

# 2018 August Reading Reports

---

## An Efficient Solution to the Five-Point Relative Pose Problem

---

The proposed five-point algorithm is as a hypothesis generator within a random sample consensus scheme (RANSAC).

The paper itself introduces a method to efficiently calculate five-point algorithm, which requires 5 corresponding points. The five-point algorithm uses  $q'^T E q = 0$  and the constraints of  $E$ ,

$$\det(E) = 0$$

and

$$E E^T E - \frac{1}{2} \text{trace}(E E^T) E = 0$$

to calculate the essential matrix  $E$ . Then SVD is used to obtain the  $R$  and  $t$  with the cheirality constraint imposed.

---

## LeGO-LOAM

---

**Point cloud segmentation** is performed to discard points that may represent unreliable features after ground separation (not use unstable features). **Two-step L-M optimization** is used, which reduces the computation time. **Planar features** extracted from the ground are used to obtain  $[t_z; roll; pitch]$  during the first step. In the second step, the rest of the transformation  $[t_x; t_y; yaw]$  is obtained by matching **edge features extracted** from the segmented point cloud. The ability to perform **loop closures** is integrated to correct motion estimation drift.

Ground points or the segmented points are used for feature extraction and odometry.

---

## Direct Visual SLAM using Sparse Depth for Camera-LiDAR System

---

1. A sliding window-based tracking method.
2. Strict pose marginalization for accurate pose-graph SLAM.
3. Depth-integrated frame matching for large-scale mapping.

Tracking-based visual odometry (VO) module, loop-closure module and a pose-graph optimizer.

Frame management uses the ratio of projectable points and certain period of time. Then marginalization is applied for old keyframes. **Point selection** is implemented by uniformly sampling from lidar data from  $2^\circ$  by  $2^\circ$  spatial bin.

In tracking, the brightness in grayscale are defined as

$$r(\mathbf{p}_m) = \mathcal{I}_n(\pi(\mathbf{T}_m^n \mathbf{p}_m)) - \mathcal{I}_m(\pi(\mathbf{p}_m))$$

$$E_{track} = \sum_{\mathbf{p} \in \Omega} \omega(r(\mathbf{p})) r(\mathbf{p})^2$$

$\omega(\cdot)$  is a weight function based on the t-distribution. And the coarse-to-fine scheme is used.

The selected points are projected to all keyframes with covisibility.

In window based optimization, the error is computed among all keyframes as

$$E_{win} = \sum_{\mathcal{F}_i \in \mathbf{W}_{\mathcal{F}}} \sum_{\mathbf{p}_k \in \mathbf{P}_I} \sum_{\mathbf{p}_k \in \Omega} \omega(r(\mathbf{p}_k)) r(\mathbf{p}_k)^2$$

ORB features and descriptor is used to do loop closure detection. Dynamic Covariance Scaling (DCS) is applied to cope with unreliable constraints.

The lidar points are sampled into 16 and 8 to simulate the sparse data.

ArUCo markers and VRS-GPS (at fixed state) are used for evaluation.

Future work includes illumination changes and moving objects.

## DEMO

A visual odometry framework that can utilize **sparse depth** information from various sources to assist motion estimation. The proposed method is able to maximally utilize image information with and without depth coverage in motion estimation. Combining both features with and without depth in solving for motion.

## Frame to frame motion estimation

For the feature points with unknown depth,  $z_i^k$  and/or  $z_i^{k-1}$  are eliminated. The  $m$  and  $n$  features with known and unknown depth forms a objective function w.r.t.  $\theta$  and  $\mathbf{T}$  with  $2m + n$  rows. Current  $\bar{\mathbf{X}}_i^k$  and the last  $\mathbf{X}_i^{k-1}$  and  $\bar{\mathbf{X}}_i^{k-1}$ .

New points in front of the camera from depth sensors are added to the **depth map**. It is downsized according to two angular coordinates to maintain a constant point density.

The depth is found by projecting onto the **planar patch**, which is formed by three points in the angular coordinates (2D KD-tree). For the points unavailable from the depth map but tracked for longer than a certain distance in the Euclidean space. The features are **triangulated** based on a Bayesian probabilistic model.

## Bundle adjustment

It takes a sequence of images and performs a **batch optimization** to refine the frame to frame motion estimation. One image out of every five images is the bundle adjustment input. The image sequence contains a number of eight images. It uses **iSAM** open source library to all selected inlier features. The information matrix contains constant first two elements and the third one is related to the depth sensor, inversely proportional to the square of the depth, or as zero if depth is known.

Two outputs (frame-to-frame and bundle adjustment) are integrated together.

In the implementation, Harris corners are tracked by the Kanade Lucas Tomasi (KLT) method. Gap between the two ends of a trajectory compared to the length of the trajectory. DEMO shares the same frame to frame motion estimation with V-LOAM. V-LOAM replaces the bundle adjustment with the scan matching.

---

## Scale Drift-Aware Large Scale Monocular SLAM

---

The most interesting part is the backend optimization. It utilizes the 7 DoF **similarity transform** optimization.

[Source](#)

---

## Tight Coupling of Laser Scanner and Inertial Measurements for a Fully Autonomous Relative Navigation Solution

---

2D lidar is used in this work with INS. **Line** extracting and matching are applied to estimate the 2D pose. **Tilted** lidar measurements are corrected by INS. Then the **lidar range** values are used as measurements for the **Kalman filter** involving INS measurements and the states (for correct the INS error). Navigation computation includes line matching and updates.

---

## Collar Line Segments for Fast Odometry Estimation from Velodyne Point Clouds

---

The Collar Line Segments are designed for overcoming the sparsity and ring structure of Velodyne data.

1. The Velodyne point cloud is transformed into a line cloud by random generation of Collar Line Segments (CLS). It is randomly generated between lines within the bins and the shortest ones are selected.
2. Registration of the line cloud representation (iteratively). The middle points of the target lines are extracted and matched with the closest ones in the source line cloud. Corresponding points (closest points) between two lines are found to operate ICP.
3. Linear prediction is used for the transformation.
4. Mean values of the transformations among multiple scans are used for the final results.

## An Explicit Loop Closing Technique for 6D SLAM

---

An explicit loop closing heuristic (ELCH) is discussed. It **dissociates** the last scan of a sequence of acquired scans, **reassociates** it to the map, built so far by scan registration, and **distributes** the difference in the pose error over the SLAM graph. Euclidean distance between the current and all previous poses and the intermediate scans are more than a threshold.

The transformation is distributed according to the ratio of covariance sum. Then iterative method LUM can be implemented.

---

## Real-Time Loop Closure in 2D LIDAR SLAM

---

A **branch-and-bound approach** and several **precomputed grids** per finished submap.

The submap is based on 2D occupancy map (binary Bayes Filters, p95 PR). IMU with the angular velocities can help to estimate the rotational component of the pose.

If finished submaps and the current scan are **close enough** based on current pose estimates, a scan matcher tries to find the scan in the submap.

Integral numbers of steps are adopted for loop scan matching. To solve this problem a **branch and bound** scan matching is introduced. It contains node selection, branching rule and computing upper bounds.

---

## Long-range GPS-denied Aerial Inertial Navigation with LIDAR Localization

---

It uses a method with tightly coupled IMU with semi-intermittent global corrections using LIDAR measurements matched against an a priori Digital Elevation Model (DEM).

It is in the category of Terrain Referenced Navigation (TRN).

A region of sufficient number of line scans are accumulated. A smooth surface-like heightmap  $\mathbf{I}_L$  is generated, which is compared with the DEM image  $\mathbf{I}_D$  by normalized cross-correlation value. First estimate the  $x$  and  $y$  axes, then a vertical correction ( $z$ ) from the elevation images (mean of the element-wise subtraction). The offset represents the state estimation error as measured by the lidar against the ground truth DEM.

Runge-Kutta integration method RK4 is used in the IMU propagation. Error state Kalman filter is used.

The filter is rolled back to the state at which the correction occurred and recompute the propagation. Joseph form of the Kalman filter measurement is used, and the covariance matrix maintains symmetry by regularly re-symmetrizing.

---

# 1-Point-RANSAC Structure from Motion for Vehicle-Mounted Cameras by Exploiting Non-holonomic Constraints

---

A restrictive motion model is introduced, allowing to parameterize the motion with only 1 point correspondence. Only one feature correspondence for computing the epipolar geometry. Once the outliers are removed, the motion can be refined using all the inliers.

It is theoretically valid under the assumption that the camera is positioned above the rear wheel axis of the vehicle.

It assume non-holonomic constraints, which result in

$$\sin\left(\frac{\theta}{2}\right) \cdot (x'z + z'x) + \cos\left(\frac{\theta}{2}\right) \cdot (y'z - z'y) = 0$$

Once the outliers are identified, the unconstrained motion estimate (6DoF) can be computed from all the remaining inliers using standard methods. The outlier removing approaches are based on RANSAC (reprojection error) and histogram voting (for max or median  $\theta^*$ , within a given distance  $t$  or using the reprojection error).

Degraded for non-perfectly planar motion. A hyperbolic mirror (KAIDAN 360 One VR) for real data. Harris detector is used. The speed of the car from **CAN-bus** is read to measure the absolute scale.

Switch between 1-point and 5-point. When the local circular planar motion is well verified, this reflects in a narrow histogram with a very distinguishable peak.

---

## Visual-Inertial Combined Odometry System for Aerial Vehicles

---

1. A method that improves estimation of translation for a high-accuracy INS.
2. A feature reparametrization technique to partially remove odometry drift (equivalent to rotating the camera to a virtual point perpendicular to the ground).

Project to a virtual horizontal plane and form a function of 5 unknown parameters. Solve the equation with rotation  $R_P^V$  as zero, then solve non-linear function iteratively. It is adapted to a robust fitting algorithm (x, y translations). For the push-broom laser, the plane is fitted into a plane, which gives the inclination of the ground.

Analysis of error propagation is provided. The upper bound of the position drift introduced by the yaw angle noise form the INS and altimeter/push-broom laser noise is proportional to the flying distance.

---

## Scan Segments Matching for Pairwise 3D Alignment

---

Object level matching based on segmentation. Coarse alignment by matching segments across scans. Matched segments are then used to constrain point-to-point associations in regular ICP (nearest neighbor search is performed in **matched** segments only).

A pipeline combining 3D object segmentation, segment matching across scans, and alignment.

The scan segmentation separate the ground and objects in *voxel grid*. Center of mass is used for the segments. The procedure of *Segment Associations* is first bipartite assignment by *Hungarian algorithm* (Symmetric Shape Distance). Then *geometric consistency* is used to check the associations and it will filter the wrong assignments. 6 DoF point-to-point ICP is applied to the associated segments, followed by point-to-plane full-scan match.

## Evaluation Metric

The evaluation metric simply consists of voxelising the point cloud and returning the number of occupied voxels. The lower this number, the more crisp the point cloud and in turn the more accurate the alignment.

But the results are not as good as point-to-plane ICP when the displacement is small.

---

## Self-Calibration and Visual SLAM with a Multi-Camera System on a Micro Aerial Vehicle

---

For multiple calibrated stereo cameras and IMU. A 3-point minimal and linear solution for motion estimation. Build a graph of keyframes and constraints and use the double-window optimization method.

### Self-Calibration

For the self-calibration, first it obtains a globally consistent map for each stereo camera via stereo vSLAM. Bundle adjustment based on dot product. Then hand-eye calibration gives an initial estimation of the inter-stereo-camera transforms. Map is merged from all stereo cameras, followed by globally consistent pose estimates. Full bundle adjustment (reprojection error of 3D scene points and chessboard corner points) allows to recover an accurate estimation. Another hand-eye for the final IMU-involved calibration (only rotation + hand-measurement of translation).

### Visual SLAM

The motion estimation is based on the rotation from IMU, and from generalized epipolar constraint by substituting  $R$  (resulting in linear form), where only minimal 3 points are needed. After RANSAC, non-linear refinement is applied to the inlier set of correspondences.

When no new keyframe for some time, pose estimation instead of motion estimation is used. The optimization includes the additional residuals corresponding to the error between the vertical direction associated with the MAV's estimated pose and the vertical direction measurement from the IMU.

## Implementation

CenSurE feature detector is used. ORB feature descriptor is extracted.  $r_2 E r_1$  is less than a threshold, the feature match is consider as valid. Vicon is used to calculate the ground truth of the extrinsic parameters with a marked chessboard. Vicon motion capture system and loop closure error metric is used.

---

## Using Many Cameras as One

The generalized camera model is simplified as that it only includes the definition of the ray that the pixel samples. Plucker Vectors are used to parameterize arbitrary lines in space.  $q$  and  $q'$  named the direction vector and moment vector. Fisher information matrix is used to give the optimal design of panoramic imaging systems constructed from multiple cameras.

### Plucker Vectors

$q$  is a vector of any length in the direction of the line.  $q' = q \times P$ , for any point  $P$  on the line. For a normalized point in the origin camera, the Plucker vector can be written as  $\langle \tilde{x}^T, 0 \rangle$ , for other cameras, the direction vector  $q = R\tilde{x}$  the moment vector  $q \times T$ .

### Motion Estimation

The Generalized Epi-polar Constraint (GEC) can be used to estimate the motion. The pair of lines with Pluck vectors intersect iff  $q_b \cdot q'_a + q'_b \cdot q_a = 0$ . To be coincident with Heng et al.'s work, I think the equation should be

$$q_2^T R q'_1 + q_2^T [T]_{\times} q_1 + q_1'^T R q_1 = 0$$

A camera system with cameras facing in opposite directions but aligned along their optic axis is perhaps the best design for two standard cameras. The "line" ambiguity is still present. Six cameras placed to view in opposite directions along coordinate axes show no ambiguities.

---

## Double Window Optimisation for Constant Time Visual SLAM

It takes a two-level approach that combines accurate **pose-point** constraints in the primary region of interest with a stabilising periphery of **pose-pose** soft constraints. It is related to relative bundle adjustment (RBA), but RBA does not enforce metric consistency within its optimisation window. The proposed method minimise the error in both windows **simultaneously**.

An efficient approximation of covariance is defined by constant for rotation and the parallax for translation. But is not clear for the reprojection error. The method does not have a fixed keyframe, but let the damping factor of LM takes care of the gauge freedom.

Metric and appearance-based loop closure are introduced.

