SCIA'05 – In Proceedings of the Fourteenth Scandinavian Conference on Image Analysis, Joensuu, Finland, June 2005.

On Aligning Sets of Points Reconstructed From Uncalibrated Affine Cameras

A. Bartoli[1], H. Martinsson[2], F. Gaspard[2], and J.-M. Lavest[1]
Adrien.Bartoli@gmail.com – Hanna.Martinsson@cea.fr

[1] LASMEA (CNRS / UBP) – Clermont-Ferrand, France
[2] CEA LIST – Gif sur Yvette, France

Abstract. The reconstruction of rigid scenes from multiple images is a central topic in computer vision. Approaches merging partial 3D models in a hierarchical manner have proven the most effective to deal with large image sequences. One of the key building blocks of these hierarchical approaches is the alignment of two partial 3D models by computing a 3D transformation. This problem has been well-studied for the cases of 3D models obtained with calibrated or uncalibrated pinhole cameras.
We tackle the problem of aligning 3D models – sets of 3D points – obtained using uncalibrated affine cameras. This requires to estimate 3D affine transformations between the 3D models. We propose a factorization-based algorithm estimating simultaneously the aligning transformations and corrected points, exactly matching the estimated transformations, such that the reprojection error over all cameras is minimized.
We experimentally compare our algorithm to other methods using simulated and real data.

1 Introduction

Three dimensional reconstruction from multiple images of a rigid scene, often dubbed Structure-From-Motion, is one of the most studied problems in computer vision. The difficulties come from the fact that, using only feature correspondences, both the 3D structure of the scene and the cameras have to be computed. Most approaches rely on an initialisation phase optionally followed by self-calibration and bundle adjustment. Existing initialisation algorithms can be divided into three families, namely *batch*, *sequential* and *hierarchical* processes. Hierarchical processes [1] have proven the most successful for large image sequences. Indeed, batch processes such as the factorization algorithms [2] which reconstruct all features and cameras in a single computation step, do not easily handle occlusions, while sequential processes reconstruct each view on turn, may typically suffer from accumulation of the errors. Hierarchical processes merge partial 3D models obtained from sub-sequences, which allows to distribute the error over the sequence, and efficiently handle open and closed sequences. A key

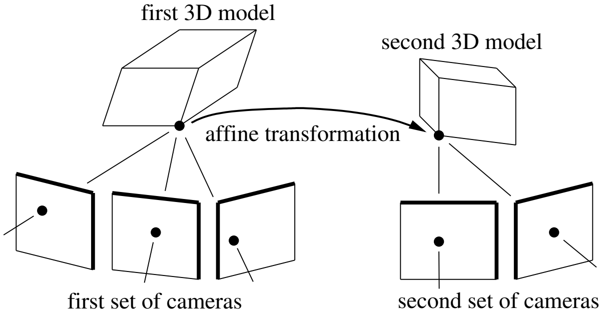


Fig. 1. This paper deals with the estimation of 3D affine transformations between two (or more) affine reconstructions obtained from uncalibrated affine cameras.

step of hierarchical processes is the fusion or the *alignment* of partial 3D models, by *computing 3D motion from 3D feature correspondences*. This problem has been extensively studied in the projective and metric cases.

We focus on the affine camera model, which is a reasonable approximation to the perspective camera model when the depth of the observed scene is small compared to the viewing distance. Partial 3D models obtained from sub-sequences, *i.e.* multiple subsets of cameras, are related by 3D affine transformations. We deal with the computation of such transformations from point correspondences, as illustrated on figure 1. We propose a Maximum Likelihood Estimator based on factorizing modified 3D point coordinates. We compute a 3D affine transformation and a set of 3D point correspondences which perfectly match, such that *the reprojection error in all sets of cameras is minimized*. The method can be embedded in a robust RANSAC-like [3] framework to deal with data sets containing outliers. It is intended to fit in hierarchical affine Structure-From-Motion processes of which the basic reconstruction block is, *e.g.* the affine factorization [2]. Our method, based on the new concept of *orthonormal bases*, requires a single Singular Value Decomposition (SVD) in the occlusion-free case.

This paper is organized as follow. We give our notation and preliminaries in §2. In §3, we review the factorization approach to uncalibrated affine Structure-From-Motion. Our alignment method is described in §4, while other methods are summarized in §5. Experimental results are reported in §6. Our conclusions are given in §7.

2 Notation and Preliminaries

Vectors are typeset using bold fonts, *e.g.* $\mathbf{x}$, and matrices using sans-serif, calligraphic and greek fonts, *e.g.* A, $\mathcal{Q}$ and Λ. We do not use homogeneous coordinates, *i.e.* image point coordinates are 2-vectors: $\mathbf{x}^\mathsf{T} = (x \ y)$, where $^\mathsf{T}$ is transposition. The different sets of cameras are indicated with primes, *e.g.* P_1, P'_1 and P''_1 are the first cameras of the three first camera sets. Index $i = 1 \ldots n$

is used for the cameras of a camera set and index $j = 1 \ldots m$ is used for the 3D points. The identity matrix is denoted I and the zero matrix and vector by $\mathsf{0}$ and $\mathbf{0}$. The Frobenius or $\mathcal{L}_2$ norm of a matrix A or a vector $\mathbf{x}$ are respectively denoted $\|\mathsf{A}\|$ and $\|\mathbf{x}\|$. The mean vector of a set of vectors, say $\{\mathbf{Q}_j\}$, is denoted $\bar{\mathbf{Q}}$. The Moore-Penrose pseudoinverse of matrix A is denoted $\mathsf{A}^{\dagger}$.

Let $\mathbf{Q}_j$ be a 3-vector and $\mathbf{x}_{ij}$ a 2-vector representing respectively a 3D and an image point. The uncalibrated affine camera is modeled by a (2×3) matrix P_i and a (2×1) translation vector $\mathbf{t}_i$, giving the projection equation:

$$\mathbf{x}_{ij} = \mathsf{P}_i \mathbf{Q}_j + \mathbf{t}_i. \tag{1}$$

Calligraphic fonts are used for the measurement matrices: *e.g.* $\mathcal{X}_{(2n \times m)}$ is made with measured point coordinates $\mathbf{x}_{ij}$ and $\mathcal{X} = \begin{pmatrix} \mathcal{Y}_1 & \cdots & \mathcal{Y}_m \end{pmatrix}$, where $\mathcal{Y}_j$ contains all the measured image coordinates for the j-th point. The so-called $(2n \times 3)$ 'joint projection' and $(3 \times m)$ 'joint structure' matrices are defined by $\mathcal{P}^{\mathsf{T}} = \begin{pmatrix} \mathsf{P}_1^{\mathsf{T}} & \cdots & \mathsf{P}_n^{\mathsf{T}} \end{pmatrix}$ and $\mathcal{Q} = \begin{pmatrix} \mathbf{Q}_1 & \cdots & \mathbf{Q}_m \end{pmatrix}$. We assume that the noise on image point positions is i.i.d., centred Gaussian. Under these hypotheses minimizing the reprojection error yields Maximum Likelihood Estimates.

3 Structure-From-Motion Using Factorization

Given a set of point matches $\{\mathbf{x}_{ij}\}$, the factorization algorithm is employed to recover all cameras $\{\hat{\mathsf{P}}_i, \hat{\mathbf{t}}_i\}$ and 3D points $\{\hat{\mathbf{Q}}_j\}$ at once [2]. Under the aforementioned hypotheses on the noise distribution, this algorithm computes Maximum Likelihood Estimates by minimizing the reprojection error:

$$\mathcal{R}^2(\mathcal{P}, \mathcal{Q}, \{\mathbf{t}_i\}) = \frac{1}{nm} \sum_{i=1}^{n} \sum_{j=1}^{m} d^2(\mathbf{x}_{ij}, \mathsf{P}_i \mathbf{Q}_j + \mathbf{t}_i), \tag{2}$$

where $d(\mathbf{x}, \mathbf{y}) = \|\mathbf{x} - \mathbf{y}\|$ is the Euclidean distance between $\mathbf{x}$ and $\mathbf{y}$. The problem is thus formulated as $\min_{\hat{\mathcal{P}}, \hat{\mathcal{Q}}, \{\hat{\mathbf{t}}_i\}} \mathcal{R}^2(\hat{\mathcal{P}}, \hat{\mathcal{Q}}, \{\hat{\mathbf{t}}_i\})$.

Step 1: Computing the translation. Given the uncalibrated affine projection (1), the first step of the algorithm is to compute the translation $\hat{\mathbf{t}}_i$ of each camera in order to cancel it out from the projection equation. This is achieved by nullifying the partial derivatives of the reprojection error (2) with respect to $\hat{\mathbf{t}}_i$: $\frac{\partial \mathcal{R}^2}{\partial \hat{\mathbf{t}}_i} = 0$. A short calculation shows that if we fix the arbitrary centroid of the 3D points to the origin, then $\hat{\mathbf{t}}_i = \bar{\mathbf{x}}_i$. Each set of image points is therefore centred on its centroid, *i.e.* $\mathbf{x}_{ij} \leftarrow \mathbf{x}_{ij} - \bar{\mathbf{x}}_i$, to obtain *centred coordinates*. Henceforth, we work in centred coordinates which allows to write the *centred projection equation* $\mathbf{x}_{ij} = \mathsf{P}_i \mathbf{Q}_j$ from (1).

Step 2: Factorizing. We rewrite $\mathcal{R}^2(\mathcal{P}, \mathcal{Q}) = \frac{1}{nm} \sum_{i=1}^{n} \sum_{j=1}^{m} d^2(\mathbf{x}_{ij}, \mathsf{P}_i \mathbf{Q}_j)$ the reprojection error. The problem is thus reformulated as $\min_{\hat{\mathcal{P}}, \hat{\mathcal{Q}}} \mathcal{R}^2(\hat{\mathcal{P}}, \hat{\mathcal{Q}})$. The reprojection error can be rewritten by gathering the terms using the measurement, the 'joint projection' and the 'joint structure' matrices as $\mathcal{R}^2(\mathcal{P},\mathcal{Q}) \propto \|\mathcal{X} - \mathcal{P}\mathcal{Q}\|^2$, and the problem is solved by computing the Singular Value Decomposition (SVD) [4] of matrix $\mathcal{X}$: $\mathcal{X}_{2n\times m} = \mathsf{U}_{2n\times m}\Sigma_{m\times m}\mathsf{V}^\mathsf{T}_{m\times m}$, where U and V are orthonormal matrices and Σ is diagonal and contains the singular values of $\mathcal{X}$. Let $\Sigma = \Sigma_u\Sigma_v$ be any decomposition of matrix Σ, *e.g.* $\Sigma_u = \Sigma_v = \sqrt{\Sigma}$. The motion and structure are obtained by, loosely speaking, 'truncating' the decomposition or nullifying all but the 3 first singular values, which leads to $\mathcal{P} = \psi(\mathsf{U}\Sigma_u)$ and $\mathcal{Q} = \psi^\mathsf{T}(\mathsf{V}\Sigma_v^\mathsf{T})$, where $\psi(\mathsf{W})$ returns the matrix formed with the 3 leading columns of matrix W. Note that the alternative solution $\mathcal{P} = \psi(\mathsf{U})$ and $\mathcal{Q} = \psi^\mathsf{T}(\mathsf{V}\Sigma)$ has the property $\mathcal{P}^\mathsf{T}\mathcal{P} = \mathrm{I}$ which is useful for our alignment method, see §4. The 3D model is obtained only up to a global affine transformation. Indeed, let B be a (3×3) invertible matrix: $\tilde{\mathcal{P}} = \hat{\mathcal{P}}\mathsf{B}$ and $\tilde{\mathcal{Q}} = \mathsf{B}^{-1}\hat{\mathcal{Q}}$ give the same reprojection error as $\mathcal{P}$ and $\mathcal{Q}$ since $\mathcal{R}^2(\tilde{\mathcal{P}},\tilde{\mathcal{Q}}) = \|\mathcal{X} - \tilde{\mathcal{P}}\tilde{\mathcal{Q}}\| = \|\mathcal{X} - \hat{\mathcal{P}}\mathsf{B}\mathsf{B}^{-1}\hat{\mathcal{Q}}\|^2 = \|\mathcal{X} - \mathcal{P}\mathcal{Q}\|^2 = \mathcal{R}^2(\mathcal{P},\mathcal{Q})$. As presented above, the factorization algorithm do not handle occlusions. Though some algorithms have been proposed, see *e.g.* [5], they are not appropriate for Structure-From-Motion from large image sequences.

4 Alignment of 3D Affine Reconstructions

We formally state the alignment problem in the two camera set case and present our algorithm, dubbed 'FactMLE'. Its extension to the multiple camera set case is trivial and is omitted.

4.1 Problem Statement

Consider two sets of cameras $\{(\mathsf{P}_i, \mathbf{t}_i)\}_{i=1}^n$ and $\{(\mathsf{P}'_i, \mathbf{t}'_i)\}_{i=1}^{n'}$ and associated structures[3] $\{\mathbf{Q}_j \leftrightarrow \mathbf{Q}'_j\}_{j=1}^m$ obtained by reconstructing a rigid scene using *e.g.* the above-described factorization algorithm. The reprojection error over these two sets is given by:

$$\mathcal{C}^2(\mathcal{Q},\mathcal{Q}') = \frac{1}{2nm}\left(\mathcal{R}^2(\mathcal{P},\mathcal{Q},\{\mathbf{t}_i\}) + \mathcal{R}'^2(\mathcal{P}',\mathcal{Q}',\{\mathbf{t}'_i\})\right). \tag{3}$$

Let $(\hat{\mathsf{A}}, \hat{\mathbf{t}})$ represent the aligning (3×3) affine transformation. The Maximum Likelihood Estimator is formulated by:

$$\min_{\hat{\mathcal{Q}},\hat{\mathcal{Q}}'} \mathcal{C}^2(\hat{\mathcal{Q}},\hat{\mathcal{Q}}') \qquad \text{s.t.} \qquad \hat{\mathbf{Q}}'_j = \hat{\mathsf{A}}\hat{\mathbf{Q}}_j + \hat{\mathbf{t}}. \tag{4}$$

4.2 A Factorization-Based Algorithm

Our method to solve problem (4) uses a three-step factorization strategy. We describe it in the occlusion-free case only. An iterative extension for the missing data case will be proposed in a forthcoming paper.

[3] Without loss of generality, we assume the same number of points to be present in the two reconstructions since only the correspondences are used for alignment.

Step 1: Orthonormalizing. We propose the important concept of *orthonormal bases.* We define a reconstruction to be in an orthonormal basis if the joint projection matrix is column-orthonormal. Given a joint projection matrix $\mathcal{P}$, one can find a 3D affine tranformation $\mathsf{N}_{(3\times 3)}$ such that $\mathcal{P}\mathsf{N}$ is column-orthonormal, *i.e.* such that $\mathsf{N}^\mathsf{T}\mathcal{P}^\mathsf{T}\mathcal{P}\mathsf{N} = \mathrm{I}_{(3\times 3)}$. We call N an *orthonormalizing transformation.* The set of orthonormalizing tranformations is 3-dimensional since for any 3D rotation matrix U, NU still is an orthonormalizing transformation for $\mathcal{P}$. We use the QR decomposition $\mathcal{P} = \mathsf{QR}$, see *e.g.* [4], giving an upper triangular orthonormalizing transformation $\mathsf{N} = \mathsf{R}^{-1}$. Other choices are possible for computing an N, *e.g.* if $\mathcal{P} = \mathsf{U}\Sigma\mathsf{V}^\mathsf{T}$ is an SVD of $\mathcal{P}$, then $\mathsf{N} = \mathsf{V}\Sigma^{-1}$ has the required property. Henceforth, we assume that all 3D models are expressed in orthonormal bases: $\mathcal{P} \leftarrow \mathcal{P}\mathsf{N}$, $\mathcal{P}' \leftarrow \mathcal{P}'\mathsf{N}'$, $\mathcal{Q} \leftarrow \mathsf{N}^{-1}\mathcal{Q}$ and $\mathcal{Q}' \leftarrow \mathsf{N}'^{-1}\mathcal{Q}'$. An interesting property of orthonormal bases is that $\mathcal{P}^\dagger = \mathcal{P}^\mathsf{T}$. Hence, triangulating points in these bases is simply done by $\mathcal{Q} = \mathcal{P}^\mathsf{T}\mathcal{X}$.

Note that the matrix $\mathcal{P}$ computed by factorization, see §3, may already satisfy $\mathcal{P}^\mathsf{T}\mathcal{P} = \mathrm{I}$. However, if at least one of the cameras is not used for the alignment, *e.g.* if none of the 3D point correspondences project in this camera, or if the cameras come as the result of the alignment of partial 3D models, then $\mathcal{P}$ will *not* satisfy $\mathcal{P}^\mathsf{T}\mathcal{P} = \mathrm{I}$, thus requiring the orthonormalization step.

Step 2: Eliminating the translation. The translation part of the sought-after transformation can not be computed directly, but can be eliminated from the equations. First, centre the image points to eliminate the translation part of the cameras: $\mathbf{x}_{ij} \leftarrow \mathbf{x}_{ij} - \mathbf{t}_i$ and $\mathbf{x}'_{ij} \leftarrow \mathbf{x}'_{ij} - \mathbf{t}'_i$. Second, consider that the partial derivatives of the reprojection error (3) with respect to $\hat{\mathbf{t}}$ must vanish: $\frac{\partial \mathcal{C}^2}{\partial \hat{\mathbf{t}}} = 0$. By using the constraint $\hat{\mathbf{Q}}'_j = \hat{\mathsf{A}}\hat{\mathbf{Q}}_j + \hat{\mathbf{t}}$ from equation (3) and expanding: $\sum_{i=1}^{n'}\sum_{j=1}^{m}\left(\mathsf{P}'^\mathsf{T}_i\mathsf{P}'_i\hat{\mathbf{t}} - \mathsf{P}'^\mathsf{T}_i\mathbf{x}'_{ij} + \mathsf{P}'^\mathsf{T}_i\mathsf{P}'_i\hat{\mathsf{A}}\hat{\mathbf{Q}}_j\right) = m\mathcal{P}'^\mathsf{T}\mathcal{P}'\hat{\mathbf{t}} - m\mathcal{P}'^\mathsf{T}\bar{\mathcal{Y}}' + m\mathcal{P}'^\mathsf{T}\mathcal{P}'\hat{\mathsf{A}}\bar{\hat{\mathcal{Q}}} = 0$, which leaves us with $\hat{\mathbf{t}} = \left(\mathcal{P}'^\mathsf{T}\mathcal{P}'\right)^{-1}(\mathcal{P}'^\mathsf{T}\bar{\mathcal{Y}}' - \mathcal{P}'^\mathsf{T}\mathcal{P}'\hat{\mathsf{A}}\bar{\hat{\mathcal{Q}}})$ that further simplifies to $\hat{\mathbf{t}} = \mathcal{P}'^\dagger\bar{\mathcal{Y}}' - \hat{\mathsf{A}}\bar{\hat{\mathcal{Q}}}$ and, thanks to the orthonormal basis property $\mathcal{P}'^\dagger = \mathcal{P}'^\mathsf{T}$, we get:

$$\hat{\mathbf{t}} = \mathcal{P}'^\mathsf{T}\bar{\mathcal{Y}}' - \hat{\mathsf{A}}\bar{\hat{\mathcal{Q}}}, \tag{5}$$

Note that if the same entire sets of reconstructed points are used for the alignment, then we directly obtain $\hat{\mathbf{t}} = \mathbf{0}$ since $\bar{\mathcal{Y}}' = \mathbf{0}$ and $\bar{\hat{\mathcal{Q}}} = \mathbf{0}$. This is rarely the case in practice, especially if the alignment is used to merge partial 3D models.

Third, consider that the m partial derivatives of the reprojection error (3) with respect to each $\hat{\mathbf{Q}}_j$ must vanish as well: $\frac{\partial \mathcal{C}^2}{\partial \hat{\mathbf{Q}}_j} = 0$, and expand as above: $\mathcal{P}^\mathsf{T}\mathcal{P}\hat{\mathbf{Q}}_j - \mathcal{P}^\mathsf{T}\mathcal{Y}_j + \hat{\mathsf{A}}^\mathsf{T}\mathcal{P}'^\mathsf{T}\mathcal{P}'\hat{\mathsf{A}}\hat{\mathbf{Q}}_j - \hat{\mathsf{A}}^\mathsf{T}\mathcal{P}'^\mathsf{T}\mathcal{Y}'_j + \hat{\mathsf{A}}^\mathsf{T}\mathcal{P}'^\mathsf{T}\mathcal{P}'\hat{\mathbf{t}} = 0$. The sum over j of all these derivatives also vanishes: $\left(\forall j, \frac{\partial \mathcal{C}^2}{\partial \mathbf{Q}_j} = 0\right) \Rightarrow \left(\sum_{j=1}^{m}\frac{\partial \mathcal{C}^2}{\partial \mathbf{Q}_j} = 0\right)$, giving $\mathcal{P}^\mathsf{T}\mathcal{P}\bar{\hat{\mathcal{Q}}} - \mathcal{P}^\mathsf{T}\bar{\mathcal{Y}} + \hat{\mathsf{A}}^\mathsf{T}\mathcal{P}'^\mathsf{T}\mathcal{P}'\hat{\mathsf{A}}\bar{\hat{\mathcal{Q}}} - \hat{\mathsf{A}}^\mathsf{T}\mathcal{P}'^\mathsf{T}\bar{\mathcal{Y}}' + \hat{\mathsf{A}}^\mathsf{T}\mathcal{P}'^\mathsf{T}\mathcal{P}'\hat{\mathbf{t}} = 0$. By replacing $\hat{\mathbf{t}}$ by its expression (5), and after some minor algebraic manipulations, we obtain $\mathcal{P}^\mathsf{T}\mathcal{P}\bar{\hat{\mathcal{Q}}} - \mathcal{P}^\mathsf{T}\bar{\mathcal{Y}} = 0$ and $\bar{\hat{\mathcal{Q}}} = \mathcal{P}^\dagger\bar{\mathcal{Y}}$. By substituting in equation (5) and using the orthonormal basis property $\mathcal{P}^{\dagger} = \mathcal{P}^{\mathsf{T}}$, we get:

$$\hat{\mathbf{t}} = \mathcal{P}'^{\mathsf{T}}\bar{\mathcal{Y}}' - \hat{\mathsf{A}}\mathcal{P}^{\mathsf{T}}\bar{\mathcal{Y}}. \tag{6}$$

It is common in factorization methods to centre the data with respect to their centroid to cancel the translation part of the transformation. Equation (6) means that, according to the reprojection error criterion, the data must be centred with respect to the *reconstructed centroid* of the image points, not with respect to the actual 3D centroid.

Obviously, if the 3D models have been obtained by the factorization method of §3, then the centroid of the 3D points corresponds to the reconstructed centroid, *i.e.* $\bar{\mathbf{Q}} = \mathcal{P}^{\mathsf{T}}\bar{\mathcal{Y}}$ and $\bar{\mathbf{Q}}' = \mathcal{P}'^{\mathsf{T}}\bar{\mathcal{Y}}'$, provided that the same sets of views are used for reconstruction and alignment.

To summarize, we cancel the translation part out of the sought-after transformation by translating the reconstructions and the image points by $\mathbf{Q}_j \leftarrow \mathbf{Q}_j - \mathcal{P}^{\mathsf{T}}\bar{\mathcal{Y}}$ and $\mathbf{x}_{ij} \leftarrow \mathbf{x}_{ij} - \mathsf{P}_i\mathcal{P}^{\mathsf{T}}\bar{\mathcal{Y}}$, and similarly for the second image set. The reprojection error (3) is rewritten:

$$\mathcal{C}^2(\mathcal{Q}, \mathcal{Q}') = \frac{1}{2nm}\left(\|\mathcal{X} - \mathcal{P}\mathcal{Q}\|^2 + \|\mathcal{X}' - \mathcal{P}'\mathcal{Q}'\|^2\right), \tag{7}$$

and problem (4) is reformulated as $\min_{\hat{\mathcal{Q}},\hat{\mathcal{Q}}'} \mathcal{C}^2(\hat{\mathcal{Q}}, \hat{\mathcal{Q}}')$ s.t. $\hat{\mathbf{Q}}'_j = \hat{\mathsf{A}}\hat{\mathbf{Q}}_j$.

Step 3: Factorizing. Thanks to the orthonormal basis property $\mathcal{P}^{\mathsf{T}}\mathcal{P} = \mathrm{I}$, and since for any column-orthonormal matrix $\mathcal{A}$, $\|\mathcal{A}\mathbf{x}\| = \|\mathbf{x}\|$, we can rewrite the reprojection error on a single set of cameras as $\mathcal{R}^2(\mathcal{P}, \mathcal{Q}) \propto \|\mathcal{X} - \mathcal{P}\mathcal{Q}\|^2 = \|\mathcal{P}^{\mathsf{T}}\mathcal{X} - \mathcal{Q}\|^2$. This allows to rewrite the reprojection error (7) as:

$$\mathcal{C}^2(\hat{\mathcal{Q}}, \hat{\mathcal{Q}}') \propto \|\mathcal{P}^{\mathsf{T}}\mathcal{X} - \hat{\mathcal{Q}}\|^2 + \|\mathcal{P}'^{\mathsf{T}}\mathcal{X}' - \hat{\mathcal{Q}}'\|^2 \quad = \quad \| \underbrace{\begin{pmatrix} \mathcal{P}^{\mathsf{T}}\mathcal{X} \\ \mathcal{P}'^{\mathsf{T}}\mathcal{X}' \end{pmatrix}}_{\mathsf{\Lambda}} - \underbrace{\begin{pmatrix} \hat{\mathcal{Q}} \\ \hat{\mathcal{Q}}' \end{pmatrix}}_{\mathsf{\Delta}} \|^2.$$

By introducing the constraint $\hat{\mathcal{Q}}' = \hat{\mathsf{A}}\hat{\mathcal{Q}}$ and, as in §3, an unknown global affine transformation B:

$$\mathsf{\Delta} \quad = \quad \begin{pmatrix} \mathrm{I} \\ \hat{\mathsf{A}} \end{pmatrix} \mathsf{B}\mathsf{B}^{-1}\hat{\mathcal{Q}} \quad = \quad \underbrace{\begin{pmatrix} \mathsf{B} \\ \hat{\mathsf{A}}\mathsf{B} \end{pmatrix}}_{\tilde{\mathcal{M}}} \underbrace{\mathsf{B}^{-1}\mathcal{Q}}_{\tilde{\mathcal{Q}}}.$$

The problem is reformulated as $\min_{\tilde{\mathcal{M}},\tilde{\mathcal{Q}}} \|\mathsf{\Lambda} - \tilde{\mathcal{M}}\tilde{\mathcal{Q}}\|^2$. A solution is given by SVD of matrix $\mathsf{\Lambda}$: $\mathsf{\Lambda}_{(6\times m)} = \mathsf{U}_{(6\times 6)}\mathsf{\Sigma}_{(6\times 6)}\mathsf{V}^{\mathsf{T}}_{(6\times m)}$. As in §3, let $\mathsf{\Sigma} = \mathsf{\Sigma}_u\mathsf{\Sigma}_v$ be any decomposition of matrix $\mathsf{\Sigma}$. We obtain $\tilde{\mathcal{M}} = \psi(\mathsf{U}\mathsf{\Sigma}_u)$ and $\tilde{\mathcal{Q}} = \psi^{\mathsf{T}}(\mathsf{V}\mathsf{\Sigma}_v^{\mathsf{T}})$. Using the partitioning $\tilde{\mathcal{M}}^{\mathsf{T}} = \begin{pmatrix} \tilde{\mathsf{M}}^{\mathsf{T}} & \tilde{\mathsf{M}}'^{\mathsf{T}} \end{pmatrix}$, we get $\mathsf{B} = \tilde{\mathsf{M}}$, $\hat{\mathsf{A}} = \tilde{\mathsf{M}}'\mathsf{B}^{-1}$ and $\hat{\mathcal{Q}} = \mathsf{B}\tilde{\mathcal{Q}}$. Obviously, one needs to undo the effect of the orthonormalizing transformations: $\hat{\mathsf{A}} \leftarrow \mathsf{N}'\hat{\mathsf{A}}\mathsf{N}^{-1}$ and $\hat{\mathcal{Q}} \leftarrow \mathsf{N}\hat{\mathcal{Q}}$. A minimal $m \geq 4$ point correspondences is required.

Note that it is possible to solve the problem without using the orthonormalizing transformations. This solution requires however to compute the SVD of a $(2(n+n') \times m)$ matrix, made by stacking the measurement matrices $\mathcal{X}$ and $\mathcal{X}'$, and is therefore much more computationally expensive than the algorithm above, and may be intractable for large sets of cameras and points.

5 Other Algorithms

We briefly describe two other alignment algorithms. They do not yield Maximum Likelihood Estimates under the previously-mentioned hypotheses on the noise distribution. They rely on 3D measurements and naturally handle missing data.

5.1 Minimizing the Non-Symmetric Transfer Error

This algorithm, dubbed 'TRERROR', is specific to the two camera set case. It is based on minimizing a non-symmetric 3D transfer error $\mathcal{E}(\hat{\mathsf{A}})$: $\min_{\hat{\mathsf{A}},\hat{\mathbf{t}}} \mathcal{E}^2(\hat{\mathsf{A}},\hat{\mathbf{t}})$ with $\mathcal{E}^2(\hat{\mathsf{A}}) = \frac{1}{m}\sum_{j=1}^{m} \|\mathbf{Q}'_j - \hat{\mathsf{A}}\mathbf{Q}_j - \hat{\mathbf{t}}\|^2$. Differentiating $\mathcal{E}^2$ with respect to $\hat{\mathbf{t}}$ and nullifying the result yields $\hat{\mathbf{t}} = \hat{\mathbf{Q}}' - \hat{\mathsf{A}}\hat{\mathbf{Q}}$. Henceforth, we assume that the translation has been eliminated by translating each 3D point set on its centroid. By rewriting the error function as $\mathcal{E}^2(\hat{\mathsf{A}}) \propto \|\mathcal{Q}' - \hat{\mathsf{A}}\mathcal{Q}\|^2$ and applying standard linear least-squares, one obtains the solution $\hat{\mathsf{A}} = \mathcal{Q}'\mathcal{Q}^\dagger$.

5.2 Direct 3D Factorization

This algorithm, dubbed 'FACT3D', is based on directly factorizing the 3D reconstructed points. It is not restricted to the two camera set case, but for simplicity, we only describe this case. Generalization to multiple camera sets is trivial. The algorithm computes the aligning transformation $(\hat{\mathsf{A}},\hat{\mathbf{t}})$ and perfectly corresponding points $\{\hat{\mathbf{Q}}_j \leftrightarrow \hat{\mathbf{Q}}'_j\}$. The reconstructed cameras are not taken into account by this algorithm, which entirely relies on 3D measures on the reconstructed points. This algorithm is equivalent to the proposed FACTMLE under certain conditions.

The problem is stated by $\min_{\hat{\mathcal{Q}},\hat{\mathcal{Q}}'} \mathcal{D}^2(\hat{\mathcal{Q}},\hat{\mathcal{Q}}')$ s.t. $\hat{\mathbf{Q}}'_j = \hat{\mathsf{A}}\hat{\mathbf{Q}}_j + \hat{\mathbf{t}}$, here the 3D error function employed is defined by $\mathcal{D}^2(\hat{\mathcal{Q}},\hat{\mathcal{Q}}') = \frac{1}{2m}\left(\|\mathcal{Q} - \hat{\mathcal{Q}}\|^2 + \|\mathcal{Q}' - \hat{\mathcal{Q}}'\|^2\right)$. Minimizing this error function means that if the noise *on the 3D point coordinates* were Gaussian, centred and i.i.d., which is *not* the case with our actual hypotheses (the noise distribution in 3D depends on the noise distribution in the images and the reconstruction method – hence it is not a priori Gaussian), then this algorithm would yield the Maximum Likelihood Estimate.

Step 1: Computing the translation. By nullifying the partial derivatives of the error function $\mathcal{D}^2$ with respect to $\hat{\mathbf{t}}$ and with respect to the $\hat{\mathbf{Q}}_j$, and substituting the latter expressions into the former one, we obtain $\hat{\mathbf{t}} = \bar{\mathbf{Q}}' - \hat{\mathsf{A}}\bar{\mathbf{Q}}$. This equation means that, as in most factorization methods, cancelling the translation part out according to the error function $\mathcal{D}$ is done by centring each set of 3D points on its actual centroid: $\hat{\mathbf{Q}}_j \leftarrow \hat{\mathbf{Q}}_j - \bar{\mathbf{Q}}$ and $\hat{\mathbf{Q}}'_j \leftarrow \hat{\mathbf{Q}}'_j - \bar{\mathbf{Q}}'$. Henceforth, we assume to work in centred coordinates. The problem is rewritten as $\min_{\hat{\mathcal{Q}},\hat{\mathcal{Q}}'} \mathcal{D}^2(\hat{\mathcal{Q}}, \hat{\mathcal{Q}}')$ s.t. $\hat{\mathbf{Q}}'_j = \hat{\mathsf{A}}\hat{\mathbf{Q}}_j$.

Step 2: Factorizing. Following the approach in §4.2, we rewrite $\mathcal{D}$ as:

$$\mathcal{D}^2(\hat{\mathcal{Q}}, \hat{\mathcal{Q}}') \;\propto\; \| \begin{pmatrix} \mathcal{Q} \\ \mathcal{Q}' \end{pmatrix} - \begin{pmatrix} \hat{\mathcal{Q}} \\ \hat{\mathcal{Q}}' \end{pmatrix} \|^2 \quad = \quad \| \underbrace{\begin{pmatrix} \mathcal{Q} \\ \mathcal{Q}' \end{pmatrix}}_{\Lambda} - \underbrace{\begin{pmatrix} \mathsf{B} \\ \mathsf{AB} \end{pmatrix}}_{\tilde{\mathcal{M}}} \underbrace{\mathsf{B}^{-1}\hat{\mathcal{Q}}}_{\tilde{\mathcal{Q}}} \|^2.$$

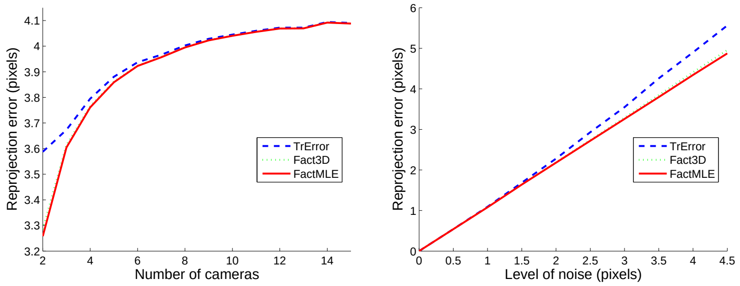
Using SVD of matrix $\Lambda = \mathsf{U}\Sigma\mathsf{V}^\mathsf{T}$, we obtain $\tilde{\mathcal{M}} = \psi(\mathsf{U}\Sigma_u)$ and $\tilde{\mathcal{Q}} = \psi^\mathsf{T}(\mathsf{V}\Sigma_v^\mathsf{T})$. By partitioning $\tilde{\mathcal{M}}^\mathsf{T} = \left(\tilde{\mathsf{M}}^\mathsf{T} \; \tilde{\mathsf{M}}'^\mathsf{T}\right)$, we get $\mathsf{B} = \tilde{\mathsf{M}}$, $\hat{\mathsf{A}} = \tilde{\mathsf{M}}'\mathsf{B}^{-1}$ and $\hat{\mathcal{Q}} = \mathsf{B}\tilde{\mathcal{Q}}$.

6 Experimental Evaluation

6.1 Simulated Data

We generated m 3D points and two sets of n weak perspective cameras: $\mathsf{P}_i = \mathsf{A}_i\bar{\mathsf{R}}_i$, where A_i is the internal calibration matrix $\mathsf{A}_i = k_i\mathrm{diag}(\tau_i, 1)$, $\bar{\mathsf{R}}_i$ a (2×3), truncated, 3D rotation matrix and $\mathbf{t}_i$ is a 2-vector. The scale factor k_i models the average depth of the object and the focal length of the camera, and τ models the aspect ratio that we choose very close to 1. The 3D points are chosen from a uniform distribution inside a thin rectangular parallelepiped with dimensions $1 \times 1 \times (1 - d)$, and the internal camera scale factors k_i are chosen so that the points are uniformly spread in 400×400 pixel images. We use m points to perform Structure-From-Motion on each camera set and m_c points for the alignment. A gaussian noise with zero mean and standard deviation σ is added in the images. We define the overlap ratio of the two camera sets to be $\theta = m_c/m$, *i.e.* for $\theta = 1$ all points are seen in all views, while for $\theta = 0$, the two sets of cameras do not share corresponding points. The comparison of the algorithms being based on the reprojection error, the point clouds used to compute it need to be re-estimated so that this error is minimized, given an estimated transformation. This must be done for TrError and Fact3D.

The default setting is: $n = 2$ views, $m = 250$ points, $\theta = 0.2$ (*i.e.* a 20% overlap and $m_c = 50$ points common to the two 3D models), $\sigma = 3.0$ pixels, $d = 0.95$ (flat 3D scene) and $a = 1$ (perfectly affine projections). Figure 2 shows the reprojection error averaged over 500 simulations for varying the number n of cameras and the level of noise σ. Whereas FactMLE and Fact3D have similar behaviors, TrError is less robust with regard to both of these parameters. Other experiments concern varying the overlap ratio and the number of points m_c, the former from 10% to 100% and the latter from 4 to 60, corresponding respectively to $m = 20$ and $m = 300$ points. For small values of m_c, FactMLE and Fact3D yield very similar errors, whereas for higher numbers of points, the



(a) (b)

Fig. 2. Reprojection error versus (a) the number n of cameras and (b) the noise σ.

difference gets larger We see that for $m_c > 20$, the number of points has a much smaller influence on the errors. However, neither the error nor the difference between the methods seems to change when the overlap changes and we thus conclude that the overlap does not much influence the alignment algorithms, when the number of points is high enough. In all cases, method TrError performs very badly compared to the two other ones. Finally, the flatness of the simulated data d is varied from 0 to 1, that is from a cube to a plane. The flatness of the scene does not change the result of the alignment, except for very flat scenes where TrError turns out to be unstable. This result was expected since planar scenes are singular for the computation of a 3D affine transformation. The result means that TrError is much more sensitive to noise than the two other methods. We conclude that although FactMLE consistently outperforms the other two algorithms, it is in critical situation that the difference seems to be the most important. In particular, TrError is less robust with regard to the number of cameras, the noise and the flatness of the scene.

6.2 Real Data

We applied the algorithms to real image sequences. For one of them, the 'cylinder head' sequence, we show results. The video camera is not calibrated, that is, the internal parameters are unknown but constant throughout the video footage.

The images are shown on figure 3 with the $m_c = 13$ original, manually entered, and reprojected points after we applied the FactMLE method. The pictures were taken with a camera with 12 mm focal length, at a distance of approximately 60 cm of the object, which is 40 cm long. The reprojection errors we obtained are: 3.7683 pixels for FactMLE, 3.7692 pixels for Fact3D and 3.7764 pixels for TrError.

The 'statuette sequence' is made of 4 images with $m_c = 14$ points lying very close to a plane, and $m = m' = 25$, giving a 56% overlap. The points were manually entered. The reprojection errors we obtained are: 2.8402 pixels for FactMLE, 2.8415 pixels for Fact3D and 2.8446 pixels for TrError.

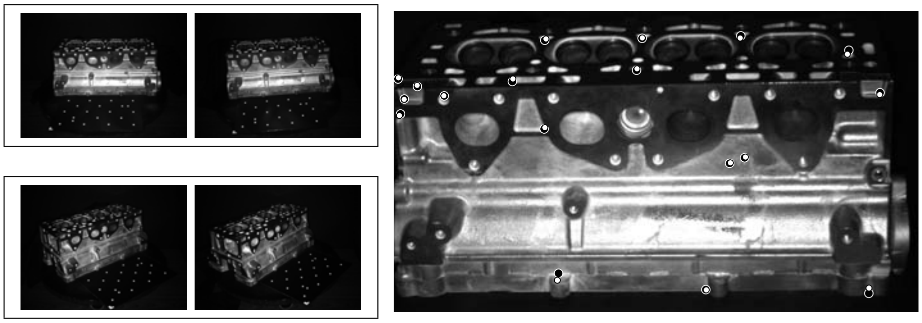


Fig. 3. (left) Both sets of images of the cylinder head sequence and (right) closeup overlaid with the original points (black) and reprojected points (white) from FactMLE.

The 'book sequence' consists of 5 images with $m_c = 196$, $m = 628$ and $m' = 634$ points given by an automatic correlation-based tracker, giving a 31% overlap. The reprojection errors we obtained are: 1.8961 pixels for FactMLE, 1.9737 pixels for Fact3D and 2.1690 pixels for TrError.

In accordance with the results on simulated data, we observe that in critical situations, FactMLE outperforms the other two methods. The reprojection errors of the order of a few pixels indicate that the data are well-modeled.

7 Conclusions

We presented a method to compute the Maximum Likelihood Estimate of 3D affine transformations, under standard hypotheses on the noise distribution, aligning sets of 3D points obtained from uncalibrated affine cameras.

Future work could be devoted to the experimental validation of the method in the missing data case, and the incorporation of other types of features, namely line, planar curve and plane correspondences.

References

1. Fitzgibbon, A., Zisserman, A.: Automatic camera recovery for closed or open image sequences. In: European Conference on Computer Vision. (1998) 311–326
2. Tomasi, C., Kanade, T.: Shape and motion from image streams under orthography: A factorization method. International Journal of Computer Vision **9** (1992) 137–154
3. Fischler, M., Bolles, R.: Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. Graphics and Image Processing **24** (1981) 381 – 395
4. Golub, G., van Loan, C.: Matrix Computation. The Johns Hopkins University Press, Baltimore (1989)
5. Jacobs, D.: Linear fitting with missing data: Applications to structure-from-motion and to characterizing intensity images. In: Computer Vision and Pattern Recognition (1997) 206–212