Globally-Optimal Inlier Set Maximisation for Camera Pose and Correspondence Estimation

Dylan Campbell, *Student Member, IEEE,* Lars Petersson, *Member, IEEE,*
Laurent Kneip, *Member, IEEE,* and Hongdong Li, *Member, IEEE*

Abstract—Estimating the 6-DoF pose of a camera from a single image relative to a 3D point-set is an important task for many computer vision applications. Perspective-n-point solvers are routinely used for camera pose estimation, but are contingent on the provision of good quality 2D–3D correspondences. However, finding cross-modality correspondences between 2D image points and a 3D point-set is non-trivial, particularly when only geometric information is known. Existing approaches to the simultaneous pose and correspondence problem use local optimisation, and are therefore unlikely to find the optimal solution without a good pose initialisation, or introduce restrictive assumptions. Since a large proportion of outliers and many local optima are common for this problem, we instead propose a robust and globally-optimal inlier set maximisation approach that jointly estimates the optimal camera pose and correspondences. Our approach employs branch-and-bound to search the 6D space of camera poses, guaranteeing global optimality without requiring a pose prior. The geometry of SE(3) is used to find novel upper and lower bounds on the number of inliers and local optimisation is integrated to accelerate convergence. The algorithm outperforms existing approaches on challenging synthetic and real datasets, reliably finding the global optimum, with a GPU implementation greatly reducing runtime.

Index Terms—camera pose estimation, registration, camera calibration, imaging geometry, global optimisation, branch-and-bound

1 Introduction

Estimating the pose of a calibrated camera given a set of 2D image points in the camera frame and a set of 3D points in the world frame, as shown in Fig. 1, is a fundamental part of the general 2D–3D registration problem of aligning an image with a 3D scene or model. The ability to find the pose of a camera and map visual information onto a 3D model is useful for many tasks, including localisation and tracking [1], [2], augmented reality [3], motion segmentation [4] and object recognition [5].

When correspondences are known, this becomes the Perspective-n-Point (PnP) problem for which many solutions exist [6], [7], [8], [9]. However, while hypothesise-and-test frameworks such as RANSAC [1] can mitigate the sensitivity of PnP solvers to outliers in the correspondence set, few approaches are able to handle the case where 2D–3D correspondences are not known in advance. There are many circumstances under which correspondences may be difficult to ascertain, including the general case of aligning an image with a textureless 3D point-set or CAD model. While feature extraction techniques provide a relatively robust and reproducible way to detect interest points such as edges or corners within each modality, finding correspondences across the two modalities is much more challenging. Even when the point-set has sufficient visual information associated with it, such as colour, reflectance or SIFT features [10], repetitive elements, occlusions and perspective distortion make the correspondence problem non-trivial. Moreover, appearance and thus visual features may change significantly between viewpoints, lighting conditions, weather and seasons, whereas scene geometry is often less affected. When relocalising a camera in a previously mapped environment or bootstrapping a tracking algorithm, geometry can be more reliable than appearance. Thus there is a need for methods that solve for both pose and correspondences.

Local optimisation algorithms for efficiently solving this joint problem have been proposed [11], [12]. However, they require a pose prior, search only for local optima and do not provide an optimality guarantee, yielding erroneous pose estimates without a reliable means of detecting failure. Global optimisation approaches for the correspondence-free problem, including hypothesise-and-test frameworks [1], [13], are not reliant on pose priors but quickly become computationally intractable as the number of points and outliers increase and do not provide an optimality guarantee. More recently, a global and ϵ-suboptimal method has been proposed [14], which uses a branch-and-bound approach to find a camera pose whose trimmed geometric error is within ϵ of the global minimum.

This work is the first to propose a globally-optimal inlier set cardinality maximisation solution to the simultaneous pose and correspondence problem. Named GOPAC, the algorithm is inherently robust to outliers and removes the assumptions that correspondences or training data are available, solving the most general form of the problem with geometry alone. The approach employs a branch-and-bound framework to guarantee global optimality without requiring a pose prior, ensuring that it is not susceptible

- *D. Campbell and L. Petersson are with the Australian National University, Acton, ACT 2601, Australia and Data61/CSIRO, Acton, ACT 2601, Australia. Email: {dylan.campbell,lars.petersson}@anu.edu.au*
- *L. Kneip is with ShanghaiTech University, Pudong, Shanghai 201210, China. Email: lkneip@shanghaitech.edu.cn*
- *H. Li is with the Australian National University and the Australian ARC Centre of Excellence for Robotic Vision (ACRV), Acton, ACT 2601, Australia. Email: hongdong.li@anu.edu.au*

This research is supported by an Australian Government Research Training Program (RTP) Scholarship.
Manuscript received January 15, 2018; revised June 11, 2018.

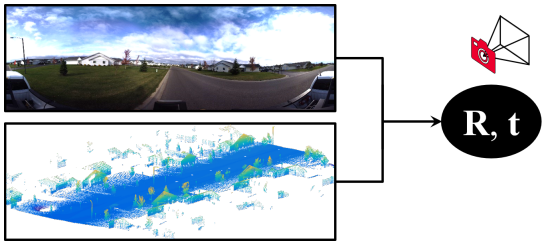
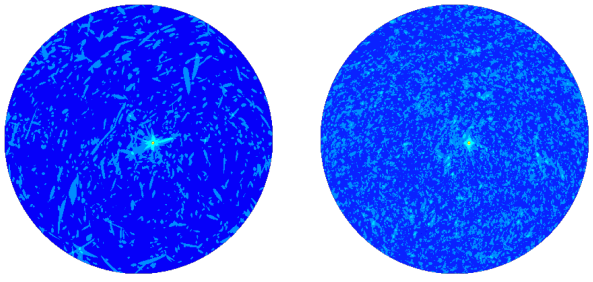


Fig. 1. Estimating the 6-DoF absolute pose of a calibrated camera from a single image, relative to a 3D point-set, with no 2D–3D correspondences. The GOPAC algorithm solves this most general form of the absolute camera pose problem, jointly estimating the position and orientation of the camera and the 2D–3D correspondences, using a globally-optimal branch-and-bound approach with tight novel bounds on the cardinality of the inlier set.



(a) 10 random points (b) 30 random points

Fig. 2. Inlier set cardinality optima for a slice of the rotation domain passing through the optimal rotation (marked in black) and the Z-axis, for two alignment problems. The colour indicates the maximum number of inliers at each rotation in the domain with lighter shades corresponding to greater cardinalities. Many local optima are evident, and are even more pervasive when solving for rotation and translation jointly, hence local optimisation in the neighbourhood of a pose prior is a bad strategy.

to local optima, such as those shown in Fig. 2. The space of rigid motions, the Special Euclidean group $SE(3)$, is parametrised in a way that facilitates branching and allows tight and novel bounds on the objective function to be derived for each branch. In addition, local optimisation methods are tightly integrated to accelerate convergence without voiding the optimality guarantee. A multi-threaded implementation on the GPU provides an additional means for greatly accelerating the algorithm. A preliminary version of this work was presented as a conference paper [15].

There are several advantages to using a cardinality maximisation approach. Firstly, it allows an exact optimiser to be found, unlike ϵ-suboptimal approaches [14]. More critically, cardinality maximisation is inherently robust to outliers without smoothing the function surface and thereby changing the location of the global optimum. In contrast, robustness conferred by trimming or robust loss functions can smooth and distort the surface of the original objective function, reducing the prominence of the global optimum and moving its location. Moreover, trimming requires the user to specify the inlier fraction, which can rarely be known in advance. If it is over- or under-estimated, the optimum may no longer occur at the correct pose. Finally, cardinality maximisation operates directly on discrete sensor data without making assumptions about the underlying structure. As a result, it can be applied in situations where the structure is not obvious from the data, such as for sparse point-sets.

2 Related Work

A large body of work exists for estimating the pose of a calibrated camera when 2D–3D correspondences are available. When the correspondences between a set of noisy image points and a 3D point-set are known perfectly, Perspective-n-Point (PnP) solvers [6], [7], [8], [9], [16] can be applied. The minimal case [6], [7] requires three 2D–3D correspondences, although greater robustness to noise can be achieved with more correspondences [8], [9], including optimality with respect to the reprojection error [16].

When outliers are present in the correspondence set, the RANSAC framework [1], [17] or robust global optimisation [18], [19], [20], [21] can be used to find the inlier set. RANSAC [1] randomly samples the correspondences, computes a pose hypothesis with a minimal PnP algorithm, and verifies the result by measuring the number of inliers, but does not provide any guarantee of optimality. Enqvist *et al*. [18] proposed a globally-optimal branch-and-bound algorithm that extended the L_∞ norm to handle outliers, but were unable to guarantee the convergence of the bounds. More recently, Svärm *et al*. [21] proposed a polynomial-time inlier maximisation approach to absolute camera pose estimation for a large-scale model, assuming that the vertical direction and height of the camera was known. Alternatively, outlier removal schemes can make quasiconvex problems more tractable [22], [23]. However, the absolute calibrated camera pose problem is not quasiconvex and many of these approaches discard inliers alongside the outliers. It should be observed that some of these approaches [1], [18] can be extended to the correspondence-free case by providing the algorithm with all possible permutations of the correspondence set. However, this leads to a hard combinatorial problem that quickly becomes infeasible.

Large-scale camera localisation, with its significant demands on outlier robustness and computational efficiency, has received a lot of attention recently [20], [21], [24], [25], [26], [27]. These methods develop sophisticated matching strategies to avoid outlier correspondences at the outset and may also incorporate RANSAC, global optimisation and outlier removal stages in their sparse feature pipeline. A recent state-of-the-art approach is Active Search [27], which prioritises those SIFT features that are more likely to yield inlier correspondences, and achieves high camera pose accuracy in feature-rich outdoor environments. However, these methods are only feasible when 2D–3D correspondences can be found. For this reason, they are often only practical for 3D models that have been constructed using stereopsis or Structure-from-Motion (SfM), associating an image feature with each 3D point and thus simplifying the correspondence problem. Generic point-sets do not have this property; a point may lie anywhere on the underlying surfaces in a laser scan, not just where strong image gradients occur.

When correspondences are unknown, the problem becomes more challenging. For the 2D–2D case, problems such as correspondence-free SfM [28], [29] and relative camera pose [30], [31] have been addressed. For the 2D–3D case, there has been a parallel investigation into geometric matching and correspondence-free absolute camera pose problems. Some approaches sidestep the full 2D–3D problem by utilising multiple cameras or a collection of images [32]

to first obtain 3D positional information from the 2D data, which is then registered against a 3D point-set.

The more general problem however is pose estimation from a single image, for which several approaches employ local optimisation. David *et al*. [11] proposed the SoftPOSIT algorithm, which alternates correspondence assignment using SoftAssign [33] with an iterative pose update algorithm. However, local methods require a pose prior and may only find a locally-optimal solution within the convergence basin of that prior. To alleviate this, these methods are frequently used within a global optimisation framework, such as random-start local search [11]. A more sophisticated approach was used in the BlindPnP algorithm [12], which represented the pose prior as a Gaussian mixture model from which a Kalman filter was initialised for matching. However, these approaches are still susceptible to local optima, require a pose prior and cannot guarantee optimality.

Grimson [13] removed the need for a pose prior by using a hypothesize-and-test approach for the simultaneous pose and correspondences problem, but the method was not optimal and quickly became intractable as the number of points increased. Pose clustering approaches [34], [35] use a similar technique, but generate all pose hypotheses before identifying dense clusters. Due to the highly combinatorial nature of searching the set of 2D–3D correspondences, these methods are limited to small input sizes. More recent approaches use machine learning techniques including regression forests and convolutional neural networks to learn 2D–3D correspondences from the data and thereby regress pose [36], [37], [38], [39]. These global optimisation methods require a large training set of images and poses and do not estimate the pose with respect to an explicit 3D model.

In contrast, globally-optimal methods find a camera pose that is guaranteed to be an optimiser of an objective function without requiring a pose prior, but tractability remains a challenge. A Branch-and-Bound (BB) [40] strategy may be applied in these cases, for which bounds need to be derived. For example, Breuel [41] used BB for 2D–2D registration problems, Hartley and Kahl [30] for optimal relative pose estimation by bounding the group of 3D rotations, Li and Hartley [42] for rotation-only 3D–3D registration, Olsson *et al*. [43] for 3D–3D registration with known correspondences, Yang *et al*. [44] for full 3D–3D registration and Campbell and Petersson [45] for robust 3D–3D registration. While not optimal, Jurie [46] used probabilistic BB for 2D–3D alignment with a linear approximation of perspective projection. More recently, Brown *et al*. [14] proposed a global and ϵ-suboptimal method using BB. It found a camera pose whose trimmed geometric error is within ϵ of the global minimum. While not susceptible to local minima, it requires the inlier fraction to be specified, which can rarely be known in advance, in order to trim outliers.

Our work is the first globally-optimal inlier set cardinality maximisation solution to the simultaneous pose and correspondence problem. It removes the assumptions that correspondences, training data or pose priors are available and is guaranteed to find the exact optimum of a robust objective function. The paper is organised as follows: we introduce the problem formulation in Section 3, develop a branch-and-bound strategy in Section 4, propose an algorithm in Section 5 and evaluate its performance in Section 6.

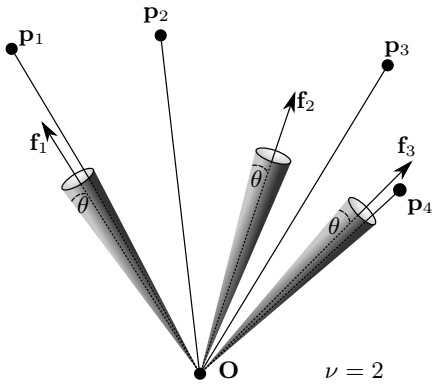


Fig. 3. The inlier set for camera pose estimation with cardinality ν, defined as the set of those bearing vectors that are less than θ from at least one 3D point with respect to the angular distance metric.

3 Inlier Set Cardinality Maximisation

The cardinality of the inlier set is a robust objective function that counts the number of inliers given a specific transformation of the data. It can operate directly on raw data representations without making assumptions about the structure of the data and is inherently robust to outliers without smoothing the objective function and thereby distorting or concealing the location of the global optimum.

For camera pose estimation, the inlier set consists of those bearing vectors that are within θ of any point in the point-set with respect to the angular distance metric, as shown in Fig. 3. Let $\mathbf{p} \in \mathbb{R}^3$ be a 3D point and $\mathbf{f} \in \mathbb{R}^3$ be a bearing vector with unit norm, corresponding to a 2D point imaged by a calibrated camera. That is, $\mathbf{f} \propto \mathbf{K}^{-1}\hat{\mathbf{x}}$ where $\mathbf{K}$ is the matrix of intrinsic camera parameters and $\hat{\mathbf{x}}$ is the homogeneous image point. Given a set of points $\mathcal{P} = \{\mathbf{p}_i\}_{i=1}^{M}$ and bearing vectors $\mathcal{F} = \{\mathbf{f}_i\}_{i=1}^{N}$ and an inlier threshold θ, the objective is to find a rotation $\mathbf{R} \in SO(3)$ and translation $\mathbf{t} \in \mathbb{R}^3$ that maximises the cardinality ν of the inlier set $\mathcal{S}_I$

$$\nu^* = \max_{\mathbf{R},\,\mathbf{t}} |\mathcal{S}_I| \tag{1}$$

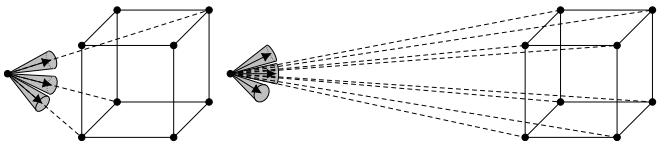
$$\mathcal{S}_I = \{\mathbf{f} \in \mathcal{F} \mid \exists \mathbf{p} \in \mathcal{P} : \angle(\mathbf{f}, \mathbf{R}(\mathbf{p} - \mathbf{t})) \leqslant \theta\} \tag{2}$$

where $\angle(\cdot,\cdot)$ denotes the angular distance between vectors. An equivalent formulation is given by

$$\nu^* = \max_{\mathbf{R},\,\mathbf{t}} f(\mathbf{R}, \mathbf{t}) \tag{3}$$

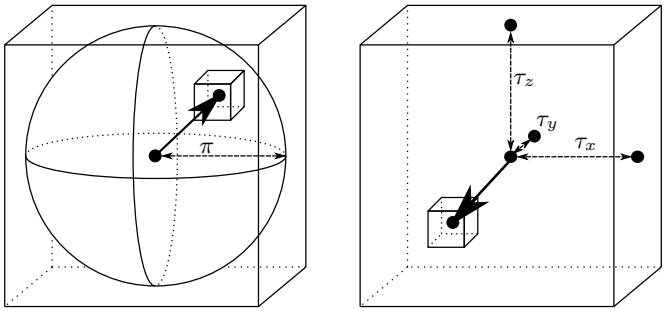
$$\nu = f(\mathbf{R}, \mathbf{t}) = \sum_{\mathbf{f} \in \mathcal{F}} \max_{\mathbf{p} \in \mathcal{P}} \mathbf{1}\big(\theta - \angle(\mathbf{f}, \mathbf{R}(\mathbf{p} - \mathbf{t}))\big) \tag{4}$$

where $\mathbf{1}(x) \triangleq \mathbf{1}_{\mathbb{R}_{\geq 0}}(x)$ is the indicator function that has the value 1 for all elements of the non-negative real numbers and the value 0 otherwise. All correspondences $(\mathbf{f}_i, \mathbf{p}_j)$ with respect to θ can be found from the optimal transformation parameters $\mathbf{R}^*$ and $\mathbf{t}^*$ by identifying all pairs for which $\angle(\mathbf{f}_i, \mathbf{R}^*(\mathbf{p}_j - \mathbf{t}^*)) \leqslant \theta$. We use the cardinality of the set of bearing vector inliers, not 3D point inliers, since this avoids the degenerate case where all 3D points become inliers when the camera is sufficiently distant, as shown in Fig. 4. Enforcing one-to-one correspondences also avoids these configurations, but introduces significant computation.



(a) Correct pose (b) Degenerate pose

Fig. 4. Maximising the cardinality of the set of bearing vector inliers instead of 3D point inliers avoids degenerate poses where all 3D points become inliers when the camera is sufficiently far from them.



(a) Rotation Domain Ω_r (b) Translation Domain Ω_t

Fig. 5. Parametrisation of $SE(3)$. (a) The rotation space $SO(3)$ is parametrised by angle-axis 3-vectors in a solid radius-π ball. (b) The translation space $\mathbb{R}^3$ is parametrised by 3-vectors bounded by a cuboid with half-widths $[\tau_x, \tau_y, \tau_z]$. The domain is branched into sub-cuboids as shown using nested octree data structures.

4 Branch-and-Bound

To solve the highly non-convex cardinality maximisation problem (1), the global optimisation technique of Branch-and-Bound (BB) [40] may be applied. To do so, a suitable means of parametrising and branching (partitioning) the function domain must be found, as well as an efficient way to calculate upper and lower bounds of the function for each branch, which converge as the branch size tends to zero. While the bounds need to be computationally efficient to calculate, the time and memory efficiency of the algorithm also depend on how tight the bounds are, since tighter bounds reduce the search space quicker by allowing suboptimal branches to be pruned. These two factors are generally in opposition and must be optimised together.

4.1 Parametrising and Branching the Domain

To find a globally-optimal solution, the cardinality of the inlier set $\mathcal{S}_I$ must be optimised over the domain of 3D motions, that is, the group $SE(3) = SO(3) \times \mathbb{R}^3$. However, the space of these transformations is unbounded. Therefore, to apply the BB paradigm, the space of translations is restricted to be within the bounded set Ω_t. For a suitably large set, it is reasonable to assume that the camera centre lies within Ω_t. That is, the camera can be assumed to be a finite distance from the 3D points. The domains are shown in Fig. 5.

Rotation space $SO(3)$ is minimally parametrised with angle-axis 3-vectors $\mathbf{r}$ with rotation angle $\|\mathbf{r}\|$ and rotation axis $\hat{\mathbf{r}} = \mathbf{r}/\|\mathbf{r}\|$. The notation $\mathbf{R}_\mathbf{r} \in SO(3)$ is used to denote the rotation matrix obtained from the matrix exponential map of the skew-symmetric matrix $[\mathbf{r}]_\times$ induced by $\mathbf{r}$. The Rodrigues' rotation formula

$$\mathbf{R}_\mathbf{r} = \exp([\mathbf{r}]_\times) = \mathbf{I} + (\sin\|\mathbf{r}\|)[\hat{\mathbf{r}}]_\times + (1 - \cos\|\mathbf{r}\|)[\hat{\mathbf{r}}]_\times^2 \quad (5)$$

can be used to efficiently calculate this mapping. Using this parametrisation, the space of all 3D rotations can be represented as a solid ball of radius π in $\mathbb{R}^3$. The mapping is one-to-one on the interior of the π-ball and two-to-one on the surface. For ease of manipulation, the 3D cube circumscribing the π-ball is used as the rotation domain Ω_r, as in [42].

Translation space $\mathbb{R}^3$ is parametrised with 3-vectors in a bounded domain chosen as the cuboid Ω_t' containing the bounding box of $\mathcal{P}$. If the camera is known to be inside the 3D scene, Ω_t' can be set to the bounding box, otherwise it is set to an expansion of the bounding box. Semantic information, such as classifying an image as 'indoors,' may also be used to restrict Ω_t'. To avoid the non-physical case where a 3D point is located within a small value ζ of the camera centre, the translation domain is restricted such that $\Omega_t = \Omega_t' \cap \{\mathbf{t} \in \mathbb{R}^3 \mid \|\mathbf{p} - \mathbf{t}\| \geqslant \zeta, \forall \mathbf{p} \in \mathcal{P}\}$.

In this implementation of BB, the domain is branched into sub-cuboids using nested octree data structures. They are defined as

$$\mathcal{C}(\mathbf{x}_0, \boldsymbol{\delta}) = \{\mathbf{x} \in \mathbb{R}^3 \mid \mathbf{e}_i^\intercal(\mathbf{x} - \mathbf{x}_0) \in [-\delta_i, \delta_i], i = 1, 2, 3\} \quad (6)$$

where $\boldsymbol{\delta}$ is the vector of half side-lengths of the cuboid and $\mathbf{e}_i$ is the i^{th} standard basis vector. To simplify the notation, we use $\mathcal{C}_r = \mathcal{C}(\mathbf{r}_0, \boldsymbol{\delta}_r)$ and $\mathcal{C}_t = \mathcal{C}(\mathbf{t}_0, \boldsymbol{\delta}_t)$ for the rotation and translation sub-cuboids respectively.

4.2 Bounding the Branches

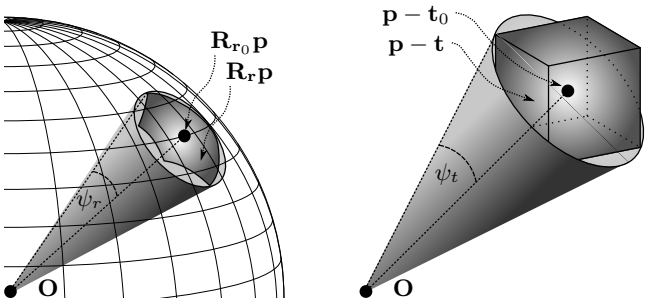
The success of a branch-and-bound algorithm is predicated on the quality of its bounds. For inlier set maximisation, the objective function (4) needs to be bounded within any sub-domain. Some preparatory material is now presented.

4.2.1 Uncertainty Angle Bounds

If a branch contained a single rotation or translation, then the new position of a point transformed by that branch would be known with certainty. However, each branch contains a contiguous set of (infinitely) many different rotations or translations. Transforming a point by one of these sets induces a transformation region, shown in Fig. 6. The transformation region lies on a sphere for rotations and in $\mathbb{R}^3$ for translations.

To bound the objective function on a branch, a bound on the maximum or worst-case angular deviation needs to be found, with respect to some arbitrary reference transformation in the branch. For simplicity, the reference transformation is the rotation or translation associated with the centroid of the cuboidal branch. In this work, the maximum deviation is termed the *uncertainty angle* because it expresses how far from the reference transformation the optimal in-branch transformation might be. The uncertainty angles induced by a rotation and translation sub-cuboid on a point $\mathbf{p}$ are shown in Fig. 6. The transformed point lies within a cone with aperture angle equal to the sum of the rotation and translation uncertainty angles.

A weak bound on the uncertainty angle due to rotation was derived in [30] using a proof, summarised in Lemma 1, that the angle between two rotated vectors is less than the Euclidean distance between their rotations' angle-axis representations in $\mathbb{R}^3$.



(a) Rotation Uncertainty Angle (b) Translation Uncertainty Angle

Fig. 6. Uncertainty angles induced by rotation and translation sub-cuboids. (a) Rotation uncertainty angle ψ_r for $\mathcal{C}_r$. The optimal rotation of $\mathbf{p}$ may be anywhere within the umbrella-shaped region on the sphere, which is entirely contained by the cone defined by $\mathbf{R}_{\mathbf{r}_0}\mathbf{p}$ and ψ_r. (b) Translation uncertainty angle ψ_t for $\mathcal{C}_t$. The optimal translation of $\mathbf{p}$ may be anywhere within the cuboidal region, which is entirely contained by the cone defined by $\mathbf{p}-\mathbf{t}_0$ and ψ_t.

Lemma 1. For an arbitrary vector $\mathbf{p}$ and two rotations, represented as $\mathbf{R}_{\mathbf{r}_1}$ and $\mathbf{R}_{\mathbf{r}_2}$ in matrix form and $\mathbf{r}_1$ and $\mathbf{r}_2$ in angle-axis form,

$$\angle(\mathbf{R}_{\mathbf{r}_1}\mathbf{p},\, \mathbf{R}_{\mathbf{r}_2}\mathbf{p}) \leqslant \|\mathbf{r}_1 - \mathbf{r}_2\|. \tag{7}$$

From this, a weak bound on the maximum angle between a vector $\mathbf{p}$ rotated by $\mathbf{r}_0$ and $\mathbf{p}$ rotated by $\mathbf{r} \in \mathcal{C}_r$ for a cube of rotation angle-axis vectors $\mathcal{C}_r$ can be found, as in [30].

Lemma 2. (Weak rotation uncertainty angle bound) Given a 3D point $\mathbf{p}$ and a rotation cube $\mathcal{C}_r$ of half side-length δ_r centred at $\mathbf{r}_0$, then $\forall \mathbf{r} \in \mathcal{C}_r$,

$$\angle(\mathbf{R}_{\mathbf{r}}\mathbf{p}, \mathbf{R}_{\mathbf{r}_0}\mathbf{p}) \leqslant \min\{\sqrt{3}\delta_r, \pi\} \triangleq \psi_r^w(\mathcal{C}_r). \tag{8}$$

Proof: Inequality (8) can be derived as follows:

$$\begin{aligned}\angle(\mathbf{R}_{\mathbf{r}}\mathbf{p}, \mathbf{R}_{\mathbf{r}_0}\mathbf{p}) &\leqslant \min\{\|\mathbf{r} - \mathbf{r}_0\|, \pi\} &(9)\\ &\leqslant \min\{\sqrt{3}\delta_r, \pi\} &(10)\end{aligned}$$

where (9) follows from Lemma 1 and the maximum possible angle between points on a sphere and (10) follows from $\max \|\mathbf{r}-\mathbf{r}_0\| = \sqrt{3}\delta_r$, the half space diagonal of the rotation cube, for $\mathbf{r} \in \mathcal{C}_r$. □

However, a tighter bound can be found by observing that a point rotated about an axis parallel to the point vector is not displaced. To exploit this, we maximise the angle $\angle(\mathbf{R}_{\mathbf{r}}\mathbf{p}, \mathbf{R}_{\mathbf{r}_0}\mathbf{p})$ over the surface $\mathcal{S}_r$ of the cube $\mathcal{C}_r$ as follows.

Lemma 3. (Rotation uncertainty angle bound) Given a 3D point $\mathbf{p}$ and a rotation cube $\mathcal{C}_r$ centred at $\mathbf{r}_0$ with surface $\mathcal{S}_r$, then $\forall \mathbf{r} \in \mathcal{C}_r$,

$$\begin{aligned}\angle(\mathbf{R}_{\mathbf{r}}\mathbf{p}, \mathbf{R}_{\mathbf{r}_0}\mathbf{p}) &\leqslant \min\left\{\max_{\mathbf{r}\in\mathcal{S}_r} \angle(\mathbf{R}_{\mathbf{r}}\mathbf{p}, \mathbf{R}_{\mathbf{r}_0}\mathbf{p}), \pi\right\} &(11)\\ &\triangleq \psi_r(\mathbf{p}, \mathcal{C}_r). &(12)\end{aligned}$$

Proof: Inequality (11) can be derived as follows:

$$\begin{aligned}\angle(\mathbf{R}_{\mathbf{r}}\mathbf{p}, \mathbf{R}_{\mathbf{r}_0}\mathbf{p}) &\leqslant \min\left\{\max_{\mathbf{r}\in\mathcal{C}_r} \angle(\mathbf{R}_{\mathbf{r}}\mathbf{p}, \mathbf{R}_{\mathbf{r}_0}\mathbf{p}), \pi\right\} &(13)\\ &= \min\left\{\max_{\mathbf{r}\in\mathcal{S}_r} \angle(\mathbf{R}_{\mathbf{r}}\mathbf{p}, \mathbf{R}_{\mathbf{r}_0}\mathbf{p}), \pi\right\} &(14)\end{aligned}$$

where (13) follows from maximising the angle over the rotation cube $\mathcal{C}_r$ and capping the angle at the maximum possible angle between points on a sphere and (14) is a consequence of the order-preserving mapping, with respect to the radial angle, from the convex cube of angle-axis vectors to the spherical surface patch (see Fig. 6a), since the mapping is obtained by projecting from the centre of the sphere to the surface of the sphere. See Section 5.4.2 for further details. □

A weak bound on the uncertainty angle due to translation was derived in [14] by enclosing the translation cuboid within a circumsphere of radius ρ_t. From this, a bound on the maximum angle between a vector $\mathbf{p}$ translated by $\mathbf{t}_0$ and $\mathbf{p}$ translated by $\mathbf{t} \in \mathcal{C}_t$ for a cube of translation vectors $\mathcal{C}_t$ can be found. For reference, the bound is reproduced here.

Lemma 4. (Weak translation uncertainty angle bound) Given a 3D point $\mathbf{p}$ and a translation cuboid $\mathcal{C}_t$ centred at $\mathbf{t}_0$ with half space diagonal ρ_t, then $\forall \mathbf{t} \in \mathcal{C}_t$,

$$\begin{aligned}\angle(\mathbf{p}-\mathbf{t}, \mathbf{p}-\mathbf{t}_0) &\leqslant \begin{cases}\arcsin\left(\dfrac{\rho_t}{\|\mathbf{p}-\mathbf{t}_0\|}\right) & \text{if } \|\mathbf{p}-\mathbf{t}_0\| \geqslant \rho_t\\ \pi & \text{else}\end{cases}\\ &\triangleq \psi_t^w(\mathbf{p}, \mathcal{C}_t). &(15)\end{aligned}$$

Proof: As given in Brown *et al*. [14]. □

However, a tighter bound can be found by using the cuboid of translated points (Fig. 6b) directly instead of its circumsphere. When the cuboid does not contain the origin, the angle can be found by maximising over the vertices.

Lemma 5. (Translation uncertainty angle bound) Given a 3D point $\mathbf{p}$ and a translation cuboid $\mathcal{C}_t$ centred at $\mathbf{t}_0$ with vertices $\mathcal{V}_t$, then $\forall \mathbf{t} \in \mathcal{C}_t$,

$$\begin{aligned}\angle(\mathbf{p}-\mathbf{t}, \mathbf{p}-\mathbf{t}_0) &\leqslant \begin{cases}\max\limits_{\mathbf{t}\in\mathcal{V}_t} \angle(\mathbf{p}-\mathbf{t}, \mathbf{p}-\mathbf{t}_0) & \text{if } \mathbf{p} \notin \mathcal{C}_t\\ \pi & \text{else}\end{cases}\\ &\triangleq \psi_t(\mathbf{p}, \mathcal{C}_t). &(16)\end{aligned}$$

Proof: Observe that for $\mathbf{p} \in \mathcal{C}_t$, the cuboid containing all translated points $\mathbf{p}-\mathbf{t}$ also contains the origin. Therefore the vectors $\mathbf{p}-\mathbf{t}$ and $\mathbf{p}-\mathbf{t}_0$ can be antiparallel (oppositely directed) and thus the maximum angle is π. For $\mathbf{p} \notin \mathcal{C}_t$,

$$\begin{aligned}\angle(\mathbf{p}-\mathbf{t}, \mathbf{p}-\mathbf{t}_0) &\leqslant \max_{\mathbf{t}\in\mathcal{C}_t} \angle(\mathbf{p}-\mathbf{t}, \mathbf{p}-\mathbf{t}_0) &(17)\\ &= \max_{\mathbf{t}\in\mathcal{V}_t} \angle(\mathbf{p}-\mathbf{t}, \mathbf{p}-\mathbf{t}_0) &(18)\end{aligned}$$

where (17) follows from maximising the angle over the translation cuboid $\mathcal{C}_t$ and (18) follows from the convexity of the angle function in this domain. The maximum of a convex function over a convex set must occur at one of its extreme points: the vertices. Geometrically, the cuboid $\mathbf{p}-\mathbf{t}$ for $\mathbf{t} \in \mathcal{C}_t$ and $\mathbf{p} \notin \mathcal{C}_t$ projects to a spherical hexagon on the unit sphere. The geodesic from an arbitrary fixed point in the hexagon to any point in the hexagon is maximised when the variable point is a vertex of the hexagon. □

4.2.2 Objective Function Bounds

The preceding lemmas are used to bound the maximum of the objective function (4) within a transformation domain $\mathcal{C}_r \times \mathcal{C}_t$. A lower bound can be found by evaluating the function at any transformation in the branch. In this case, the transformation at the centre of the rotation and translation cuboids is convenient and quick to evaluate.

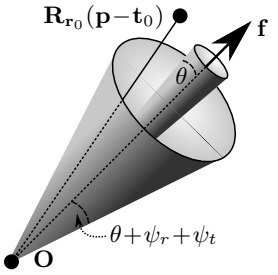


Fig. 7. Geometric intuition for the upper bound. The inlier threshold is relaxed by the two uncertainty angle bounds ψ_r and ψ_t, creating a more permissive inlier set and hence an upper bound on the cardinality.

Theorem 1. (Lower bound) For the transformation domain $\mathcal{C}_r \times \mathcal{C}_t$ centred at $(\mathbf{r}_0, \mathbf{t}_0)$, the lower bound of the inlier set cardinality can be chosen as

$$\underline{\nu} \triangleq f(\mathbf{R}_{\mathbf{r}_0}, \mathbf{t}_0). \tag{19}$$

Proof: The validity of the lower bound follows from

$$f(\mathbf{R}_{\mathbf{r}_0}, \mathbf{t}_0) \leqslant \max_{\substack{\mathbf{r}\in\mathcal{C}_r \\ \mathbf{t}\in\mathcal{C}_t}} f(\mathbf{R}_{\mathbf{r}}, \mathbf{t}). \tag{20}$$

That is, the function value at a specific point within the domain is less than or equal to the maximum. □

An upper bound on the objective function within a transformation domain $\mathcal{C}_r \times \mathcal{C}_t$ can be found using the bounds on the uncertainty angles ψ_r and ψ_t. The geometric intuition for the upper bound is that it relaxes the inlier threshold by the two uncertainty angles, creating a more permissive inlier set, as shown in Fig. 7.

Theorem 2. (Upper bound) For the transformation domain $\mathcal{C}_r \times \mathcal{C}_t$ centred at $(\mathbf{r}_0, \mathbf{t}_0)$, the upper bound of the inlier set cardinality can be chosen as

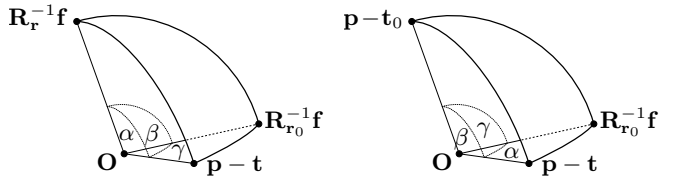
$$\bar{\nu} \triangleq \sum_{\mathbf{f}\in\mathcal{F}} \max_{\mathbf{p}\in\mathcal{P}} \mathbf{1}\Big(\theta - \angle\big(\mathbf{f}, \mathbf{R}_{\mathbf{r}_0}(\mathbf{p}-\mathbf{t}_0)\big) + \psi_r(\mathbf{f}, \mathcal{C}_r) + \psi_t(\mathbf{p}, \mathcal{C}_t)\Big). \tag{21}$$

Proof: Observe that $\forall(\mathbf{r}, \mathbf{t}) \in (\mathcal{C}_r \times \mathcal{C}_t)$,

$$\begin{aligned}
\angle\big(\mathbf{f}, \mathbf{R}_{\mathbf{r}}(\mathbf{p}-\mathbf{t})\big) &= \angle\big(\mathbf{R}_{\mathbf{r}}^{-1}\mathbf{f}, \mathbf{p}-\mathbf{t}\big) && (22)\\
&\geqslant \angle\big(\mathbf{R}_{\mathbf{r}_0}^{-1}\mathbf{f}, \mathbf{p}-\mathbf{t}\big) - \angle\big(\mathbf{R}_{\mathbf{r}}^{-1}\mathbf{f}, \mathbf{R}_{\mathbf{r}_0}^{-1}\mathbf{f}\big) && (23)\\
&\geqslant \angle\big(\mathbf{R}_{\mathbf{r}_0}^{-1}\mathbf{f}, \mathbf{p}-\mathbf{t}_0\big) - \angle\big(\mathbf{R}_{\mathbf{r}}^{-1}\mathbf{f}, \mathbf{R}_{\mathbf{r}_0}^{-1}\mathbf{f}\big) \\
&\quad - \angle\big(\mathbf{p}-\mathbf{t}, \mathbf{p}-\mathbf{t}_0\big) && (24)\\
&\geqslant \angle\big(\mathbf{f}, \mathbf{R}_{\mathbf{r}_0}(\mathbf{p}-\mathbf{t}_0)\big) - \psi_r - \psi_t && (25)
\end{aligned}$$

where (23) and (24) follow from the triangle inequality in spherical geometry (see Fig. 8) and (25) follows from Lemmas 3 and 5. Substituting (25) into (4) completes the proof. □

By inspecting the translation component of Theorem 2 and removing one of the two applications of the triangle inequality (24), a tighter upper bound can be found. A similar approach cannot be taken for the rotation component since $\mathbf{R}_{\mathbf{r}}^{-1}\mathbf{f}$ is a complex surface due to the nonlinear conversion from angle-axis to rotation matrix representations. To reduce computation, it is only necessary to evaluate this tighter bound when $\angle\big(\mathbf{f}, \mathbf{R}_{\mathbf{r}_0}(\mathbf{p}-\mathbf{t}_0)\big) \leqslant \theta + \psi_r(\mathbf{f}, \mathcal{C}_r) + \psi_t(\mathbf{p}, \mathcal{C}_t)$, since otherwise the point is definitely an outlier and does not need to be investigated further.



(a) Triangle inequality for (23) (b) Triangle inequality for (24)

Fig. 8. The triangle inequality in spherical geometry, given by $\gamma \leqslant \alpha + \beta$. The transformed points have been normalised to lie on the unit sphere.

Theorem 3. (Tighter upper bound) For the transformation domain $\mathcal{C}_r \times \mathcal{C}_t$ centred at $(\mathbf{r}_0, \mathbf{t}_0)$, the upper bound of the inlier set cardinality can be chosen as

$$\bar{\nu} \triangleq \sum_{\mathbf{f}\in\mathcal{F}} \max_{\mathbf{p}\in\mathcal{P}} \Gamma(\mathbf{f}, \mathbf{p}) \tag{26}$$

where

$$\Gamma(\mathbf{f}, \mathbf{p}) = \max_{\mathbf{t}\in\mathcal{C}_t} \mathbf{1}\big(\theta - \angle\big(\mathbf{f}, \mathbf{R}_{\mathbf{r}_0}(\mathbf{p}-\mathbf{t})\big) + \psi_r(\mathbf{f}, \mathcal{C}_r)\big). \tag{27}$$

Proof: Observe that $\forall(\mathbf{r}, \mathbf{t}) \in (\mathcal{C}_r \times \mathcal{C}_t)$,

$$\begin{aligned}
\mathbf{1}\big(\theta - \angle\big(\mathbf{f}, \mathbf{R}_{\mathbf{r}}(\mathbf{p}-\mathbf{t})\big)\big) &= \mathbf{1}\big(\theta - \angle(\mathbf{R}_{\mathbf{r}}^{-1}\mathbf{f}, \mathbf{p}-\mathbf{t})\big) && (28)\\
&\leqslant \mathbf{1}\big(\theta - \angle\big(\mathbf{f}, \mathbf{R}_{\mathbf{r}_0}(\mathbf{p}-\mathbf{t})\big) \\
&\quad + \angle(\mathbf{R}_{\mathbf{r}}^{-1}\mathbf{f}, \mathbf{R}_{\mathbf{r}_0}^{-1}\mathbf{f})\big) && (29)\\
&\leqslant \max_{\mathbf{t}\in\mathcal{C}_t} \mathbf{1}\big(\theta - \angle\big(\mathbf{f}, \mathbf{R}_{\mathbf{r}_0}(\mathbf{p}-\mathbf{t})\big) \\
&\quad + \psi_r(\mathbf{f}, \mathcal{C}_r)\big) && (30)
\end{aligned}$$

where (29) follows from the triangle inequality in spherical geometry (see Fig. 8) and (30) follows from Lemma 3 and maximising over $\mathbf{t}$. Substituting (30) into (4) completes the proof. See Section 5.4.3 for implementation details. □

4.2.3 Comparison of Uncertainty Angle Bounds

The cuboid-based uncertainty angle bounds ψ_r (11) and ψ_t (16) original to this work are smaller than the sphere-based uncertainty angle bounds ψ_r^w (8) and ψ_t^w (15) from Hartley and Kahl [30] and Brown *et al*. [14] respectively. Specifically, the maximum angular difference between ψ_t and ψ_t^w is at least 117°, shown in Fig. 9. This leads to tighter bounds on the objective function: it is clear from Theorems 1 and 2 that $\bar{\nu} - \underline{\nu}$ is smaller when ψ_r and ψ_t are smaller. The proofs that $\psi_r \leqslant \psi_r^w$ and $\psi_t \leqslant \psi_t^w$ will now be given.

Lemma 6. (Rotation uncertainty angle bounds inequality) Given a 3D point $\mathbf{p}$ and a rotation cube $\mathcal{C}_r$ centred at $\mathbf{r}_0$ with surface $\mathcal{S}_r$ and half side-length δ_r, then

$$\psi_r(\mathbf{p}, \mathcal{C}_r) \leqslant \psi_r^w(\mathcal{C}_r). \tag{31}$$

Proof: Inequality (31) can be derived as follows:

$$\begin{aligned}
\psi_r(\mathbf{p}, \mathcal{C}_r) &= \min\left\{\max_{\mathbf{r}\in\mathcal{S}_r} \angle(\mathbf{R}_{\mathbf{r}}\mathbf{p}, \mathbf{R}_{\mathbf{r}_0}\mathbf{p}), \pi\right\} && (32)\\
&= \min\left\{\angle(\mathbf{R}_{\mathbf{r}^*}\mathbf{p}, \mathbf{R}_{\mathbf{r}_0}\mathbf{p}), \pi\right\} && (33)\\
&\leqslant \min\left\{\sqrt{3}\delta_r, \pi\right\} && (34)\\
&= \psi_r^w(\mathcal{C}_r) && (35)
\end{aligned}$$

where (33) replaces the maximisation with an $\arg\max$ rotation $\mathbf{r}^*$ and (34) follows from Lemma 2. □

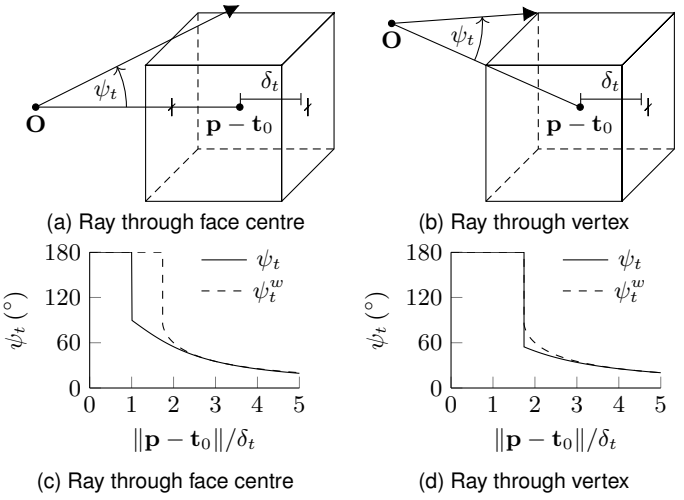


Fig. 9. Comparison of translation uncertainty angle bounds when the centre $\mathbf{p}-\mathbf{t}_0$ of the translation cuboid $\mathbf{p}-\mathbf{t}$ lies along a ray from the origin towards (a) any face centre and (b) any vertex. (c)–(d) The novel bound ψ_t is tighter across the entire domain in both cases.

Lemma 7. (Translation uncertainty angle bounds inequality) Given a 3D point $\mathbf{p}$ and a translation cuboid $\mathcal{C}_t$ centred at $\mathbf{t}_0$ with vertices $\mathcal{V}_t$ and half space diagonal ρ_t, then

$$\psi_t(\mathbf{p},\mathcal{C}_t) \leqslant \psi_t^w(\mathbf{p},\mathcal{C}_t). \tag{36}$$

Proof: Inequality (36) can be derived as follows. For $\|\mathbf{p}-\mathbf{t}_0\| \geqslant \rho_t$, which is guaranteed for $\rho_t \leqslant \zeta$,

$$\begin{aligned}\psi_t(\mathbf{p},\mathcal{C}_t) &= \max_{\mathbf{t}\in\mathcal{V}_t} \angle(\mathbf{p}-\mathbf{t},\mathbf{p}-\mathbf{t}_0) &(37)\\ &\leqslant \max_{\mathbf{t}\in S_t^2} \angle(\mathbf{p}-\mathbf{t},\mathbf{p}-\mathbf{t}_0) &(38)\\ &= \arcsin\left(\frac{\rho_t}{\|\mathbf{p}-\mathbf{t}_0\|}\right) &(39)\\ &= \psi_t^w(\mathbf{p},\mathcal{C}_t) &(40)\end{aligned}$$

where (38) follows from maximising the angle over the circumsphere S_t^2 of the cuboid instead of the vertices and (39) is shown in [14] with ρ_t being the half space diagonal of the translation sub-cuboid $\mathcal{C}_t$. For the alternate case,

$$\psi_t(\mathbf{p},\mathcal{C}_t) \leqslant \pi = \psi_t^w(\mathbf{p},\mathcal{C}_t) \tag{41}$$

when $\|\mathbf{p}-\mathbf{t}_0\| < \rho_t$. □

5 The GOPAC Algorithm

The Globally-Optimal Pose And Correspondences (GOPAC) algorithm is outlined in Algorithms 1 and 2.

5.1 Nested Branch-and-Bound Structure

A nested BB structure is employed for computational efficiency, as in [44]. In the outer breadth-first BB search, upper and lower bounds are found for each translation cuboid $\mathcal{C}_t \in \Omega_t$ by running an inner BB search over rotation space $SO(3)$ (denoted RBB). The upper bound $\bar{\nu} \triangleq \bar{\nu}_t$ (21) for the cuboid $\mathcal{C}_t$ is found by running RBB until convergence with the following bounds

$$\underline{\nu}_r \triangleq \sum_{\mathbf{f}\in\mathcal{F}} \max_{\mathbf{p}\in\mathcal{P}} \mathbf{1}\big(\theta - \angle\big(\mathbf{f}, \mathbf{R}_{\mathbf{r}_0}(\mathbf{p}-\mathbf{t}_0)\big) + \psi_t\big) \tag{42}$$

$$\bar{\nu}_r \triangleq \sum_{\mathbf{f}\in\mathcal{F}} \max_{\mathbf{p}\in\mathcal{P}} \mathbf{1}\big(\theta - \angle\big(\mathbf{f}, \mathbf{R}_{\mathbf{r}_0}(\mathbf{p}-\mathbf{t}_0)\big) + \psi_t + \psi_r\big). \tag{43}$$

Algorithm 1 GOPAC: a branch-and-bound algorithm for globally-optimal camera pose & correspondence estimation

Input: bearing vector set $\mathcal{F}$, point-set $\mathcal{P}$, inlier threshold θ, initial domains Ω_r and Ω_t
Output: optimal number of inliers ν^*, camera pose $(\mathbf{r}^*,\mathbf{t}^*)$, 2D–3D correspondences

$\nu^* \leftarrow 0$
Add translation domain Ω_t to priority queue Q_t
loop
  Update greatest upper bound $\bar{\nu}_t$ from Q_t
  Remove cuboid $\mathcal{C}_t$ with greatest width δ_{tx} from Q_t
  if $\nu^* \geqslant \bar{\nu}_t$ **then** terminate
  for all sub-cuboids $\mathcal{C}_{ti} \in \mathcal{C}_t$ **do**
    $(\underline{\nu}_{ti}, \mathbf{r}) \leftarrow \text{RBB}(\nu^*, \mathbf{t}_{0i}, \psi_t = 0)$
    if $\nu^* < 2\underline{\nu}_{ti}$ **then** $(\nu^*,\mathbf{r}^*,\mathbf{t}^*) \leftarrow \text{Refine}(\mathbf{r},\mathbf{t}_{0i})$
    $(\bar{\nu}_{ti}, \emptyset) \leftarrow \text{RBB}(\nu^*, \mathbf{t}_{0i}, \psi_t)$
    if $\nu^* < \bar{\nu}_{ti}$ **then** add $\mathcal{C}_{ti}$ to queue Q_t

Algorithm 2 RBB: a rotation search subroutine for GOPAC

Input: bearing vector set $\mathcal{F}$, point-set $\mathcal{P}$, inlier threshold θ, initial domain Ω_r, best-so-far cardinality ν^*, translation $\mathbf{t}_0$, translation uncertainty ψ_t
Output: optimal number of inliers ν_r^*, rotation $\mathbf{r}^*$

$\nu_r^* \leftarrow \nu^*$
Add rotation domain Ω_r to priority queue Q_r
loop
  Read cube $\mathcal{C}_r$ with greatest upper bound $\bar{\nu}_r$ from Q_r
  if $\nu_r^* \geqslant \bar{\nu}_r$ **then** terminate
  for all sub-cubes $\mathcal{C}_{ri} \in \mathcal{C}_r$ **do**
    Calculate $\underline{\nu}_{ri}$ by (42) or (44) with $\mathbf{r}_{0i}$, $\mathbf{t}_0$, ψ_t
    if $\nu_r^* < \underline{\nu}_{ri}$ **then** $\nu_r^* \leftarrow \underline{\nu}_{ri}, \mathbf{r}^* \leftarrow \mathbf{r}_0$
    Calculate $\bar{\nu}_{ri}$ by (43) or (45) with $\mathbf{r}_{0i}$, $\mathbf{t}_0$, ψ_t, ψ_r
    if $\nu_r^* < \bar{\nu}_{ri}$ **then** add $\mathcal{C}_{ri}$ to queue Q_r

The tighter upper bound (26) instead uses

$$\underline{\nu}_r \triangleq \sum_{\mathbf{f}\in\mathcal{F}} \max_{\mathbf{p}\in\mathcal{P}} \max_{\mathbf{t}\in\mathcal{C}_t} \mathbf{1}\big(\theta - \angle\big(\mathbf{f}, \mathbf{R}_{\mathbf{r}_0}(\mathbf{p}-\mathbf{t})\big)\big) \tag{44}$$

$$\bar{\nu}_r \triangleq \sum_{\mathbf{f}\in\mathcal{F}} \max_{\mathbf{p}\in\mathcal{P}} \max_{\mathbf{t}\in\mathcal{C}_t} \mathbf{1}\big(\theta - \angle\big(\mathbf{f}, \mathbf{R}_{\mathbf{r}_0}(\mathbf{p}-\mathbf{t})\big) + \psi_r\big). \tag{45}$$

The lower bound $\underline{\nu} \triangleq \underline{\nu}_t$ (19) is found by running RBB using bounds (42) and (43) with ψ_t set to zero.

The nested structure has better memory and computational efficiency than directly branching over 6D transformation space, since it maintains a queue for each 3D subproblem, rather than one for the entire 6D problem. This requires significantly fewer simultaneously enqueued subcubes, reducing the runtime of priority queue operations. Moreover, with rotation search nested inside translation search, ψ_t only has to be calculated once per translation $\mathbf{t}$ not once per pose $(\mathbf{r},\mathbf{t})$, and $\mathcal{F}$ can be rotated (by $\mathbf{R}^{-1}$) instead of $\mathcal{P}$ which typically has more elements. This makes it possible to precompute the rotated bearing vectors and rotation bounds for the top 5 levels of the rotation octree to reduce the amount of computation required in the inner BB subroutine. Finally, nesting does not weaken the optimality guarantee. In contrast, ϵ-suboptimality cannot be guaranteed when ϵ-suboptimal BB algorithms are nested.

5.2 Integrating Local Optimisation

Line 9 of Algorithm 1 shows how local optimisation methods are incorporated into the algorithm to refine the camera pose, in a similar manner to [14] and [44]. Whenever the BB algorithm finds a sub-cube pair $(\mathcal{C}_r, \mathcal{C}_t)$ with a greater lower bound $\underline{\nu}$ than half the best-so-far cardinality ν^*, the Perspective-n-Point (PnP) problem is solved, with correspondences given by the inlier pairs at the pose $(\mathbf{r}_0, \mathbf{t}_0)$. For this algorithm, a nonlinear optimisation solver [47] was selected, minimising the sum of angular distances between corresponding bearing vectors and points. The local optimisation method SoftPOSIT [11] is also applied at this stage to refine the camera pose without correspondences. In this way, BB and the refinement methods collaborate, with PnP finding the best pose given correspondences, SoftPOSIT finding the nearest local maxima without correspondences and BB guiding the search for correspondences and jumping out of local maxima. PnP and SoftPOSIT accelerate convergence since the faster ν^* is increased, the sooner sub-cubes (with $\bar{\nu} \leqslant \nu^*$) can be culled (Alg. 1, Line 11).

5.3 Parallel Implementations

To improve the runtime characteristics of GOPAC, CPU multithreading was implemented. This program variant divides the initial translation domain into sub-domains and runs GOPAC for each sub-domain in separate threads. It returns the greatest ν^* and the associated pose and correspondences. However, sub-optimal branches may not be pruned as quickly with this approach because the best ν value found so far is not communicated between threads.

In view of this, a massively parallel version of GOPAC was implemented on the GPU with regular communication between the threads. It directly branches over 6D transformation space with each thread computing the bounds for a single branch. We use 16384 concurrent threads and an adaptive branching strategy that chooses to subdivide the rotation or translation dimensions based on which has the greater angular uncertainty, reducing redundant branching.

Source code is publicly available on the author's website and at the DOI 10.4225/08/5a014a042bcfd.

5.4 Further Implementation Details

5.4.1 Initialising the Number of Inliers

If the best-so-far number of inliers ν^* is initialised to a value close to the optimal value, sub-optimal branches are pruned sooner, reducing the overall runtime. However, the user is unlikely to know a tight lower bound on the optimal value. Therefore, we propose (i) a P3P-RANSAC initialisation and (ii) a guess-and-verify strategy without loss of optimality. The latter provides especial benefit when 2D outliers are rare: set $\nu^* = n$; run GOPAC; stop if an optimality guarantee is found, otherwise update $n \leftarrow \max(n - s, 0)$ and repeat. We initialise n to $N - 1$ and s to $\lceil 0.1N \rceil$.

5.4.2 Rotation Uncertainty Angle Bound

Lemma 3 requires the evaluation of the angle maximiser $\max_{\mathbf{r} \in \mathcal{S}_r} \angle(\mathbf{R}_\mathbf{r}\mathbf{p}, \mathbf{R}_{\mathbf{r}_0}\mathbf{p})$, where $\mathcal{S}_r$ is the surface of the rotation cube $\mathcal{C}_r$. While it is possible to calculate the bound by sampling the cube surface using a grid of step-size σ_g, evaluating the angle at each sample and adding $\sqrt{2}/2 \times \sigma_g$ to the greatest angle calculated (by Lemma 1), it is significantly more computationally efficient to use a different approach.

The alternative approach is contingent on two assumptions: (i) the maximum always occurs on the cube skeleton (edges and vertices), not the faces; and (ii) the angle function along each edge is unimodal. Assumption (i) has been demonstrated empirically in simulations and assumption (ii) has been demonstrated for all rotation cubes used in the GOPAC algorithm (octree subdivisions of the angle-axis cube $[-\pi, \pi]^3$). In the vast majority of cases, the function is (quasi)convex and thus the edge angle maximiser occurs at one of the two vertices (extreme points). Otherwise, the maximum occurs on the edge and can be found using the efficient golden-section search routine [48], which assumes unimodality but does not require the time-consuming evaluation of the derivative. However, the sign of the derivative at the vertices needs to be evaluated to identify when the angle maximiser occurs on an edge. The derivative of the rotation angle function is obtained in Lemma 8.

Lemma 8. (Derivative of the rotation angle function) Given a unit 3D bearing vector $\mathbf{f}$ and a rotation cube $\mathcal{C}_r$ centred at $\mathbf{r}_0$ with vertices $\{\mathbf{r}_i\}_{i \in [1,8]}$, then the derivative of the rotation angle function

$$A_{ij}(\lambda) = \arccos\big((\mathbf{R}_{\mathbf{r}_0}^{-1}\mathbf{f}) \cdot (\mathbf{R}_{\mathbf{r}_{ij}(\lambda)}^{-1}\mathbf{f})\big) \tag{46}$$

with respect to λ, for an edge parametrisation of $\mathbf{r}_{ij}(\lambda) = \mathbf{r}_i + \lambda(\mathbf{r}_j - \mathbf{r}_i)$ with $\lambda \in [0, 1]$, is given by

$$\frac{\mathrm{d}A_{ij}}{\mathrm{d}\lambda} = \frac{\mathbf{f}^\intercal \mathbf{R}_{\mathbf{r}_0} \mathbf{R}_{\mathbf{r}_{ij}}^\intercal [\mathbf{f}]_\times \Big(\mathbf{r}_{ij}\mathbf{r}_{ij}^\intercal - (\mathbf{R}_{\mathbf{r}_{ij}} - I)[\mathbf{r}_{ij}]_\times\Big)\Big(\mathbf{r}_i - \mathbf{r}_j\Big)}{\|\mathbf{r}_{ij}\|^2 \sqrt{1 - (\mathbf{f}^\intercal \mathbf{R}_{\mathbf{r}_0} \mathbf{R}_{\mathbf{r}_{ij}}^\intercal \mathbf{f})^2}}. \tag{47}$$

Proof: See appendix. □

To calculate the rotation uncertainty angle bound ψ_r:

(i) for each edge $\mathbf{r}_{ij}$, evaluate the sign of the derivative of the angle function at $\lambda = 0$ and $\lambda = 1$ using (47);
(ii) if $\operatorname{sgn} \frac{\mathrm{d}A_{ij}}{\mathrm{d}\lambda}\big|_{\lambda=0} > 0$ and $\operatorname{sgn} \frac{\mathrm{d}A_{ij}}{\mathrm{d}\lambda}\big|_{\lambda=1} < 0$, use golden-section search with a tolerance of $\pi/2048$ to find the angle maximiser on that edge and add $\pi/2048$;
(iii) otherwise, the angle maximiser on that edge is one of the vertices: evaluate the angle with respect to the projected cube centre at both vertices and choose the maximum; and
(iv) choose the maximum angle over all edges as ψ_r.

Note that golden-section search terminates at a tolerance of $\pi/2048$. By Lemma 1, the bound is therefore incorrect by at most $\pi/2048 = 0.088°$, a value that is added to the upper bound to ensure optimality.

5.4.3 Tighter Upper Bound

The upper bound given in Theorem 3 requires the evaluation of $\Gamma(\mathbf{f}, \mathbf{p})$ for a given translation cuboid $\mathcal{C}_t$. Γ may be evaluated by observing that the minimum angle between a ray $\mathbf{f}$ and a cuboid $\mathbf{R}_{\mathbf{r}_0}(\mathbf{p} - \mathbf{t})$ for $\mathbf{t} \in \mathcal{C}_t$ is (a) the angle between the ray and the point on the skeleton $\mathcal{S}k_t$ of the cuboid (vertices and edges) with least angular displacement from $\mathbf{f}$ or (b) zero if the ray passes through the cuboid. Thus,

$$\Gamma(\mathbf{f}, \mathbf{p}) = \begin{cases} \max\limits_{\mathbf{t} \in \mathcal{S}k_t} \mathbf{1}\big(\theta - \angle\big(\mathbf{f}, \mathbf{R}_{\mathbf{r}_0}(\mathbf{p} - \mathbf{t})\big) + \psi_r\big) & \text{if (a)} \\ 1 & \text{if (b)} \end{cases} \tag{48}$$

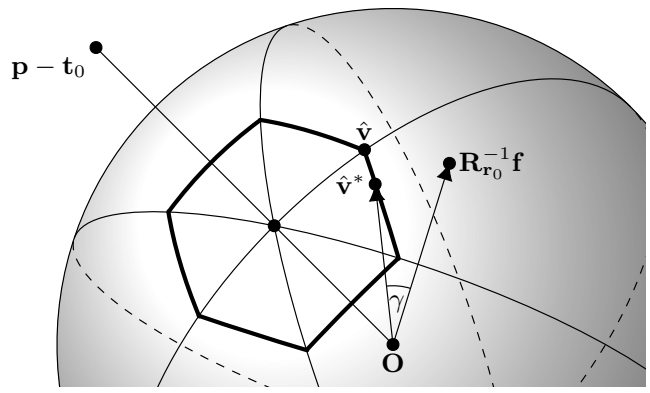


Fig. 10. Projection of the translation cuboid $\mathbf{p}-\mathbf{t}$ for $\mathbf{t} \in \mathcal{C}_t$ onto the unit sphere. The resulting spherical hexagon reduces the angle calculation to finding in which spherical lune (wedge) the rotated bearing vector $\mathbf{R}_{\mathbf{r}_0}^{-1}\mathbf{f}$ resides and then solving for the closest point $\hat{\mathbf{v}}^*$ on the geodesic of the hexagon edge in that lune. γ is the smallest angle between the rotated bearing vector and any point in the translation cuboid.

The key here is finding $\angle(\mathbf{f}, \mathbf{R}_{\mathbf{r}_0}(\mathbf{p}-\mathbf{t}))$ that maximises Γ over the skeleton. For the first case in (48), this can be done by finding $\mathbf{p}-\mathbf{t}$ with least angular displacement from $\mathbf{R}_{\mathbf{r}_0}^{-1}\mathbf{f}$. The following technique is applied:

(i) find the octant of $\mathbf{p}-\mathbf{t}_0$ with respect to the coordinate axes and project the cube to the unit sphere as a spherical hexagon;
(ii) determine in which lune induced by the spherical hexagon $\mathbf{R}_{\mathbf{r}_0}^{-1}\mathbf{f}$ resides; and
(iii) solve for the point on the hexagon edge in that lune with least angular displacement from $\mathbf{R}_{\mathbf{r}_0}^{-1}\mathbf{f}$.

By design, the cuboid of translated points $\mathbf{p}-\mathbf{t}$ for $\mathbf{t} \in \mathcal{C}_t$ lies entirely in one octant of $\mathbb{R}^3$. By finding the octant (i), the cuboid can be projected to a spherical hexagon on the unit sphere, as shown in Fig. 10. This simplifies the problem to finding the closest point $\hat{\mathbf{v}}^*$ on the spherical hexagon to the rotated bearing vector. Finding in which lune the rotated bearing vector lies (ii) further simplifies the problem to one of finding the closest point on a geodesic to the rotated bearing vector. This can be solved in closed form (iii).

5.5 Convergence Analysis

In order for the algorithm to converge, the bounds must converge as the size of the branch tends to zero. The upper bound (21) is equal to the lower bound (19) when the uncertainty angle bounds ψ_r and ψ_t are zero. Similarly, the tighter upper bound (26) is equal to the lower bound when the rotation uncertainty angle bound ψ_r is zero and the translation sub-cuboid $\mathcal{C}_t$ is of size zero, since then $\angle(\mathbf{f}, \mathbf{R}_{\mathbf{r}_0}(\mathbf{p}-\mathbf{t})) = \angle(\mathbf{f}, \mathbf{R}_{\mathbf{r}_0}(\mathbf{p}-\mathbf{t}_0))$ for $\mathbf{t} \in \mathcal{C}_t$. It remains to be seen that ψ_r and ψ_t tend to zero as the size of the sub-cuboids $\mathcal{C}_r$ and $\mathcal{C}_t$ tend to zero, irrespective of $\mathbf{f}$ or $\mathbf{p}$.

The uncertainty bound ψ_r involves a maximisation over all rotations on the surface of the sub-cube $\mathcal{C}_r$. As the sub-cube size tends to zero, in the limit the surface and centre of the cube become identified and therefore the angle $\angle(\mathbf{R}_\mathbf{r}\mathbf{f}, \mathbf{R}_{\mathbf{r}_0}\mathbf{f})$ equals zero. A similar argument applies for the uncertainty bound ψ_t, with the additional observation that a point $\mathbf{p}$ cannot lie inside a sufficiently small sub-cuboid $\mathcal{C}_t$, since the translation domain has been restricted to exclude translations for which $\|\mathbf{p}-\mathbf{t}\| < \zeta$. Therefore the bounds converge as the size of the sub-cuboids tend to zero.

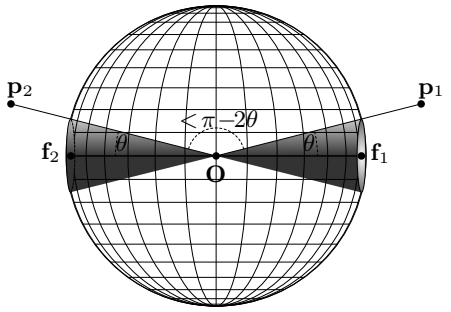


Fig. 11. A rotation-only critical configuration. The angle $\angle(\mathbf{p}_1, \mathbf{p}_2)$ is infinitesimally less than $\pi - 2\theta$. Proving that there is only 1 inlier would require infinitesimally small rotation sub-cubes.

However, an advantage of the inlier maximisation formulation is that the gap between the bounds becomes exactly zero well before the branch size becomes infinitesimal. This is not the case for critical configurations of points and bearing vectors that will only converge in the limit, as demonstrated in Fig. 11. Therefore, to guarantee that the algorithm terminates in finite time, a small tolerance value η must be subtracted from the uncertainty angles. That is, the uncertainty angles in all the formulae must be replaced with their primed versions: $\psi_r' = \psi_r - \eta$ and $\psi_t' = \psi_t - \eta$. For the tighter upper bound, η also has to be added to $\angle(\mathbf{f}, \mathbf{R}_{\mathbf{r}_0}(\mathbf{p}-\mathbf{t}))$. Except where noted, we set η to machine epsilon to ensure optimality. In C++, this can be accessed by using the command `std::numeric_limits<float>::epsilon()`.

5.6 Time Complexity Analysis

Explicitly including the tolerance η makes it possible to derive a bound on the worst-case search tree depth and thereby obtain the time complexity of the algorithm as $\mathcal{O}(MN)$. However, the notation conceals a large constant.

Theorem 4. (Time Complexity of GOPAC) Let ρ_{t_0} be the half space diagonal of the initial translation sub-cuboid $\mathcal{C}_{t_0}$, ζ be the translation restriction parameter, η be the uncertainty angle tolerance, M be the number of 3D points and N be the number of bearing vectors, then the time complexity of the GOPAC algorithm is given by

$$\mathcal{O}\left(\rho_{t_0}^3 \zeta^{-3} \eta^{-6} MN\right). \tag{49}$$

Proof: Calculating the bounds involves a summation over $\mathcal{F}$ and a maximisation over $\mathcal{P}$, therefore the complexity is $\mathcal{O}(MN)$. For the nested structure, the number of bound calculations is at worst $4N_tN_r$ where N_t and N_r are the maximum number of translation and rotation sub-cuboids examined. N_t and N_r are exponential in the worst-case tree search depths D_t and D_r, but the depths are logarithmic in η^{-1}. Therefore, the number of examined sub-cuboids is polynomial in η^{-1}. Combining these analyses gives the result (49). For a full derivation, see the appendix. □

However, experimental evaluation is more revealing for BB algorithms than time complexity analysis, since BB can prune large regions of the search space, reducing the size of the problem. This is not reflected in the complexity analysis.

6 Results

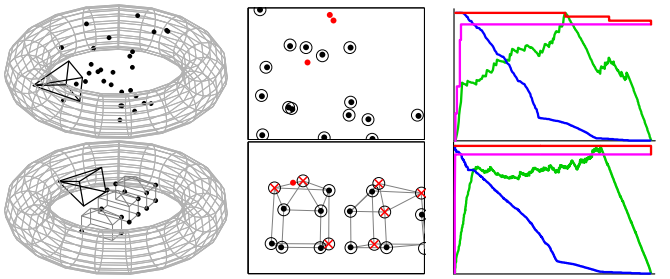
The GOPAC algorithm, denoted GP, was evaluated with respect to the baseline algorithms RANSAC [1], SoftPOSIT [11] and BlindPnP [12], denoted RS, SP and BP respectively, using both synthetic and real data. The RANSAC approach uses the OpenGV framework [47] and the P3P algorithm [7] with randomly-sampled correspondences. SoftPOSIT and BlindPnP are local optimisation algorithms and hence require a pose prior. A torus or cube prior was used in the synthetic experiments for a fair comparison. In general, the space of camera poses is much larger and a good prior can rarely be known in advance. The algorithm of Brown *et al*. [14] was not evaluated because the code and feature sets were not released publicly. However, our bounds were proved to be tighter in Section 4.2.3 and this is shown experimentally in Section 6.1. Except where otherwise specified, the inlier threshold θ was set to 1°, the lower bound from Theorem 1 and the upper bound from Theorem 2 were used, SoftPOSIT and nonlinear PnP refinement were applied and the point-to-camera limit ζ was set to 0.1. All experiments were run on a PC with a 3.4GHz quad core CPU, 8 threads were used for CPU multithreading, and up to 4 GeForce GTX 1080Ti GPUs were used for GPU multithreading.

6.1 Synthetic Data Experiments

To evaluate GOPAC in a setting where the true camera pose was known, 50 independent Monte Carlo simulations were performed per parameter setting, using the framework of Moreno Noguer *et al*. [12]: M random 3D points were generated from $[-1, 1]^3$; a fraction ω_{3D} of the 3D points were randomly selected as outliers to model occlusion; the inliers were projected to a 640×480 virtual image with an effective focal length of 800; normal noise was added to the 2D points with a standard deviation σ of 2 pixels; and random points were added to the image such that a fraction ω_{2D} of the 2D points were outliers. In addition to these random point experiments, the same procedure was applied to a repetitive CAD structure with $M = 27$ 3D points. Examples of both datasets and alignment results are shown in Fig. 12.

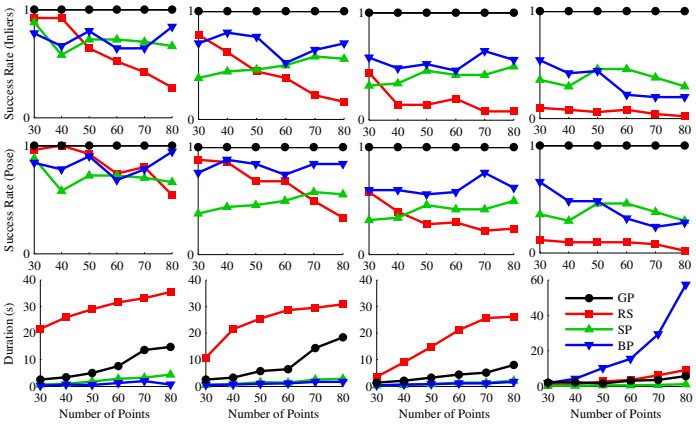
The evolution of the global lower and upper bounds over time is shown in Fig. 12c. Branch-and-Bound (BB) and the local refinement methods collaborate to increase the lower bound with BB guiding the search into convergence basins with increasingly higher local maxima and the refinement methods jumping to the nearest local maximum (the staircase pattern). It can be observed that the majority of the runtime was spent decreasing the upper bound, indicating that it will often find the global optimum when terminated early, albeit without an optimality guarantee.

To facilitate fair comparison with SoftPOSIT and BlindPnP, pose priors were used for these experiments. The torus prior constrains the camera centre to a torus around the 3D point-set with the optical axis directed towards the model, as in [12]. For BlindPnP, the poses were represented by a 20 component Gaussian Mixture Model (GMM) generated from the torus. For SoftPOSIT, the 20 mean poses from the mixture model were used to initialise the algorithm. GOPAC was given a set of translation cubes that approximated the torus and was not given any rotation prior. The cube prior constrains the camera centre to a cube



(a) 3D Results (b) 2D Results (c) Bound Evolution

Fig. 12. Sample 2D and 3D results for two experiments using the random points and CAD structure datasets. (a) 3D models, true and GOPAC-estimated camera fulcra (completely overlapping) and toroidal pose priors. (b) 2D alignment results. True projections of non-occluded 3D points are shown as black dots, 2D outliers as red dots, GOPAC projections as black circles and GOPAC-classified 3D outliers as red crosses. (c) Evolution over time of the upper (red) and lower (magenta) bounds, remaining unexplored translation volume (blue) and translation queue size (green) as a fraction of their maximum values.

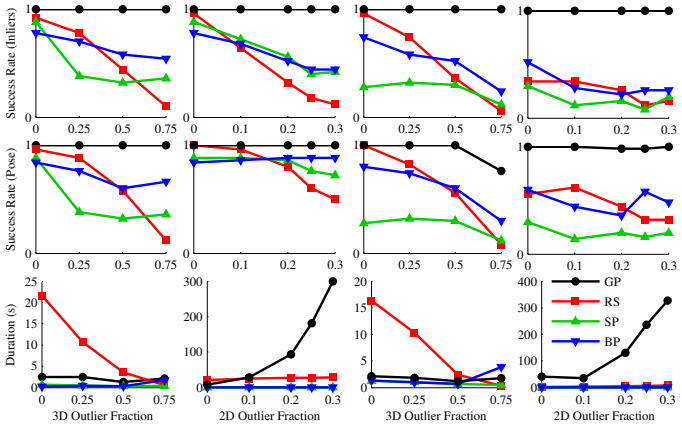


(a) $\omega_{3D} = 0$ (b) $\omega_{3D} = 0.25$ (c) $\omega_{3D} = 0.5$ (d) $\omega_{3D} = 0.75$

Fig. 13. Results for the random points dataset with the torus prior. The mean success rates and median runtimes are plotted with respect to the number of random 3D points and the 3D outlier fraction, with $\omega_{2D} = 0$ and 50 Monte Carlo simulations per parameter value.

centred randomly in $[-1, 1]^3$ with side-length 0.5 and has no restriction on rotation. This prior is more realistic since it assumes much less about the camera pose. To model the increased rotation uncertainty, 50 GMM components or pose initialisations were provided to the local methods.

The results are shown in Fig. 13, 14 and 15. Two success rates are reported: the fraction of trials where the true maximum number of inliers was found and the fraction where the correct pose was found, where the angle between the output rotation and the ground truth rotation is less than 0.1 radians and the camera centre error $\|\mathbf{t} - \mathbf{t}_{GT}\|/\|\mathbf{t}_{GT}\|$ relative to the ground truth $\mathbf{t}_{GT}$ is less than 0.1, as used in [12]. The 2D and 3D outlier fractions were fixed to 0 when not being varied and CPU multithreading was used when 2D outliers were present ($\omega_{2D} > 0$). GOPAC outperformed the other methods, reliably finding the global optimum while still being relatively efficient, particularly when the fraction of 2D outliers was low. For the repetitive CAD structure, GOPAC retrieved some incorrect poses when 75% of the 3D points were occluded, due to symmetries, while still finding the optimal number of inliers.



(a) Random Points $M = 30$ (b) CAD Structure $M = 27$

Fig. 14. Results for the random points and CAD structure datasets with the torus prior. The mean success rates and median runtimes are plotted with respect to the 3D and 2D outlier fractions, for 50 Monte Carlo simulations per parameter value.

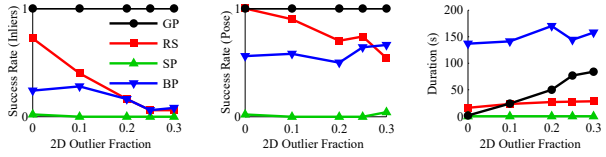


Fig. 15. Results for the random points dataset with the cube prior. The mean success rates and median runtimes are plotted with respect to the 2D outlier fraction, for 50 Monte Carlo simulations per parameter value.

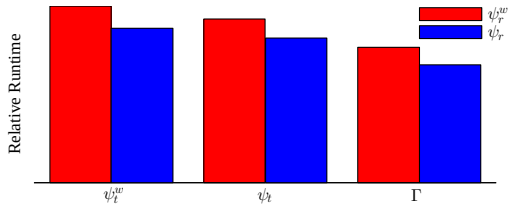


Fig. 16. Comparison of the different upper bound functions. Runtime is plotted relative to the maximum value. The weakest upper bound using ψ_r^w and ψ_t^w (leftmost) is 50% slower than the tightest upper bound using ψ_r and bounding function Γ (rightmost).

To exploit the observation that the majority of the runtime was spent decreasing the upper bound once the optimum had been attained, the random point experiments were repeated using "truncated GOPAC," halting after 30s. Despite the truncated runtime, it achieved success rates of 100%, albeit sometimes without an optimality guarantee. Finally, Fig. 16 shows the improvement attributable to the tighter upper bounds. We measured the runtime with 10 random 3D points and 50% 2D outliers using upper bounds with different combinations of the uncertainty angle bounds ψ_r^w, ψ_t^w, ψ_r and ψ_t, and the tighter bounding function Γ (26).

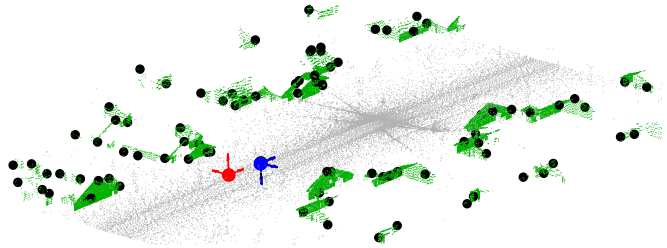
6.2 Real Data Experiments

To evaluate the algorithm on real data, the Data61/2D3D (formerly NICTA) [49] and Stanford 2D-3D-Semantics (2D-3D-S) [50] datasets were used. They are both large and repetitive multi-modal datasets with panoramic 2D images, large-scale 3D point-sets, and semantic annotations for both modalities. The former is an outdoor dataset collected from a survey vehicle with a laser scanner and 360° camera and the latter is an indoor dataset collected from a structured-light RGBD camera. Finding the pose of a camera with respect to a point-set collected by a depth sensor from a single image and without a good initialisation is an unsolved problem. The sub-problem of extracting points that correspond to known pixels in an image is itself a challenging unsolved problem for 2D–3D registration pipelines. However, since GOPAC jointly solves for pose and correspondences, this problem can be relaxed to that of isolating regions of the point-set that appear in the image and vice versa. To do this, semantic labels of the images and point-set were used to select regions that were potentially observable in both modalities: building points for the outdoor dataset and furniture points for the indoor dataset. The number of selected pixels and points were then reduced to a manageable size using grid downsampling and k-means clustering, and the pixels were converted to bearing vectors. As a result, there is a good chance that each bearing vector has a 3D point inlier, despite not knowing the correspondences in advance.

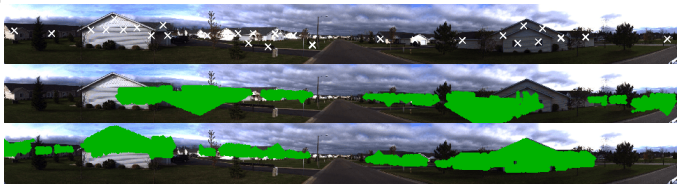
6.2.1 *Outdoor Dataset*

For the first set of experiments, a dataset was generated using this pre-processing technique for scene 1 of the Data61/2D3D dataset, consisting of a point-set with 88 points and 11 sets of 30 bearing vectors, shown in Fig. 17. The inlier threshold θ was set to $2°$, the 2D outlier fraction guess ω_{2D} was set to 0.25 and the translation domain was set to $50 \times 5 \times 5$m, covering two lanes of the road since the camera was known to be mounted on a vehicle. Results for the GOPAC and RANSAC algorithms are shown in Fig. 17 and Table 1. GOPAC found the optimal number of inliers for all frames and the correct camera pose for the majority of frames, despite the naïvety of the 2D/3D point extraction process. The failure modes for GOPAC were $180°$ rotation flips, due to ambiguities arising from the low angular separation of points in the vertical direction. In contrast, RANSAC was rarely able to find the correct pose even after 200 million iterations, due to the hard combinatorial problem of searching over all possible correspondences. In addition, SoftPOSIT and BlindPnP were unable to find the correct camera pose for any image in this dataset, even when supplied the ground truth pose as a prior, due to being sensitive to 3D points behind the camera and not natively supporting panoramic imagery, requiring an artificially restricted field of view. We also evaluated truncated GOPAC, terminating after 30s. It found the optimal number of inliers for 45% of the images and the correct pose for 64%, illustrating the difficulty of this ill-posed problem and motivating the necessity for globally-optimal guided search.

Table 1 also compares the serial and parallel (CPU and GPU) implementations of the GOPAC algorithm. The runtime of the single GPU implementation was two orders of magnitude faster than the serial implementation, as shown in Fig. 18, without any loss of optimality or accuracy. In addition, the effect of relaxing the angular tolerance η from 0 (machine epsilon) to 10^{-3} radians is reported. Some reduction in runtime is observed, without any loss of optimality. However, if the angular tolerance is too large, the algorithm may discard branches containing the optimal pose. Thus, η should be at least an order of magnitude smaller than θ.



(a) 3D point-set (grey and green), 3D features (black dots) and ground-truth (black), RANSAC (red) and GOPAC (blue) camera poses. The ground-truth and GOPAC camera poses coincide, whereas the RANSAC pose has a translation offset and a 180° rotation offset.



(b) Panoramic photograph and extracted 2D features (top), building points projected onto the image using the RANSAC camera pose (middle) and building points projected using the GOPAC camera pose (bottom).

Fig. 17. Qualitative camera pose results for scene 1 of the Data61/2D3D dataset, showing the pose of the camera when capturing image 10 and the projection of 3D building points onto image 10.

TABLE 1
Camera pose results for serial and parallel (CPU and GPU) implementations of GOPAC and RANSAC for scene 1 of the Data61/2D3D dataset. The median translation error, rotation error and runtime and the mean inlier recall and success rates are reported.

Implementation	Serial		Parallel				RAN
	CPU		CPU		GPU		SAC
Angular tolerance η	0	10^{-3}	0	10^{-3}	0	10^{-3}	–
Translation error (m)	2.30	**2.22**	2.30	2.29	**2.22**	**2.22**	28.5
Rotation error (°)	2.18	**2.08**	**2.08**	2.09	2.10	2.09	179
Recall (inliers)	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	0.81
Success rate (inliers)	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	0.00
Success rate (pose)	**0.82**	**0.82**	**0.82**	**0.82**	**0.82**	**0.82**	0.09
Runtime (s)	614	352	477	323	8	**6**	471

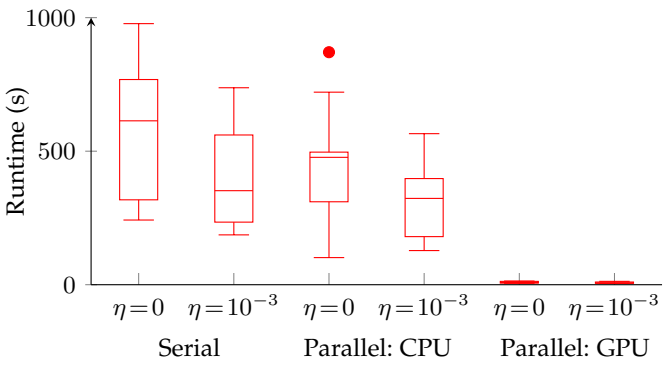


Fig. 18. Comparing the runtime of the serial and parallel (CPU and GPU) implementations of GOPAC for scene 1 of the Data61/2D3D dataset.

For the next set of experiments, the number of 2D and 3D features were increased to 50 2D and 500 3D features on average (2m^3 voxel downsampling). All 10 scenes from the Data61/2D3D dataset were processed, with 11 images per scene. The inlier threshold θ was set to $1°$, the angular tolerance η was set to 10^{-3}, and the translation domain was set to $50 \times 5 \times 5$m. Quantitative results for the quad-GPU implementation of GOPAC are given in Table 2. The single pose failure case ($< 1\%$) was caused by a symmetry in the bearing vector set. In contrast, RANSAC was able to find only 13% of the poses when run for 2 minutes per frame.

TABLE 2
Camera pose results for the quad-GPU implementation of GOPAC for the Data61/2D3D dataset. The median translation error, rotation error and runtime and the mean inlier recall and success rates are reported.

Scene	1	2	3	4	5	6	7	8	9	10
Point-set size M	514	572	721	314	259	234	245	439	819	899
Translation error (m)	1.1	1.0	1.1	1.6	1.1	1.1	0.3	1.5	0.9	0.8
Rotation error (°)	0.7	1.5	1.5	1.4	1.2	0.8	0.6	1.4	0.8	1.5
Recall (inlier)	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
Success rate (inlier)	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
Success rate (pose)	1.0	1.0	1.0	1.0	1.0	1.0	1.0	0.9	1.0	1.0
Runtime (s)	15	27	11	7	11	25	7	20	25	24

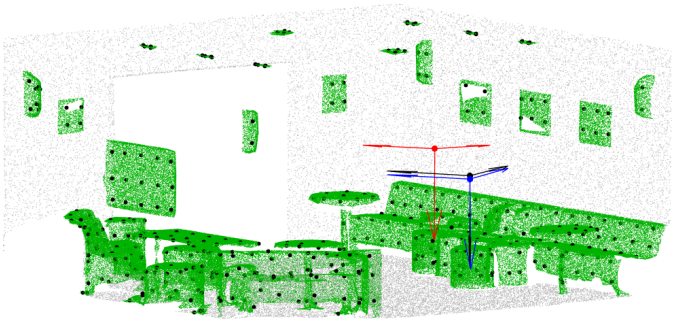
TABLE 3
Camera pose results for the quad-GPU implementation of GOPAC (GP) and RANSAC (RS) for area 3 of the Stanford 2D-3D-S dataset. The median translation error, rotation error and runtime and the mean inlier recall and success rates are reported.

Room type	lounge		office		other	
Mean point-set size	534		299		365	
Method	GP	RS	GP	RS	GP	RS
Translation error (m)	**0.07**	0.68	0.18	1.85	0.13	1.87
Rotation error (°)	**1.74**	13.0	3.40	89.7	2.95	37.5
Recall (inliers)	**1.00**	0.62	**1.00**	0.63	**1.00**	0.59
Success rate (inliers)	**1.00**	0.00	**1.00**	0.00	**1.00**	0.00
Success rate (pose)	**1.00**	0.20	0.80	0.10	**1.00**	0.14
Runtime (s)	**12**	121	40	121	35	121

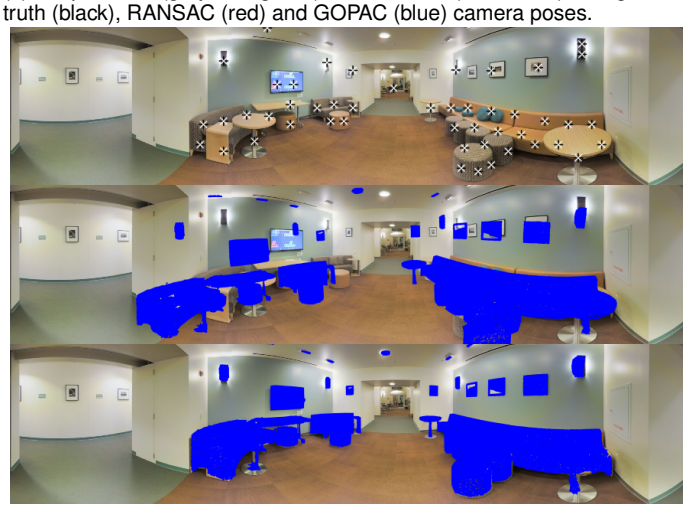
6.2.2 *Indoor Dataset*

For these experiments, a dataset was generated from area 3 of the 2D-3D-S dataset, using the same pre-processing technique as the previous section with 0.3m^3 voxel downsampling. It consists of 15 rooms (lounges, offices, WCs and a conference room) and 27 sets of 50 bearing vectors, where the camera is at least 80cm from any item of furniture. The rooms were treated as separate point-sets to model visibility constraints, which assumes that the location of the camera is known to the room level. The inlier threshold θ was set to $2.5°$, the angular tolerance η was set to $0.25°$, and the translation domain was set to the room size. Results for the quad-GPU implementation of GOPAC and RANSAC are given in Fig. 19 and Table 3.

We also tested GOPAC with regular, non-panoramic images and a more sophisticated data pre-processing strategy, extracting furniture edge points in 2D and 3D using the instance-level segmentations. The edge-features A and B datasets were generated from the 2D-3D-S dataset (lounge 1, area 3) and consist of a point-set (296 points) and image sets A and B (161 and 103 images respectively) with up to 66 2D features. B is a subset of A in which 5 or more objects were at least partially visible in each frame. Results for the quad-GPU implementation with $\theta = 1°$ and $\eta = 0.1°$ are given



(a) 3D point-set (grey and green), 3D features (black dots) and ground-truth (black), RANSAC (red) and GOPAC (blue) camera poses.



(b) Panoramic photograph and extracted 2D features (top), furniture points projected onto the image using the RANSAC camera pose (middle) and furniture points projected using the GOPAC camera pose (bottom).

Fig. 19. Qualitative camera pose results for lounge 1 of the Stanford 2D-3D-S dataset, showing the pose of the camera when capturing the image and the projection of 3D furniture points onto it.

TABLE 4
Camera pose results for the quad-GPU implementation of GOPAC (GP) and RANSAC (RS) for the edge-features A and B datasets. The median translation error, rotation error and runtime and the mean inlier recall and success rates are reported.

Dataset	A		B	
Method	GP	RS	GP	RS
Translation error (m)	0.13	6.11	**0.10**	5.77
Rotation error (°)	2.90	141	**1.88**	129
Recall (inliers)	**1.00**	0.69	**1.00**	0.56
Success rate (inliers)	**1.00**	0.00	**1.00**	0.00
Success rate (pose)	0.68	0.00	**0.99**	0.00
Runtime (s)	**44**	120	52	120

in Table 4. GOPAC was more effective on set B, since set A contains many images with few features, such as close-up images of blank walls that are not amenable to alignment.

7 Conclusion

In this paper, we have introduced a robust and globally-optimal solution to the simultaneous camera pose and correspondence problem using inlier set cardinality maximisation. The method applies the branch-and-bound paradigm to guarantee optimality regardless of initialisation and uses local optimisation to accelerate convergence. The pivotal contribution is the derivation of the function bounds using the geometry of $SE(3)$. The algorithm outperformed other local and global methods on challenging synthetic and real datasets, finding the global optimum reliably, with a GPU implementation greatly reducing runtime. Further investigation is warranted to develop a complete 2D–3D pipeline, from segmentation and clustering to alignment.

Acknowledgements

The authors would like to thank the anonymous reviewers and chairs of ICCV'17 for their careful comments and recommendations which significantly improved the paper. HL is grateful for the support provided by the ARC Centre of Excellence for Robotic Vision (CE140100016).

References

[1] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, 1981.

[2] L. Kneip, Z. Yi, and H. Li, "SDICP: Semi-dense tracking based on iterative closest points," in *Proc. British Mach. Vision Conf.* BMVA Press, Sep. 2015, pp. 100.1–100.12.

[3] E. Marchand, H. Uchiyama, and F. Spindler, "Pose estimation for augmented reality: a hands-on survey," *IEEE Trans. Vis. Comput. Graphics*, vol. 22, no. 12, pp. 2633–2651, 2016.

[4] C. F. Olson, "A general method for geometric feature matching and model extraction," *Int. J. Comput. Vision*, vol. 45, no. 1, pp. 39–54, 2001.

[5] M. Aubry, D. Maturana, A. A. Efros, B. C. Russell, and J. Sivic, "Seeing 3D chairs: exemplar part-based 2D-3D alignment using a large dataset of CAD models," in *Proc. Conf. Comput. Vision Pattern Recognition*. IEEE, 2014, pp. 3762–3769.

[6] B. M. Haralick, C.-N. Lee, K. Ottenberg, and M. Nölle, "Review and analysis of solutions of the three point perspective pose estimation problem," *Int. J. Comput. Vision*, vol. 13, no. 3, pp. 331–356, 1994.

[7] L. Kneip, D. Scaramuzza, and R. Siegwart, "A novel parametrization of the perspective-three-point problem for a direct computation of absolute camera position and orientation," in *Proc. Conf. Comput. Vision Pattern Recognition*. IEEE, 2011, pp. 2969–2976.

[8] V. Lepetit, F. Moreno-Noguer, and P. Fua, "EPnP: An accurate O(n) solution to the PnP problem," *Int. J. Comput. Vision*, vol. 81, no. 2, pp. 155–166, 2009.

[9] J. A. Hesch and S. I. Roumeliotis, "A direct least-squares (DLS) method for PnP," in *Proc. Int. Conf. Comput. Vision*. IEEE, 2011, pp. 383–390.

[10] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[11] P. David, D. Dementhon, R. Duraiswami, and H. Samet, "SoftPOSIT: simultaneous pose and correspondence determination," *Int. J. Comput. Vision*, vol. 59, no. 3, pp. 259–284, 2004.

[12] F. Moreno-Noguer, V. Lepetit, and P. Fua, "Pose priors for simultaneously solving alignment and correspondence," in *Proc. European Conf. Comput. Vision*. Springer, 2008, pp. 405–418.

[13] W. E. L. Grimson, *Object Recognition by Computer: The Role of Geometric Constraints*. Cambridge, MA, USA: MIT Press, 1990.

[14] M. Brown, D. Windridge, and J.-Y. Guillemaut, "Globally optimal 2D-3D registration from points or lines without correspondences," in *Proc. Int. Conf. Comput. Vision*, 2015, pp. 2111–2119.

[15] D. Campbell, L. Petersson, L. Kneip, and H. Li, "Globally-optimal inlier set maximisation for simultaneous camera pose and feature correspondence," in *Proc. Int. Conf. Comput. Vision*. IEEE, Oct. 2017, pp. 1–10.

[16] C. Olsson, F. Kahl, and M. Oskarsson, "Optimal estimation of perspective camera pose," in *Proc. Int. Conf. Pattern Recognition*, vol. 2. IEEE, 2006, pp. 5–8.

[17] O. Chum and J. Matas, "Optimal randomized RANSAC," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 8, pp. 1472–1482, 2008.

[18] O. Enqvist and F. Kahl, "Robust optimal pose estimation," in *Proc. European Conf. Comput. Vision*. Springer, 2008, pp. 141–153.

[19] E. Ask, O. Enqvist, and F. Kahl, "Optimal geometric fitting under the truncated L_2-norm," in *Proc. Conf. Comput. Vision Pattern Recognition*. IEEE, 2013, pp. 1722–1729.

[20] O. Enqvist, E. Ask, F. Kahl, and K. Åström, "Tractable algorithms for robust model estimation," *Int. J. Comput. Vision*, vol. 112, no. 1, pp. 115–129, 2015.

[21] L. Svärm, O. Enqvist, F. Kahl, and M. Oskarsson, "City-scale localization for cameras with known vertical direction," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 7, pp. 1455–1461, 2016.

[22] K. Sim and R. Hartley, "Removing outliers using the L_∞ norm," in *Proc. Conf. Comput. Vision Pattern Recognition*, vol. 1. IEEE, 2006, pp. 485–494.

[23] T.-J. Chin, Y. Heng Kee, A. Eriksson, and F. Neumann, "Guaranteed outlier removal with mixed integer linear programs," in *Proc. Conf. Comput. Vision Pattern Recognition*. IEEE, 2016, pp. 5858–5866.

[24] T. Sattler, B. Leibe, and L. Kobbelt, "Fast image-based localization using direct 2D-to-3D matching," in *Proc. Int. Conf. Comput. Vision*. IEEE, 2011, pp. 667–674.

[25] Y. Li, N. Snavely, D. Huttenlocher, and P. Fua, "Worldwide pose estimation using 3D point clouds," in *Proc. European Conf. Comput. Vision*. Springer-Verlag, 2012, pp. 15–29.

[26] B. Zeisl, T. Sattler, and M. Pollefeys, "Camera pose voting for large-scale image-based localization," in *Proc. Int. Conf. Comput. Vision*. IEEE, 2015, pp. 2704–2712.

[27] T. Sattler, B. Leibe, and L. Kobbelt, "Efficient effective prioritized matching for large-scale image-based localization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 9, pp. 1744–1756, Sept 2017.

[28] A. Makadia, C. Geyer, and K. Daniilidis, "Correspondence-free structure from motion," *Int. J. Comput. Vision*, vol. 75, no. 3, pp. 311–327, 2007.

[29] W.-Y. Lin, L.-F. Cheong, P. Tan, G. Dong, and S. Liu, "Simultaneous camera pose and correspondence estimation with motion coherence," *Int. J. Comput. Vision*, vol. 96, no. 2, pp. 145–161, 2012.

[30] R. I. Hartley and F. Kahl, "Global optimization through rotation space search," *Int. J. Comput. Vision*, vol. 82, no. 1, pp. 64–79, 2009.

[31] J. Fredriksson, V. Larsson, C. Olsson, and F. Kahl, "Optimal relative pose with unknown correspondences," in *Proc. Conf. Comput. Vision Pattern Recognition*. IEEE, 2016, pp. 1728–1736.

[32] D. P. 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 *Proc. Int. Conf. Comput. Vision*, 2015, pp. 2048–2056.

[33] S. Gold and A. Rangarajan, "A graduated assignment algorithm for graph matching," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 18, no. 4, pp. 377–388, 1996.

[34] T. A. Cass, "Polynomial-time geometric matching for object recognition," *Int. J. Comput. Vision*, vol. 21, no. 1, pp. 37–61, 1997.

[35] C. F. Olson, "Efficient pose clustering using a randomized algorithm," *Int. J. Comput. Vision*, vol. 23, no. 2, pp. 131–147, 1997.

[36] J. Shotton, B. Glocker, C. Zach, S. Izadi, A. Criminisi, and A. Fitzgibbon, "Scene coordinate regression forests for camera relocalization in RGB-D images," in *Proc. Conf. Comput. Vision Pattern Recognition*. IEEE, June 2013, pp. 2930–2937.

[37] A. Kendall, M. Grimes, and R. Cipolla, "Posenet: A convolutional network for real-time 6-dof camera relocalization," in *Proc. Int. Conf. Comput. Vision*, Dec 2015, pp. 2938–2946.

[38] E. Brachmann, A. Krull, S. Nowozin, J. Shotton, F. Michel, S. Gumhold, and C. Rother, "DSAC – Differentiable RANSAC for camera localization," in *Proc. Conf. Comput. Vision Pattern Recognition*, July 2017, pp. 2492–2500.

[39] A. Kendall and R. Cipolla, "Geometric loss functions for camera pose regression with deep learning," in *Proc. Conf. Comput. Vision Pattern Recognition*, July 2017, pp. 6555–6564.

[40] A. H. Land and A. G. Doig, "An automatic method of solving discrete programming problems," *Econometrica: Journal of the Econometric Society*, pp. 497–520, 1960.

[41] T. M. Breuel, "Implementation techniques for geometric branch-and-bound matching methods," *Computer Vision and Image Understanding*, vol. 90, no. 3, pp. 258–294, 2003.

[42] H. Li and R. Hartley, "The 3D-3D registration problem revisited," in *Proc. Int. Conf. Comput. Vision*. IEEE, 2007, pp. 1–8.

[43] C. Olsson, F. Kahl, and M. Oskarsson, "Branch-and-bound methods for Euclidean registration problems," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 5, pp. 783–794, 2009.

[44] J. Yang, H. Li, D. Campbell, and Y. Jia, "Go-ICP: A globally optimal solution to 3D ICP point-set registration," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 11, pp. 2241–2254, 2016.

[45] D. Campbell and L. Petersson, "GOGMA: Globally-Optimal Gaussian Mixture Alignment," in *Proc. Conf. Comput. Vision Pattern Recognition*. IEEE, June 2016, pp. 5685–5694.

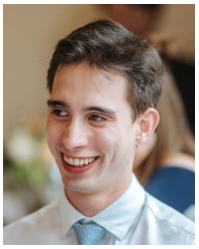
[46] F. Jurie, "Solution of the simultaneous pose and correspondence problem using Gaussian error model," *Computer Vision and Image Understanding*, vol. 73, no. 3, pp. 357–373, 1999.

[47] L. Kneip and P. Furgale, "OpenGV: A unified and generalized approach to real-time calibrated geometric vision," in *Proc. Int. Conf. Robotics Automation*. IEEE, 2014, pp. 1–8.

[48] J. Kiefer, "Sequential minimax search for a maximum," *Proc. Am. Math. Soc.*, vol. 4, no. 3, pp. 502–506, 1953.

[49] S. T. Namin, M. Najafi, M. Salzmann, and L. Petersson, "A multi-modal graphical model for scene analysis," in *Proc. Winter Conf. Applications Comput. Vision*. IEEE, 2015, pp. 1006–1013.

[50] I. Armeni, A. Sax, A. R. Zamir, and S. Savarese, "Joint 2D-3D-semantic data for indoor scene understanding," *ArXiv e-prints*, Feb. 2017. [Online]. Available: http://arxiv.org/abs/1702.01105



Dylan Campbell received a BE degree in mechatronic engineering in 2012 from the University of New South Wales in Sydney, Australia. He is currently working towards a PhD degree in computer vision at the Australian National University and Data61, CSIRO in Canberra, Australia. His research interests include visual geometry, 3D vision, registration, global optimisation, SLAM and SfM. His work won the Marr Prize (Honourable Mention) at ICCV 2017.

Lars Petersson is a Principal Research Scientist at Data61, CSIRO, Australia where he leads a team specialising in resource-constrained computer vision. Previously, he was a Principal Researcher and Research Leader at NICTA where he led the Smart Cars, AutoMap and Distributed Large Scale Vision projects. Prior to this, he did postdoctoral research at the Australian National University. He received his PhD from KTH Sweden in 2002, from where he also holds a Master's degree in Engineering Physics.

Laurent Kneip obtained a Dipl.-Ing. degree in mechatronics from Friedrich-Alexander University Erlangen-Nürnberg in 2009 and a PhD degree from ETH Zurich in 2013, in the fields of mobile robotics and computer vision. After his PhD, he was granted a Discovery Early-Career Researcher Award from the Australian Research Council and became a senior researcher at the Australian National University and a member of the ARC Center of Excellence for Robotic Vision. Since 2017, he has been an Assistant Professor at ShanghaiTech, where he founded and directs the Mobile Perception Lab. His most well-known research output, the open-source project OpenGV, summarises his contributions in geometric computer vision.

Hongdong Li is a Chief Investigator of the Australian ARC Centre of Excellence for Robotic Vision, Australian National University. His research interests include geometric computer vision, pattern recognition, computer graphics and combinatorial optimization. He is an Associate Editor for IEEE TPAMI and served as Area Chair for recent CVPR, ICCV and ECCV conferences. Jointly with coworkers, he has won a number of prestigious computer vision awards, including the CVPR Best Paper Award, the Marr Prize (Honorable Mention), the DSTO Best Fundamental Contribution to Image Processing Paper award and the best algorithm prize at the NRSFM Challenge at CVPR 2017. He is a program co-chair for ACCV 2018.