2013 IEEE/RSJ International Conference on
Intelligent Robots and Systems (IROS)
November 3-7, 2013. Tokyo, Japan

Structureless Pose-Graph Loop-Closure with a Multi-Camera System on a Self-Driving Car

Gim Hee Lee[1], Friedrich Fraundorfer[2], and Marc Pollefeys[1]

[1]Computer Vision and Geometry Lab, Department of Computer Science, ETH Zürich
[2]Remote Sensing Technology, Faculty of Civil Engineering and Surveying, Technische Universität München
glee@student.ethz.ch, friedrich.fraundorfer@tum.de, marc.pollefeys@inf.ethz.ch

***Abstract*— In this paper, we propose a method to compute the pose-graph loop-closure constraints using multiple non/minimal overlapping field-of-views cameras mounted rigidly on a self-driving car without the need to reconstruct any 3D scene points. In particular, we show that the relative pose with metric scale between two loop-closing pose-graph vertices can be directly obtained from the epipolar geometry of the multi-cameras system. As a result, we avoid the additional time complexities and uncertainties from the reconstruction of 3D scene points which are needed by standard monocular and stereo approaches. In addition, there is a greater flexibility in choosing a configuration for the multi-camera system to cover a wider field-of-view so as to avoid missing out any loop-closure opportunities. We show that by expressing the point correspondences between two frames as Plücker lines and enforcing the planar motion constraint on the car, we are able to use multiple cameras as one and formulate the relative pose problem for loop-closure as a minimal problem which requires 3-point correspondences that yields up to six real solutions. The RANSAC algorithm is used to determine the correct solution and for robust estimation. We verify our method with results from multiple large-scale real-world data.**

I. Introduction

In the recent years, pose-graph Simultaneous Localization and Mapping (SLAM) has been a subject of intensive research [1]–[4] because of the enormous simplification and speed-up to the SLAM problem. Essentially, pose-graph SLAM represents the SLAM problem as an undirected graph where the vertices are the predicted robot poses and the edges are the observed constraints between any two robot poses. The simplification and speed-up of the SLAM problem are achieved by marginalizing out all the 3D scene points from the pose-graph. The errors between the predicted and observed robot poses are minimized with solvers such as the Levenberg-Marquardt algorithm. The predicted robot poses represented by the vertices are usually obtained from wheel or visual odometry readings and the constraints represented by the edges connecting consecutive vertices are the relative poses between the vertices. The constraints between non-consecutive vertices are from loop-closures and play an important role in mitigating the errors accumulated from large drifts of the wheel or visual odometry in the pose-graph.

Loop-closure opportunities are detected with the aid of the camera and vocabulary-tree [5] which returns the similarity scores between the image taken at the current robot pose and all the images took from previous robot poses. A geometric verification using, for example, the 5-point RANSAC algorithm [6] is applied to the list of loop-closure candidate with the top similarity scores. The candidate image pair with the highest inlier count and inlier count exceeding a given threshold is taken to be the loop-closure image pair. The loop-closure constraint between the two robot poses associated with the loop-closure image pair is usually computed by first retrieving the 3D scene points followed by solving the Perspective-n-Point (PnP) problem [7] for monocular cameras or the absolute orientation problem [8] for stereo cameras. It is essential to reconstruct the 3D scene points for the recovery of the metric scale between the relative pose since it is well known that the metric scale cannot recovered with pure epipolar geometry [9] of a monocular camera. The need for the 3D scene points to compute the relative pose with metric scale for the loop-closure constraint however introduces additional time complexities and uncertainties which contradicts the idea of simplification and speed-up by marginalizing out 3D scene points in pose-graph SLAM. In addition, a single monocular or stereo camera usually has limited field-of-view thus missing out on some potential loop-closing opportunities.

Fig. 1. Our car equipped with a multi-camera system with minimal overlapping field-of-views and GPS/INS system for ground truth.

In this paper, we propose a multi-camera system with non/minimal overlapping field-of-views on a car for loop-closure. We show that the relative pose with metric scale for the loop-closure constraint can be computed directly from the epipolar geometry of a multi-camera system without the need to reconstruct any 3D scene points. In particular, we adopt the generalized epipolar constraint (GEC) proposed by Pless [10]. The GEC expresses the point correspondences as

Plücker lines which turns the epipolar geometry of a multi-camera system to be in a similar structure as a single monocular camera (see Section III for more details). 17- or 16-point correspondences are needed to solve linearly for the relative pose with metric scale in [10], [11] which made it inefficient to be used within RANSAC [12] for robust estimation. We show that by enforcing the planar motion constraint of a car, we are able to formulate the relative pose with metric scale problem as a minimal problem which requires only 3-point correspondences to solve for the 3 degree-of-freedoms on a plane. The low number of point correspondences (3-point) makes it possible for robust estimation with RANSAC. We solve the system of polynomials from the minimal problem with the Hidden Variable Resultant and Companion Matrix methods [13]. A maximum of up to six real solutions can be found and the correct solution gives the highest number of inliers from RANSAC. We do a final estimate of the loop-constraint by doing a least-squares estimate using the linear algorithm from [11] with all the inliers found from RANSAC. Finally, we close the loops by doing a robust pose-graph optimization [14] with all the loop-constraints. We verify our algorithm with results from multiple large-scale real-world data.

Figure 1 shows our car equipped with four fish-eye cameras and a GPS/INS system for ground truth. It is important to note that our method is not restricted to the camera configuration shown in Figure 1 but works for any arbitrary non-degenerated [11] multi-camera configurations. We chose this camera configuration based on the maximal coverage it provides.

II. Related Works

Pless [10] first proposed the idea of the generalized camera model where a single epipolar constraint known as the generalized epipolar constraint (GEC) is used to describe the relative motion of a multi-camera system over two different frames. In this work, Pless showed that the problem of the absence of a single camera projection center can be circumvented by expressing the point correspondences as the Plücker lines and this allows any arbitrary frame to be chosen as the reference frame for the multi-camera system. In addition, the use of the Plücker lines also made it possible to formulate the GEC in the same structure as the epipolar constraint for a single camera. The so-called generalized essential matrix (GEM) is a 6×6 matrix with 18 unique entries. This means that a total of 17-point correspondences are needed to solve for the GEM linearly. Sturm showed similar derivations in [15]. Both works are however largely theoretical and showed only results from simulated data. The high number of point correspondences needed for the GEC prevented efficient use of the method within RANSAC for robust estimation with real-world data.

In [11], Li $et\ al.$ did further research on the GEC by identifying the degenerated cases for locally-central generalized camera. A locally-central generalized camera is a multi-camera system where frame-to-frame point correspondences are matched locally in each of the respective camera that made up the multi-camera system. This differs from the general case where the matching of the frame-to-frame point correspondences over different cameras is possible. Li $et\ al.$ showed that the rank of the GEC drops from 17 to 16 for the locally-central generalized camera because a null motion is always a solution to the GEC. He further showed that the null motion is always the solution found from the Singular Value Decomposition (SVD) approach [9] to solve the locally-central generalized camera GEC linearly. He proposed a new linear method which avoids the degenerated null motion solution and showed that the same approach can also be used to find the solution for the axial and locally-central-and-axial-cameras which exhibit the same degeneracy. Their approach needed 16-point correspondences which also inhibited it from being used within RANSAC for robust estimation. They showed results from a small-scale dataset collected with a Point-Grey ladybug camera in a controlled laboratory environment where point correspondences were chosen manually.

Stewénius $et\ al.$ solved the minimal problem for the GEC in [16] with the Gröbner basis [13]. The minimal problem uses 6-point correspondences to solve for the 6 degree-of-freedom in the GEC which made it possible for robust estimation within RANSAC. The approach however involved solving a system of polynomials that yields up to 64 real solutions. The high number of solutions made their approach computationally inefficient and tedious to determine the correct solution within RANSAC. In contrast, our method uses 3-point correspondences and yields up to only six real solutions. They showed results from a RANSAC implementation of the 6-point algorithm on synthetic data but not on any real-world data.

More recently, we showed in our previous work [17] that by incorporating the Ackermann motion model into the GEC, we are able to solve for the relative motion between two consecutive frames from the multi-camera system mounted on a car as a minimal problem. 2-point correspondences are needed to solve for the 2 degree-of-freedom - yaw angle and scale from the Ackermann motion model. The low number of point correspondences needed to solve the GEC made it possible for very efficient robust estimation within RANSAC. The approach was verified with results from large-scale dataset collected from a multi-camera system mounted on a car. However, the approach does not work for finding the relative pose between two loop-closure vertices in the pose-graph since the Ackermann motion constraint would be violated. In contrast, in this paper we relax the Ackermann motion constraint to a planar constraint with 3 degree-of-freedom - x, y and yaw where the additional degree-of-freedom allows the computation of the loop-closure constraints.

III. Generalized Camera Model

Our work on finding the loop-closure constraint with a multi-camera system is based on the generalized camera model. We briefly describe the concept of the generalized camera model and the GEC in this section which is needed to understand the remaining paper. More details can be found

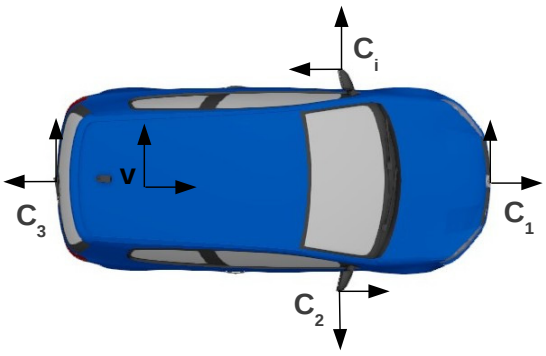


Fig. 2. Illustration of a multi-camera system mounted on a car.

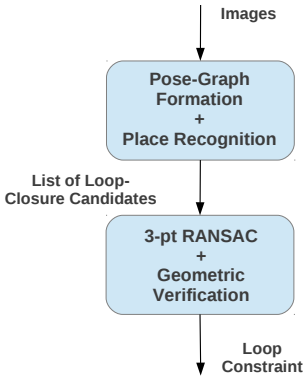


Fig. 3. System overview for our structureless loop-closure framework. Structureless loop-closure is possible because our 3-Point RANSAC computes the relative pose with metric scale directly from the image correspondences of the multi-camera system.

in [10], [15]. Figure 2 shows an illustration of a multi-camera system mounted on arbitrary locations on a car. The multi-camera system consists of individual cameras denoted by C_i and an arbitrary chosen reference frame denoted by V. We denote the intrinsics and extrinsics of the cameras with K_i and $[R_{C_i},\ t_{C_i}]$, and the normalized image coordinate of a point $\mathbf{x}_{ij}$ is given by $\hat{\mathbf{x}}_{ij} = K_i^{-1}\mathbf{x}_{ij}$. The problem of the absence of a single camera projection center for the multi-camera system is circumvent by expressing the image point as a 6-vector Plücker line given by

$$\mathbf{l}_{ij} = [\mathbf{u}_{ij}^T,\ (t_{C_i} \times \mathbf{u}_{ij})^T]^T \tag{1}$$

where $\mathbf{l}_{ij}$ describes a ray that passes through the camera center C_i and an image point $\mathbf{x}_{ij}$ seen by the camera. The unit direction of the ray expressed in the reference frame V is given $\mathbf{u}_{ij} = R_{C_i}\hat{\mathbf{x}}_{ij}$. Notice that $\mathbf{l}_{ij}$ for all cameras are now unanimously expressed in the same reference frame V and this results in the GEC given by

$$\mathbf{l}_{ij}^{\prime T} \underbrace{\begin{bmatrix} E & R \\ R & 0 \end{bmatrix}}_{E_{GC}} \mathbf{l}_{ij} = 0 \tag{2}$$

where $\mathbf{l}_{ij}^{\prime T}$ and $\mathbf{l}_{ij}^{T}$ are the point correspondences between frames V' and V expressed as Plücker lines. E_{GC} is the generalized essential matrix which consists of the relative rotation matrix R and the essential matrix E which is the same essential matrix from the epipolar geometry of a single camera. The relative translation t can be obtained from the decomposition [9] of $E = \lfloor t \rfloor_{\mathbf{x}} R$.

IV. Structureless Loop-Closure

Figure 3 shows the system overview for our structureless loop-closure framework with the multi-camera system. We form the pose-graph by computing the visual odometry [17] with every new coming image. Alternatively, the wheel odometry can also be used. The images are used to create a database with the vocabulary-tree [5] and the current image is matched against the database for a list of visually similar loop-closure candidates. We compute the point correspondences for all the loop-closure candidates and used them in our 3-point RANSAC algorithm to compute the relative poses for all the loop-closure candidates. The candidate with the highest inlier count and inlier count that exceeds a given threshold passes the geometric verification test is selected as the loop-closure constraint. A final estimate of the loop-closure constraint is done by doing a least-squares estimate using the linear algorithm from [11] with all the inliers. Finally, we do the robust pose-graph optimization based on [14] to close the loops. The robust pose-graph optimization is done to remove the detrimental effects of the outlier loop-closure constraints caused by the wrong recognitions from the vocabulary-tree in highly similar scenes. Note that the whole process is done without the need to reconstruct any 3D scene point and this is possible because our 3-Point RANSAC is derived from the epipolar geometry of the multi-camera system which directly allows the computation of the relative pose with metric scale based on only image correspondences.

A. Pose-Graph Formation and Place Recognition

The edge constraints that link consecutive vertices in the pose-graph are obtained from relative poses estimated with generalized visual odometry described in our previous work [17]. Alternatively, the wheel odometry readings can also be used. These relative poses are concatenated together to get the global poses which are represented by the vertices in the pose-graph. Loop-closure opportunities for edges that link non-consecutive vertices in the pose-graph are obtained from a vocabulary-tree [5] based place recognizer. The vocabulary-tree based place recognizer consists of the training and query phases. In the training phase, a vocabulary tree is trained offline with SURF features [18] extracted from a set of given training images. In the query phase, the acquired images are assigned unique IDs and are inserted into the vocabulary-tree database in the form of an inverted file for efficient retrieval. The database is queried with the SURF features extracted from every incoming image and the output of the query is a list of image IDs ranked according to their similarity scores with the query image. The list of database images with the top similarity scores is selected as the list of loop-closure candidates. It is important to note that we maintain only one vocabulary-tree for all the cameras from our multi-camera system. We do so by assigning unique image IDs given by

imageID = frameID $\times$ n + cameraID, where n is the total number of cameras in the multi-camera system.

B. 3-point Minimal Solution

Our multi-camera system is mounted on a car which can be assumed to be moving on a plane or at least locally planar between two loop-closing poses. Hence, we are able to write the relative transformation $[R,\ t]$ between two loop-closing frames V' and V as

$$R = \frac{1}{1+q^2}\begin{bmatrix} 1-q^2 & -2q & 0 \\ 2q & 1-q^2 & 0 \\ 0 & 0 & 1+q^2 \end{bmatrix}, \quad t = \begin{bmatrix} x \\ y \\ 0 \end{bmatrix} \tag{3}$$

where $q = tan(\frac{\theta}{2})$, hence $cos(\theta) = \frac{1-q^2}{1+q^2}$ and $sin(\theta) = \frac{2q}{1+q^2}$ according to the double-angle trigonometry identities. θ is the yaw angle. We do this trigonometric identity substitution to get rid of the difficulties in dealing with sines and cosines in the system of equations. Putting the relative transformation $[R,\ t]$ from Equation 3 into the generalized essential matrix E_{GC} from Equation 2, we get

$$E_{GC} = \begin{bmatrix} 0 & 0 & y & \frac{1-q^2}{1+q^2} & \frac{-2q}{1+q^2} & 0 \\ 0 & 0 & -x & \frac{2q}{1+q^2} & \frac{1-q^2}{1+q^2} & 0 \\ \frac{2xq-y(1-q^2)}{1+q^2} & \frac{2yq+x(1-q^2)}{1+q^2} & 0 & 0 & 0 & 1 \\ \frac{1-q^2}{1+q^2} & \frac{-2q}{1+q^2} & 0 & 0 & 0 & 0 \\ \frac{2q}{1+q^2} & \frac{1-q^2}{1+q^2} & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} \tag{4}$$

which is the generalized essential matrix with the planar constraint. Dropping the camera and image point indices i and j for brevity, we get the GEC with planar constraint from Equations 2 and 3 which is given by

$$\begin{aligned} a_1xq^2 + a_2xq + a_3x + a_4yq^2 + \\ a_5yq + a_6y + a_7q^2 + a_8q + a_9 = 0 \end{aligned} \tag{5}$$

where

$$\begin{aligned}
a_1 &= -u_yu'_w - u_wu'_y, \quad a_2 = 2u_wu'_x \\
a_3 &= u_wu'_y - u_yu'_w, \quad a_4 = u_xu'_w + u_wu'_x \\
a_5 &= 2u_wu'_y, \quad a_6 = u_xu'_w - u_wu'_x \\
a_7 &= t_{cx}(u_yu'_w + u_wu'_y) - t_{cy}(u_xu'_w + u_wu'_x) - \\
&\quad t_{cz}(u_xu'_y - u_yu'_x) + t'_{cx}(u_yu'_w + u_wu'_y) - \\
&\quad t'_{cy}(u_xu'_w + u_wu'_x) + t'_{cz}(u_xu'_y - u_yu'_x) \\
a_8 &= 2(t_{cz}u_xu'_x - t_{cx}u_wu'_x - t_{cy}u_wu'_y + t_{cz}u_yu'_y + \\
&\quad t'_{cx}u_xu'_w - t'_{cz}u_xu'_x + t'_{cy}u_yu'_w - t'_{cz}u_yu'_y) \\
a_9 &= t_{cx}(u_yu'_w - u_wu'_y) - t_{cy}(u_xu'_w - u_wu'_x) + \\
&\quad t_{cz}(u_xu'_y - u_yu'_x) - t'_{cx}(u_yu'_w - u_wu'_y) + \\
&\quad t'_{cy}(u_xu'_w - u_wu'_x) - t'_{cz}(u_xu'_y - u_yu'_x)
\end{aligned}$$

Here, $t'_c = [t'_{cx}, t'_{cy}, t'_{cz}]^T$, $t_c = [t_{cx}, t_{cy}, t_{cz}]^T$, $u' = [u'_x, u'_y, u'_w]^T$ and $u = [u_x, u_y, u_w]^T$, are the camera centers and the rays that connect the respective camera centers and image point with respect to the respective loop-closing frames V' and V defined in Section III. We solve for the 3 unknowns x, y and q in Equation 5 as a minimal problem which requires 3-point correspondences and we get the following system of polynomials

$$\begin{aligned} a_1xq^2 + a_2xq + a_3x + a_4yq^2 + \\ a_5yq + a_6y + a_7q^2 + a_8q + a_9 = 0 \end{aligned} \tag{6a}$$

$$\begin{aligned} b_1xq^2 + b_2xq + b_3x + b_4yq^2 + \\ b_5yq + b_6y + b_7q^2 + b_8q + b_9 = 0 \end{aligned} \tag{6b}$$

$$\begin{aligned} c_1xq^2 + c_2xq + c_3x + c_4yq^2 + \\ c_5yq + c_6y + c_7q^2 + c_8q + c_9 = 0 \end{aligned} \tag{6c}$$

where b and c are the coefficients from the additional two point correspondences with similar definition as the coefficient a. The Hidden Variable Resultant method [13] is used to solve for the unknowns in the system of polynomials. We write the system of polynomials from Equation 6 into the form of

$$\beta(q)\mathbf{X} = 0 \tag{7}$$

where $\beta(q)$ is given by

$$\begin{bmatrix} a_1q^2 + a_2q + a_3 & a_4q^2 + a_5q + a_6 & a_7q^2 + a_8q + a_9 \\ b_1q^2 + b_2q + b_3 & b_4q^2 + b_5q + b_6 & b_7q^2 + b_8q + b_9 \\ c_1q^2 + c_2q + c_3 & c_4q^2 + c_5q + c_6 & c_7q^2 + c_8q + c_9 \end{bmatrix} \tag{8}$$

and

$$\mathbf{X} = \begin{bmatrix} x & y & 1 \end{bmatrix}^{\mathbf{T}} \tag{9}$$

We know from Linear Algebra that since $\beta(q)$ is a square matrix, Equation 7 has a non-trivial solution when $det(\beta(q)) = 0$. This gives a six degree polynomial in terms of q.

$$Aq^6 + Bq^5 + Cq^4 + Dq^3 + Eq^2 + Fq + G = 0 \tag{10}$$

where the coefficients A, B, C, D and E are made up of the coefficients from Equation 6. We drop the full expressions for brevity. The roots of Equation 10 can be obtained from the eigen-values of the following Companion matrix [13]

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & -\frac{G}{A} \\ 1 & 0 & 0 & 0 & 0 & -\frac{F}{A} \\ 0 & 1 & 0 & 0 & 0 & -\frac{E}{A} \\ 0 & 0 & 1 & 0 & 0 & -\frac{D}{A} \\ 0 & 0 & 0 & 1 & 0 & -\frac{C}{A} \\ 0 & 0 & 0 & 0 & 1 & -\frac{B}{A} \end{bmatrix} \tag{11}$$

A maximum of up to six real eigen-values (i.e. six real roots to q) can be obtained from the Companion matrix and the correct solution is determined by checking the number of inliers within the RANSAC (see Section IV-D) loops. We solve for the yaw angle as $\theta = 2tan^{-1}(q)$. With q known, x can now be solved with

$$x = -\frac{d_6q^4 + d_7q^3 + d_8q^2 + d_9q + d_{10}}{d_1q^4 + d_2q^3 + d_3q^2 + d_4q + d_5} \tag{12}$$

which is obtained by eliminating y from Equations 6a and 6b. Here, d is made up of the coefficients from Equation 6.

We show only the full expression of d_{10} which has a special property in the degenerated case (see Section IV-C).

$$d_{10} = -a_9 b_6 + b_9 a_6 \tag{13}$$

Finally, y can be solved by back-substitutions of x and q into Equations 5. We also verified with the Gröbner basis [13] that six solutions is the minimal solution for our parametrization and choice of the coordinate system of the problem.

C. Degenerated Case

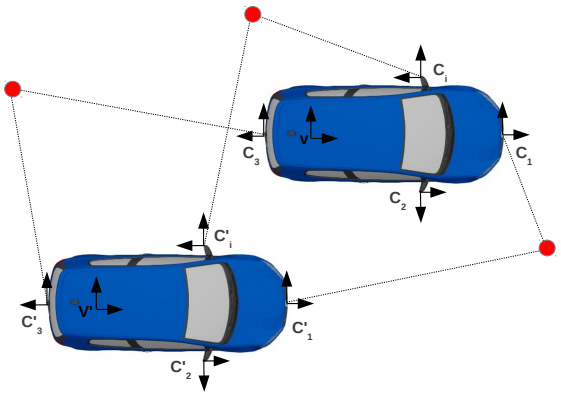


Fig. 4. Illustration of a degenerated case for the GEC.

Figure 4 shows an illustration of the degenerated case for the GEC. It happens when all the selected 3-point correspondences are matched locally over the same respective camera over the loop-closing frames V' and V, and the yaw angle is zero, i.e. $q = 0$. In this case, we observe that the camera centers stay the same over the two frames, i.e. $t'_c = t_c$, hence the first and last three terms of the coefficient a_9 from Equation 5 cancel out, i.e. $a_9 = 0$. Similarly, $b_9 = 0$ and $c_9 = 0$. We immediately see from Equation 13 that the coefficients $d_{10} = 0$, and Equation 12 becomes $d_5 x = 0$. This means that x and y cannot be uniquely identified.

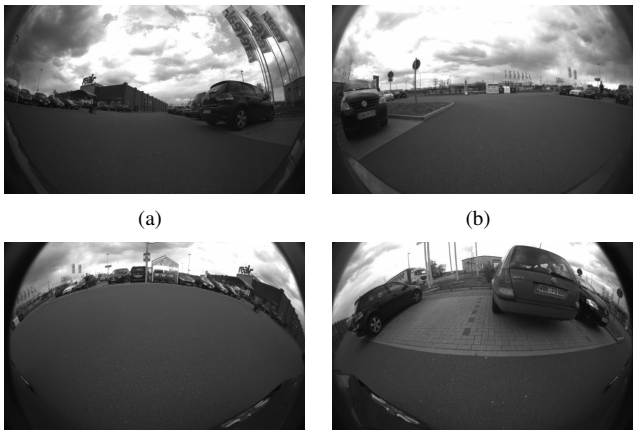
We observed experimentally that it is very rare for the yaw angle θ to be exactly or very close to zero during loop-closures for the degeneracy to happen and the overall pose-graph will not be affected by omitting the loop-closure opportunities when the yaw angle is zero. Hence, we disregard the solution of $q = 0$ and do geometric verifications for all other solutions from $q \neq 0$ when $t'_c = t_c$. Note from Equation 5 that $q = 0$ is always one of the solutions when $t'_c = t_c$ since $a_9 = 0$. In this case, no loop-closure opportunities exist if the solution from $q \neq 0$ with the highest inlier count does not pass the geometric verification test. As noted in [11], the solution does not have any scale ambiguity if it exists.

D. Robust Estimation

We reject outlier point correspondences by putting our 3-point algorithm within RANSAC [12]. Similar to our previous work [17], we do this by checking the Sampson error [9] for each point correspondence in the respective camera where the essential matrices can be computed from the hypotheses of the relative motion R and t between the loop-closing frames V' and V, and the extrinsics T_{C_i} of the camera. We also determine the correct solution from the multiple solutions of the 3-point minimal problem by counting the number of inlier within RANSAC. The correct solution gives the highest number of inliers.

The number of iterations m needed in RANSAC is given by $m = \frac{\ln(1-p)}{\ln(1-\upsilon^n)}$ where n is the number of correspondences needed to form the hypothesis, p is the probability that all selected features are inliers and υ is the probability that any selected correspondence is an inlier. Assuming that $p = 0.99$ and $\upsilon = 0.5$, a total of 34 iterations are needed for our 3-point algorithm which is a significant improvement in terms of computational efficiency compared to the 6-point, 16-point and 17-point algorithms which need 292, 301802 and 603606 iterations respectively.

V. Results

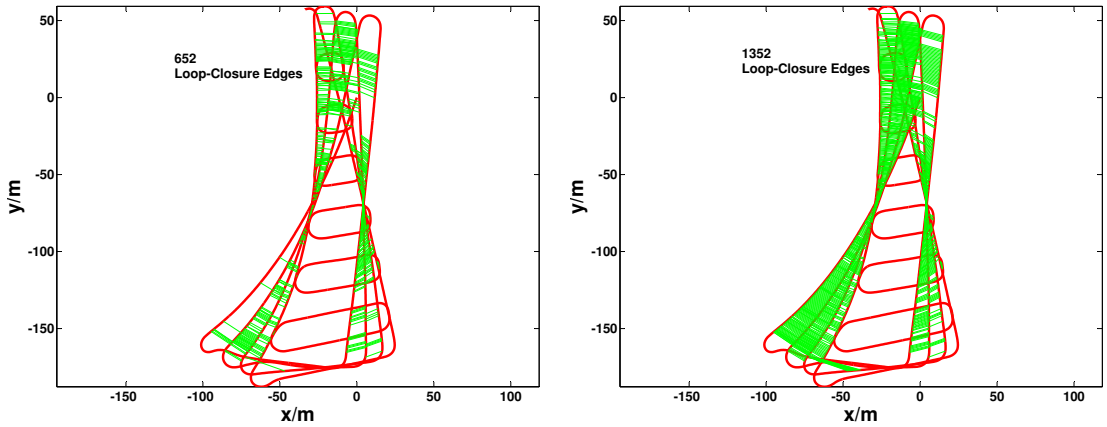


(a) (b) (c) (d)

Fig. 5. Images from the four cameras with fish-eye lens on the car. (a) Front, (b) Rear, (c) Left, (d) Right.

We implement our structureless loop-closure algorithm on the multi-camera system mounted on the car shown in Figure 1. Our multi-camera system consists of four cameras with fish-eye lens looking front, rear, left and right. Figure 5 shows an example of the images captured from the cameras. We calibrate the intrinsics of the fish-eye cameras with [19] and the extrinsics are provided by the car manufacturer. The full pipeline which includes formation of the pose-graph, place recognition, 3-Point RANSAC and geometric verification is running at approximately 8 fps on a Intel Core2 Quad CPU @ 2.40GHz $\times$ 4 with 4G of memory and GeForce GTX 285 GPU. The run-time can be further optimized by replacing the GPU SURF features that we are currently using with a more efficient feature such as ORB [20]. We implement the robust pose-graph optimization with the Google Ceres solver [1]. We show results from three datasets - (1) ParkingGarage01, (2) ParkingGarage02 and (3) Campus01. The pose-graphs of ParkingGarage01 and ParkingGarage02 are formed from wheel odometry and the

[1] http://code.google.com/p/ceres-solver/



(a) (b)

Fig. 6. Pose-graph of the ParkingGarage01 dataset from wheel odometry (red) before pose-graph optimization. Comparison on the total loop-closure edges (green) found from (a) a forward looking monocular camera and (b) our multi-camera system.

pose-graph of Campus01 is formed from visual odometry [17].

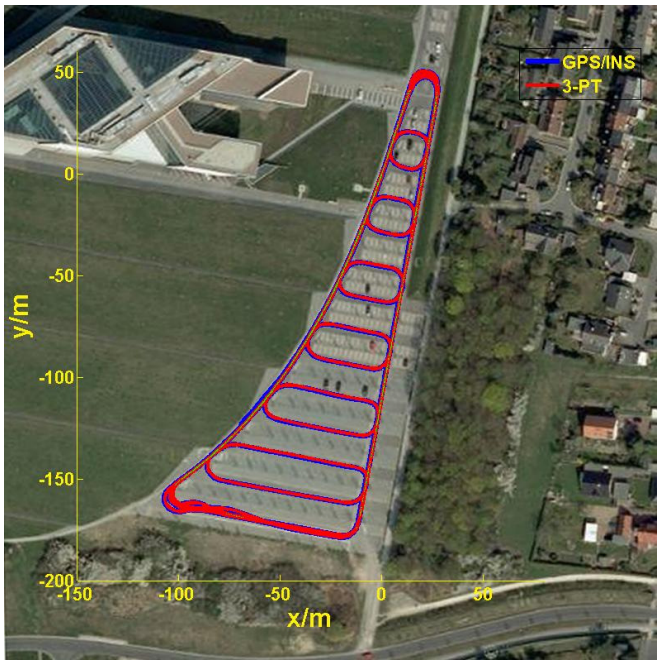
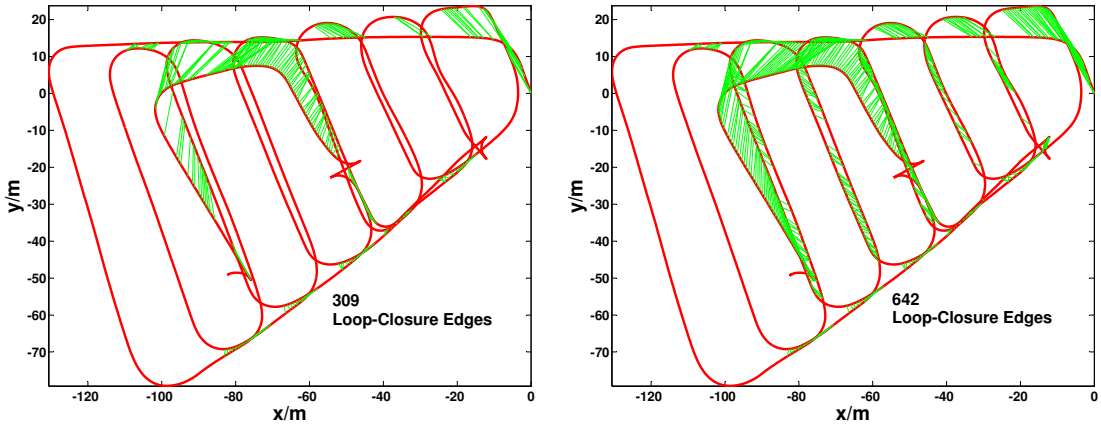


Fig. 7. Pose-graph for ParkingGarage01 dataset after pose-graph optimization (red) compared with trajectory from GPS/INS (blue) overlaid on the satellite image.

Figures 6 and 7 show the results of our algorithm on the ParkingGarage01 dataset. This dataset consists of a total of 12000 frames, i.e. 12000 $\times$ 4 images from all the four cameras covering approximately 3.5km of trajectory around a parking garage. The trajectory make a total of 4 outer and 6 nested loops. We process the dataset at a 3 frames interval and the pose-graph is formed with the wheel odometry readings (red). Figure 6 shows a comparison of the total number of loop-closure edges (green) detected by the monocular front camera in Figure 6(a) and the multi-camera system in Figure 6(b) before pose-graph optimization. Our multi-camera system successfully detected 1352 loop-closure edges which is more than twice the total of 652 loop-closure edges detected by the monocular front looking camera. We compute the loop-constraints with metric scale with our algorithm without the need to reconstruct any 3D scene points which is impossible for the monocular camera. Figure 7 shows the pose-graph after pose-graph optimization. We show the accuracy of the pose-graph after pose-graph optimization by plotting it with the GPS/INS ground truth overlaid on the satellite image. It can be seen that the pose-graph after pose-graph optimization follows the GPS/INS ground truth very closely. Note that we use the robust pose-graph optimization [14] to remove the detrimental effects of outliers and noise.

Figures 8 and 9 show the results of our algorithm on the ParkingGarage02 dataset. This dataset consists of a total of 7084 frames, i.e. 7084 $\times$ 4 images from all the four cameras covering approximately 1km of trajectory around a parking garage (a different parking garage from the ParkingGarage01 dataset). The trajectory made a total of 1 outer and 7 nested loops. Similar to the ParkingGarage01 dataset, we process the dataset at a 3 frames interval and form the pose-graph with the wheel odometry readings (red). Figure 8 shows a comparison of the total number of loop-closure edges (green) detected by the monocular front camera in Figure 8(a) and the multi-camera system in Figure 8(b) before pose-graph optimization. Our multi-camera system successfully detected 642 loop-closure edges which is more than twice the total of 309 loop-closure edges detected by the monocular front looking camera. We noticed that most of the additional loop-closure edges that were detected from our multi-camera system came from frames where the current frame and loop-closure frame are facing opposite directions. In these cases, the monocular front camera which has a limited field-of-view would not be able to detect any loop-closure. Note that this would also be true for a stereo camera. Figure 9 shows the pose-graph after pose-graph optimization. Similar to the ParkingGarage01 dataset, we show the accuracy of the pose-graph after pose-graph optimization by plotting it with the GPS/INS ground truth overlaid on the satellite image. It can



(a) (b)

Fig. 8. Pose-graph of the ParkingGarage02 dataset from wheel odometry (red) before pose-graph optimization. Comparison on the total loop-closure edges (green) found from (a) a forward looking monocular camera and (b) our multi-camera system.

be seen that the pose-graph after pose-graph optimization follows the GPS/INS ground truth very closely.

Fig. 9. Pose-graph for ParkingGarage02 dataset after pose-graph optimization (red) compared with trajectory from GPS/INS (blue) overlaid on the satellite image.

Figures 10 and 11 show the results of our algorithm on the Campus01 dataset. This dataset consists of a total of 4460 frames, i.e. 4460 $\times$ 4 images from all the four cameras covering approximately 900m of trajectory along a stretch of road within the ETH campus. The trajectory made a total of 6 loops in the shape of "∞". We process all the 4460 frames and form the pose-graph with visual odometry (red) from multi-camera based on our previous work [17]. Note that the visual odometry based on [17] is also computed without the need to reconstruct any 3D scene points. Figure 10 shows a comparison of the total number of loop-closure edges (green) detected by the monocular front camera in Figure 10(a) and the multi-camera system in Figure 10(b) before pose-graph optimization. Our multi-camera system detects 2258 loop-constraints while the front-looking monocular camera detects 2098 loop-constraints. In this case, the high number of loop-constraints detected by the front-looking camera is because most of the loop-closure paths are facing the same directions. Figure 11 shows the pose-graph after pose-graph optimization. Since we do not have the GPS/INS ground truth for this dataset, we do triangulation for the 3D scene points from the poses obtained after pose-graph optimization and overlaid these points on the satellite image to show the accuracy. It is important to note that these 3D scene points are purely for visualization of the accuracy of our algorithm and they are not used at all in the computation of the loop-closure constraints.

VI. Conclusion

In this paper, we proposed an algorithm to compute the relative pose with metric scale between two loop-closing pose-graph vertices directly from the epipolar geometry of a multi-cameras system with non/minimal overlapping field-of-views mounted on a self-driving car without the need to compute any 3D scene points. As a result, we avoid the additional time complexities and uncertainties from the reconstruction of 3D scene points which are needed by standard monocular and stereo approaches. We derived the minimal solution which requires 3-Point correspondences and showed that our 3-Point minimal solution can be implemented efficiently with RANSAC for robust estimation. We also showed that the greater flexibility in choosing a configuration for the multi-camera system to allow wider field-of-views resulted in finding more loop-closure constraints as compared to a single front-looking camera. We evaluated our algorithm with multiple large-scale datasets and the results clearly showed the viability of our algorithm.

VII. Acknowledgement

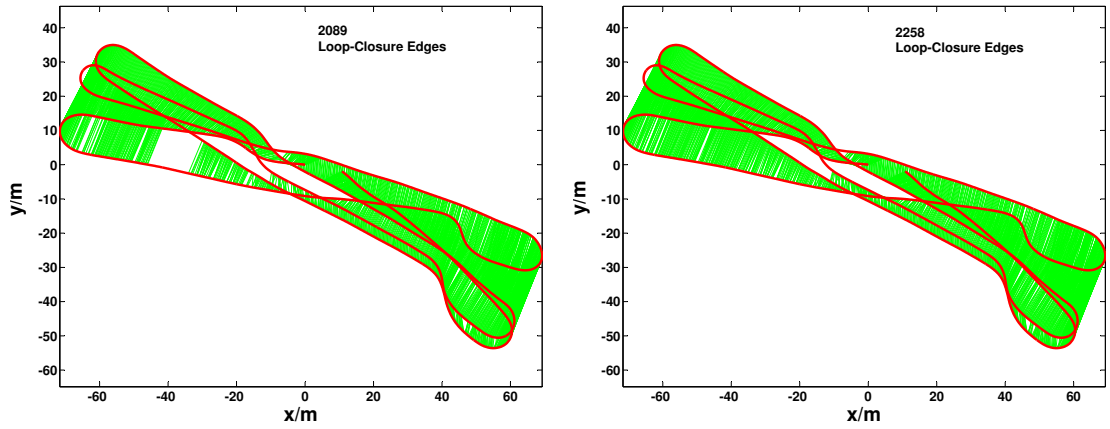
This work is supported in part by the European Community's Seventh Framework Programme (FP7/2007-2013) under grant #269916 (v-charge) and 4DVideo ERC Starting Grant Nr. 210806.

References

[1] K. Konolige, J. Bowman, J. D. Chen, P. Mihelich, M. Calonder, V. Lepetit, and P. Fua, "View-based maps," in *Robotics: Science and Systems (RSS)*, June 2009.

[2] E. Olson, J. Leonard, and S. Teller, "Fast iterative optimization of pose graphs with poor initial estimates," in *International Conference on Robotics and Automation (ICRA)*, 2006, pp. 2262–2269.

[3] G. Grisetti, R. Kummerle, C. Stachniss, and W. Burgard, "A tutorial on graph-based slam," *Intelligent Transportation Systems Magazine, IEEE*, vol. 2, no. 4, pp. 31–43, 2010.



(a) (b)

Fig. 10. Pose-graph of the Campus01 dataset from visual odometry (red) before pose-graph optimization. Comparison on the total loop-closure edges (green) found from (a) a forward looking monocular camera and (b) our multi-camera system.

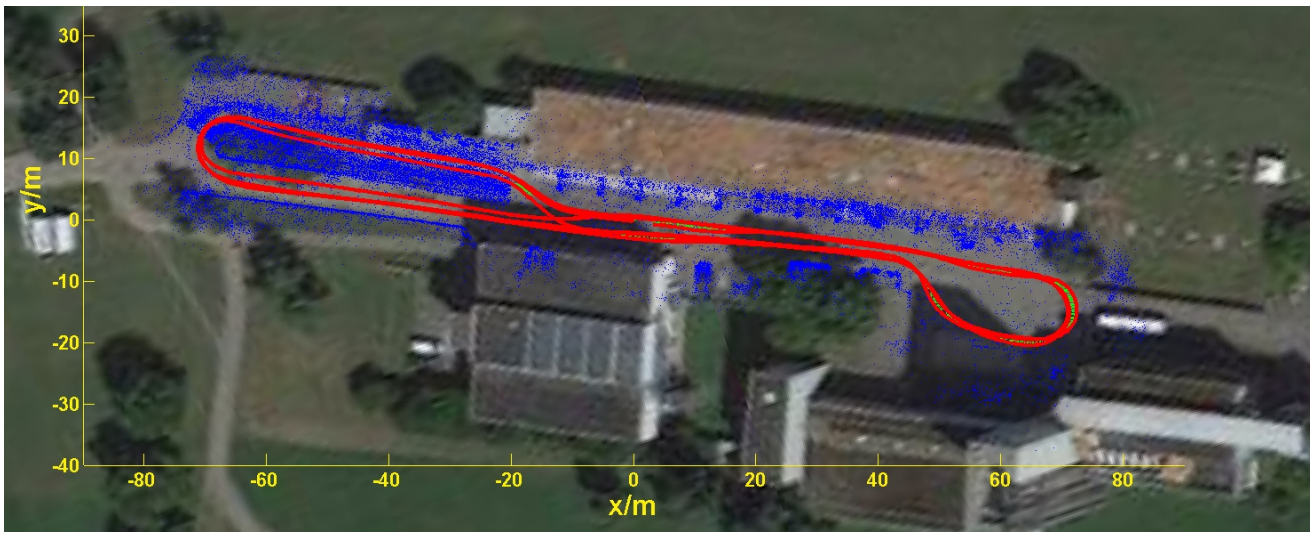


Fig. 11. Pose-graph (red) and 3D scene points (blue) for Campus01 dataset after pose-graph optimization overlaid on the satellite image.

[4] G. Grisetti, C. Stachniss, S. Grzonka, and W. Burgard, "A tree parameterization for efficiently computing maximum likelihood maps using gradient descent," in *Robotics: Science and Systems (RSS)*, 2007.

[5] D. Nistér and H. Stewénius, "Scalable recognition with a vocabulary tree," in *Computer Vision and Pattern Recognition (CVPR)*, 2006, pp. 2161–2168.

[6] D. Nister, "An efficient solution to the five-point relative pose problem," in *Pattern Analysis and Machine Intelligence (PAMI)*, vol. 26, no. 6, 2004, pp. 756–770.

[7] F. Moreno-Noguer, V. Lepetit, and P. Fua, "Accurate non-iterative o(n) solution to the pnp problem," in *International Conference on Computer Vision (ICCV)*, October 2007.

[8] S. N. Berthold K. P. Horn, Hugh M. Hilden, "Closed-form solution of absolute orientation using unit quaternions," *Journal of the Optical Society of America A*, 1987.

[9] R. I. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed. Cambridge University Press, ISBN: 978-0-521-54051-3, 2004.

[10] R. Pless, "Using many cameras as one," in *Computer Vision and Pattern Recognition (CVPR)*, vol. 2, June 2003, pp. 587–93.

[11] H. Li, R. Hartley, and J. Kim, "A linear approach to motion estimation using generalized camera models," in *Computer Vision and Pattern Recognition (CVPR)*, June 2008, pp. 1–8.

[12] 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, June 1981.

[13] D. A. Cox, J. Little, and D. O'Shea, *Ideals, varieties, and algorithms - an introduction to computational algebraic geometry and commutative algebra*, 2nd ed. Springer, ISBN: 978-0-387-94680-1, 1997.

[14] G. H. Lee, F. Faundorfer, and M. Pollefeys, "Robust pose-graph loop-closures with expectation-maximization," in *Intelligent Robots and Systems (IROS)*, 2013.

[15] P. Sturm, "Multi-view geometry for general camera models," in *Computer Vision and Pattern Recognition (CVPR)*, vol. 1, June 2005, pp. 206–212.

[16] H. Stewénius, D. Nistér, M. Oskarsson, and K. Åström, "Solutions to minimal generalized relative pose problems," in *The Workshop on Omnidirectional Vision, Camera Networks and Non-classical Cameras (OMNIVIS)*, 2005.

[17] G. H. Lee, F. Fraundorfer, and M. Pollefeys, "Motion estimation for a self-driving car with a generalized camera," in *Computer Vision and Pattern Recognition (CVPR)*, June 2013.

[18] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-up robust features (surf)," *Computer Vision and Image Understanding*, vol. 110, no. 3, pp. 346–359, June 2008.

[19] C. Mei and P. Rives, "Calibrage non biaise d'un capteur central catadioptrique," in *RFIA*, January 2006.

[20] E. Rublee, V. Rabaud, K. Konolige, and G. R. Bradski, "Orb: An efficient alternative to sift or surf," in *International Conference on Computer Vision (ICCV)*, 2011, pp. 2564–2571.