Optimal Transformation Estimation with Semantic Cues

Danda Pani Paudel
Computer Vision Laboratory
D-ITET, ETH Zurich
paudel@vision.ee.ethz.ch

Adlane Habed
ICube Laboratory
CNRS, University of Strasbourg
adlane.habed@icube.unistra.fr

Luc Van Gool
Computer Vision Laboratory
D-ITET, ETH Zurich
vangool@vision.ee.ethz.ch

Abstract

This paper addresses the problem of estimating the geometric transformation relating two distinct visual modalities (e.g. an image and a map, or a projective structure and a Euclidean 3D model) while relying only on semantic cues, such as semantically segmented regions or object bounding boxes. The proposed approach differs from the traditional feature-to-feature correspondence reasoning: starting from semantic regions on one side, we seek their possible corresponding regions on the other, thus constraining the sought geometric transformation. This entails a simultaneous search for the transformation and for the region-to-region correspondences. This paper is the first to derive the conditions that must be satisfied for a convex region, defined by control points, to be transformed inside an ellipsoid. These conditions are formulated as Linear Matrix Inequalities and used within a Branch-and-Prune search to obtain the globally optimal transformation. We tested our approach, under mild initial bound conditions, on two challenging registration problems for aligning: (i) a semantically segmented image and a map via a 2D homography; (ii) a projective 3D structure and its Euclidean counterpart.

1. Introduction

Estimating a geometric transformation relating two overlapping instances of a scene, be it images or 3D point clouds, typically relies on establishing cross-instance (e.g. 2D-2D, 2D-3D or 3D-3D) correspondences of low-level features such as points [38], lines [6], planes [18], skylines [35], or scene constraints [22]. The success of establishing such correspondences may be undermined by the absence of such features or by the difficulty of matching them. Such difficulty would be high in case of very different modalities (e.g. matching an image and a map). Methods, such as [13, 32], do not require initial correspondences yet they rely on detecting such low-level features.

Mainly spurred by significant advances in Machine Learning [21, 20], detecting higher level features (objects, regions, and their semantic labels) is nowadays reaching unparalleled levels of performance in a variety of imaging modalities including 2D images [20], videos [16], and 3D point clouds [41]. In many applications, the use of traditional hand-crafted features (such as SIFT) is outperformed by that of high-level features learned by Neural Networks.

As semantic labels for 2D images and 3D data can be obtained quite accurately nowadays [23, 24, 48], it is becoming particularly appealing to use these for solving geometric problems. Matching high-level features and estimating the underlying aligning transformation is a challenging problem that we address in this paper. Supporting transformation estimation with semantic cues has the potential to improve the success rate, speed and accuracy of the process.

Owing to their success in many applications, Machine Learning techniques have been tried to solve a wide range of problems in Computer Vision, including that of learning geometric parameters directly from images [12, 45, 44, 25]. This said, attempts towards the latter have only met with limited success. The results were often not on par with those of model-based methods. Although model-based transformation estimation may potentially benefit from semantics, methods to do so have obtained little attention.

Semantic cues have been successfully exploited to support several 3D vision tasks such as, keypoint matching [19], 3D reconstruction [10, 39], and robot navigation [7]. In the context of transformation estimation, the method in [11] learns and estimates the relative homography between a pair of images using deep convolutional neural networks. For producing a semantic map from multi-view street-level imagery, [40] considers a homography relationship between semantically labeled pixels of the ground plane in one image and a sub-set in another. These methods fully rely on the ability of learning methods and do not provide guaranties on the optimality of the estimated parameters. In this regard, a global method for uncalibrated 2D-3D alignment was proposed in [31]. The method relies on multi-convex conditions for associating corresponding multi-view 2D pixels and 3D bounding boxes. Yet, that

approach does not allow one to establish correspondences between high-level features in both imaging modalities.

In this paper, we address the problem of the globally optimal estimation of geometric transformations using only semantic cues. These features could materialize in the form of regions with semantic labels or as bounding boxes of previously detected/known scene parts. Regions of interest or bounding boxes are represented by polytopes in one instance of the data (the source) and by ellipsoids in the other instance (the target). Based on this representation, we propose Linear Matrix Inequality (LMI) conditions that must be satisfied by a projectively transformed polytope from the source to lie within an ellipsoid in the target. We also propose a convex optimization formulation to estimate a covering ellipsoid around the set of all transformed polytopes emanating from all possible transformations within given parameter bounds. The covering ellipsoid, the so-called Löwner-John ellipsoid, is estimated using the polynomial Sum-of-Squares (SoS) theory. Unlike the multi-convex formulation of [31] for bounding-box estimation, our covering ellipsoid estimation problem is purely convex.

Based on the proposed polytope-ellipsoid assignment conditions, we use parameter bounds to infer correspondences. Correspondences contradicting the semantic cues or that are geometrically inconsistent are eliminated. Geometrically consistent correspondences not contradicting the semantic cues are further investigated. Regions in the target data are shrunk and subdivided leading to new correspondences. These are then used to estimate new parameter bounds. A dynamically Branch-and-Prune search tree is built by recursively repeating this process. The nodes of the tree are potential correspondences to prune or to investigate. Using some initial bounds on transformation parameters, we applied our approach on two challenging problems: (i) Image-to-map registration to align an image and a semantic map via a 2D projective homography, (ii) Projective 3D-to-3D registration of an uncalibrated reconstruction.

Image-to-map registration: Registering an image to a map is a challenging problem in which establishing low-level feature correspondences is not possible. Exploiting semantic cues might as well be the only plausible way for achieving this task. To our knowledge, there exists no globally optimal method that registers images to maps using semantic cues. Existing methods perform image localization either by image retrieval [37] or by direct learning [44]. These methods are non-optimal and purely data driven. In our experiments, we push the challenge of image-to-map registration one step further by not relying on landmarks (such as unique buildings/objects) for correspondences.

Projective 3D-to-3D registration: When cameras are uncalibrated, the transformation relating the Structure-from-Motion (SfM) reconstruction to its Euclidean counterpart is a 3D projective homography. The projective 3D-to-3D registration problem may appear whenever a Euclidean reconstruction and another from uncalibrated SfM need to be combined. One may attempt to solve this problem in two steps: camera auto-calibration for projective-to-metric upgrade followed by Metric-to-Euclidean registration. As argued in [31], camera auto-calibration methods are known to be impractical due to their sensitivity to noise, critical motions, and large variations in the camera parameters. Using the proposed method, we accurately estimate this transformation by exploiting semantic cues and bounds on camera centers. In this context, our problem is similar to [31]. However, our method performs significantly faster under the same conditions while obtaining better results.

2. Background and Notations

This section is devoted to an overview of the convex optimization machinery we employ in our solution to the problem of optimal transformation estimation. The notations used throughout the paper are also introduced herein. For instance, when dealing with matrices, $\mathsf{A} \succ 0$ (resp. $\mathsf{A} \succeq 0$) means that A is a symmetric positive definite (resp. semidefinite) matrix. Should A be a $m \times n$ matrix, we refer to the upper-left $(m-1) \times (n-1)$ block of A by $\hat{\mathsf{A}}$. The j^{th} element of a vector x is denoted by x_j.

Linear Matrix Inequalities: Considering $\mathsf{A}(\mathsf{x})$ to be a matrix whose entries are affine functions of a vector x, the problem of finding a realization of x such that $\mathsf{A}(\mathsf{x}) \succ 0$ or $\mathsf{A}(\mathsf{x}) \succeq 0$ is a Linear Matrix Inequality (LMI) feasibility problem. Minimizing or maximizing a linear cost subject to LMI constraints is a Semidefinite Programming problem (SDP). LMI and SDP problems can efficiently be solved using interior-point methods [3]. Furthermore, when dealing with convex nonlinear matrix inequalities, Schur's complement lemma, a fundamental tool in the theory of LMIs, may be used to turn such inequalities to equivalent LMIs:

Lemma 2.1 (Schur Complement [14]) *For a symmetric block-partitioned matrix* $\mathsf{D} = \begin{bmatrix} \mathsf{A} & \mathsf{B} \\ \mathsf{B}^\top & \mathsf{C} \end{bmatrix}$ *and its Schur complement* $\mathsf{D}/\mathsf{A} = \mathsf{C} - \mathsf{B}^\top \mathsf{A}^{-1} \mathsf{B}$, $\mathsf{D} \succeq 0 \iff \mathsf{A} \succeq 0, \mathsf{D}/\mathsf{A} \succeq 0$.

Another important result in this theory is the so-called S-Procedure: a tool for verifying whether or not one quadratic inequality is a consequence of another quadratic inequality:

Lemma 2.2 (S-Procedure [43]) *Let* A_0 *and* A_1 *be symmetric matrices.* $\mathsf{z}^\top \mathsf{A}_0 \mathsf{z} \leq 0$ *holds for all* z *such that* $\mathsf{z}^\top \mathsf{A}_1 \mathsf{z} \leq 0$ *if there exists* $\lambda \geq 0$ *such that* $\lambda \mathsf{A}_1 \succeq \mathsf{A}_0$.

Ellipsoids: Without loss of generality, an ellipsoid $\mathcal{E}$ in a $(d-1)$-dimensional space can be represented by a $d \times d$ matrix $\mathsf{Q} \succeq 0$ whose $(d-1) \times (d-1)$ upper-left block $\hat{\mathsf{Q}}$ satisfies $\hat{\mathsf{Q}} \succ 0$. Using homogeneous coordinate vectors, in which points in $(d-1)$-space are represented by $\mathsf{z} \in \mathbb{R}^d$, $\mathcal{E}$

is defined by $\mathcal{E} = \{\mathsf{z} : \mathsf{z}^\top \mathsf{Q}\mathsf{z} \leq z_d^2\}$. One is often interested in finding the so-called Löwner-John ellipsoid: the tightest ellipsoid that covers a given set of points:

Definition 2.3 (Löwner-John ellipsoid [17]) *Löwner-John ellipsoid of a compact and non-empty set $\mathcal{S} \subseteq \mathbb{R}^d$ is the minimum volume ellipsoid $\mathcal{E}$ that covers $\mathcal{S}$.*

In general, finding the Löwner-John ellipsoid is a NP-complete problem [29]. However, should $\mathcal{S}$ be convex, the volume of an ellipsoid $\mathcal{E}$ being proportional to $\sqrt{\det(\hat{\mathsf{Q}}^{-1})}$ [4] (p.48), the Löwner-John ellipsoid can be obtained by solving the concave maximization problem:

$$\begin{aligned} \underset{\mathsf{Q}}{\text{maximize}} \quad & \log\det \hat{\mathsf{Q}} \\ \text{s.t.} \quad & \mathsf{z}^\top \mathsf{Q}\mathsf{z} \leq z_d^2, \ \forall \mathsf{z} \in \mathcal{S}, \ \hat{\mathsf{Q}} \succ 0, \ \mathsf{Q} \succeq 0. \end{aligned} \tag{1}$$

This is again a problem for which the optimal solution can be obtained using interior point methods [3].

Polynomial Sum-of-Squares: More general results, which we also use in this work, on the characterization of nonlinear polynomial inequalities, have to do with the theory of polynomial Sum-of-Squares (SoS).

Definition 2.4 (SoS) *Let $\mathbb{R}[\mathsf{x}]$ be the ring of polynomials parameterized by variables $\mathsf{x} \in \mathbb{R}^n$ with real valued coefficients. A polynomial $f(\mathsf{x}) \in \mathbb{R}[\mathsf{x}]$ is Sum-of-Squares (SoS), if there exist polynomials $f_i(\mathsf{x}) \in \mathbb{R}[\mathsf{x}]$ such that $f(\mathsf{x}) = \sum_i f_i(\mathsf{x})^2$.*

In general, establishing the nonnegativity of polynomials is NP-hard while testing whether it is SoS is a LMI feasibility problem involving the so-called Gram matrix of the polynomial. Not all nonnegative polynomials are SoS but, fortunately enough, for some classes, such as quadratic polynomials, nonnegativity and SoS are equivalent:

Theorem 2.5 (Nonnegativity and SoS [15]) *A second degree polynomial is nonnegative, iff it is SoS.*

Definition 2.6 (Gram matrix [33]) *For a second degree polynomial $p(\mathsf{x}) \in \mathbb{R}[\mathsf{x}]$, the matrix G such that $p(\mathsf{x}) = \begin{bmatrix}\mathsf{x}^\top & 1\end{bmatrix} \mathsf{G} \begin{bmatrix}\mathsf{x} \\ 1\end{bmatrix}$ is a Gram matrix of $p(\mathsf{x})$.*

Theorem 2.7 (SoS and Gram matrix [5, 33]) *The polynomial $p(\mathsf{x})$ is SoS iff there exists a Gram matrix $\mathsf{G} \succeq 0$.*

The following result (an extension of the S-lemma) allows one to test the positivity of a quadratic polynomial within some predefined parameters interval:

Theorem 2.8 (Polynomial within bounds [32]) *A second degree polynomial $f(\mathsf{x})$ is positive within the interval $[\underline{\mathsf{x}}, \overline{\mathsf{x}}]$ if there exist non-negative scalars σ_k such that*

$$p(\mathsf{x}) = f(\mathsf{x}) - \sum_{k=1}^{n} g_k(\mathsf{x})\sigma_k \tag{2}$$

is SoS, for $g_k(\mathsf{x}) = (x_k - \underline{x}_k)(\overline{x}_k - x_k)$. More importantly, as the size of the interval tends to zero, $p(\mathsf{x})$ is guaranteed to be a SoS, if $f(\mathsf{x})$ is positive within that interval.

3. Transformation Estimation w/ Semantics

The transformation estimation method proposed in this paper, and outlined in Section 3.3, relies on exploring potential correspondences inferred by bounds on the parameters of the sought transformation matrix. At each iteration, bounds on the transformation parameters are considered. The idea is to characterize the region in the target data covering all possible mappings of a point or region from the source data. The resulting regions in the target data are then checked for semantic consistency against their source counterparts. If the semantics in the target and source data are consistent, then the ellipsoids are re-estimated to better fit the region/object in the target data and the parameters bounds. The correspondences thus obtained are further checked for geometric consistency: i.e. whether they may emanate from applying a common transformation.

3.1. Bounded Geometric Transformations

Consider a transformation matrix $\mathsf{T}(\mathsf{x}) \in \mathbb{R}^{d\times r}$, linearly parameterized by $\mathsf{x} \in \mathbb{R}^n$. With a given $\mathsf{T}(\mathsf{x})$, a point from the source data, with homogeneous coordinate vector y, is mapped into a point with homogeneous coordinate vector $\mathsf{z} \simeq \mathsf{T}(\mathsf{x})\mathsf{y}$ in the target data. Typically, registering the source and target data requires estimating the unknown parameters x. This, however, requires correspondences between z and y to be established. In our work, such correspondences are unknown. Instead, we consider that, because $\mathsf{T}(\mathsf{x})$ is dependent on x, the location z corresponding to some given y is a function of x. Hence, we write:

$$\mathsf{z}(\mathsf{x}) = \mathsf{T}(\mathsf{x})\mathsf{y}. \tag{3}$$

We define the set $\mathcal{S}_b$ of all possible points in the target data that y can be mapped into when considering all possible values of x in the interval $\mathsf{x} \in [\underline{\mathsf{x}}, \overline{\mathsf{x}}]$

$$\mathcal{S}_b = \{\mathsf{z} : \mathsf{z} \simeq \mathsf{z}(\mathsf{x}), \ \mathsf{x} \in [\underline{\mathsf{x}}, \overline{\mathsf{x}}]\}. \tag{4}$$

We are interested in a convex characterization of the region defined by $\mathcal{S}_b$. To this end, we seek the Löwner-John ellipsoid, the ellipsoid with minimum volume, $\mathcal{E}_b = \{\mathsf{z} : \mathsf{z}^\top \mathsf{Q}_b \mathsf{z} \leq z_d^2, \ \mathsf{z} \in \mathcal{S}_b\}$ where Q_b is the solution of (1) when $\mathcal{S} = \mathcal{S}_b$. However, this problem involves nonconvex inequalities of the form $\mathsf{z}(\mathsf{x})^\top \mathsf{Q}\mathsf{z}(\mathsf{x}) \leq z_d(\mathsf{x})^2$ in which both Q and x are unknown. As an alternative, we propose a relaxed formulation of this problem. Denoting by $f_b(\mathsf{x})$ the polynomial

$$f_b(\mathsf{x}) = z_d(\mathsf{x})^2 - \mathsf{z}(\mathsf{x})^\top \mathsf{Q}\mathsf{z}(\mathsf{x}) \tag{5}$$

the problem turns into:

Problem 3.1 *Find the smallest ellipsoid $\tilde{\mathcal{E}}_b$ such that $p_b(\mathsf{x}) = f_b(\mathsf{x}) - \sum_{k=1}^{n} g_k(\mathsf{x})\sigma_k$ is SoS, for $g_k(\mathsf{x}) = (x_k - \underline{x}_k)(\overline{x}_k - x_k)$ and scalars σ_k, $k = 1, \ldots, n$.*

Note that, based on Theorem 2.8, if $p_b(\mathsf{x})$ is SoS then $\mathsf{z}(\mathsf{x})^\top \mathsf{Q}\mathsf{z}(\mathsf{x}) \leq z_d(\mathsf{x})^2 \ \forall \mathsf{x} \in [\underline{\mathsf{x}}, \overline{\mathsf{x}}]$. The volume of an ellipsoid being proportional to $\sqrt{\det(\hat{\mathsf{Q}}^{-1})}$, we now state the following proposition without further proof:

Proposition 3.2 *Consider the Gram matrix $\mathsf{G}(\mathsf{Q}, \sigma_1, \sigma_2, \ldots, \sigma_n)$ of the polynomial $p_b(\mathsf{x})$. The smallest ellipsoid $\tilde{\mathcal{E}}_b$, optimal solution of Problem 3.1, can be obtained by solving the concave maximization problem:*

$$\begin{aligned} \underset{\mathsf{Q}, \sigma_1, \sigma_2, \ldots, \sigma_n}{\text{maximize}} \quad & \log\det \hat{\mathsf{Q}} \\ \text{s.t.} \quad & \mathsf{G}(\mathsf{Q}, \sigma_1, \sigma_2, \ldots, \sigma_n) \succeq 0, \\ & \mathsf{Q} \succeq 0, \ \hat{\mathsf{Q}} \succ 0, \\ & \sigma_k \geq 0, \ k = 1, 2 \ldots, n. \end{aligned} \tag{6}$$

As in (1), the optimization problem (6) can efficiently be solved using interior-point methods [3].

Now let us assume that several points y_i, $i = 1, 2 \ldots, p$ from the source data are mapped onto $\mathsf{z}_i(\mathsf{x}) = \mathsf{T}(\mathsf{x})\mathsf{y}_i$ in the target data. Furthermore, assume that points $\mathsf{z}_i(\mathsf{x})$, $i = 1, 2, \ldots, p$, are correctly assigned to ellipsoids. Then, the estimation of minimum volume ellipsoid associated with a yet-to-be-assigned point $\mathsf{z}(\mathsf{x}) = \mathsf{T}(\mathsf{x})\mathsf{y}$ may benefit from these additional constraints to obtain a tighter ellipsoid:

Proposition 3.3 *For a given set of point-to-ellipsoid correspondences $\{(\mathsf{z}_i(\mathsf{x}), \mathcal{E}_i)\}_{i=1}^{p}$, consider polynomials $f_i(\mathsf{x})$ constructed as in (5) from these correspondences and with known Q_i, $i = 1, 2, \ldots p$.*

Based on the S-procedure Lemma 2.2, a tighter ellipsoid $\tilde{\mathcal{E}}_b$ around $\mathsf{z}(\mathsf{x}) = \mathsf{T}(\mathsf{x})\mathsf{y}$, supported by point-to-ellipsoid correspondences, can be estimated by solving:

$$\begin{aligned} \underset{\mathsf{Q}, \sigma_k, \tau_i}{\text{maximize}} \quad & \log\det \hat{\mathsf{Q}} \\ \text{s.t.} \quad & \mathsf{G}_b(\mathsf{Q}, \sigma_1, \sigma_2, .., \sigma_n) - \sum_{i=1}^{p} \tau_i \mathsf{G}_i(\mathsf{Q}_i) \succeq 0, \\ & \mathsf{Q} \succeq 0, \ \hat{\mathsf{Q}} \succ 0, \ \sigma_k \geq 0, \ k = 1, 2, \ldots, n, \\ & \tau_i \geq 0, \ i = 1, 2, \ldots, p. \end{aligned} \tag{7}$$

This is a concave maximization problem where G_b and G_e are the Gram matrices of $p_b(\mathsf{x})$ and $f_i(\mathsf{x})$, respectively.

Remark 3.4 *For $\tau_i = 0, \forall i$, (7) is equivalent to (6). This means that the solution of (7) is at least as good as that of (6). For any $\mathsf{x} \in [\underline{\mathsf{x}}, \overline{\mathsf{x}}]$ that satisfies the assignment $\{(\mathsf{z}_i(\mathsf{x}), \mathcal{E}_i)\}_{i=1}^{p}$, $f_i(\mathsf{x})$ are always nonnegative. Therefore, the solution of (7) never violates the given assignments. On the contrary, these additional assignment constraints help restricting the set $\mathcal{S}_b$ by eliminating the regions not respected by the given assignments. This can lead only to a tighter estimation of $\tilde{\mathcal{E}}_b$.*

Finally, we consider the case in which multiple points y_i, $i = 1, 2 \ldots, p$, from the source data are mapped onto $\mathsf{z}_i(\mathsf{x}) = \mathsf{T}(\mathsf{x})\mathsf{y}_i$ in the target. None of these $\mathsf{z}_i(\mathsf{x})$ are assigned to any ellipsoid and the interest here is to estimate a single minimum volume ellipsoid covering all the points $\mathsf{z}_i(\mathsf{x})$, $i = 1, 2, \ldots, p$, $\forall \mathsf{x} \in [\underline{\mathsf{x}}, \overline{\mathsf{x}}]$:

Proposition 3.5 *Consider a given set of correspondences $\{(\mathsf{z}_i(\mathsf{x}), \mathcal{E})\}_{i=1}^{p}$ between multiple points and a single ellipsoid. Consider the Gram matrices $\mathsf{G}_i(\mathsf{Q}, \sigma_k^i)$ of the polynomial $p_b^i(\mathsf{x})$ constructed as in the formulation of Problem 3.1 for the points $\mathsf{z}_i(\mathsf{x})$, $i = 1, 2, \ldots, p$, $\forall \mathsf{x} \in [\underline{\mathsf{x}}, \overline{\mathsf{x}}]$, and a single unknown matrix Q. The minimum volume ellipsoid $\hat{\mathcal{E}}_b$ covering $\mathsf{z}_i(\mathsf{x})$, $i = 1, 2, \ldots, p$, $\forall \mathsf{x} \in [\underline{\mathsf{x}}, \overline{\mathsf{x}}]$, is obtained by solving the concave maximization problem:*

$$\begin{aligned} \underset{\mathsf{Q}, \sigma_k^i}{\text{maximize}} \quad & \log\det \hat{\mathsf{Q}} \\ \text{s.t.} \quad & \mathsf{G}_i(\mathsf{Q}, \sigma_k^i) \succeq 0, \\ & \mathsf{Q} \succeq 0, \ \hat{\mathsf{Q}} \succ 0, \\ & \sigma_k^i \geq 0, \ i = 1, 2, \ldots, p, \ k = 1, 2, \ldots, n. \end{aligned} \tag{8}$$

3.2. Geometric Consistency

When bounds on the transformation parameters are considered, Proposition 3.2 allows one to estimate the smallest ellipsoid, in the target data, containing all potential mappings of a point from the source data. Proposition 3.3 allows to additionally characterize this ellipsoid from available point-to-ellipsoid assignments. Proposition 3.5 allows one to estimate the ellipsoid covering the mappings of several points from the source data: this is particularly useful when such points are the vertices of a polytope delimiting a region/object in the source data. The ellipsoids obtained using these propositions may be re-estimated to better fit the target data. It is then necessary to check whether the resulting correspondences are geometrically consistent, i.e. there exists a common transformation leading to them.

Definition 3.6 *A set of point-to-ellipsoid putative assignments $\{\mathsf{z}_i(\mathsf{x}), \mathcal{E}_i\}_{i=1}^{p}$ is said to be geometrically consistent if there exists x such that each point with coordinates $\mathsf{z}_i(\mathsf{x})$ lies inside its associated ellipsoid $\mathcal{E}_i$.*

Proposition 3.7 *A set of point-to-ellipsoid assignments $\{\mathsf{z}_i(\mathsf{x}), \mathcal{E}_i\}_{i=1}^{p}$ is geometrically consistent iff there exists $\mathsf{x} \in \mathbb{R}^n$ such that LMIs*

$$\begin{bmatrix} \delta_i z_d(\mathsf{x}) \hat{\mathsf{Q}}_i^{-1} & \hat{\mathsf{z}}_i(\mathsf{x}) \\ \hat{\mathsf{z}}_i(\mathsf{x})^\top & \delta_i z_d(\mathsf{x}) \end{bmatrix} \succeq 0 \ \ \forall i, \ \mathsf{x} \in [\underline{\mathsf{x}}, \overline{\mathsf{x}}]. \tag{9}$$

are simultaneously feasible for $\delta_i = \pm 1$. *In (9)* $\hat{\mathsf{z}}_\mathsf{i}(\mathsf{x})$ *refers to the* $(d-1)$*-dimensional vector such that* $\mathsf{z}_\mathsf{i}(\mathsf{x})^\top = (\hat{\mathsf{z}}_\mathsf{i}(\mathsf{x})^\top \; z(\mathsf{x})_d)$ *and* $\hat{\mathsf{Q}}_i$ *is the* $(d-1)\times(d-1)$ *upper-left block of* Q_i.

Proof Lemma 2.1 demonstrates that (9) is equivalent to $\hat{\mathsf{z}}_\mathsf{i}(\mathsf{x})^\top \hat{\mathsf{Q}}_\mathsf{i} \hat{\mathsf{z}}_\mathsf{i}(\mathsf{x}) \leq \delta_i^2 z_d(\mathsf{x})$. Hence, the simultaneous feasibility of the LMIs means that there exists a x mapping each point $\mathsf{z}_\mathsf{i}(\mathsf{x})$ to its designated ellipsoid. ∎

Due to the presence of noise and/or outliers, the optimization problem (9) may turn out to be infeasible for given parameter intervals and assignments. Under such circumstances, we suggest to consider the following remark to obtain a closest feasible solution.

Remark 3.8 *In the absence of a strictly feasible solution, the closest feasible solution within the parameter intervals can be obtained by relaxing the volume of assigned ellipsoids. Such solution can be obtained by solving the following convex optimization problem:*

$$\begin{aligned} \underset{s_i,\mathsf{x}}{\text{minimize}} \quad & \sum_i s_i \\ \text{s.t.} \quad & \begin{bmatrix} s_i + \delta_i z_d(\mathsf{x})\hat{\mathsf{Q}}_i^{-1} & \hat{\mathsf{z}}_\mathsf{i}(\mathsf{x}) \\ \hat{\mathsf{z}}_\mathsf{i}(\mathsf{x})^\top & s_i + \delta_i z_d(\mathsf{x}) \end{bmatrix} \geq 0 \qquad (10) \\ & s_i \geq 0, \;\; \forall i, \;\; \mathsf{x} \in [\underline{\mathsf{x}}, \overline{\mathsf{x}}]. \end{aligned}$$

3.3. The BnP Algorithm

We have devised an optimal transformation estimation method using our formulations within a Branch-and-Prune (BnP) search. Our method relies on a small set of reliable semantic cues to bound the problem as well as to speed up the processing. We assume that at least a minimal set of control points, say $\mathcal{S} \subset \mathcal{V} = \{\mathsf{y}_i\}_{i=1}^p$, in one modality from different semantic cues are detected. We refer to these as the support points. In our BnP search, subdivision is carried out by dividing the regions that the support points are assigned to. The branching process progressively reduces the regions to which the support points can be assigned. After obtaining the subdivided regions, we fit ellipsoids around them using (1). While doing so, we make use of only the sparsely selected boundary points. Once the ellipsoids for all the support points are estimated, we induce the ellipsoids for the rest of the points using Proposition 3.3. If the induced ellipsoid does not cover the sought semantic cues, it indicates that the assignment of the support points, in the current branch, is surely incorrect. Therefore, the branch is pruned. Once the ellipsoids for all the control points are estimated, we shrink the ellipsoids corresponding to the support points. With newly shrunk ellipsoids, we test the assignments feasibility using Proposition 3.7. If the problem is feasible for all the assignments, we re-estimate the parameters' bounds $\mathcal{B} = [\underline{\mathsf{x}}, \overline{\mathsf{x}},]$ by solving for each x_k as follows:

$$\begin{aligned} \min_{\mathsf{x}}/\max_{\mathsf{x}} \quad & x_k \\ \text{s.t.} \quad & \begin{bmatrix} \delta_i z_d(\mathsf{x})\hat{\mathsf{Q}}_i^{-1} & \hat{\mathsf{z}}_\mathsf{i}(\mathsf{x}) \\ \hat{\mathsf{z}}_\mathsf{i}(\mathsf{x})^\top & \delta_i z_d(\mathsf{x}) \end{bmatrix} \geq 0, \;\; \forall i, \qquad (11) \\ & \mathsf{x} \in [\underline{\mathsf{x}}, \overline{\mathsf{x}}]. \end{aligned}$$

After updating $\mathcal{B}$, we compute the cost for a feasible x to assign all points on one side to the closest points, on the other, with the sought semantics and lying inside their respective ellipsoids. Denoting the Euclidean distance by $\mathbf{d}(.,.)$, this cost is

$$\xi(\mathsf{x}) = \sum_i \min_{e \in \mathcal{E}_i} \mathbf{d}(\mathsf{e}, \mathsf{T}(\mathsf{x})\mathsf{y}_i). \qquad (12)$$

If the feasibility test fails, the cost is computed for the solution obtained from Remark 3.8, and this branch is pruned after recording the solution. In the next step, the branch with the lowest cost is selected and the support ellipsoid with the largest volume is divided into two new ellipsoids (each for a new branch) by slicing it along the smallest variation direction. This process is repeated recursively until the desired solution is obtained. In each iteration, every node is processed using Algorithm 1.

Algorithm 1 $[\mathcal{B},\mathcal{S},\mathcal{V},\xi,\eta]$ = NodeProcessing($\mathcal{B},\mathcal{S},\mathcal{V}$)

1. Induce $\mathcal{E}_b$ for $\mathsf{y} \in \mathcal{V}$ with $\mathcal{B}$ and $\mathcal{S}$ (Proposition 3.3/3.5).
2. If any $\mathcal{E}_b$ is empty, set $\eta = 0$ (for pruning).
3. Shrink $\mathcal{E}_b$ for all $\mathsf{y} \in \mathcal{S}$.
4. Test the feasibility using Proposition 3.7.

$\Longrightarrow$ If a feasible solution exists, compute ξ using (12), update $\mathcal{B}$ using (11) and set $\eta = 1$ (to continue).

$\Longrightarrow$ Otherwise, compute ξ for x from (10) and set $\eta = 0$.

Initialization: For image-to-map registration, we assume that at least 4 support points belonging to two non-overlapping regions are provided. Similarly, at least 5 support points belonging to 3 different regions are assumed to be known for projective 3D-to-3D registration. The initial parameter bounds are either derived from vague knowledge on the acquisition setups, or using (11) from given region correspondences. More details are provided in Section 4.
Termination: To ensure optimality, BnP explores branches until all control points are assigned to the ellipsoids of volume smaller than a threshold. For given semantic cues, there may exist multiple configurations where all control points are assigned to the desired volume ellipsoids. Therefore, the algorithm can be terminated when (i) all branches are pruned, or (ii) the cost $\xi(\mathsf{x})$ is below a predefined objective, or (iii) all control points are assigned to the desired volume ellipsoids. When the cost $\xi(\mathsf{x})$ reaches the predefined objective, the obtained solution is said to be optimal.

4. Experiments

We tested our transformation estimation method on two registration problems: 2D-2D image-to-map and projective 3D-3D. All reported experiments are conducted on real datasets. The semantic cues were obtained in two different ways: (1) manual annotation, (2) automatic detection. For automatic detection, we used the state-of-art methods for object detection [36] and semantic segmentation [47]. Our algorithm is implemented in MATLAB2015a and all the optimization problems are solved using MOSEK [27].

4.1. Image-to-Map Registration

We used images acquired by three roundshot cameras [1] mounted on the top of buildings in Zürich, Switzerland. These cameras provide 360° panoramic views with high quality 2051×9002 pix images. The local map, covering about $14km^2$, was downloaded from OpenStreetMap [2].

We registered images to the map with 2D homographies, whose initial bounds were obtained by inferring some vague knowledge about the acquisition setup. We assumed that the cameras were mounted at a height between $20 - 50m$, they looked roughly towards the ground plane, ground plane's normal coordinates are close to $(0, 0, 1)$, and a region of size $50 \times 50m^2$ or $100 \times 100m^2$ that includes the camera is known. Furthermore, we also assume that the detected river parts (in images) are located in of $1 \times 1km^2$ known map region, with camera search region in the center. Similarly, the region for tramways is set to $400 \times 400m^2$. Given the form of the sought homography $\mathsf{H} = \mathsf{R} - \frac{1}{d}\mathsf{t}\mathsf{n}^\top$, we first establish the individual bounds R, t, and d using only the above mentioned assumptions. Then, the bounds on the entries of H are estimated using interval arithmetics [26].

First, we present the effect of different bound gaps on ellipse estimation. Figure 1 shows, for a selected set of image points, the ellipses induced on the map using Proposition 3.2 with 1.0 bound gap homographies. The corresponding feature points are also displayed on the map. The expected result of registration is the map warped to fit the image. It is shown in the same figure using the ground-truth homography matrix. The effect on ellipse estimation with various bound gaps is shown in Figure 2. One can observe that the ellipse areas decrease with the decreasing bound gap. It comes as no surprise that points near infinity have higher uncertainties than those close to the camera.

Experiments with manual image annotations and others with automatic detection from images were carried out independently. For registration supported by manual annotation, two semantic regions with labels "river" and "tramway" were chosen on the images. Similarly, the automatic detection consists of three categories: "car", "tram", and "boat" obtained using [36]. We use the assumption that cars, trams, and boats can be found only on roads, tramways, and rivers, respectively. The corresponding labels on the map were extracted from the OpenStreetMap data. Both manual and automatic semantic labels used for one dataset are also shown in Figure 3. In the same figure, we also show the qualitative results obtained by our method before and after refinement (refer to Remark 3.8).

The final results of our method for all cameras are shown in Table 1, after decomposing the homography into more geometrically meaningful parameters. It can be seen that the accuracy of the rotation, translation, and normal estimation is very satisfactory. Note that the semantic labels from manual source has only two categories, whereas, the automatic source consists of three. To analyze the behavior of our BnP search, we report the number of nodes and remaining areas (inside the ellipses induced by the support points) for the first 100 iterations and all three cameras, on only two semantic categories, in Figure 4. Note that the search volume of the parameter space is guaranteed to decrease. Figure 4(right) shows the area left in the map calculated as the sum of ellipsoids including overlaps. Although this is an overestimation, the remaining area in the map is clearly decreasing. Figure 4(left) shows that the number of nodes naturally increases in the initial iterations while remaining fairly low (about 20 nodes for Alstadt after 100 iterations).

Experiments on object detection simulation data: Another set of experiments on image-to-map registration was conducted for assessing the influence of object sizes and number of control points. Although real data are used, we refer to these experiments as simulations as we randomly simulated bounding boxes on the road or on the river (mimicking cars and boats of different size). We used the Altsatdt dataset setup. All objects belong to one of two categories: either river or tramway. Objects of various sizes were simulated on the map along with their corresponding bounding boxes on the image. The number of the support points and object sizes were varied. The computation time and the homography estimation accuracy are reported in Figure 5.

Note that the accuracy of homography estimation naturally deteriorates with larger object sizes (with a fixed number of support points). The speed of the algorithm depends on the number of support points. Interestingly, with only 20 support points the algorithm is faster because of the lower computation time for processing each node. But for 25 support points, the algorithm gets slower and for 30 support points the algorithm gets faster again. This is because (for fixed object size) more support points lead to less BnP iterations to obtain the predefined objective. Each iteration however would involve more computations than using fewer points. The remaining plots demonstrate the trade-off that BnP makes on registration parameters to reach the desired objective for different experimental setups. All the experiments were conducted for the same predefined objective.

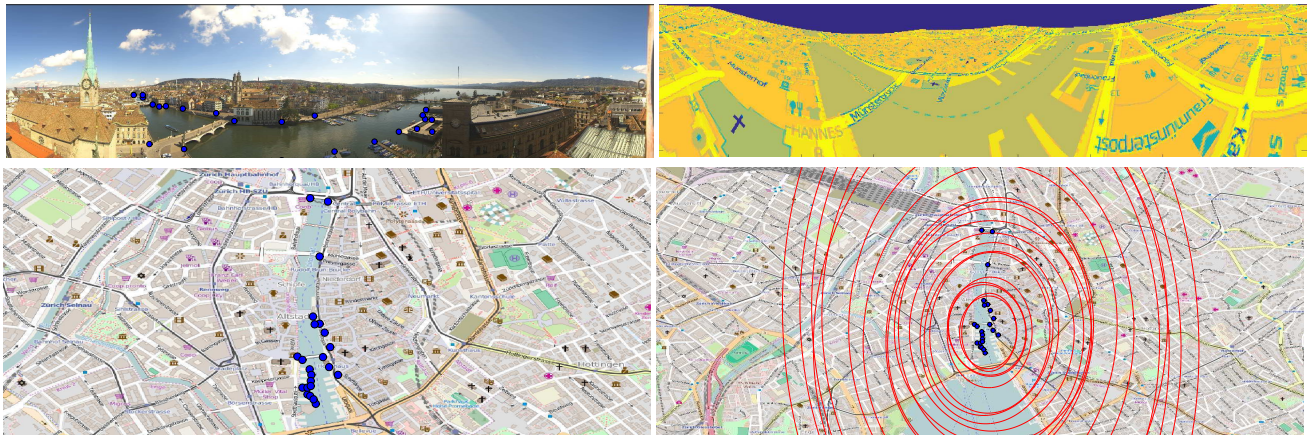

Figure 1: Selected image points (top-left) and the warped map on the image (top-right). Corresponding points on the map (bottom-left) and the ellipses around them induced by a set of homographies with 1.0 bound gap in each entry (bottom-right).

Datasets	Search area (m^2)	Semantics	Source	Support points	Time (sec)	ΔR(degrees)	$\Delta t(\%)$	$\Delta n(\%)$
Altstadt	100 × 100	river, tramway	manual	31	28.28	6.6918	10.96	5.46
		boat, tram, car	automatic	83	232.96	2.97	8.82	11.92
Sechselautenplatz	50 × 50	river, tramway	manual	17	134.56	15.48	3.95	13.33
		boat, tram, car	automatic	26	64.19	6.28	3.72	9.21
ZurichWest	50 × 50	river, tramway	manual	39	166.54	17.21	4.95	10.27
		boat, tram, car	automatic	35	144.09	10.31	8.65	13.83

Table 1: Results on three datasets for manual and automatic semantic labels. Experimental setups (search area, semantic cues, and support points) with their corresponding running time and 2D homography estimation error.

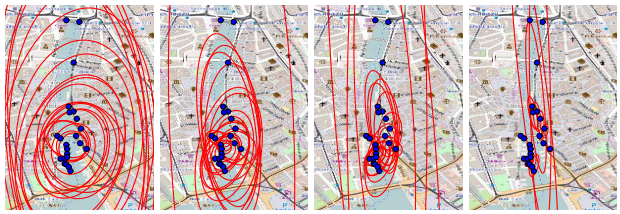

Figure 2: Left to right: ellipses induced around corresponding points by homographies of 0.5, 0.3, 0.2, 0.1 bound gaps.

4.2. Projective 3D-to-3D registration

We also tested our method to register uncalibrated 3D projective SfM reconstructions and Euclidean 3D models of the same scenes, using two real datasets: Fountain-P11 and Herz-Jesu-P8 obtained from [42]. These datasets respectively consist of 11 and 8 images of size 3072 × 2048 captured by a moving camera of $f_x = 2759.5$, $f_y = 2764.2$, $u = 1520.7$ and $v = 1006.8$, along with the laser scanned 3D scenes. We first obtained a projective reconstruction from feature point correspondences across images using [30] in Rabauds SfM Toolbox [34]. Then, the registration was carried out between the projective structure and a laser scanned 3D scene using our method. In these experiments, we used the semantic image segmentation cues automatically obtained from [47], trained on the categories of [8]. As in [31], boxes around cameras were also used: One can consider the camera centers to be within a known bounding box should GPS/IMU measurements be available. The semantic labels on the 3D scene were extracted manually although these can also be obtained automatically using methods similar to [24]. Input semantic labels and the final results obtained by our algorithm on Fountain dataset are shown in Figure 6.

Our results were compared against three other methods: SSR [31], RISAG [9], and Go-ICP [46]. For the sake of comparison, we conducted both experiments in the same setup as that of SSR. However, we could not do the so for the other methods. Therefore, these methods were conducted under their most favorable conditions. RISAG and Go-ICP require Metric and Euclidean reconstructions, respectively. The metric reconstruction required for RISAG was obtained using openMVG [28]. For Go-ICP, this reconstruction was upgraded to Euclidean using the ground-truth reconstruction scale. Table 2 summarizes the results of all four methods. Notice that our method performs significantly faster than SSR on both datasets while producing better results in terms of accuracy. Our method also produces better results than the other two methods, yet a direct comparison between them may be unfair (also because of the difference in input data for registration). RISAG and Go-ICP's reported results are rather meant to give the reader an overall impression.

Figure 3: Top left: Segmented regions (river in red and tramway in blue) selected manually; top right: automatically detected objects (boats in blue, trams in red, and cars in green) from an image sequence recorded over a day. An example of the warped map (on the image) obtained using our method before (bottom left) and after (bottom right) the refinement step.

Datasets	Methods	Semantics	Sup. points	Cameras	Bbox (m)	Time (sec)	Δf	Δuv	$\Delta R(^\circ)$	Δt	3D Error
Fountain	Ours	ground, pole, vegetation	28	11/11	2.00	47.7201	0.0602	0.1172	1.5881	0.0600	0.0167
	SSR	ground, pole, vegetation	28	11/11	2.00	238.0209	0.0984	0.4353	8.1879	0.2474	0.0169
	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	Ours	ground, pole	23	8/8	1.00	50.1480	0.0202	0.1288	2.8400	0.0843	0.0187
	SSR	ground, pole	23	8/8	1.00	348.8915	0.0421	0.2166	4.4098	0.1377	0.0190
	RISAG	-	4024	-/8	-	160.064	-	-	17.6378	0.0570	0.1830
	Go-ICP	-	4024	-/8	-	31.254	-	-	3.2618	0.169	0.0725

Table 2: Results with four different methods. Cameras: p/q cameras are bounded within boxes of size Bbox. The reported errors are RMS errors on estimating $\{f_x, f_y\}$,{u, v}, rotation, transaltion, and 3D point clouds registration.

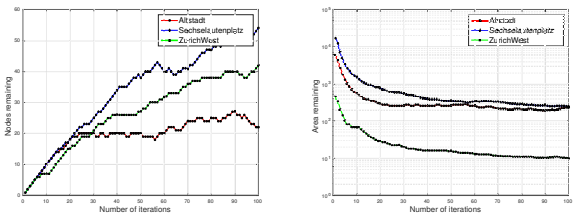

Figure 4: Number of nodes remaining to be processed (left) and area measure of ellipses (right) for three datasets with $50 \times 50m^2$ search region (shown for the first 100 iterations).

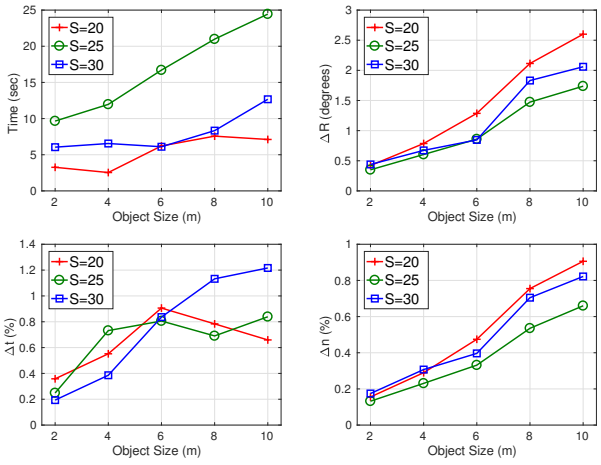

Figure 5: Quantitative results for various object sizes and support points (S). Running time in seconds (top-left) and errors in rotation (top-right), translation (bottom-left), and normal (bottom-right).

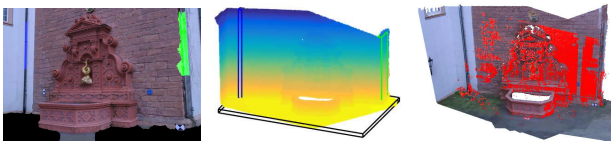

Figure 6: Left to right: image and 3D semantic cues (pole in blue, ground in black, and vegetation in green) and reconstructed point cloud (in red) registered to the scene.

5. Conclusion

We have proposed a method that demonstrates the potential of using semantic cues to estimate geometric transformations. We have applied our method for solving two challenging registration problems in which using semantic cues might as well be the most plausible solution. Our approach differs from traditional methods because we search for region-to-region correspondences rather than low-level feature correspondences. The proposed optimization formulations are purely convex and can be solved efficiently using the existing optimization algorithms. These formulations have the potential to be useful for solving many other computer vision problems.

Acknowledgements

This work was supported by the European Research Council project VarCity, under grant agreement No. 273940.

References

[1] http://www.roundshot.com. 6

[2] http://www.openstreetmap.org. 6

[3] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, New York, NY, USA, 2004. 2, 3, 4

[4] S. P. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan. *Linear matrix inequalities in system and control theory*, volume 15. SIAM, 1994. 3

[5] M. Choi, T. Lam, and B. Reznick. Sums of squares of real polynomials. *Proceedings of Symposia in Pure Mathematics*, 2(58):103–126, 1995. 3

[6] S. Christy and R. Horaud. Iterative pose computation from line correspondences. In *Comput. Vis. Image Underst (CVIU)*, pages 137–144, January 1999. 1

[7] J. Civera, D. Gálvez-López, L. Riazuelo, J. D. Tardós, and J. Montiel. Towards semantic slam using a monocular camera. In *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1277–1284. IEEE, 2011. 1

[8] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3213–3223, 2016. 7

[9] 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. 7

[10] M. Crocco, C. Rubino, and A. Del Bue. Structure from motion with objects. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016. 1

[11] D. DeTone, T. Malisiewicz, and A. Rabinovich. Deep image homography estimation. *arXiv preprint arXiv:1606.03798*, 2016. 1

[12] D. Eigen, C. Puhrsch, and R. Fergus. Depth map prediction from a single image using a multi-scale deep network. In *Advances in neural information processing systems*, pages 2366–2374, 2014. 1

[13] J. Fredriksson, V. Larsson, C. Olsson, and F. Kahl. Optimal relative pose with unknown correspondences. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1728–1736, 2016. 1

[14] E. V. Haynsworth. On the schur complement. Technical report, DTIC Document, 1968. 2

[15] D. Hilbert. Über die darstellung definiter formen als summe von formenquadraten. *Mathematische Annalen*, 32(3):342–350, 1888. 3

[16] Z. Huang, R. Wang, S. Shan, and X. Chen. Projection metric learning on grassmann manifold with application to video based face recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 140–149, 2015. 1

[17] F. John. Extremum problems with inequalities as subsidiary conditions. In *Studies and Essays Presented to R. Courant on his 60th Birthday*, pages 187–204. Interscience Publishers, Inc., New York, 1948. 3

[18] 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

[19] N. Kobyshev, H. Riemenschneider, and L. Van Gool. Matching features correctly through semantic understanding. In *2014 2nd International Conference on 3D Vision*, volume 1, pages 472–479. IEEE, 2014. 1

[20] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012. 1

[21] Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015. 1

[22] 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

[23] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3431–3440, 2015. 1

[24] A. Martinovic, J. Knopp, H. Riemenschneider, and L. Van Gool. 3d all the way: Semantic segmentation of urban scenes from start to end in 3d. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4456–4465, 2015. 1, 7

[25] M. Milford, C. Shen, S. Lowry, N. Suenderhauf, S. Shirazi, G. Lin, F. Liu, E. Pepperell, C. Lerma, B. Upcroft, et al. Sequence searching with deep-learnt depth for condition-and viewpoint-invariant route-based place recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 18–25, 2015. 1

[26] R. E. Moore and F. Bierbaum. *Methods and applications of interval analysis*, volume 2. SIAM, 1979. 6

[27] A. MOSEK. The mosek optimization toolbox for matlab manual, version 8.0. *MOSEK ApS, Denmark*, 2015. 6

[28] 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. 7

[29] K. Murthy and S. Kabadi. Some np-complete problems in quadratic and linear programming. *Mathematical Programming*, 39:117–129, 1987. 3

[30] 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

[31] D. Pani Paudel, A. Habed, C. Demonceaux, and P. Vasseur. Lmi-based 2d-3d registration: from uncalibrated images to euclidean scene. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4494–4502, 2015. 1, 2, 7

[32] D. Pani Paudel, A. Habed, C. Demonceaux, and P. Vasseur. Robust and optimal sum-of-squares-based point-to-plane

registration of image sets and structured scenes. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2048–2056, 2015. 1, 3

[33] V. Powers and T. Wörmann. An algorithm for sums of squares of real polynomials. *Journal of Pure and Applied Algebra*, 127(1):99–104, 1998. 3

[34] V. Rabaud. Vincent's Structure from Motion Toolbox. http://vision.ucsd.edu/~vrabaud/toolbox/. 7

[35] 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

[36] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 779–788, 2016. 6

[37] T. Sattler, M. Havlena, K. Schindler, and M. Pollefeys. Large-scale location recognition and the geometric burstiness problem. June 2016. 2

[38] 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

[39] N. Savinov, C. Haene, L. Ladicky, and M. Pollefeys. Semantic 3d reconstruction with continuous regularization and ray potentials using a visibility consistency constraint. June 2016. 1

[40] S. Sengupta, P. Sturgess, P. H. Torr, et al. Automatic dense visual semantic mapping from street-level imagery. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 857–862. IEEE, 2012. 1

[41] R. Socher, B. Huval, B. Bath, C. D. Manning, and A. Y. Ng. Convolutional-recursive deep learning for 3d object classification. In *Advances in Neural Information Processing Systems*, pages 665–673, 2012. 1

[42] 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. 7

[43] F. Uhlig. A recurring theorem about pairs of quadratic forms and extensions: A survey. *Linear algebra and its applications*, 25:219–237, 1979. 2

[44] T. Weyand, I. Kostrikov, and J. Philbin. Planet-photo geolocation with convolutional neural networks. *arXiv preprint arXiv:1602.05314*, 2016. 1, 2

[45] Z. Wu, S. Song, A. Khosla, F. Yu, L. Zhang, X. Tang, and J. Xiao. 3d shapenets: A deep representation for volumetric shapes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1912–1920, 2015. 1

[46] 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. 7

[47] S. Zheng, S. Jayasumana, B. Romera-Paredes, V. Vineet, Z. Su, D. Du, C. Huang, and P. H. Torr. Conditional random fields as recurrent neural networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1529–1537, 2015. 6, 7

[48] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba. Learning deep features for discriminative localization. *arXiv preprint arXiv:1512.04150*, 2015. 1