



LMI-based 2D-3D Registration: from Uncalibrated Images to Euclidean Scene

Danda Pani Paudel, Adlane Habed, Cedric Demonceaux, Pascal Vasseur

► To cite this version:

Danda Pani Paudel, Adlane Habed, Cedric Demonceaux, Pascal Vasseur. LMI-based 2D-3D Registration: from Uncalibrated Images to Euclidean Scene. CVPR 2015 - 28th IEEE Conference on Computer Vision and Pattern Recognition, Jun 2015, Boston, United States, Jun 2015, Boston, United States. hal-01149529v3

HAL Id: hal-01149529

<https://inria.hal.science/hal-01149529v3>

Submitted on 8 Jun 2015

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

LMI-based 2D-3D Registration: from Uncalibrated Images to Euclidean Scene

Danda Pani Paudel¹, Adlane Habed², Cédric Demonceaux¹, Pascal Vasseur³

¹Le2i laboratory, University of Bourgogne, CNRS, France

²ICube laboratory, University of Strasbourg, CNRS, France

³LITIS EA laboratory, University of Rouen, France,

{danda-pani.paudel, cedric.demonceaux}@u-bourgogne.fr, adlane.habed@icube.unistra.fr,
pascal.vasseur@univ-rouen.fr

Abstract

This paper investigates the problem of registering a scanned scene, represented by 3D Euclidean point coordinates, and two or more uncalibrated cameras. An unknown subset of the scanned points have their image projections detected and matched across images. The proposed approach assumes the cameras only known in some arbitrary projective frame and no calibration or autocalibration is required. The devised solution is based on a Linear Matrix Inequality (LMI) framework that allows simultaneously estimating the projective transformation relating the cameras to the scene and establishing 2D-3D correspondences without triangulating image points. The proposed LMI framework allows both deriving triangulation-free LMI cheirality conditions and establishing putative correspondences between 3D volumes (boxes) and 2D pixel coordinates. Two registration algorithms, one exploiting the scene's structure and the other concerned with robustness, are presented. Both algorithms employ the Branch-and-Prune paradigm and guarantee convergence to a global solution under mild initial bound conditions. The results of our experiments are presented and compared against other approaches.

1. Introduction

With the ongoing surge in affordable high quality 3D and 2D capture sensors, the two modalities are increasingly often jointly used in vision-based systems. Using information from both 2D and 3D sensors, such as a laser scanner or RGB-D camera, provides several advantages ranging from texture mapping to scene understanding. It, however, may also come with its share of difficulties and challenges. When the 3D sensor and 2D cameras are rigidly attached, the system can be calibrated at once. If the two

sensor modalities are free, many such systems require the 2D cameras to be internally calibrated and registered with the 3D sensor at all time. Camera calibration can be carried out either by using a dedicated pattern in the scene or via autocalibration. The registration of the 2D cameras and the 3D sensor is generally achieved by establishing correspondences in two ways: (i) between scanned 3D data and 2D features; (ii) between scanned 3D points and 3D triangulated points from calibrated cameras.

The problem of establishing correspondences between 3D data and 2D features has been tackled in [23] for points, [5] for lines, [12] using planes, [22] using skylines and [13] by relying on scene constraints. The success of such methods is often undermined by the absence of reliable 3D descriptors and their lack of compatibility with 2D descriptors. They may also be undermined by the likewise unreliable descriptors for certain image features such as lines. Other registration methods are based on mutual information [29] and region segmentation [27] but suffer from similar drawbacks. Establishing correspondences between scanned and triangulated 3D points is often carried out using the Iterative Closest Point (ICP) algorithm. The image-induced scene may however be reconstructed only up to a scale ambiguity making the problem particularly difficult to solve. In [6], this ambiguity is handled by an extension of the 4-points congruent sets algorithm [1]. Other scale-invariant registration methods perform registration either by a voting approach [19] or performing mean-shift in the scale invariant space [20]. None of these methods guarantees convergence to a globally optimal solution. The recent Go-ICP method, proposed in [30], is a globally optimal method for registering 3D point clouds with the same scale. The scale of the image-induced 3D scene can only be corrected with additional knowledge. However, using any of [6, 19, 20, 30] requires the cameras to be calibrated.

This paper investigates the problem of registering a scanned scene, represented by Euclidean 3D point coordinates, and two or more uncalibrated cameras. An unknown subset of the scanned points have their image projections detected and matched across images. The proposed approach assumes camera matrices to be calculated in some arbitrarily chosen projective frame and no calibration or autocalibration is required. We argue here that camera calibration may turn out to be impractical due to possible changes in the cameras' internal geometry when zooming and focusing. As for camera autocalibration, although globally convergent methods [8, 3, 4, 9] do exist, the process fails for numerous critical motions of the cameras and is generally sensitive to 2D pixel localization errors. When cameras are uncalibrated, the transformation relating the cameras to the scene is projective. Our proposed registration solution is based on a Linear Matrix Inequality (LMI) framework that allows simultaneously estimating this unknown projective transformation and establishing 2D-3D correspondences without triangulating image points. The proposed LMI framework allows both deriving triangulation-free LMI cheirality conditions and establishing putative correspondences between 3D volumes (boxes) and 2D pixel coordinates. Directly using raw 2D points in lieu of triangulated 3D points is believed to yield more accurate motion computation [24]. In practice, triangulation results are rather uncertain in the depth direction. Using a small set of such reconstructed points for alignment may have a devastating effect on the results [17].

Two registration algorithms, one exploiting the scene's structure and the other concerned with robustness, are presented. Both algorithms employ the Branch-and-Prune paradigm and guarantee convergence to a global solution under some mild initial bounding conditions. Our algorithms require initial box-2D correspondences with 5 non-overlapping boxes to guarantee convergence to a global solution. Alternatively, non-overlapping bounds on camera centers can also be used. Finding initial bounds on camera positions is relatively easy as far as hand-held or GPS-equipped cameras are concerned. The results of our experiments, on both simulated and real data, are also presented.

2. Background and notations

Consider a static scene consisting of m 3D points $\{X_j\}_{j=1}^m$ observed by $n \geq 2$ uncalibrated 2D pinhole cameras $\{P^i\}_{i=1}^n$. Scene points and cameras may be retrieved from point correspondences across images up to an unknown projective ambiguity. Let P^i be the 3×4 matrix representation of P^i and X_j the homogeneous coordinate vector of X_j , all expressed in a common projective frame. The special point C_i , with coordinates C_i in this frame satisfying $P^i C_i = 0$ (0 is the null-vector), is the optical center of camera P^i . The 2D pixel projection x_j^i of a scene point

X_j onto camera P^i is given by $x_j^i \sim P^i X_j$ where $\sim$ refers to the equality up to an unknown scale.

Triangulation: Any point X_j can be triangulated in a 3D coordinate frame given camera matrices and 2D pixel correspondences $\{x_j^i\}_{i=1}^n$ across images. So long as at least two 2D points are matched in at least two images, if a x_j^i is unknown in one given image (no corresponding feature point detected and/or matched in that image), it can safely be replaced by the null vector without prejudice for what follows. Let S_j be the $3n \times 3n$ block-diagonal matrix

$$S_j = \text{diag}([x_j^1]_{\times}, [x_j^2]_{\times}, [x_j^3]_{\times}, \dots, [x_j^n]_{\times}) \quad (1)$$

with matrices $[x_j^i]_{\times}$, $i = 1 \dots n$, on the diagonal blocks and zeros elsewhere. $[x_j^i]_{\times}$ denotes the 3×3 skew-symmetric matrix associated with the cross-product and constructed using the projection x_j^i of X_j on camera P^i . Let M be the $3n \times 4$ matrix obtained by stacking all camera matrices:

$$M^T = [P^1{}^T \ P^2{}^T \ P^3{}^T \ \dots \ P^n{}^T]. \quad (2)$$

The coordinate vector of X_j can then be obtained by solving $S_j M X_j = 0$. Note that matrix $S_j M$ must be of rank-3, or else assumed to be enforced as such throughout this paper.

Cheirality: As far as the true Euclidean camera matrices and 3D points are concerned, the depth of any scene point, relative to a camera in which it is visible, must be positive. The sign of this depth is referred to as the *cheirality* of the point with respect to the considered camera [11, 10]. However, in addition to the projective ambiguity, projective points and cameras are each retrieved up to a different unknown scale generally not preserving cheirality. It is possible though to assign signatures $\zeta_i = \pm 1$ to cameras and signatures $\eta_j = \pm 1$ to points to ensure that: (i) each point has a consistent cheirality with respect to all cameras in which it is visible, and (ii) all points have a consistent cheirality with respect to any one camera in which they are visible.

Camera signatures: Let X be a point visible in camera P . The cheirality of X with respect to any camera P^i in which it is also visible must be identical to that of its cheirality with respect to P . This can be enforced by considering the signatures ζ and ζ_i of, respectively, P and P^i such that

$$(\zeta P X)_3 (\zeta_i P^i X)_3 > 0 \text{ for } X \text{ visible in } P \text{ and } P^i. \quad (3)$$

Note that (3) can be used to deduce the signature of one camera given the signature of the other. Indeed, this can be done by initially assigning an arbitrarily chosen signature ζ to one given camera P and iteratively assigning signatures to all cameras observing X . Every P^i with assigned signature can in turn be used to deduce signatures of cameras sharing visible points with it. A robust version of such algorithm may be found in [16].

Point signatures: Correcting the signatures of cameras suffices to enforce identical cheirality for any given point in all

the views in which it is visible. It, however, remains that any two points X and X_j , visible in the same camera P , may have different cheiralities, $(PX)_3(PX_j)_3 < 0$, with respect to that camera. To make such points share the same cheirality relative to one such camera, one seeks the signatures η and η_j of these points such that

$$(\eta PX)_3(\eta_j PX_j)_3 > 0 \text{ for } X \text{ and } X_j \text{ visible in } P. \quad (4)$$

Using (4), one may arbitrarily assign a signature η to one of the points X and recover the signatures of the remaining visible points. Once a signature is assigned to a point, it can be used to assign signatures to points visible in other views. **Cheirality inequalities:** Note that (3) allows to assign signatures to cameras independently from the homogeneous representation of the considered visible points. Likewise, signatures are assigned to points through (4) independently from the camera signatures. However, this suffices to guarantee that the cheirality of any point to be identical with respect to all cameras in which it is visible. It also guarantees that all points visible by one camera carry the same cheirality with respect to it. As in the Euclidean frame, once signatures are assigned to cameras and points, the plane at infinity Π_∞ neither cuts through the convex hull of scene points nor does it cut through the convex hull of camera centers. The projective coordinates of Π_∞ must satisfy:

$$\eta_j \Pi_\infty^T X_j > 0 \text{ for } j = 1 \dots m, \quad (5)$$

$$\delta \zeta_i \Pi_\infty^T C_i > 0 \text{ for } i = 1 \dots n \quad (6)$$

for some $\delta = \pm 1$. Note that the coordinate vectors C_i referred to in (6) ought to be obtained exactly through the identity $C_i^T \Pi = \det([P^i{}^T \mid \Pi])$ for some 4-vector Π .

Upgrade: The plane at infinity plays a key role in upgrading a projective reconstruction to its Euclidean or affine counterpart. For instance, the Euclidean coordinates X_j^E of points X_j and Euclidean camera matrices P^{Ei} of P^i , satisfying $x_j^i \sim P^{Ei} X_j^E$, are only a projective transformation, say H , away from their projective counterparts: $X_j^E \sim H X_j$ and $P^{Ei} \sim P^i H^{-1}$. The full-rank 4×4 matrix H is the matrix representation of H . Unless the cameras are calibrated and their pose calculated, H is unknown. However, the last row of H is the homogeneous coordinate vector Π_∞ of the plane at infinity in the projective frame. If the latter is known, for arbitrarily chosen remaining rows of H , points at infinity in the true scene are mapped back onto the canonical plane. In this case, the scene and cameras are said to be reconstructed in an affine frame. Π_∞ being generally unknown, one may use a surrogate plane, say $\tilde{\Pi}_\infty$ whose coordinate vector $\tilde{\Pi}_\infty$ in the projective frame satisfies (5) and (6). The resulting reconstruction is then said to be quasi-affine with respect to the considered points and camera centers.

LMIs and SDPs: When dealing with matrices, $A > 0$ (resp. $A \geq 0$) means that the symmetric matrix A is positive-definite (resp. positive semi-definite). A Linear

Matrix Inequality (LMI) is a constraint on a real-valued vector $y = (y_1, y_2, \dots)$ such that $A(y) > 0$. The matrix $A(y) = A_0 + \sum_i y_i A_i$ is an affine function of y involving symmetric matrices $A_0, A_1, A_2, \dots$. A LMI feasibility problem consists in finding y that satisfies the considered LMI or determining that no solution exists. It is a convex optimization problem that can be efficiently solved using interior-point methods [2]. When a LMI $A(y) > 0$ arises in **homogeneous form**, i.e. $A(y) = \sum_i y_i A_i$, it is replaced by a non-homogeneous counterpart $A(y) \geq I$ as to avoid numerical issues since $A(0) = 0$. A **Semidefinite program (SDP)** consists in minimizing or maximizing a linear objective subject to LMI constraints. A key ingredient in the work presented in this paper is the following lemma:

Lemma 2.1 (Finsler's) *Let Y be a vector, Q a symmetric matrix, B a rectangular matrix - all real-valued and of appropriate dimensions - and γ a scalar. The following statements are equivalent:*

$$(i) \quad Y^T Q Y > 0 \quad \forall Y \neq 0: BY = 0.$$

$$(ii) \quad \exists \gamma: Q + \gamma B^T B > 0.$$

Lemma 2.1 is due to Paul Finsler [7]. It allows to convert the problem of checking the sign of a quadratic form over a subspace into solving a LMI problem.

Further notations: Additional notations are used throughout the paper: the canonical vectors are denoted e_k , $k = 1, 2, 3$, such that $e_1 = (1 \ 0 \ 0)^T$, $e_2 = (0 \ 1 \ 0)^T$ and $e_3 = (0 \ 0 \ 1)^T$. The superscript $*$ refers to the symmetric part of a square matrix. For example, the symmetric part Q^* of a square matrix Q is given by $Q^* = \frac{1}{2}(Q + Q^T)$.

3. LMI-based 2D-to-3D registration

In this section, we first introduce a set of LMI and bounding conditions that constitute the backbone of our 2D-3D registration algorithms. The proposed algorithms are also presented in this section. We consider the scene imaged by a sequence of uncalibrated cameras and scanned by a 3D sensor. In addition to 2D point correspondences across images, the scanned scene points are given by their Euclidean coordinates X_j^E , $j = 1 \dots m$. In the absence of Euclidean-to-projective (3D-3D) point correspondences and Euclidean-to-image (3D-2D) point correspondences, the scanned points are an unknown projective transformation away, $X_j^E \sim H X_j$, from the image-induced 3D points X_j . Note that H can be linearly calculated if 3D-3D point correspondences are available. It can also be estimated from 3D-2D point correspondences via $x_j^i \sim P^i H^{-1} X_j^E$. It goes without saying that, if H is known, then the correspondences can be established. However, when neither H nor 3D-3D or 3D-2D correspondences are known, the problem is particularly challenging and difficult to solve. Our goal is precisely to simultaneously establish such unknown correspondences

and estimate H while using only 2D pixel coordinates and the Euclidean coordinates of the scanned points: i.e. without triangulating image points in 3-space. Once the correspondences established and H estimated, the Euclidean matrices P^{Ei} , camera pose and internal calibration parameters can be extracted. Our proposed solution heavily depends upon finding a surrogate plane at infinity $\tilde{\Pi}_\infty$ that wouldn't cross the scene and cameras. This however traditionally requires the so-called cheirality inequalities involving image points to be triangulated. Therefore, prior to presenting our registration conditions and methods, we first provide a LMI formulation of the cheirality inequalities for obtaining such “quasi-affine” plane without triangulating image points.

3.1. Cheirality LMIs

Consider a point X visible in camera P . The signature ζ of P and that of any camera P^i in which X is also visible must satisfy (3). Note that (3) can be rewritten as $\zeta \zeta_i X^T P^T e_3 e_3^T P^i X > 0$. One can only notice that the latter inequality is equivalent to $\zeta \zeta_i X^T (P^T e_3 e_3^T P^i)^* X > 0$ when employing the symmetric part of the involved matrix. Finsler's lemma can then be used to deduce the LMI

$$\exists \gamma : \zeta \zeta_i (P^T e_3 e_3^T P^i)^* + \gamma^i (SM)^T SM > 0 \quad (7)$$

for X visible in P and P^i . In (7), γ^i is a scalar and matrices S and M are constructed as in (1) and (2) from the image projections of point X and camera matrices. Note that LMI (7) is equivalent to (3). It allows to correct the signature of a camera given the signature of another camera. Unlike (3), LMI (7) does not require triangulating any point X . As in (3), an arbitrary signature ζ can initially be assigned to camera P and every matrix whose signature is recovered can be used to deduce the signatures of other cameras.

An alternative to (4) would be to enforce that all points X_j visible in some camera P^i have positive cheirality, i.e. $\zeta_i \eta_j e_3^T P^i X_j > 0$ as demanded when using the true Euclidean points and cameras. From (5), one can deduce that η_j and $X_j^T \Pi_\infty$ must carry the same sign. Because Π_∞ is homogeneous, we can choose the plane at infinity such that $\zeta_i X_j^T \Pi_\infty e_3^T P^i X_j > 0$. The latter inequality remains true when considering the symmetric part of the matrix involved: $X_j^T (\zeta_i \Pi_\infty e_3^T P^i)^* X_j > 0$. Using Finsler's lemma and accounting for homogeneity, we deduce that LMI

$$(\zeta_i \Pi_\infty e_3^T P^i)^* + \gamma_j^i (S_j M)^T S_j M > I \quad (8)$$

must hold for any point X_j visible in P^i for some scalar γ_j^i and the true Π_∞ . Given the signatures of all cameras obtained via (7), LMI (8) is an equivalent alternative to using (5) to calculate a “quasi-affine” plane $\tilde{\Pi}_\infty$, satisfying (8), not cutting through the convex-hull of visible points. Unlike (5), LMI (8) neither requires the calculation of point signatures nor does it require the reconstruction of the observed points in 3-space. A surrogate plane at infinity $\tilde{\Pi}_\infty$

can be obtained by solving the LMIs (8) along with inequalities (6) (with $\delta = \pm 1$) for all cameras and visible points.

3.2. Bounding LMIs

Definition 3.1 (Positive vs. negative sides of a plane)

Consider a plane Π with Euclidean coordinate vector Π^E . We say that a point X with coordinates X^E in this frame lies on the positive side with respect to this plane if and only if $X^E \Pi_\infty^E \Pi_\infty^E X^E > 0$. The coordinate vector $\Pi_\infty^E = (0 \ 0 \ 0 \ 1)^T$ is that of the plane at infinity in the Euclidean frame. Points on the negative side with respect to Π satisfy $X^E \Pi_\infty^E \Pi_\infty^E X^E < 0$.

Definition 3.2 (Boxing a point)

Let $\mathcal{B} = \{(\underline{\Pi}_k, \bar{\Pi}_k)\}_{k=1}^3$ be a set of three pairs of planes with Euclidean coordinate vectors $\underline{\Pi}_k^E = (e_k^T - \underline{d}_k)^T$ and $\bar{\Pi}_k^E = (e_k^T - \bar{d}_k)^T$ such that the signed distances $\underline{d}_k$ and $\bar{d}_k$ of the planes to the origin of the frame satisfy $\underline{d}_k < \bar{d}_k$. Without loss of generality, the normal vectors e_k of the planes are assumed to be the canonical basis vectors. We say that a point X is boxed by $\mathcal{B}$ if for each pair $(\underline{\Pi}_k, \bar{\Pi}_k)$, X is on the positive side with respect to $\underline{\Pi}_k$ and negative side with respect to $\bar{\Pi}_k$.

Let S_j be the matrix constructed as in (1) from 2D matches. Let M^E (resp. M) be, as in (2), the stack of Euclidean (resp. projective) camera matrices P^{Ei} (resp. P^i), $i = 1 \dots m$. Based on the above definitions, the following corollary can be directly deduced from Finsler's lemma.

Corollary 3.3 A point X_j projecting onto x_j^i in cameras $\{P^i\}_{i=1}^n$ is boxed by $\mathcal{B}_j = \{(\underline{\Pi}_k, \bar{\Pi}_k)\}_{k=1}^3$ if and only if the following LMIs are simultaneously feasible for some scalars γ_{jk} and $\bar{\gamma}_{jk}$:

$$(\Pi_\infty^E \underline{\Pi}_k^E)^* + \gamma_{jk} (S_j M^E)^T S_j M^E > 0 \quad k = 1, 2, 3 \quad (9)$$

$$\bar{\gamma}_{jk} (S_j M^E)^T S_j M^E - (\Pi_\infty^E \bar{\Pi}_k^E)^* > 0 \quad k = 1, 2, 3. \quad (10)$$

Remark 3.4 Note that if any of LMIs (9) and (10) is feasible for some γ_{jk} , then the same LMI is also feasible for any $\gamma > \gamma_{jk}$. Hence, one can seek a single γ simultaneously satisfying (9) and (10) rather than six scalars γ_{jk} , $\bar{\gamma}_{jk}$ ($k = 1, 2, 3$) for each point. This also means that a single γ can be sought for the LMIs induced by multiple points X_j . We henceforth express all our LMIs using a common γ .

Corollary 3.3 allows to express the correspondence between a box in 3D and 2D point matches. It basically states that if the 3D point X_j was to be triangulated from 2D correspondences $\{x_j^i\}_{i=1}^m$, then it would be within the box $\mathcal{B}_j$ if LMIs (9) and (10) were feasible and outside this box otherwise. However, LMIs (9) and (10) depend upon the unknown Euclidean camera matrices and the true plane at infinity. Let us now consider the block-diagonal matrix

$$B_j = \text{diag}(B_j^1, B_j^2, B_j^3, \bar{B}_j^1, \bar{B}_j^2, \bar{B}_j^3) \quad (11)$$

whose blocks $\underline{B}_j^k = (\tilde{\Pi}_\infty \underline{\Pi}_k^E \mathbf{H})^* + \gamma (\mathbf{S}_j \mathbf{M})^T \mathbf{S}_j \mathbf{M}$ and $\bar{B}_j^k = \gamma (\mathbf{S}_j \mathbf{M})^T \mathbf{S}_j \mathbf{M} - (\tilde{\Pi}_\infty \bar{\Pi}_k^E \mathbf{H})^*$ are expressed using projective camera matrices, an unknown 4×4 transformation matrix $\mathbf{H}$, and the surrogate plane at infinity $\tilde{\Pi}_\infty$ (calculated as in Section 3.1). The following holds for visible scene points:

Proposition 3.5 *Let $\mathcal{S}x = \{(X_j, \mathcal{B}_j)\}_{j=1}^m$ be a set of putative point-to-box correspondences (i.e. each point X_j , projecting onto image points $\{x_j^i\}_{i=1}^n$, is assigned to a box $\mathcal{B}_j$). If $\mathcal{S}x$'s correspondences are correct then LMIs*

$$\mathbf{B}_j \geq \mathbf{I}, j = 1 \dots m \quad (12)$$

must be simultaneously feasible for a scalar γ and at least the true transformation matrix $\mathbf{H}$ satisfying $\mathbf{X}_j^E \sim \mathbf{H} \mathbf{X}_j$.

Proof The proof relies on Corollary 3.3. Recall that $\mathbf{M}^E \sim \mathbf{M} \mathbf{H}^{-1}$ and $\Pi_\infty = \mathbf{H}^T \Pi_\infty^E$. It is well-known that congruence transformations preserve definiteness. Hence, pre- and post-multiplying the left-hand side of each of LMIs (9) and (10) by $\mathbf{H}$ and $\mathbf{H}^T$, these can be respectively rewritten as $(\Pi_\infty \underline{\Pi}_k^E \mathbf{H})^* + \gamma (\mathbf{S}_j \mathbf{M})^T \mathbf{S}_j \mathbf{M} > 0$ and $\gamma (\mathbf{S}_j \mathbf{M})^T \mathbf{S}_j \mathbf{M} - (\Pi_\infty \bar{\Pi}_k^E \mathbf{H})^* > 0$. As per Remark 3.4, a common γ is used. Because $\mathbf{X}_j^T \Pi_\infty \bar{\Pi}_\infty^T \mathbf{X}_j$ carry the same sign for all points X_j , one can replace Π_∞ by $\tilde{\Pi}_\infty$ thus leading to all $\underline{B}_j^k$ and $\bar{B}_j^k$ being simultaneously either positive or negative definite. Since $\mathbf{H}$ is unknown, one may choose to enforce positive rather than negative definiteness. $\mathbf{B}_j > 0$ then arises naturally since a block-diagonal matrix is positive-definite if and only if each of its diagonal blocks is positive-definite. Because $\mathbf{B}_j > 0$ is homogeneous, it is replaced by (12). ■

Similarly, consider the matrix

$$\mathbf{D}_i = \text{diag}(\underline{D}_i^1, \underline{D}_i^2, \underline{D}_i^3, \bar{D}_i^1, \bar{D}_i^2, \bar{D}_i^3) \quad (13)$$

with blocks $\underline{D}_i^k = \delta (\tilde{\Pi}_\infty \underline{\Pi}_k^E \mathbf{H})^* + \gamma \mathbf{P}^{i,T} \mathbf{P}^i$ and $\bar{D}_i^k = \gamma \mathbf{P}^{i,T} \mathbf{P}^i - \delta (\tilde{\Pi}_\infty \bar{\Pi}_k^E \mathbf{H})^*$. Given $\tilde{\Pi}_\infty$ and δ , both obtained by solving (6) and cheirality LMIs (8), the following holds:

Proposition 3.6 *Let $\mathcal{S}c = \{(C_i, \mathcal{C}_i)\}_{i=1}^n$ be a set of putative camera-to-box correspondences (i.e. each camera center C_i is assigned to a box $\mathcal{C}_i$). If $\mathcal{S}c$'s correspondences are correct, then LMIs*

$$\mathbf{D}_i \geq \mathbf{I}, i = 1 \dots n \quad (14)$$

must be simultaneously feasible for a scalar γ and at least the true transformation matrix $\mathbf{H}$ satisfying $\mathbf{X}_j^E \sim \mathbf{H} \mathbf{X}_j$.

Proof The proof, omitted here, is along the lines of that of Proposition 3.5. It employs Finsler's lemma while relying on the fact that $\mathbf{P}^i \mathbf{C}_i = 0$ and that $\delta \mathbf{C}_i^T \Pi_\infty \bar{\Pi}_\infty^T \mathbf{C}_i > 0$. ■

When a set of points and/or camera centers are putatively assigned to bounding boxes $\mathcal{B}_j$ and/or $\mathcal{C}_i$, LMIs (12) and (14) can be simultaneously tested for feasibility. Should they be infeasible, one is guaranteed that at least one point or one camera center has wrongly been assigned to a box. Alternatively to assigning multiple points to boxes, one may use bounds on the entries of the sought matrix $\mathbf{H}$ to check whether or not a single point (or camera center)-to-box hypothesis is viable. Assuming the origin of the projective scene/cameras frame coincides with the centroid of the camera centers and SfM-deduced points, δ and $(\mathbf{H})_{44}$ can both be set to 1 (the last row of $\mathbf{H}$ being the plane at infinity - see [11] p. 526). The following corollary can be deduced:

Corollary 3.7 *Let $\underline{\mathbf{H}}$ and $\bar{\mathbf{H}}$ be the 4×4 matrices whose entries are valid, respectively, lower and upper bounds on the entries of the sought matrix $\mathbf{H}$. If a point X (resp. camera center C) is boxed by $\mathcal{B}$ (resp. $\mathcal{C}$), the LMI problem*

$$\begin{aligned} \mathbf{B} > 0 \text{ (resp. } \mathbf{D} > 0), (\mathbf{H})_{44} &= 1 \\ (\underline{\mathbf{H}})_{k\ell} < (\mathbf{H})_{k\ell} < (\bar{\mathbf{H}})_{k\ell}, k, \ell &= 1, 2 \dots 4 \end{aligned} \quad (15)$$

is feasible for a scalar γ and the true matrix $\mathbf{H}$.

3.3. Registration

We have devised two algorithms for registering 2D corresponding points across images with their 3D scanned counterparts. The first algorithm, named here SSR (Scene Structure Registration), is based on Propositions 3.5 and 3.6 and exploits the scene's structure. SSR is relatively fast, considering the problem at hand, but requires that matched 2D features have their corresponding 3D points scanned. This requirement is relaxed in our second registration method, named RR (Robust Registration), that allows a predefined number of 2D matches not to have scanned 3D counterparts. RR is based on Corollary 3.7 and considers each point-to-box assignment independently from the others. Both algorithms exploit the Branch-and-Prune (BnP) paradigm but explore different spaces. On the one hand, SSR subdivides non-empty bounding boxes to which points are assigned in order to iteratively obtain tighter boxes. This algorithm exploits the fact that scanned scenes consist of surface points and much of the explored space is void. A point that can only be assigned to an empty box indicates that the correspondence hypotheses for such assignment are surely incorrect. On the other hand, RR subdivides the space of parameters defined by the 15 bounded entries of the sought transformation matrix in order to obtain tighter bounds on this matrix while guaranteeing that at least a predefined number of points are assigned to non-empty boxes.

Initialization: In both SSR and RR algorithms, all scanned points are initially assigned to the scene's bounding box. Some applications and/or setups may allow to assign some

of the points to smaller boxes. Camera centers are initially assigned to bounding boxes obtained either from GPS information or a good guess (possibly application-specific). Because estimating H requires 5 pairs of 3D-3D correspondences (no 4 points on one plane), 5 distinct non-overlapping bounding boxes in general position are required for the boundedness of the optimization problems at hand. These could be non-overlapping boxes on 4 cameras in addition to the scene's bounding box, or boxes around 3 cameras and 2 boxes in the scene, etc. Such assumption is considered satisfied throughout. Based on Corollary 3.7, the initial bounds on the entries of H can be obtained by solving a series of SDPs. That is, for each entry $(H)_{k\ell}$, solve

$$\max_{H, \gamma} / \min_{H, \gamma} (H)_{k\ell} \text{ s.t. } B_j > 0, D_i > 0, (H)_{44} = 1. \quad (16)$$

SSR: At any given iteration of the SSR algorithm, one is given the sets $\mathcal{S}x = \{(X_j, \mathcal{B}_j)\}_{j=1}^m$ and $\mathcal{S}c = \{(C_i, \mathcal{C}_i)\}_{i=1}^n$ of respectively point-to-box and camera-to-box assignments. The set $\mathcal{S}x \cup \mathcal{S}c$ defines a node in a dynamically-built search tree. The point or camera-to-box assignments therein have feasible H and γ simultaneously satisfying their corresponding LMIs (12) and (14). Algorithm 1, that requires solving Problem 1 below, is used to reassess the boxes of all points and camera centers such that smaller boxes contribute to shrinking larger ones and all boxes best fit the scanned points within. If any box assigned to a point turns out to be empty, the branch is marked for dismissal and the hypothetical assignments are dropped. If a branch is not dismissed, then the feasible H for LMIs (12) and (14) is used to initialize a projective ICP-like refinement (discussed below). The branch with the lowest cost (19) is processed first. The box in $\mathcal{S}x$ with the longest edge is subdivided (along the latter edge) into two boxes resulting in two new branches to be explored.

Algorithm 1 $[\mathcal{S}x, \mathcal{S}c] = \text{SSR-NodeProcessing}(\mathcal{S}x, \mathcal{S}c)$

```

for each  $a \in \mathcal{S}x \cup \mathcal{S}c$  do
  ( $a \in \mathcal{S}x \cup \mathcal{S}c$  consists in a tuple  $(a.X, a.B)$ )
  Refine  $a.B$  by solving Problem 1
  if  $a \in \mathcal{S}x$  (i.e.  $a.X$  is a point) then
    if refined  $a.B$  is empty (i.e. no scanned points) then
       $\mathcal{S}x \leftarrow \emptyset$ ;  $\mathcal{S}c \leftarrow \emptyset$  (branch to dismiss)
    else
      Shrink  $a.B$  to best fit scanned points within
      Update  $a$  in  $\mathcal{S}x$ 
    end if
  else
    Update  $a$  in  $\mathcal{S}c$ 
  end if
end for
return  $[\mathcal{S}x, \mathcal{S}c]$ 

```

Problem 1: Let $\mathcal{S}x = \{(X_j, \mathcal{B}_j)\}_{j=1}^m$ and $\mathcal{S}c = \{(C_i, \mathcal{C}_i)\}_{i=1}^n$ be sets of putative, respectively, point-to-box

and camera-to-box assignments. Let $X \in \{X_j\}_{j=1}^m$ be boxed by $\mathcal{B} = \{(\underline{\Pi}_k, \bar{\Pi}_k)\}_{k=1}^3 \in \{\mathcal{B}_j\}_{j=1}^m$ for which a possibly tighter box may exist. Recalling that $\underline{\Pi}_k^E = (e_k^T - d_k)^T$ and $\bar{\Pi}_k^E = (e_k^T - \bar{d}_k)^T$, a new upper bound $\bar{d}_k$ for some fixed k can be obtained by solving

$$\begin{aligned} \max_{H, \gamma, d_k} \quad & d_k \\ \text{s.t.} \quad & (\tilde{\Pi}_\infty (e_k^T - d_k)H)^* + \gamma(SM)^T SM > I, \\ & B_j \geq I \quad j = 1 \dots m, \quad D_i \geq I \quad i = 1 \dots n. \end{aligned} \quad (17)$$

This can be solved by binary search over d_k in the range $[d_k, \bar{d}_k]$. Intuitively, this is equivalent to pushing $\underline{\Pi}_k^E$ towards $\bar{\Pi}_k^E$ until either the two planes coincide (no smaller bound on $\bar{d}_k$) or X cannot be mapped on the positive side of $(e_k^T - d_k)^T$. This latter case means that X can only be mapped on the negative side of $(e_k^T - d_k)^T$ thus making the resulting d_k the new upper bound $\bar{d}_k$. A new lower bound $\underline{d}_k$ can be obtained by solving a similar problem to (17) by minimizing d_k while $\gamma(S_j M)^T S_j M - (\tilde{\Pi}_\infty (e_k^T - d_k)H)^* > I$ and subjected to points and cameras' bounding LMIs.

RR: At any given iteration of the RR algorithm, one is given bounds on the 15 entries of H (given $(H)_{44} = 1$) and a set $\mathcal{S}x$ of point-to-box putative assignments. The set $\mathcal{S}x$ and H 's bounds define a node in a dynamically-built search tree. Algorithm 2 refines the box assigned to each point based on the bounds on H it has been provided. This algorithm returns a new set $\mathcal{S}x$ with updated boxes and, more importantly, empty box assignments taken away. The cardinality of $\mathcal{S}x$ hence provides the number of points actually assigned to non-empty boxes. The node is dropped if the number of such point-to-box assignments is below a predefined threshold or LMIs (15) are infeasible when considered simultaneously for all assignments in the refined $\mathcal{S}x$. Otherwise, the feasible H satisfying LMIs (15) due to $\mathcal{S}x$ is used to initialize the projective ICP-like refinement. The branch with the lowest cost (19) is processed first. In this case, H is branched along its longest edge thus creating two new branches (inheriting the refined $\mathcal{S}x$) to explore.

Algorithm 2 $\mathcal{S}x = \text{RR-NodeProcessing}(\mathcal{S}x, H, \bar{H})$

```

for each  $a \in \mathcal{S}x$  do
  Refine  $a.B$  by solving Problem 2
  if  $a.B$  is empty (i.e. no scanned points) then
    Remove  $a$  from  $\mathcal{S}x$ 
  else
    Update  $a$  in  $\mathcal{S}x$ 
  end if
end for
return  $\mathcal{S}x$ 

```

Problem 2: Now consider bounds on H are given and $\mathcal{B} = \{(\underline{\Pi}_k, \bar{\Pi}_k)\}_{k=1}^3$ be the box to which X is assigned. With tighter bounds on H one can obtain new tighter bounds on

X. A new upper bound $\bar{d}_k$ can be obtained by solving

$$\begin{aligned} \max_{H, \gamma, d_k} \quad & d_k \\ \text{s.t.} \quad & (\tilde{\Pi}_\infty (e_k^T - d_k) H)^* + \gamma (SM)^T SM > 0, \\ & (H)_{k\ell} < (H)_{k\ell} < (\bar{H})_{k\ell} \quad k, \ell = 1, 2 \dots 4, \\ & (H)_{44} = 1 \end{aligned} \quad (18)$$

assuming the SfM-scene and cameras centered at the origin of the projective frame. As in (17), this can be solved by binary search over d_k in the range $[d_k, \bar{d}_k]$. The largest d_k is the new $\bar{d}_k$. A lower bound $\underline{d}_k$ can very much be obtained in the same manner as discussed for Problem 1.

Termination: Both SSR and RR algorithms terminate when the cost of the projective ICP-like refinement reaches a predefined objective or when all branches have been processed (up to bound gap in the branching parameters). In the latter case the best solution is returned.

Projective ICP-like refinement: Let $\mathcal{X}_j$ be the set of scanned 3D points boxed by some box $\mathcal{B}_j$. Given initialization on H , it can be refined by minimizing the cost,

$$\mathcal{C}(H) = \sum_{i=1}^n \sum_{j=1}^m \min_{X \in \mathcal{X}_j} \mathbf{d}(x_j^i, P^i H^{-1} X^E)^2 \quad (19)$$

where $\mathbf{d}(\cdot, \cdot)$ is the Euclidean distance. This is carried out by alternating matching 3D scanned points in bounding boxes and 2D points (based on re-projection error) and re-estimating H .

4. Experiments

We tested the proposed methods using synthetic and real images. Projective reconstruction was obtained using [18] and refined via Bundle Adjustment [28] in [14] using Rabauds SfM Toolbox [21]. The algorithms were implemented in MATLAB2012a and the LMI problems were solved using the LMI Control Toolbox. All experiments were carried out on a Pentium i7/2.50GHz with 8GB RAM.

Synthetic data: We generated a set of 800 random 3D points scattered on the surface of four faces of a $20m \times 20m \times 20m$ scene box. The cameras were placed about $20 \pm 2m$ away from the scene's centroid with randomly generated rotations while looking towards the scene. 800 additional points were also generated on the surface of a hemisphere placed at a corner of the box. Of these points, 1000 were randomly selected and projected onto 512×512 images with zero-skew, 200 pix. focal length and an image-centered principal point. The projected points were imposed 0.0 to 2.0 pixels random noise (with a step of 0.4). Only 20 image points were assumed to be matched across the image sequence. The SSR method was tested by changing various parameters while conducting 50 experiments for each setup. The number of views was varied from 5 to 15

(with a step of 2) while bounding camera centers inside cubic bounding boxes (denoted Bbx) of different sizes (sides of 20cm, 2m, and 4m), with no constraints on the scene points. The number of branching was allowed to be no more than 50 to restrict the maximum processing time. The 2D projection error threshold was set to 10^{-2} .

The median time taken for various experiments against the number of bounded cameras and image noise are shown in Figure 1. Similarly, Figure 2 shows the success count over 50 experiments. 2D-to-3D registration accuracy was measured by computing the 3D registration error of all 1000 reconstructed points to the scene. Measured 3D RMS registration error is shown in Figure 3. An experiment is assumed to be successful if it produces less than 0.1 3D error. The estimated camera intrinsics and pose were compared against that of ground truth. The Euclidean projection matrix of the first camera was recovered using $P^{E1} = K^1[R^1 \ t^1] = P^1 H^{-1}$. For the evaluation, error measurement metrics for N number of experiments are defined as follows

$$\begin{aligned} \Delta f &= \sqrt{\frac{\sum_{i=1}^N (\alpha_i^1 - \alpha)^2 + (\beta_i^1 - \beta)^2}{N(\alpha^2 + \beta^2)}}, & \Delta R &= \sqrt{\frac{\sum_{i=1}^N \|r_i^1 - r\|^2}{3N}}, \\ \Delta uv &= \sqrt{\frac{\sum_{i=1}^N (u_i^1 - u)^2 + (v_i^1 - v)^2}{N(u^2 + v^2)}}, & \Delta t &= \sqrt{\frac{\sum_{i=1}^N \|t_i^1 - t\|^2}{N(\|t\|^2)}}, \end{aligned}$$

where α^1, β^1 represent two focal lengths, and (u^1, v^1) is the principal point. r^1 is a vector obtained by stacking three rotation angles in degrees. These angles are obtained from R^1 after enforcing its orthogonality. The corresponding variables without subscript represent the ground truth. The errors in camera intrinsics and pose are shown in Figure 4. The success, speed, and accuracy improve with the increase in number of views and decrease in the box size.

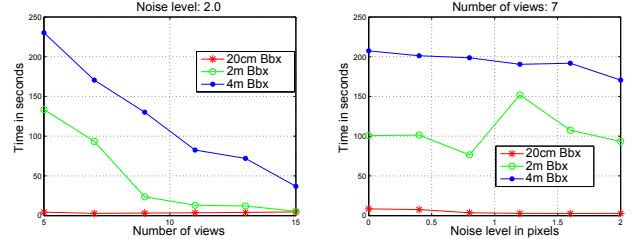


Figure 1: Time vs. number of views and noise.

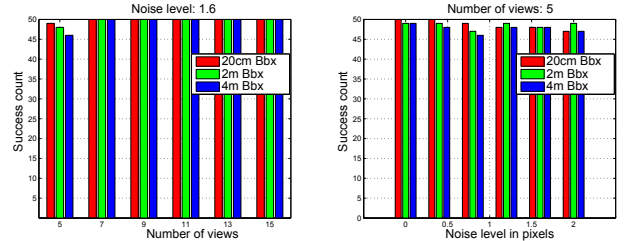


Figure 2: Success count vs. number of views and noise.

Sequence	Method	Points Bbx.		Cameras Bbx.		Time (sec)	Δf	Δuv	ΔR	Δt	3D Error
		Num.	Size (m)	Num.	Size (m)						
Fountain	SSR	0/36	-	11	2.00	78.423	0.0162	0.0275	0.2471	0.0196	0.0164
		18/36	2.00	11	2.00	3.912	0.0162	0.0275	0.2471	0.0196	0.0164
	RR	10/36	1.00	11	1.00	52.796	0.0162	0.0275	0.2471	0.0196	0.0164
	RISAG	-/4601	-	11	-	805.680	-	-	8.6825	0.1408	0.3275
	Go-ICP	-/4601	-	11	-	529.415	-	-	0.7225	0.0163	0.0348
Herz-Jesu	SSR	0/29	-	8	2.00	57.442	0.0207	0.0280	0.1441	0.0137	0.0221
		18/29	2.00	8	2.00	3.999	0.0207	0.0280	0.1441	0.0137	0.0221
	RR	5/29	1.00	8	1.00	8.212	0.0039	0.0089	0.3223	0.0061	0.0019
	RISAG	-/4024	-	8	-	160.064	-	-	17.6378	0.0570	0.1830
	Go-ICP	-/4024	-	8	-	31.254	-	-	3.2618	0.169	0.0725

Table 1: Four different methods with real data. Points Bbx. p/q means p number of points are bounded out of q sought.

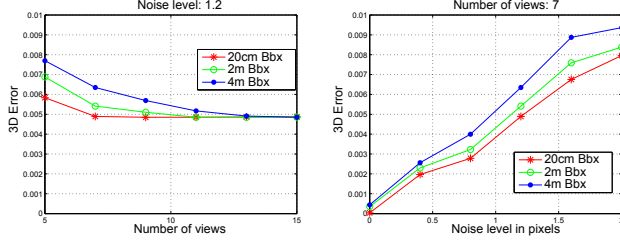


Figure 3: Registration error vs. number of views and noise.

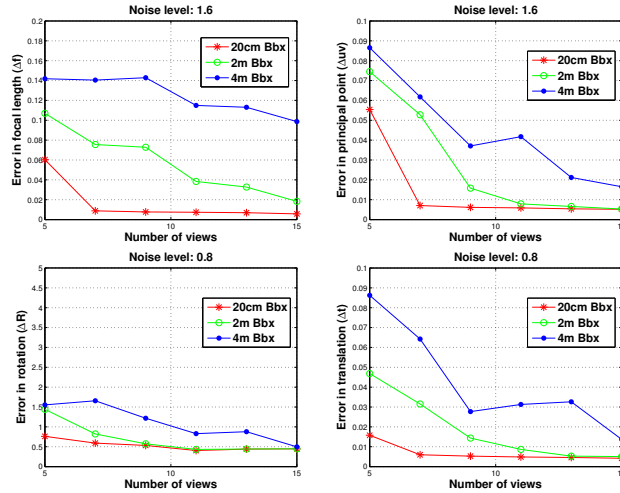


Figure 4: Intrinsic and pose errors vs. number of views.



Figure 5: Fountain: (left) 11 cameras 2m Bbx and scene, (right) estimated cameras in textured scene using SSR.

Real data: We tested our method with two real datasets: Fountain-P11 and Herz-Jesu-P8 (from [25]). These datasets consist, respectively, of 11 and 8 images of size 3072×2048 captured by a moving camera of $\alpha = 2759.5$, $\beta = 2764.2$, $u = 1520.7$ and $v = 1006.8$, along with the laser scanned



Figure 6: Herz-Jesu: (left) matched 2D features with outliers in red, (right) texture-mapped scene using RR.

3D scenes. Our results were compared against two methods: RISAG [6] and Go-ICP [30]. RISAG requires metric reconstruction, hence works only for the calibrated case. Likewise, Go-ICP requires an Euclidean reconstruction, which was obtained by upgrading the metric reconstruction using ground truth projection matrices. The metric reconstruction was obtained using openMVG [15]. The results obtained for all four methods are shown in Table 1. For qualitative analysis, estimated projection matrices were used for texture mapping. The obtained results using our methods were very accurate. These are shown in Figures 5-6 which also provide the results after further refinement using [29]. Note that a small error in pose can significantly affect the texture mapping. For the Fountain sequence, both SSR and RR converged to the same solution. RR, however, converged to a better solution for Herz-Jesu.

5. Conclusion

We have presented a novel approach for registering two or more uncalibrated cameras to a 3D scanned scene. The proposed approach only assumes point correspondences across images. Our solution allows estimating the unknown projective transformation relating the cameras to the scene and establishing 2D-3D correspondences. A LMI framework was used to overcome the image-induced point triangulation requirement. Using this framework, we have derived triangulation-free LMI cheirality conditions and LMI constraints for establishing putative correspondences between 3D boxes and 2D points. Two globally convergent algorithms, one exploiting the scene's structure and the other concerned with robustness, have been presented.

Acknowledgments

This research has been funded by the International Project NRF-ANR DrAACaR: ANR-11-ISO3-0003.

References

- [1] D. Aiger, N. J. Mitra, and D. Cohen-Or. 4-points congruent sets for robust surface registration. *ACM Transactions on Graphics*, 27(3):85, 1–10, 2008. **1**
- [2] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, New York, NY, USA, 2004. **3**
- [3] M. Chandraker, S. Agarwal, F. Kahl, D. Nister, and D. Kriegman. Autocalibration via rank-constrained estimation of the absolute quadric. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2007. **2**
- [4] M. Chandraker, S. Agarwal, D. Kriegman, and S. Belongie. Globally optimal algorithms for stratified autocalibration. *International Journal of Computer Vision (IJCV)*, pages 236–254, November 2009. **2**
- [5] S. Christy and R. Horaud. Iterative pose computation from line correspondences. In *Comput. Vis. Image Underst. (CVIU)*, pages 137–144, January 1999. **1**
- [6] M. Corsini, M. Dellepiane, F. Ganovelli, R. Gherardi, A. Fusiello, and R. Scopigno. Fully automatic registration of image sets on approximate geometry. *International Journal of Computer Vision (IJCV)*, pages 91–111, March 2013. **1, 8**
- [7] P. Finsler. Über das vorkommen definiter und semidefiniter formen in scharen quadratischer formen. *Comment. Math. Helv.*, 9, pages 188–192, 1936/37. **3**
- [8] A. Fusiello, A. Benedetti, M. Farenzena, and A. Busti. Globally convergent autocalibration using interval analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, pages 1633–1638, December 2004. **2**
- [9] A. Habed, D. Pani Paudel, C. Demonceaux, and D. Fofi. Efficient pruning lmi conditions for branch-and-prune rank and chirality-constrained estimation of the dual absolute quadric. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 493–500, 2014. **2**
- [10] R. I. Hartley. Chirality. *International Journal of Computer Vision (IJCV)*, pages 41–61, January 1998. **2**
- [11] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, second edition, 2004. **2, 5**
- [12] J. Knopp, J. Sivic, and T. Pajdla. Avoiding confusing features in place recognition. In *European Conference on Computer Vision (ECCV)*, pages 748–761, 2010. **1**
- [13] L. Liu and I. Stamos. Automatic 3d to 2d registration for the photorealistic rendering of urban scenes. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 137–143, 2005. **1**
- [14] M. A. Lourakis and A. Argyros. SBA: A software package for generic sparse bundle adjustment. In *ACM Trans. Math. Software*, pages 1–30, 2009. **7**
- [15] P. Moulon, P. Monasse, and R. Marlet. Adaptive structure from motion with a contrario model estimation. In *Asian Conference on Computer Vision (ACCV)*, pages 257–270, 2013. **8**
- [16] D. Nistér. Untwisting a projective reconstruction. *International Journal of Computer Vision (IJCV)*, pages 165–183, November 2004. **2**
- [17] D. Nister, O. Naroditsky, and J. Bergen. Visual odometry. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 652–659, 2004. **2**
- [18] J. Oliensis and R. Hartley. Iterative extensions of the sturm/triggs algorithm: Convergence and nonconvergence. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, pages 2217–2233, December 2007. **7**
- [19] D. P. Paudel, C. Demonceaux, A. Habed, and P. Vasseur. Localization of 2d cameras in a known environment using direct 2d-3d registration. In *IEEE International Conference on Pattern Recognition (ICPR)*, pages 1–6, 2014. **1**
- [20] M.-T. Pham, O. J. Woodford, F. Perbet, A. Maki, B. Stenger, and R. Cipolla. A new distance for scale-invariant 3d shape recognition and registration. In *IEEE International Conference on Computer Vision (ICCV)*, pages 145–152, 2011. **1**
- [21] V. Rabaud. Vincent’s Structure from Motion Toolbox. <http://vision.ucsd.edu/~vrabaud/toolbox/>. **7**
- [22] S. Ramalingam, S. Bouaziz, P. Sturm, and M. Brand. Geolocalization using skylines from omni-images. In *IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*, pages 23–30, 2009. **1**
- [23] T. Sattler, B. Leibe, and L. Kobbelt. Fast image-based localization using direct 2d-to-3d matching. In *IEEE International Conference on Computer Vision (ICCV)*, pages 667–674, 2011. **1**
- [24] D. Scaramuzza and F. Fraundorfer. Visual odometry, part i: The first 30 years and fundamentals. In *IEEE ROBOTICS and AUTOMATION MAGAZINE*, pages 80–92, December 2011. **2**
- [25] 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 *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8, 2008. **8**
- [26] P. Sturm. Critical motion sequences for monocular self-calibration and uncalibrated euclidean reconstruction. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1100–1105, 1997.
- [27] A. Taneja, L. Ballan, and M. Pollefeys. Registration of spherical panoramic images with cadastral 3d models. In *3DIMPVT*, pages 479–486, 2012. **1**
- [28] B. Triggs, P. Mclauchlan, R. Hartley, and A. Fitzgibbon. Bundle adjustment a modern synthesis. In *Vision Algorithms: Theory and Practice, LNCS*, pages 298–375. Springer Verlag, 2000. **7**
- [29] P. Viola and W. M. Wells, III. Alignment by maximization of mutual information. *International Journal of Computer Vision (IJCV)*, pages 137–154, September 1997. **1, 8**
- [30] J. Yang, H. Li, and Y. Jia. Go-icp: Solving 3d registration efficiently and globally optimally. In *IEEE International Conference on Computer Vision (ICCV)*, pages 1457–1464, December 2013. **1, 8**