Published in Image Processing On Line on 2019–08–05.
Submitted on 2019–02–01, accepted on 2019–07–11.
ISSN 2105–1232 © 2019 IPOL & the authors CC–BY–NC–SA
This article is available online with supplementary materials,
software, datasets and online demo at
https://doi.org/10.5201/ipol.2019.248

The Orthographic Projection Model for Pose Calibration of Long Focal Images

Laura F. Julià[1], Pascal Monasse[1], Marc Pierrot-Deseilligny[2]

[1] LIGM (UMR 8049), École des Ponts ParisTech, UPE, Marne-la-Vallée, France
({laura.fernandez-julia, pascal.monasse}@enpc.fr)
[2] IGN/LOEMI, Université Paris-Est (marc.pierrot-deseilligny@ensg.eu)

Abstract

Most stereovision and Structure from Motion (SfM) methods rely on the pinhole camera model based on perspective projection. From this hypothesis the fundamental matrix and the epipolar constraints are derived, which are the milestones of pose estimation. In this article we present a method based on the matrix factorization due to Tomasi and Kanade that relies on a simpler camera model, resulting in orthographic projection. This method can be used for the pose estimation of perspective cameras in configurations where other methods fail, in particular, when using cameras with long focal length lenses. We show this projection is an approximation of the pinhole camera model when the camera is far away from the scene. The performance of our implementation of this pose estimation method is compared to that given by the perspective-based methods for several configurations using both synthetic and real data. We show through some examples and experiments that the accuracy achieved and the robustness of this method make it worth considering in any SfM procedure.

Source Code

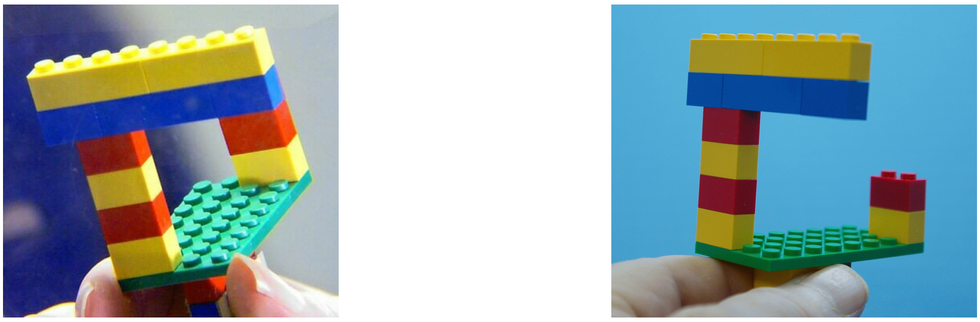
The Matlab implementation of this algorithm is available in the IPOL web page of this article[1]. Usage instructions are included in the README.txt file of the archive. Note that the input data are the image correspondences, so it might be necessary to launch an independent matching algorithm in a first step.

Keywords: pose estimation; calibration; camera model

[1] https://doi.org/10.5201/ipol.2019.248

1 Introduction

The pose estimation task consists in computing the relative motion (rotation and translation) of two or more cameras from a set of matching points in the images obtained from the same scene. This is possible for two views with the epipolar equations: the fundamental matrix can be computed from corresponding image points [6] and the relative rotation and translation can be extracted via singular value decomposition provided the internal calibration of the cameras [4]. The epipolar geometry relies on the pinhole camera model, which describes the generation of photographic images as a perspective projection, but many other geometric projection models exist that are of interest for their applications. For instance, the orthographic model has been used in technical domains for its lack of perspective deformation so that measurements can be taken from the orthographic images, proportional to the 3D object dimensions. Real orthographic images (Figure 1a) can be obtained with a telecentric lens system, a camera recreating the orthographic projection.


(a) orthographic image

(b) perspective image

Figure 1: Two photographs of the same object: one taken with a telecentric lens system (a), creating a visual illusion, and the other (b) taken with a normal lens. Note the lack of perspective deformation on the left, where parallel lines remain parallel and the object depth does not decrease its size, as opposed to the perspective image.

However, there are other cameras that can be modeled after the orthographic projection and our interest resides on seeing the orthographic camera as an approximation of a perspective camera when the scene is far away with respect to its size. It is precisely in such situations, generally generated by the use of long focal lengths, that the standard pose estimation approach based on the perspective model is badly suited and shows great errors in the estimated rotations and translations [16]. For this reason, the use of other projection models that are more robust in this kind of scenes becomes of interest and the orthographic projection has been explored in several works in the literature [14, 12, 8].

The orientation of orthographic views has been discussed as early as 1962 [1] and it is solvable when at least three views are considered. The factorization method by Tomasi and Kanade [14] provides a simple method for pose estimation of orthographic image streams and it can also be used to calibrate perspective images. The other constraint of the method is that it requires full visibility: only points visible in all views can be exploited.

In this article we aim to show the advantages of using the orthographic projection model for pose estimation of perspective images. We investigate the types of scenes where the orthographic model outperforms a perspective approach and test the robustness of the method for degenerate scenes. Moreover, we provide a Matlab implementation and a corresponding web demo that estimates the pose of three views from enough matching points. In Section 2 the definition and equations of the orthographic model are explained, as well as its link with the perspective camera. The factorization method is presented in detail in Section 3 and our implementation of the associated pose estimation method in Section 4. Finally, the results of the experiments on synthetic and real data are discussed in Section 5.

2 The Orthographic Model

The orthographic model consists of a projection of the space points onto a plane along its orthogonal direction, the plane's normal. Rotation, scaling and translation can be applied to the points in the plane after the projection. We can parameterize the projection by the plane's axes, orthonormal vectors $\vec{\imath}$ and $\vec{\jmath}$ that will also characterize the rotation, the origin of coordinates (a, b) in the plane and a scaling factor s. The direction of the projection is $\vec{k} = \vec{\imath} \times \vec{\jmath}$. The orthographic projection can be defined by the linear function ρ_{O}:

$$\begin{aligned} \rho_{\text{O}} : \mathbb{R}^3 &\longrightarrow \mathbb{R}^2 \\ \mathbf{X} &\longmapsto \mathbf{x} = s \begin{pmatrix} \vec{\imath}^\top \\ \vec{\jmath}^\top \end{pmatrix} \mathbf{X} + \begin{pmatrix} a \\ b \end{pmatrix} . \end{aligned} \tag{1}$$

2.1 Pinhole Camera at Infinity

The most commonly used model to explain the geometry of a standard camera is the pinhole camera. In this model the camera lens is omitted and the rays are considered to all go through a tiny hole, a point in space $\mathbf{C}$ that we call camera center. This clearly corresponds to a **perspective projection** of the points in space to a determined plane (the sensor plane) and where the camera center is the center of projection. This projection is usually written in homogeneous coordinates (see [4] for reference) with a linear function given by the projection matrix $P = K \begin{bmatrix} R & \vec{t} \end{bmatrix}$ where K is the calibration matrix, R is the rotation matrix and $\vec{t}$ is the translation vector ($\vec{t} = -R\mathbf{C}$, where $\mathbf{C}$ is the vector of coordinates of the camera center). However, we can also write this perspective projection in Cartesian coordinates with the non-linear function ρ_{P}

$$\begin{aligned} \rho_{\text{P}} : \mathbb{R}^3 &\longrightarrow \mathbb{R}^2 \\ \mathbf{X} &\longmapsto \mathbf{x} = \frac{f}{\vec{k}^\top \mathbf{X} + t^z} \left[\begin{pmatrix} \vec{\imath}^\top \\ \vec{\jmath}^\top \end{pmatrix} \mathbf{X} + \begin{pmatrix} t^x \\ t^y \end{pmatrix} \right] + \begin{pmatrix} c^x \\ c^y \end{pmatrix} , \end{aligned} \tag{2}$$

with $R = \begin{pmatrix} \vec{\imath} & \vec{\jmath} & \vec{k} \end{pmatrix}^\top$, $\vec{t} = (t^x, t^y, t^z)^\top$, f is the focal length (in pixels) and (c^x, c^y) is the principal point.

The perspective projection for pinhole camera becomes an orthographic projection when the camera center is placed at an infinite distance of the scene. To see that, we study the effect of increasing the distance from the camera center to the center of the scene $\mathbf{O}$ (concept defined later) in the direction orthogonal to the image plane, $d := \vec{k}^\top (\mathbf{O} - \mathbf{C}) = \vec{k}^\top O + t^z$. In order to get the camera further away from the scene while maintaining the projection of the scene inside the image, we fix the ratio $\alpha = f/d$, and we put the focal length as a function of the distance to the scene $f = \alpha d$. With this we can compute the limit of the projection $\mathbf{x} = \rho_{\text{P}}(\mathbf{X})$ for $X \in \mathbb{R}^3$ when d approaches infinity

$$\lim_{d \to +\infty} \mathbf{x} = \lim_{d \to +\infty} \frac{\alpha d}{\vec{k}^\top (\mathbf{X} - \mathbf{O}) + d} \left[\begin{pmatrix} \vec{\imath}^\top \\ \vec{\jmath}^\top \end{pmatrix} \mathbf{X} + \begin{pmatrix} t^x \\ t^y \end{pmatrix} \right] + \begin{pmatrix} c^x \\ c^y \end{pmatrix} = \alpha \begin{pmatrix} \vec{\imath}^\top \\ \vec{\jmath}^\top \end{pmatrix} \mathbf{X} + \alpha \begin{pmatrix} t^x \\ t^y \end{pmatrix} + \begin{pmatrix} c^x \\ c^y \end{pmatrix} . \tag{3}$$

This result indicates that the perspective projection with camera center at an infinite distance from the scene is an orthographic projection with scale $\alpha = \frac{f}{d}$, axes $\vec{\imath}$, $\vec{\jmath}$ and translation $\alpha(t^x,\ t^y)^\top + (c^x,\ c^y)^\top$. This happens because as the camera gets further away from the scene, the rays arriving at the image plane become more and more parallel to each other, almost orthogonal to the image plane (see Figure 2).

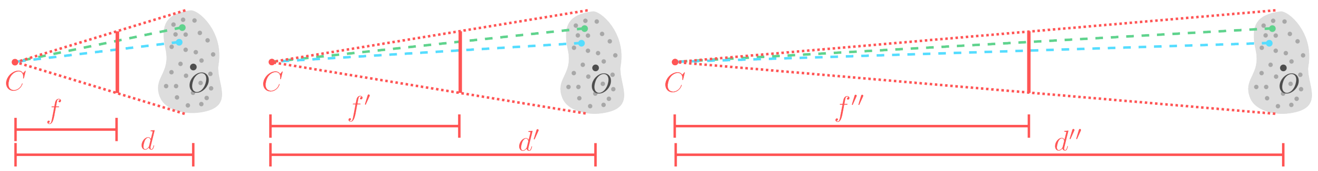


Figure 2: For the same scene, we can see how the projection lines of two points (the ones in blue and green) become more and more parallel as we get the camera away from the scene. In the image, we increased the distance d along with the focal length f so that $\frac{f}{d} = \frac{f'}{d'} = \frac{f''}{d''}$.

2.2 The Scaled-Orthographic Model

In Poelman and Kanade [10] a specific orthographic projection is defined to emulate the pinhole camera at infinity: the Scaled-Orthographic Model, also known as the weak-perspective camera. This model is defined by fixing the scaling factor to the ratio between the focal length and the distance to the scene of the emulated pinhole camera, $s = \alpha$. Also, we will write the translation vector in terms of the camera parameters to match those of the limit at infinity. In this case, the projection function will be a linear function ρ_{SO}

$$\begin{aligned} \rho_{\text{SO}} : \mathbb{R}^3 &\longrightarrow \mathbb{R}^2 \\ \mathbf{X} &\longmapsto \mathbf{x} = \frac{f}{\vec{k}^\top \mathbf{O} + t^z} \left[\begin{pmatrix} \vec{\imath}^\top \\ \vec{\jmath}^\top \end{pmatrix} \mathbf{X} + \begin{pmatrix} t^x \\ t^y \end{pmatrix} \right] + \begin{pmatrix} c^x \\ c^y \end{pmatrix}. \end{aligned} \tag{4}$$

Notice that this definition is the same as ρ_{P}, except that the point $\mathbf{X}$ in the denominator has been replaced by the center of the scene $\mathbf{O}$ so that it coincides with the perspective model at infinity. The new parameter $\mathbf{O}$ will be defined as the centroid of all the observed points of the scene, i.e. $\mathbf{O} = \frac{1}{N}\sum_j \mathbf{X}^j$. For that, we make the assumption that only a finite number N of space points is observed.

The fact that the perspective model is an orthographic projection at infinity is a good indication that the scaled orthographic projection of a scene will be a good approximation of the perspective projection for scenes far away from the camera. To further show this, we can study the distance between the two projections $\rho_{\text{P}}(\mathbf{X})$ and $\rho_{\text{SO}}(\mathbf{X})$. Let the scene be a set of space points $\{\mathbf{X}^j\}$ and the coordinates be centered on the centroid $\mathbf{O} = \frac{1}{N}\sum_j \mathbf{X}^j = \mathbf{0}$. We define a camera with center of projection $\mathbf{C}$, axes $\vec{\imath}$, $\vec{\jmath}$ and $\vec{k}$, focal length f and principal point $(c^x,\ c^y) = (0,\ 0)$, to simplify. Using these parameters also for the scaled-orthographic model we can compute both a perspective projection $\mathbf{x}_{\text{P}} = \rho_{\text{P}}(\mathbf{X})$ and a scaled orthographic projection $\mathbf{x}_{\text{SO}} = \rho_{\text{SO}}(\mathbf{X})$ for any point $\mathbf{X} = (X, Y, Z)^\top$. The distance between these two projected points is

$$d(\mathbf{x}_{\text{P}}, \mathbf{x}_{\text{SO}}) = \|\rho_{\text{P}}(\mathbf{X}) - \rho_{\text{SO}}(\mathbf{X})\| = \left| \frac{f}{Z+d} - \frac{f}{d} \right| \left\| \begin{pmatrix} \vec{\imath}^\top \\ \vec{\jmath}^\top \end{pmatrix} \mathbf{X} + \begin{pmatrix} t^x \\ t^y \end{pmatrix} \right\| = \left| \frac{Z}{Z+d} \right| \|\mathbf{x}_{\text{SO}}\| \ . \tag{5}$$

Therefore, the difference between the orthographic and perspective images is proportional to the image point distance to the "center" of the image and the ratio $\frac{z}{Z+d}$. When this ratio decreases, so does the difference between the two projections. This proves that the scaled orthographic model can be a good approximation of the perspective model in situations where $\frac{Z}{Z+d}$ is small. This is generally the case for long focal length images, where the scene observed (and the matched points) are far away in comparison to the relative depth of the scene. It is also the case for close-to-planar scenes, but we will see that those are not easy to handle with the scaled-orthographic model.

3 Tomasi-Kanade Factorization of the Orthographic Model

Tomasi and Kanade [14] presented a method to factorize the matrix built from the image measurements into two matrices representing shape and motion under the orthographic model. Later, Poelman and Kanade [10] extended this work to the scaled-orthographic model. We present the latter below.

Suppose we have a set of M scaled-orthographic cameras with parameters $\mathbf{C}_i$, $\vec{\imath}_i$, $\vec{\jmath}_i$, $\vec{k}_i$, focal length f_i and principal point $(0,\ 0)$. Let the scene be a set of N space points $\{\mathbf{X}^j\}$ and the origin of the world coordinates the centroid $\mathbf{O} = \frac{1}{N}\sum_j \mathbf{X}^j$. Then, the projection of the points $\{\mathbf{X}^j\}$ by each camera onto the image points $\{\mathbf{x}_i^j\}$ is described by the following equations:

$$\mathbf{x}_i^j = \begin{pmatrix} \vec{m}_i^\top \\ \vec{n}_i^\top \end{pmatrix} \mathbf{X}^j + \begin{pmatrix} a_i \\ b_i \end{pmatrix} \qquad \text{for} \quad i = 1, \dots, M \quad j = 1, \dots, N \tag{6}$$

where,

$$\begin{pmatrix} \vec{m}_i^\top \\ \vec{n}_i^\top \end{pmatrix} = \frac{f_i}{t_i^z} \begin{pmatrix} \vec{\imath}_i^\top \\ \vec{\jmath}_i^\top \end{pmatrix} , \qquad \begin{pmatrix} a_i \\ b_i \end{pmatrix} = \frac{f_i}{t_i^z} \begin{pmatrix} t^x \\ t^y \end{pmatrix} . \tag{7}$$

We can gather all image points in a $2M \times N$ measurement matrix $\mathcal{W}$, all the scaled axes on the $2M \times 3$ motion matrix $\mathcal{R}$, all 3D points in the $3 \times N$ shape matrix $\mathcal{S}$ and the translation vectors in a global translation vector $\mathcal{T}$ of length $2M$. The projection equations can be written then in matrix form

$$\underbrace{\begin{pmatrix} \mathbf{x}_1^1 & \mathbf{x}_1^2 & \cdots & \mathbf{x}_1^N \\ \vdots & \vdots & & \vdots \\ \mathbf{x}_M^1 & \mathbf{x}_M^2 & \cdots & \mathbf{x}_M^N \end{pmatrix}}_{\mathcal{W}} = \underbrace{\begin{pmatrix} \vec{m}_1^\top \\ \vec{n}_1^\top \\ \vdots \\ \vec{m}_M^\top \\ \vec{n}_M^\top \end{pmatrix}}_{\mathcal{R}} \underbrace{\left(\mathbf{X}^1 \dots \mathbf{X}^N\right)}_{\mathcal{S}} + \underbrace{\begin{pmatrix} a_1 \\ b_1 \\ \vdots \\ a_M \\ b_M \end{pmatrix}}_{\mathcal{T}} \left(1 \overset{N}{\dots} 1\right) . \tag{8}$$

Notice that the global translation vector $\mathcal{T}$ can be computed from the mean of image points thanks to the centroid being placed at the origin of space coordinates: if we multiply (8) on the right by the vector $\frac{1}{N}(1 \dots 1)^\top$ of size N,

$$\frac{1}{N}\sum_{j=1}^{N} \begin{pmatrix} \mathbf{x}_1^j \\ \vdots \\ \mathbf{x}_M^j \end{pmatrix} = \mathcal{R}\left(\frac{1}{N}\sum_{j=1}^{N} \mathbf{X}^j\right) + \mathcal{T} = \mathcal{R}\,\mathbf{O} + \mathcal{T} = \mathcal{T} . \tag{9}$$

Therefore, we define the matrix $\mathcal{W}^* := \mathcal{W} - \mathcal{T}(1 \ \dots \ 1)$ that can be computed from image data only. This matrix will have, when no noise is present, at most rank three due to the fact that $\mathcal{W}^* = \mathcal{R}\mathcal{S}$. This decomposition of $\mathcal{W}^*$ is a rank factorization and it is the key to Tomasi and Kanade [14] pose estimation method.

Remark. *The rank factorization of any matrix A of rank $r > 0$ is not unique. However, if we have two rank factorizations of the same matrix, $A = A_1A_2 = A_1'A_2'$, there always exists an $r \times r$ invertible matrix Π s.t. $A_1 = A_1'\Pi$ and $A_2 = \Pi^{-1}A_2'$.*

The matrix $\mathcal{R}$ from the factorization verifies several constraints given the fact that it is the motion matrix. These constraints are about the norm and scalar products of its rows, these are

$$\|\vec{m}_i\| = \|\vec{n}_i\| = \left|\frac{f_i}{t_i^z}\right| \quad \text{and} \quad \vec{m}_i^\top \vec{n}_i = 0 \quad \forall i = 1, \dots, M \tag{10}$$

In addition, there is a rotation and scaling ambiguity. For any rotation matrix R and scalar s, the matrices $s\mathcal{R}R^\top$ and $\frac{1}{s}R\mathcal{S}$ give equally valid motion and shape matrices respectively.

Minimal M. For $M \leq 2$ there is no unique reconstruction. The case $M = 1$ is obvious. In the case of two views, $M = 2$, given a possible reconstruction, any rotation of one of the views around the axis perpendicular to the projection direction of both views will produce another valid reconstruction with different motion and shape. This means that we need at least three views to proceed to unambiguous pose estimation.

Minimal N. We can see by studying the rank of $\mathcal{W}^*$ with respect to N that the minimal number of correspondences is 4. We know that $\text{rank}(\mathcal{W}) \leq 3$ and the number of correspondences N should not limit in any way this rank, otherwise we would not be able to compute $\mathcal{R}$ and $\mathcal{S}$ from a degenerate $\mathcal{W}^*$. If we write the measurement matrix like $\mathcal{W} = (w_1, \dots, w_N)$ and $\mathcal{W}^* = (w_1^*, \dots, w_N^*)$ we will have

$$w_j^* = w_j - \frac{1}{N}\sum_{j'=1}^{N} w_{j'} \quad \Rightarrow \quad \sum_{j=1}^{N} w_j^* = 0 \tag{11}$$

and then $\text{rank}(\mathcal{W}^*) < N$ since its N columns are linearly dependent. For $N \leq 3$, the matrix $\mathcal{W}^*$ would have rank lower than 3, so we need $N \geq 4$ in order to have a non-deficient $\mathcal{W}^*$ from which we can compute the motion and shape matrices.

3.1 The Pose Estimation Method

The factorization leads to the following pose estimation method. Let us have $N \geq 4$ corresponding image points throughout $M \geq 3$ images $\{\mathbf{x}_i^j\}$ and the (estimated) focal length of each view $\{f_i\}$.

1. From the image data we can compute $\tilde{\mathcal{W}}^* = \tilde{\mathcal{W}} - \tilde{\mathcal{T}}(1 \ \dots \ 1)$ (the tilde indicates that noise is present in the measurements).

2. $\tilde{\mathcal{W}}^*$ might have rank higher than three due to error and noise. We impose the rank deficiency by using the singular value decomposition. For $\tilde{\mathcal{W}}^* = U\Sigma V^\top$, we define $\mathcal{W}^* = U'\Sigma'(V')^\top$, where U' and V' are the first 3 columns of U and V respectively and Σ' is the upper-left 3×3 sub-matrix of Σ, formed by the three largest singular values.

3. A first rank factorization $\mathcal{W}^* = \hat{\mathcal{R}}\hat{\mathcal{S}}$ is given by $\hat{\mathcal{R}} = U'(\Sigma')^{\frac{1}{2}}$ and $\hat{\mathcal{S}} = (\Sigma')^{\frac{1}{2}}(V')^\top$.

4. We search for a 3×3 invertible matrix Q such that the new rank factorization $\mathcal{W}^* = (\hat{\mathcal{R}}Q)(Q^{-1}\hat{\mathcal{S}})$ is a valid motion-shape decomposition. Hence, $\mathcal{R} = \hat{\mathcal{R}}Q$ should verify the constraints in (10). They translate to

$$\hat{m}_i^\top QQ^\top \hat{m}_i - \hat{n}_i^\top QQ^\top \hat{n}_i = 0 \quad \text{and} \quad \hat{m}_i^\top QQ^\top \hat{n}_i = 0 \qquad \forall i = 1, \dots, M \tag{12}$$

The numerical value of $\|\hat{m}_i\|$ is unknown since we do not have t_i^z. In consequence, only the equality of norms $\|\hat{m}_i\| = \|\hat{n}_i\|$ can be imposed and we have a linear system of $2M$ homogeneous equations on the coefficients of the matrix $\Pi = QQ^\top$. Since the matrix Π is positive semi-definite, its coefficients are reduced to 6 unknowns and the $2M \geq 6$ equations are enough to determine the coefficients up-to-scale by finding the kernel of the matrix of the homogeneous system (the matrix Π could also be estimated imposing its positive semi-definiteness using convex optimization). Afterwards, we recover Q through the Cholesky factorization of Π.

5. The final factorization of $\mathcal{W}^*$ is $\mathcal{R} = \hat{\mathcal{R}}Q$ and $\mathcal{S} = Q^{-1}\hat{\mathcal{S}}$. From it we can compute all camera parameters. For the axes,

$$\vec{\imath}_i = \frac{\vec{m}_i}{\|\vec{m}_i\|}, \quad \vec{\jmath}_i = \frac{\vec{n}_i}{\|\vec{n}_i\|}, \quad \vec{k}_i = \vec{\imath}_i \times \vec{\jmath}_i \tag{13}$$

and for the translation vector we estimate the value of f_i/t_i^z using the mean of the norms of both $\vec{m}_i$ and $\vec{n}_i$ (since they might not be exactly equal),

$$\vec{t}_i = \frac{2}{\|\vec{m}_i\| + \|\vec{n}_i\|}(a_i,\ b_i,\ f_i)^\top. \tag{14}$$

The pose estimation is summed up in Algorithm 1. Then Algorithm 2 recovers the complete rotations and translations.

```
Algorithm 1: Pose estimation algorithm
  Input: Point projection matrix $\tilde{\mathcal{W}} = \left(x_i^j\right)_{ij} \in \mathbb{R}^{2M\times N}$, projections of $N$ 3D points $X^j$ into $M$
         views indexed by $i$.
  Output: Matrix $\mathcal{R} \in \mathbb{R}^{2M\times 3}$ of incomplete rotations, vector $\mathcal{T} \in \mathbb{R}^{2M}$ of incomplete
          translations, and structure matrix $\mathcal{S} \in \mathbb{R}^{3\times N}$, so that $\tilde{\mathcal{W}} \approx \mathcal{R}\mathcal{S} + \mathcal{T}$.
  1  $\mathcal{T} \leftarrow \frac{1}{N}\tilde{\mathcal{W}}\begin{pmatrix}1 & \dots & 1\end{pmatrix}^\top$                                  // See (9)
  2  $\tilde{\mathcal{W}}^* \leftarrow \tilde{\mathcal{W}} - \mathcal{T}\begin{pmatrix}1 & \dots & 1\end{pmatrix}$
  3  SVD decomposition of $\tilde{\mathcal{W}}^* = U\Sigma V^\top$, $U \in O(2M)$, $V \in O(N)$ and diagonal $\Sigma \in \mathbb{R}^{M\times N}$
  4  $U' \leftarrow U_{:,1:3}$, $V' \leftarrow V_{:,1:3}$, $\Sigma' \leftarrow \Sigma_{1:3,1:3}$                   // Matlab slicing notations
  5  $\hat{\mathcal{R}} \leftarrow U'(\Sigma')^{1/2}$, $\hat{\mathcal{S}} \leftarrow (\Sigma')^{1/2}(V')^\top$
  6  Write $2M \times 6$ matrix $D$ such that for $i = 1 \dots M$

         $\begin{aligned} D_{2i-1,:} &= \begin{pmatrix} B^i_{11}/2 & B^i_{22}/2 & B^i_{33}/2 & B^i_{12} & B^i_{13} & B^i_{23}\end{pmatrix} \\ D_{2i,:} &= \begin{pmatrix} C^i_{11}/2 & C^i_{22}/2 & C^i_{33}/2 & C^i_{12} & C^i_{13} & C^i_{23}\end{pmatrix} \end{aligned}$        (15)

     with $B^i = 2(\hat{\mathcal{R}}_{i,:}^\top\hat{\mathcal{R}}_{i,:} - \hat{\mathcal{R}}_{i+1,:}^\top\hat{\mathcal{R}}_{i+1,:})$ and $C^i = (\hat{\mathcal{R}}_{i,:}^\top\hat{\mathcal{R}}_{i+1,:} + \hat{\mathcal{R}}_{i+1,:}^\top\hat{\mathcal{R}}_{i,:})$.      // Writing (12)
  7  SVD decomposition of $D = \hat{U}\hat{\Sigma}\hat{V}^\top$, $\hat{V} \in O(6)$.
  8  $W \leftarrow \frac{\hat{V}_{16}}{|\hat{V}_{16}|}\hat{V}_6$.                                                // Make $W_1 > 0$
  9  $Q_2 \leftarrow \begin{pmatrix} W_1 & W_4 & W_5 \\ W_4 & W_2 & W_6 \\ W_5 & W_6 & W_3 \end{pmatrix}$
 10  Cholesky decomposition of $Q_2 = QQ^\top$
 11  $\mathcal{R} \leftarrow \hat{\mathcal{R}}Q$, $\mathcal{S} = Q^{-1}\hat{\mathcal{S}}$
```

Rotation ambiguity. Any rotation can be applied to the final result. The usual approach is to bring $\vec{m}_1$ and $\vec{n}_1$ to $(1,0,0)^\top$ and $(0,1,0)^\top$.

Depth ambiguity. The final factorization obtained in step 4 can be transformed into a new valid factorization $\mathcal{W}^* = (\mathcal{R}A)(A\mathcal{S})$ by using matrix

$$A = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{pmatrix}. \tag{16}$$

This transformation gives an equivalent 3D reconstruction but the views are placed “at the back” of the scene. For orthographic data this ambiguity cannot be solved. In Figure 3 there is an example of two possible configurations of the same reconstruction.

Algorithm 3 returns the two possible normalized configurations of structure and motion.

Algorithm 2: Normalization of rotations and translations

```
Input: Matrix $\mathcal{R} \in \mathbb{R}^{2M\times 3}$ of incomplete rotations and vector $\mathcal{T} \in \mathbb{R}^{2M}$ of translations, issued
       from Algorithm 1, focal lengths $f_i$.
Output: Rotation matrices $R^i \in O(3)$ and translations $T^i$
for $i = 1 \dots M$ do
    $R^i_{1:2,:} \leftarrow \begin{pmatrix} \mathcal{R}_{2i-1,:}/\|\mathcal{R}_{2i-1,:}\| \\ \mathcal{R}_{2i,:}/\|\mathcal{R}_{2i,:}\| \end{pmatrix}$
    $R^i_{3,:} \leftarrow \left((R^i_{1,:})^\top \times (R^i_{2,:})^\top\right)^\top$
    $T^i_{1:2} \leftarrow \mathcal{T}_{2i-1:2i}$, $T^i_3 \leftarrow f_i$
    $T_i \rightarrow 2\,T_i/(\|\mathcal{R}_{2i-1,:}\| + \|\mathcal{R}_{2i,:}\|)$
```

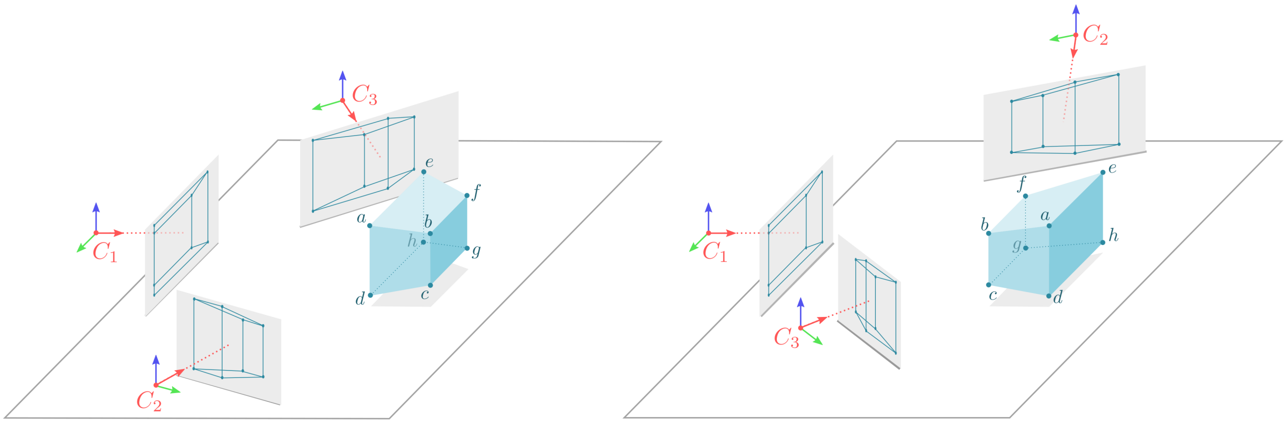


Figure 3: Two possible configurations for the same image points in three orthographic views. Notice that for each view the relative depth of the observed points swaps from one configuration to the other.

Algorithm 3: Recover ambiguous multi-view structure and motion

```
Input: Rotations $\{R^i \in SO(3)\}$, translations $\{T^i \in \mathbb{R}^3\}$, structure $\mathcal{S} \in \mathbb{R}^{3\times N}$
Output: Two normalized possible multi-view configurations $(\{R^i_k\}, \{T^i_k\}, \mathcal{S}^i_k)$, $k = 1, 2$
for $i = 1 \dots M$ do                    // Change coordinate frame so that $R^1 = I$ and $T^1 = 0$
    $R^i_1 \leftarrow R^i$
    $R^i_2 \leftarrow A\,R^i\,A$                                    // Matrix $A$ defined in (16)
    $R^i_1 \leftarrow R^i_1\,(R^1_1)^\top$, $T^i_1 \leftarrow T^i - R^i_1\,T^1$
    $R^i_2 \leftarrow R^i_2\,(R^1_2)^\top$, $T^i_2 \leftarrow T^i - R^i_2\,T^1$
$\alpha_1 \leftarrow 1/\|T^2_1\|$, $\alpha_2 \leftarrow 1/\|T^2_2\|$    // Scale so that first two views are at unit distance
for $i = 1 \dots M$ do
    $T^i_1 \leftarrow \alpha_1 T^i_1$, $T^i_2 \leftarrow \alpha_2 T^i_2$
$\mathcal{S}_1 \leftarrow \alpha_1 \mathcal{S}$, $\mathcal{S}_2 \leftarrow \alpha_2 A\,\mathcal{S}$
```

3.2 Application to Perspective Cameras

As seen in Section 2.2 the scaled-orthographic model can be seen as an estimation of the perspective model in some particular scenes like the images acquired with long focal lengths. It turns out that the pose estimation of narrow-angle images is a particular difficult task for perspective-based methods like the fundamental matrix and, therefore, we propose to use the scaled-orthographic model which proves to be more robust.

To apply the method of the previous section to pose estimation of perspective cameras we can use the image data as if it was produced by a scaled orthographic camera. Once we estimate the camera parameters we can reinterpret them as perspective cameras instead of orthographic and proceed with a bundle adjustment refinement if desired. The depth ambiguity can be solved by two different approaches. We can manually choose between the two possible solutions by identifying one image point from the correspondences in the images that is closer to the camera (on the first plane of the scene). Otherwise, if no manual help is intended, there is no other solution than to keep both solutions and to choose the one that gives smaller reprojection error after the bundle adjustment step.

4 Implementation

The pose estimation of perspective cameras based on the scaled-orthographic method is implemented by the Matlab function `OrthographicPoseEstimation`. For this method, the input data are the matching points throughout $M \geq 3$ views and the calibration parameters for each camera (calibration matrix K with the focal length and principal point). The output are the two possible orientations in variables `Sol1` and `Sol2` that contain a rotation matrix R and translation vector $\vec{t}$ for each view and also matrices $\mathcal{R}$, $\mathcal{S}$, $\mathcal{T}$ from the Tomasi and Kanade factorization ($\mathcal{S}$ contains the 3D reconstruction of the matching points passed as input). The general pipeline for pose estimation from images to a refined solution would be the following (see Algorithm 4):

Matching between pairs: For each pair of images the matching features should be found. This step can be done with any available software and it is not implemented in our code. In the demo, we use SIFT point matches, as implemented in IPOL [11].

Extract tracks: From the couples of matching points of each pair of images, the consistent tracks between three views are extracted by the function `matches2triplets`. At least 4 tracks are needed to continue with the procedure.

RANSAC: A set of inliers from the tracks between the three views can be chosen by Random Sample Consensus. For our implementation we chose an a contrario approach (adapted from [7]) with the scaled-orthographic model for three views. The AC-RANSAC is programmed in function `AC_RANSAC_Orthographic`, where the set of inliers is computed using the orthographic model with a minimal sample of 4 tracks. The output is the maximal set of inliers.

Initial pose estimation: The scaled orthographic method is applied to all the inliers to get a first estimation of the pose for each camera. Calling the function `OrthographicPoseEstimation` will provide two possible configurations of the cameras. As seen in the description of this method in Section 3.1, the operations needed in the pose estimation are computationally simple and only involve the SVD and the Cholesky decomposition.

Bundle Adjustment: The configurations obtained in the last step are used as two possible initializations for the orientations of perspective cameras in a bundle adjustment [15]. This method consists in minimizing the reprojection error over the possible cameras orientations and space points

$$\min_{\{R_i,\vec{t}_i\}_i,\,\{\mathbf{X}^j\}_j} \sum_{j=1}^{N}\sum_{i=1}^{M} \|\mathbf{x}_i^j - \rho_{\mathrm{P}_i}(\mathbf{X}^j)\|^2, \tag{17}$$

where ρ_{P_i} is the perspective projection as in (2) associated to camera i. The optimization is carried out by the function `BundleAdjustment` using the Levenberg-Marquardt algorithm [5] (already implemented in Matlab). After using the two possible initializations, the solution with lower reprojection error at the end of the optimization is chosen as the correct final pose estimation.

Algorithm 4: Pose estimation pipeline

```
  Input: Images $I_1,\dots,I_M$, $M \geq 3$, focal length $f$.      // Only for $M = 3$ in our code
  Output: Rotations $R^i$ and translation $T^i$ $(i = 1\dots M)$, 3D point coordinates in matrix $\mathcal{S}$.
1 For each pair of images $\{I_i, I_j\}$, compute matching points $(x_i^k, x_j^k)$      // SIFT matches
2 Extract tracks of points $\mathcal{W} = (x_i^j)$, $i = 1\dots M$, $j = 1\dots N$
3 Apply AC-RANSAC. Each sample of 4 tracks yields two hypotheses of structure and motion.
4 Reestimate structure and motion from all inlier tracks (yielding again two solutions).
5 Apply bundle adjustment to both solutions, using perspective projection $\rho_{P_i}$
6 Discard the solution of structure and motion with higher reprojection error
```

In the code provided, the function `mainPoseEstimation.m` carries out the last four steps of the pipeline described above for a given set of matching points between the pairs of a triplet of images. In the online demo, the entire pipeline is carried out for a triplet of images provided by the user, and the keypoint matching step is handled by the IPOL published implementation of SIFT [11]. Additionally, the user must provide the focal length value. However, this focal length f should be expressed in pixel units, while it is usually known in millimeters, f_{mm}. The latter can be extracted from the EXIF information of a photographic file in JPEG format. To do the conversion, it is necessary to know the size of a pixel in millimeters, which can be computed by dividing the CCD width W (in millimeters) by the horizontal number of pixels w. The value of W is *not* part of the EXIF information, but the camera model is part of EXIF. From the latter, the sensor size can be looked up in a database, such as [2]. Finally, original photographs are often too large for the demo system to answer in an acceptable time, so a zoom-out of factor z may be applied. Finally, the conversion follows the formula

$$f = z \cdot \frac{w}{W} \cdot f_{mm}. \tag{18}$$

5 Experiments

To study the robustness and performance of this method, we evaluated its results when applied to different scenes, comparing them to other algorithms based on the perspective model. Synthetic data was used to test the sensibility to different parameters such as noise, focal length and configuration of the space points and cameras. We also applied the method to real images taken with long focal lengths to test its performance in real situations to evaluate its usefulness. Since the minimum number of views needed in the orthographic pose estimation is three, we evaluated the results on scenes composed of three views.

The errors used for evaluation are the **reprojection error** and the **angular error** in rotations and translations[2]. The first does not depend on a known ground truth and can always be computed. It is computed by triangulating the 3D points using the image points and the estimated poses and then reprojecting them onto the images. The error is defined as the RMS of the distances between the reprojected points and the original measured image points. For M views and N correspondences

$$e_{\text{repr}} = \sqrt{\frac{1}{N}\sum_{j=1}^{N}\left(\frac{1}{M}\sum_{i=1}^{M}\|\mathbf{x}_i^j - \rho_{\mathrm{P}_i}(\mathbf{X}^j)\|^2\right)}, \tag{19}$$

where $\mathbf{X}^j$ is the triangulated point using $\mathbf{x}_1^j, \ldots, \mathbf{x}_j^M$. The triangulation is carried out in a linear fashion with the DLT method, described in [4].

The angular error compares the estimated poses to the ground truth. To do the comparison one of the cameras is aligned with the known true position and the scale is adjusted, then the rotations and translations defining the other cameras are compared to the true ones and the angle of their differences is computed. For instance, if the true poses of the second and third camera with respect to the first one are $[R^0_{21},\ \vec{t}^{\,0}_{21}]$ and $[R^0_{31},\ \vec{t}^{\,0}_{31}]$ respectively and the estimated poses are $[R_{21},\ \vec{t}_{21}]$ and $[R_{31},\ \vec{t}_{31}]$, then the angular errors are

$$e_{\text{rot}} = \frac{1}{2}\big(\angle(R^0_{21}, R_{21}) + \angle(R^0_{31}, R_{31})\big) \qquad e_{\text{trans}} = \frac{1}{2}\big(\widehat{\vec{t}^{\,0}_{21}\ \vec{t}_{21}} + \widehat{\vec{t}^{\,0}_{31}\ \vec{t}_{31}}\big), \tag{20}$$

where $\angle(R, R')$ denotes the angle of the rotation corresponding to the matrix $RR'^{\top}$. It can be computed using the trace $\angle(R, R') = \arccos((\mathrm{tr}(RR'^{\top}) - 1)/2)$.

For comparison with a perspective-based method, we evaluate the results given by the pose estimation using the fundamental matrix on the same scenes. More specifically, the two fundamental matrices F_{21} and F_{31} are computed from their correspondences by the 8-point algorithm [3]. Then, the relative orientations $[R_{21},\ \vec{t}_{21}]$ and $[R_{31},\ \vec{t}_{31}]$ are extracted by singular value decomposition of the essential matrices [4]. Finally, we choose the global poses $[\mathrm{Id}\ 0]$, $[R_{21},\ \vec{t}_{21}]$, $[R_{31},\ \lambda\vec{t}_{31}]$ with λ the solution of

$$\arg\min_{\lambda\in\mathbb{R}} \sum_{j=1}^{N} \|\bar{\mathbf{x}}_3^j \times \big(K_3(R_{31}\mathbf{X}^j + \lambda\vec{t}_{31})\big)\|^2, \tag{21}$$

where the space points $\{\mathbf{X}^j\}_{j=1,\ldots,N}$ are reconstructed using only the image points in the first two images and the camera matrices $K_1[\mathrm{Id}\ 0]$, $K_2[R_2,\ \vec{t}_2]$ and the notation $\bar{\mathbf{x}}$ is used to indicate the homogeneous coordinates of $\mathbf{x}$. This is not the minimization of a geometric error but an algebraic one. The solution has the closed form

$$\lambda = \frac{\sum_{j=1}^{N}\big(\bar{\mathbf{x}}_3^j \times (K_3R_{31}\mathbf{X}^j)\big)^{\top}\big(\bar{\mathbf{x}}_3^j \times (K_3\vec{t}_{31})\big)}{\sum_{j=1}^{N}\|\bar{\mathbf{x}}_3^j \times (K_3\vec{t}_{31})\|}. \tag{22}$$

We also compare the results to the final pose given after a bundle adjustment based on the perspective model with the initial pose obtained by one of the previous methods. The optimization is carried out by the Levenberg-Marquardt algorithm. In most of our experiments, the same minimum is reached regardless of the method used as initialization, so the results will be represented as one.

5.1 Synthetic Data

The standard scene (Figure 4) for our experiments is composed of a set of space points contained in a cube of 400mm×400mm centered in the world's origin. The points are projected onto three cameras

[2] since only translation direction can be estimated, not its norm.

and Gaussian noise is added to the image points with $\sigma = 1$ pixel, if not stated otherwise. A sample of 20 points is used for the computations of the different models and a test set of 100 to evaluate the results. The image size is 1800×1200 pixels that correspond to a 36mm×24mm sensor. The cameras all point at the origin and their placement depends on the chosen focal length, $\mathbf{C}_1 = f \cdot (0, -28, 8)$, $\mathbf{C}_2 = f \cdot (-8, -20, 0)$ and $\mathbf{C}_3 = f \cdot (12, -16, -4)$, so that the ratio α is fixed, see Figure 4. Results are averaged over 20 simulations of data.

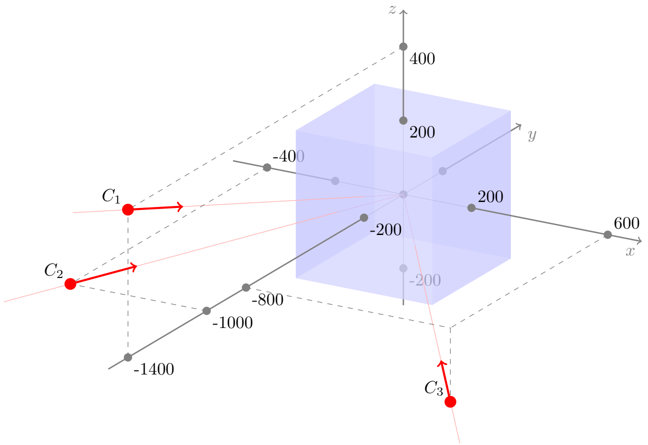


Figure 4: Illustration of the configuration of the synthetic data scene for f = 50mm. When the focal length is increased, the camera centers move along the red lines a proportional distance from the origin.

5.1.1 Focal Length

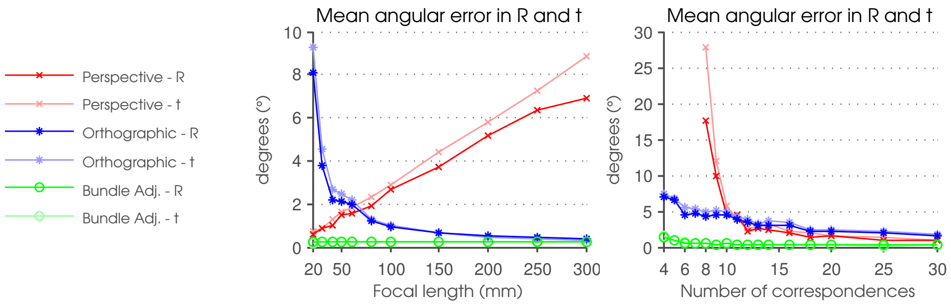
In a first approach, we look at the angular error for different focal lengths, f varying in $[20, 300]$mm. This changes not only f but also the position of the cameras as stated previously. We can see in Figure 5a that the initial pose given by the perspective-based estimation loses accuracy linearly with the increase of the focal length. On the other hand, the orthographic method has bad results for short lengths but rapidly gains accuracy and gets better results than the perspective-based method for lengths starting at $f = 60$mm. For $f \geq 200$mm the orthographic solution is really close to the final solution given by the bundle adjustment, both solutions getting less than 0.5° error.

5.1.2 Number of Correspondences

The orthographic method shows a good stability also in the number of correspondences used for the pose estimation. Here we vary the cardinal M of the sample of image points used in the computation of the estimated pose for all methods. The fundamental matrix can be computed with the 8-point algorithm for $M \geq 8$ while the orthographic method only needs $M \geq 4$. Figure 5b shows how the fundamental matrix gets bad results when a minimal or close to minimal set of correspondences is used and how the orthographic solution is not so affected by M. This also means that when $4 \leq M \leq 7$ and the fundamental matrix cannot be computed, the orthographic model can give a first pose estimation not so far from the ground truth. The stability of the factorization method even with few correspondences can be attributed to the strong rank-3 constraint, which prevents overfitting to the observations.

5.1.3 Robustness to Noise

Varying the Gaussian noise added to the image measurements with $\sigma \in [0, 3]$ pixels, the orthographic method proves to be much more robust to noise than a perspective-based method. The angular error

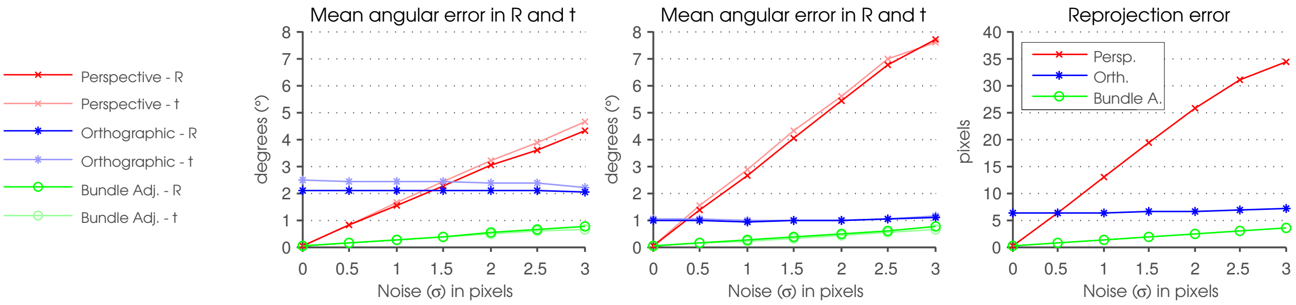


(a) (b)

Figure 5: Angular errors for each method changing different parameters. Two lines are drawn for each method, one for the mean angular error in rotations (-R) and another for the mean error in translations (-t). In (a) the focal length is varied and in (b) the focal is fixed at $f = 50$mm while the number of correspondences is varied.

of rotation and translation stays almost constant in comparison to the increasing error proportional to σ of the initial pose given by the perspective-based method. Testing with $f = 100$mm (Figure 6b) we see that the orthographic solution has a constant error smaller than the perspective solution for $\sigma \geq 0.5$ pixel. However, with a shorter length $f = 50$mm (Figure 6a) the performance of the orthographic method is not as good, though still almost constant, and it is surpassed by the perspective solution when noise is not high.

When we look at the reprojection error, for example with $f = 50$ mm and varying the noise in Figure 6c, we can see that the error in the perspective solution is extremely affected by increasing the noise while the orthographic solution has a smaller constant error for $\sigma \geq 0.5$ pixel. On the other hand, we have seen that the angular error for this focal length is smaller in the perspective solution for noise with $\sigma \leq 1.5$ pixels.



(a) $f = 50$mm (b) $f = 100$mm (c) $f = 50$mm

Figure 6: The angular errors in rotations and translations for varying Gaussian noise are shown in (a) and (a) for two different fixed focal lengths. The reprojection error in pixels is shown in (c) for a short focal length.

5.1.4 Planar Scene

There is a particular case where the orthographic method fails: the case of quasi planar or planar scenes. We evaluated the failures of the orthographic and perspective methods when the scene becomes planar. The planarity of a scene is measured with the percentage

$$\text{planarity} = 100 \cdot \left(1 - \frac{v_3}{v_1}\right), \tag{23}$$

where v_1 and v_3 are respectively the first and third eigenvalues of the matrix $X^\top X$ formed from the point cloud[3].

We consider that a pose estimation is a **valid solution** when after applying bundle adjustment it gets small angular errors, $e_{\text{rot}} \leq 5°$ and $e_{\text{trans}} \leq 10°$. In Figure 7a it is clear how the failures in the orthographic model rapidly rise for a planarity greater than 95%. The method based on the fundamental matrix is not robust to planarity either, but it manages to give at least 50% of valid solutions. It is known that for a perspective camera model observing a planar scene the homography between images should be used to compute the pose [17], so we also included the results with this perspective method.

The failures of the orthographic model for the planar scenes can be explained by the fact that the situation is similar to having only $N = 3$ points. If all the points projected onto the three orthographic cameras lie on the same 3D plane, the matrix $\mathcal{S}$ in (8) has rank 2 which makes rank$(\mathcal{W}^*) \leq 2$. Therefore, the factorization $\mathcal{W}^* = \mathcal{RS}$ is no longer a rank decomposition and there is no guarantee that the first factorization given by SVD in step 3 of the pose estimation method will lead to the correct solution. This usually translates into obtaining a non positive semi-definite matrix Q in step 4 or simply getting a wrong pose estimation. Even if the planar hypothesis is assumed, the new pose estimation problem involves solving a system of 6 quadratic equations for 6 unknowns leading up to 64 possible solutions, a reconstruction ambiguity not easy to address.

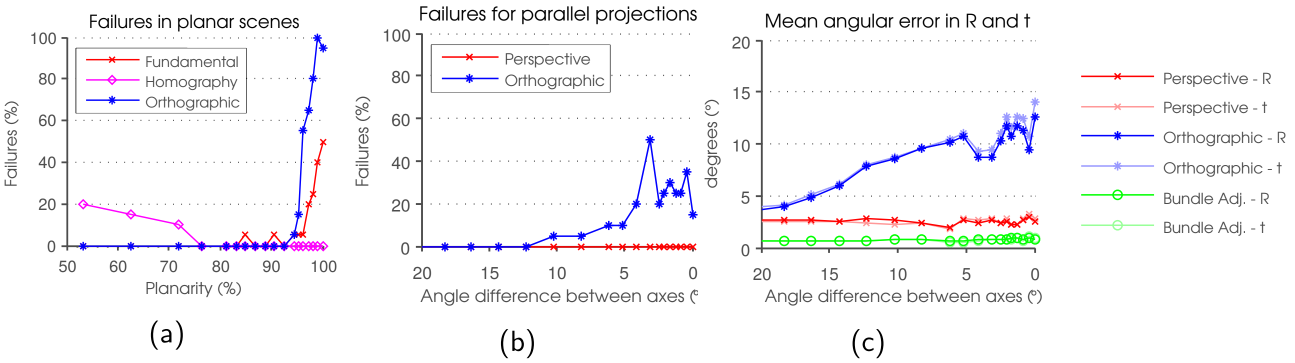


Figure 7: In (a) the percentage of failures is shown for the orthographic model and for two perspective methods (fundamental matrix and homography) in the case of the space points lying on a quasi-planar surface. The focal length is fixed at 100mm. The case of quasi-parallel camera axes is studied in (b) with the percentage of failures and in (c) with the angular errors.

5.1.5 Parallel Camera Axes

The projection rays in the orthographic model are parallel to each other, no matter the depth or position of the space points. For this reason, having M orthographic cameras with different position but same direction of projection $(\vec{k}_i)$ will, in fact, create M equivalent orthographic images, i.e. related only by scaling, translation and rotation on the plane. This translates into not having enough information for the depth recovery of the space points since in Equation (8) the matrix $\mathcal{R}$ will have only rank 2. Even if we are working with perspective data, this configuration generates ambiguous data for the proposed pose estimation method.

We analyzed the potential instability of the orthographic pose estimation on these type of scenes by modifying the synthetic data used. In Figure 8 we can see the modified scene where the cameras positions are fixed but the orientation varies from all cameras pointing at the origin to three parallel camera axes. The experiments showed that the method starts to fail as the camera axes get close to parallel (Figure 7b) and the angular error of the obtained pose increases above 10° (Figure 7c).

[3] $X^\top X$ is the estimation of the covariance matrix of the point cloud.

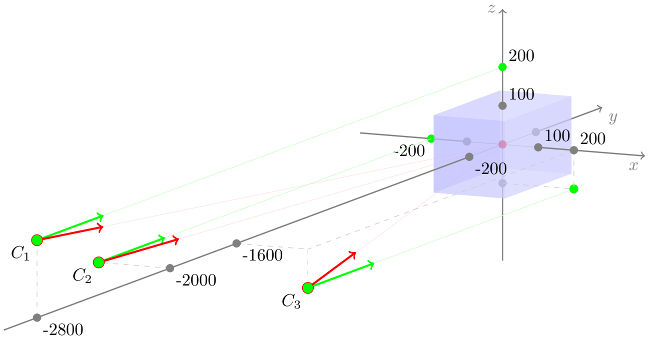


Figure 8: The synthetic data scene to study parallel camera axes. The cameras orientations are varied from the configuration in red (all cameras pointing at the origin) to configuration in green, where all three camera axes are parallel and the cameras are directed toward the green points.

However, the pose estimation based on the fundamental matrix is not at all affected by this special configuration.

5.2 Real Data

We have evaluated the orthographic method along with the perspective method based on the fundamental matrix on several real databases. Again, we test the two methods on triplets of images and compare the errors and the failures of each method. We also show the results for the solutions after applying a bundle adjustment (BA) step to both methods. In these real experiments, we chose to use a small sample of 50 correspondences on the bundle adjustment in order to improve the speed of the optimization.

To select the inlier tracks of each evaluated triplet, we use an a contrario RANSAC adapted to each method. In the case of the fundamental matrix, the RANSAC method is applied to each pair of views separately and the common inliers in the three pairs are selected.

In Tables 1–4 we show, for each dataset, several statistics averaged over all evaluated triplets: the resulting reprojection error (e_{repr}), rotation and translation errors (e_{rot} and e_{trans}), the percentage of correspondences selected as inliers, the percentage of valid cases (VC) as described in Section 5.1.4 and the number of iterations of the bundle adjustment.

Reims. This dataset is made out of a total of 55 images all acquired with a long focal length ($f = 400$mm) and it covers the main interior wall of the cathedral in Reims, France (see Figure 9). We tested the 200 triplets of images with the most correspondences and we compared the results to an "artificial" ground truth, the solution given by the SfM realized by MicMac [9] with not only the long focal length images but also with 16 other images with shorter focal length (7 with $f = 100$mm and 9 with $f = 50$mm) to stabilize the process. The internal calibration and distortion on the images was estimated by the same MicMac process and used in our computations. For this particular dataset, most of the scenes are quasi planar and the cameras are aligned. We can see in Table 1 that while both methods have problems to estimate the translations, the orthographic model gives a better estimate according to the lower number of mean iterations needed to reach the minimum and the higher percentage of pose estimation success.

EPFL fountain-P11 [13]. In juxtaposition to the previous case, we tested our algorithm in a short focal length dataset to evaluate its robustness. The fountain-P11 dataset is made out of a total of 10 images taken with a 35mm equivalent focal length of 32.5mm and has an available ground truth. We tested 70 of the possible triplets and the results are shown in Table 2. The orthographic

Figure 9: Some images of the Reims dataset. The first four on the left were taken with a focal length of 400mm and used in our experiments. The last one on the right was taken with a focal length of 100mm and used in the computations of the MicMac SfM algorithm.

	e_{repr}	e_{rot}	e_{trans}	inliers	VC	# iter
Perspective	240.995	1.554	42.70	88.6%	75.5%	68.9
after BA	0.410	0.461	1.52			
Orthographic	7.926	2.660	36.78	84.9%	82.0%	41.5
after BA	0.413	0.457	1.56			

Table 1: Statistics for 200 triplets of the Reims dataset.

method does not manage to get similar results to the perspective method; moreover, it has much higher reprojection and angular errors. However, the orthographic model is able to give as much as 91.43% of valid results, meaning that, even if the perspective method is the obvious choice for this kind of scenes, the orthographic method would still be a viable option to get a first pose estimation to proceed to bundle adjustment.

	e_{repr}	e_{rot}	e_{trans}	inliers	VC	# iter
Perspective	13.084	0.49	1.87	95.9%	100.0%	4.5
after BA	0.366	0.16	0.17			
Orthographic	76.219	29.38	26.50	48.0%	91.4%	36.2
after BA	0.369	0.16	0.18			

Table 2: Statistics for 70 triplets of the EPFL fountain-P11 dataset.

Statue. To show the accuracy of the orthographic method for long focal lengths, we acquired a new set of images with a focal of length approximately 1000mm. The images cover the face of a statue (Figure 10) positioned in the middle of one of the fountains of Château de Champs-sur-Marne, France. The dataset consists of a set of 58 images and we tested 200 possible triplets. We can see in Table 3 how the mean reprojection error for the perspective-based method is extremely high while for the orthographic model it is just a little above 1 pixel in the initial pose estimation. Looking at the number of iterations needed to reach the minimum in the bundle adjustment it becomes clear that the orthographic method is more suited for the long focal length case.

5.2.1 Example

As an example, we show the results for one triplet of images from the Statue dataset. In Figure 11 the images are shown along with the 982 tracks between them. The RANSAC based on the orthographic

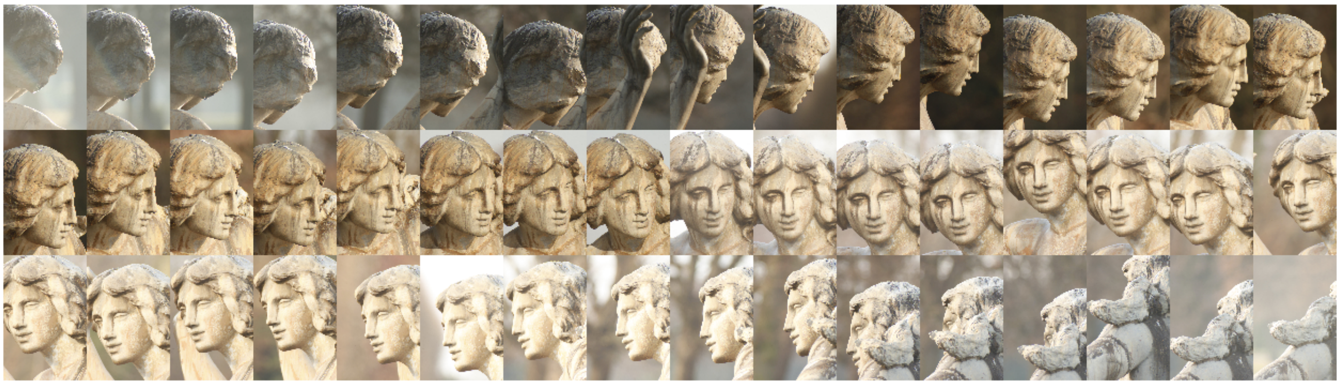


Figure 10: A sample of 48 images of the Statue dataset.

	e_{repr}	inliers	VC	# iter
Perspective	4964.494	59.9%	100%	64.3
after BA	0.751			
Orthographic	1.526	58.3%	100%	10.1
after BA	0.647			

Table 3: Statistics for 200 triplets of the Statue dataset.

model gives a big enough set of inliers for the pose estimation but leaves out many true tracks. However, the method manages to correctly label the true outliers (notice on the right of the third image in Figure 11 a set of points that were marked as a correspondence but are clearly not visible in the other two images).

For this triplet of images, the solution found by the orthographic method is much closer to the final refined solution than the initial pose given by the perspective method. We can see this quantitatively, in the number of iterations needed to reach the minimum in Table 4, where the iterations taken by the perspective solution are much higher. Also qualitatively in Figure 12, where we can see how the initial pose given by the orthographic method is practically the same as the final refined solution, while the perspective method gives a much worse initial guess. Looking at the orthogonal views of the estimated poses in Figure 12, it becomes clear that the error of the perspective-based method is mainly in the direction of the projection, which is related to a bad estimation in the depth of the object with respect to the camera.

	e_{repr}	e_{repr}-BA	inliers	# iter
Perspective	5617.4	0.0218	528	24(+4)
Orthographic	112.6	0.0218	496	7(+4)

Table 4: Results for the calibration of the triplet of images in Figure 11. The final solution of the bundle adjustment was produced with an extra minimization step involving all inliers that took 4 iterations in both cases. That is why it appears (+4) in the iterations column.

6 Conclusion

In this article we have described a non-perspective model for pose estimation of perspective cameras and we have provided our own implementation. The model is based on the orthographic projection that we have theoretically shown to model a pinhole camera with center at infinity. With the

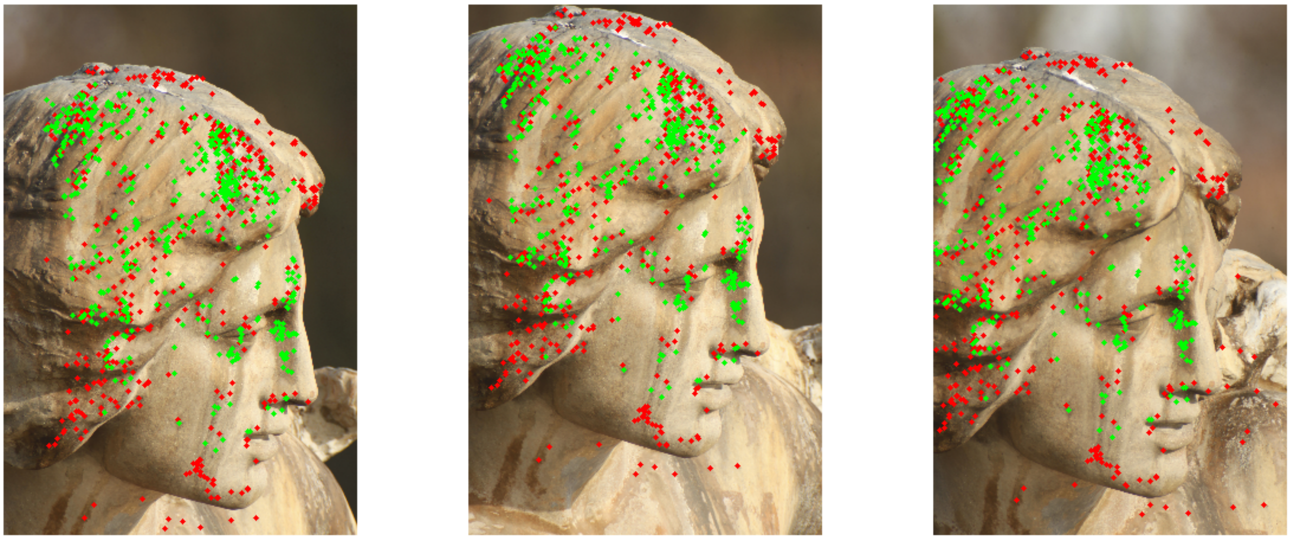


Figure 11: A triplet of images from a statue in Château de Champs. The matches are drawn in green if they are considered inliers by the orthographic AC-RANSAC and red otherwise.

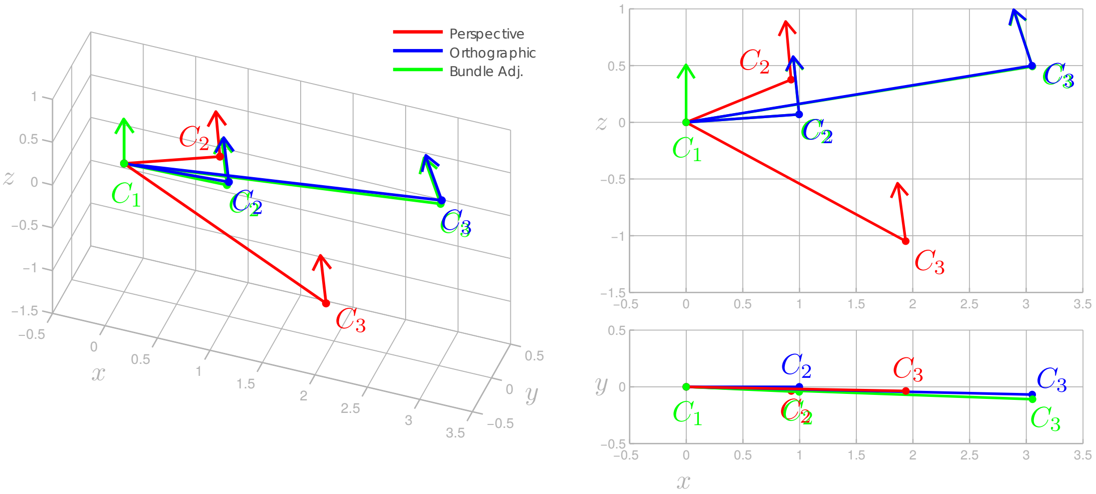
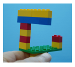


Figure 12: Pose estimation of the triplet of images in Figure 11 given by the orthographic and perspective methods along with the solution of the bundle adjustment. Different views are shown for the configuration of the poses: a general view on the left and two orthogonal views on the right; one perpendicular to the y-axis (top) and one perpendicular to the z-axis (bottom).

experiments we have proven that this translates into a good performance in the pose estimation task in the case of cameras with long focal length lenses. Moreover, the method robustness to noise is outstanding. Using this estimation as an initial guess for a bundle adjustment procedure we have seen that the minimization converges faster. We conclude that this method based on the orthographic model is well suited for the external calibration of perspective cameras with relatively long focal lengths outperforming the perspective-based pose estimation and being comparable in computational complexity to the linear computation of the fundamental matrix.

Image Credits

All images by the authors (license CC-BY-SA) except:

 Donald Simanek's Pages[4]

References

[1] D. W. G. Arthur, *Model formation with narrow-angle photography*, The Photogrammetric Record, 4 (1962), pp. 49–53. http://dx.doi.org/10.1111/j.1477-9730.1962.tb00325.x.

[2] G. Brdnik and OpenMVG authors, *Camera sensor size database*. https://github.com/openMVG/CameraSensorSizeDatabase.

[3] R. I. Hartley, *In defense of the eight-point algorithm*, IEEE Transactions on Pattern Analysis and Machine Intelligence, 19 (1997), pp. 580–593. http://dx.doi.org/10.1109/34.601246.

[4] R. I. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, Cambridge University Press, second ed., 2004. ISBN 0521540518.

[5] K. Levenberg, *A method for the solution of certain non-linear problems in least squares*, Quarterly of Applied Mathematics, 2 (1944), pp. 164–168. http://www.jstor.org/stable/43633451.

[6] L. Moisan, P. Moulon, and P. Monasse, *Fundamental Matrix of a Stereo Pair, with A Contrario Elimination of Outliers*, Image Processing On Line, 6 (2016), pp. 89–113. http://doi.org/10.5201/ipol.2016.147.

[7] P. Moulon, P. Monasse, and R. Marlet, *Adaptive structure from motion with a contrario model estimation*, in Proceedings of the 11th Asian Conference on Computer Vision (ACCV), Springer, 2012, pp. 257–270. http://doi.org/10.1007/978-3-642-37447-0_20.

[8] T. Ono and S. Hattori, *Fundamental principle of image orientation using orthogonal projection model*, International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, 34 (2002), pp. 194–199.

[9] M. P. Pierrot-Deseilligny, *MicMac, un logiciel pour la mise en correspondance automatique d'images dans le contexte gographique*, Bulletin d'Information Scientifique et Technique de l'IGN n, 77 (2007), p. 1.

[4] https://www.lockhaven.edu/~dsimanek/TTT-impossible/TTT-impossible.htm

[10] C. J. Poelman and T. Kanade, *A paraperspective factorization method for shape and motion recovery*, IEEE Transactions on Pattern Analysis and Machine Intelligence, 19 (1997), pp. 206–218. http://doi.org/10.1109/34.584098.

[11] I. Rey Otero and M. Delbracio, *Anatomy of the SIFT method*, Image Processing On Line, 4 (2014), pp. 370–396. https://doi.org/10.5201/ipol.2014.82.

[12] C. Stamatopoulos and C. S. Fraser, *Calibration of long focal length cameras in close range photogrammetry*, The Photogrammetric Record, 26 (2011), pp. 339–360. http://dx.doi.org/10.1111/j.1477-9730.2011.00648.x.

[13] C. Strecha, W. von Hansen, L. Van Gool, P. Fua, and U. Thoennessen, *On benchmarking camera calibration and multi-view stereo for high resolution imagery*, in 2008 IEEE Conference on Computer Vision and Pattern Recognition, June 2008, pp. 1–8. http://doi.org/10.1109/CVPR.2008.4587706.

[14] C. Tomasi and T. Kanade, *Shape and motion from image streams under orthography: a factorization method*, International Journal of Computer Vision, 9 (1992), pp. 137–154. http://doi.org/10.1007/BF00129684.

[15] B. Triggs, P. F. McLauchlan, R. I. Hartley, and A. W. Fitzgibbon, *Bundle adjustment — A modern synthesis*, in International Workshop on Vision Algorithms, Springer, 1999, pp. 298–372. https://doi.org/10.1007/3-540-44480-7_21.

[16] X. Yang and S. Fang, *Effect of field of view on the accuracy of camera calibration*, Optik - International Journal for Light and Electron Optics, 125 (2014), pp. 844 – 849. http://doi.org/10.1016/j.ijleo.2013.07.089.

[17] Z. Zhang, *3D reconstruction based on homography mapping*, Proceedings of ARPA96, (1996), pp. 1007–1012. http://ci.nii.ac.jp/naid/10030410351/en/.