}$  for the image displacements from position 1 to position 2. In the general case, this is a 2D problem: for each pixel  $s$ ,  $d_x$  and  $d_y$  have to be recovered.

Using the epipolar constraint, it is possible to decompose the displacement vector  $\mathbf{d}_s$  into normal and tangential components with respect to the epipolar line (see Figure 1). The brightness constancy assumption is therefore rewritten as:

$$DFD(\mathbf{s}, \mathbf{d}_s) = I_1(\mathbf{s}) - I_2(\mathbf{s} + \mathbf{n}_s + \lambda_s \mathbf{v}_s) = 0. \quad (7)$$

The normal component  $\mathbf{n}_s$  and the unit vector on the epipolar line  $\mathbf{v}_s$  can be computed from the fundamental matrix for any position  $\mathbf{s}$  (see eq. 4). The enforcement of the epipolar constraint

at every point reduces the original 2D estimation problem to a 1D problem: the estimation of a scalar field  $\lambda = \{\lambda_s, s \in S\}$  along epipolar lines.

The DFD expression is highly non linear with respect to the displacements. To avoid a tough non linear estimation, a Taylor expansion of this equation is considered around point  $\mathbf{s} + \mathbf{n}_s$ . This linearization leads to a constrained optical flow equation:

$$I_1(\mathbf{s}) - I_2(\mathbf{s} + \mathbf{n}_s) - \lambda_s \mathbf{v}_s \cdot \nabla I_2(\mathbf{s} + \mathbf{n}_s) = 0, \quad (8)$$

where  $\nabla$  is the spatial gradient.

This equation relies nevertheless on an inherent ambiguity. The fundamental matrix defining epipolar lines is well known to be far more reliably estimated for large displacements between two camera view points. This large displacements assumption somewhat contradicts the infinitesimal disparity hypothesis implicitly associated to the Taylor expansion. To overcome this incompatibility, the estimation is embedded in a coarse-to-fine multiresolution scheme.

### B. Multiresolution scheme

In motion estimation, a multiresolution setup consists to rely on two pyramid images  $I_i^k$ ,  $i = 1, 2$ ,  $k = 0, \dots, K$  derived from the original images by successive Gaussian smoothing and a regular subsampling by a factor of two in each direction. The resolution index  $k$  spans from  $K$  (the coarsest resolution) to 0 (the finest resolution). The created pyramids allow to incrementally estimate the unknown disparity field by successive estimation on data spaces defined at different scales. The low resolution component of the disparity is estimated at the coarsest level where the domain of validity of the linearized data model is larger due to the joint reduction of the displacement magnitude (through subsampling) and of the image gradient (through smoothing).

To that end, at a given level  $k$ , the disparity  $\lambda^k = \{\lambda_s^k, s \in S\}$  is decomposed into an unknown refinement  $d\lambda^k = \{d\lambda_s^k, s \in S\}$  to be estimated and a known coarse disparity field  $\widehat{\lambda}^k$  obtained from the projection onto the current level  $k$  of the previous level displacement field,  $\Phi(\mathbf{d}^{k-1})$ . The projection is here defined through the interpolation operator  $\Phi$ .

Considering the brightness constancy assumption for the total displacement at level  $k$  yields the following equation:

$$I_1^k(\mathbf{s}) - I_2^k(\mathbf{s} + \mathbf{n}_s^k + [\widehat{\lambda}_s^k + d\lambda_s^k] \mathbf{v}_s^k) = 0, \quad (9)$$

to be solved for all points  $s$  with respect to  $d\lambda_s^k$ .

In addition to pyramids of images  $I_i^k = \{I_i^k(\mathbf{s}), s \in S\}$ ,  $i = 1, 2$ , this equation involves pyramids of tangent and normal vectors  $\mathbf{n}^k = \{\mathbf{n}_s^k, s \in S\}$ ,  $\mathbf{v}^k = \{\mathbf{v}_s^k, s \in S\}$ . These pyramids are obtained considering fundamental matrices  $\{F^k\}$ ,  $k = 0, \dots, K$  deduced for each level from the initial matrix  $F$  and a change of coordinates. More precisely, we have:

$$F^k = M^{kT} F M^k, \quad (10)$$

where  $M^k = \text{diag}(2^k, 2^k, 1)$  is the matrix associated with the considered change of basis involved in the pyramidal representation. The matrix  $F^k$  allows us to compute  $\mathbf{n}_s^k$  and  $\mathbf{v}_s^k$  at resolution  $k$  for each pixel  $s$ . As it is just a change of basis, there is no approximation in the estimation of  $F^k$ . Matrix  $F^k$  provides the exact same epipolar geometry for level  $k$  as  $F$  for full resolution.

Nevertheless, it must be pointed out that due to the interpolation process involved in the projection of displacement field  $\mathbf{d}$  between two consecutive resolution levels, the projected field  $\Phi(\mathbf{d}^{k-1})$  may not respect the epipolar constraint at level  $k$ .

To avoid this problem, and to guarantee that the projected disparity field follows the current epipolar geometry, the coarse component  $\hat{\lambda}^k$  is deduced by projecting the displacement  $\Phi(\mathbf{d}^{k-1})$  onto the epipolar lines at level  $k$  (see Fig. 2).

Such a process allows to properly define a coarse disparity field belonging to the set of admissible solutions at level  $k$ .

### C. Global estimation method

For the sake of clarity, we will omit the resolution upper-script  $k$  in all expressions throughout the remainder of this paper. All the expressions will be meant to concern level  $k$ . Following the same principle as previously, the DFD expression (9) is linearized around point  $\mathbf{s} + \mathbf{n}_s + \hat{\lambda}_s \mathbf{v}_s$ . This leads to a displaced version of the constrained optical flow equation:

$$d\lambda_s \mathbf{v}_s \cdot \nabla \tilde{I}_2(\mathbf{s}) + \tilde{I}_2(\mathbf{s}) - I_1(\mathbf{s}) = 0, \quad (11)$$

where  $\tilde{I}_2(\mathbf{s}) \triangleq I_2(\mathbf{s} + \mathbf{n}_s + \hat{\lambda}_s \mathbf{v}_s)$  is the backward registered version of the second image. Assuming this equation is almost satisfied everywhere and the disparity field is piecewise smooth, the disparity estimation problem may be addressed by the following minimization problem:

$$\hat{d\lambda} = \arg \min_{d\lambda} H(d\lambda) = \arg \min_{d\lambda} H_1(d\lambda, I) + \alpha[H_2(d\lambda)], \quad (12)$$



where  $\alpha$  is an arbitrary fixed constant. The first term  $H_1$  of the objective function  $H$  represents the data model based on the constrained optical flow equation assumption:

$$H_1(d\lambda, I) = \sum_{s \in S} \rho \left[ d\lambda_s \mathbf{v}_s \cdot \nabla \tilde{I}_2(\mathbf{s}) + \tilde{I}_2(\mathbf{s}) - I_1(\mathbf{s}) \right], \quad (13)$$

The second term  $H_2$  is a smoothness term which favors piecewise smooth disparity solutions. This term is expressed over all pairs  $\langle s, r \rangle \in \mathcal{C}$  of mutual neighbors (according to a 4-neighborhood system in our implementation):

$$H_2(d\lambda) = \sum_{\langle s, r \rangle \in \mathcal{C}} \rho(\|\mathbf{d}_s - \mathbf{d}_r\|). \quad (14)$$

To cope with large deviations from the data model (resp. to allow disparity depth discontinuities),  $H_1$  (resp.  $H_2$ ) includes a  $M$ -estimator,  $\rho$ . Under some simple conditions [4], [6], [11], (mainly the concavity of  $\rho(\sqrt{u})$ ), any multidimensional minimization of the form “ $\arg \min_{x_i} \sum_s \rho(g_s(x))$ ” may be turned into a so called semi-quadratic optimization problem of the form “ $\arg \min_{x, z_s} \sum_s \tau z_s g_s(x)^2 + \psi(z_s)$ ” involving new weight variables  $z_s$  continuously lying in  $(0, 1]$ . The function  $\psi$  (which is never used in practice) is a decreasing function depending on  $\rho$ . Parameter  $\tau$  is a scalar normalization depending on the chosen  $M$ -estimator. In our case, the weights are of two natures:

- (a) the data outliers weights,  $\delta = \{\delta_s, s \in S\}$  provided by the semi-quadratic formulation of  $H_1$ , and
- (b) the discontinuity weights  $\beta = \{\beta_{sr}, \langle s, r \rangle \in \mathcal{C}\}$  related to the semi-quadratic formulation of  $H_2$  and lying on the dual edge grid.

The estimation problem is now expressed as a global minimization in  $(d\lambda, \beta, \delta)$  of an extended energy function  $\mathcal{H} = \mathcal{H}_1 + \alpha \mathcal{H}_2$  where:

$$\begin{cases} \mathcal{H}_1 = \sum_{s \in S} \tau_1 \delta_s [d\lambda_s \mathbf{v}_s \cdot \nabla \tilde{I}_2(\mathbf{s}) + \tilde{I}_2(\mathbf{s}) - I_1(\mathbf{s})]^2 + \psi(\delta_s) \\ \mathcal{H}_2 = \sum_{\langle s, r \rangle \in \mathcal{C}} \tau_2 \beta_{sr} \|d\lambda_s \mathbf{v}_s + \hat{\lambda}_s \mathbf{v}_s + \mathbf{n}_s - \mathbf{d}_r\|^2 + \psi(\beta_{sr}) \end{cases},$$

where  $\mathbf{d}_r = (\hat{\lambda}_r + d\lambda_r) \mathbf{v}_r + \mathbf{n}_r$ .

The energy contribution of a point  $s$  to  $\mathcal{H}_1$  is thus weighted by a factor  $\delta_s \in (0, 1]$ : the larger the contribution, the smaller the weight. Similarly, each pair of neighbors  $\langle s, r \rangle \in \mathcal{C}$  contributes to  $\mathcal{H}_2$  with a weight  $\beta_{sr} \in (0, 1]$  depending on their displacement vector difference  $\|\mathbf{d}_s - \mathbf{d}_r\|$ . The larger the difference, the smaller the weight.

The resulting semi-quadratic minimization problem is conducted alternatively with respect to the different variables (here the scalar field  $d\lambda$  and the two weight fields  $\delta$  and  $\beta$ ). The minimization with respect to weights are given in the following closed form [11]:

$$\arg \min_{z_s} \sum_s z_s g_s(x)^2 + \psi(z_s) = \frac{\rho'(g_s(x))}{2g_s(x)}. \quad (15)$$

Now considering weights as being frozen, the minimization with respect to  $d\lambda$  is a classical weighted quadratic problem solved using an iterative method. Using a Gauss-Seidel scheme, the local update  $d\lambda^{(n)}$  at iteration  $n$  of the iterative solver is given for each point  $s$  by:

$$d\lambda_s^{(n)} = \frac{-\tau_1 \delta_s (\tilde{I}_2(\mathbf{s}) - I_1(\mathbf{s})) \mathbf{v}_s \cdot \nabla \tilde{I}_2(\mathbf{s}) + \alpha \tau_2 (\mathbf{v}_s \cdot \boldsymbol{\omega}_s^{n-1} - \hat{\lambda}_s \bar{\beta}_s)}{\tau_1 \delta_s (\mathbf{v}_s \cdot \nabla \tilde{I}_2(\mathbf{s}))^2 + \alpha \tau_2 \bar{\beta}_s}, \quad (16)$$

where  $\boldsymbol{\omega}_s^{n-1}$  is the weighted average of neighboring disparity vectors at iteration  $n-1$  and  $\bar{\beta}_s$  is the sum the spatial discontinuity variables between  $s$  and its neighbors. The whole estimation process problem can be seen as a non linear least squares minimization (Gauss-Newton minimization) of the energy function:

$$\rho(I_1(\mathbf{s}) - I_2(\mathbf{s} + \mathbf{d}_s)) + \alpha \sum_{\langle s,r \rangle \in \mathcal{C}} \rho(\|\mathbf{d}_s - \mathbf{d}_r\|). \quad (17)$$

Such a minimization consists in linearizing the non linear term around a known solution. In motion estimation context, such a minimization is usually embedded into a multiresolution framework [16] and the successive linearizations take place around solutions estimated at coarser resolutions. As for the convergence such a method shares the same deficiency as the Gauss-Newton minimization: it may not converge if the initial solution is too far from the sought minimum. It is therefore important, in case of long range displacements, to initialize the coarsest resolution level ( $k = K$ ) with a reasonable initial disparity field. In our case, we consider an initialization derived from the interpolation of the initial matched points of interest used for the computation of the fundamental matrix. We used here a bilinear interpolation based on a Delaunay triangulation. The resulting displacement field  $\mathbf{d}^K$  is projected on the top level of the pyramid to provide an initial disparity field  $\hat{\lambda}^K$  for the coarsest resolution level with respect to the associated epipolar geometry (projection on the associated epipolar lines (see fig. 2)). Let us note that this dense disparity estimation method shares some common principles with the

method proposed in [1]. Compared to our approach the differences consist mainly in the use of a variational framework together with a Gaussian scale-space approach to recover long range disparities (instead of a hierarchical multiresolution setup). Another difference lies also in the use of the Nagel-Enkelmann operator as a smoothness term[17].

#### IV. SEGMENTATION

As our final goal is to provide a 3D reconstruction of the scene easy to handle, we now introduce a segmentation method of the dense disparity field obtained at the previous step. The method we propose is based on an adaptive triangular mesh structure. The idea of our technique consists in recursively splitting an initial mesh until each triangular element corresponds to a 3D planar element. The associated splitting criterion is based on the homographic parametric model-description of the disparity field. It can be easily shown that, according to a pinhole camera model, the disparity associated with a planar surface projected respectively as  $\Pi_1$  in the first image and  $\Pi_2$  in the second image satisfies an homographic model. This model is linear using homogeneous coordinates. For the sake of clarity, all the following expressions are meant to be expressed in homogeneous coordinates. The homographic model links two corresponding points  $\mathbf{s}$  and  $\mathbf{s} + \mathbf{d}_s$  of  $\Pi_1$  and  $\Pi_2$  with a  $3 \times 3$  homogeneous homography matrix named  $H$  up to a scalar factor  $\mu$ :

$$\forall \mathbf{s} \in \Pi_1, H\mathbf{s} = \mu(\mathbf{s} + \mathbf{d}_s). \quad (18)$$

The segmentation step we propose consists thus in triangulating the disparity map until the disparity vectors associated with each patch correspond to a single representative homographic model. An initial Delaunay triangulation is first performed by taking four arbitrary points near the corners of the image. This triangulation is then refined until each triangle verifies a distance criterion between the dense estimation disparity and a homographic model estimated within the considered triangle.

##### A. Homography estimation

The homography estimation is performed using a method proposed by Robert and Faugeras [20]. The method relies on the epipolar geometry to efficiently estimate the homography matrix from three or more corresponding pairs of points.

For  $H$  to be consistent with epipolar geometry, the homogeneous symmetric matrix  $F^T H + H^T F$  must be null. This leads to 6 homogeneous equations with unknowns  $h_{ij}$  (the coefficients of  $H$ ). In our case, each point of a considered triangle  $T$  accounts for one scalar equation. We have therefore the following system of equations:

$$\forall \mathbf{s} \in T, [\mathbf{s} + \mathbf{d}_s, F\mathbf{s}, H\mathbf{s}] = 0, \quad (19)$$

where  $[a, b, c]$  denotes the triple product.

This over-constrained system can be rewritten in matrix notation as  $A\mathbf{h} = 0$ , where  $\mathbf{h}$  is an 8 components vector gathering the unknown coefficients of  $H$  and  $A$  is a  $(|\{\mathbf{s} \in T\}| + 6) \times 8$  matrix. An estimate of  $\mathbf{h}$  is computed using a SVD (singular value decomposition) of the matrix  $A^t A$ .

To be robust to problematic situations where the estimated disparities are likely to be biased or erroneous (such as occlusion areas or range discontinuities), we exclude from this system points which are not simultaneously in accordance with the data model and the smoothing model (points for which the data outliers and the discontinuity weights approach zero).

### B. Splitting criterion

The distance criterion we chose to handle the splitting of the triangular mesh is decomposed in two terms:

- The first one measures the adequacy of  $H$  to the disparity field. The influence of each point  $\mathbf{s}$  of the triangle is weighted by the data model weight  $\delta_{\mathbf{s}}$  coming from the robust estimator associated with the data model of the dense disparity estimator (occlusion areas do not influence the distance measurement). The resulting adequacy term is given by:

$$C_1(T, H, \mathbf{d}) = \frac{1}{\sum_{\mathbf{s} \in T} \delta_{\mathbf{s}}} \sum_{\mathbf{s} \in T} \delta_{\mathbf{s}} [\|H\mathbf{s} - (\mathbf{s} + \mathbf{d}_s)\|^2 + \|H^{-1}(\mathbf{s} + \mathbf{d}_s) - \mathbf{s}\|^2], \quad (20)$$

where  $\| \cdot \|$  denotes the Euclidean distance.

- The second term is related to the presence of disparity discontinuities within the considered triangle. This term is defined as the mean of discontinuity weights included in the considered triangle. It is expressed as follows:

$$\begin{cases} C_2(T, \beta) = \frac{\sum_{\langle \mathbf{s}, \mathbf{r} \rangle \in \mathcal{C}_T} \beta_{\mathbf{s}, \mathbf{r}}}{|\mathcal{C}_T|} \\ C_T \triangleq \{\langle \mathbf{s}, \mathbf{r} \rangle, \mathbf{s} \in T, \mathbf{r} \in T\}, \mathcal{C}_T \subset \mathcal{C} \end{cases} \quad (21)$$

where  $\langle s, r \rangle$  denotes neighboring pixel of image 1.

More precisely, a given triangle  $T$  will be refined if the global criterion  $C_1(T, H, \mathbf{d}) + \gamma C_2(T, \beta)$  exceeds a given threshold  $\epsilon$ . The parameter  $\gamma$  is an arbitrary fixed positive constant.

### C. Triangulation refinement

The triangulation is refined by adding new points at the “centers of mass” of each triangle which verifies the splitting criterion; the mass of each vertex being here given by the value of their associated data outliers weights. A new Delauney triangulation is then performed taking into account the new set of points (the previous vertices and the new added points). The new triangular mesh is then considered again for splitting. This process is repeated until no triangle verifies the splitting criterion. Using Delaunay triangulation guarantees optimal aspect ratio of the triangular mesh and prevents degeneracy. Moreover the classical incremental algorithm is used, which enables fast updating of the triangulation when the new vertices are inserted.

The overall synopsis of the segmentation scheme is presented in

## V. 3D RECONSTRUCTION

So far we have obtained a 2D triangulation of the first image and an associated disparity information. The last step of our method consists in recovering 3D information in order to build the final 3D model. To that end, the calibration parameters of the cameras have to be provided or estimated.

### A. Estimation of camera parameters

As the aim is not an accurate reconstruction but a visually satisfactory 3D representation we have used a weak calibration technique. This approach consists in assigning some arbitrary values the intrinsic parameters (represented by matrix  $A$ ) and then estimating the extrinsic parameters. The intrinsic parameters are chosen in order to respect the following assumptions: the projection of the optical center is supposed to be at the center of the image, coordinate image axes are perpendicular, horizontal and vertical pixel sizes are fixed and equal to one and the focal length is assigned to a realistic value. The fundamental matrix  $F$  allows to estimate the *essential matrix*  $E$ . This matrix only depends on the extrinsic parameters, i.e. the rotation matrix  $R$  and the

translation vector  $\mathbf{t}$  between the first and the second camera location:

$$E = A^T F A = [\mathbf{t}]_{\times} R, \quad (22)$$

where  $[\mathbf{t}]_{\times}$  is the antisymmetric cross product matrix associated with the translation vector  $\mathbf{t}$ . As shown by Tsai and Huang in [23], the essential matrix can be decomposed in order to recover rotation and translation parameters. Using a singular value decomposition,  $E$  can be written as follows:

$$E = \Delta E' \Theta^t, \quad (23)$$

where  $\Delta$  and  $\Theta$  are two orthogonal matrices and  $E'$  is a diagonal matrix. It can be shown that an essential matrix has one null singular value, while the two others have the same value (they can be assigned to 1 because of the homogeneous property) [23]. Matrix  $E'$  can thus be rewritten as follows:

$$\begin{aligned} E' &= T_1 R_1 \\ &= \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & -1 & 0 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{pmatrix}. \end{aligned} \quad (24)$$

Injecting this decomposition in equation (23) leads to write the matrix  $E$  as a product of an antisymmetric matrix by an orthogonal one. By identification with equation (22), we can extract rotation and translation matrices:

$$E = \underbrace{\Delta T_1 \Delta}_{[\mathbf{t}]_{\times}} \underbrace{\Delta^t R_1 \Theta^t}_R \quad (25)$$

We must notice that this decomposition is not unique. The rotation is only defined up to  $\pi$  while the translation is defined up to a scalar factor. The adequate pair of matrices is obtained by ensuring all 3D reconstructed points to be in view of both of the cameras.

### B. 3D geometry computation and texture correction

The resulting intrinsic and extrinsic parameters lead to two projection matrices. The vertices of the image triangulation are then back-projected into the 3D space by solving system (1) according to the dense disparity field. This 3D point set defines a 3D continuous triangular mesh. Triangular facets are textured using image  $I_1$ , and the result is saved as a VRML format

file. The textured triangular mesh description is well adapted for visualization with real time interactivity thanks to specialized hardware which perform fast rendering of OpenGL calls.

Since texture is extracted from a real image, perspective correction has to be performed prior to mapping. As for most existing formats, VRML renderers assume texture is provided in a front view, with a possible scale factor. In real images, texture of planar areas appear with perspective distortion, which differs from a front view by a 2D homography.

During visualization, texture mapping is usually performed through affine transformation. For a 3D triangular mesh, an affinity is defined for each 3D facet. The 3 vertices of a 2D triangle in the texture image are mapped onto the corresponding 3D facet vertices. Thus any affine transformation of a triangle front-view is a correct texture. A perspective view is not a suitable texture, and it is not compensated by affine mapping. The 3D model then suffers from distorted texture.

The usual way to solve this point is to perform texture correction on each facet. For each triangular facet in the 3D mesh, a front view of the triangle is generated, by computing the appropriate homography. This homography is applied to the corresponding triangle in the 2D mesh, thus generating a corrected texture triangle of different shape and size. One texture image per facet is then necessary, which is time and memory expensive.

We propose a technique that corrects perspective distortion for all triangles and provides a single texture image for the whole scene.

The idea is to perform the correction without modifying the 2D mesh positioning. Only the texture inside each triangle is modified. To do so, we define an homography which transforms the perspective view of a 3D facet into an affine view of the same 3D facet, with conservation of the triangle vertices. The affine mapping performed during visualization will then be correct.

A 2D homography is defined by four points. For a given 2D triangle the proper correction homography  $H$  is defined by the three triangle vertices  $a, b, c$  and a fourth point  $g$ , which is the projection of the 3D facet center of gravity  $G$ .  $H$  leaves  $a, b$  and  $c$  unchanged and transforms  $g$  into  $g'$ , the 2D triangle center of gravity (see figure 4). Once  $H$  is known, the homography is applied on the triangle texture using  $H^{-1}$  and bilinear interpolation. Figure 5 illustrates our texture correction on a synthetic scene.

## VI. RESULTS

The proposed method has been applied on different kinds of image sequences. It has been run both on real world sequences and synthetic sequences for which the actual motion field is known.

The first sequence we are considering here is the well known synthetic “Yosemite” sequence (figure 7). In order to satisfy the rigidity assumption, a major part of the sky containing moving clouds has been removed. Two different image pairs of this sequence have been considered. In the first one, which is composed of two consecutive images (images 11 and 12) the small range of the displacements (not more than 4 pixels) makes critical the estimation of the epipolar geometry. The second image pair, composed of far apart images in the sequence (images 3 and 12), constitutes a difficult benchmark towards the differential aspect of our method (up to 30 pixels of displacement).

As expected and shown on the recovered disparity map (figure 8(a)), the disparities are larger in the mountain area in the foreground and continuously decrease while we move towards the valley. The global aspect of this map is in accordance with what could be expected from visual inspection.

Following [2], we provide quantitative comparative results on this pair of images. Angular deviations with respect to the actual flow field have been computed. The table in figure 6 lists the mean angular value error and associated the standard deviation. It gathers some results presented in [2], and by other authors (only the higher and the lower mean square error obtained by state of the art methods are presented in comparison with the classical Horn and Schunck algorithm). Let us note that we report here only performances of similar algorithms (energy based dense estimators). Other results of more complex method combining motion estimation with a joint segmentation may be found in the literature.

As may be observed, compared to others our method yields to a higher angular discrepancy. Let us note that, nevertheless, it stays satisfactory. A few remarks must be done at this point. First: unlike the best methods mentioned in the table, our method uses a simple iterative solver (Gauss-Seidel). It could be therefore improved by using more efficient solvers. Second: it must be pointed out that our method is a *one-dimensional method*. It is therefore much faster than the others. Besides, due to small motion the epipolar geometry is quite difficult to estimate



accurately.

Let us now consider the second sequence, composed of far apart images (images 3 and 12) of the “Yosemite” sequence. Experiments on this sequence have shown that, due to the presence of very large displacements (up to 30 pixels of displacements), non constrained optical flow estimators (even embedded in a multiresolution framework) do not converge towards acceptable solutions. As shown in the disparity map presented figure 8(b) our method provides consistent results. The foreground mountain is characterized by important disparity values whereas in the background, disparities decrease smoothly. The dense disparity field estimation performs well for an image presenting both small and large displacements. The resulting field is globally smooth and nevertheless presents discontinuities on important depth changes.

The disparity field computed from images 3 and 12 has been then iteratively triangulated to obtain a 3D model of the valley. The final 2D triangulation is shown figure 10(b). Figure 10(a) presents the initial triangulation obtained through a Delaunay triangulation of initial matched points of interest. The associated VRML model has been computed by arbitrarily fixing the focal length to 1000. Figure 11 presents some interpolated images. The camera displacement along the z-axis is not far away from the real 3D motion. The resulting images are visually satisfactory. Figure 12 exemplifies more complex displacements illustrating occlusion problems.

Some reconstruction results obtained for a static scene shot by a moving commercial camera are shown in figures 13, 14, 15, 16 and 17. Two reconstructions are presented here (the views are obtained with the same 3D displacement of the virtual camera). The first one comes from the “image-matching” software, developed by Zhang [24], which gives a list of matching points of interest that respect the epipolar geometry. Those points are triangulated and re-projected to obtain a 3D model. The examples presented in figures 14(a), 15(a), 16(a) and 17(a) are constructed from 89 automatically extracted matching points. The synthesized views outline the presence of outliers points that make the model visually uncomfortable. This effect can mostly be explained by the presence of spurious matches that respect epipolar geometry and luminance consistency. The second 3D model results from our algorithm (figures 14(b), 15(b), 16(b) and 17(b)). A visual inspection of reconstructed images shows far less artifact. Such results could now be used in the context of video manipulation applications.

Let us point out that others results may be found in the PhD thesis of Lionel Oisel [18](in

French).

## VII. CONCLUSION

In this paper we have presented a method for the reconstruction of a complex scene from a pair of weakly calibrated images. This method relies on the estimation of a dense disparity field. The estimator proposed here is constrained by the epipolar geometry and incorporates robust estimation. We have experimentally demonstrated that the recovered fields are of good quality even in unfavorable case (very close views). The final 3D reconstruction is obtained through a segmentation process handled as a recursive adaption of a triangular mesh. The outliers detection provided by the dense robust estimation is also used in the segmentation step to improve the quality of the final reconstruction. The efficiency of our approach has been validated on both polyhedral and non polyhedral complex scenes. The models obtained are sufficiently good to be used in a comfortable way in the context of video manipulation applications. Nevertheless, more accurate results could be expected using self-calibration methods available in the literature. An extension of our algorithm would consist in considering the trifocal tensor (associated with three images [22]) instead of the fundamental matrix  $F$ , to avoid many degenerate estimation cases of  $F$ . This could naturally lead to take into account more than two images to improve the VRML model quality.

## REFERENCES

- [1] L. Alvarez, R. Deriche, J. Weickert and X. Sanchez. Dense disparity map estimation respecting image discontinuities: a pde and scale-space based approach. *J. Visual Com. and Image Representation*, 13(1-2):3-21, 2002.
- [2] J. Barron, D. Fleet, and S. Beauchemin. Performance of optical flow techniques. *International Journal on Computer Vision*, 12(1):43–77, 1994.
- [3] M. Black. Recursive non-linear estimation of discontinuous flow fields. In *Proceedings of the European Conf. on Computer Vision*, pages 138–145, Stockholm, Sweden, 1994.
- [4] M.J. Black and A. Rangarajan. On the unification of line processes, outlier rejection, and robust statistics with applications in early vision. *International Journal on Computer Vision*, 19(1):57–91, July 1996.
- [5] B. Boufama and R. Mohr. Epipole and fundamental matrix estimation using the virtual parallax property. In *Proc. Int. Conf. Computer Vision*, pages 1030–1036, Cambridge, Massachusetts, 1995.
- [6] P. Charbonnier, L. Blanc-Féraud, G. Aubert, and M. Barlaud. Deterministic edge-preserving regularization in computed imaging. *IEEE Trans. on Image Processing*, 6(2):298–311, 1997.
- [7] O. Faugeras and S. Laveau. Representing three-dimensional data as a collection of images and fundamental matrices for image synthesis. In *Proceedings of International Conf. on Pattern Recognition*, pages 689–691, Jerusalem, Israel, 1994.

- [8] O. Faugeras, Q.T. Luong. *The Geometry of Multiple Images*. MIT press 2001.
- [9] R. Hartley, A. Zisserman. *Multiple view geometry in computer vision*. Cambridge University Press, 2000.
- [10] P. Havaladar, M-S. Lee, and G. Medioni. View synthesis from unregistered 2-d images. In *Graphics Interface*, pages 61–69, Toronto, Canada, May 1996.
- [11] D. Geman and G. Reynolds. Constrained Restoration and The Recovery Of Discontinuities *IEEE Trans. on Pattern Analysis and Machine intelligence*, 3(14):367-383, 1992
- [12] T. Kanade, P.J. Narayanan, and P.W. Rander. Virtualized reality: Concept and early results. In *Workshop on Representation of Visual Scenes*, Cambridge, USA, 1995.
- [13] R. Koch, M. Pollefeys, and L. Van Gool. Automatic 3d model acquisition from uncalibrated image sequences. In *Computer Graphics International*, pages 597–604, Hannover, 1998.
- [14] S. Lai and B. Vemuri. Reliable and efficient computation of optical flow. *International Journal on Computer Vision* , 29(2):87–105, 1998.
- [15] H.C. Longuet-Higgins. The reconstruction of a scene from two projections: configurations that defeat the 8-point algorithm. In *Proceedings of the 1st Conference on Artificial intelligence applications*, pages 395–397, Denver, 1984.
- [16] E. Mémin and P. Pérez Hierarchical estimation and segmentation of dense motion fields *International Journal on Computer Vision*, 46(2):129–155, 2002.
- [17] H.H. Nagel and W. Enkelmann. An Investigation of smoothness constraints for the estimation of displacement vector fields from image sequences. *IEEE Trans. on Pattern Analysis and Machine intelligence*, 8:565-593, 1986.
- [18] L. Oisel *Reconstruction 3D de scènes complexes à partir de séquences vidéo non calibrées: estimation et maillage d'un champ de disparité* Thèse de l'université de Rennes I. 1998
- [19] L. Robert and R. Deriche. Dense depth map reconstruction: A minimization and regularization approach which preserves discontinuities. In *Proceedings of the 4th European Conf. on Computer Vision*, volume 1, pages 439–451, 1996.
- [20] L. Robert and O. Faugeras. Relative 3-D positioning and 3-D convex hull computation from a weakly calibrated stereo pair. *Image and Vision Computing*, 13(3):189–197, 1995.
- [21] P.J. Rousseeuw and A.M. Leroy. *Robust Regression and Outlier Detection*. John Wiley & Sons, New York, 1987.
- [22] P.H.S. Torr and A. Zisserman. Robust parameterization and computation of the trifocal tensor. *Image and Vision Computing*, 15:591–605, 1997.
- [23] R. Tsai and T. Huang. Uniqueness and estimation of three dimensional motion parameters of rigid objects with curved surface. *IEEE Trans. on Pattern Analysis and Machine intelligence*, 6:13–26, 1984.
- [24] Z. Zhang. Estimating motion and structure from correspondences of line segments between two perspective images. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 17(12):1129–1139, 1995.

PLACE  
PHOTO  
HERE

**Lionel Oisel** received the PhD degree in computer science from the University of Rennes, FRANCE in 1998. In 2000, he joined Thomson Research , Rennes France. His research interests include computer vision, 3D reconstruction, video indexing using audio-visual analysis.

PLACE  
PHOTO  
HERE

**Étienne Mémin** received the PhD degree in computer science from the university of Rennes I, France, in 1993. From 1994 to 1999, he was assistant professor in computer science at the University of South Brittany (UBS Vannes). He is currently assistant professor in computer science at the University of Rennes I. His major research interest concerns motion analysis (motion estimation, motion segmentation and tracking), energy based modeling and parallel processing. Since a couple of years, he is particularly interested in motion analysis issues for image sequences depicting fluid phenomenon. He will be the scientific leader of the vista group for this project.

PLACE  
PHOTO  
HERE

**Luce Morin** was born in Grenoble, France. She received her engineer degree in physics and numerical images from ENSPS, Strasbourg in 1989 and her PhD degree from INPG, Grenoble in 1993. Since then she has been an assistant-professor at University of Rennes 1, where she teaches computer science, image processing and computer vision. Her research activities, as a member of the Temics project in the IRISA/INRIA Rennes laboratory, deal with 3D modelisation for video sequence communication.

PLACE  
PHOTO  
HERE

**Franck Galpin** was born in Laval, France. He did his graduate studies at DIIC/IFSIC school of the University of Rennes 1, followed by a PhD thesis in computer sciences at IRISA/INRIA Rennes laboratory in the Temics project. His topics of interest deal with video coding and computer vision. He is now a post-doctoral student in Deguchi Laboratory at the University of Tohoku in Japan.

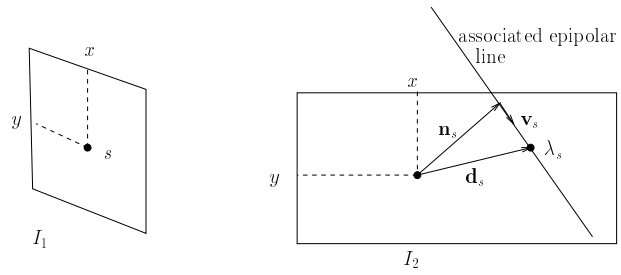


Fig. 1. Displacement vector decomposition.

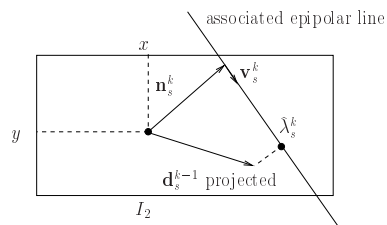


Fig. 2. Projection of the displacement vector  $\mathbf{d}^{k-1}$  according to epipolar geometry at level  $k$ .

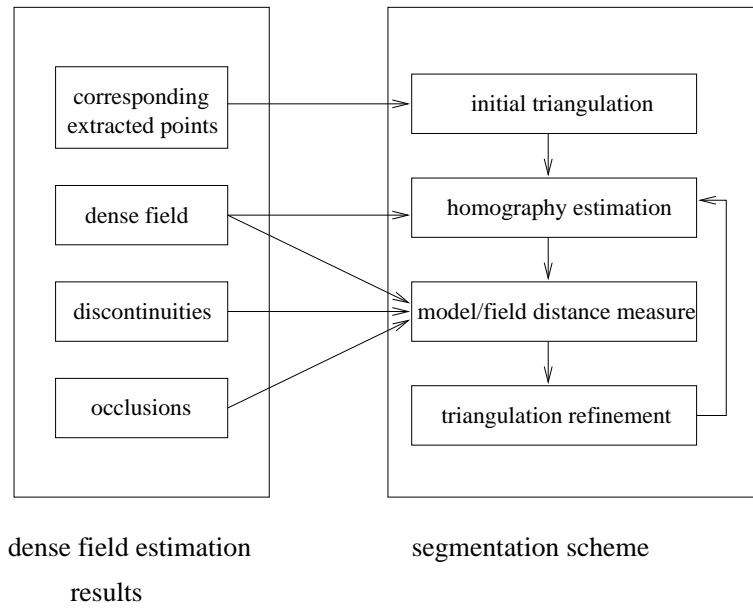


Fig. 3. Segmentation scheme

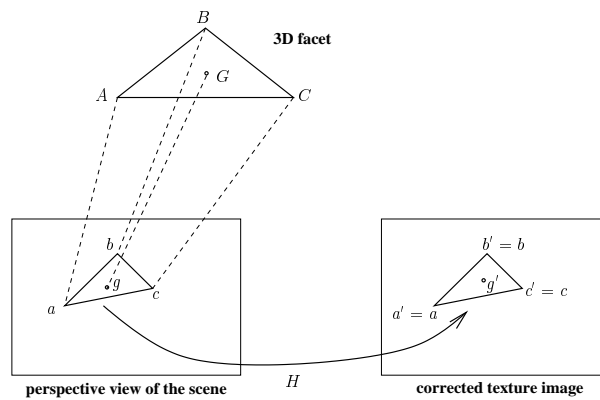


Fig. 4. Extraction and correction of a texture triangle

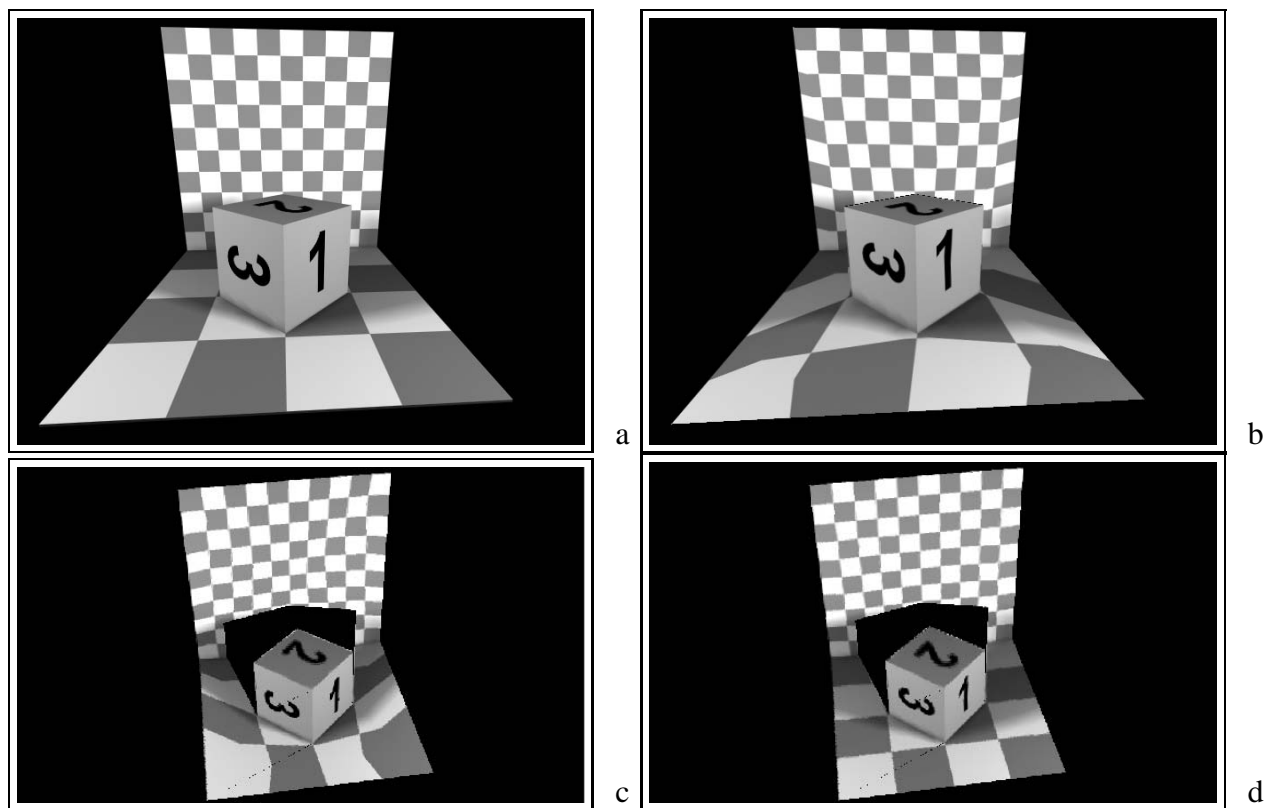


Fig. 5. View of the scene used as texture image, before correction (a) - after correction (b) - View of the 3D reconstructed model, using texture image without correction (c) - using corrected texture image (d).

Technique	Mean error	Standard deviation
Horn and Schunck [2]	$9.78^\circ$	$16.19^\circ$
Black [3]	$3.52^\circ$	$3.25^\circ$
Lai and Vemuri[14]	$1.99^\circ$	$1.45^\circ$
Our method	$4.82^\circ$	$3.27^\circ$

Fig. 6. Comparative results on “Yosemite” sequence.



Fig. 7. Original images 3 (a), 11 (b) and 12 (c) of the “Yosemite” sequence.

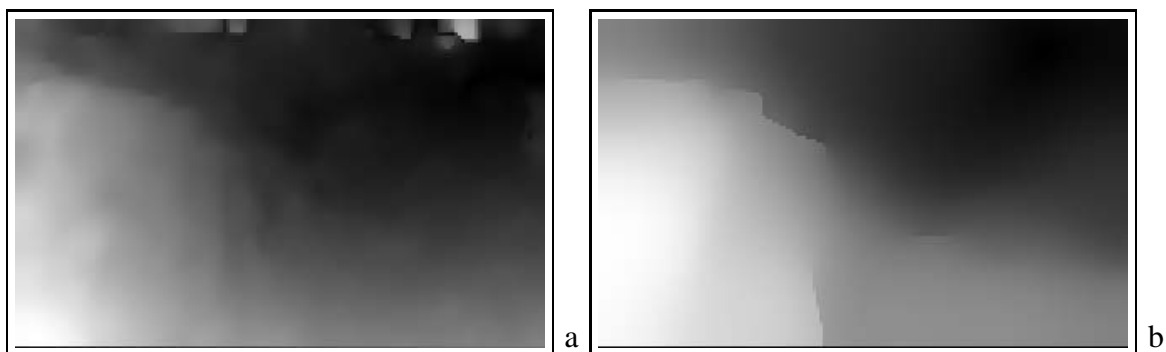


Fig. 8. Disparity map for images 11 and 12 (a) and 3 and 12 (b) (the darker the smaller the disparity value)

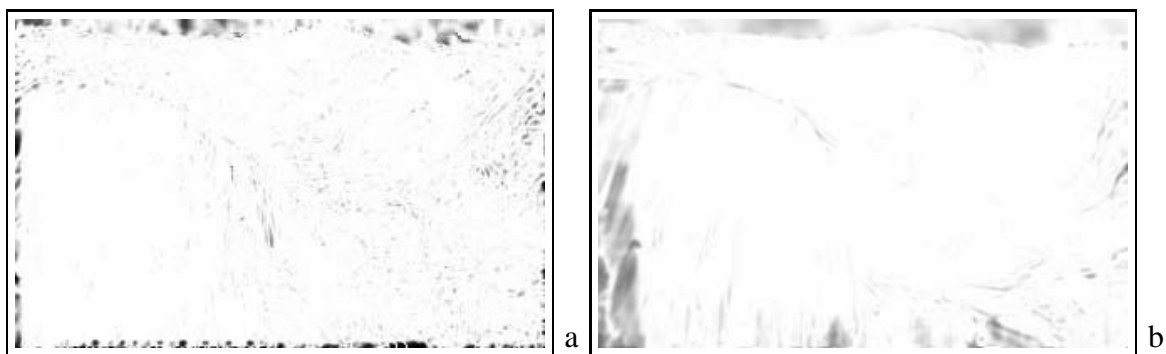


Fig. 9. Outliers due to occlusion for images 11 and 12 (a) and 3 and 12 (b) (the darker the smaller the weight  $\delta_s$ )

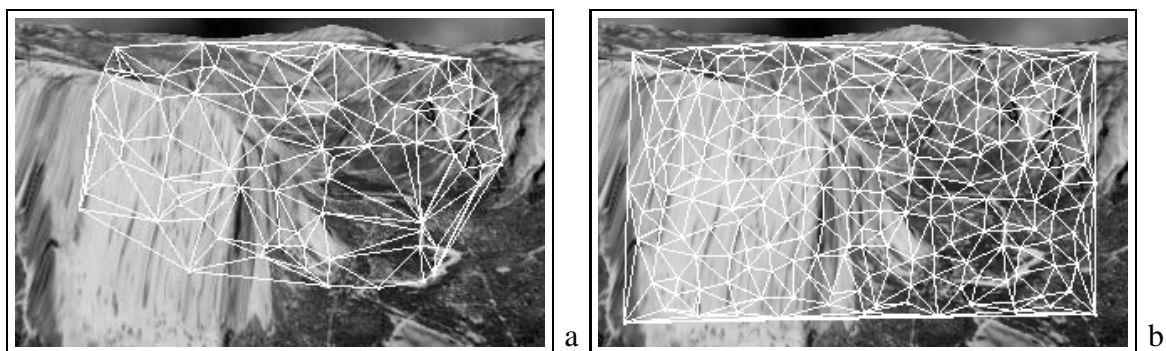


Fig. 10. Points of interest triangulation (a) and resulting 2D triangulation (b) associated with image 3



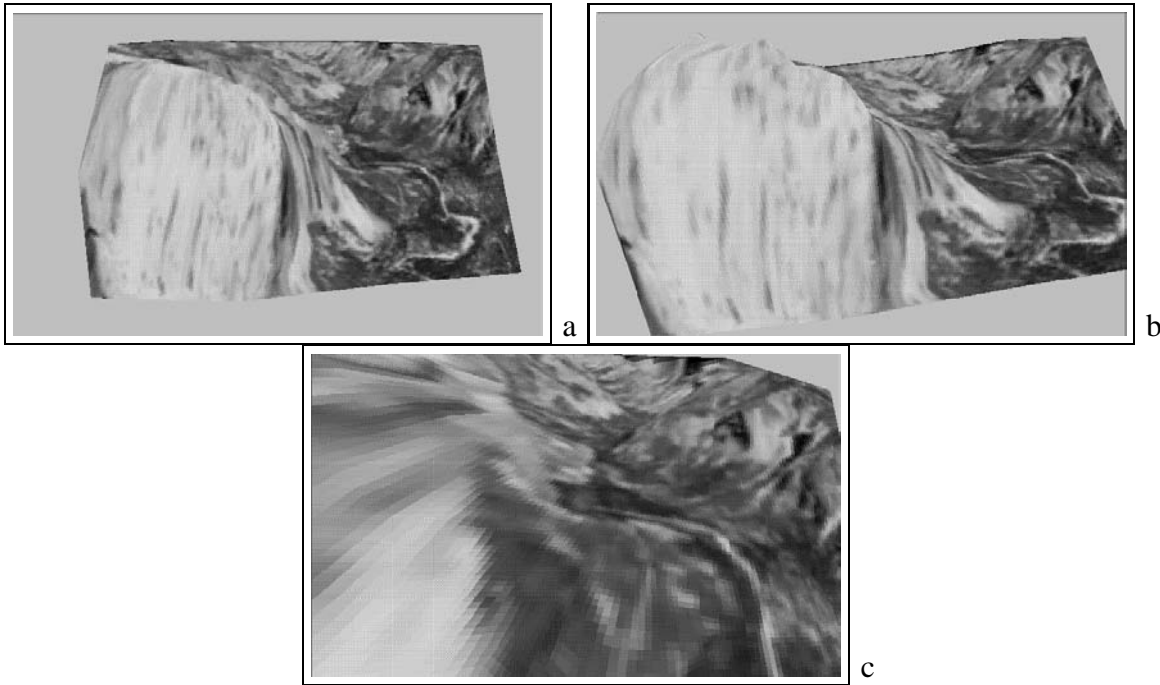


Fig. 11. Translation along Z axis simulations.

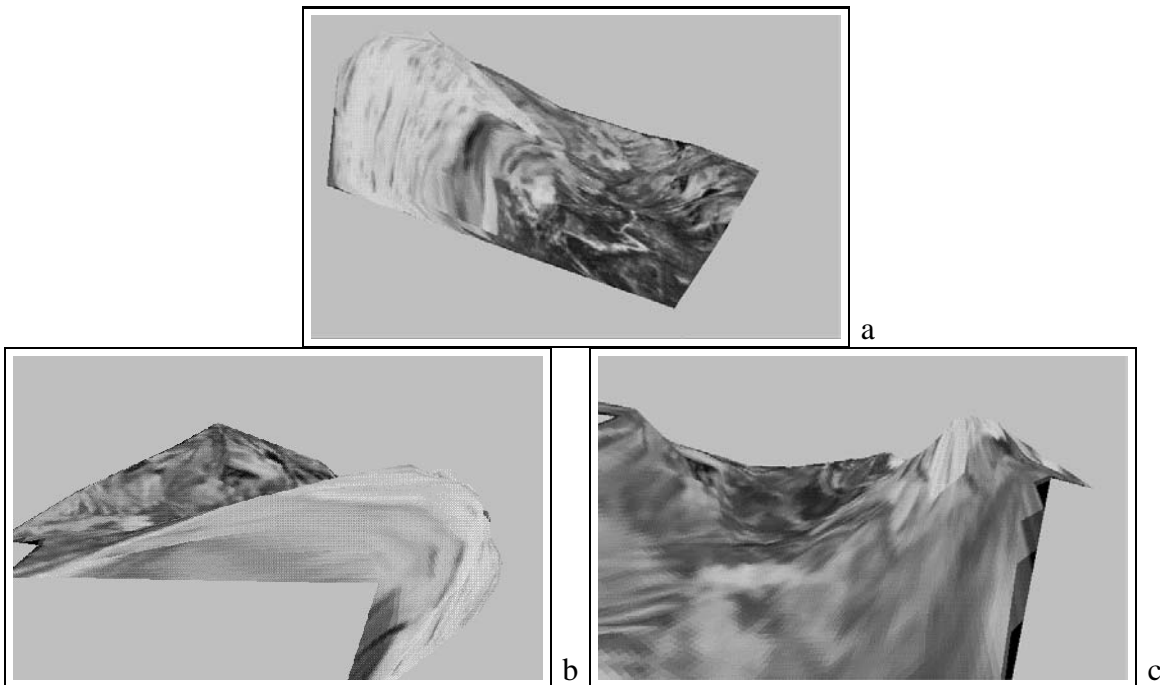


Fig. 12. Complex motion simulation: viewpoint on the right (a) viewpoint on the left (b) and viewpoint behind the foreground mountain (c).

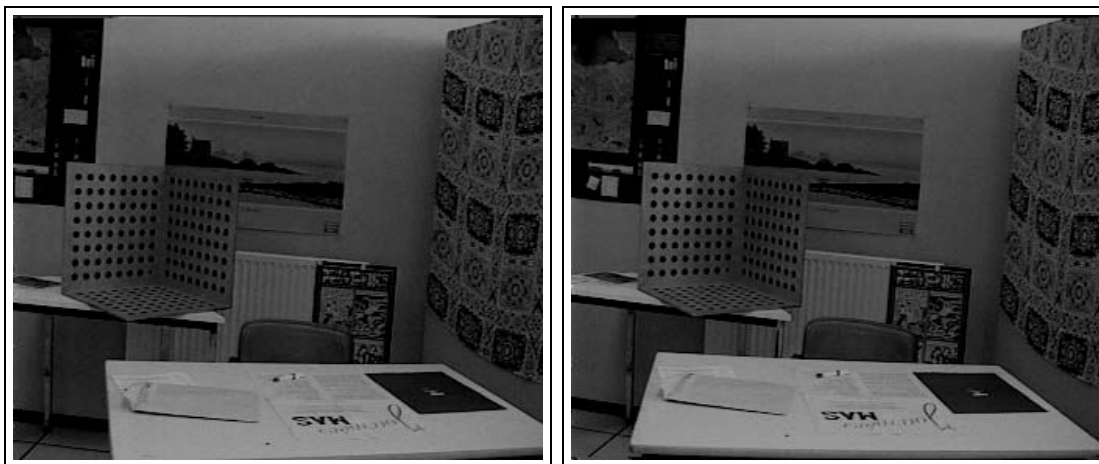


Fig. 13. Two original images of an indoor sequence.

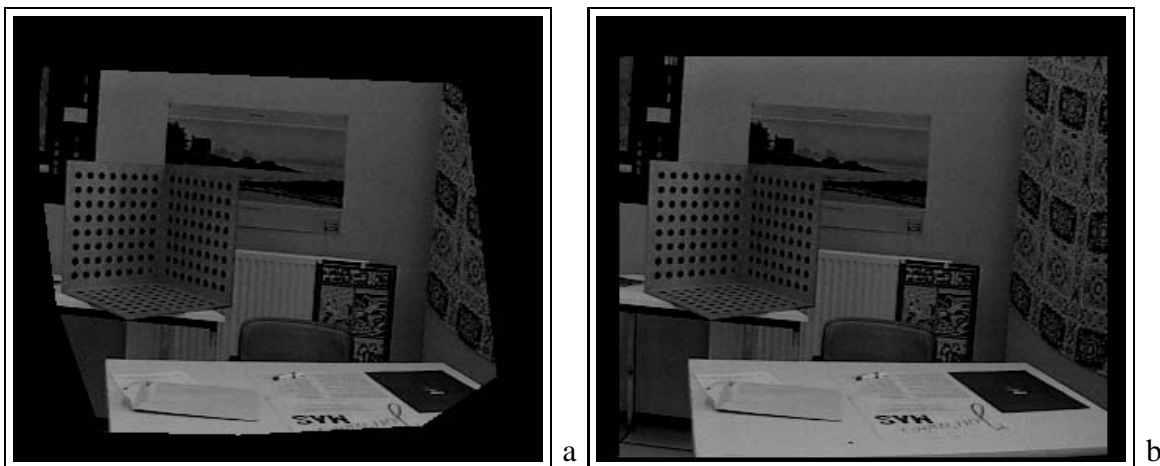


Fig. 14. Left original image resynthesized: model computed directly from automatically extracted and matched points (a), model obtained by our method (b)

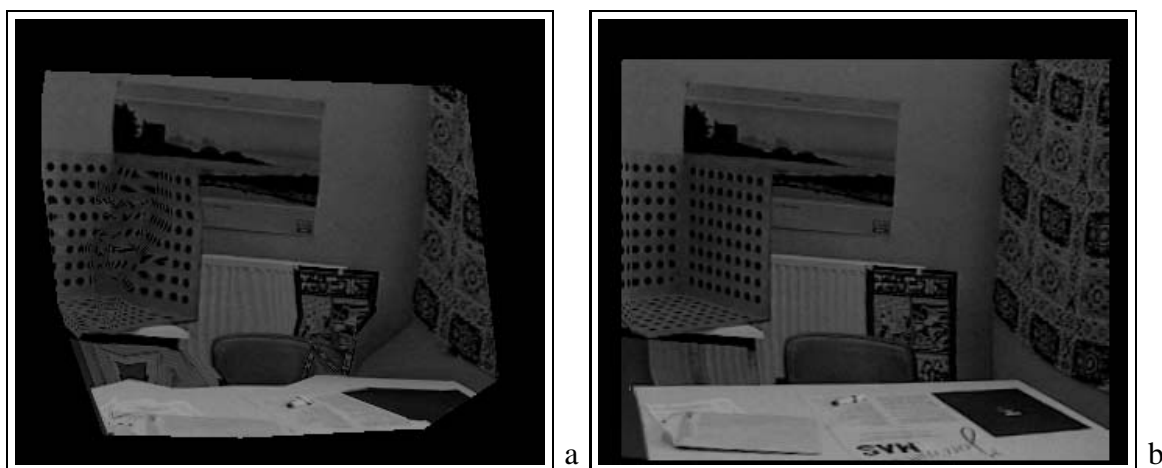


Fig. 15. Two synthesized views from the same view point: model computed directly from automatically extracted and matched points (a), model obtained by our method (b)

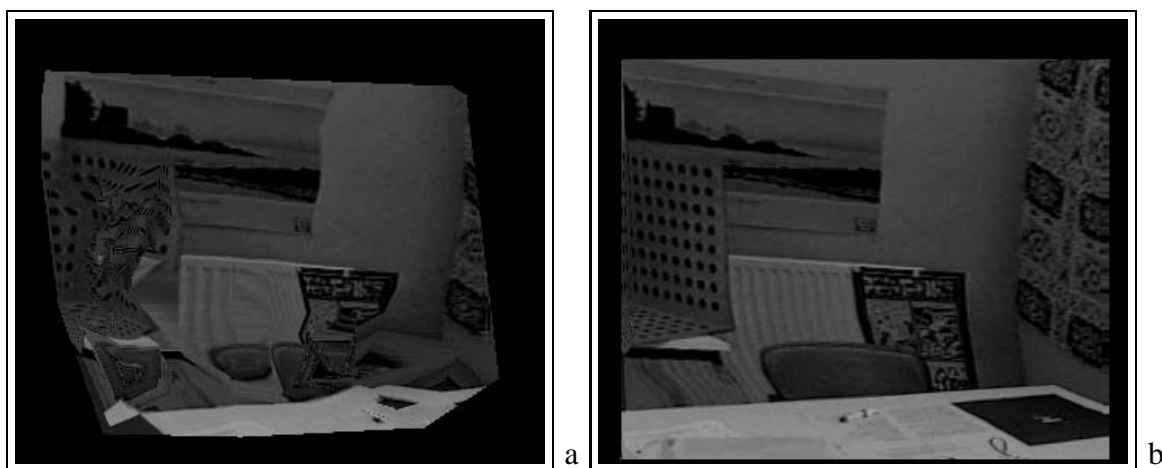


Fig. 16. Two synthesized views from the same view point: model computed directly from automatically extracted and matched points (a), model obtained by our method (b)

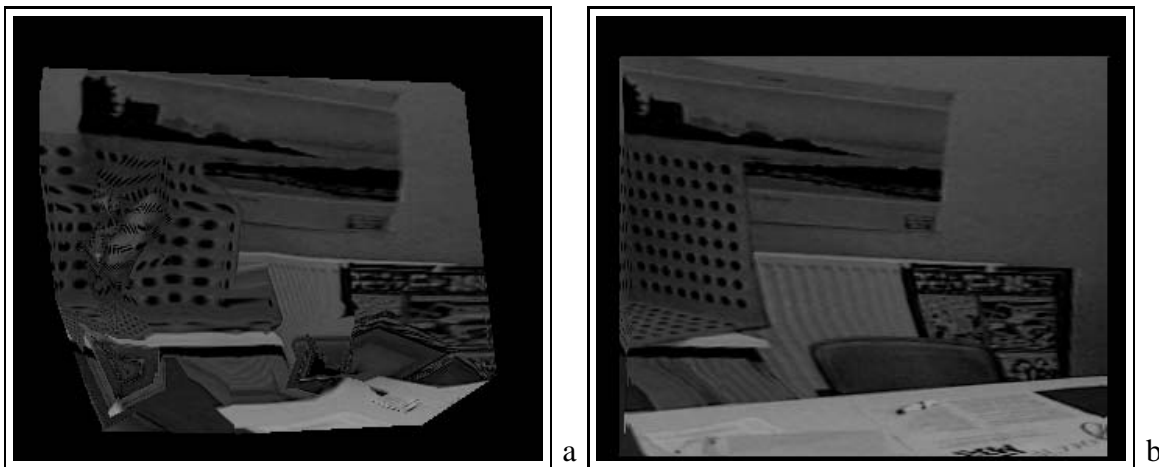


Fig. 17. Two synthesized views from the same view point: model computed directly from automatically extracted and matched points (a), model obtained by our method (b)