# A global camera network calibration method with Linear Programming

Jérôme Courchay Arnak Dalalyan Renaud Keriven
IMAGINE Université Paris-Est, LIGM/ENPC/CSTB
http://imagine.enpc.fr

Peter Sturm
INRIA
http://www.inrialpes.fr

## Abstract

*We propose a novel global calibration method for a network of cameras. Given a set of unknown cameras linked by epipolar geometry, we transform it into a graph of camera triplets. Underlying this new graph is a set of trifocal tensors. Based on (i) a linear computation of the trifocal tensor given two of the three related fundamental matrices, and (ii) considering a maximal tree embedded in our graph, we first design a way to globally recover the cameras. Then, we observe that considering the whole graph consists in simply adding constraints that can be easily linearized. A global estimation of the cameras thus boils down to solving a linear program. Numerical experiments show the ability of our method to recover accurate geometries, without any incremental process, and dealing naturally with loops in the network of cameras. Moreover, the number of unknown parameters is limited: we only consider the cameras (the* motion*), not the three-dimensional points (the* structure*).*

## 1. Introduction

Camera calibration from images has always been a central issue in Computer Vision. The success of textbooks like [6, 4] attests this interest, although misleads somehow the community toward the idea that most of the problem have been solved. Indeed, in recent years, many methods have been proposed [12, 20, 19, 13, 11, 5, 7, 24, 18, 2, 1]. Yet, very few of them achieve calibration in a global way i.e. without any incremental process.

Since the seminal work on global calibration for orthographic cameras [23], much has been done to cope with its inherent limits (total matching and orthographic camera model). Nevertheless, despite some remarkable results to compute factorization with perspective cameras [21], and factorization with affine cameras and missing data [8], most of today methods are incremental.

In [11, 13, 20, 19], an incremental process is used by estimating new cameras from previously obtained cameras and three-dimensional ($3D$) points. It is subject to an error accumulation and results in a visible and annoying drift for closed scenes. This drift can be attenuated by a final bundle adjustment [25] that enforces the closure constraints, but is prompt to get stuck in local minima. As a result, such methods have to fight against errors during the incremental process (usually by running the bundle adjustment regularly), hoping that the drift does not grow up too quickly and prevents the solution of the optimization problem from falling into a local minimum.

A number of recent studies, oriented toward city modeling from car or aerial sequences, investigate such loop closing. [10] merges partial reconstructions, [16] constrains coherent rotations for loops and planar motion. Adapted to their specific input, these papers often rely on trajectory regularization or dense matching[3, 22]. [24] is a notable exception, where loop constraints are added to sparse Structure from Motion (SfM), yet taking as input an ordered omnidirectional sequence.

In [5], a method is proposed based on trifocal tensors to recover the cameras in one common projective space by estimating the best projective transform between two consecutive camera triplets. However, as noticed in [5], the noise in correspondences may severely affect the coherence of two estimated trifocal tensors. Therefore, in general, the estimated homography does not offer a perfect merging between cameras. As a next step, it is proposed in [2] to connect two consecutive fundamental matrices using the trifocal tensor as an intermediate "glue", which guarantees the uniqueness of subsequent camera matrices without recovering the underlying $3D$ model. The approach we propose in the present paper is linked to that of [2] in that the recovery of camera matrices is carried out by enforcing the fundamental matrices and tensors to be coherent. Nevertheless, unlike our method, that of [2] is incremental, based

on a sliding window, and does not take into account loop constraints.

Substantial progress in performing global calibration is made in [18], where the authors compute coherent fundamental matrices and produce compatible cameras. However, the methodology of [18] does not handle loop constraints and the way to compute the third compatible fundamental matrix is not directly based on images data, *i.e.* correspondences. Perhaps the most extended recent work on global calibration has been conducted by Martinec [12], who proposed a global framework, to calibrate a set of cameras, encoding the loop constraints in a global way. The most important limitation of Martinec's approach is that intrinsic parameters are assumed to be constant. We achieve here the same goal, but in the fully general case of unknown intrinsic parameters.

Our method consists of the following points:

- As in [12], our starting point is a set of unknown cameras linked by estimated epipolar geometries. We assume that along with the estimated fundamental matrices, reliable epipolar correspondences are known. These correspondences are made robust by simultaneously considering several camera pairs, like in [19]. This produces a set of three-camera correspondences that will be used in the sequel.
- We group cameras into triplets. Some of the estimated epipolar geometries are ignored, so that inside a triplet, only two of the three fundamental matrices are considered known. The advantage of this strategy is that we do not need to enforce the coherence of fundamental matrices. At a first sight, this can be seen as a loss of information. However, this information is actually recovered via trifocal tensors.
- We define a graph having as vertices camera triplets. Two triplets are connected by an edge if they share two cameras. We consider a maximal tree embedded in this graph. We demonstrate that for each triplet there exists a 4-vector such that all the entries of the three camera matrices are affine functions of this 4-vector with known coefficients. Writing this relationship for all the vertices of the tree, we get a linear estimate of all the cameras from the aforementioned three-cameras correspondences.
- Finally, considering the same relationships for the complete graph adds constraints. Linearizing these constraints, our method boils down to linear programming which can be efficiently solved by fast algorithms even for very large graphs.
- Once we have all cameras in a projective space, we recover the metric space using an implementation of [14], and a single euclidean bundle adjustment in order to refine the metric space and camera positions.

Thus, we propose a method that accurately recovers geometries, without any incremental process, and deals naturally with loops in the network of cameras. Moreover, the number of unknown parameters is fairly small, since we consider only the cameras (four unknowns for each triplet) and not the 3D points which are usually involved in most structure from motion approaches. Our results could be further refined by standard bundle adjustment techniques. Taking loops into account and avoiding error accumulation, the proposed solution is less prone to get stuck in local minima. Numerical experiments confirm these nice properties.

The remainder of the paper is organized as follows. Section 2 presents the background theory and terminology. Our algorithm is thoroughly described in Section 4. The results of numerical experiments conducted on two real datasets as well as a comparison to state-of-the-art software is provided in Section 5. Discussion concludes the paper.

## 2. Background and notation

In this work, we consider a network of $N$ uncalibrated cameras. In what follows, we assume that for some pairs of cameras $(i, j)$, where $i, j = 1, \dots, N$, $i \neq j$, an estimation of the fundamental matrix, denoted by $\mathtt{F}^{ij}$, is available. Let us denote by $\mathbf{e}^{ij}$ the unit norm epipole in view $j$ of camera center $i$. Recall that the fundamental matrix leads to a projective reconstruction of camera matrices $(\mathtt{P}^i, \mathtt{P}^j)$, which is unique up to a homography.

The geometry of three views $i$, $j$ and $k$ is described by the Trifocal Tensor, hereafter denoted by $\mathcal{T}^{ijk}$. It consists of three $3 \times 3$ matrices: $\mathtt{T}_1^{ijk}$,$\mathtt{T}_2^{ijk}$ and $\mathtt{T}_3^{ijk}$ and provides a particularly elegant description of point-line-line correspondences in terms of linear equations

$$\mathbf{p}_i^\top \begin{bmatrix} \mathbf{l}_j^\top \mathtt{T}_1^{ijk} \\ \mathbf{l}_j^\top \mathtt{T}_2^{ijk} \\ \mathbf{l}_j^\top \mathtt{T}_3^{ijk} \end{bmatrix} \mathbf{l}_k = 0, \tag{1}$$

where $\mathbf{p}_i$ is a point in image $i$ (seen as a point in projective space $\mathbb{P}^2$) which is in correspondence with the line $\mathbf{l}_j$ in image $j$ and with the line $\mathbf{l}_k$ in image $k$. Considering the entries of the Trifocal Tensor as unknowns, we get thus one linear equation for each point-line-line correspondence. Therefore, one point-point-point correspondence $\mathbf{p}_i \longleftrightarrow \mathbf{p}_j \longleftrightarrow \mathbf{p}_k$ leads to 4 independent linear equations by combining an independent pair of lines passing through $\mathbf{p}_j$ in image $j$ with an independent pair of lines passing through $\mathbf{p}_k$ in image $k$.

Since a Trifocal Tensor has 27 entries, the previous argument shows that 7 point-point-point correspondences suffice for recovering the Trifocal Tensor as a solution of an overdetermined system of linear equations. Recall however that the Trifocal Tensor has only 18 degrees of freedom. Most algorithms estimating a Trifocal Tensor from noisy point-point-point correspondences compute an approximate solution to the overdetermined linear system by

least squares estimator (LSE) and then perform a post-processing in order to get a valid Trifocal Tensor. An alternative approach consists in using a minimal solution that determines the three-view geometry from six points [15, 17].

## 3. Main ingredients of our approach

In the present section, we describe two elementary results that represent the building blocks of our approach. It relies on the fact that when two out of three fundamental matrices are known, the Trifocal Tensor has exactly 4 degrees of freedom (recall that a fundamental matrix has 7 degrees of freedom).

**Proposition 1.** *For three views $i$, $j$ and $k$, given two fundamental matrices $\mathtt{F}^{ij}$ and $\mathtt{F}^{ik}$, there exists a 4-vector $\boldsymbol{\gamma} = [\gamma_0, \dots, \gamma_3]$ such that the Trifocal Tensor $\mathcal{T}^{ijk}$ is given by:*

$$T_t^{ijk} = A_t^{ij} \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & \gamma_0 & \gamma_t \end{bmatrix} \left(A_t^{ik}\right)^\top \tag{2}$$

*for every $t = 1, 2, 3$, where*

$$A_t^{is} = \left[ \left(F_{t,1:3}^{is}\right)^\top \,,\, \left(F_{t,1:3}^{is}\right)^\top \times \mathbf{e}^{is} \,,\, \mathbf{e}^{is} \right],\ s = j, k.$$

*Moreover, this Trifocal Tensor is geometrically valid,* i.e.*, there exist 3 camera matrices $\mathtt{P}^i$, $\mathtt{P}^j$ and $\mathtt{P}^k$ compatible with the fundamental matrices $\mathtt{F}^{ij}$ and $\mathtt{F}^{ik}$ and having $\mathcal{T}^{ijk}$ as the Trifocal Tensor.*

The proof of this result is postponed to the Appendix. It is noteworthy that the claims of Proposition 1 hold true under full generality, even if the centers of three cameras are collinear. It is important to present the form of a triplet of camera matrices parameterized by $\boldsymbol{\gamma}$ that are compatible with the fundamental matrices $\mathtt{F}^{ij}$ and $\mathtt{F}^{ik}$ as well as with the Trifocal Tensor defined by Eq. 2. In fact, the triplet of cameras, which is unique up to a projective homography, is

$$\mathtt{P}^i = [\mathtt{I}_{3\times 3} \,|\, \mathbf{0}_{3\times 1}], \tag{3}$$

$$\mathtt{P}^k = [\gamma_0 [\mathbf{e}^{ik}]_\times \mathtt{F}^{ki} \,|\, \mathbf{e}^{ik}], \tag{4}$$

$$\mathtt{P}^j = \mathrm{kron}\big([\boldsymbol{\gamma}_{1:3}, 1]; \mathbf{e}^{ij}\big) - [[\mathbf{e}^{ij}]_\times \mathtt{F}^{ji} \,|\, \mathbf{0}_{3\times 1}], \tag{5}$$

where $\mathrm{kron}(\cdot,\cdot)$ stands for the Kronecker product of two matrices.

In the noiseless setting, Proposition 1 offers a minimal way of computing the 4 remaining unknowns from point-point-point correspondences. At first sight, one could think that one point-point-point correspondence leading to 4 equations is enough for retrieving the 4 unknowns. This is not true, since as the epipolar geometry is known, only one equation brings new information from one point-point-point correspondence. So we need at least 4 point-point-point correspondences to compute the Trifocal Tensor compatible with the two given fundamental matrices. In the noisy case, if we use all 4 equations associated to point-point-point correspondences, the system is then overdetermined and one usually proceeds by computing the LSE. One can also, from 4 point-point-point correspondences, use only one equation for each correspondence and get an estimator by solving a linear system.

The second ingredient in our approach is the parameterization of the homography that bridges two camera triplets having one fundamental matrix in common. Let $i$, $j$, $k$ and $\ell$ be four views such that (a) for views $i$ and $k$ we have successfully estimated the fundamental matrix $\mathtt{F}^{ik}$ and (b) for each triplet $(i,j,k)$ and $(k,i,\ell)$ the estimates of two fundamental matrices are available. Thus, the triplets $(i,j,k)$ and $(k,i,\ell)$ share the same fundamental matrix $\mathtt{F}^{ik}$. Using equations (3)-(5), one obtains two projective reconstructions of camera matrices of views $i$ and $j$ based on two 4-vectors $\boldsymbol{\gamma}$ and $\boldsymbol{\gamma}'$. Let us denote the reconstruction from the triplet $(i,j,k)$ (resp. $(k,i,\ell)$) by $\mathtt{P}^i_{\boldsymbol{\gamma}}$ and $\mathtt{P}^k_{\boldsymbol{\gamma}}$ (resp. $\mathtt{P}^i_{\boldsymbol{\gamma}'}$ and $\mathtt{P}^k_{\boldsymbol{\gamma}'}$). These matrices are connected by a homography $\mathtt{H}_{\boldsymbol{\gamma},\boldsymbol{\gamma}'}$, that is

$$\mathtt{P}^i_{\boldsymbol{\gamma}} \mathtt{H}_{\boldsymbol{\gamma},\boldsymbol{\gamma}'} = \mathtt{P}^i_{\boldsymbol{\gamma}'}, \quad \mathtt{P}^k_{\boldsymbol{\gamma}} \mathtt{H}_{\boldsymbol{\gamma},\boldsymbol{\gamma}'} = \mathtt{P}^k_{\boldsymbol{\gamma}'}. \tag{6}$$

Considering the camera matrices as known, one can solve (6) w.r.t. $\mathtt{H}_{\boldsymbol{\gamma},\boldsymbol{\gamma}'}$. The solution of (6) can be easily computed[1] and is

$$\mathtt{H}_{\boldsymbol{\gamma},\boldsymbol{\gamma}'} = \left[\begin{array}{c|c} \mathrm{kron}(\boldsymbol{\gamma}'_{1:3}, \mathbf{e}^{ki}) - [\mathbf{e}^{ki}]_\times \mathtt{F}^{ik} & \mathbf{e}^{ki} \\ \hline -\frac{\gamma_0}{2} \,\mathrm{tr}([\mathbf{e}^{ik}]_\times \mathtt{F}^{ki} [\mathbf{e}^{ki}]_\times \mathtt{F}^{ik}) (\mathbf{e}^{ik})^\top & 0 \end{array}\right]. \tag{7}$$

To sum up this section, let us stress that the main message to retain from all these formulas is that the homography $\mathtt{H}_{\boldsymbol{\gamma},\boldsymbol{\gamma}'}$, as well as the camera matrices given by (3)-(5) are linear in $(\boldsymbol{\gamma}, \boldsymbol{\gamma}')$.

## 4. Global network calibration

This section contains the core of our contribution which is based on a graph-based representation of the triplets of cameras.

### 4.1. Graph of triplets of cameras

The starting point for our algorithm is a set of estimated epipolar geometries. Based on these geometries, one can define a graph $\mathcal{G}_{\text{cam}}$ such that (a) $\mathcal{G}_{\text{cam}}$ has $N$ vertices corresponding to the $N$ cameras and (b) two vertices of $\mathcal{G}_{\text{cam}}$ are connected by an edge if a reliable estimation of the corresponding epipolar geometry is available. As a preprocessing step, we discard some edges from $\mathcal{G}_{\text{cam}}$ in such a way that the resulting graph has no loops of length 3.

[1] See Appendix for more details

From this graph, we deduce the graph $\mathcal{G}_{\text{triplet}} = (\mathcal{V}_{\text{triplet}}, \mathcal{E}_{\text{triplet}})$ of triplets of cameras as follows:

- each vertex $v \in \mathcal{V}_{\text{triplet}}$ of $\mathcal{G}_{\text{triplet}}$ is a triplet of cameras for which two fundamental matrices are available,
- two vertices $v, v' \in \mathcal{V}_{\text{triplet}}$ are connected by an edge if the corresponding triplets have one fundamental matrix in common.

In view of Proposition 1, the global calibration of the network is equivalent to the estimation of a 4-vector for each triplet of cameras. Thus, to each vertex $v$ of the graph of triplets we associate a vector $\boldsymbol{\gamma}^v \in \mathbb{R}$. The large vector $\Gamma = (\boldsymbol{\gamma}^v : v \in \mathcal{V}_{\text{triplet}})$ is the parameter of interest in our framework.

### 4.2. Calibration of an acyclic graph

If, by some chance, it turns out that the graph of triplets is acyclic, then the problem of estimating $\Gamma$ reduces to estimating $N_V = \text{Card}(\mathcal{V}_{\text{triplet}})$ independent vectors $\boldsymbol{\gamma}^v$. This task can be effectively accomplished using point-point-point correspondences and the equation (1). As explained in Section 2, a few point-point-point correspondences suffice for computing an estimator of $\boldsymbol{\gamma}_v$ by least squares.

### 4.3. Calibration as constrained optimization

Acyclic graphs are however the exception rather than the rule. Even if the camera graph is acyclic, the resulting triplet graph may contain loops. To explain how the loops in the graph $\mathcal{G}_{\text{triplet}}$ are handled, let us remark that one can associate a homography (cf. (7)) to each adjacent pair $(v, v')$ of vertices of $\mathcal{G}_{\text{triplet}}$. Using these homographies, each loop of the graph of triplets yields a constraint on the homographies and, therefore, on the parameter vector $\Gamma$. For instance, the 3-loop $v \longleftrightarrow v' \longleftrightarrow v'' \longleftrightarrow v$ gives raise to the constraint

$$\mathtt{H}_{\boldsymbol{\gamma}^v,\boldsymbol{\gamma}^{v'}} \mathtt{H}_{\boldsymbol{\gamma}^{v'},\boldsymbol{\gamma}^{v''}} \mathtt{H}_{\boldsymbol{\gamma}^{v''},\boldsymbol{\gamma}^{v}} = \mathtt{I}_{4\times 4}, \tag{8}$$

where the equality is understood in the projective sense, *i.e.*, up to a multiplicative constant. Equation (8) defines a set of 15 polynomial constraints on the unknown vector $\Gamma$. If the triplet graph contains $N_{\text{cycle}}$ loops, then we end up with $15N_{\text{cycle}}$ constraints. Our proposal—in the case of general graphs of triplets—is to estimate $\Gamma$ by minimizing the algebraic error derived from the equations (1) and point-point-point correspondences (similarly to the least squares estimation proposed in the previous subsection) subject to $15N_{\text{cycle}}$ constraints.

The main advantage of this approach is that if a solution to the proposed optimization problem is found, it is guaranteed to be consistent w.r.t. the loops, which is not the case for the incremental procedures.

### 4.4. Sequential linear programming

Instead of solving the optimization problem that is obtained by combining the LSE with the loop-constraints, we propose here to replace it by a linear program that can be solved fairly rapidly even for a very large network of cameras. To give more details, let us remark that every loop-constraint (8) can be rewritten as $f_j(\Gamma) = 0, \quad j = 1, \dots, 15$, for some polynomial functions $f_j$. Writing these constraints for all $N_{\text{cycle}}$ loops, we get

$$f_j(\Gamma) = 0, \quad j = 1, \dots, 15N_{\text{cycle}}. \tag{9}$$

On the other hand, in view of (1) and (2), the point-point-point correspondences can be expressed as an inhomogeneous linear equation system in $\Gamma$

$$\mathtt{M}_p\Gamma = \mathbf{m}_p, \qquad p = 1, \dots, 4N_{\text{3-corr}}, \tag{10}$$

where $N_{\text{3-corr}}$ is the number of correspondences across three views. The matrices $\mathtt{M}_p$ and the vectors $\mathbf{m}_p$ are computed using the known fundamental matrices. Since in practice these matrices are estimated from available data, the system (10) need not be satisfied exactly. Then, it is natural to estimate the parameter-vector $\Gamma$ by solving the problem

$$\begin{array}{c} \min \sum_p \|\mathtt{M}_p\Gamma - \mathbf{m}_p\|_q^q \\ \text{subject to } f_j(\Gamma) = 0, \ \forall j = 1, \dots, 15N_{\text{cycle}}, \end{array} \tag{11}$$

for some $q \geq 1$. Unfortunately, there is no $q$ for which this problem is convex and, therefore, it is very hard to solve. To cope with this issue, we propose a strategy based on local linearization.

We start by computing an initial estimator of $\Gamma$, *e.g.*, by solving the unconstrained (convex) problem with some $q \geq 1$. Then, given an initial estimator $\Gamma_0$, we define the sequence $\Gamma_k$ by the following recursive relation: $\Gamma_{k+1}$ is the solution to the linear program

$$\begin{array}{c} \min \sum_p \|\mathtt{M}_p\Gamma - \mathbf{m}_p\|_1 \\ \text{subject to } |f_j(\Gamma_k) + \nabla f_j(\Gamma_k)(\Gamma - \Gamma_k)| \leq \epsilon, \end{array} \tag{12}$$

where $\epsilon$ is a small parameter. In practice one can always set $\epsilon = 10^{-6}$. There are many softwares—such as GLPK, SeDuMi, SDP3—for solving the problem (12) with highly attractive execution times even for thousands of constraints and variables. Furthermore, empirical experience shows that the sequence $\Gamma_k$ converges very rapidly. Typically, a solution with satisfactory accuracy is obtained after five to ten iterations. Such a behavior is also observed on the numerical experiments reported in this work. The only flaw of this strategy is that there is no guarantee that the solution we get is a global minimum.

Another strategy could be to design a suitable version of the branch and bound algorithm for finding the global minimum of (11), inspired by the recent work [9]. However,

actual implementation of this strategy for the task of global calibration of a network results in an algorithm which is prohibitively time consuming.

## 5. Experiments

The proposed algorithm has been implemented in $C++$ with a call to the SeDuMi package of Matlab for solving the linear program. We have tested our algorithm on three real datasets. The *Dinosaur* dataset (Figure 1 on the top)[2] composed of 36 images, the *Detenice Fountain* dataset[3] composed of 34 images and the *Calvary* dataset[4] composed of 52 images.

One of the main contributions of the paper is the accurate computation of cameras respecting the loop-constraints without neither performing bundle adjustment (BA) nor recovering $3D$ points. As one can see in Figure 1, in the three datasets, without constrained optimization, the epipolar lines in the last image, corresponding to a point clicked in the last but one image, are quite accurate. This is not surprising since the method computes cameras based on a tree, fitting exactly previously computed fundamental matrices. On the other hand, one can observe that epipolar lines in the first image, corresponding to a point clicked in the last but one image, are quite inaccurate, when the loop-constraints are not enforced. This is particularly true for the *Dinosaur* dataset, and even more for the *Calvary* dataset. Once we add the constraint optimization, one can see on the right panel of Figure 1, that epipolar lines closing the sequence become far more accurate.

Since no ground truth is available for the Detenice Fountain and the Calvary datasets, we use the result of the BA as a benchmark for evaluating the impact of the loop-constraints on the estimated cameras. In Figure 2, we can observe that the camera locations estimated without the loop-constraint are very different from those obtained by the BA. In particular, camera $33$ is in front of camera $00$, while according to the estimator provided by the BA it is slightly behind camera $00$. In contrast with this, if the constrained optimization is performed, relative positions of cameras $33$ and $00$ are almost the same as those obtained by the BA. Generally speaking, it is noticeable that the cameras estimated by the BA and those estimated by our constrained optimization approach are very close. This demonstrates the potential of our approach as a non-incremental alternative to the BA.

Figure 3, shows an example of a long sequence in which the unconstrained optimization results in a strong drift (camera $00$ is far behind camera $51$). With the constrained optimization, camera $51$ is close to camera $00$. As one can observe on the small images in Figure 3, the first and last photos were taken from almost the same viewpoint. This configuration is confirmed by the BA. In order to get a quantitative evaluation of the improvement achieved by enforcing the loop-constraints, we successively performed BA with constrained estimators and unconstrained estimators as initial values. The root mean square reprojection error on the whole sequence is 0.87 for the BA with constrained estimator as initial value, while it equals 1.61 for the BA with unconstrained estimator as initial value.

[2] http://www.robots.ox.ac.uk/vgg/

[3] courtesy of D. Martinec

[4] courtesy of Imagine Group http://imagine.enpc.fr/

## 6. Conclusion

In this paper, we have proposed a new approach to the problem of autocalibration of a network of cameras. Our approach is based on a representation of the network of cameras by a graph of trifocal tensors and on a natural parameterization of camera matrices and relating homographies. We have proposed to estimate the unknown parameters by a constrained optimization that can be recast in a linear program. Thanks to the sparsity of the matrices involved in this linear program, the running times of the proposed algorithm are very attractive even for large scale datasets.

The main advantage of the proposed methodology is that it offers a global approach for estimating the network of cameras, whereas the most popular approaches are incremental. The experiments reported in this paper show the potential of our method on datasets containing loops. We stay in the projective context and do not rely on known or estimated internal parameters. Yet, if needed, one final Euclidean bundle adjustment could be conducted to refine the metric space and the camera positions.

Future work includes to incorporate the possible heteroscedasticity of the errors into the objective of the optimization problem. In addition, the performance of the algorithm should be tested on a wider range of experiments. We are currently acquiring larger sequences with several loops for which our framework is expected to be considerably more efficient than classical incremental methods. Another natural path for further investigation is to use our methodology in conjunction with a branch and bound approach in order to improve the robustness with respect to local minima.

**Acknowledgments** This work was partially supported by ANR under grant Callisto.

## References

[1] S. Agarwal, N. Snavely, I. Simon, S. M. Seitz, and R. Szeliski. Building rome in a day. In *ICCV*, 2009. 1

[2] S. Avidan and A. Shashua. Threading fundamental matrices. *IEEE Trans. Pattern Anal. Mach. Intell.*, 23:73–77, 2001. 1

[3] N. Cornelis, K. Cornelis, and L. Van Gool. Fast compact city modeling for navigation pre-visualization. In *CVPR*, 2006. 1

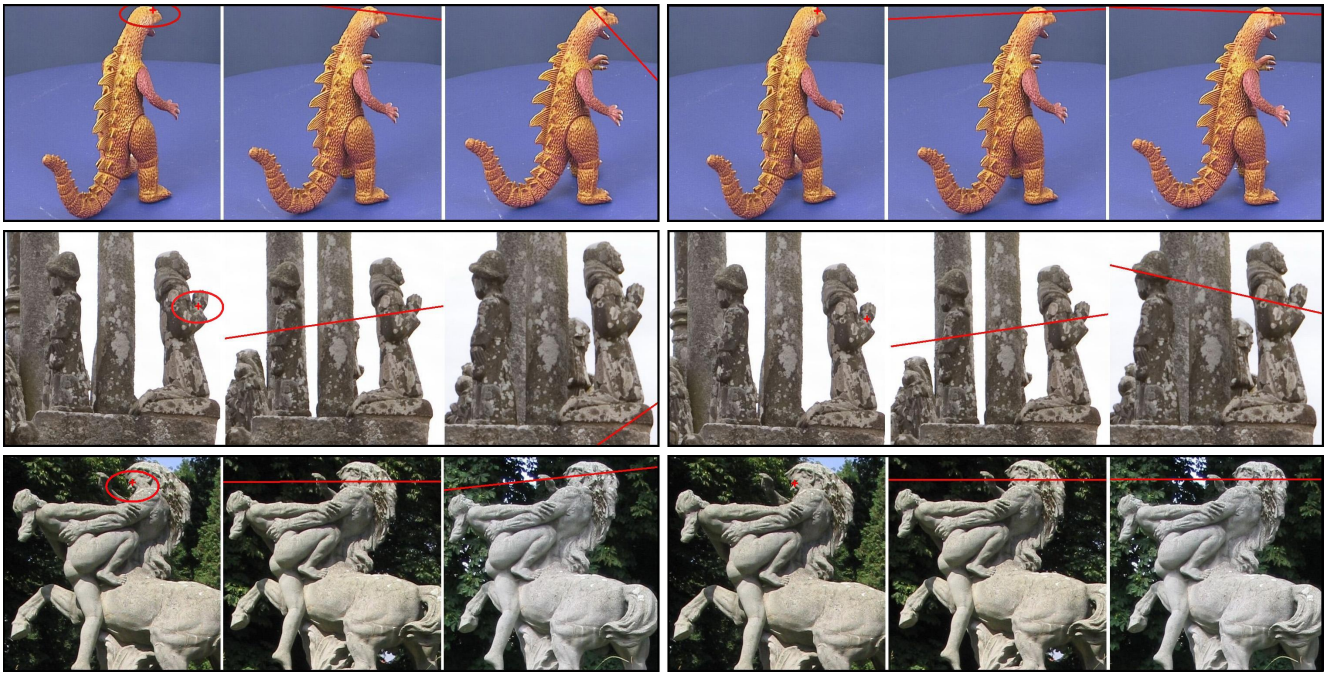

Figure 1. For each cyclic dataset composed of $N$ images from 0 to $N-1$, we show recovered epipolar geometry. Epipolar lines for images $N-1$ and 0 corresponding to a point clicked in the image $N-2$ are drawn. For each row, the three images on the left correspond to calibration without loop constraint, the three images on the right to calibration with loop constraints. As expected, epipolar lines in image 0 are far more accurate with constrained optimization.

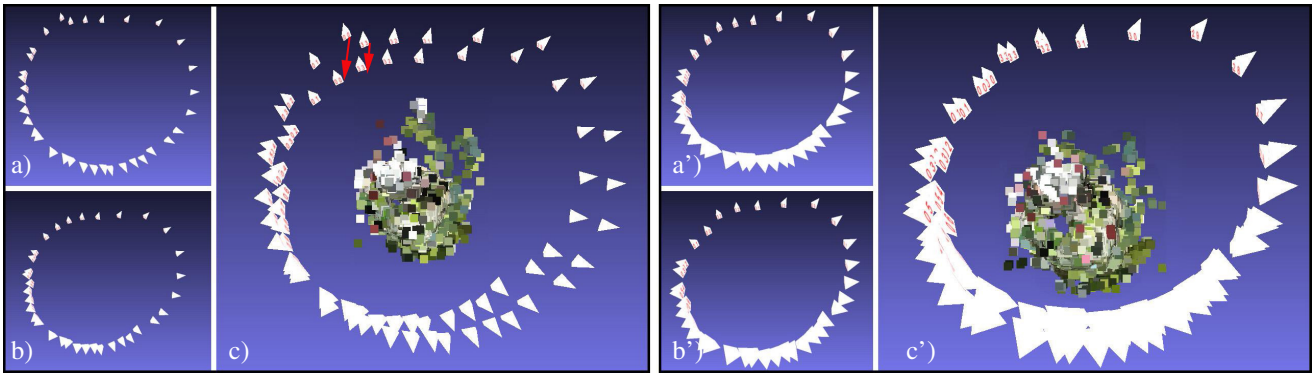


Figure 2. For the *Detenice Fountain* dataset, we compare our method without the loop-constraint on the left, and with constrained optimization on the right. a) and a') show the resulting numbered cameras (tetrahedrons) in both cases, before applying BA. b) and b') show the retrieved cameras after BA. c) and c') show cameras before and after BA to compare camera motion during BA. c) and c') contain $3D$ points cloud in the center. As one can see, without the loop-constraint there is a strong motion during BA, while with constrained optimization the motion during BA is small, this shows the potential of our approach as an alternative to the BA.


[4] O. Faugeras, Q.-T. Luong, and T. Papadopoulou. *The Geometry of Multiple Images: The Laws That Govern The Formation of Images of A Scene and Some of Their Applications*. MIT Press, Cambridge, MA, USA, 2001. 1

[5] A. Fitzgibbon and A. Zisserman. Automatic camera recovery for closed or open image sequences. In *ECCV*, 1998. 1

[6] R. Hartley and A. Zisserman. *Multiple view geometry in computer vision.* Cambridge University, 2nd ed.,, 2003. 1, 8

[7] M. Havlena, A. Torii, J. Knopp, and T. Pajdla. Randomized structure from motion based on atomic 3d models from camera triplets. In *CVPR*, 2009. 1

[8] D. Jacobs. Linear fitting with missing data: Applications to structure-from-motion and to characterizing intensity images. In *CVPR*, 1997. 1

[9] F. Kahl, S. Agarwal, M. Chandraker, D. Kriegman, and S. Belongie. Practical global optimization for multiview ge-

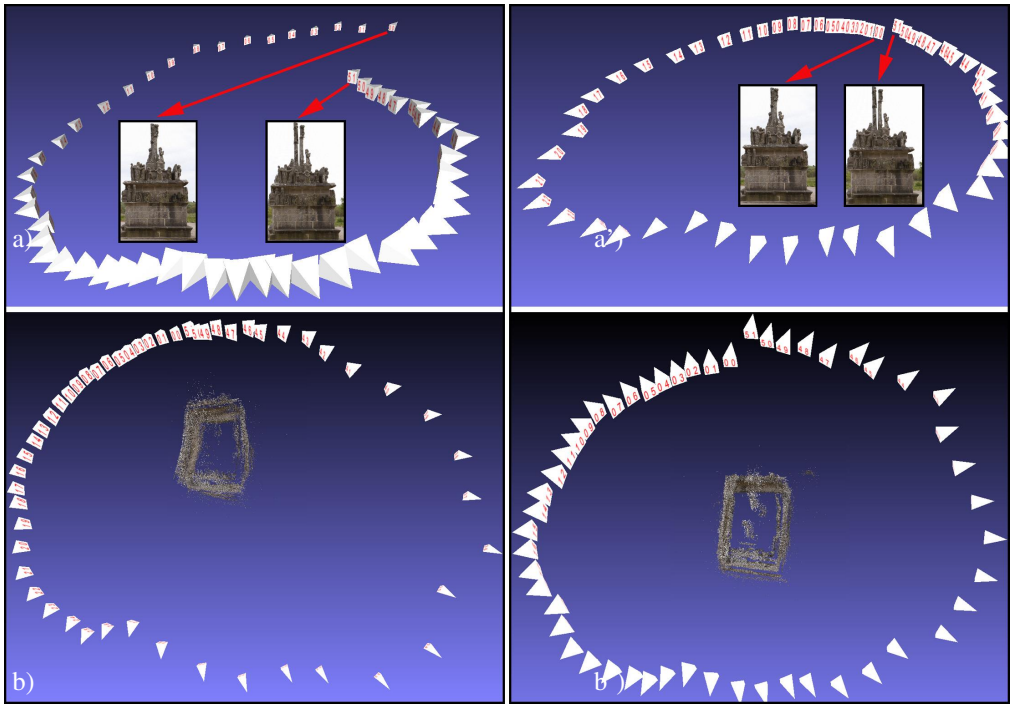


Figure 3. *Calvary* dataset: in a) and a') we show the retrieved cameras (tetrahedrons) before BA without loop-constraint on the left, and with constrained optimization on the right. In b) and b') we show the cameras after BA without loop-constraints (left), and with constrained optimization (right). As one can see in small images at the top, the first and last photos were taken from almost the same viewpoint. This fact is not recovered without loop-constraints, while it is with constrained optimization. After BA in b') the circular configuration is well retrieved, while in b), without loop-constraints, the cameras configuration is not circular and is stretched towards the right.

ometry. *Int. J. Comput. Vision*, 79(3):271–284, 2008. 4

[10] M. Klopschitz, C. Zach, A. Irschara, and D. Schmalstieg. Generalized detection and merging of loop closures for video sequences. In *3DPVT*, 2008. 1

[11] R. Koch, M. Pollefeys, and L. V. Gool. Robust calibration and 3d geometric modeling from large collections of uncalibrated images. In *DAGM*, pages 413–420, 1999. 1

[12] D. Martinec and T. Pajdla. Robust rotation and translation estimation in multiview reconstruction. In *CVPR*, 2007. 1, 2

[13] M. Pollefeys, L. Van Gool, M. Vergauwen, F. Verbiest, K. Cornelis, J. Tops, and R. Koch. Visual modeling with a hand-held camera. *Int. J. Comput. Vision*, 59(3):207–232, 2004. 1

[14] J. Ponce, K. McHenry, T. Papadopoulo, M. Teillaud, and B. Triggs. On the absolute quadratic complex and its application to autocalibration. In *CVPR*, pages 780–787, Washington, DC, USA, 2005. IEEE Computer Society. 2

[15] L. Quan. Invariants of six points and projective reconstruction from three uncalibrated images. *IEEE Trans. Pattern Anal. Mach. Intell.*, 17(1):34–46, 1995. 3

[16] D. Scaramuzza, F. Fraundorfer, and M. Pollefeys. Closing the loop in appearance-guided omnidirectional visual odometry by using vocabulary trees. *Robotics and Autonomous System Journal*, page to appear, 2010. 1

[17] F. Schaffalitzky, A. Zisserman, R. Hartley, and P. Torr. A six point solution for structure and motion. In *ECCV*, pages 632–648, London, UK, 2000. Springer-Verlag. 3

[18] S. N. Sinha, M. Pollefeys, and L. McMillan. Camera network calibration from dynamic silhouettes. In *CVPR*, 2004. 1, 2

[19] N. Snavely, S. M. Seitz, and R. Szeliski. *Photo tourism: Exploring photo collections in 3D*. ACM Press, New York, NY, USA, 2006. 1, 2

[20] N. Snavely, S. M. Seitz, and R. Szeliski. Modeling the world from Internet photo collections. *Int. J. Comput. Vision*, 80:189–210, 2008. 1

[21] P. Sturm and B. Triggs. A factorization based algorithm for multi-image projective structure and motion. In *ECCV (2)*, pages 709–720, 1996. 1

[22] J. Tardif, Y. Pavlidis, and K. Daniilidis. Monocular visual odometry in urban environments using an omnidirectional camera. In *IROS*, pages 2531–2538, 2008. 1

[23] C. Tomasi and T. Kanade. Shape and motion from image streams under orthography: a factorization method. *Int. J. Comput. Vision*, 9(2):137–154, 1992. 1

[24] A. Torii, M. Havlena, and T. Pajdla. From google street view to 3d city models. In *OMNIVIS*, 2009. 1

[25] B. Triggs, P. McLauchlan, R. Hartley, and A. Fitzgibbon. Bundle adjustment - a modern synthesis. In *Workshop on Vision Algorithms*, pages 298–372, 1999. 1

# Appendix

## 6.1. Proof of Proposition 1

We begin by considering the case where all the 3 rows of the fundamental matrices $\mathtt{F}^{ij}$ and $\mathtt{F}^{ik}$ are different from the zero vector of $\mathbb{R}^3$. This implies that the columns of the matrices $\mathtt{A}_t^{ij}$ and $\mathtt{A}_t^{ij}$ form two orthogonal bases of $\mathbb{R}^3$. (Indeed, it is well-known that the epipole $\mathbf{e}^{ij}$ is orthogonal to the rows of $\mathtt{F}^{ij}$, while $(\mathtt{F}_{t,1:3}^{ij})^\top \times \mathbf{e}^{ij}$ is orthogonal to $\mathtt{F}_{t,1:3}^{ij}$ and to $\mathbf{e}^{ij}$ by virtue of the definition of the vector product.) Therefore, $\mathtt{A}_t^{ik}$ and $\mathtt{A}_t^{ij}$ are invertible. Let us define

$$\begin{bmatrix} a_t & b_t & c_t \\ d_t & e_t & f_t \\ g_t & h_t & i_t \end{bmatrix} = (\mathtt{A}_t^{ij})^{-1} \mathtt{T}_t^{ijk} (\mathtt{A}_t^{ik})^{-\top}. \tag{13}$$

Let us show that $a_t = b_t = c_t = 0$. Recall that the matrix $\mathtt{T}_t^{ijk}$ relates a point $\mathbf{p} = (p_1, p_2, p_3) \in \mathbb{P}^2$ to its epipolar line $\mathbf{l} = \mathtt{F}^{ij\top}\mathbf{p}$ through the equation $\mathbf{l}^\top \sum_{s=1}^3 p_s \mathtt{T}_s^{ijk} = \mathbf{0}^\top$ [6, p. 373]. Choosing as $\mathbf{p}$ the vector $(\delta_{t1}, \delta_{t2}, \delta_{t3})^\top$, where $\delta_{t\ell}$ stands for the Kronecker symbol that equals one if $t = \ell$ and zero otherwise, we get $\mathtt{F}_{t,1:3}^{ij} \mathtt{T}_t^{ijk} = \mathbf{0}^\top$. This equation, in conjunction with (13), the definition of $\mathtt{A}_t^{ij}$ and the invertibility of $\mathtt{A}_t^{ik}$ entails that

$$\begin{bmatrix} a_t & d_t & g_t \\ b_t & e_t & h_t \\ c_t & f_t & i_t \end{bmatrix} \begin{bmatrix} \|(\mathtt{F}_{t,1:3}^{ij})^\top\|^2 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

This yields $a_t = b_t = c_t = 0$. By a symmetric argument, we also check that $d_t = g_t = 0$. Thus, the Trifocal Tensor necessarily reduces to the form:

$$\mathtt{T}_t^{ijk} = \mathtt{A}_t^{ij} \begin{bmatrix} 0 & 0 & 0 \\ 0 & e_t & f_t \\ 0 & h_t & i_t \end{bmatrix} (\mathtt{A}_t^{ik})^\top \tag{14}$$

Since the rank of any fundamental matrix is equal to two, there exists an index $t' \in \{1, 2, 3\}$ such that $\mathtt{F}_{t',1:3}^{ij}$ and $\mathtt{F}_{t,1:3}^{ij}$ are not collinear. We have already checked that $\mathtt{F}_{t,1:3}^{ij} \mathtt{T}_t^{ijk} = \mathbf{0}^\top$ and, similarly, $\mathtt{F}_{t',1:3}^{ij} \mathtt{T}_{t'}^{ijk} = \mathbf{0}^\top$. Therefore, substituting $\mathbf{p} = (\delta_{t1}, \delta_{t2}, \delta_{t3})^\top + (\delta_{t'1}, \delta_{t'2}, \delta_{t'3})^\top$ in the equation $\mathbf{p}^\top \mathtt{F}^{ij} \sum_{s=1}^3 p_s \mathtt{T}_s^{ijk} = 0$, we get

$$\mathtt{F}_{t',1:3}^{ij} \mathtt{T}_t^{ijk} + \mathtt{F}_{t,1:3}^{ij} \mathtt{T}_{t'}^{ijk} = \mathbf{0}^\top. \tag{15}$$

Now, let us observe that $(\mathtt{F}_{t',1:3}^{ij})^\top((\mathtt{F}_{t,1:3}^{ij})^\top \times \mathbf{e}^{ij}) = -(\mathtt{F}_{t,1:3}^{ij})^\top((\mathtt{F}_{t',1:3}^{ij})^\top \times \mathbf{e}^{ij}) := \beta_{t,t'}$. Moreover, $\beta_{t,t'} \neq \mathbf{0}$ since the vectors $\mathtt{F}_{t,1:3}^{ij}$ and $\mathtt{F}_{t',1:3}^{ij}$ are linearly independent and orthogonal to $\mathbf{e}^{ij}$. This observation together with (14) and (15) leads to

$$\mathtt{A}_t^{ik} \begin{bmatrix} 0 & 0 & 0 \\ 0 & e_t & h_t \\ 0 & f_t & i_t \end{bmatrix} \begin{bmatrix} \alpha \\ \beta \\ 0 \end{bmatrix} + \mathtt{A}_{t'}^{ik} \begin{bmatrix} 0 & 0 & 0 \\ 0 & e_{t'} & h_{t'} \\ 0 & f_{t'} & i_{t'} \end{bmatrix} \begin{bmatrix} \alpha \\ -\beta \\ 0 \end{bmatrix} = \mathbf{0},$$

where we have used the shorthands $\alpha = \alpha_{t,t'}$ and $\beta = \beta_{t,t'}$. Matrix multiplication yields

$$\beta_{t,t'} \Big( [0,\ e_t,\ f_t]\, (\mathtt{A}_t^{ik})^\top - [0,\ e_{t'},\ f_{t'}]\, (\mathtt{A}_{t'}^{ik})^\top \Big) = \mathbf{0}^\top.$$

The last display is equivalent to

$$(e_t \mathtt{F}_{t,1:3}^{ik} - e_{t'} \mathtt{F}_{t',1:3}^{ik})^\top \times \mathbf{e}^{ik} + (f_t - f_{t'})\mathbf{e}^{ik} = \mathbf{0},$$

which is possible if and only if $f_t = f_{t'}$ and $e_t \mathtt{F}_{t,1:3}^{ik} - e_{t'} \mathtt{F}_{t',1:3}^{ik} = \mathbf{0}$. Since $\mathtt{F}_{t,1:3}^{ik}$ and $\mathtt{F}_{t',1:3}^{ik}$ are linearly independent, we conclude that $e_t = e_{t'} = 0$. In addition, using a symmetric argument, we get $h_t = h_{t'}$ and thus

$$\begin{aligned} \mathtt{T}_t^{ijk} &= \mathtt{A}_t^{ij} \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & f_t \\ 0 & h_t & i_t \end{bmatrix} (\mathtt{A}_t^{ik})^\top, \\ \mathtt{T}_{t'}^{ijk} &= \mathtt{A}_{t'}^{ij} \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & f_t \\ 0 & h_t & i_{t'} \end{bmatrix} (\mathtt{A}_{t'}^{ik})^\top \end{aligned} \tag{16}$$

Let now $t''$ be the element of the index set $\{1, 2, 3\}$ that is different from $t$ and $t'$. Repeating the same arguments with $t$ replaced by $t''$ leads to a formula similar to (16) in which $t'$ is replaced by $t''$ everywhere. The first claim of the proposition follows by dividing all the entries of $[\mathtt{T}^{ijk}]$ by $f_t$, which is $\neq 0$ since otherwise the fundamental matrix computed from this trifocal tensor is of rank $< 2$, which is impossible.

In the case where the matrix $\mathtt{F}^{ij}$ or $\mathtt{F}^{ik}$ contain zero rows, one can merely use the fact that Eq. (3)-(5) characterize all the triplets of camera matrices (up to a projective homography) which are compatible with the fundamental matrices $\mathtt{F}^{ij}$ and $\mathtt{F}^{ik}$. In view of [6, Eq. 15.1], the trifocal tensor corresponding to these camera matrices coincides with the one defined in the statement of the proposition. This completes the proof.

## 6.2. Proof of Equation 7

Let $\mathtt{P}^i$ and $\mathtt{P}^k$ be the camera matrices computed from the triplet $(i, j, k)$ with parameter $\boldsymbol{\gamma}$ and let $\bar{\mathtt{P}}^i$ and $\bar{\mathtt{P}}^k$ be those computed from the triplet $(k, i, \ell)$ with parameter $\boldsymbol{\gamma}'$. Thus, we are looking for a homography $\mathtt{H}$ such that

$$[\mathtt{I}_{3\times 3}|\mathbf{0}]\mathtt{H} \cong \text{kron}([\boldsymbol{\gamma}'_{1:3}, 1], \mathbf{e}^{ki}) - \big[[\mathbf{e}^{ki}]_\times \mathtt{F}^{ik}|\mathbf{0}\big], \tag{17}$$

$$\big[\gamma_0 [\mathbf{e}^{ik}]_\times \mathtt{F}^{ki} | \mathbf{e}^{ik}\big]\mathtt{H} \cong [\mathtt{I}_{3\times 3}|\mathbf{0}], \tag{18}$$

where $\cong$ stands for the proportionality. The first equation yields $\mathtt{H}_{1:3,1:4} = \big[\text{kron}(\boldsymbol{\gamma}'_{1:3}, \mathbf{e}^{ki}) - [\mathbf{e}^{ki}]_\times \mathtt{F}^{ik} | \mathbf{e}^{ki}\big]$. Inserting this in (18) and using $\mathtt{F}^{ki}\mathbf{e}^{ki} = \mathbf{0}$, we get

$$-\gamma_0 [\mathbf{e}^{ik}]_\times \mathtt{F}^{ki} [\mathbf{e}^{ki}]_\times \mathtt{F}^{ik} + \mathbf{e}^{ik} \mathtt{H}_{4,1:3} = \alpha \mathtt{I}_{3\times 3} \tag{19}$$

and $\mathbf{e}^{ik}\mathtt{H}_{4,4} = 0$. This implies that $\mathtt{H}_{4,4} = 0$. Furthermore, multiplying both sides of (19) by $(\mathbf{e}^{ik})^\top$, we get $\mathtt{H}_{4,1:3} = \alpha(\mathbf{e}^{ik})^\top$. To complete the proof, it remains to determine the value of $\alpha$. This is done by computing the trace of both sides in (19).