3-D Reconstruction of Urban Scenes from Sequences of Images

Olivier Faugeras, Stéphane Laveau, Luc Robert, Gabriella Csurka, Cyril Zeller

INRIA

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

3-D Reconstruction of Urban Scenes from Sequences of Images

Olivier Faugeras, Stéphane Laveau, Luc Robert

and

Gabriella Csurka, Cyril Zeller

N° 2572

Juin 1995

PROGRAMME 4

ISSN 0249-6399

INRIA
SOPHIA ANTIPOLIS

3-D Reconstruction of Urban Scenes from Sequences of Images*

Olivier Faugeras, Stéphane Laveau, Luc Robert
and
Gabriella Csurka, Cyril Zeller**

Programme 4 — Robotique, image et vision
Projet Robotvis

Rapport de recherche n°2572 — Juin 1995 — 24 pages

Abstract: In this paper, we address the problem of the recovery of the Euclidean geometry of a scene from a sequence of images without any prior knowledge either about the parameters of the cameras, or about the motion of the camera(s). We do not require any knowledge of the absolute coordinates of some control points in the scene to achieve this goal. Using various computer vision tools, we establish correspondences between images and recover the epipolar geometry of the set of images, from which we show how to compute the complete set of perspective projection matrices for each camera position. These being known, we proceed to reconstruct the scene. This reconstruction is defined up to an unknown projective transformation (i.e. is parameterized with 15 arbitrary parameters). Next we show how to go from this reconstruction to a more constrained class of reconstructions, defined up to an unknown affine transformation (i.e. parameterized with 12 arbitrary parameters) by exploiting known geometric relations between features in the scene such as parallelism. Finally, we show how to go from this reconstruction to another class, defined up to an unknown similitude (i.e. parameterized with 7 arbitrary parameters). This means that in an Euclidean frame attached to the scene or to one of the cameras, the reconstruction depends only upon one parameter, the global scale. This parameter is easily fixed as soon as one absolute length measurement is known. We see this vision system as a building block, a vision server, of a CAD system that is used by a human to model a scene for such applications as simulation, virtual or augmented reality. We believe that such a system can save a lot of tedious work to the human observer as well as play a leading role in keeping the geometric data base accurate and coherent.

Key-words: projective geometry, reconstruction, CAD, architecture

(Résumé : tsvp)

*Stéphane Laveau is supported by a grant under DRET contract No 91-815/DRET/EAR. This work was also partially funded by the EEC under Esprit project 6448, Viva and Esprit project 8878, Realise.
**Email : {faugeras,laveau,lucr,csurka,zeller}@sophia.inria.fr

Unité de recherche INRIA Sophia-Antipolis
2004 route des Lucioles, BP 93, 06902 SOPHIA-ANTIPOLIS Cedex (France)
Téléphone : (33) 93 65 77 77 – Télécopie : (33) 93 65 77 65

Reconstruction tridimensionnelle de scènes urbaines à partir de séquences d'images

Résumé : Dans cet article, nous nous penchons sur le problème de retrouver la structure géométrique Euclidienne d'une scène à partir d'une séquence d'images, sans information préalable sur les paramètres ou le mouvement des caméras. Pour cela, il ne nous est pas nécessaire de connaitre les coordonnées de points de controle dans la scène. En utilisant diverses techniques de vision par ordinateur, nous établissons des correspondances entre les images, et nous retrouvons la géométrie épipolaire de l'ensemble des images, à partir de laquelle nous calculons les matrices de projection correspondant a toutes les prises de vue. A partir de celles-ci, nous pouvons reconstruire la scène tridimensionnelle. Cette reconstruction est définie à une transformation projective inconnue près, c'est-à-dire qu'elle dépend de 15 paramètres arbitraires. Ensuite, nous montrons comment passer de cette reconstruction à une classe plus restreinte de reconstructions, définies à une transformation affine près (il reste alors 12 paramètres arbitraires) en exploitant la connaissance de relations géométriques entre certains éléments de la scène, comme le parallélisme. Enfin, nous montrons comment se ramener a une classe encore plus restreinte de reconstructions, définies cette fois à une similitude près (7 paramètres arbitraires). Cela signifie que dans un repère Euclidien attaché à la scène ou à l'une des caméras, la reconstruction ne dépend que d'un paramètre: l'échelle globale. Pour fixer ce paramètre, il suffit de connaitre une mesure de distance dans la scène. Ce système de vision est concu comme un module (serveur de vision) d'un système de CAO utilisé par un humain afin de créer des modèles de scènes en vue d'applications comme la simulation, ou la réalité virtuelle ou augmentée. Nous croyons qu'un tel système peut permettre d'économiser un temps de travail considérable dans la construction d'une base de donnée tridimensionnelle, tout en garantissant son exactitude.

Mots-clé : géométrie projective, reconstruction, CAO, architecture

1 Introduction

The problem which is tackled in this paper and for which we propose a number of partial solutions is the following: we want to reconstruct a three-dimensional model of a static environment viewed by one or several cameras whose motions or relative positions are unknown and whose intrinsic parameters are also unknown and may vary.

The sequence of images that is used can be either a video sequence, or a film, or a number of snapshots taken from usually fairly distinct viewpoints. In the first two cases, which we will denote as the M(ovie)-situation, it is, as explained in section 2.1 possible to use the continuity in time of the images to help simplify the problem. In the third case, which we will denote by the S(napshot)-situation, this is not possible, and we must work a little harder, as also explained in the same section.

Solving this problem at relatively low cost is extremely important for such applications as image synthesis, simulation, virtual and augmented reality where 3-D models are required and are obtained today through lengthy manual interaction with the images. Our techniques build upon the knowledge which has been acquired in computer vision and photogrammetry in the last 20 years or so and can potentially reduce by a significant factor the amount of manual interaction which is currently necessary to get 3-D models of the world in the computer.

The Esprit project 8878, *Realise*, has set up as one of its goals the partial automation of the acquisition of computerized 3-D models of urban scenes from sequences of images taken from the ground or from an aircraft flying at a low altitude. As was mentioned before, no hypotheses are made upon the relative positions of the cameras, their intrinsic parameters, all of them are assumed unknown, or upon the presence in the environment to be modeled of control points with known coordinates in some fixed frame of reference. We nonetheless show in this article that the complete projective, affine, and Euclidean geometry (up to a global scale factor) of the scene can be accurately captured by a combination of techniques which have been developed over the years in computer vision and photogrammetry. We would like to emphasize the fact that these techniques encompass a wide range of traditionally distinct subjects such as feature detection (edges, corners, junctions) using non-parametric (image-based) and parametric (snake-like) models, tracking of image features in a sequence of images, geometric modeling of image correspondences at the projective, affine, and Euclidean levels with a clear distinction between those levels and the amount of information they require, robust estimation of algebraic instantiation of this geometry (i.e. the perspective projection matrices).

This is only one aspect of the *Realise* project, the geometric aspect; this project also aims at developing a deeper understanding of the photometry and colorimetry of real scenes in order, for example, to improve significantly the quality of the rendering of simulated scenes. This important topic is not covered in this article.

Before we describe in detail the techniques which lead from a set of images to a 3-D model of the scene they represent, we sketch in figure 1 the general flow of information in our method. We start with either a set of N snapshots (S-situation) or with a set of video sequences from which we select N. In the second case, we can use a variety of feature trackers to establish a number of correspondences between the N images, as explained in section 2.1. Note that we do not require that all features be tracked in all sequences, a notoriously non-realistic hypothesis. Also, even though this makes the next process of estimating the perspective projection matrices easier, it is by no means mandatory.

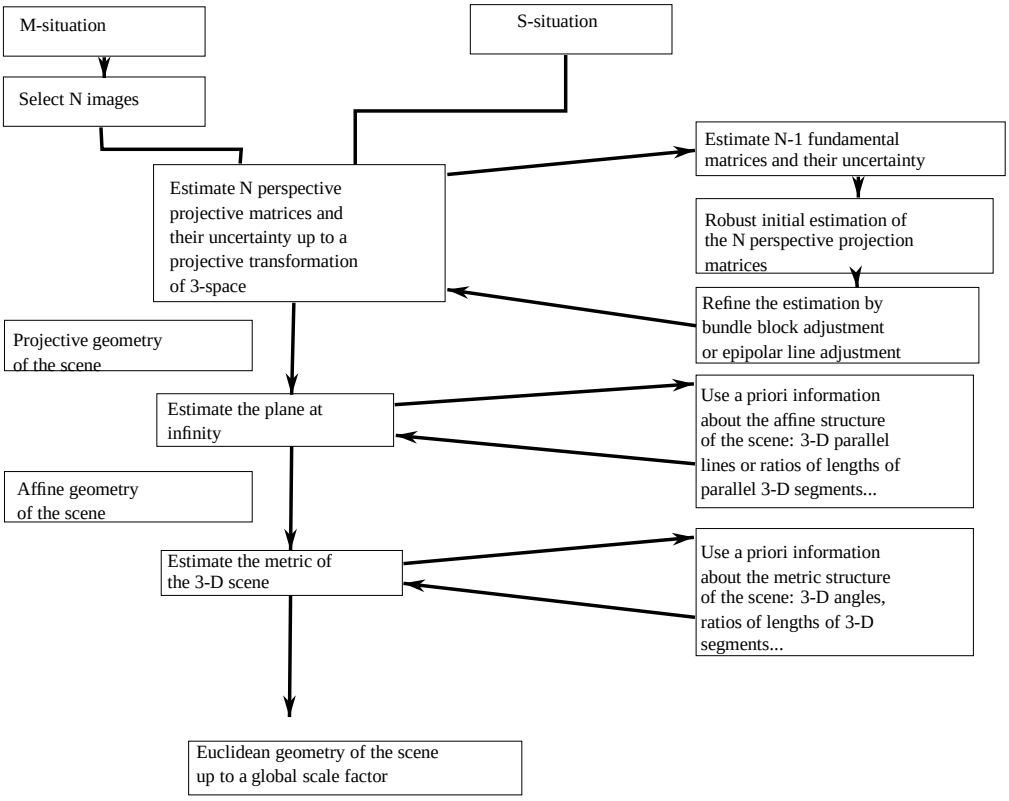

Figure 1: Flowchart of the method described in the paper: this is the backbone of the visual server which is being developed for the Esprit project *Realise*.

The next step is to estimate the perspective projection matrices of each of the N images. We found that this was the best way of representing the geometry of a set of cameras rather than, for example, the fundamental matrices describing the epipolar geometry. At this stage the perspective projection matrices are defined up to an unknown projective transformation, i.e. they can be multiplied on the right by the same arbitrary 4×4 full rank matrix. This reflects the fact that if only image correspondences are available, only the projective geometry of the 3-D scene, i.e. those properties which are invariant under the action of the group of projective transformations, can be recovered. Since projective geometry is now widely used in computer vision and has been in computer graphics since the early days, we refer the interested reader to the corresponding literature [39, 34, 9]. The estimation of the perspective projection matrices is done in three steps:

1. We estimate the $N - 1$ fundamental matrices defining the epipolar geometry between consecutive images,
2. these are then used to obtain a first estimate of the perspective projection matrices,
3. which is then refined using possibly several methods, one of them, being the famous bundle adjustment [5, 6, 16, 17].

Once this estimation has been completed, the 3-D scene can be reconstructed up to an arbitrary projective transformation. To be complete, let us mention that we track the uncertainty at all levels, starting from the pixel level, through the level of the fundamental matrices, the perspective projection matrices, and the reconstructed 3-D points. This is described in detail in section 2.

The next step, if required by the application, is to estimate the affine geometry of the scene, i.e. those properties of the scene which are invariant under the action of 3-D affine transformations. This can be achieved in several ways as described in [10] depending upon whether or not one can control the motion of the sensors. In the application described in this article we only use a priori information about the scene such as parallel lines or known ratios of collinear line segments. This is described in section 3.

The final step, again, if required by the application, is to estimate the Euclidean geometry of the scene, i.e. those properties of the scene which are invariant under the action of 3-D Euclidean transformations, i.e. similitudes. This can again be achieved in several ways as discussed also in [10]. In the application described in this article we use a priori information about the scene such as angles of lines, planes, or known ratios of line segments, to take a few examples. This is also described in section 3.

To summarize, by carefully distinguishing the various geometric levels at which the scene can be reconstructed, i.e. projective, affine, and Euclidean, we are able to determine the minimum amount of information necessary to access a given level from image measurements only. Note that, contrary to the case of the so-called "Self-Calibration" methods in photogrammetry [17], our method does not require the knowledge of the Euclidean coordinates of a small number of control points in the scene. Nonetheless, it does require, in the version presented in this article, some knowledge about the 3-D geometry of the scene such as parallel lines and angles. But we have shown in previous work [33, 28, 12] that even this assumption is not necessary. We use it here because there is such a rich geometric information in images of urban scenes and because the system we are developing is partially interactive.

This last point is extremely important since it is the stepping stone of the philosophy of the design of the *Realise* project. Far from being after a complete automation of the modeling of the scene, the project aims at providing to a human agent working with a CAD system from a set of images of the environment he wants to model, the most advanced tools in computer vision to help him solve such problems as accurate (i.e. subpixel) detection of image features, matching of those features across views, estimation of the geometry of the set of views, computation of the 3-D coordinates of scene points, curves, and surfaces. This framework is depicted in figure 2.

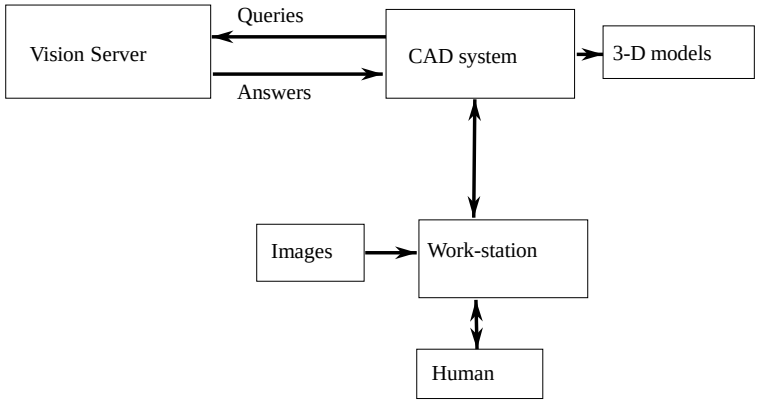

Figure 2: The principle of the interactive recovery of the 3-D geometry of the scene in the Esprit *Realise* Project.

We now briefly review the necessary background material that will be used in the rest of the article.

1.1 Epipolar Geometry

Recently, it has been discovered that the full calibration of the cameras (intrinsic and extrinsic parameters) is not needed to obtain a useful reconstruction of a scene viewed by a stereo system [8, 21]. This theory make use of epipolar geometry which can be retrieved easily from point correspondences in pair of images.

Since these first attempts at an uncalibrated stereovision, a lot of work has been done on the estimation of the epipolar geometry of two images [29, 26, 32, 31, 30, 22, 20, 36, 4]. Robust programs which work automatically are now publicly available. We will consider this problem as solved for the rest of this article; the interested reader is referred to the bibliography.

We will use the *fundamental* matrix representation of the epipolar geometry. In this representation, 2 points in correspondence in images 1 and 2 (expressed in homogeneous coordinates) $\mathbf{m}_1$ and $\mathbf{m}_2$ satisfy the following projective relation: $\mathbf{m}_2^T \mathbf{F}_{12} \mathbf{m}_1 = 0$. The fundamental matrix is defined up to a scale factor and satisfies $\mathbf{F}_{21}\mathbf{e}_{21} = \mathbf{F}_{12}\mathbf{e}_{12} = 0$, with $\mathbf{e}_{ij}$ being the epipole in image i generated by image j (or equivalently, the image in i of the optical center of camera j).

From this, we can conclude that the fundamental matrix depends upon at most seven parameters. It is shown elsewhere [28] that it depends exactly on seven parameters. Therefore, the set of all possible fundamental matrices between N cameras would depend on $7N(N-1)/2$ parameters if there were no constraints between them. However, in our simple perspective model, each camera depends on a fixed number of parameters (we use 6 for the pose and orientation and 5 for the intrinsic or internal parameters). This leads to $O(N)$ parameters for the cameras and since the fundamental matrices are represented by $O(N^2)$ parameters, there exist constraints between them. These constraints can be enumerated [13, 14], but they are difficult to use and we prefer the more compact representation of the projective camera matrices.

1.2 Perspective Projection Matrices

A point $\mathbf{M}$ in space projects to a point $\mathbf{m}$ in the image if and only if $\mathbf{m} = \mathbf{P}\mathbf{M}$ where $\mathbf{P}$ is the 3×4 so-called perspective projection matrix of the camera. All quantities are defined up to an unknown scale factor. It has been shown by [27] that given a set of fundamental matrices satisfying the constraints, one can find corresponding projection matrices. The solution is unique up to an unknown projective transformation in space if the optical centers are not aligned.

The relation of the projection matrices to the fundamental matrices is extremely simple: If we write $\mathbf{P}_i$ as $[\mathbf{M}_i|\mathbf{t}_i]$, the epipoles $\mathbf{e}_{ij}$ satisfy Equation (1) by definition (as images of an optical center).

$$\mathbf{e}_{ij} = \mathbf{t}_i - \mathbf{M}_i\mathbf{M}_j^{-1}\mathbf{t}_j \tag{1}$$

For the fundamental matrices, an elimination scheme leads to

$$\mathbf{F}_{ij} = [\mathbf{e}_{ij}]_\times \mathbf{M}_j\mathbf{M}_i^{-1} \tag{2}$$

where $[\mathbf{e}_{ij}]_\times$ is the 3×3 matrix representing the cross-product with the vector $\mathbf{e}_{ij}$.

It is understood that these equations are projective and therefore are defined only up to an unknown scale factor. We assume that $\mathbf{M}_i$ and $\mathbf{M}_j$ are invertible. If they are not, we can always transform them by a proper choice of a projective transformation to a new frame such that they are invertible and satisfy our assumptions.

1.3 Reconstruction

From the consistent epipolar geometry, we can recover the 3-D scene up to an unknown projective transformation of space. This is not as far from a Euclidean reconstruction as it seems. The set of projection matrices for N cameras depends upon $11N-7$ parameters in the Euclidean case, whereas only $11N-15$ in the projective case.

These 8 additional free parameters are the *low* price to pay for not knowing the internal parameters of the cameras and their relative positions in space. We will later see how these unknown parameters can be recovered using very little information. As stated previously, the epipolar geometry of a set of cameras is best represented by a set of projection matrices that we will in the sequel assume to be defined up to a projective transformation in space.

2 Robust recovery of the geometry

In the M-situation, we select (presently manually but we plan to automate this process in the near future) a subset of the images in the sequence so that we end up in the S-situation. The important difference is that the intermediate images can be used, as explained below, to simplify the process of establishing correspondences between the views.

2.1 Obtaining correspondences between images

The algorithm used to compute the projection matrices needs correspondences and a few epipoles in order to work. We first obtain feature points using very simple corner detectors (we use [19], but other possibilities are [23, 35, 18, 15]) and we refine their position using a model-based approach [3].

In the S-situation, we then establish correspondences between the corners using grey-level correlation between neighboring regions of those feature points. For a given point in one image several candidate matches are in general possible in another image. In order to reduce the number of hypotheses, we make use of relaxation methods [42]. Figure 3(bottom) shows a subset of the correspondences which have been automatically obtained between the images shown on the top row.

In the M-situation we track the feature points in the sequence whenever possible (small motion between frames). If a given point can be tracked all the way between two of the selected views, a correspondence is established. Tracking of a point of interest is performed by predicting its position from one image to the next and searching in a small neighborhood of the predicted point for an actual point. An example of such a tracking is shown in Figure 4. We can of course also use the method of the previous paragraph.

2.2 Estimating the fundamental matrices between pairs of images

At this stage, we have obtained a number of correspondences between some images. Correspondences between pairs of images are input to a program that reliably and robustly estimates the fundamental matrices between those pairs [42]. In particular this program has the capability of rejecting some of the correspondences as outliers.

2.3 Estimating the uncertainty of the fundamental matrices

The uncertainty associated with points of interest (typically between 0.1 and 1 pixel) is propagated to the fundamental matrices. In order to compute an estimate of this uncertainty, we parameterize the fundamental matrix with the minimum number of parameters, namely 7, and compute the covariance matrix of the corresponding vector of size 7. There are several technical difficulties in doing this. First, the parameterization using 7 parameters is nonlinear, not unique, and has singularities. We therefore have to find the best one in the sense that it is the most remote from singularities. Second, the criterion which is minimized in order to estimate the fundamental matrix is also nonlinear and does not provide an analytical expression of the solution as a function of the point correspondences. We therefore have to use the implicit function theorem to actually compute the covariance matrix of

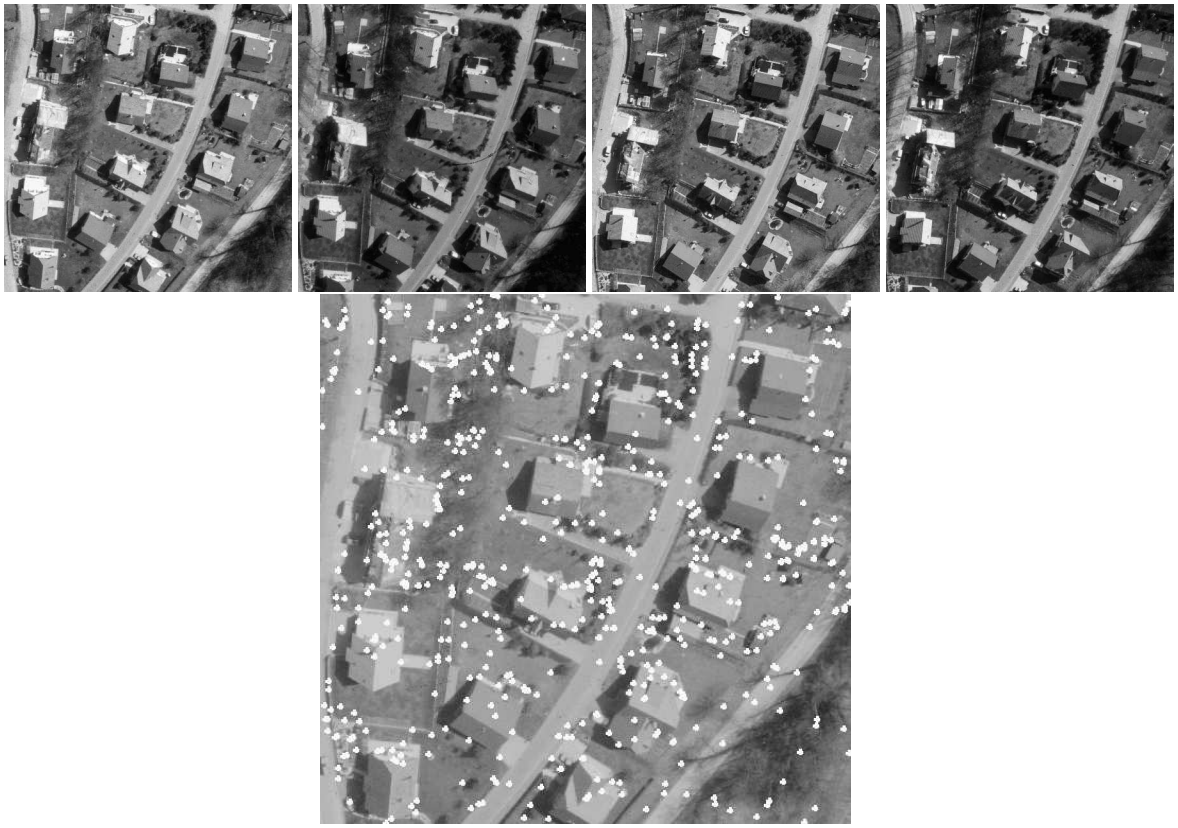

Figure 3: Top: Four images (labeled 5888, 5889, 5897, 5898) of the "Avenches" series provided by ETH, Zurich. Bottom: Points extracted from image 5888 which have been matched by the Image-Matching algorithm with at least one point in another image.

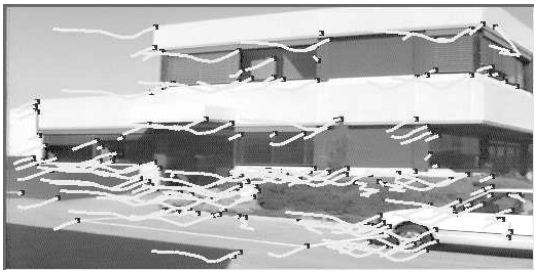

Figure 4: Tracking of points of interest in the library sequence.

the parameter vector of the fundamental matrix. The details of those computations can be found in [7].

2.4 Recovering the geometry of the N cameras

Up to this stage in the processing, we have estimated the fundamental matrices of consecutive pairs of images as well as obtained a number of point correspondences between the views. Nonetheless, we usually still have false matches to account for. The set of false matches for the epipolar geometry between pairs of images is a strict subset of the false matches for the projective geometry: this simply means that the images of a point can satisfy the epipolar geometry between pairs of images and still be incorrect when considering the complete set of images. The geometry estimation algorithm has been designed to deal with this problem.

We have the choice of the projective basis in which we want to express the perspective projection matrices, knowing that the choice of a particular basis will not have any influence on the final results. We are going to use this property to compute our projection matrices. Using the theory developed in [8], from 5 points in correspondence in a pair of images and the epipoles, we can obtain the projection matrices, expressed in the projective basis defined by those points. This step is just a matter of writing equations of the type:

$$\mathbf{m}_i = \mathbf{P}_j.\mathbf{E}_i, i \in \{1, \ldots, 5\}, j \in \{1, 2\} \tag{3}$$

where the $\mathbf{E}_i$ represent the canonic 3-D projective basis: $[0,0,0,1]^T$, $[0,0,1,0]^T$, $[0,1,0,0]^T$, $[1,0,0,0]^T$, $[1,1,1,1]^T$. The 20 scalar equations given by (3) need to the be completed by two equations exploiting the fact that the epipoles are known.

In order to obtain a set of projection matrices, we first choose 5 points in correspondence in the N images. These points are usually chosen in the scene. We then proceed pairwise. For each consecutive pair of images, we compute the projection matrices in the basis of the 5 chosen points. For this, we make use of the coordinates of the points in the images and of the coordinates of the epipoles that we determined previously. Of course, there can be conflicts: the projection matrix $\mathbf{P}_j$ computed from the pair $(j-1, j)$ can be different form the one computed with $(j, j+1)$[1]. We do not consider this as a major problem because there are usually not very different and because this initial estimate is just a starting point for a refinement procedure. Only one of the possible projection matrices is kept.

Of course, running through this process only once has very little chance to succeed because of the possible outliers. If one of the correspondences is erroneous, then the projection matrix and its neighbors will be useless[2]. To overcome this problem, we use robust methods.

2.4.1 Least Median of Squares

The Least Median of Squares (LMedS) is a classic method in outlier detection. A very good introduction can be found in [38]. We need a quality measure of the set of projection matrices for each

[1] This can only be due to the epipoles, because the coordinates of the 5 points remain unchanged.

[2] Note that all other matrices will be correct. This quality of localness is desirable and cannot be achieved with iterative (image after image) techniques

point. We define r_i as the distance between a given point and the reprojection of the reconstructed 3-D point with these projections.

$$r_i = \sum_j d(\mathbf{m}_{ij}, \mathbf{P}_j.\mathbf{M}_i) \tag{4}$$

$\mathbf{M}_i$ is obtained with the reconstruction algorithms mentioned in section 3. The LMedS method estimates the parameters by solving the non-linear minimization problem:

$$\min(\text{med}_i(r_i^2)) \tag{5}$$

That is, the estimator must yield the smallest value for the median of squared residuals computed for the complete set of points. Of course, it is not reasonable to generate all the possible subsets of 5 point correspondences. Rather, we use a Monte-Carlo technique to draw m random subsamples of $p = 5$ different point correspondences. For each subsample J, we estimate the set of projection matrices $\mathbf{P}_j^J$ by the methods previously described. For each set $\mathbf{P}_j^J$, we can determine the median of the squared residuals denoted $M^J = \text{med}_i(r_i^2)$, with respect to the whole set of point correspondences. We retain the estimate of the $\mathbf{P}_j$ leading to the minimal M^J.

The question now is: *how do we determine m?* A subsample is considered good if it consists of p good correspondences across the N images. Assuming that the probability of a point correspondence across 2 images being an outlier is ϵ, the probability of a point correspondence across the N images being an outlier is $1 - (1 - \epsilon)^{N-1}$. The probability that at least one of the m subsamples is good is given by

$$P = 1 - (1 - (1 - \epsilon)^{(N-1)p})^m \tag{6}$$

In our implementation, we assume that $\epsilon = 15\%$ and require $P = 0.99$, thus $m = 6907$. Note that the algorithm can be sped up by means of parallel computation, because the processing for each subsample is done separately.

The five points of a subsample may be very close to each other. Such a situation should be avoided because the estimation of the 3-D structure from such a projective basis is highly unstable and the result is useless. It is a waste of time to evaluate such a subsample. Bucketing techniques were developed to ensure that such configurations are avoided. The images are evenly divided in buckets and we impose that the points be drawn from different buckets (i.e. different regions of the image). The previous formula determining m still holds under the assumption that the outliers are uniformly distributed over the image.

2.4.2 Block estimation

Over a long sequence, it is very difficult or even impossible to find correspondences for the same five points across the whole sequence of images. We therefore split our estimation process over different consecutive blocks of images, with the precaution that the intersection of two consecutive blocks of images contains at least two images. Knowing the projective bases used in each such pair of blocks, we can compute the projective transformation from one to the other and apply it to the matrices of the second block. This process glues the blocks together.

2.5 Refinement

Once a correct set of projection matrices has been computed, we can refine it by two possible methods. Of course, the outliers found at the previous step are marked as invalid and are not taken into account any further.

2.5.1 Epipolar line adjustment

From the set of projection matrices, we can compute a consistent set of fundamental matrices. The points that we have matched must satisfy the epipolar constraints. We then minimize the sum of the distances between the points and the epipolar lines generated by their correspondents by varying the projection matrices. However, this method is slow because we have to recompute the epipolar lines at each step. In other words, there is no possible decoupling of the minimization because of its high non-linearity.

2.5.2 Bundle adjustment

This classical method in photogrammetry [5, 6, 16, 40, 17] is very well suited to our problem. With our initial estimation, the optical rays used in the method approximately intersect because the reprojections of the 3-D point are close to the initial points. This method has the advantage of being fast. The only modification that we have made is that instead of reconstructing the actual Euclidean 3-D points, we reconstruct the points in a projective basis. Although our problem is over-parameterized (we allow the projective basis in space to change), the minimization converges because of the nice properties of the Levenberg-Marquardt algorithm. The average distance between one point and the reprojection of its reconstruction is initially around 1 pixel and typically goes down to 0.3 pixels.

The epipolar line adjustment is slower than the bundle adjustment but performs best in the bad cases when the set of projection matrices is not very well initialized.

Figure 5 shows the epipolar geometry obtained with the calibration data on the aerial images provided by ETH, the epipolar geometry obtained by our algorithm without any prior knowledge and a close-up. We worked with images of reduced size (1000×1000 pixels). The computation took 6.3 minutes for 10000 projective bases tried. The average image error was less than 0.3 pixels on the 183 points used for the refinement.

3 Reconstruction

As mentioned in section 2, the projection matrices that we have determined allow us to compute three-dimensional structure from image correspondences, *up to an unknown projective transformation*. This fact had been first stated in [24] in the case of two affine cameras (in this case, the unknown transformation in space is affine). The case of two projective cameras has been then presented in [8, 21].

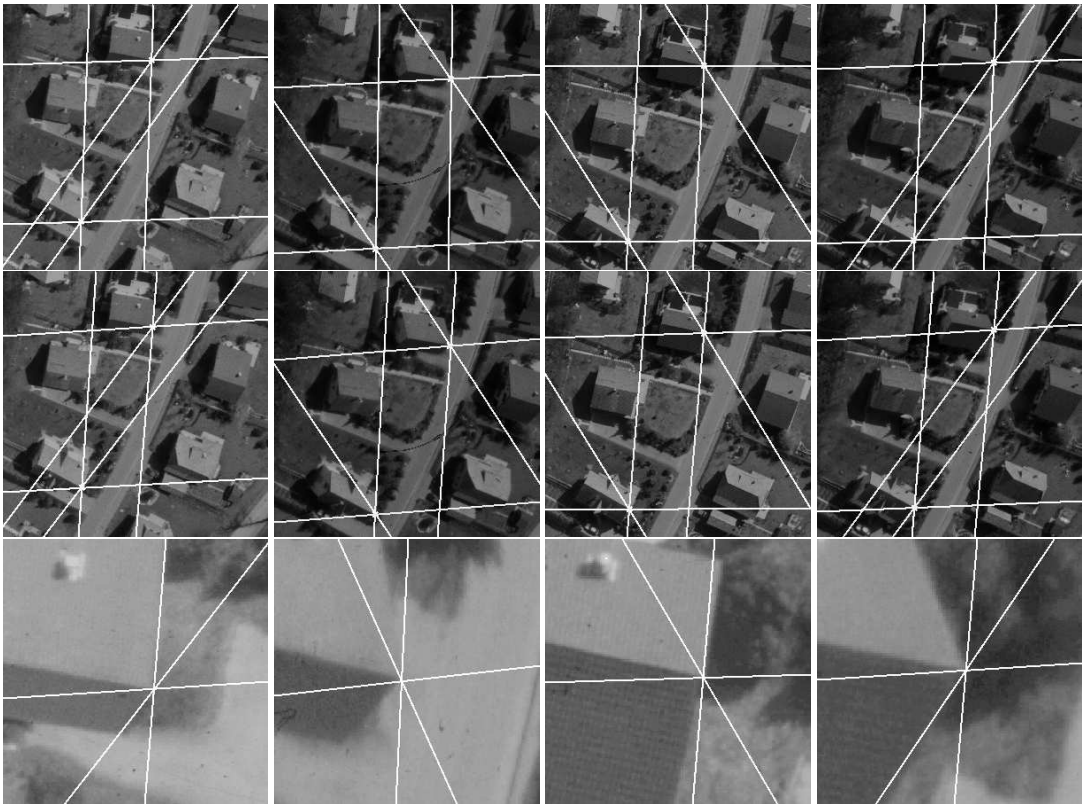

Figure 5: The top row shows the epipolar geometry obtained with the calibration data provided by ETH, the middle row shows the epipolar geometry obtained by the method described in the article without any prior knowledge and the bottom row demonstrates on four subimages the high quality of the estimated epipolar geometry.

From a practical standpoint, this means that from image correspondences, we can compute three-dimensional points represented by their homogeneous coordinates (4-vectors). Knowing the coordinates of a point, we can back-project it onto any of the cameras for which a projection matrix has been computed. We can even project it onto an arbitrary virtual camera: this way we can produce new views of the scene. This process, usually called view transfer [2, 41], has been used for image synthesis[11, 25].

However, for simple reasons, the projective reconstruction may not be sufficient: for instance within *Realise*, the generated building models are used for realistic rendering and virtual walk-through, requiring fast-rendering hardware which can only handle Euclidean descriptions.

Without any additional information, recovering Euclidean structure is impossible: all the geometric relations induced by point correspondences have been already used. We need to use additional information, either on the viewing system, or on the scene.

Approaches have been developed which deal with the former case [33, 28, 12]: assuming that the intrinsic parameters of the camera do not vary across at least three views, they can be computed from a number of point correspondences in the three views. Though good results have been obtained, this approach tends to be very sensitive to noise.

In the latter case, we assume that some information is known about the scene:

- If we know the coordinates of at least five reconstructed points in general configuration (i.e., a projective basis) with respect to a Euclidean frame, we can compute the projective transformation which changes projective coordinates into Euclidean ones. This principle of using a few "anchor points" to derive Euclidean coordinates is also used in self-calibration [17]. It supposes that one has performed manual measurements on the real scene.

- A Euclidean frame can be characterized as a frame where parallel lines intersect at infinity, and where orthogonal lines are indeed orthogonal (the dot-product of their directions is zero). The first property characterizes affine structure, whereas the second one characterizes Euclidean structure up to an unknown scale. As shown below, using images of parallel lines we can recover affine structure; Using pairs of orthogonal direction, we then reach scaled Euclidean structure. Less restrictive than the above approach (here, no manual measurement is performed), this approach is perfectly suited to the requirements of *Realise*, because there are in general many images of parallel or orthogonal lines in views of buildings.

Let us now see in more detail the various stages of the process. In the first stage, we recover affine structure, in which parallelism is preserved. In the second one, we recover Euclidean structure in which orthogonality is preserved as well.

After the first stage described in the previous section, the world is modeled as a three-dimensional projective space $\mathcal{P}^3$. We use the standard embedding of a three-dimensional affine space $\mathcal{A}^3$ into $\mathcal{P}^3$ obtained by identifying $\mathcal{A}^3$ with $\mathcal{P}^3 \setminus \Pi_\infty$, where Π_∞ is a plane, called the plane at infinity and which can be thought as the set of directions of lines in $\mathcal{A}^3$. In particular, two parallel lines of $\mathcal{A}^3$, seen as lines of $\mathcal{P}^3$ intersect at a point of Π_∞ (called their point at infinity). Such a point is not as mysterious as it sounds since when viewed by a camera, the images of the two lines usually intersect at a point (called their vanishing point) which can be thought of as (in fact in some sense *is*) the image of the point at infinity of the two lines. Hence, in order to determine the plane at infinity, it is in principle

sufficient to have in the scene three pairs of non coplanar parallel lines. Once the plane at infinity has been determined, an affine coordinate system can be chosen and affine coordinates of the points in the scene computed.

In the remainder of this article, we use the standard embedding of $\mathcal{A}^3$ into $\mathcal{P}^3$ which maps a point of affine coordinates $[x, y, z]^T$ onto its corresponding point of $\mathcal{P}^3 \setminus \Pi_\infty$ of homogeneous coordinates $[x, y, z, 1]^T$. This embedding simply means that the plane at infinity is the plane of equation $T = 0$ in the projective space with homogeneous coordinates X, Y, Z, T.

3.1 Parallel lines: affine structure

Two lines in space are parallel if and only if they intersect each other in the plane at infinity. A projective transformation preserves affine structure if and only if preserves parallelism, which means that it leaves the plane at infinity (the set of all points at infinity) globally invariant.

Thus, the problem of recovering affine structure is equivalent to finding a projective transformation H_a which maps the plane at infinity onto the plane represented by $[0, 0, 0, 1]^T$. This is a very simple operation provided that we can compute the coordinates of the plane at infinity in the initial projective frame. For this purpose, we first need to determine at least three non-aligned points on this plane, i.e. three non coplanar directions of lines. Since we observe images of lines, each of these points is computed as the vanishing points point of a set of parallel lines observed in the images. This is shown in Figure 6.

In this figure the three pairs of lines $(L_1,\ L_2)$, $(M_1,\ M_2)$, and $(N_1,\ N_2)$ are respectively parallel and the images of their points at infinity $V_L,\ V_M,\ V_N$ are the points of intersection $v_l,\ v_m,\ v_m$ of the pairs of image lines (l_1, l_2), (m_1, m_2), (n_1, n_2), respectively.

From a practical standpoint, parallel directions are defined manually in the images. For each image, we compute a polygonal approximation of the edge chains extracted with a sub-pixel feature detector. Line segments representing parallel lines are selected in the different images. Each group of segments representing parallel lines in space allows computing one point at infinity. In Figure 7 we show the line segments in image 5888 with which points at infinity have been computed.

3.1.1 Computing points at infinity

Let us assume that we measure in the images the projections of parallel space lines $\langle D_i \rangle$. $\langle d_{ij} \rangle$, represented by the three-dimensional homogeneous vector $\mathbf{d}_{ij}$, is the image of $\langle D_i \rangle$ in view j. We want to compute their point of intersection $\mathbf{V} = \bigcap_i \langle D_i \rangle$ which we know to be in Π_∞. The image $\mathbf{v}_j$ of $\mathbf{V}$ in view j is the vanishing point of the image lines $\mathbf{d}_{ij}$. The problem that we need to solve is the following: given $\mathbf{d}_{ij}$, compute $\mathbf{V}$.

Since the image of $\mathbf{V}$ in camera j lies on line $\mathbf{d}_{ij}$, we have:

$$\forall i, j \qquad \mathbf{d}_{ij}^T \mathbf{P_j} \mathbf{V} = 0.$$

This is system of P linear, homogeneous equations in the four homogeneous coordinates of $\mathbf{V}$, where P is the total number of lines observed in the images. We solve it using SVD, obtaining the homogeneous vector $\mathbf{V}$.

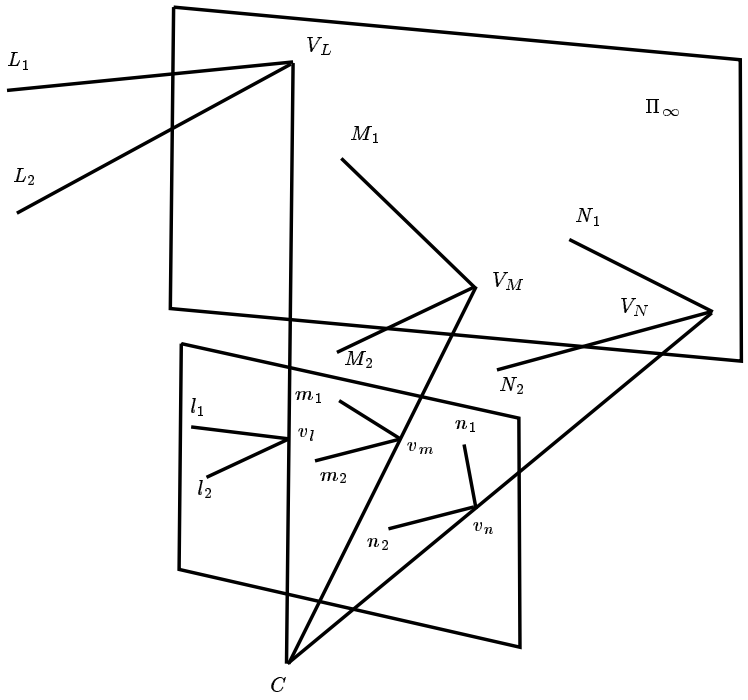

Figure 6: The plane at infinity is determined from the vanishing points of the images of three sets of non-coplanar parallel lines $(L_1,\ L_2)$, $(M_1,\ M_2)$, and $(N_1,\ N_2)$.

Figure 7: Line segments of image 5888 used for the computation of Π_∞.

Remark: Another method for computing $\mathbf{V}$ consists of first computing in each image the vanishing points v_j as solutions of the homogeneous system:

$$\forall i \qquad \mathbf{d}_{ij}^T \mathbf{v}_j = 0$$

Then, $\mathbf{V}$ is obtained using standard reconstruction of the computed vanishing points. This method tends to provide much worse results than the previous one. It is probably because vanishing points are computed independently in all the images, so they are not constrained to satisfy the epipolar constraint before they are reconstructed. In the previous method, they are constrained to be the image of one single point $\mathbf{V}$, so they necessarily satisfy the epipolar constraint.

3.1.2 Computing any point on the plane at infinity

The previous process can be applied to all the directions for which parallel lines are observed, yielding points at infinity $\mathbf{V}_k$. Provided that there are at least three non-aligned points at infinity, we can compute the plane at infinity Π_∞, represented by the four-dimensional homogeneous vector $\mathbf{\Pi}_\infty$, as the non-zero solution of the linear homogeneous system:

$$\forall k \qquad \mathbf{V}_k^T \mathbf{\Pi}_\infty = 0$$

Once we know $\mathbf{\Pi}_\infty$, we can compute the point at infinity of any line as long as this line can be reconstructed in space, by computing the intersection of this line with the plane at infinity.

3.1.3 Deriving an affine reconstruction

To define the transformation which maps the plane at infinity onto $[0,0,0,1]^T$, we proceed as follows:

First, we compute a projective reconstruction of the scene, using standard multi-camera reconstruction based on SVD [37].

One reconstructed point of the scene, denoted by $\mathbf{C}$, is chosen as the origin: in the new frame, it has coordinates $[0,0,0,1]^T$. Then, three arbitrary independent directions are selected as coordinate axes, and their points at infinity $\mathbf{V}_X, \mathbf{V}_Y, \mathbf{V}_Z$ are computed using the method described above. They are respectively mapped onto $[1,0,0,0]^T$, $[0,1,0,0]^T$, $[0,0,1,0]^T$. To define a projective transformation in space, we need a fifth point mapping. We select another reconstructed point $\mathbf{S}$, which does not lie on any of the three planes defined by the origin and two of the three axes. In the new frame, this point is assigned arbitrary non-zero coordinates $[\alpha, \beta, \gamma, 1]$.

$\mathbf{H}_a$ is then computed as the projective transformation which maps the initial projective basis $\mathbf{V}_X, \mathbf{V}_Y, \mathbf{V}_Z, \mathbf{C}, \mathbf{S}$ onto the final one. Since $\mathbf{V}_X, \mathbf{V}_Y, \mathbf{V}_Z$ are all mapped onto points whose fourth component is zero, any point of the plane at infinity will also be mapped onto a point whose fourth component is zero, which is precisely what we needed for the reconstruction to be affine. Figure 8 shows an example of affine reconstruction. Two of the three directions chosen as coordinate axes form a very small angle (left). This implies a strong affine skew effect on the reconstructed scene (right).

Parameters α, β, γ have a very simple meaning: they represent scale factors along the three coordinate axes. For instance, if α is multiplied by a non-zero factor, then the reconstructed scene will be stretched by the same scale factor along the $\mathbf{V}_X$ axis. This is visible in Figure 9, which displays two affine reconstructions which differ by only one scale factor (along the direction of the top edge of the roof).

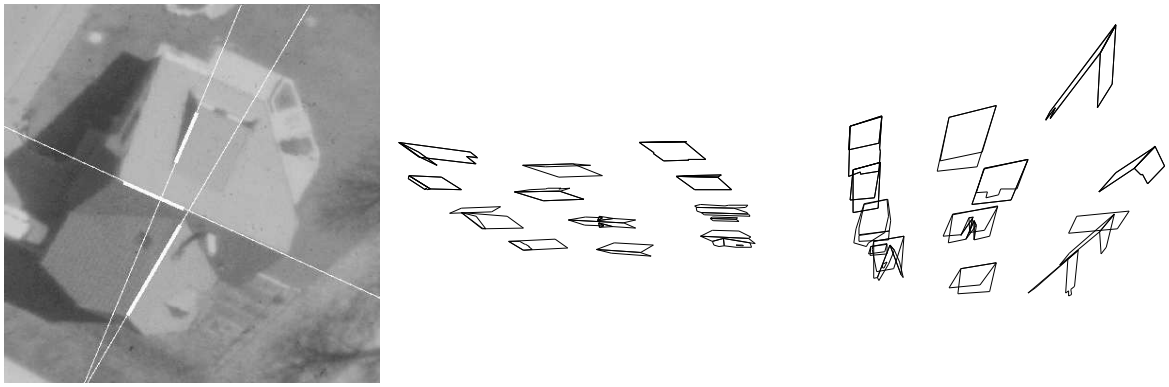

Figure 8: Line segments and directions used for defining the coordinate axes (left), top-view (middle) and side-view (right) of the affine reconstruction (see text).

3.2 Euclidean structure up to three scale factors

As we have seen in Figure 8, the choice of non-orthogonal reference directions may cause severe affine distortion of the reconstructed scene. A first step toward the recovery of Euclidean structure is to use three pairwise orthogonal directions.

In this case, the directions of edges parallel to the reference directions are preserved. In fact, the recovered structure is equivalent to Euclidean structure scaled with three scale factors along the three coordinate axes. As a consequence, two edges aligned with two orthogonal coordinate axes remain orthogonal in the final affine reconstruction, for any value of the scale parameters α, β, γ (e.g. previous section). This is for instance the case of the roof on which the two horizontal directions have been defined (left). The relative values of the scale factors used for the two displayed affine reconstructions (middle,right) are very different. This modifies drastically the aspect of the reconstructed roof (at the bottom-right in each view), but the principal directions remain orthogonal in both reconstructions.

Of course, angles between lines which are not aligned with the coordinate axes are not preserved. In particular, orthogonality is not preserved for such directions (see the roof on the bottom-left). We will now see how this property can be used for recovering Euclidean structure up to one global scale factor.

3.3 Euclidean structure up to one scale factor

We now assume that some pairs of orthogonal lines are known a priori. The points at infinity of these lines, $\mathbf{V}_i$, are computed as described above. In a Euclidean frame, lines i, j are orthogonal if and only if $\mathbf{V}_i^T \mathbf{V}_j = 0$.

If we consider the orthogonal frame defined in the previous paragraph, finding Euclidean structure is equivalent to finding relative values of the scale parameters α, β, γ for which the dot-products $\mathbf{V}_i^T \mathbf{V}_j$ are zero for all pairs (i, j) of orthogonal lines.

If the three reference axes used for affine reconstruction are not orthogonal, three additional parameters ("skew" parameters) are introduced which account for the non-orthogonality of the reference affine frame. More precisely, instead of using the mapping defined in 3.1.3, we respectively map $\mathbf{V}_X, \mathbf{V}_Y, \mathbf{V}_Z$ onto $[1, 0, 0, 0]^T$ (this has not changed), $[\lambda, 1, 0, 0]^T$, $[\mu, \nu, 1, 0]^T$.

We end up with the following criterion to be minimized over the scale parameters and the skew parameters: :

$$E(\alpha, \beta, \gamma, \lambda, \mu, \nu) = \sum_{i,j \text{ orthogonal}} (\mathbf{V}_i^T \mathbf{V}_j)^2$$

The global scale of the scene cannot be recovered. So, we search for the particular solution for which $\alpha = 1$. Minimizing $E(1, \beta, \gamma, \lambda, \mu, \nu)$ with the standard Levenberg-Marquardt iterative technique (the initial values of the five parameters are the ones used for affine reconstruction), we end up with a Euclidean reconstruction up to a global scale factor (see Figure 10).

This way we have computed a Euclidean reconstruction of the scene without any knowledge of the camera parameters, nor of the scene coordinates. Only information about point and line matches, parallel and orthogonal relations have been used.

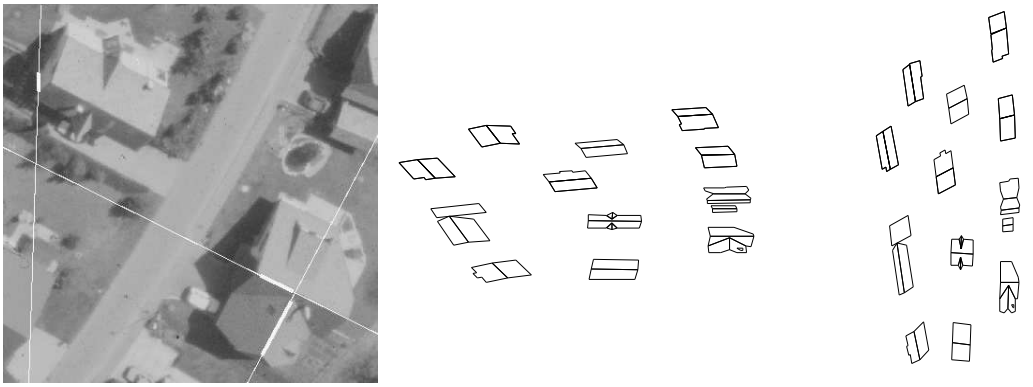

Figure 9: Line segments and directions used for defining the coordinate axes (left); two top-views (left) and (right) of the scene reconstructed with two different values of the scale-factor along one of the horizontal axes (see text).

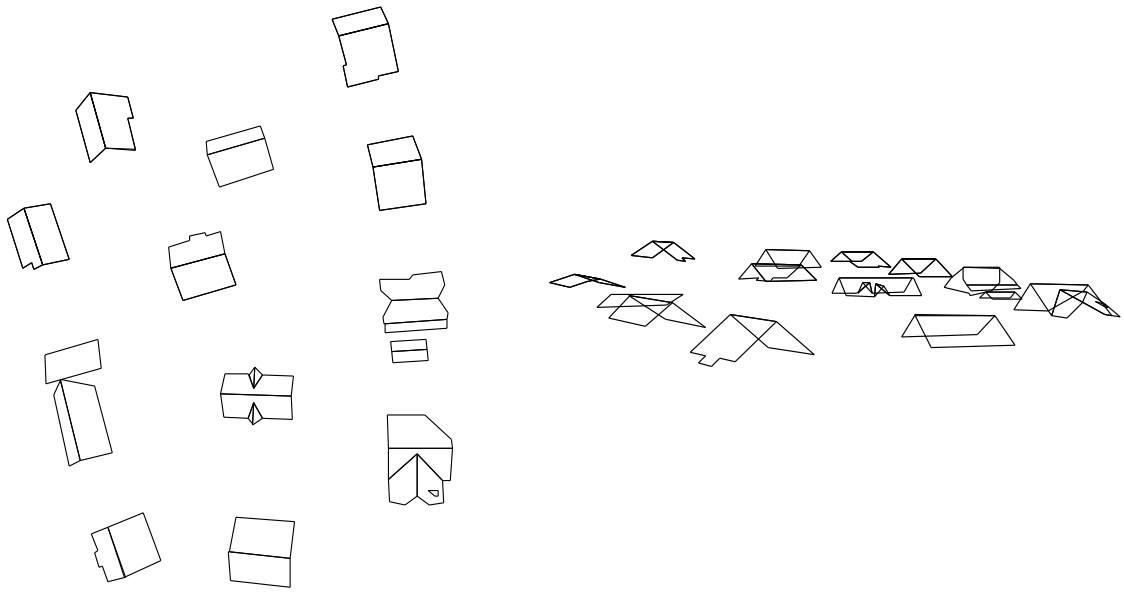

Figure 10: Top-view (left) and side-view (right) of the scene reconstructed after automatic adjustment of the scale factors in order to preserve orthogonality. The frame of reference is the same as in Figure 9.

4 Conclusion

We have described in this article the skeleton of a system based on computer vision that is going to be used to partially automate the 3-D modeling of urban scenes. The system can use any number of cameras and images of the scenes to be modeled and proceeds to estimate automatically the perspective projection matrices corresponding to all the images by matching image features such as corners, junctions, lines. The resulting matrices do not allow to recover a metric model of the scene since no metric information has been used so far, only a projective one which can be used for some applications. In order to go further, the system can use information provided by the user about the actual affine or Euclidean structure of the scene, such as parallel lines, ratios of lengths, and angles. This information allows the system to specialize its representation of the environment from a projective one to an affine one and finally a Euclidean one. The whole system uses sophisticated computer vision tools and has been developed as a flexible server, a vision server, that can be queried by a human user who is using a CAD system to develop a 3-D model of the scene.

One of the advantages of this system is that it does not require any prior knowledge about the cameras, which is handy in applications like video-based modeling for example. The user is then allowed to use his camera the way he likes, without any special set-up.

We think that this concept of an interactive approach in which the user can have at its fingertips the most sophisticated tools in computer vision and photogrammetry can save a lot of tedious work that is presently done by the human modeler.

Acknowledgement
The authors would like to acknowledge the fact that the work presented here would not have been possible without the contributions of Thierry Blaska, Rachid Deriche, Cyrille Gauclin, Charlie Rothwell, and Zhengyou Zhang.

References

[1] Boston, Ma, November 1995. IEEE Computer Society Press.

[2] Eamon B. Barrett, Michael H. Brill, Nils N. Haag, and Paul M. Payton. Invariant Linear Methods in Photogrammetry and Model-Matching. In Joseph L. Mundy and Andrew Zimmerman, editors, *Geometric Invariance in Computer Vision*, chapter 14. MIT Press, 1992.

[3] Thierry Blaszka and Rachid Deriche. Recovering and characterizing image features using an efficient model based approach. Technical Report 2422, INRIA, November 1994.

[4] Boubakeur Boufama and Roger Mohr. Epipole and fundamental matrix estimation using the virtual parallax property. In *Proceedings of the 5th Proc. International Conference on Computer Vision* [1]. To appear.

[5] Duane C. Brown. A solution to the general problem of multiple station analytical stereotriangulation. Technical Report 43, RCA Data Reduction Technical Report, Patrick Air Force base, Florida, 1958.

[6] Duane C. Brown. Close-Range Camera Calibration. *Photogrammetric Engineering*, 37(8):855–866, 1971.

[7] Gabriella Csurka, Cyril Zeller, Zhengyou Zhang, and Olivier Faugeras. Characterizing the uncertainty of the fundamental matrix. Technical report, INRIA, 1995.

[8] Olivier Faugeras. What can be seen in three dimensions with an uncalibrated stereo rig. In Giulio Sandini, editor, *Proceedings of the 2nd European Conference on Computer Vision*, volume 588 of *Lecture Notes in Computer Science*, pages 563–578, Santa Margherita Ligure, Italy, May 1992. Springer-Verlag.

[9] Olivier Faugeras. *Three-Dimensional Computer Vision: a Geometric Viewpoint.* The MIT Press, 1993.

[10] Olivier Faugeras. Stratification of 3-d vision: projective, affine, and metric representations. *Journal of the Optical Society of America A*, 12(3):465–484, March 1995.

[11] Olivier Faugeras and Stéphane Laveau. Representing three-dimensional data as a collection of images and fundamental matrices for image synthesis. In *Proceedings of the International Conference on Pattern Recognition*, pages 689–691, Jerusalem, Israel, October 1994. Computer Society Press.

[12] Olivier Faugeras, Tuan Luong, and Steven Maybank. Camera self-calibration: theory and experiments. In Giulio Sandini, editor, *Proc. Second European Conference on Computer Vision*, number 588 in Lecture Notes in Computer Science, pages 321–334, Santa-Margherita, Italy, May 1992. Springer-Verlag.

[13] Olivier Faugeras and Bernard Mourrain. On the geometry and algebra of the point and line correspondences between n images. In *Proceedings of the 5th Proc. International Conference on Computer Vision* [1]. To appear.

[14] Olivier Faugeras and Bernard Mourrain. On the geometry and algebra of the point and line correspondences between n images. Technical report, INRIA, 1995.

[15] M. A. Förstner and E. Gülch. A fast operator for detection and precise location of distinct points, corners and centers of circular features. In *Proceedings of the Intercommission Workshop of the International Society for Photogrammetry and Remote Sensing*, Interlaken, Switzerland, 1987.

[16] Armin Gruen. Accuracy, reliability and statistics in close-range photogrammetry. In *Proceedings of the Symposium of the ISP Commission V*, Stockholm, 1978.

[17] Armin Gruen and Horst A. Beyer. System calibration through self-calibration. In *Proceedings of the Workshop on Calibration and Orientation of Cameras in Computer Vision*, Washington D.C., August 1992.

[18] A. Guiducci. Corner characterization by differential geometry techniques. *Pattern Recognition Letters*, 8:311–318, 1988.

[19] C. Harris and M. Stephens. A combined corner and edge detector. In *Proceedings Alvey Conference*, pages 189–192, 1988.

[20] R.I. Hartley. Euclidean reconstruction from uncalibrated views. In Joseph Mundy and Andrew Zisserman, editors, *Applications of Invariance in Computer Vision*, volume 825 of *Lecture Notes in Computer Science*, pages 237–256, Berlin, 1993. Springer-Verlag.

[21] Richard Hartley, Rajiv Gupta, and Tom Chang. Stereo from uncalibrated cameras. In *Proc. International Conference on Computer Vision and Pattern Recognition*, pages 761–764, Champaign, Illinois, 1992. IEEE.

[22] Richard I. Hartley. Estimation of relative camera positions for uncalibrated cameras. In *European Conference on Computer Vision*, pages 579–587, Santa Margherita Ligure, Italy, 1992. Springer-Verlag.

[23] L. Kitchen and A. Rosenfeld. Gray-level corner detection. *Pattern Recognition Letters*, pages 95–102, 1982.

[24] Jan J. Koenderink and Andrea J. van Doorn. Affine Structure from Motion. *Journal of the Optical Society of America*, A8:377–385, 1991.

[25] Stéphane Laveau and Olivier Faugeras. 3-D scene representation as a collection of images andfundamental matrices. Technical Report 2205, INRIA, 1994.

[26] Q. T. Luong, R. Deriche, O. D. Faugeras, and T. Papadopoulo. On determining the Fundamental matrix: analysis of different methods and experimental results. In *Israelian Conf. on Artificial Intelligence and Computer Vision*, Tel-Aviv, Israel, December 1993. A longer version is INRIA Tech Report RR-1894.

[27] Q.-T. Luong and T. Viéville. Canonic representations for the geometries of multiple projective views. In Jan-Olof Eklundh, editor, *Proceedings of the 3rd European Conference on Computer Vision*, pages 589–599, Vol. I. Springer-Verlag, Lecture Notes in Computer Science 800-801, 1994.

[28] Quang-Tuan Luong. *Matrice Fondamentale et Calibration Visuelle sur l'Environnement-Vers une plus grande autonomie des systèmes robotiques*. PhD thesis, Université de Paris-Sud, Centre d'Orsay, December 1992.

[29] Quang-Tuan Luong, Rachid Deriche, Olivier Faugeras, and Théodore Papadopoulo. On determining the fundamental matrix: Analysis of different methods and experimental results. Technical Report 1894, INRIA, 1993.

[30] Quang-Tuan Luong and Olivier Faugeras. Camera Calibration, Scene Motion and Structure recovery from point correspondences and Fundamental matrices. *The International Journal of Computer Vision*, 1994. Submitted.

[31] Quang-Tuan Luong and Olivier Faugeras. The fundamental matrix: theory, algorithms, and stability analysis. *The International Journal of Computer Vision*, 1994. To appear.

[32] Tuan Luong and Olivier Faugeras. A stability analysis of the fundamental matrix. In J-O. Eklundh, editor, *Proceedings of the 3rd European Conference on Computer Vision*, pages 577–588, Stockholm, Sweden, May 1994. Springer-Verlag.

[33] S. J. Maybank and O. D. Faugeras. A theory of self-calibration of a moving camera. *The International Journal of Computer Vision*, 8(2):123–152, August 1992.

[34] Joseph L. Mundy and Andrew Zisserman, editors. *Geometric Invariance in Computer Vision*. MIT Press, 1992.

[35] J.A. Noble. Finding corners. *Image and Vision Computing*, 6:121–128, May 1988.

[36] S.I. Olsen. Epipolar line estimation. In *Proc. Second European Conference on Computer Vision*, pages 307–311, Santa Margherita Ligure, Italy, May 1992.

[37] C. Rothwell, G. Csurka, and O. Faugeras. A comparison of projective reconstruction methods for pairs of views. In *Proceedings of the 5th Proc. International Conference on Computer Vision* [1]. To appear.

[38] P.J. Rousseeuw and A.M. Leroy. *Robust Regression and Outlier Detection*. John Wiley & Sons, New York, 1987.

[39] J.G. Semple and G.T. Kneebone. *Algebraic Projective Geometry*. Oxford: Clarendon Press, 1952. Reprinted 1979.

[40] C. C. Slama, editor. *Manual of Photogrammetry*. American Society of Photogrammetry, fourth edition, 1980.

[41] Shimon Ullman and Ronen Basri. Recognition by Linear Combinations of Models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(10):992–1006, 1991.

[42] Z. Zhang, R. Deriche, O. Faugeras, and Q.-T. Luong. A robust technique for matching two uncalibrated images through the recovery of the unknown epipolar geometry. Research Report 2273, INRIA Sophia-Antipolis, France, May 1994. submitted to *Artificial Intelligence Journal*.

Unité de recherche INRIA Lorraine, Technopôle de Nancy-Brabois, Campus scientifique,
615 rue du Jardin Botanique, BP 101, 54600 VILLERS LÈS NANCY
Unité de recherche INRIA Rennes, Irisa, Campus universitaire de Beaulieu, 35042 RENNES Cedex
Unité de recherche INRIA Rhône-Alpes, 46 avenue Félix Viallet, 38031 GRENOBLE Cedex 1
Unité de recherche INRIA Rocquencourt, Domaine de Voluceau, Rocquencourt, BP 105, 78153 LE CHESNAY Cedex
Unité de recherche INRIA Sophia-Antipolis, 2004 route des Lucioles, BP 93, 06902 SOPHIA-ANTIPOLIS Cedex

Éditeur
INRIA, Domaine de Voluceau, Rocquencourt, BP 105, 78153 LE CHESNAY Cedex (France)
ISSN 0249-6399