Maximum Likelihood 3D Reconstruction from One or More Images under Geometric Constraints

Etienne Grossmann*and José Santos-Victor
{etienne|jasv}@isr.ist.utl.pt

Abstract

We address the 3D reconstruction of scenes in which some planarity, collinearity, symmetry and other geometric properties are known à-priori. Our main contribution is a reconstruction method that has advantages of both constraint-based and model-based methods.

Like in the former, the reconstructed object needs not be an assemblage of predefined shapes. Like in the latter, the reconstruction is a maximum likelihood estimate and its precision can be estimated. Moreover, we improve on other constraint-based methods by using symmetry and other forms of regularity in the scene, and by working indifferently with one or more images.

A second contribution is a method for parameterising a configuration of 3D points subject to geometric constraints. Using this parameterisation, the maximum likelihood reconstruction is obtained by solving an unconstrained optimisation problem.

Another contribution lies in validating experimentally the assumption under which the maximum likelihood estimator was defined, namely, that the errors in hand-identified 2D points behave approximately like identically distributed independent Gaussian random variables. With this assumption validated, benchmarking is performed on synthetic data and the precision obtained on real-world data is shown. These experiments show that the maximum likelihood estimator is well-behaved and give insight on the precision obtained in real-world situations.

1 Introduction

When reconstructing a 3D scene from observed 2D points, knowing à-priori geometric information facilitates the task [2, 6, 1] and in some cases allows to obtain a reconstruction from a single image [5, 13, 14]. These advantages allow many new applications such as rapid acquisition of virtual models of buildings [6, 5, 13, 4, 14], reconstruction from paintings and precise measurements [5].

We consider the reconstruction of structured scenes from 2D points in one or more images, when a user gives beforehand geometric information about the scene, such as planarity, collinearity, symmetry etc. This problem has been addressed in the past, either

*This work was supported by PRAXIS XXI Grant BD / 19594 / 99

in model-based approaches [12, 6, 17], in which predefined geometric shapes are fitted to image features or in constraint-based [2, 1, 15, 13, 5, 14, 17] methods, in which geometric constraints are imposed on the reconstructed 3D points. The presented method shares aspects of both approaches.

Like in model-based methods, the shape of the reconstruction depends on some parameters. However, we do not parameterise an assemblage of predefined shapes, so that the user has more freedom in choosing how to represent the scene. Because it is independent from the reconstruction process, the presented parameterisation technique could be used in a model-based reconstruction method [6] and increase its flexibility.

Like in constraint-based methods, the geometric information is given in the form of constraints on the 3D points. We improve over other methods by using extended geometric constraints, such as symmetry and other forms of geometric regularity and by using one or more images indifferently. Because we refine by least-squares fitting a reconstruction obtained by an algebraic method[1], there is a superficial resemblance with some previously published single-view methods [13, 14]. However, these authors do not detail how optimisation is done, while we treat this question in detail. By transforming the reconstruction problem into one of unconstrained optimisation, the dimensionality of the problem is reduced, classical optimisation tools can be used and it is moreover possible to estimate the precision of the reconstruction [10].

Determining whether the user gives information that is consistent and sufficient to define a unique reconstruction is an important issue that is not addressed here ([8]) because it is quite distinct from the maximum likelihood method described in the present article.

There are two principal steps in the method, first to express the geometric information as equality constraints on the estimated quantities (Section 3) and then to transform the reconstruction problem into one of unconstrained optimisation. Due to the geometric constraints, a constrained optimisation problem is first obtained. Then, like in [1], we use our differentiable parameterisation of the set of 3D points subject to the constraints (Section 4) and obtain an unconstrained optimisation problem. Unlike in [1], advanced polynomial manipulations are not needed and a result of differential matrix calculus [8] is used instead.

The reconstruction is obtained by least-squares, which is the right thing to do when the errors in the 2D observations are Gaussian, an assumption that is often made, but is rarely verified experimentally. We are only aware of [16, Appendix B] where such a verification is done, in conditions different from ours[2], so that it is important (Section 5.1) to perform an experimental verification. Then, the estimator is benchmarked on synthetic data and the precision obtained on real-world examples is shown. Finally further discussion and conclusions are given in Section 6.

2 Problem statement

We now define formally the problem of reconstruction. We assume that F images are available, in which N 2D observations $\mathbf{x}_1, ..., \mathbf{x}_N$ are the perspective projections of 3D

[1] We do not discuss here this algebraic method whose details have been published elsewhere.

[2] In [16], automatically tracked points are considered, while we consider hand identified-points which are not necessarily tracked and further subject to error due to inexactitude in geometric constraints.

points $\mathbf{X}_1, ..., \mathbf{X}_N$. If $\mathbf{x}_m$ is observed in image number f ($1 \leq f \leq F$), one has [11] :

$$\begin{bmatrix} \mathbf{x}_m \\ 1 \end{bmatrix} = \lambda_m^f K R_f \left(\mathbf{X}_m - \mathbf{T}_f \right) + \begin{bmatrix} \varepsilon_m \\ 0 \end{bmatrix}, \tag{1}$$

where λ_m^f is the "inverse depth" , $\mathbf{T}_f$ is the position of the camera in world coordinates, R_f is the rotation matrix that relates the camera frame to the world frame and K is the matrix of intrinsic parameters (constant in all images). We assume for now that only the focal length is unknown, although more intrinsic parameters could be estimated [3, 14].

The errors in the observations are represented by the ε_m, which are independent, Gaussian random variables with covariance $\text{cov}(\varepsilon_m) = \sigma^2 I_2$, where I_2 is the 2×2 identity matrix and σ is unknown. Other forms of the covariance matrix could be considered, e.g. that returned by an automatic feature tracker, but we assume that the scaled identity models faithfully the error caused by the human operator and the inexactitude in the geometric model. The validity of this assumption is verified experimentally in Section 5.1.

To facilitate later discussion, we define the "observation function" $\mathcal{P}$ that associates to a 3D point $\mathbf{X}$, a rotation matrix R, a camera position $\mathbf{T}$ and camera calibration matrix K, the perspective projection of $\mathbf{X}$,

$$\mathbf{x} = \mathcal{P}\left(\mathbf{X}, R, \mathbf{T}, K\right) = \lambda \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} K R \left(\mathbf{X} - \mathbf{T} \right), \tag{2}$$

For convenience, the parameters $\mathbf{X}_m$, R_f, $\mathbf{T}_f$ and K are grouped in a single entity :

$$\mathcal{X} = \left(\mathbf{X}_1, ...\mathbf{X}_N, R_1, ..., R_F, \mathbf{T}_1, ..., \mathbf{T}_F, K \right), \tag{3}$$

and another function called $\mathcal{P}$ is defined :

$$\mathcal{P}\left(\mathcal{X}\right) = \begin{bmatrix} \mathcal{P}\left(\mathbf{X}_1, R_{f_1}, \mathbf{T}_{f_1}, K\right) \\ \vdots \\ \mathcal{P}\left(\mathbf{X}_N, R_{f_N}, \mathbf{T}_{f_N}, K\right) \end{bmatrix} \in \mathbb{R}^{2N}. \tag{4}$$

where f_m is the number of the image in which $\mathbf{x}_m$ is observed. This function $\mathcal{P}()$ is disambiguated from that in Eq. (2) by the context.

Hence, our goal consists in estimating the $\mathbf{X}_m$, R_f, $\mathbf{T}_f$ and K, based on noisy observations, $\mathbf{x}_m$. In addition, some geometric information provided by the user will be used, as is detailed in the next section.

3 Geometric constraints

We now explain how the geometric information given by the user is transformed into equality constraints on the coordinates of the 3D points in the scene.

Planarity, collinearity and parallelism. If two points $\mathbf{X}_m$ and $\mathbf{X}_n$ are known to belong to a plane with unit normal $\mathbf{v}$, the following equation must hold :

$$\mathbf{v}^\top \left(\mathbf{X}_m - \mathbf{X}_n \right) = 0. \tag{5}$$

Collinearities are represented combining two planarities, by using the property that two points lie on a segment parallel to a 3D direction $\mathbf{v}$, if and only if they belong to two planes that have (non collinear) normals orthogonal to $\mathbf{v}$.

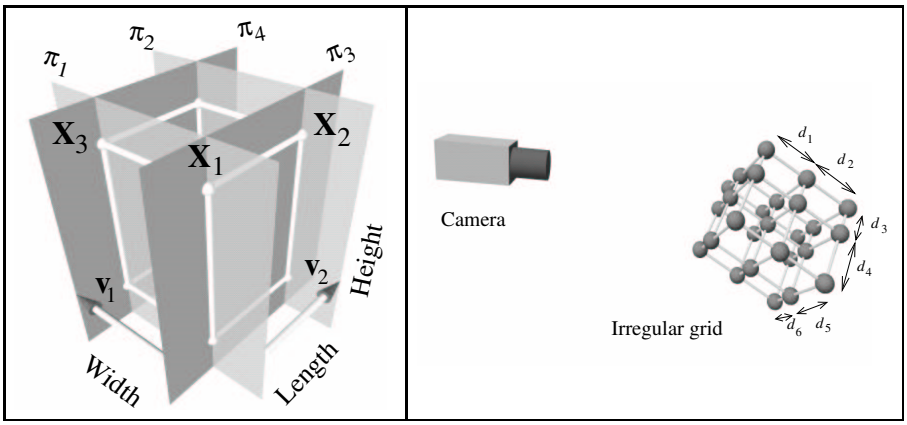

Figure 1: **Left** : The fact that the parallelepiped is as wide as long is expressed by the equation $\mathbf{v}_2^\top (\mathbf{X}_1 - \mathbf{X}_2) = \mathbf{v}_1^\top (\mathbf{X}_1 - \mathbf{X}_3)$. **Right** : Synthetic setup used for benchmarking the estimator.

Ratios of distances between pairs of parallel planes. Figure 1 (left) shows a parallelepiped whose width and length are equal. This equality is expressed, if $\mathbf{v}_1$ and $\mathbf{v}_2$ are the normals of the vertical planes, by :

$$\mathbf{v}_2^\top (\mathbf{X}_1 - \mathbf{X}_2) = \mathbf{v}_1^\top (\mathbf{X}_1 - \mathbf{X}_3) .$$

More generally, the distances between the planes need not be equal, but only have a known ratio $\alpha \in \mathbb{R}$. For arbitrary points $\mathbf{X}_m$, $\mathbf{X}_n$, $\mathbf{X}_p$ and $\mathbf{X}_q$, this type of constraint takes the form :

$$\mathbf{v}_i^\top (\mathbf{X}_m - \mathbf{X}_n) - \alpha \mathbf{v}_j^\top (\mathbf{X}_p - \mathbf{X}_q) = 0. \tag{6}$$

All the known equations resulting from planarities and known ratios of distances are assembled into an expression involving all the 3D points and geometric constraints. The coordinates of the points are joined in a single $3N \times 1$ vector $\mathbf{X} = [\mathbf{X}_1; ...; \mathbf{X}_N]$ and a single matrix equation is obtained

$$B(\mathbf{v}_1, ..., \mathbf{v}_D)\, \mathbf{X} = \mathbb{0}_{M \times 1}, \tag{7}$$

where M is the total number of constraints and the distinct plane normals are $\mathbf{v}_1, ..., \mathbf{v}_D$. The notation $B(\mathbf{v}_1, ..., \mathbf{v}_D)$ emphasises that this matrix depends only on the $\mathbf{v}_i$.

If $U(\mathbf{v}_1, ..., \mathbf{v}_D)$ is a $3N \times P$ matrix whose columns form an orthonormal basis of the nullspace of $B(\mathbf{v}_1, ..., \mathbf{v}_D)$, then all solutions to Eq. (7) can be written

$$\mathbf{X} = U(\mathbf{v}_1, ..., \mathbf{v}_D)\, \mathbf{V}, \tag{8}$$

for some $\mathbf{V} \in \mathbb{R}^P$.

Constraints on the 3D directions Typically, some information on the $\mathbf{v}_i$ is also available, such as known angles and planarities. Usually, three of the directions form a right trihedron, and in the present work, we take $\mathbf{v}_1 = [1, 0, 0]^\top$,$\mathbf{v}_2 = [0, 1, 0]^\top$ and $\mathbf{v}_3 = [0, 0, 1]^\top$. Section 4 explains how constraints on the other $\mathbf{v}_i$ are used within the estimation process.

4 Maximum likelihood estimation

Having presented all the involved geometric entities and the usage of geometric information, we take a probabilistic point of view on the observation process. From the hypothesis of Gaussian i.i.d. noise introduced in Section 2, the negative log-likelihood function takes the familiar form of a sum of squared residues. Finding the maximum likelihood estimate is equivalent to finding the minimum of that function amongst the configurations that verify all the geometric constraints.

From the observation model in Eq. (2), and the notation introduced in Eq. (4), the negative logarithm of the probability of observing $\mathbf{x} = [\mathbf{x}_1, ..., \mathbf{x}_N]$, for a given $\mathcal{X}$ is

$$Q(\mathbf{x}, \mathcal{X}) = \frac{1}{2\sigma^2} \|\mathbf{x} - \mathcal{P}(\mathcal{X})\|^2 + \text{Constant}, \tag{9}$$

where we grouped in the "Constant" term all the terms that do not depend on $\mathcal{X}$. Note that σ needs not be known to locate the $\mathcal{X}$ that minimises $Q(\mathbf{x}, \mathcal{X})$.

Solving the reconstruction problem, for a given $\mathbf{x}$, is equivalent to minimising $Q(\mathbf{x}, \mathcal{X})$, subject to Eq. (7) and to the constraints on the directions $\mathbf{v}_i$. This is thus an equality-constrained optimisation problem in which the feasible set consists of all configurations $\mathcal{X}$ that verify Eq. (7) and the constraints on the directions.

There exist many ways to solve such problems. We choose to transform it into an equivalent unconstrained optimisation problem by parameterising the feasible set. In this approach, a function $\mathcal{X}(\Theta)$ is found such that for all $\Theta \in \mathbb{R}^P$ (for some P), the configuration $\mathcal{X}(\Theta)$ belongs to the feasible set, and such that the feasible set is entirely covered by this mapping. This approach has the advantage that classical tools of unconstrained optimisation can be used and that the new problem has a smaller dimension.

Parameterisation of the estimated quantities One of the main contributions of this paper, with respect to other work, lies in the parameterisation of the collection 3D points. This mapping has parameters $\boldsymbol{\theta}_1, ..., \boldsymbol{\theta}_D$ and $\mathbf{V} \in \mathbb{R}^M$ and is defined by :

$$\begin{array}{rcll} \mathbf{v}_i & = & \mathbf{v}_i(\boldsymbol{\theta}_i) & \text{for all } i \in \{1, ..., D\} \\ \mathbf{X} & = & U(\mathbf{v}_1, ..., \mathbf{v}_D)\,\mathbf{V}. & \end{array} \tag{10}$$

The directions $\mathbf{v}_i(\boldsymbol{\theta}_i)$ act as intermediate quantities, whose parameterisation is defined below. The $3N \times M$ matrix $U(\mathbf{v}_1, ..., \mathbf{v}_D)$ forms an orthonormal basis of the nullspace of the matrix $B(\mathbf{v}_1, ..., \mathbf{v}_D)$ of geometric constraints.

We assume that the $\mathbf{v}_i$ are ordered so that they can be computed sequentially, starting from $\mathbf{v}_1$. Each $\mathbf{v}_i$ is defined by one of the following rules :

Known direction $\mathbf{v}_i$ = Known. No parameter $\boldsymbol{\theta}_i$ is needed, e.g. $\mathbf{v}_1 = [1,\ 0,\ 0]^\top$.

Arbitrary direction $\mathbf{v}_i = \boldsymbol{\theta}_i / \|\boldsymbol{\theta}_i\|$ for some $\boldsymbol{\theta}_i \in \mathbb{R}^3$.

Fixed angle $\mathbf{v}_i$ is constrained to verify $\mathbf{v}_i^\top \mathbf{v}_j = \cos(\alpha)$ for some previously computed $\mathbf{v}_j$ (i.e. $j < i$) and some fixed α. $\mathbf{v}_i$ is computed from a vector $\boldsymbol{\theta}_i \in \mathbb{R}^3$ that is projected orthogonally onto the 3D circle $\left\{\mathbf{v} \mid \mathbf{v}^\top \mathbf{v}_j = \cos(\alpha), \|\mathbf{v}\| = 1\right\}$.

Coplanarity $\mathbf{v}_i$ is coplanar with two direction $\mathbf{v}_j$ and $\mathbf{v}_k$, for some $j < i$, $k < i$. This is implemented in the same way as a fixed angle, by taking $\mathbf{v}_i^\top (\mathbf{v}_j \times \mathbf{v}_k) = \cos(\pi/2)$.

Cross-product $\mathbf{v}_i$ is constrained to be orthogonal to two directions, $\mathbf{v}_j$ and $\mathbf{v}_k$, and is defined e.g. by $\mathbf{v}_i = \mathbf{v}_j \times \mathbf{v}_k / \|\mathbf{v}_j \times \mathbf{v}_k\|$, so that no parameter $\boldsymbol{\theta}_i$ is needed.

Although this parameterisation is less general than arbitrary angle and planarity constraints between directions, it has the advantage of always being feasible (one cannot give incompatible constraints) and of defining explicitly a differentiable parameterisation.

Having defined the mappings $\mathbf{v}_i(\boldsymbol{\theta}_i)$, a function $U(\mathbf{v}_1, .., \mathbf{v}_D)$ is still needed to complete the mapping defined in Eq. (10). We show in [8] that, if $B(\mathbf{v}_1, .., \mathbf{v}_D)$ is a differentiable function such that the rank of B does not vary with the $\mathbf{v}_i$ (this is true for almost all configurations of the $\mathbf{v}_i$), then there exists a differentiable function $U(\mathbf{v}_1, .., \mathbf{v}_D)$ whose columns form an orthonormal basis of the nullspace of $B(\mathbf{v}_1, .., \mathbf{v}_D)$. Because we also show how to compute this function and its differential, we have obtained a complete parameterisation for 3D points subject to geometric constraints.

The other parameters -camera position, orientation and calibration are commonly used in optimisation problems and their parameterisation does not need to be detailed here. The vector of all the parameters is written $\Theta \in \mathbb{R}^P$.

Covariance of the estimator: The reconstruction is thus obtained by minimising a differentiable function $Q(\mathbf{x}, \mathcal{X}(\Theta))$. Having found the minimum $\hat{\Theta}$, it is possible [10] to estimate the covariance σ of the error in the observations and the covariance $\hat{\Theta}$. Because the expression of the covariance can be found in this last publication, we do not include it here, and only indicate that the validity of the resulting expressions was verified by numerical simulations [10, 5, 8].

5 Experimental results

This section starts by validating the Gaussian hypothesis on the errors in the observations. Then, the maximum likelihood estimator is benchmarked with synthetic data and finally the precision obtained in real-world reconstructions is shown.

5.1 Justification of Gaussian assumption

We show that the error of one of the authors clicking on an image is approximately Gaussian by examining the residues from real-world reconstructions.

Because the residues $\mathbf{x} - \mathcal{P}\left(\mathcal{X}\left(\hat{\Theta}\right)\right)$ are not i.i.d. variables, we cannot use these quantities directly. Indeed, the residues have the form :

$$\mathbf{x} - \mathcal{P}\left(\mathcal{X}\left(\hat{\Theta}\right)\right) \simeq \Pi\varepsilon,$$

where Π is the orthogonal projector onto the nullspace of the Jacobian of $\mathcal{P}(\mathcal{X}\left(\hat{\Theta}\right))$. However, we may compute, e.g. using the SVD, a rectangular unitary matrix V whose columns form an orthonormal basis of this space. and define the "normalised residues"

$$\mathbf{y} = V^\top \left(\mathbf{x} - \mathcal{P}\left(\mathcal{X}\left(\hat{\Theta}\right)\right)\right)$$

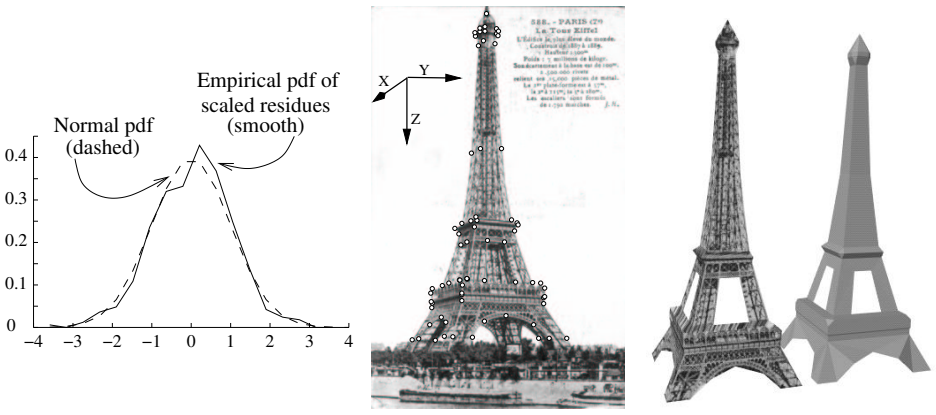

Figure 2: **Left** : Histogram of scaled residues obtained from the real-world reconstructions. **Middle**: Outdoors scene. **Right**: Reconstruction obtained from 70 image points, 56 planes and 17 distance equalities, mostly due to the symmetry around a vertical axis passing through the topmost point.

whose mean and covariance are, if the Gaussian assumption is correct :

$$E(\mathbf{y}) \simeq \mathbb{0} \text{ and } \operatorname{cov}(\mathbf{y}) \simeq \sigma^2 I.$$

If the hypothesis were wrong, the histogram of the normalised residues $\mathbf{y}$ would differ from that of a normal distribution. The hypothesis is validated by comparing the histogram obtained from real-world reconstructions to the probability density function of a normal random variable.

For each setup in Section 5.3, and some other real-world setups, the normalised residues $\mathbf{y}$ were computed and the covariance estimated by $\hat{\sigma}^2 = \|\mathbf{y}\|^2 / N$. The elements of $\hat{\sigma}^{-1}\mathbf{y}$ are thus, if our model is correct, realizations of normal Gaussian random variables. The histogram, obtained by grouping the 392 normalised residues in 18 bins (Figure 2, left), shows that the empirical probability function is approximately normal. The rather good fit between the empirical and assumed probability density function justifies the choice of the Gaussian probability model and the resulting sum-of-squares function Q of Eq. 9. In conclusion, when the author clicks on image features, the error behaves approximately like a Gaussian random variable.

5.2 Error analysis

In this section, the estimator is benchmarked using synthetic data. The setup consists in 27 points disposed on a $3 \times 3 \times 3$ irregular grid (Figure 1, right). The camera orientation is a rotation matrix whose axis is given by a $N(\mathbb{0}, I_3)$ random vector, the norm being the angle of the rotation. The natural logarithm of the focal length is a $N(0, 0.1)$ random variable. The error in the observations is made to vary from 0 (no noise) to 3.12 percent (30dB). Percents (%) or decibels -rather than in pixels- are used so that the figures be independent of the size of the image.

For each error level, the reconstruction algorithm was run 50 times and the mean absolute error in the results measured. Figure 3 shows the curves of the error in the

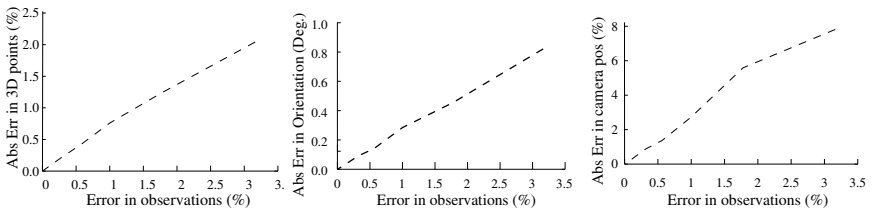

Figure 3: Mean absolute error in the estimated 3D points (**left**), in the orientation of the cameras, measured in degrees (**middle**) and in the position of the cameras (**right**).

Quantity (Unit)	$\mathbf{X}$ (%)	Orient. (deg.)	$\mathbf{T}$ (%)	F. Length (%)	Residue (%)	N	P
Saint-Michel	5.49	0.49	4.19	5.27	0.86	114	180
Tour Eiffel	0.51	0.21	0.64	6.52	0.52	70	58
Conciergerie	2.05	0.26	1.64	1.13	0.62	72	119

Table 1: Theoretical error in the real-world reconstructions. Each row shows the standard deviation of the error in the 3D points, camera orientations, positions and focal length, the relative amplitude of the residues $\mathbf{x} - \mathcal{P}\left(\mathcal{X}\left(\hat{\Theta}\right)\right)$ with respect to $\mathbf{x}$, the number of 3D points, N, and the dimension P of the parameter vector Θ.

estimated 3D points (left), camera orientation (middle) and position (right). The error in the 3D points and camera positions is measured in percents of the mean amplitude of these quantities. The error in the camera orientation is measured by the mean absolute angle, in degrees, between the columns of the true and estimated orientation matrices (R_f, in Eq. (1)). This figure shows that the estimator behaves well when noise becomes very high and gives exact results in the absence of noise. The relatively high error in the location of the camera is in accordance with previous publications [9].

5.3 Real-world examples

Finally, we show some results obtained from real-world scenes. For each dataset, the covariance of the estimator is estimated as in [10]. Table 1 shows the standard deviation of the error in the 3D points, camera orientation, position and focal length, as well as the amplitude of the residues $\mathbf{x} - \mathcal{P}\left(\mathcal{X}\left(\hat{\Theta}\right)\right)$ relatively to that of the observations $\mathbf{x}$.

Tour-Eiffel This dataset illustrates how symmetry relations can be used. Figure 2 (left) shows an image with superposed image features. Horizontal planes are easily identified. Because few vertical planes "connect" points at different heights, symmetry (around a vertical axis passing through the topmost point) is used to obtain the reconstruction (Figure 2 (right), with and without texture). Altogether, there are 70 points, 56 planes and 43 known length ratios and Θ has length 58.

Saint-Michel This example illustrates that single-view reconstruction is not limited to block-like objects. The original image, with superposed 2D points, is shown in Figure 4 (left). Planes are identified using the floor, horizontal rows of bricks, walls and arcades. Altogether, the reconstruction shown in Figure 4 (middle, right) has 114 points and 39

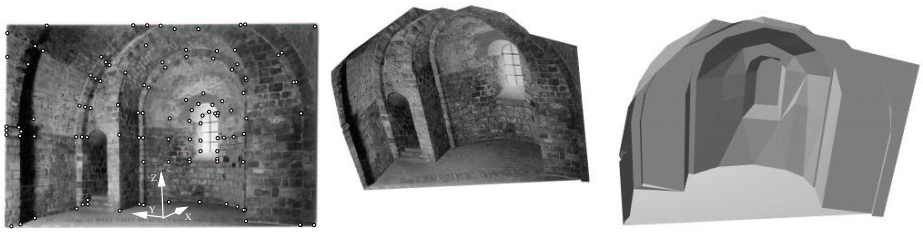

Figure 4: **Left:** Indoor scene with 114 identified points. **Middle** and **Right:** Reconstruction, shown with and without texture, obtained from these points and 39 known planes.

Figure 5: **Left** and **middle** : Two outdoor images. **Right**: Untextured reconstruction

planes. In this model, the principal point of the camera is estimated and Θ has length 180.

Conciergerie This example shows the result of a reconstruction obtained from two views. Figure 5 shows two outdoors image with some overlap and the reconstruction. Seventy-two points (24 in the first image, 48 in the second), 21 planes and two known ratios of length are used and Θ has dimension 119.

All implementations were done with the Octave [7] programming language.

6 Conclusions and future work

The problem of reconstructing a scene from 2D points and à-priori geometric information has been addressed using a method that shares positive aspects of both model-based and constraint-based methods. Like the former, it integrates well in the framework of estimation theory. Like the latter, predefined shapes are not needed. In comparison to other constraint-based methods, greater generality is achieved by using extended geometric constraints and by indifferently using one or more images.

Then, foundation of the method is verified by showing that the errors in hand-identified 2D points is approximately Gaussian and the good behaviour of the method was assessed by benchmarking and real-world experiments.

Further developments can be envisioned, such as using constraints on the camera positions and improving the optimisation method. Also, a statistical test could be used after the optimisation process, like in [16], to check whether the residues are to be Gaussian. Finally, the differentiable parameterisation of 3D points subject to geometric constraints can be used separately, e.g. in a model-based method [17], which would thus gain some flexibility.

References

[1] D. Bondyfalat and S. Bougnoux. Imposing euclidean constraint during self-calibration processes. In *Proc. SMILE Workshop*, pages 224–235, Freiburg, Germany, 1998.

[2] B. Boufama, R. Mohr, and F. Veillon. Euclidean constraints for uncalibrated reconstruction. In *Proc. ICCV*, pages 466–470, Berlin, Germany, May 1993.

[3] B. Caprile and V. Torre. Using vanishing points for camera calibration. *IJCV*, 4:127–140, 1990.

[4] R. Cipolla and E. Boyer. 3D model acquisition from uncalibrated images. In *IAPR Workshop on Machine Vision Applications*, pages 559–568, 1998.

[5] A. Criminisi. *Accurate visual metrology from single and multiple uncalibrated images*. PhD thesis, University of Oxford, 1999.

[6] P.E. Debevec, C.J. Taylor, and J. Malik. Modeling and rendering architecture from photographs : A hybrid geometry- and image-based approach. Technical Report CSD/96-893, UCB, 1996.

[7] J. W. Eaton. Octave—a high-level interactive language for numerical computations. Technical Report CCSR-93-003, U. of Texas. http://www.octave.org/.

[8] Etienne Grossmann. *Maximum Likelihood 3D Reconstruction From One or More Uncalibrated Views Under Geometric Constraints*. PhD thesis, IST, 2002.

[9] Etienne Grossmann and José Santos-Victor. Uncertainty analysis of 3D reconstruction from uncalibrated views. *Image and Vision Computing*, 18:685–696, May 2000.

[10] R.M. Haralick. Propagating covariance in computer vision. In *Proc. Workshop on Performance Characteristics of Vision Algorithms*, pages 1–12, Cambridge, 1996.

[11] R. Hartley and A. Zissermann. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2000.

[12] David G. Lowe. Fitting parameterized three-dimensional models to images. *IEEE Trans. PAMI*, 14(5):441–450, May 1991.

[13] Heung-Yeung Shum, Mei Han, and Richard Szeliski. Interactive construction of 3D models from panoramic mosaics. In *CVPR*, pages 427–433, 1998.

[14] P.F. Sturm and S.J. Maybank. A method for interactive 3D reconstruction of piecewise planar objects from single views. In *Proc. BMVC*, pages 265–274, 1999.

[15] R. Szeliski and P.H.S Torr. Geometrically constrained structure from motion: points on planes. In *Proc. SMILE Workshop*, pages 171–186, Freiburg, Germany, 1998.

[16] P. H. S. Torr, A. Zisserman, and S. Maybank. Robust detection of degenerate configurations for the fundamental matrix. *CVIU*, 71(3):312–333, 1998.

[17] F.A. van den Heuvel. Trends in cad-based photogrammetric measurement. *International Archives of Photogrammetry and Remote Sensing*, 33(5/2):852–863, 2000.