Efficient and Consistent Bundle Adjustment on Lidar Point Clouds (Supplementary)

Zheng Liu, Xiyuan Liu and Fu Zhang

Please note that equation numbers and section numbers from the main manuscript are labelled in this letter in red.

I. Applications

A. Lidar-inertial odometry with sliding window optimization

A local bundle adjustment in a sliding window of keyframes has been widely used in visual odometry and proved to be very effective in lowering the odometry drift [1]–[4]. Similar idea could apply to the lidar-inertial odometry based on our BA method. To demonstrate the effectiveness, we design a lidar-inertial odometry system as shown in Fig. 12. The system is divided into two parts: the EKF-based front-end, which provides initial yet timely pose estimation (EKF odometry), and the BA-based back-end, which refines the pose estimation in a sliding window (Local mapping). The EKF design is similar to FAST-LIO2 [5] which compensates the motion distortion in the incoming lidar scans and performs EKF propagation and update on the state manifold. The sliding window optimization performs a local BA among the most recent 20 scans considering constraints of IMU preintegration [6] and constraints from plane features co-visible among all scans in the window (Section III). After the convergence of the local BA, points in the local window are built into a k-d tree to register the next incoming scan. Furthermore, the oldest scan is removed from window and its contained points are merged into the global point cloud map.

We compare our proposed system with one state-of-art lidar-inertial odometry, FAST-LIO2 [5], which performs incremental pairwise scan registration via GICP. The comparison is conducted on "*utbm*" , "*uclk*", and "*nclt*" dataset that evaluated by FAST-LIO2, so the results of FAST-LIO2 are directly read from the original paper [7]. As can be seen in Table V, benefiting from the abundant multi-view constraints in the local sliding window, our system consistently outperforms FAST-LIO2 in terms of accuracy with considerable margins except the sequence "nclt4". The improvements in odometry accuracy confirm the effectiveness of the local BA optimization. The improvement in accuracy comes with increased computation costs as shown in the last two columns of Table V, where for FAST-LIO2, we record the time of each scan-to-map registration and for our LIO system, the time consumption is divided into two parts: one is the front-end scan-to-local map registration and the back-end local BA optimization. As can be seen, our system has a little more time consumption in front-end than FAST-LIO2 because of the building of a k-d tree, which takes a constant time overhead, while FAST-LIO2 uses a more efficient incremental k-d tree structure. In addition, our system requires an average of 73 ms in the back-end local BA optimization, which ensures a 10 Hz running frequency for both the front-end and back-end. Since the back-end runs in parallel in a separate thread, the overall odometry latency of our system is only 7 ms more than that of FAST-LIO2.

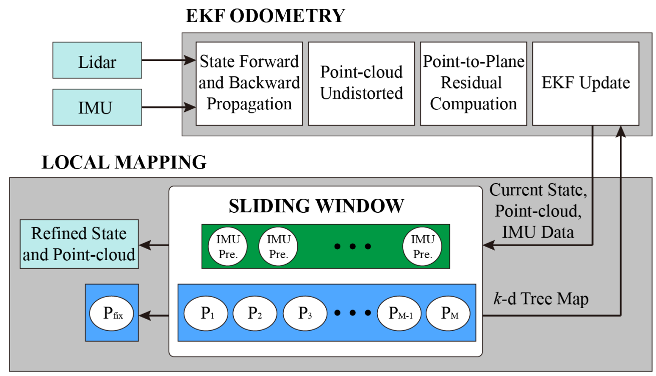


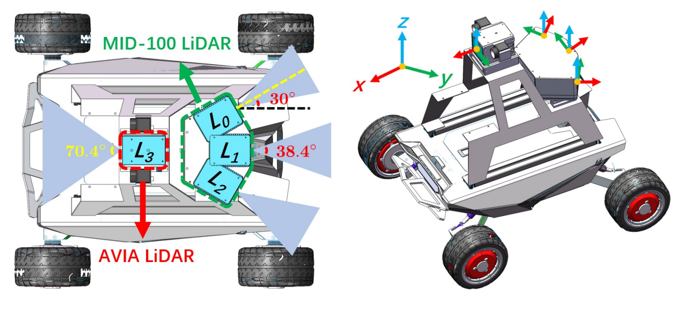
Fig. 12. Overview of the LIO system with BA and IMU preintegration

TABLE V
ATE and time cost per scan of FAST-LIO2 and LIO with BA

	ATE (m)		Time (ms)	
	FAST-LIO2	Local-BA	FAST-LIO2	Local-BA (Odom/BA Opt.)
utbm8	23.7	**20.4**	**22.05**	31.73/92.11
utbm9	45.9	**39.1**	**25.44**	33.42/95.43
utbm10	16.8	**13.1**	**22.48**	30.46/97.81
uclk4	1.31	**1.15**	**20.14**	19.53/69.89
nclt4	**8.50**	9.23	**15.72**	21.77/59.83
nclt5	6.65	**6.25**	**16.60**	26.13/61.23
nclt6	20.57	**20.34**	**15.84**	25.14/63.19
nclt7	6.58	**5.69**	**16.87**	23.18/62.53
nclt8	30.08	**26.24**	**14.25**	24.80/69.29
nclt9	5.56	**5.07**	**13.65**	22.98/67.36
nclt10	16.29	**14.10**	**21.79**	23.41/64.45
Average	16.54	**14.61**	**18.62**	25.68/73.01

B. Multiple-lidar Calibration

With the ability of concurrent optimization of multiple lidar poses, our BA method can be applied to multi-lidar extrinsic calibration. We consider the problem in [8], which aims to calibrate the extrinsic of multiple solid-state lidars shown in Fig. 13. Due to the very small FoV, these lidars have very small or even no FoV overlap. To create co-visible features, the vehicle is rotated for one cycle, during which a set of point cloud scans are collected by all lidars. Due to the rotation,



(a) (b)

Fig. 13. The customized multi-sensor vehicle platform in [8]. The MID-100 lidar consists of three lidars: L_0, L_1 and L_2, with FoV overlap between adjacent lidars 8.4°. The AVIA lidar is denoted as L_3. (a) The FoV coverage of each lidar sensor. (b) Each lidar sensor's orientation denoted in the right-handed coordinate system.

it further introduces unknown lidar poses, in addition to the extrinsic, to estimate.

To formulate the simultaneous localization and extrinsic calibration problem, we choose a reference lidar as the base and calibrate the extrinsic of the rest lidars relative to it. Denote ${}^G_R\mathbf{T}_j, j = 1, \cdots, M_p$, the pose of the reference lidar at j-th scan in a rotation, ${}^R_{L_k}\mathbf{T}$ the extrinsic of the k-th lidar relative to the base lidar. Then, following (25), the cost item corresponding to the i-th plane feature is

$$
\begin{aligned}
c_i({}^G_R\mathbf{T}_j, {}^R_{L_k}\mathbf{T}) = \lambda_3 \Bigg(\mathbf{A} \Bigg(& \sum_{j=1}^{M_p} {}^G_R\mathbf{T}_j \cdot \mathbf{C}^r_{f_{ij}} \cdot {}^G_R\mathbf{T}_j^T \\
& + \sum_{k \neq r} {}^G_R\mathbf{T}_j \cdot {}^R_{L_k}\mathbf{T} \cdot \mathbf{C}^k_{f_{ij}} \cdot \left({}^G_R\mathbf{T}_j \cdot {}^R_{L_k}\mathbf{T} \right)^T \Bigg) \Bigg)
\end{aligned} \tag{57}
$$

where $\mathbf{C}^r_{f_{ij}}, \mathbf{C}^k_{f_{ij}}$ are the point cluster of the reference lidar and the k-th lidar, respectively, observed on the i-th feature at the j-th scan. Following Theorem 4, the first and second order derivatives of (57) can be obtained by chain rules, whose details are omitted here due to limited space.

We demonstrate the advantage of the proposed BA approach in multi-lidar calibration with the latest state-of-the-art calibration methods based on ICP [9] and BALM [8]. The ICP-based method [9] optimizes the extrinsic parameter ${}^R_{L_k}\mathbf{T}$ and base lidar pose ${}^G_R\mathbf{T}_j$ by repeatedly registering the point cloud by each lidar at each scan to the rest lidar points until convergence, whereas in both BALM-based method [8] and our method, the extrinsic parameter ${}^R_{L_k}\mathbf{T}$ and base lidar pose ${}^G_R\mathbf{T}_j$ are optimized concurrently. In [9] the feature correspondence searching is conducted by a k-d tree data structure, while in [8] and our method, this is resolved by adaptive voxelization proposed in BALM [10]. To restrain the computation time, the point number in each voxel of [8] have been down-sampled to four, whereas in our method, all feature points are used.

We test these methods on two lidar setups in the system shown in Fig. 13: lidars with small field-of-view (FoV) overlap (MID-100 self calibration) and lidars without FoV overlap (AVIA and MID-100 calibration). Since the extrinsic between

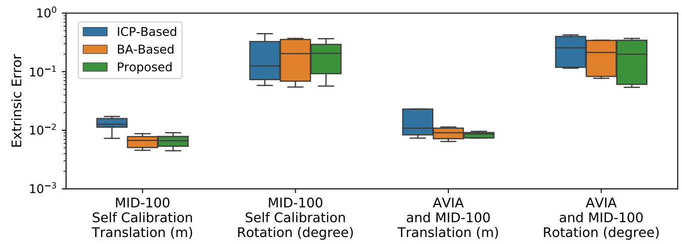


Fig. 14. Extrinsic calibration results of the ICP-based [9], BALM-based [8] and our proposed method in two experiment setups (with small or without FoV overlap).

the internal lidars of MID-100 (i.e., ${}^{L_1}_{L_0}\mathbf{T}$, ${}^{L_1}_{L_2}\mathbf{T}$) is known fromt he manufacturer's in-factory calibration, it is thus served as the ground truth in both setups. Moreover, in the MID-100 self calibration setup, the middle L_1 is chosen as the reference lidar to calibrate the extrinsic of adjacent lidars, i.e., ${}^{L_1}_{L_0}\mathbf{T}$, ${}^{L_1}_{L_2}\mathbf{T}$. This lidar setup is tested with data collected in both [9] and [8] under two scenes which contributes to eight calibration results. In the MID-100 and AVIA calibration setup, the L_3 is selected as the reference lidar to calibrate the extrinsic of each internal lidar of MID-100, i.e., ${}^{L_3}_{L_0}\mathbf{T}$, ${}^{L_3}_{L_1}\mathbf{T}$ and ${}^{L_3}_{L_2}\mathbf{T}$. We then calculate the relative pose ${}^{L_1}_{L_0}\mathbf{T}$, ${}^{L_1}_{L_2}\mathbf{T}$ from the calibrated results and compare them with the ground truth. This lidar setup is tested with data collected under two scenes in [8] since no AVIA lidar was used in [9], which contributes another four calibration results.

The total twelve independent calibration results are illustrated in Fig. 14 and Fig. 15. It is seen the our proposed method outperforms the ICP-based method [9] especially in translation. This is due to the concurrent optimization of all poses and extrinsic at the same time, which leads to full convergences within a few iterations. In contrast, the repetitive pairwise ICP registration leads to very slow convergence. The optimization did not fully converge after the maximum iteration number (*max_iter=40*), a phenomenon detailed in [8]. Furthermore, since all raw points on a feature are utilized in our method, the accuracy of our proposed work is also slightly improved when compared with the BALM-based method [8], which sample a few points on a feature to restrain the computation time. The averaged time consumption in each step of these methods have been summarized in Table VI. It is seen our proposed work and [8] has significantly shortened the calibration time in each step compared with the ICP-based method. This is due to the use of adaptive voxelization which saves a great amount of time in k-d tree build and nearest neighbor search used in [9]. Compared with [8], our proposed BA has further increased the computation speed due to the use of point cluster technique avoiding the enumerating of individual points in the BALM-based method [8].

C. Global BA on Large-Scale Dataset

In this section, we show that our proposed BA method could also be used to globally refine the quality of a large-scale lidar point cloud [11]. It is popular to use pose graph optimization (PGO) to increase the overall SLAM accuracy [12]. In PGO, the poses are optimized by minimizing the error between the

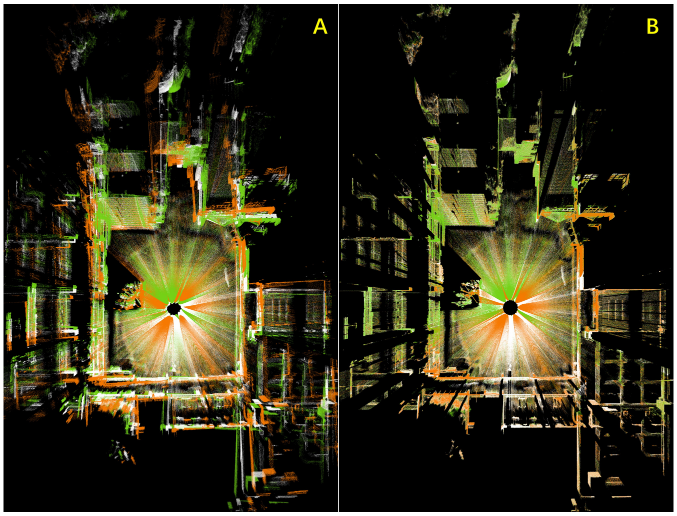


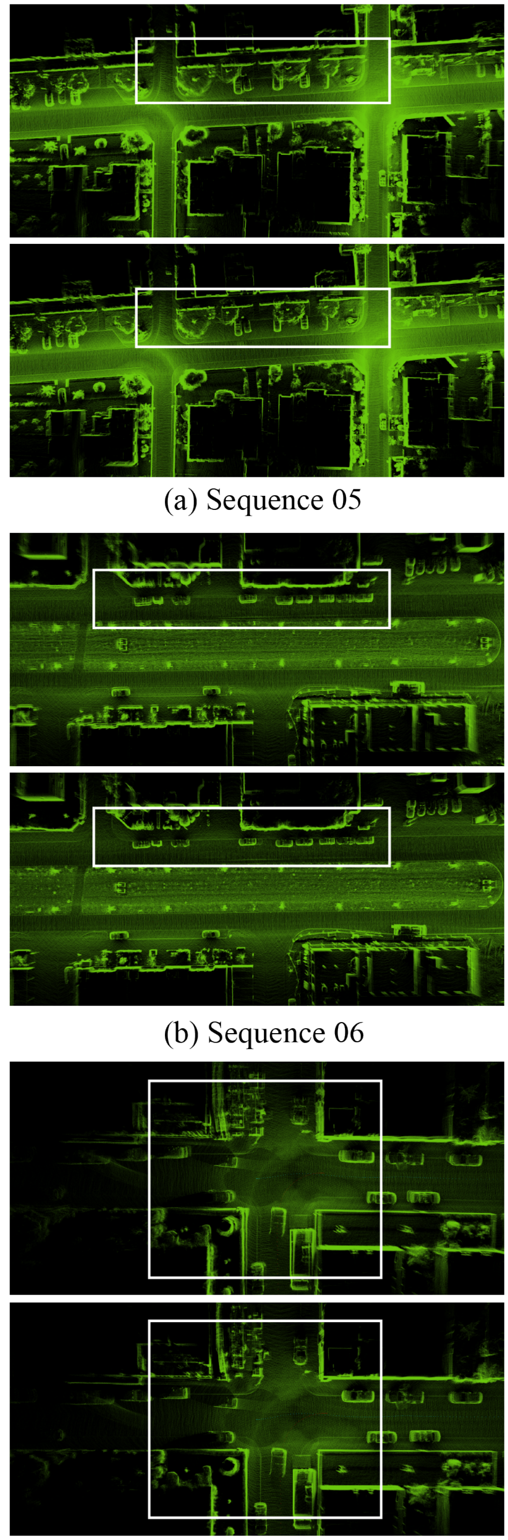
Fig. 15. The point cloud of MID-100 lidar in Scene-1 reconstructed before (A) and after (B) extrinsic optimization. The base lidar (L_1) point cloud is colored in white and the two lidars to be calibrated (L_0, L_2) are colored in orange and green, respectively.

TABLE VI
AVERAGE TIME (SECOND) PER ITERATION FOR MULTI-LIDAR CALIBRATION

Method	Pose Optimization	Extrinsic Optimization	Joint Optimization
ICP-Based [9]	5.16	6.10	14.98
BALM-Based [8]	0.23	0.58	3.09
Proposed	**0.19**	**0.20**	**0.80**

relative transformation of these poses and that estimated by the odometry. One drawback of the PGO is that it cannot reinforce the map quality directly. A divergence in point cloud mapping would occur if the estimation of the relative poses near the loop is ill. We show that the mapping quality (and odometry accuracy) could be further improved with our BA method even after PGO is performed.

We choose to validate on KITTI dataset [13]. The initial lidar odometry is provided by the state-of-the-art SLAM algorithm MULLS [12] with loop closure function enabled. We directly feed this odometry to our BA algorithm [11] (and all other BA methods under comparison) and concurrently optimize the entire poses. The optimization results are summarized in Table VII and Fig. 16. It is seen that even with loop closure function, some divergence still exist in the area near the loop of the point cloud due to ill estimations. With our proposed global BA refinement, the accuracy in odometry is further improved and the divergence in point cloud is eliminated. When compared to other BA methods, our method achieves higher accuracy while converges significantly faster as shown in Fig. 17. Moreover, we compare the results with one state-of-the-art lidar SLAM, CT-ICP [14] augmented with loop-closure [15]. The ATE of our method is lower than CT-ICP in most sequences and also the average value. For sequences that our method is outperformed by CT-ICP (i.e., sequence 02, 08, and 09), we compare the mapping quality, which is shown in Fig. 18. As can be seen, despite the larger ATE of our method, the obtained map is more consistent, which is due to the direct optimization on the mapping consistency of our global BA.



(a) Sequence 05

(b) Sequence 06

(c) Sequence 07

Fig. 16. Comparison of map consistency on KITTI dataset [13]. The image above of each subplot depicts the point cloud reconstructed by MULLS [12] which is used as our initial value. The image blow of each subplot is the point cloud optimized by our proposed global BA method.

TABLE VII
ABSOLUTE TRAJECTORY ERROR (RMSE,METERS) ON KITTI.

Sequence	00	01	02	03	04	05	06	07	08	09	10	Mean
MULLS	1.09	1.96	5.42	0.74	0.89	0.97	0.31	0.44	2.93	2.12	1.13	1.63
CT-ICP	1.68	2.25	**4.06**	0.67	0.67	0.76	0.34	0.40	**2.52**	**0.91**	0.83	1.40
EF	1.02	1.94	5.28	0.70	0.82	0.84	0.31	0.43	2.80	1.98	0.99	1.55
BALM	0.96	1.90	5.21	0.68	0.75	0.72	0.28	0.40	2.72	1.75	0.92	1.48
PA (inner)	*0.86*	*1.84*	5.08	*0.58*	**0.64**	*0.66*	*0.23*	*0.31*	2.63	1.50	*0.80*	*1.37*
BAREG	0.89	1.88	5.12	0.65	0.70	0.69	0.24	0.35	2.68	1.59	0.88	1.42
Our	**0.84**	**1.83**	*5.06*	**0.57**	**0.64**	**0.62**	**0.21**	**0.30**	*2.59*	*1.48*	**0.78**	**1.34**

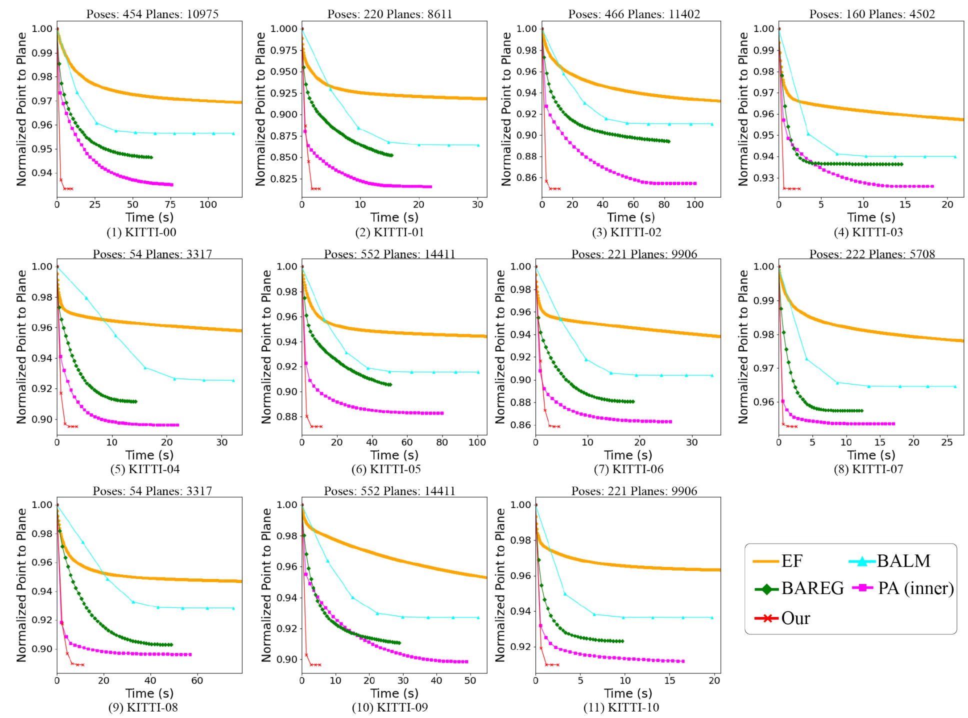


Fig. 17. Point-to-plane distance versus optimization time in KITTI dataset. All methods have the same initial pose (hence the same initial point-to-plane distance) and have their point-to-plane distance all normalized by the initial values.

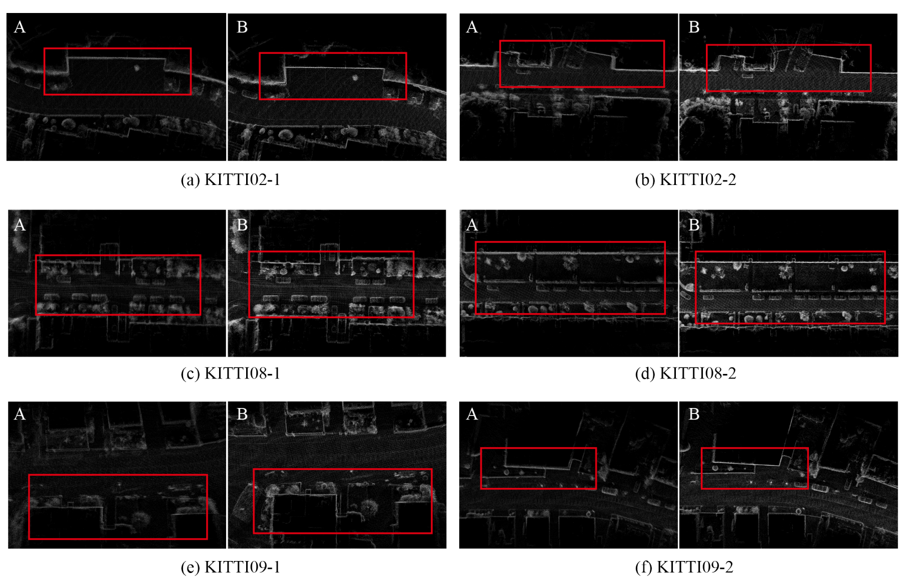


Fig. 18. Comparison of map consistency between CT-ICP(A) and our method(B) in KITTI sequence 02, 08 and 09.

II. Lemmas

Lemma 1. *For a scalar $x \in \mathbb{R}$ and a matrix $\mathbf{A} \in \mathbb{S}^{3\times 3}$ which depends on x, we have the two following conclusions.*

$$\frac{\partial \lambda_l(x)}{\partial x} = \mathbf{u}_l(x)^T \frac{\partial \mathbf{A}(x)}{\partial x} \mathbf{u}_l(x) \tag{58}$$

$$\frac{\partial \mathbf{u}_l(x)}{\partial x} = \sum_{k=1,k\neq l}^{3} \frac{1}{\lambda_l - \lambda_k} \mathbf{u}_k(x)\mathbf{u}_k(x)^T \frac{\partial \mathbf{A}(x)}{\partial x} \mathbf{u}_l(x) \tag{59}$$

where λ_l $(l = 1, 2, 3)$ denotes the l-th largest eigenvalue and $\mathbf{u}_l$ is the corresponding eigenvector.

Proof. Since the matrix $\mathbf{A}(x)$ is symmetric, its singular value decomposition is,

$$\mathbf{A}(x) = \mathbf{U}(x)\boldsymbol{\Lambda}(x)\mathbf{U}(x)^T \tag{60}$$

where $\boldsymbol{\Lambda}(x) = \operatorname{diag}(\lambda_1(x), \lambda_2(x), \lambda_3(x))$ consists of all the eigenvalues and $\mathbf{U}(x) = \begin{bmatrix}\mathbf{u}_1(x) & \mathbf{u}_2(x) & \mathbf{u}_3(x)\end{bmatrix}$ is an orthonormal matrix consisting of the eigenvectors. Therefore,

$$\boldsymbol{\Lambda}(x) = \mathbf{U}(x)^T \mathbf{A}(x)\mathbf{U}(x) \tag{61}$$

Both sides take the derivative of x,

$$\begin{aligned}\frac{\partial \boldsymbol{\Lambda}(x)}{\partial x} &= \mathbf{U}(x)^T \frac{\partial \mathbf{A}(x)}{\partial x}\mathbf{U}(x) + \mathbf{U}(x)^T\mathbf{A}(x)\frac{\partial \mathbf{U}(x)}{\partial x} \\ &\quad + \left(\frac{\partial \mathbf{U}(x)}{\partial x}\right)^T \mathbf{A}(x)\mathbf{U}(x)\end{aligned} \tag{62}$$

Since $\mathbf{U}(x)^T\mathbf{A}(x) = \boldsymbol{\Lambda}(x)\mathbf{U}(x)^T$ and $\mathbf{A}(x)\mathbf{U}(x) = \mathbf{U}(x)\boldsymbol{\Lambda}(x)$, the equation is

$$\begin{aligned}\frac{\partial \boldsymbol{\Lambda}(x)}{\partial x} &= \mathbf{U}(x)^T \frac{\partial \mathbf{A}(x)}{\partial x}\mathbf{U}(x) + \boldsymbol{\Lambda}(x)\underbrace{\mathbf{U}(x)^T\frac{\partial \mathbf{U}(x)}{\partial x}}_{\mathbf{D}(x)} \\ &\quad + \underbrace{\left(\frac{\partial \mathbf{U}(x)}{\partial x}\right)^T \mathbf{U}(x)}_{\mathbf{D}^T(x)}\boldsymbol{\Lambda}(x)\end{aligned} \tag{63}$$

Denote $\mathbf{D}(x) \triangleq \mathbf{U}(x)^T\frac{\partial \mathbf{U}(x)}{\partial x}$. Since $\mathbf{U}(x)\mathbf{U}(x)^T = \mathbf{I}$, differentiating both sides with respect to x leads to,

$$\begin{aligned}&\mathbf{U}(x)^T\frac{\partial \mathbf{U}(x)}{\partial x} + \left(\frac{\partial \mathbf{U}(x)}{\partial x}\right)^T \mathbf{U}(x) = \mathbf{0} \\ \Rightarrow\ &\mathbf{D}(x) + \mathbf{D}^T(x) = \mathbf{0}\end{aligned}$$

It is seen that $\mathbf{D}(x)$ is a skew symmetric matrix whose diagonal elements are zeros. Moreover, since $\boldsymbol{\Lambda}(x)$ is diagonal, the last two items of the right side of (63) sum to zero on diagonal positions. Only considering the diagonal elements in (63) leads to

$$\frac{\partial \lambda_l(x)}{\partial x} = \mathbf{u}_l(x)^T \frac{\partial \mathbf{A}(x)}{\partial x} \mathbf{u}_l(x), l \in \{1, 2, 3\} \tag{64}$$

which yields the first conclusion. Now we aims to prove the second one. In (63), $\frac{\partial \boldsymbol{\Lambda}(x)}{\partial x}$ is diagonal matrix and thus for the off-diagonal, k-th row, l-th column, element $(k \neq l)$,

$$0 = \mathbf{u}_k(x)^T \frac{\partial \mathbf{A}(x)}{\partial x} \mathbf{u}_l(x) + \lambda_k D_x^{k,l} - D_x^{k,l}\lambda_l \tag{65}$$

where $D_x^{k,l}$ is the k-th row, l-th column element in the skew symmetric $\mathbf{D}(x)$ and satisfy $D_x^{k,l} = -D_x^{l,k}$. From (65), we can solve $D_x^{k,l}$

$$D_x^{k,l} = \begin{cases} \dfrac{1}{\lambda_l - \lambda_k}\mathbf{u}_k(x)^T \dfrac{\partial \mathbf{A}(x)}{\partial x}\mathbf{u}_l(x), k \neq l \\ 0 \qquad\qquad , k = l\end{cases} \tag{66}$$

Since $\mathbf{D}(x) \triangleq \mathbf{U}(x)^T\frac{\partial \mathbf{U}(x)}{\partial x}$, we have $\frac{\partial \mathbf{U}(x)}{\partial x} = \mathbf{U}(x)\mathbf{D}(x)$. Taking the l-th column on both sides leads to

$$\frac{\partial \mathbf{u}_l(x)}{\partial x} = \mathbf{U}(x)\mathbf{D}_x^{:,l}, \tag{67}$$

where $\mathbf{D}_x^{:,l} \in \mathbb{R}^3$ represents the l-th column of $\mathbf{D}(x)$. Finally, substituting $\mathbf{U}(x) = [\mathbf{u}_1(x) \quad \mathbf{u}_2(x) \quad \mathbf{u}_3(x)]$ and (66) into (67), we obtain

$$\frac{\partial \mathbf{u}_l(x)}{\partial x} = \sum_{k=1,k\neq l}^{3} \frac{1}{\lambda_l - \lambda_k} \mathbf{u}_k(x)\mathbf{u}_k(x)^T \frac{\partial \mathbf{A}(x)}{\partial x} \mathbf{u}_l(x), \tag{68}$$

which yields the second conclusion.

□

Lemma 2. *Given*

(1) *Matrices* $\mathbf{C}_j = \begin{bmatrix}\mathbf{P}_j & \mathbf{v}_j \\ \mathbf{v}_j^T & N_j\end{bmatrix} \in \mathbb{S}^{4\times4}, j = 1, \cdots, M_p$;

(2) *Poses* $\mathbf{T}_j \in SE(3), j = 1, \cdots, M_p$;

(3) *A matrix* $\mathbf{C} = \begin{bmatrix}\mathbf{P} & \mathbf{v} \\ \mathbf{v}^T & N\end{bmatrix} \triangleq \sum_{j=1}^{M_p}\mathbf{T}_j\mathbf{C}_j\mathbf{T}_j^T \in \mathbb{S}^{4\times4}$, *which is the aggregation of* $\mathbf{C}_j$, *and a matrix function* $\mathbf{A}(\mathbf{C}) \triangleq \frac{1}{N}\mathbf{P} - \frac{1}{N^2}\mathbf{v}\mathbf{v}^T \in \mathbb{S}^{3\times3}$ *;*

(4) *Two constant vectors* $\mathbf{u}_k, \mathbf{u}_l \in \mathbb{R}^3$,

then the first and second order derivatives of $\mathbf{u}_k^T\mathbf{A}(\mathbf{T})\mathbf{u}_l$ *w.r.t.* $\mathbf{T}$ *are:*

$$\mathbf{g}_{kl} \triangleq \frac{\partial \mathbf{u}_k^T\mathbf{A}(\delta\mathbf{T})\mathbf{u}_l}{\partial \delta\mathbf{T}} = \begin{bmatrix}\cdots & \mathbf{g}_{kl}^j & \cdots\end{bmatrix} \in \mathbb{R}^{1\times 6M_p}, \tag{69}$$

$$\mathbf{Q}_{kl} \triangleq \frac{\partial^2 \mathbf{u}_k^T\mathbf{A}(\delta\mathbf{T})\mathbf{u}_l}{\partial (\delta\mathbf{T})^2} = \begin{bmatrix} & \vdots & \\ \cdots & \mathbf{Q}_{kl}^{ij} & \cdots \\ & \vdots & \end{bmatrix} \in \mathbb{R}^{6M_p\times 6M_p}, \tag{70}$$

where $\mathbf{g}_{kl}^j \in \mathbb{R}^{1\times6}, \mathbf{Q}_{kl}^{ij} \in \mathbb{R}^{6\times6}, \forall i, j \in \{1, \cdots, M_p\}$, *are block elements of* $\mathbf{g}_{kl}$ *and* $\mathbf{Q}_{kl}$ *defined as below*

$$\begin{aligned}\mathbf{g}_{kl}^j &= \frac{1}{N}\mathbf{u}_l^T\mathbf{S}_\mathbf{P}(\mathbf{T}_j - \frac{1}{N}\mathbf{C}\mathbf{F})\mathbf{C}_j\mathbf{T}_j^T\mathbf{V}_k^T \\ &\quad + \frac{1}{N}\mathbf{u}_k^T\mathbf{S}_\mathbf{P}(\mathbf{T}_j - \frac{1}{N}\mathbf{C}\mathbf{F})\mathbf{C}_j\mathbf{T}_j^T\mathbf{V}_l^T\end{aligned} \tag{71}$$

$$\begin{aligned}\mathbf{Q}_{kl}^{ij} &= -\frac{2}{N^2}\mathbf{V}_k\mathbf{T}_i\mathbf{C}_i\mathbf{F}\mathbf{C}_j\mathbf{T}_j^T\mathbf{V}_l^T + \mathbb{1}_{i=j}\cdot\Big(\frac{2}{N}\mathbf{V}_k\mathbf{T}_j\mathbf{C}_j\mathbf{T}_j^T\mathbf{V}_l^T \\ &\quad + \begin{bmatrix}\mathbf{K}_{kl}^j & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3}\end{bmatrix}\Big)\end{aligned} \tag{72}$$

$$\begin{aligned}\mathbf{K}_{kl}^j &= \frac{1}{N}\lfloor \mathbf{S}_\mathbf{P}\mathbf{T}_j\mathbf{C}_j(\mathbf{T}_j - \frac{1}{N}\mathbf{C}\mathbf{F})^T\mathbf{S}_\mathbf{P}^T\mathbf{u}_k\rfloor\lfloor\mathbf{u}_l\rfloor \\ &\quad + \frac{1}{N}\lfloor\mathbf{u}_k\rfloor\lfloor \mathbf{S}_\mathbf{P}\mathbf{T}_j\mathbf{C}_j(\mathbf{T}_j - \frac{1}{N}\mathbf{C}\mathbf{F})^T\mathbf{S}_\mathbf{P}^T\mathbf{u}_l\rfloor\end{aligned} \tag{73}$$

where

$$\mathbf{V}_l = \begin{bmatrix} -\lfloor \mathbf{u}_l \rfloor & \mathbf{0}_{3\times1} \\ \mathbf{0}_{3\times3} & \mathbf{u}_l \end{bmatrix} \quad \mathbf{F} = \begin{bmatrix} \mathbf{0}_{3\times3} & \mathbf{0}_{3\times1} \\ \mathbf{0}_{1\times3} & 1 \end{bmatrix} \tag{74}$$

$$\mathbf{S_P} = \begin{bmatrix} \mathbf{I}_{3\times3} & \mathbf{0}_{3\times1} \end{bmatrix} \qquad \mathbb{1}_{i=j} = \begin{cases} 1, & i=j \\ 0, & i \neq j \end{cases}. \tag{75}$$

Additionally, the block elements $\mathbf{g}_{kl}^j, \mathbf{Q}_{kl}^{ij}$ *satisfy that* $\forall \mathbf{u}_k, \mathbf{u}_l$,

$$\mathbf{g}_{kl}^j = \mathbf{0}_{1\times6}, \text{ if } \mathbf{C}_j = 0, \tag{76}$$

$$\mathbf{Q}_{kl}^{ij} = \mathbf{0}_{6\times6}, \text{ if } \mathbf{C}_i = 0 \text{ or } \mathbf{C}_j = 0. \tag{77}$$

Proof. Partition the matrix $\mathbf{C}$ as

$$\mathbf{C} = \begin{bmatrix} \mathbf{P} & \mathbf{v} \\ \mathbf{v}^T & N \end{bmatrix} = \sum_{j=1}^{M_p} \mathbf{T}_j \mathbf{C}_j \mathbf{T}_j^T, \tag{78}$$

then

$$\mathbf{P} = \mathbf{S_P C S_P}^T = \sum_{j=1}^{M_p} \left(\mathbf{R}_j \mathbf{P}_j \mathbf{R}_j^T + \mathbf{R}_j \mathbf{v}_j \mathbf{t}_j^T + \mathbf{t}_j \mathbf{v}_j^T \mathbf{R}_j^T + N_j \mathbf{t}_j \mathbf{t}_j^T \right), \tag{79}$$

$$\mathbf{v} = \mathbf{S_P C S_v}^T = \sum_{j=1}^{M_p} (\mathbf{R}_j \mathbf{v}_j + N_j \mathbf{t}_j), \tag{80}$$

$$N = \sum_{j=1}^{M_p} N_j, \tag{81}$$

where

$$\mathbf{S_P} = \begin{bmatrix} \mathbf{I}_{3\times3} & \mathbf{0}_{3\times1} \end{bmatrix} \in \mathbb{R}^{3\times4}, \tag{82}$$

$$\mathbf{S_v} = \begin{bmatrix} \mathbf{0}_{1\times3} & 1 \end{bmatrix} \in \mathbb{R}^{1\times4}. \tag{83}$$

Therefore,

$$\mathbf{A}\left(\sum_{j=1}^{M_p} \mathbf{T}_j \mathbf{C}_j \mathbf{T}_j^T\right) = \frac{1}{N}\mathbf{P} - \frac{1}{N^2}\mathbf{v}\mathbf{v}^T \tag{84}$$

$$= \frac{1}{N}\mathbf{S_P}\left(\mathbf{C} - \frac{1}{N}\mathbf{C}\mathbf{S_v}^T\mathbf{S_v}\mathbf{C}^T\right)\mathbf{S_P}^T \tag{85}$$

$$= \frac{1}{N}\mathbf{S_P}\left(\sum_{j=1}^{M_p} \mathbf{T}_j \mathbf{C}_j \mathbf{T}_j^T - \frac{1}{N}\sum_{i=1}^{M_p}\sum_{j=1}^{M_p} \mathbf{T}_i \mathbf{C}_i \mathbf{T}_i^T \mathbf{S_v}^T \mathbf{S_v} \mathbf{T}_j \mathbf{C}_j \mathbf{T}_j^T\right)\mathbf{S_P}^T. \tag{86}$$

Since $\mathbf{T}_i^T \mathbf{S_v}^T \mathbf{S_v} \mathbf{T}_j = \begin{bmatrix} \mathbf{0}_{3\times3} & \mathbf{0}_{3\times1} \\ \mathbf{0}_{1\times3} & 1 \end{bmatrix} \triangleq \mathbf{F}$, we obtain

$$\mathbf{A} = \frac{1}{N}\mathbf{S_P}\left(\sum_{j=1}^{M_p} \mathbf{T}_j \mathbf{C}_j \mathbf{T}_j^T - \frac{1}{N}\sum_{i=1}^{M_p}\sum_{j=1}^{M_p} \mathbf{T}_i \mathbf{C}_i \mathbf{F} \mathbf{C}_j \mathbf{T}_j^T\right)\mathbf{S_P}^T, \tag{87}$$

where we omitted the input argument of $\mathbf{A}$ for the sake of notation simplicity. Since $N = \sum_{j=1}^{M_p} N_j$ is a constant number that is irrelevant to the pose $\mathbf{T}$, perturbing the pose $\mathbf{T}$ (i.e., the input of $\mathbf{A}$) by $\delta\mathbf{T}$ yields

$$\mathbf{u}_k^T \mathbf{A}(\delta\mathbf{T})\mathbf{u}_l = \frac{1}{N}\mathbf{u}_k^T \mathbf{S_P}\left(\sum_{j=1}^{M_p} (\mathbf{T}_j \boxplus \delta\mathbf{T}_j)\,\mathbf{C}_j\,(\mathbf{T}_j \boxplus \delta\mathbf{T}_j)^T - \frac{1}{N}\sum_{i=1}^{M_p}\sum_{j=1}^{M_p} (\mathbf{T}_i \boxplus \delta\mathbf{T}_i)\,(\mathbf{C}_i \mathbf{F} \mathbf{C}_j)\,(\mathbf{T}_j \boxplus \delta\mathbf{T}_j)^T\right)\mathbf{S_P}^T \mathbf{u}_l. \tag{88}$$

Based on the definition of $\boxplus$ on $SE(3)$ in (27), we define,

$$\mathbf{w}_{jl}(\delta\mathbf{T}_j) \triangleq (\mathbf{T}_j \boxplus \delta\mathbf{T}_j)^T \mathbf{S_P}^T \mathbf{u}_l = \begin{bmatrix} \mathbf{R}_j^T \exp^T(\lfloor \delta\boldsymbol{\phi}_j \rfloor)\mathbf{u}_l \\ \mathbf{t}_j^T \exp^T(\lfloor \delta\boldsymbol{\phi}_j \rfloor)\mathbf{u}_l + \mathbf{u}_l^T \delta\mathbf{t}_j \end{bmatrix}. \tag{89}$$

When $\delta\boldsymbol{\phi}_j$ is small, which is indeed the case for the purpose of derivative computation, we have

$$\exp(\lfloor \delta\boldsymbol{\phi}_j \rfloor) \approx \mathbf{I} + \lfloor \delta\boldsymbol{\phi}_j \rfloor + \frac{1}{2}\lfloor \delta\boldsymbol{\phi}_j \rfloor^2 \tag{90}$$

Substituting (90) into $\mathbf{w}_{jl}(\delta\mathbf{T}_j)$, we obtain

$$\mathbf{w}_{jl}(\delta\mathbf{T}_j) \approx \begin{bmatrix} \mathbf{R}_j^T(\mathbf{I} - \lfloor \delta\boldsymbol{\phi}_j \rfloor + \frac{1}{2}\lfloor \delta\boldsymbol{\phi}_j \rfloor^2)\mathbf{u}_l \\ \mathbf{t}_j^T(\mathbf{I} - \lfloor \delta\boldsymbol{\phi}_j \rfloor + \frac{1}{2}\lfloor \delta\boldsymbol{\phi}_j \rfloor^2)\mathbf{u}_l + \mathbf{u}_l^T \delta\mathbf{t}_j \end{bmatrix} \approx \underbrace{\begin{bmatrix} \mathbf{R}_j^T \mathbf{u}_l \\ \mathbf{t}_j^T \mathbf{u}_l \end{bmatrix}}_{\bar{\mathbf{w}}_{jl}} + \underbrace{\begin{bmatrix} \mathbf{R}_j^T \lfloor \mathbf{u}_l \rfloor & \mathbf{0}_{3\times3} \\ \mathbf{t}_j^T \lfloor \mathbf{u}_l \rfloor & \mathbf{u}_l^T \end{bmatrix}}_{\mathbf{J}_{\mathbf{w}_{jl}} \in \mathbb{R}^{4\times6}} \underbrace{\begin{bmatrix} \delta\boldsymbol{\phi}_j \\ \delta\mathbf{t}_j \end{bmatrix}}_{\delta\mathbf{T}_j} + \underbrace{\begin{bmatrix} \frac{1}{2}\mathbf{R}_j^T \lfloor \delta\boldsymbol{\phi}_j \rfloor^2 \mathbf{u}_l \\ \frac{1}{2}\mathbf{t}_j^T \lfloor \delta\boldsymbol{\phi}_j \rfloor^2 \mathbf{u}_l \end{bmatrix}}_{\delta\boldsymbol{\Phi}_{jl}} \tag{91}$$

where $\bar{\mathbf{w}}_{jl}$, $\delta\boldsymbol{\Phi}_{jl}$ and $\mathbf{J}_{\mathbf{w}_{jl}}$ can be simplified as

$$\bar{\mathbf{w}}_{jl} = (\mathbf{S_P}\mathbf{T}_j)^T \mathbf{u}_l, \quad \delta\boldsymbol{\Phi}_{jl} = \frac{1}{2}(\mathbf{S_P}\mathbf{T}_j)^T \lfloor \delta\boldsymbol{\phi}_j \rfloor^2 \mathbf{u}_l$$

$$\mathbf{J}_{\mathbf{w}_{jl}} = \mathbf{T}_j^T \mathbf{V}_l^T, \quad \text{where } \mathbf{V}_l = \begin{bmatrix} -\lfloor \mathbf{u}_l \rfloor & \mathbf{0}_{3\times1} \\ \mathbf{0}_{3\times3} & \mathbf{u}_l \end{bmatrix} \tag{92}$$

Substituting (91) into (88) and keeping terms up to the second order lead to

$$\mathbf{u}_k^T \mathbf{A}(\delta\mathbf{T})\mathbf{u}_l = \frac{1}{N}\sum_{j=1}^{M_p} \mathbf{w}_{jk}^T(\delta\mathbf{T}_j)\mathbf{C}_j \mathbf{w}_{jl}(\delta\mathbf{T}_j) - \frac{1}{N^2}\sum_{i=1}^{M_p}\sum_{j=1}^{M_p} \mathbf{w}_{ik}^T(\delta\mathbf{T}_i)\mathbf{C}_i \mathbf{F} \mathbf{C}_j \mathbf{w}_{jl}(\delta\mathbf{T}_j) \tag{93}$$

$$= \frac{1}{N}\sum_{j=1}^{M_p}\Big((\bar{\mathbf{w}}_{jk}^T \mathbf{C}_j \mathbf{J}_{\mathbf{w}_{jl}} + \bar{\mathbf{w}}_{jl}^T \mathbf{C}_j \mathbf{J}_{\mathbf{w}_{jk}})\delta\mathbf{T}_j + \bar{\mathbf{w}}_{jk}^T \mathbf{C}_j \bar{\mathbf{w}}_{jl} + \delta\mathbf{T}_j^T \mathbf{J}_{\mathbf{w}_{jk}}^T \mathbf{C}_j \mathbf{J}_{\mathbf{w}_{jl}} \delta\mathbf{T}_j + \underbrace{\bar{\mathbf{w}}_{jk}^T \mathbf{C}_j \delta\boldsymbol{\Phi}_{jl} + \delta\boldsymbol{\Phi}_{jk}^T \mathbf{C}_j \bar{\mathbf{w}}_{jl}}_{\frac{1}{2}\delta\mathbf{T}_j^T \mathbf{Y}_{kl}^j \delta\mathbf{T}_j}\Big) \tag{94}$$

$$- \frac{1}{N^2}\sum_{i=1}^{M_p}\sum_{j=1}^{M_p}\Big(\delta\mathbf{T}_i^T \mathbf{J}_{\mathbf{w}_{ik}}^T \mathbf{C}_i \mathbf{F} \mathbf{C}_j \mathbf{J}_{\mathbf{w}_{jl}} \delta\mathbf{T}_j + \bar{\mathbf{w}}_{ik}^T \mathbf{C}_i \mathbf{F} \mathbf{C}_j \mathbf{J}_{\mathbf{w}_{jl}} \delta\mathbf{T}_j + \delta\mathbf{T}_i^T \mathbf{J}_{\mathbf{w}_{ik}}^T \mathbf{C}_i \mathbf{F} \mathbf{C}_j \bar{\mathbf{w}}_{jl} + \bar{\mathbf{w}}_{ik}^T \mathbf{C}_i \mathbf{F} \mathbf{C}_j \bar{\mathbf{w}}_{jl} + \underbrace{\bar{\mathbf{w}}_{ik}^T \mathbf{C}_i \mathbf{F} \mathbf{C}_j \delta\boldsymbol{\Phi}_{jl}}_{\frac{1}{2}\delta\mathbf{T}_j^T \mathbf{N}_{kl}^{ij} \delta\mathbf{T}_j} + \underbrace{\delta\boldsymbol{\Phi}_{ik}^T \mathbf{C}_i \mathbf{F} \mathbf{C}_j \bar{\mathbf{w}}_{jl}}_{\frac{1}{2}\delta\mathbf{T}_i^T \mathbf{M}_{kl}^{ij} \delta\mathbf{T}_i}\Big). \tag{95}$$

To compute $\mathbf{Y}_{kl}^j$ in $\frac{1}{2}\delta\mathbf{T}_j^T\mathbf{Y}_{kl}^j\delta\mathbf{T}_j$, note that $\mathbf{a}^T\lfloor\delta\boldsymbol{\phi}\rfloor^2\mathbf{b} = \delta\boldsymbol{\phi}^T\lfloor\mathbf{a}\rfloor\lfloor\mathbf{b}\rfloor\delta\boldsymbol{\phi}, \forall\mathbf{a},\mathbf{b},\delta\boldsymbol{\phi}\in\mathbb{R}^3$, we have

$$\begin{aligned}\bar{\mathbf{w}}_{jk}^T\mathbf{C}_j\delta\boldsymbol{\Phi}_{jl} &= \frac{1}{2}\mathbf{u}_k^T\mathbf{S_P}\mathbf{T}_j\mathbf{C}_j\mathbf{T}_j^T\mathbf{S_P}^T\lfloor\delta\boldsymbol{\phi}_j\rfloor^2\mathbf{u}_l \\ &= \frac{1}{2}\delta\boldsymbol{\phi}_j^T\lfloor\mathbf{S_P}\mathbf{T}_j\mathbf{C}_j\mathbf{T}_j^T\mathbf{S_P}^T\mathbf{u}_k\rfloor\lfloor\mathbf{u}_l\rfloor\delta\boldsymbol{\phi}_j. \end{aligned} \tag{96}$$

Similarly,

$$\delta\boldsymbol{\Phi}_{jk}^T\mathbf{C}_j\bar{\mathbf{w}}_{jl} = \frac{1}{2}\delta\boldsymbol{\phi}_j^T\lfloor\mathbf{u}_k\rfloor\lfloor\mathbf{S_P}\mathbf{T}_j\mathbf{C}_j\mathbf{T}_j^T\mathbf{S_P}^T\mathbf{u}_l\rfloor\delta\boldsymbol{\phi}_j. \tag{97}$$

Summing (96) and (97) and extending the $\delta\boldsymbol{\phi}$ into $\delta\mathbf{T}$:

$$\mathbf{Y}_{kl}^j = \begin{bmatrix}\lfloor\mathbf{S_P}\mathbf{T}_j\mathbf{C}_j\mathbf{T}_j^T\mathbf{S_P}^T\mathbf{u}_k\rfloor\lfloor\mathbf{u}_l\rfloor + \lfloor\mathbf{u}_k\rfloor\lfloor\mathbf{S_P}\mathbf{T}_j\mathbf{C}_j\mathbf{T}_j^T\mathbf{S_P}^T\mathbf{u}_l\rfloor & \mathbf{0} \\ \mathbf{0} & \mathbf{0}\end{bmatrix} \tag{98}$$

For $\frac{1}{2}\delta\mathbf{T}_j^T\mathbf{N}_{kl}^{ij}\delta\mathbf{T}_j$ and $\frac{1}{2}\delta\mathbf{T}_i^T\mathbf{M}_{kl}^{ij}\delta\mathbf{T}_i$,

$$\begin{aligned}\bar{\mathbf{w}}_{ik}^T\mathbf{C}_i\mathbf{F}\mathbf{C}_j\delta\boldsymbol{\Phi}_{jl} &= \frac{1}{2}\mathbf{u}_k^T\mathbf{S_P}\mathbf{T}_i\mathbf{C}_i\mathbf{F}\mathbf{C}_j\mathbf{T}_j^T\mathbf{S_P}^T\lfloor\delta\boldsymbol{\phi}_j\rfloor^2\mathbf{u}_l \\ &= \frac{1}{2}\delta\boldsymbol{\phi}_j^T\lfloor\mathbf{S_P}\mathbf{T}_j\mathbf{C}_j\mathbf{F}\mathbf{C}_i\mathbf{T}_i^T\mathbf{S_P}^T\mathbf{u}_k\rfloor\lfloor\mathbf{u}_l\rfloor\delta\boldsymbol{\phi}_j \end{aligned} \tag{99}$$

$$\begin{aligned}\delta\boldsymbol{\Phi}_{ik}^T\mathbf{C}_i\mathbf{F}\mathbf{C}_j\bar{\mathbf{w}}_{jl} &= \frac{1}{2}\mathbf{u}_k^T\lfloor\delta\boldsymbol{\phi}_i\rfloor^2\mathbf{S_P}\mathbf{T}_i\mathbf{C}_i\mathbf{F}\mathbf{C}_j\mathbf{T}_j^T\mathbf{S_P}^T\mathbf{u}_l \\ &= \frac{1}{2}\delta\boldsymbol{\phi}_i^T\lfloor\mathbf{u}_k\rfloor\lfloor\mathbf{S_P}\mathbf{T}_i\mathbf{C}_i\mathbf{F}\mathbf{C}_j\mathbf{T}_j^T\mathbf{S_P}^T\mathbf{u}_l\rfloor\delta\boldsymbol{\phi}_i \end{aligned} \tag{100}$$

Thus, extending the $\delta\boldsymbol{\phi}$ into $\delta\mathbf{T}$, we obtain

$$\mathbf{N}_{kl}^{ij} = \begin{bmatrix}\lfloor\mathbf{S_P}\mathbf{T}_j\mathbf{C}_j\mathbf{F}\mathbf{C}_i\mathbf{T}_i^T\mathbf{S_P}^T\mathbf{u}_k\rfloor\lfloor\mathbf{u}_l\rfloor & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3}\end{bmatrix} \tag{101}$$

$$\mathbf{M}_{kl}^{ij} = \begin{bmatrix}\lfloor\mathbf{u}_k\rfloor\lfloor\mathbf{S_P}\mathbf{T}_i\mathbf{C}_i\mathbf{F}\mathbf{C}_j\mathbf{T}_j^T\mathbf{S_P}^T\mathbf{u}_l\rfloor & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3}\end{bmatrix} \tag{102}$$

It can be seen that (95) is quadratic w.r.t. $\delta\mathbf{T}$, so we cast it into the following standard form

$$\mathbf{u}_k^T\mathbf{A}(\delta\mathbf{T})\mathbf{u}_l = r_{kl} + \mathbf{g}_{kl}\cdot\delta\mathbf{T} + \frac{1}{2}\delta\mathbf{T}^T\cdot\mathbf{Q}_{kl}\cdot\delta\mathbf{T}, \tag{103}$$

where $\mathbf{g}_{kl}$ and $\mathbf{Q}_{kl}$ are partitioned as

$$\mathbf{g}_{kl} = \begin{bmatrix}\cdots & \mathbf{g}_{kl}^j & \cdots\end{bmatrix} \in \mathbb{R}^{1\times6M_p} \tag{104}$$

$$\mathbf{Q}_{kl} = \begin{bmatrix} & \vdots & \\ \cdots & \mathbf{Q}_{kl}^{ij} & \cdots \\ & \vdots & \end{bmatrix} \in \mathbb{R}^{6M_p\times6M_p}. \tag{105}$$

For $\mathbf{g}_{kl}^j \in \mathbb{R}^{1\times6}, \forall j\in\{1,\cdots,M_p\}$, the j-th column block of $\mathbf{g}_{kl}$ in (104), it is

$$\begin{aligned}\mathbf{g}_{kl}^j &= \frac{1}{N}(\bar{\mathbf{w}}_{jk}^T\mathbf{C}_j\mathbf{J}_{\mathbf{w}_{jl}} + \bar{\mathbf{w}}_{jl}^T\mathbf{C}_j\mathbf{J}_{\mathbf{w}_{jk}}) \\ &\quad - \frac{1}{N^2}\sum_{i=1}^{M_p}(\bar{\mathbf{w}}_{ik}^T\mathbf{C}_i\mathbf{F}\mathbf{C}_j\mathbf{J}_{\mathbf{w}_{jl}} + \bar{\mathbf{w}}_{il}^T\mathbf{C}_i\mathbf{F}\mathbf{C}_j\mathbf{J}_{\mathbf{w}_{jk}}) \end{aligned} \tag{106}$$

$$\begin{aligned} &= \frac{1}{N}(\mathbf{u}_k^T\mathbf{S_P}\mathbf{T}_j\mathbf{C}_j\mathbf{T}_j^T\mathbf{V}_l^T + \mathbf{u}_l^T\mathbf{S_P}\mathbf{T}_j\mathbf{C}_j\mathbf{T}_j^T\mathbf{V}_k^T) \\ &\quad - \frac{1}{N^2}\sum_{i=1}^{M_p}(\mathbf{u}_k^T\mathbf{S_P}\mathbf{T}_i\mathbf{C}_i\mathbf{F}\mathbf{C}_j\mathbf{T}_j^T\mathbf{V}_l^T + \mathbf{u}_l^T\mathbf{S_P}\mathbf{T}_i\mathbf{C}_i\mathbf{F}\mathbf{C}_j\mathbf{T}_j^T\mathbf{V}_k^T) \\ &= \frac{1}{N}\mathbf{u}_l^T\mathbf{S_P}(\mathbf{T}_j\mathbf{C}_j - \frac{1}{N}\sum_{i=1}^{M_p}\mathbf{T}_i\mathbf{C}_i\mathbf{F}\mathbf{C}_j)\mathbf{T}_j^T\mathbf{V}_k^T \\ &\quad + \frac{1}{N}\mathbf{u}_k^T\mathbf{S_P}(\mathbf{T}_j\mathbf{C}_j - \frac{1}{N}\sum_{i=1}^{M_p}\mathbf{T}_i\mathbf{C}_i\mathbf{F}\mathbf{C}_j)\mathbf{T}_j^T\mathbf{V}_l^T \end{aligned} \tag{107}$$

Since $\mathbf{v} = \sum_{j=1}^{M_p}(\mathbf{R}_j\mathbf{v}_j + N_j\mathbf{t}_j)$ from (80) and $\mathbf{F} = \begin{bmatrix}\mathbf{0}_{3\times1} & \mathbf{0}_{3\times1} \\ \mathbf{0}_{1\times3} & 1\end{bmatrix}$, we have

$$\begin{aligned}\sum_{i=1}^{M_p}\mathbf{T}_i\mathbf{C}_i\mathbf{F} &= \sum_{i=1}^{M_p}\begin{bmatrix}\mathbf{R}_i & \mathbf{t}_i \\ \mathbf{0} & 1\end{bmatrix}\begin{bmatrix}\mathbf{P}_i & \mathbf{v}_i \\ \mathbf{v}_i^T & N_i\end{bmatrix}\begin{bmatrix}\mathbf{0}_{3\times1} & \mathbf{0}_{3\times1} \\ \mathbf{0}_{1\times3} & 1\end{bmatrix} \\ &= \sum_{i=1}^{M_p}\begin{bmatrix}\mathbf{0}_{3\times1} & \mathbf{R}_i\mathbf{v}_i + N_i\mathbf{t}_i \\ \mathbf{0}_{1\times3} & N_i\end{bmatrix} = \begin{bmatrix}\mathbf{0} & \mathbf{v} \\ \mathbf{0} & N_i\end{bmatrix} = \mathbf{CF}. \end{aligned} \tag{108}$$

Hence, the part $\left(\mathbf{T}_j\mathbf{C}_j - \frac{1}{N}\sum_{i=1}^{M_p}\mathbf{T}_i\mathbf{C}_i\mathbf{F}\mathbf{C}_j\right)$ in (107) is

$$\begin{aligned}\mathbf{T}_j\mathbf{C}_j - \frac{1}{N}\sum_{i=1}^{M_p}\mathbf{T}_i\mathbf{C}_i\mathbf{F}\mathbf{C}_j &= \mathbf{T}_j\mathbf{C}_j - \frac{1}{N}\underbrace{\left(\sum_{i=1}^{M_p}\mathbf{T}_i\mathbf{C}_i\mathbf{F}\right)}_{\mathbf{CF}}\mathbf{C}_j \\ &= (\mathbf{T}_j - \frac{1}{N}\mathbf{CF})\mathbf{C}_j. \end{aligned} \tag{109}$$

Therefore,

$$\begin{aligned}\mathbf{g}_{kl}^j &= \frac{1}{N}\mathbf{u}_l^T\mathbf{S_P}(\mathbf{T}_j - \frac{1}{N}\mathbf{CF})\mathbf{C}_j\mathbf{T}_j^T\mathbf{V}_k^T \\ &\quad + \frac{1}{N}\mathbf{u}_k^T\mathbf{S_P}(\mathbf{T}_j - \frac{1}{N}\mathbf{CF})\mathbf{C}_j\mathbf{T}_j^T\mathbf{V}_l^T, \end{aligned} \tag{110}$$

which yields the result in (71). Additionally, it is seen that if $\mathbf{C}_j = \mathbf{0}_{4\times4}$, $\mathbf{g}_{kl}^j = \mathbf{0}_{1\times6}, \forall\mathbf{u}_k,\mathbf{u}_l$.

For $\mathbf{Q}_{kl}^{ij}\in\mathbb{R}^{6\times6}, \forall i,j\in\{1,\cdots,M_p\}$, the i-th row, j-th column block of $\mathbf{Q}_{kl}$ in (105), it is

$$\begin{aligned}\mathbf{Q}_{kl}^{ij} &= -\frac{2}{N^2}\mathbf{J}_{\mathbf{w}_{ik}}^T\mathbf{C}_i\mathbf{F}\mathbf{C}_j\mathbf{J}_{\mathbf{w}_{jl}} + \mathbb{1}_{i=j}\cdot\Big(\frac{2}{N}\mathbf{J}_{\mathbf{w}_{jk}}^T\mathbf{C}_j\mathbf{J}_{\mathbf{w}_{jl}} \\ &\quad + \frac{1}{N}\mathbf{Y}_{kl}^j - \frac{1}{N^2}\sum_{\nu=1}^{M_p}(\mathbf{N}_{kl}^{\nu j} + \mathbf{M}_{kl}^{i\nu})\Big) \end{aligned} \tag{111}$$

where

$$\mathbb{1}_{i=j} = \begin{cases} 1, & i = j \\ 0, & i \neq j \end{cases} \tag{112}$$

$$\mathbf{J}_{\mathbf{w}_{ik}}^T \mathbf{C}_i \mathbf{F} \mathbf{C}_j \mathbf{J}_{\mathbf{w}_{jl}} = \mathbf{V}_k \mathbf{T}_i \mathbf{C}_i \mathbf{F} \mathbf{C}_j \mathbf{T}_j^T \mathbf{V}_l^T \tag{113}$$

$$\mathbf{J}_{\mathbf{w}_{jk}}^T \mathbf{C}_j \mathbf{J}_{\mathbf{w}_{jl}} = \mathbf{V}_k \mathbf{T}_j \mathbf{C}_j \mathbf{T}_j^T \mathbf{V}_l^T \tag{114}$$

$$\begin{aligned}\sum_{\nu=1}^{M_p} (\mathbf{N}_{kl}^{\nu j} + \mathbf{M}_{kl}^{i\nu}) = &\sum_{\nu=1}^{M_p} \begin{bmatrix} \lfloor \mathbf{S_P} \mathbf{T}_j \mathbf{C}_j \mathbf{F} \mathbf{C}_\nu \mathbf{T}_\nu^T \mathbf{S_P}^T \mathbf{u}_k \rfloor \lfloor \mathbf{u}_l \rfloor & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \\ &+ \sum_{\nu=1}^{M_p} \begin{bmatrix} \lfloor \mathbf{u}_k \rfloor \lfloor \mathbf{S_P} \mathbf{T}_i \mathbf{C}_i \mathbf{F} \mathbf{C}_\nu \mathbf{T}_\nu^T \mathbf{S_P}^T \mathbf{u}_l \rfloor & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}\end{aligned} \tag{115}$$

By using similar method in (108), we have

$$\sum_{\nu=1}^{M_p} \mathbf{F} \mathbf{C}_\nu \mathbf{T}_\nu^T = \mathbf{F} \mathbf{C} \tag{116}$$

and

$$\begin{aligned}&\sum_{\nu=1}^{M_p} (\mathbf{N}_{kl}^{\nu j} + \mathbf{M}_{kl}^{i\nu}) = \\ &\begin{bmatrix} \lfloor \mathbf{S_P} \mathbf{T}_j \mathbf{C}_j \mathbf{F} \mathbf{C} \mathbf{S_P}^T \mathbf{u}_k \rfloor \lfloor \mathbf{u}_l \rfloor + \lfloor \mathbf{u}_k \rfloor \lfloor \mathbf{S_P} \mathbf{T}_i \mathbf{C}_i \mathbf{F} \mathbf{C} \mathbf{S_P}^T \mathbf{u}_l \rfloor & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}\end{aligned} \tag{117}$$

Thus, the matrix $\mathbf{Q}_{kl}^{ij}$ is

$$\begin{aligned}\mathbf{Q}_{kl}^{ij} = -\frac{2}{N^2} \mathbf{V}_k \mathbf{T}_i \mathbf{C}_i \mathbf{F} \mathbf{C}_j \mathbf{T}_j^T \mathbf{V}_l^T + \mathbb{1}_{i=j} \cdot \Big(&\frac{2}{N} \mathbf{V}_k \mathbf{T}_j \mathbf{C}_j \mathbf{T}_j^T \mathbf{V}_l^T \\ &+ \begin{bmatrix} \mathbf{K}_{kl}^j & \mathbf{0}_{3\times 3} \\ \mathbf{0}_{3\times 3} & \mathbf{0}_{3\times 3} \end{bmatrix} \Big)\end{aligned} \tag{118}$$

$$\begin{aligned}\mathbf{K}_{kl}^j &= \frac{1}{N} \mathbf{Y}_{kl}^j - \frac{1}{N^2} \sum_{\nu=1}^{M_p} (\mathbf{N}_{kl}^{\nu j} + \mathbf{M}_{kl}^{i\nu}), \quad \text{with } i = j \\ &= \frac{1}{N} \lfloor \mathbf{S_P} \mathbf{T}_j \mathbf{C}_j (\mathbf{T}_j - \frac{1}{N} \mathbf{C} \mathbf{F})^T \mathbf{S_P}^T \mathbf{u}_k \rfloor \lfloor \mathbf{u}_l \rfloor \\ &\quad + \frac{1}{N} \lfloor \mathbf{u}_k \rfloor \lfloor \mathbf{S_P} \mathbf{T}_j \mathbf{C}_j (\mathbf{T}_j - \frac{1}{N} \mathbf{C} \mathbf{F})^T \mathbf{S_P}^T \mathbf{u}_l \rfloor,\end{aligned} \tag{119}$$

which yields the results in (72) and (73). Additionally, it can be seen that if $\mathbf{C}_i = \mathbf{0}_{4\times 4}$ or $\mathbf{C}_j = \mathbf{0}_{4\times 4}$, we have $\mathbf{C}_i \mathbf{F} \mathbf{C}_j = \mathbf{0}_{4\times 4}$, $\mathbf{K}_{kl}^j = \mathbf{0}_{3\times 3}$ and then $\mathbf{Q}_{kl}^{ij} = \mathbf{0}_{6\times 6}, \forall \mathbf{u}_k, \mathbf{u}_l$. □

III. Proof of Theorems

A. Proof of formula (6) and (8)

Proof. The variable to be optimized is $\boldsymbol{\pi}_i = (\mathbf{n}_i, \mathbf{q}_i)$ and the cost function is

$$c_i = \min_{\boldsymbol{\pi}_i = (\mathbf{n}_i, \mathbf{q}_i)} \left(\frac{1}{N_i} \sum_{j=1}^{M_p} \sum_{k=1}^{N_{ij}} \left\| \mathbf{h}_i^T (\mathbf{p}_{ijk} - \mathbf{q}_i) \right\|_2^2 \right) \tag{120}$$

where $\mathbf{h}_i = \mathbf{n}_i$ for plane feature and $\mathbf{h}_i = (\mathbf{I} - \mathbf{n}_i \mathbf{n}_i^T)$ for edge feature. The dimensions of $\mathbf{h}_i$ may be different for these two features but it has no influence on following derivation.

$$\begin{aligned}c_i &= \min_{\mathbf{n}_i} \min_{\mathbf{q}_i} \left(\frac{1}{N_i} \sum_{j=1}^{M_p} \sum_{k=1}^{N_{ij}} \left\| \mathbf{h}_i^T (\mathbf{p}_{ijk} - \mathbf{q}_i) \right\|_2^2 \right) \\ &= \min_{\mathbf{n}_i} \left(\min_{\mathbf{q}_i} \left(\frac{1}{N_i} \sum_{j=1}^{M_p} \sum_{k=1}^{N_{ij}} (\mathbf{p}_{ijk} - \mathbf{q}_i)^T \mathbf{h}_i \mathbf{h}_i^T (\mathbf{p}_{ijk} - \mathbf{q}_i) \right) \right).\end{aligned} \tag{121}$$

As can be seen, the inner optimization on $\mathbf{q}_i$ is a standard quadratic optimization problem. So, the optimum $\mathbf{q}_i^\star$ can be solved by setting the derivative to zero:

$$\begin{aligned}&2 \sum_{j=1}^{M_p} \sum_{k=1}^{N_{ij}} \mathbf{h}_i \mathbf{h}_i^T (\mathbf{p}_{ijk} - \mathbf{q}_i) = \mathbf{0} \implies \\ &2 \mathbf{h}_i \mathbf{h}_i^T \left(\sum_{j=1}^{M_p} \sum_{k=1}^{N_{ij}} \mathbf{p}_{ijk} - N_i \mathbf{q}_i \right) = \mathbf{0}\end{aligned} \tag{122}$$

where $N_i = \sum_{j=1}^{M_p} \sum_{k=1}^{N_{ij}} 1 = \sum_{j=1}^{M_p} N_{ij}$. This equation does not lead to a unique solution of $\mathbf{q}_i$, one particular optimum solution is $\mathbf{q}_i^\star = \frac{1}{N_i} \sum_{j=1}^{M_p} \sum_{k=1}^{N_{ij}} \mathbf{p}_{ijk} = \bar{\mathbf{p}}_i$ as is defined in (7).

Now, substituting the optimum solution $\mathbf{q}_i^\star = \bar{\mathbf{p}}_i$ into (121) leads to:

$$c_i = \min_{\mathbf{n}_i} \left(\frac{1}{N_i} \sum_{j=1}^{M_p} \sum_{k=1}^{N_{ij}} \left\| \mathbf{h}_i^T (\mathbf{p}_{ijk} - \bar{\mathbf{p}}_i) \right\|_2^2 \right) \tag{123}$$

To solve for the optimal parameter $\mathbf{n}_i$ in the above optimization problem, we discuss the case of plane and edge features separately, as follows.

1) Plane feature: $\mathbf{h}_i = \mathbf{n}_i$.

$$\begin{aligned}c_i &= \min_{\|\mathbf{n}_i\|_2 = 1} \left(\frac{1}{N_i} \sum_{j=1}^{M_p} \sum_{k=1}^{N_{ij}} \left\| \mathbf{n}_i^T (\mathbf{p}_{ijk} - \bar{\mathbf{p}}_i) \right\|_2^2 \right) \\ &= \min_{\|\mathbf{n}_i\|_2 = 1} \left(\frac{1}{N_i} \sum_{j=1}^{M_p} \sum_{k=1}^{N_{ij}} \left(\mathbf{n}_i^T (\mathbf{p}_{ijk} - \bar{\mathbf{p}}_i)(\mathbf{p}_{ijk} - \bar{\mathbf{p}}_i)^T \mathbf{n}_i \right) \right) \\ &= \min_{\mathbf{n}_i} \mathbf{n}_i^T \mathbf{A}_i \mathbf{n}_i,\end{aligned} \tag{124}$$

where $\mathbf{A}_i$ is defined in (7) and is a symmetric matrix. Performing Singular Value Decomposition (SVD) of $\mathbf{A}_i$

$$\mathbf{A}_i = \mathbf{U}_i \boldsymbol{\Lambda}_i \mathbf{U}_i^T \tag{125}$$

where

$$\mathbf{U}_i = [\mathbf{u}_1 \quad \mathbf{u}_2 \quad \mathbf{u}_3] \quad \boldsymbol{\Lambda}_i = \operatorname{diag}(\lambda_1 \quad \lambda_2 \quad \lambda_3) \tag{126}$$

with $\lambda_1 \geq \lambda_2 \geq \lambda_3$ and $\mathbf{U}_i^T \mathbf{U}_i = \mathbf{I}$.

Denote $\mathbf{m} = \mathbf{U}_i \mathbf{n}_i = \begin{bmatrix} m_1 & m_2 & m_3 \end{bmatrix}^T$, $\|\mathbf{m}\|_2 = \sqrt{\mathbf{n}_i^T \mathbf{U}_i^T \mathbf{U}_i \mathbf{n}_i} = 1$, then (124) reduces to

$$\begin{aligned}c_i &= \min_{\|\mathbf{n}_i\|_2 = 1} (\mathbf{n}_i^T \mathbf{U}_i \boldsymbol{\Lambda}_i \mathbf{U}_i^T \mathbf{n}_i) = \min_{\|\mathbf{m}\|_2 = 1} (\mathbf{m}^T \boldsymbol{\Lambda}_i \mathbf{m}) \\ &= \min_{\|\mathbf{m}\|_2 = 1} (\lambda_1 m_1^2 + \lambda_2 m_2^2 + \lambda_3 m_3^2) \\ &\geq \min_{\|\mathbf{m}\|_2 = 1} (\lambda_3 m_1^2 + \lambda_3 m_2^2 + \lambda_3 m_3^2) = \lambda_3,\end{aligned} \tag{127}$$

where the minimum value λ_3 is reached when $m_3 = 1$, i.e., $\mathbf{m}^\star = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}^T$ and $\mathbf{n}_i^\star = \mathbf{U}_i \mathbf{m}^\star = \mathbf{u}_3$.

Therefore, the optimal cost is $\lambda_3(\mathbf{A}_i)$ and the optimum solution is $\mathbf{n}^\star = \mathbf{u}_3(\mathbf{A}_i)$ and $\mathbf{q}^\star = \bar{\mathbf{p}}_i$.

2) Edge feature: $\mathbf{h}_i = \mathbf{I} - \mathbf{n}_i\mathbf{n}_i^T$, then $\mathbf{h}_i\mathbf{h}_i^T = \mathbf{h}_i$ and

$$\begin{aligned}
c_i &= \min_{\mathbf{n}_i} \left(\frac{1}{N_i} \sum_{j=1}^{M_p} \sum_{k=1}^{N_{ij}} \left\| (\mathbf{I} - \mathbf{n}_i\mathbf{n}_i^T)(\mathbf{p}_{ijk} - \bar{\mathbf{p}}_i) \right\|_2^2 \right) \\
&= \min_{\mathbf{n}_i} \left(\frac{1}{N_i} \sum_{j=1}^{M_p} \sum_{k=1}^{N_{ij}} (\mathbf{p}_{ijk} - \bar{\mathbf{p}}_i)^T (\mathbf{I} - \mathbf{n}_i\mathbf{n}_i^T)(\mathbf{p}_{ijk} - \bar{\mathbf{p}}_i) \right) \\
&= \min_{\mathbf{n}_i} \left(\frac{1}{N_i} \sum_{j=1}^{M_p} \sum_{k=1}^{N_{ij}} (\mathbf{p}_{ijk} - \bar{\mathbf{p}}_i)^T (\mathbf{p}_{ijk} - \bar{\mathbf{p}}_i) \right. \\
&\qquad \left. - \mathbf{n}_i^T \left(\frac{1}{N_i} \sum_{j=1}^{M_p} \sum_{k=1}^{N_{ij}} (\mathbf{p}_{ijk} - \bar{\mathbf{p}}_i)^T (\mathbf{p}_{ijk} - \bar{\mathbf{p}}_i) \right) \mathbf{n}_i \right) \\
&= \min_{\mathbf{n}_i} (\operatorname{trace}(\mathbf{A}_i) - \mathbf{n}_i^T \mathbf{A}_i \mathbf{n}_i) \\
&= \lambda_1 + \lambda_2 + \lambda_3 - \underbrace{\max_{\mathbf{n}_i} \mathbf{n}_i^T \mathbf{A}_i \mathbf{n}_i}_{= \lambda_1, \text{ when } , \mathbf{n}_i^\star = \mathbf{u}_1} \\
&= \lambda_2 + \lambda_3.
\end{aligned} \tag{128}$$

Therefore, the optimal cost is $\lambda_2(\mathbf{A}_i) + \lambda_3(\mathbf{A}_i)$ and the optimum solution is $\mathbf{n}^\star = \mathbf{u}_1(\mathbf{A}_i)$ and $\mathbf{q}^\star = \bar{\mathbf{p}}_i$.

□

B. Proof of Theorem 1

For the point collections $\mathcal{C} = \{\mathbf{p}_k \in \mathbb{R}^3 | k = 1, \cdots, n\}$, its point cluster is

$$\mathfrak{R}(\mathcal{C}) = \sum_{k=1}^{n} \begin{bmatrix} \mathbf{p}_k \\ 1 \end{bmatrix} \begin{bmatrix} \mathbf{p}_k^T & 1 \end{bmatrix} \tag{129}$$

The rigid transformation of $\mathcal{C}$ by pose $\mathbf{T} = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0}_{3\times 1} & 1 \end{bmatrix}$ is

$$\mathbf{T} \circ \mathcal{C} = \{\mathbf{R}\mathbf{p}_k + \mathbf{t} \in \mathbb{R}^3 | k = 1, \cdots, n\} \tag{130}$$

whose point cluster is

$$\begin{aligned}
\mathfrak{R}(\mathbf{T} \circ \mathcal{C}) &= \sum_{k=1}^{n} \begin{bmatrix} \mathbf{R}\mathbf{p}_k + \mathbf{t} \\ 1 \end{bmatrix} \begin{bmatrix} (\mathbf{R}\mathbf{p}_k + \mathbf{t})^T & 1 \end{bmatrix} \\
&= \sum_{k=1}^{n} \mathbf{T} \begin{bmatrix} \mathbf{p}_k \\ 1 \end{bmatrix} \begin{bmatrix} \mathbf{p}_k^T & 1 \end{bmatrix} \mathbf{T}^T = \mathbf{T} \mathfrak{R}(\mathcal{C}) \mathbf{T}^T
\end{aligned} \tag{131}$$

which yields the solution. □

C. Proof of Theorem 2

For two point collections $\mathcal{C}_1 = \{\mathbf{p}_k^1 \in \mathbb{R}^3 | k = 1, \cdots, n_1\}$ and $\mathcal{C}_2 = \{\mathbf{p}_k^2 \in \mathbb{R}^3 | k = 1, \cdots, n_2\}$ in the same reference frame, their point clusters are respectively

$$\mathfrak{R}(\mathcal{C}_l) = \sum_{k=1}^{n_l} \begin{bmatrix} \mathbf{p}_k^l \\ 1 \end{bmatrix} \begin{bmatrix} (\mathbf{p}_k^l)^T & 1 \end{bmatrix}, \quad l = 1, 2 \tag{132}$$

The merge of $\mathcal{C}_1$ and $\mathcal{C}_2$ is

$$\mathcal{C}_1 \oplus \mathcal{C}_2 = \{\mathbf{p}_k^l \in \mathbb{R}^3 | l = 1, 2; k = 1, \cdots, n_l\} \tag{133}$$

whose point cluster is

$$\begin{aligned}
\mathfrak{R}(\mathcal{C}_1 \oplus \mathcal{C}_2) &= \sum_{k=1}^{n_1} \begin{bmatrix} \mathbf{p}_k^1 \\ 1 \end{bmatrix} \begin{bmatrix} (\mathbf{p}_k^1)^T & 1 \end{bmatrix} + \sum_{k=1}^{n_2} \begin{bmatrix} \mathbf{p}_k^2 \\ 1 \end{bmatrix} \begin{bmatrix} (\mathbf{p}_k^2)^T & 1 \end{bmatrix} \\
&= \mathfrak{R}(\mathcal{C}_1) + \mathfrak{R}(\mathcal{C}_2)
\end{aligned} \tag{134}$$

which yields the solution. □

D. Proof of Theorem 3

Let $\mathbf{T}_0 = \begin{bmatrix} \mathbf{R}_0 & \mathbf{t}_0 \\ 0 & 1 \end{bmatrix}$ and $\bar{\mathbf{C}} = \mathbf{T}_0 \mathbf{C} \mathbf{T}_0^T = \begin{bmatrix} \bar{\mathbf{P}} & \bar{\mathbf{v}} \\ \bar{\mathbf{v}}^T & N \end{bmatrix}$, then

$$\bar{\mathbf{P}} = \mathbf{R}_0 \mathbf{P}_0 \mathbf{R}_0^T + \mathbf{R}_0 \mathbf{v}_0 \mathbf{t}_0^T + \mathbf{t}_0 \mathbf{v}_0^T \mathbf{R}_0^T + N \mathbf{t}_0 \mathbf{t}_0^T, \tag{135}$$

$$\bar{\mathbf{v}} = \mathbf{R}_0 \mathbf{v}_0 + N \mathbf{t}_0, \tag{136}$$

$$\mathbf{A}\left(\mathbf{T}_0 \mathbf{C} \mathbf{T}_0^T\right) = \frac{1}{N} \bar{\mathbf{P}} - \frac{1}{N^2} \bar{\mathbf{v}} \bar{\mathbf{v}}^T = \mathbf{R}_0 \mathbf{A}(\mathbf{C}) \mathbf{R}_0^T. \tag{137}$$

Since $\mathbf{A}\left(\mathbf{T}_0 \mathbf{C} \mathbf{T}_0^T\right)$ and $\mathbf{A}(\mathbf{C})$ are similar by transformation $\mathbf{R}_0$, they have the same eigenvalue. □

E. Proof of Theorem 4

Denote λ_l the l-th largest eigenvalue of $\mathbf{A}$ and $\mathbf{u}_l$ the corresponding vector, i.e., $\lambda_l \mathbf{u}_l = \mathbf{A}\mathbf{u}_l$. Since $\mathbf{A}$ is symmetric, $\mathbf{u}_l$ is an orthonormal vector. Multiplying both sides of $\lambda_l \mathbf{u}_l = \mathbf{A}\mathbf{u}_l$ by $\mathbf{u}_l^T$ leads to

$$\lambda_l = \mathbf{u}_l^T \mathbf{A} \mathbf{u}_l. \tag{138}$$

Note that in the above equation, λ_l, $\mathbf{u}_l$ and $\mathbf{A}_l$ all depend on the pose $\mathbf{T}$. To avoid any confusion, we write them as explicit functions of $\mathbf{T}$:

$$\lambda_l(\mathbf{T}) = \mathbf{u}_l^T(\mathbf{T}) \mathbf{A}(\mathbf{T}) \mathbf{u}_l(\mathbf{T}). \tag{139}$$

Parameterizing the pose $\mathbf{T}$ by $\delta\mathbf{T}$ leads to

$$\lambda_l(\delta\mathbf{T}) = \mathbf{u}_l^T(\delta\mathbf{T}) \mathbf{A}(\delta\mathbf{T}) \mathbf{u}_l(\delta\mathbf{T}). \tag{140}$$

From the first conclusion of Lemma 1 (i.e., (58)), we know that for a vector $\mathbf{x} = \begin{bmatrix} x_1 & \cdots & x_m \end{bmatrix} \in \mathbb{R}^m$ that the matrix $\mathbf{A}$ depends on, we have

$$\frac{\partial \lambda_l(\mathbf{x})}{\partial x_i} = \frac{\partial \left(\mathbf{u}_l^T(\mathbf{x}) \mathbf{A}(\mathbf{x}) \mathbf{u}_l(\mathbf{x}) \right)}{\partial x_i} = \mathbf{u}_l^T(\mathbf{x}) \frac{\partial \mathbf{A}(\mathbf{x})}{\partial x_i} \mathbf{u}_l(\mathbf{x}),$$
$$\forall i = 1, \cdots, m, x_i \in \mathbb{R} \tag{141}$$

Directly applying this result to the entire vector $\mathbf{x}$ leads to the notation of $\frac{\partial \mathbf{A}(\mathbf{x})}{\partial \mathbf{x}}$, which is a tensor. To avoid it, we fix the vector $\mathbf{u}_l(\mathbf{x})$ at its current value and lump it with the matrix $\mathbf{A}(\mathbf{x})$ within the derivative, i.e.,

$$\mathbf{u}_l^T(\mathbf{x}) \frac{\partial \mathbf{A}(\mathbf{x})}{\partial x_i} \mathbf{u}_l(\mathbf{x}) := \frac{\partial \mathbf{u}_l^T \mathbf{A}(\mathbf{x}) \mathbf{u}_l}{\partial x_i} \tag{142}$$

where on the right hand side, $\mathbf{u}_l$ is fixed (so we remove its argument $\mathbf{x}$) and the derivative is only applied on the component $\mathbf{A}(\mathbf{x})$ (so we keep its argument $\mathbf{x}$). If applying (142) to the entire vector $\mathbf{x} \in \mathbb{R}^m$, the result would be $\frac{\partial \mathbf{u}_l^T \mathbf{A}(\mathbf{x}) \mathbf{u}_l}{\partial \mathbf{x}}$, which is now a row vector of dimension m.

Since the input parameter is the poses parameterized by $\delta\mathbf{T}$, setting $\mathbf{x}$ to $\delta\mathbf{T}$ in (141) and applying the notation trick in (142) lead to

$$\frac{\partial\lambda_l(\delta\mathbf{T})}{\partial\delta\mathbf{T}} = \frac{\partial\mathbf{u}_l^T\mathbf{A}(\delta\mathbf{T})\mathbf{u}_l}{\partial\delta\mathbf{T}}. \tag{143}$$

Recalling Lemma 2 with $k = l$, we obtain:

$$\mathbf{u}_l^T\mathbf{A}(\delta\mathbf{T})\mathbf{u}_l = \frac{1}{2}\delta\mathbf{T}^T\cdot\mathbf{Q}_{ll}\cdot\delta\mathbf{T} + \mathbf{g}_{ll}\cdot\delta\mathbf{T} + r_{ll}. \tag{144}$$

where $\mathbf{g}_{ll}$ and $\mathbf{Q}_{ll}$ are defined in (69) and (70) with $k = l$, respectively.

Therefore, the first order derivative of $\lambda_l(\mathbf{T})$ w.r.t. $\mathbf{T}$ is

$$\mathbf{J}_l \triangleq \frac{\partial\lambda_l(\delta\mathbf{T})}{\partial\delta\mathbf{T}} = \frac{\partial\mathbf{u}_l^T\mathbf{A}(\delta\mathbf{T})\mathbf{u}_l}{\partial\delta\mathbf{T}} = \mathbf{g}_{ll}, \tag{145}$$

which is the result of (30) in Theorem 4.

Next, we derive the second order derivative of $\lambda_l(\mathbf{T})$ w.r.t. $\mathbf{T}$. From (141), we have,

$$\frac{\partial\lambda_l(\mathbf{x})}{\partial x_i} = \mathbf{u}_l^T(\mathbf{x})\frac{\partial\mathbf{A}(\mathbf{x})}{\partial x_i}\mathbf{u}_l(\mathbf{x}), \forall x_i\in\mathbb{R}, \tag{146}$$

Differentiating it w.r.t. the second parameter $x_j\in\mathbb{R}$ leads to

$$\begin{aligned}\frac{\partial^2\lambda_l(\mathbf{x})}{\partial x_j\partial x_i} &= \frac{\partial}{\partial x_j}\left(\mathbf{u}_l^T(\mathbf{x})\frac{\partial\mathbf{A}(\mathbf{x})}{\partial x_i}\mathbf{u}_l(\mathbf{x})\right)\\ &= \left(\frac{\partial\mathbf{u}_l(\mathbf{x})}{\partial x_j}\right)^T\left(\frac{\partial\mathbf{A}(\mathbf{x})\mathbf{u}_l}{\partial x_i}\right) + \frac{\partial}{\partial x_j}\left(\frac{\partial\mathbf{u}_l^T\mathbf{A}(\mathbf{x})\mathbf{u}_l}{\partial x_i}\right)\\ &+ \left(\frac{\partial\mathbf{A}(\mathbf{x})\mathbf{u}_l}{\partial x_i}\right)^T\left(\frac{\partial\mathbf{u}_l(\mathbf{x})}{\partial x_j}\right)\end{aligned} \tag{147}$$

Applying the above results to each elements x_i, x_j leads to

$$\begin{aligned}\frac{\partial^2\lambda_l(\mathbf{x})}{\partial\mathbf{x}^2} &= \left(\frac{\partial\mathbf{u}_l(\mathbf{x})}{\partial\mathbf{x}}\right)^T\left(\frac{\partial\mathbf{A}(\mathbf{x})\mathbf{u}_l}{\partial\mathbf{x}}\right)\\ &+ \frac{\partial}{\partial\mathbf{x}}\left(\frac{\partial\mathbf{u}_l^T\mathbf{A}(\mathbf{x})\mathbf{u}_l}{\partial\mathbf{x}}\right) + \left(\frac{\partial\mathbf{A}(\mathbf{x})\mathbf{u}_l}{\partial\mathbf{x}}\right)^T\left(\frac{\partial\mathbf{u}_l(\mathbf{x})}{\partial\mathbf{x}}\right).\end{aligned} \tag{148}$$

To compute $\frac{\partial\mathbf{u}_l(\mathbf{x})}{\partial\mathbf{x}}$, we apply the second conclusion in Lemma 1 (i.e., (59)) to all components of $\mathbf{x}$ and use the notation trick similar to (142):

$$\frac{\partial\mathbf{u}_l(\mathbf{x})}{\partial\mathbf{x}} = \sum_{k=1,k\neq l}^{3}\frac{1}{\lambda_l-\lambda_k}\mathbf{u}_k\mathbf{u}_k^T\frac{\partial\mathbf{A}(\mathbf{x})\mathbf{u}_l}{\partial\mathbf{x}} \tag{149}$$

Now, the input parameter is the pose vector parameterized by $\delta\mathbf{T}$, substituting $\mathbf{x} = \delta\mathbf{T}$ into (148) leads to

$$\begin{aligned}\frac{\partial^2\lambda_l(\delta\mathbf{T})}{\partial\delta\mathbf{T}^2} &= \underbrace{\left(\frac{\partial\mathbf{u}_l(\delta\mathbf{T})}{\partial\delta\mathbf{T}}\right)^T\left(\frac{\partial\mathbf{A}(\delta\mathbf{T})\mathbf{u}_l}{\partial\delta\mathbf{T}}\right)}_{\mathbf{D}_l^T}\\ &+ \underbrace{\frac{\partial^2\mathbf{u}_l^T\mathbf{A}(\delta\mathbf{T})\mathbf{u}_l}{\partial\delta\mathbf{T}^2}}_{\mathbf{Q}_{ll}} + \underbrace{\left(\frac{\partial\mathbf{A}(\delta\mathbf{T})\mathbf{u}_l}{\partial\delta\mathbf{T}}\right)^T\left(\frac{\partial\mathbf{u}_l(\delta\mathbf{T})}{\partial\delta\mathbf{T}}\right)}_{\mathbf{D}_l},\end{aligned} \tag{150}$$

where term $\mathbf{Q}_{ll}$ is from (144), which is further from (70) with $k = l$. To obtain the term $\mathbf{D}_l$, we substitute $\mathbf{x} = \delta\mathbf{T}$ into (149):

$$\frac{\partial\mathbf{u}_l(\delta\mathbf{T})}{\partial\delta\mathbf{T}} = \sum_{k=1,k\neq l}^{3}\frac{1}{\lambda_l-\lambda_k}\mathbf{u}_k\mathbf{u}_k^T\frac{\partial\mathbf{A}(\delta\mathbf{T})\mathbf{u}_l}{\partial\delta\mathbf{T}} \tag{151}$$

$$= \sum_{k=1,k\neq l}^{3}\frac{1}{\lambda_l-\lambda_k}\mathbf{u}_k\frac{\partial\mathbf{u}_k^T\mathbf{A}(\delta\mathbf{T})\mathbf{u}_l}{\partial\delta\mathbf{T}}. \tag{152}$$

From Lemma 2, we have

$$\frac{\partial\mathbf{u}_k^T\mathbf{A}(\delta\mathbf{T})\mathbf{u}_l}{\partial\delta\mathbf{T}} = \mathbf{g}_{kl} \tag{153}$$

with $\mathbf{g}_{kl}$ defined in (69). Hence,

$$\frac{\partial\mathbf{u}_l(\delta\mathbf{T})}{\partial\delta\mathbf{T}} = \sum_{k=1,k\neq l}^{3}\frac{1}{\lambda_l-\lambda_k}\mathbf{u}_k\mathbf{g}_{kl} \tag{154}$$

and

$$\mathbf{D}_l = \left(\frac{\partial\mathbf{A}(\delta\mathbf{T})\mathbf{u}_l}{\partial\delta\mathbf{T}}\right)^T\left(\frac{\partial\mathbf{u}_l(\delta\mathbf{T})}{\partial\delta\mathbf{T}}\right) \tag{155}$$

$$= \left(\frac{\partial\mathbf{A}(\delta\mathbf{T})\mathbf{u}_l}{\partial\delta\mathbf{T}}\right)^T\sum_{k=1,k\neq l}^{3}\frac{1}{\lambda_l-\lambda_k}\mathbf{u}_k\mathbf{g}_{kl} \tag{156}$$

$$= \sum_{k=1,k\neq l}^{3}\left(\frac{\partial\mathbf{u}_k^T\mathbf{A}(\delta\mathbf{T})\mathbf{u}_l}{\partial\delta\mathbf{T}}\right)^T\frac{1}{\lambda_l-\lambda_k}\mathbf{g}_{kl} \tag{157}$$

$$= \sum_{k=1,k\neq l}^{3}\frac{1}{\lambda_l-\lambda_k}\mathbf{g}_{kl}^T\mathbf{g}_{kl}. \tag{158}$$

Therefore, the second order derivative of $\lambda_l(\mathbf{T})$ w.r.t. $\mathbf{T}$ is

$$\begin{aligned}\mathbf{H}_l \triangleq \frac{\partial^2\lambda_l(\delta\mathbf{T})}{\partial\delta\mathbf{T}^2} &= \mathbf{D}_l^T + \mathbf{Q}_{ll} + \mathbf{D}_l\\ &= \mathbf{W}_l + \sum_{k=1,k\neq l}^{3}\frac{2}{\lambda_l-\lambda_k}\mathbf{g}_{kl}^T\mathbf{g}_{kl}.\end{aligned} \tag{159}$$

where $\mathbf{W}_l = \mathbf{Q}_{ll}$ defined in (70) with $k = l$. (159) gives the result of (31) in Theorem 4. □

F. Proof of Corollary 4.1

First, we show that

$$\mathbf{J}\delta\mathbf{T} = 0, \delta\mathbf{T}^T\mathbf{H}\delta\mathbf{T} = 0, \delta\mathbf{T} = \begin{bmatrix}\mathbf{w}\\ \vdots\\ \mathbf{w}\end{bmatrix}, \forall\mathbf{w}\in\mathbb{R}^6 \tag{160}$$

From (30) in Theorem 4, we can obtain

$$\mathbf{J}_l\delta\mathbf{T} = \mathbf{g}_{ll}\delta\mathbf{T} = \sum_{j=1}^{M_p}\mathbf{g}_{ll}^j\mathbf{w} \tag{161}$$

where $\mathbf{g}_{ll}^j$ is defined in (71) with $k = l$:

$$\mathbf{g}_{ll}^j = \frac{2}{N}\mathbf{u}_l^T\mathbf{S_P}(\mathbf{T}_j - \frac{1}{N}\mathbf{CF})\mathbf{C}_j\mathbf{T}_j^T\mathbf{V}_l^T \tag{162}$$

$$\mathbf{V}_l = \begin{bmatrix}-\lfloor\mathbf{u}_l\rfloor & \mathbf{0}_{3\times1}\\ \mathbf{0}_{3\times3} & \mathbf{u}_l\end{bmatrix} \quad \mathbf{F} = \begin{bmatrix}\mathbf{0}_{3\times3} & \mathbf{0}_{3\times1}\\ \mathbf{0}_{1\times3} & 1\end{bmatrix} \tag{163}$$

$$\mathbf{S_P} = \begin{bmatrix}\mathbf{I}_{3\times3} & \mathbf{0}_{3\times1}\end{bmatrix} \quad \mathbf{C} = \sum_{j=1}^{M_p}\mathbf{T}_j\mathbf{C}_j\mathbf{T}_j^T \tag{164}$$

Thus,

$$\sum_{j=1}^{M_p} \mathbf{g}_{ll}^j \mathbf{w} = \frac{2}{N} \sum_{j=1}^{M_p} \mathbf{u}_l^T \mathbf{S}_\mathbf{P} (\mathbf{T}_j - \frac{1}{N}\mathbf{CF})\mathbf{C}_j \mathbf{T}_j^T \mathbf{V}_l^T \mathbf{w}$$
$$= \frac{2}{N} \mathbf{u}_l^T \mathbf{S}_\mathbf{P} \left(\sum_{j=1}^{M_p} (\mathbf{T}_j \mathbf{C}_j \mathbf{T}_j^T - \frac{1}{N}\mathbf{CFC}_j \mathbf{T}_j^T) \right) \mathbf{V}_l^T \mathbf{w}.$$

Since $\sum_{j=1}^{M_p} \mathbf{T}_j \mathbf{C}_j \mathbf{T}_j^T \triangleq \mathbf{C}$ and $\sum_{j=1}^{M_p} \mathbf{FC}_j \mathbf{T}_j^T = \mathbf{FC}$ from equation (108), we have

$$\sum_{j=1}^{M_p} \mathbf{g}_{ll}^j \mathbf{w} = \frac{2}{N} \mathbf{u}_l^T \mathbf{S}_\mathbf{P} (\mathbf{C} - \frac{1}{N}\mathbf{CFC}) \mathbf{V}_l^T \mathbf{w}.$$

Partition $\mathbf{C} = \begin{bmatrix} \mathbf{P} & \mathbf{v} \\ \mathbf{v}^T & N \end{bmatrix}$ and recalling $\mathbf{A} = \frac{1}{N}\mathbf{P} - \frac{1}{N^2}\mathbf{v}\mathbf{v}^T$,

$$\mathbf{C} - \frac{1}{N}\mathbf{CFC}$$
$$= \begin{bmatrix} \mathbf{P} & \mathbf{v} \\ \mathbf{v}^T & N \end{bmatrix} - \frac{1}{N} \begin{bmatrix} \mathbf{P} & \mathbf{v} \\ \mathbf{v}^T & N \end{bmatrix} \begin{bmatrix} \mathbf{0}_{3\times3} & \mathbf{0}_{3\times1} \\ \mathbf{0}_{1\times3} & 1 \end{bmatrix} \begin{bmatrix} \mathbf{P} & \mathbf{v} \\ \mathbf{v}^T & N \end{bmatrix}$$
$$= \begin{bmatrix} \mathbf{P} & \mathbf{v} \\ \mathbf{v}^T & N \end{bmatrix} - \begin{bmatrix} \frac{1}{N}\mathbf{v}\mathbf{v}^T & \mathbf{v} \\ \mathbf{v}^T & N \end{bmatrix} = \begin{bmatrix} N\mathbf{A} & \mathbf{0}_{3\times1} \\ \mathbf{0}_{1\times3} & 0 \end{bmatrix} \tag{165}$$

Thus,

$$\sum_{j=1}^{M_p} \mathbf{g}_{ll}^j \mathbf{w} = \frac{2}{N} \mathbf{u}_l^T \mathbf{S}_\mathbf{P} \begin{bmatrix} N\mathbf{A} & \mathbf{0}_{3\times1} \\ \mathbf{0}_{1\times3} & 0 \end{bmatrix} \mathbf{V}_l^T \mathbf{w} \tag{166}$$
$$= 2 \begin{bmatrix} \mathbf{u}_l^T \mathbf{A} & \mathbf{0}_{3\times1} \end{bmatrix} \begin{bmatrix} -\lfloor \mathbf{u}_l \rfloor & \mathbf{0}_{3\times1} \\ \mathbf{0}_{3\times3} & \mathbf{u}_l \end{bmatrix}^T \mathbf{w} \tag{167}$$
$$= 2 \begin{bmatrix} \mathbf{u}_l^T \mathbf{A} \lfloor \mathbf{u}_l \rfloor & \mathbf{0}_{3\times1} \end{bmatrix} \mathbf{w} \tag{168}$$

Since $\mathbf{A}\mathbf{u}_l = \lambda_l \mathbf{u}_l$,

$$\mathbf{u}_l^T \mathbf{A} \lfloor \mathbf{u}_l \rfloor = \lambda_l \mathbf{u}_l^T \lfloor \mathbf{u}_l \rfloor = \mathbf{0}_{1\times3}. \tag{169}$$

Therefore,

$$\mathbf{J}_l \delta\mathbf{T} = \sum_{j=1}^{M_p} \mathbf{g}_{ll}^j \mathbf{w} = 0. \tag{170}$$

For the proof of $\delta\mathbf{T}^T \mathbf{H}_l \delta\mathbf{T} = 0$, from (31) in Theorem 4,

$$\delta\mathbf{T}^T \mathbf{H}_l \delta\mathbf{T} = \delta\mathbf{T}^T (\mathbf{W}_l + \sum_{k=1,k\neq l}^{3} \frac{2}{\lambda_l - \lambda_k} \mathbf{g}_{kl}^T \mathbf{g}_{kl}) \delta\mathbf{T}$$
$$= \mathbf{w}^T \sum_{i=1}^{M_p} \sum_{j=1}^{M_p} \left(\mathbf{W}_l^{ij} + \sum_{k=1,k\neq l}^{3} \frac{2}{\lambda_l - \lambda_k} (\mathbf{g}_{kl}^i)^T \mathbf{g}_{kl}^j \right) \mathbf{w} \tag{171}$$

where

$$\mathbf{W}_l^{ij} = -\frac{2}{N^2} \mathbf{V}_l \mathbf{T}_i \mathbf{C}_i \mathbf{FC}_j \mathbf{T}_j^T \mathbf{V}_l^T + \mathbb{1}_{i=j}$$
$$\cdot \left(\frac{2}{N} \mathbf{V}_l \mathbf{T}_j \mathbf{C}_j \mathbf{T}_j^T \mathbf{V}_l^T + \begin{bmatrix} \mathbf{K}_l^j & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \end{bmatrix} \right), \tag{172}$$

$$\mathbf{K}_l^j = \frac{1}{N} \lfloor \mathbf{S}_\mathbf{P} \mathbf{T}_j \mathbf{C}_j (\mathbf{T}_j - \frac{1}{N}\mathbf{CF})^T \mathbf{S}_\mathbf{P}^T \mathbf{u}_l \rfloor \lfloor \mathbf{u}_l \rfloor$$
$$+ \frac{1}{N} \lfloor \mathbf{u}_l \rfloor \lfloor \mathbf{S}_\mathbf{P} \mathbf{T}_j \mathbf{C}_j (\mathbf{T}_j - \frac{1}{N}\mathbf{CF})^T \mathbf{S}_\mathbf{P}^T \mathbf{u}_l \rfloor, \tag{173}$$

$$\mathbf{g}_{kl}^j = \frac{1}{N} \mathbf{u}_l^T \mathbf{S}_\mathbf{P} (\mathbf{T}_j - \frac{1}{N}\mathbf{CF}) \mathbf{C}_j \mathbf{T}_j^T \mathbf{V}_k^T$$
$$+ \frac{1}{N} \mathbf{u}_k^T \mathbf{S}_\mathbf{P} (\mathbf{T}_j - \frac{1}{N}\mathbf{CF}) \mathbf{C}_j \mathbf{T}_j^T \mathbf{V}_l^T, \tag{174}$$

We will divide (171) into two parts to discuss. For the first part,

$$\sum_{i=1}^{M_p} \sum_{j=1}^{M_p} \mathbf{W}_l^{ij} = -\frac{2}{N^2} \sum_{i=1}^{M_p} \sum_{j=1}^{M_p} \mathbf{V}_l \mathbf{T}_i \mathbf{C}_i \mathbf{FC}_j \mathbf{T}_j^T \mathbf{V}_l^T$$
$$+ \frac{2}{N} \sum_{j=1}^{M_p} \mathbf{V}_l \mathbf{T}_j \mathbf{C}_j \mathbf{T}_j^T \mathbf{V}_l^T + \begin{bmatrix} \sum_{j=1}^{M_p} \mathbf{K}_l^j & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \end{bmatrix} \tag{175}$$

Since $\sum_{j=1}^{M_p} \mathbf{T}_j \mathbf{C}_j \mathbf{T}_j^T \triangleq \mathbf{C}$, $\sum_{j=1}^{M_p} \mathbf{FC}_j \mathbf{T}_j^T = \mathbf{FC}$ and $\sum_{j=1}^{M_p} \mathbf{T}_j \mathbf{C}_j \mathbf{F} = \mathbf{CF}$ from (108), we have

$$\sum_{i=1}^{M_p} \sum_{j=1}^{M_p} \mathbf{T}_i \mathbf{C}_i \mathbf{FC}_j \mathbf{T}_j^T$$
$$= \sum_{i=1}^{M_p} \mathbf{T}_i \mathbf{C}_i \sum_{j=1}^{M_p} \mathbf{FC}_j \mathbf{T}_j^T = \sum_{i=1}^{M_p} \mathbf{T}_i \mathbf{C}_i \mathbf{FC} = \mathbf{CFC}, \tag{176}$$

and

$$\sum_{j=1}^{M_p} \mathbf{K}_l^j = \frac{1}{N} \sum_{j=1}^{M_p} \lfloor \mathbf{S}_\mathbf{P} \mathbf{T}_j \mathbf{C}_j (\mathbf{T}_j - \frac{1}{N}\mathbf{CF})^T \mathbf{S}_\mathbf{P}^T \mathbf{u}_l \rfloor \lfloor \mathbf{u}_l \rfloor$$
$$+ \frac{1}{N} \sum_{j=1}^{M_p} \lfloor \mathbf{u}_l \rfloor \lfloor \mathbf{S}_\mathbf{P} \mathbf{T}_j \mathbf{C}_j (\mathbf{T}_j - \frac{1}{N}\mathbf{CF})^T \mathbf{S}_\mathbf{P}^T \mathbf{u}_l \rfloor$$
$$= \frac{1}{N} \lfloor \mathbf{S}_\mathbf{P} (\mathbf{C} - \frac{1}{N}\mathbf{CFC}) \mathbf{S}_\mathbf{P}^T \mathbf{u}_l \rfloor \lfloor \mathbf{u}_l \rfloor$$
$$+ \frac{1}{N} \lfloor \mathbf{u}_l \rfloor \lfloor \mathbf{S}_\mathbf{P} (\mathbf{C} - \frac{1}{N}\mathbf{CFC}) \mathbf{S}_\mathbf{P}^T \mathbf{u}_l \rfloor. \tag{177}$$

Then, from (165) and $\mathbf{A}\mathbf{u}_l = \lambda_l \mathbf{u}_l$,

$$\sum_{j=1}^{M_p} \mathbf{K}_l^j = \lfloor \mathbf{A}\mathbf{u}_l \rfloor \lfloor \mathbf{u}_l \rfloor + \lfloor \mathbf{u}_l \rfloor \lfloor \mathbf{A}\mathbf{u}_l \rfloor = 2\lambda_l \lfloor \mathbf{u}_l \rfloor^2. \tag{178}$$

Now, substituting the results in (176) and (178) into (175):

$$\sum_{i=1}^{M_p}\sum_{j=1}^{M_p}\mathbf{W}_l^{ij}$$
$$= -\frac{2}{N^2}\mathbf{V}_l\mathbf{CFCV}_l^T + \frac{2}{N}\mathbf{V}_l\mathbf{CV}_l^T + \begin{bmatrix}2\lambda_l\lfloor\mathbf{u}_l\rfloor^2 & \mathbf{0}\\ \mathbf{0} & \mathbf{0}\end{bmatrix} \quad (179)$$
$$= \frac{2}{N}\mathbf{V}_l\underbrace{\left(\mathbf{C}-\frac{1}{N}\mathbf{CFC}\right)}_{\text{Recall (165)}}\mathbf{V}_l^T + \begin{bmatrix}2\lambda_l\lfloor\mathbf{u}_l\rfloor^2 & \mathbf{0}\\ \mathbf{0} & \mathbf{0}\end{bmatrix} \quad (180)$$
$$= 2\mathbf{V}_l\begin{bmatrix}\mathbf{A} & \mathbf{0}\\ \mathbf{0} & 0\end{bmatrix}\mathbf{V}_l^T + \begin{bmatrix}2\lambda_l\lfloor\mathbf{u}_l\rfloor^2 & \mathbf{0}\\ \mathbf{0} & \mathbf{0}\end{bmatrix}$$
$$= 2\begin{bmatrix}\lambda_l\lfloor\mathbf{u}_l\rfloor^2 - \lfloor\mathbf{u}_l\rfloor\mathbf{A}\lfloor\mathbf{u}_l\rfloor & \mathbf{0}\\ \mathbf{0} & \mathbf{0}\end{bmatrix}. \quad (181)$$

For the second part in (171),

$$\sum_{i=1}^{M_p}\sum_{j=1}^{M_p}\left(\sum_{k=1,k\neq l}^{3}\frac{2}{\lambda_l-\lambda_k}(\mathbf{g}_{kl}^i)^T\mathbf{g}_{kl}^j\right)$$
$$= \sum_{k=1,k\neq l}^{3}\frac{2}{\lambda_l-\lambda_k}\left(\sum_{i=1}^{M_p}\mathbf{g}_{kl}^i\right)^T\left(\sum_{j=1}^{M_p}\mathbf{g}_{kl}^j\right) \quad (182)$$

where

$$\sum_{j=1}^{M_p}\mathbf{g}_{kl}^j = \frac{1}{N}\mathbf{u}_l^T\mathbf{S}_\mathbf{P}\sum_{j=1}^{M_p}(\mathbf{T}_j\mathbf{C}_j\mathbf{T}_j^T - \frac{1}{N}\mathbf{CFC}_j\mathbf{T}_j^T)\mathbf{V}_k^T$$
$$+\frac{1}{N}\mathbf{u}_k^T\mathbf{S}_\mathbf{P}\sum_{j=1}^{M_p}(\mathbf{T}_j\mathbf{C}_j\mathbf{T}_j^T - \frac{1}{N}\mathbf{CFC}_j\mathbf{T}_j^T)\mathbf{V}_l^T. \quad (183)$$

Again, since $\sum_{j=1}^{M_p}\mathbf{T}_j\mathbf{C}_j\mathbf{T}_j^T \triangleq \mathbf{C}$ and $\sum_{j=1}^{M_p}\mathbf{FC}_j\mathbf{T}_j^T = \mathbf{FC}$ from (108), we have

$$\sum_{j=1}^{M_p}\mathbf{g}_{kl}^j = \frac{1}{N}\mathbf{u}_l^T\mathbf{S}_\mathbf{P}(\mathbf{C}-\frac{1}{N}\mathbf{CFC})\mathbf{V}_k^T$$
$$+\frac{1}{N}\mathbf{u}_k^T\mathbf{S}_\mathbf{P}(\mathbf{C}-\frac{1}{N}\mathbf{CFC})\mathbf{V}_l^T \quad (184)$$

Further, from (165), $\mathbf{A}\mathbf{u}_l = \lambda_l\mathbf{u}_l$, and $\mathbf{A}\mathbf{u}_k = \lambda_k\mathbf{u}_k$, we have

$$\sum_{j=1}^{M_p}\mathbf{g}_{kl}^j = \mathbf{u}_l^T\begin{bmatrix}\mathbf{A} & \mathbf{0}_{3\times1}\end{bmatrix}\mathbf{V}_k^T + \mathbf{u}_k^T\begin{bmatrix}\mathbf{A} & \mathbf{0}_{3\times1}\end{bmatrix}\mathbf{V}_l^T$$
$$= \begin{bmatrix}\mathbf{u}_l^T\mathbf{A}\lfloor\mathbf{u}_k\rfloor & \mathbf{0}_{3\times1}\end{bmatrix} + \begin{bmatrix}\mathbf{u}_k^T\mathbf{A}\lfloor\mathbf{u}_l\rfloor & \mathbf{0}_{3\times1}\end{bmatrix}$$
$$= \begin{bmatrix}\lambda_l\mathbf{u}_l^T\lfloor\mathbf{u}_k\rfloor + \lambda_k\mathbf{u}_k^T\lfloor\mathbf{u}_l\rfloor & \mathbf{0}_{3\times1}\end{bmatrix} \quad (185)$$

Therefore,

$$\sum_{k=1,k\neq l}^{3}\frac{2}{\lambda_l-\lambda_k}\left(\sum_{i=1}^{M_p}\mathbf{g}_{kl}^i\right)^T\left(\sum_{j=1}^{M_p}\mathbf{g}_{kl}^j\right)$$
$$= \sum_{k=1,k\neq l}^{3}\frac{2}{\lambda_l-\lambda_k}\begin{bmatrix}\mathbf{r}_{kl} & \mathbf{0}_{3\times1}\\ \mathbf{0}_{1\times3} & 0\end{bmatrix} \quad (186)$$

where due to $\lfloor\mathbf{u}_k\rfloor\mathbf{u}_l = -\lfloor\mathbf{u}_l\rfloor\mathbf{u}_k$ and $\mathbf{u}_k^T\lfloor\mathbf{u}_l\rfloor = -\mathbf{u}_l^T\lfloor\mathbf{u}_k\rfloor$

$$\mathbf{r}_{kl} = \left(\lambda_l\mathbf{u}_l^T\lfloor\mathbf{u}_k\rfloor + \lambda_k\mathbf{u}_k^T\lfloor\mathbf{u}_l\rfloor\right)^T\left(\lambda_l\mathbf{u}_l^T\lfloor\mathbf{u}_k\rfloor + \lambda_k\mathbf{u}_k^T\lfloor\mathbf{u}_l\rfloor\right)$$
$$= -\lambda_l^2\lfloor\mathbf{u}_k\rfloor\mathbf{u}_l\mathbf{u}_l^T\lfloor\mathbf{u}_k\rfloor - \lambda_l\lambda_k\lfloor\mathbf{u}_k\rfloor\mathbf{u}_l\mathbf{u}_k^T\lfloor\mathbf{u}_l\rfloor$$
$$-\lambda_k^2\lfloor\mathbf{u}_l\rfloor\mathbf{u}_k\mathbf{u}_k^T\lfloor\mathbf{u}_l\rfloor - \lambda_l\lambda_k\lfloor\mathbf{u}_l\rfloor\mathbf{u}_k\mathbf{u}_l^T\lfloor\mathbf{u}_k\rfloor$$
$$= -(\lambda_l-\lambda_k)^2\lfloor\mathbf{u}_l\rfloor\mathbf{u}_k\mathbf{u}_k^T\lfloor\mathbf{u}_l\rfloor. \quad (187)$$

Thus,

$$\sum_{k=1,k\neq l}^{3}\frac{2}{\lambda_l-\lambda_k}\left(\sum_{i=1}^{M_p}\mathbf{g}_{kl}^i\right)^T\left(\sum_{j=1}^{M_p}\mathbf{g}_{kl}^j\right)$$
$$= 2\sum_{k=1,k\neq l}^{3}\begin{bmatrix}(\lambda_k-\lambda_l)\lfloor\mathbf{u}_l\rfloor\mathbf{u}_k\mathbf{u}_k^T\lfloor\mathbf{u}_l\rfloor & \mathbf{0}\\ \mathbf{0} & 0\end{bmatrix} \quad (188)$$

Now, from (181) and (188), the equation (171) is turned into

$$\delta\mathbf{T}^T\mathbf{H}_l\delta\mathbf{T} = \mathbf{w}^T\begin{bmatrix}2\mathbf{L}_l & \mathbf{0}_{3\times1}\\ \mathbf{0}_{1\times3} & 0\end{bmatrix}\mathbf{w}, \quad (189)$$

where

$$\mathbf{L}_l = \lambda_l\lfloor\mathbf{u}_l\rfloor^2 - \lfloor\mathbf{u}_l\rfloor\mathbf{A}\lfloor\mathbf{u}_l\rfloor + \sum_{k=1,k\neq l}^{3}(\lambda_k-\lambda_l)\lfloor\mathbf{u}_l\rfloor\mathbf{u}_k\mathbf{u}_k^T\lfloor\mathbf{u}_l\rfloor$$
$$= \lambda_l\lfloor\mathbf{u}_l\rfloor^2 - \lfloor\mathbf{u}_l\rfloor\mathbf{A}\lfloor\mathbf{u}_l\rfloor$$
$$+\lfloor\mathbf{u}_l\rfloor\left(\sum_{k=1,k\neq l}^{3}\lambda_k\mathbf{u}_k\mathbf{u}_k^T\right)\lfloor\mathbf{u}_l\rfloor - \lambda_l\lfloor\mathbf{u}_l\rfloor\left(\sum_{k=1,k\neq l}^{3}\mathbf{u}_k\mathbf{u}_k^T\right)\lfloor\mathbf{u}_l\rfloor. \quad (190)$$

Since $\mathbf{u}_k$ ($k = 1, 2, 3$) is the eigenvector (with eigenvalue λ_k) of matrix $\mathbf{A}$, which is symmetric, we have the following two conditions from the singular value decomposition of $\mathbf{A}$:

$$\mathbf{A} = \sum_{k=1}^{3}\lambda_k\mathbf{u}_k\mathbf{u}_k^T, \quad \mathbf{I} = \sum_{k=1}^{3}\mathbf{u}_k\mathbf{u}_k^T, \quad (191)$$

which imply

$$\sum_{k=1,k\neq l}^{3}\lambda_k\mathbf{u}_k\mathbf{u}_k^T = \mathbf{A} - \lambda_l\mathbf{u}_l\mathbf{u}_l^T, \quad \sum_{k=1,k\neq l}^{3}\mathbf{u}_k\mathbf{u}_k^T = \mathbf{I} - \mathbf{u}_l\mathbf{u}_l^T. \quad (192)$$

Substituting the above results into $\mathbf{L}_l$ in (190):

$$\mathbf{L}_l = \lambda_l\lfloor\mathbf{u}_l\rfloor^2 - \lfloor\mathbf{u}_l\rfloor\mathbf{A}\lfloor\mathbf{u}_l\rfloor$$
$$+\lfloor\mathbf{u}_l\rfloor\left(\mathbf{A}-\lambda_l\mathbf{u}_l\mathbf{u}_l^T\right)\lfloor\mathbf{u}_l\rfloor - \lambda_l\lfloor\mathbf{u}_l\rfloor(\mathbf{I}-\mathbf{u}_l\mathbf{u}_l^T)\lfloor\mathbf{u}_l\rfloor$$
$$= \lambda_l\lfloor\mathbf{u}_l\rfloor^2 - \lfloor\mathbf{u}_l\rfloor\mathbf{A}\lfloor\mathbf{u}_l\rfloor + \lfloor\mathbf{u}_l\rfloor\mathbf{A}\lfloor\mathbf{u}_l\rfloor - \lambda_l\lfloor\mathbf{u}_l\rfloor^2 = \mathbf{0}. \quad (193)$$

As a result,

$$\delta\mathbf{T}^T\mathbf{H}\delta\mathbf{T} = \mathbf{w}^T\begin{bmatrix}2\mathbf{L}_l & \mathbf{0}_{3\times1}\\ \mathbf{0}_{1\times3} & 0\end{bmatrix}\mathbf{w} = 0. \quad (194)$$

Finally, if $\mathbf{C}_j = \mathbf{0}$, we have $\mathbf{g}_{kl}^j = \mathbf{0}_{1\times6}$ from (76) of Lemma 2, hence $\mathbf{J}^j = \mathbf{g}_{ll}^j = \mathbf{0}$, the result (40) of Corollary 4.1; If $\mathbf{C}_i = \mathbf{0}$ or $\mathbf{C}_j = \mathbf{0}$, we have $\mathbf{Q}_{kl}^{ij} = \mathbf{0}_{6\times6}$ from (77) of Lemma 2, hence $\mathbf{W}_l^{ij} = \mathbf{Q}_{ll}^{ij} = \mathbf{0}_{6\times6}$ and $\mathbf{H}_l^{ij} = \mathbf{W}_l^{ij} + \sum_{k=1,k\neq l}^{3}\frac{2}{\lambda_l-\lambda_k}(\mathbf{g}_{kl}^j)^T\mathbf{g}_{kl}^j = \mathbf{0}$, the result (41) of Corollary 4.1. □

G. Derivation of pose covariance

The quantity $\delta \mathbf{C}_{f_{ij}}$ can be obtained by substituting (44) into the definition of $\mathbf{C}^{\text{gt}}_{f_{ij}}$ in (45) and retaining only the first order items:

$$\delta \mathbf{C}_{f_{ij}} = \begin{bmatrix} \delta \mathbf{P}_{f_{ij}} & \delta \mathbf{v}_{f_{ij}} \\ \delta \mathbf{v}_{f_{ij}}^T & 0 \end{bmatrix}, \quad \text{where} \tag{195}$$

$$\delta \mathbf{P}_{f_{ij}} = \sum_{k=1}^{N_{ij}} (\mathbf{p}_{f_{ijk}} \delta \mathbf{p}_{f_{ijk}}^T + \delta \mathbf{p}_{f_{ijk}} \mathbf{p}_{f_{ijk}}^T), \tag{196}$$

$$\delta \mathbf{v}_{f_{ij}} = \sum_{k=1}^{N_{ij}} \delta \mathbf{p}_{f_{ijk}}. \tag{197}$$

To derive $\frac{\partial \mathbf{J}^T(\mathbf{T}^\star, \mathbf{C}_f)}{\partial \mathbf{C}_f} \delta \mathbf{C}_f$ without involving any tensor, we parameterize the matrix $\mathbf{C}_{f_{ij}}$ by a column vector $\mathbf{c}_{f_{ij}}$, which consists of the independent elements in $\mathbf{C}_{f_{ij}}$:

$$\begin{aligned} \mathbf{c}_{f_{ij}} = \text{vec}(\mathbf{C}_{f_{ij}}) \triangleq \big[& \mathbf{e}_1^T \mathbf{C}_{f_{ij}} \mathbf{e}_1 \quad \mathbf{e}_1^T \mathbf{C}_{f_{ij}} \mathbf{e}_2 \quad \mathbf{e}_1^T \mathbf{C}_{f_{ij}} \mathbf{e}_3 \\ & \mathbf{e}_2^T \mathbf{C}_{f_{ij}} \mathbf{e}_2 \quad \mathbf{e}_2^T \mathbf{C}_{f_{ij}} \mathbf{e}_3 \quad \mathbf{e}_3^T \mathbf{C}_{f_{ij}} \mathbf{e}_3 \\ & \mathbf{e}_1^T \mathbf{C}_{f_{ij}} \mathbf{e}_4 \quad \mathbf{e}_2^T \mathbf{C}_{f_{ij}} \mathbf{e}_4 \quad \mathbf{e}_3^T \mathbf{C}_{f_{ij}} \mathbf{e}_4 \big]^T \in \mathbb{R}^9 \end{aligned} \tag{198}$$

where $\text{vec}(\cdot) : \mathbb{S}^{4\times 4} \mapsto \mathbb{R}^9$ maps a symmetric matrix to its column vector representation, $\mathbf{e}_l \in \mathbb{R}^4$ $(l \in \{1, 2, 3, 4\})$ is a vector with all zero elements except for the l-th element being one. Note that the constant N in the 4-th row, 4-th column of $\mathbf{C}_{f_{ij}}$ is not contained in $\mathbf{c}_{f_{ij}}$ since it is a constant number independent of the noise. Correspondingly, noises in $\mathbf{C}_{f_{ij}}$ becomes the noise of $\mathbf{c}_{f_{ij}}$ as below:

$$\begin{aligned} \delta \mathbf{c}_{f_{ij}} = & \big[\mathbf{e}_1^T \delta \mathbf{C}_{f_{ij}} \mathbf{e}_1 \quad \mathbf{e}_1^T \delta \mathbf{C}_{f_{ij}} \mathbf{e}_2 \quad \mathbf{e}_1^T \delta \mathbf{C}_{f_{ij}} \mathbf{e}_3 \\ & \mathbf{e}_2^T \delta \mathbf{C}_{f_{ij}} \mathbf{e}_2 \quad \mathbf{e}_2^T \delta \mathbf{C}_{f_{ij}} \mathbf{e}_3 \quad \mathbf{e}_3^T \delta \mathbf{C}_{f_{ij}} \mathbf{e}_3 \\ & \mathbf{e}_1^T \delta \mathbf{C}_{f_{ij}} \mathbf{e}_4 \quad \mathbf{e}_2^T \delta \mathbf{C}_{f_{ij}} \mathbf{e}_4 \quad \mathbf{e}_3^T \delta \mathbf{C}_{f_{ij}} \mathbf{e}_4 \big]^T \end{aligned} \tag{199}$$

$$= \sum_{k=1}^{N_{ij}} \begin{bmatrix} 2\mathbf{p}_{f_{ijk}}^T \mathbf{S}_\text{P} \mathbf{E}_{11} \mathbf{S}_\text{P}^T \delta \mathbf{p}_{f_{ijk}} \\ \mathbf{p}_{f_{ijk}}^T \mathbf{S}_\text{P} \mathbf{E}_{12} \mathbf{S}_\text{P}^T \delta \mathbf{p}_{f_{ijk}} \\ \mathbf{p}_{f_{ijk}}^T \mathbf{S}_\text{P} \mathbf{E}_{13} \mathbf{S}_\text{P}^T \delta \mathbf{p}_{f_{ijk}} \\ 2\mathbf{p}_{f_{ijk}}^T \mathbf{S}_\text{P} \mathbf{E}_{22} \mathbf{S}_\text{P}^T \delta \mathbf{p}_{f_{ijk}} \\ \mathbf{p}_{f_{ijk}}^T \mathbf{S}_\text{P} \mathbf{E}_{23} \mathbf{S}_\text{P}^T \delta \mathbf{p}_{f_{ijk}} \\ 2\mathbf{p}_{f_{ijk}}^T \mathbf{S}_\text{P} \mathbf{E}_{33} \mathbf{S}_\text{P}^T \delta \mathbf{p}_{f_{ijk}} \\ \delta \mathbf{p}_{f_{ijk}} \end{bmatrix} = \sum_{k=1}^{N_{ij}} \mathbf{B}_{f_{ijk}} \delta \mathbf{p}_{f_{ijk}}, \tag{200}$$

where $\mathbf{S}_\text{P} = [\mathbf{I}_{3\times 3}, 0_{3\times 1}]$, $\mathbf{E}_{kl} = \mathbf{e}_k \mathbf{e}_l^T + \mathbf{e}_l \mathbf{e}_k^T \in \mathbb{S}^{4\times 4}, \ k, l \in \{1, 2, 3, 4\}$, and

$$\mathbf{B}_{f_{ijk}} = \begin{bmatrix} 2\mathbf{p}_{f_{ijk}}^T \mathbf{S}_\text{P} \mathbf{E}_{11} \mathbf{S}_\text{P}^T \\ \mathbf{p}_{f_{ijk}}^T \mathbf{S}_\text{P} \mathbf{E}_{12} \mathbf{S}_\text{P}^T \\ \mathbf{p}_{f_{ijk}}^T \mathbf{S}_\text{P} \mathbf{E}_{13} \mathbf{S}_\text{P}^T \\ 2\mathbf{p}_{f_{ijk}}^T \mathbf{S}_\text{P} \mathbf{E}_{22} \mathbf{S}_\text{P}^T \\ \mathbf{p}_{f_{ijk}}^T \mathbf{S}_\text{P} \mathbf{E}_{23} \mathbf{S}_\text{P}^T \\ 2\mathbf{p}_{f_{ijk}}^T \mathbf{S}_\text{P} \mathbf{E}_{33} \mathbf{S}_\text{P}^T \\ \mathbf{I}_{3\times 3} \end{bmatrix} \in \mathbb{R}^{9\times 3}. \tag{201}$$

With the column representation of each $\mathbf{C}_{f_{ij}}$ contained in $\mathbf{C}_f$, $\frac{\partial \mathbf{J}^T(\mathbf{T}^\star, \mathbf{C}_f)}{\partial \mathbf{C}_f} \delta \mathbf{C}_f$ can now be computed as

$$\frac{\partial \mathbf{J}^T(\mathbf{T}^\star, \mathbf{C}_f)}{\partial \mathbf{C}_f} \delta \mathbf{C}_f = \sum_{i=1}^{M_f} \sum_{j=1}^{M_p} \frac{\partial \mathbf{J}^T(\mathbf{T}^\star, \mathbf{C}_{f_{ij}})}{\partial \mathbf{c}_{f_{ij}}} \delta \mathbf{c}_{f_{ij}}. \tag{202}$$

To derive the quantity $\frac{\partial \mathbf{J}^T(\mathbf{T}^\star, \mathbf{C}_{f_{ij}})}{\partial \mathbf{c}_{f_{ij}}}$, we give two lemmas which are useful for subsequent derivations.

Lemma 3. *For $\mathbf{w} \in \mathbb{R}^4, \mathbf{C} \in \mathbb{S}^{4\times 4}$ and its vector form $\mathbf{c} = \text{vec}(\mathbf{C})$, we have*

$$\frac{\partial \mathbf{C}\mathbf{w}}{\partial \mathbf{c}} = \mathbf{g}_1(\mathbf{w}) \in \mathbb{R}^{4\times 9},$$

where

$$\begin{aligned} \mathbf{g}_1(\mathbf{w}) = \big[& \mathbf{E}_{11}\mathbf{w} \quad \mathbf{E}_{12}\mathbf{w} \quad \mathbf{E}_{13}\mathbf{w} \quad \mathbf{E}_{22}\mathbf{w} \quad \mathbf{E}_{23}\mathbf{w} \\ & \mathbf{E}_{33}\mathbf{w} \quad \mathbf{E}_{14}\mathbf{w} \quad \mathbf{E}_{24}\mathbf{w} \quad \mathbf{E}_{34}\mathbf{w} \big], \end{aligned}$$

where $\mathbf{E}_{kl} \in \mathbb{S}^{4\times 4}, \ k, l \in \{1, 2, 3, 4\}$, is

$$\mathbf{E}_{kl} = \begin{cases} \mathbf{e}_k \mathbf{e}_l^T + \mathbf{e}_l \mathbf{e}_k^T & (k \neq l) \\ \mathbf{e}_l \mathbf{e}_l^T & (k = l) \end{cases} \tag{203}$$

where $\mathbf{e}_l \in \mathbb{R}^4$ $(l \in \{1, 2, 3, 4\})$ is a vector with all zero elements except for the l-th element being one.

Proof. For the k-th row, l-th column element of $\mathbf{C}$, denoted by $\mathbf{C}_{k,l}$, we have

$$\frac{\partial \mathbf{C}\mathbf{w}}{\partial \mathbf{C}_{k,l}} = \mathbf{E}_{kl}\mathbf{w} \in \mathbb{R}^4.$$

Hence,

$$\begin{aligned} \frac{\partial \mathbf{C}\mathbf{w}}{\partial \mathbf{c}} &= \Big[\tfrac{\partial \mathbf{C}\mathbf{w}}{\partial \mathbf{C}_{1,1}} \quad \tfrac{\partial \mathbf{C}\mathbf{w}}{\partial \mathbf{C}_{1,2}} \quad \tfrac{\partial \mathbf{C}\mathbf{w}}{\partial \mathbf{C}_{1,3}} \quad \tfrac{\partial \mathbf{C}\mathbf{w}}{\partial \mathbf{C}_{2,2}} \quad \tfrac{\partial \mathbf{C}\mathbf{w}}{\partial \mathbf{C}_{2,3}} \\ &\qquad \tfrac{\partial \mathbf{C}\mathbf{w}}{\partial \mathbf{C}_{3,3}} \quad \tfrac{\partial \mathbf{C}\mathbf{w}}{\partial \mathbf{C}_{1,4}} \quad \tfrac{\partial \mathbf{C}\mathbf{w}}{\partial \mathbf{C}_{2,4}} \quad \tfrac{\partial \mathbf{C}\mathbf{w}}{\partial \mathbf{C}_{3,4}} \Big] \\ &= \big[\mathbf{E}_{11}\mathbf{w} \quad \mathbf{E}_{12}\mathbf{w} \quad \mathbf{E}_{13}\mathbf{w} \quad \mathbf{E}_{22}\mathbf{w} \quad \mathbf{E}_{23}\mathbf{w} \\ &\qquad \mathbf{E}_{33}\mathbf{w} \quad \mathbf{E}_{14}\mathbf{w} \quad \mathbf{E}_{24}\mathbf{w} \quad \mathbf{E}_{34}\mathbf{w} \big]. \end{aligned}$$

□

Lemma 4. *For $\mathbf{w} \in \mathbb{R}^4, \mathbf{u}_l \in \mathbb{R}^3$ and*

$$\mathbf{V}_l = \begin{bmatrix} -\lfloor \mathbf{u}_l \rfloor & \mathbf{0}_{3\times 1} \\ \mathbf{0}_{3\times 3} & \mathbf{u}_l \end{bmatrix} \in \mathbb{R}^{6\times 4}$$

we have

$$\frac{\partial \mathbf{V}_l \mathbf{w}}{\partial \mathbf{u}_l} = \mathbf{g}_2(\mathbf{w}) = \begin{bmatrix} \lfloor \mathbf{w}_{1:3} \rfloor \\ w_4 \mathbf{I}_{3\times 3} \end{bmatrix} \in \mathbb{R}^{6\times 3}.$$

where $\mathbf{w}_{1:3}$ represents the first three elements in $\mathbf{w}$ and w_4 is the 4-th element of $\mathbf{w}$.

Proof.

$$\mathbf{V}_l \mathbf{w} = \begin{bmatrix} -\lfloor \mathbf{u}_l \rfloor & \mathbf{0} \\ \mathbf{0} & \mathbf{u}_l \end{bmatrix} \begin{bmatrix} \mathbf{w}_{1:3} \\ w_4 \end{bmatrix} = \begin{bmatrix} \lfloor \mathbf{w}_{1:3} \rfloor \mathbf{u}_l \\ w_4 \mathbf{u}_l \end{bmatrix} = \begin{bmatrix} \lfloor \mathbf{w}_{1:3} \rfloor \\ w_4 \mathbf{I}_{3\times 3} \end{bmatrix} \mathbf{u}_l$$

Thus,

$$\frac{\partial \mathbf{V}_l \mathbf{w}}{\partial \mathbf{u}_l} = \begin{bmatrix} \lfloor \mathbf{w}_{1:3} \rfloor \\ w_4 \mathbf{I}_{3\times 3} \end{bmatrix} = \mathbf{g}_2(\mathbf{w})$$

□

With these two lemmas, next we will continue to derive $\frac{\partial \mathbf{J}^T(\mathbf{T}^\star, \mathbf{C}_{f_{ij}})}{\partial \mathbf{c}_{f_{ij}}}$ in (202). Theorem 4 gives the Jacobian for one cost item while the required Jacobian consists of items from all features, to distinguish the different cost item, we add a subscript ν to (30) to denote the ν-th item and replace $\mathbf{C}_j$

with the actual point cluster notation $\mathbf{C}_{f_{\nu j}}$ corresponding to the ν-th item, leading to:

$$\mathbf{J} = \sum_{\nu=1}^{M_f} \mathbf{J}_\nu \tag{204}$$

$$\mathbf{J}_\nu = \begin{bmatrix} \cdots & \mathbf{J}_\nu^{\mathrm{p}} & \cdots \end{bmatrix} \in \mathbb{R}^{1\times 6M_p} \tag{205}$$

where $\mathbf{J}_\nu^{\mathrm{p}} \in \mathbb{R}^{1\times 6}$ is the p-th column block of $\mathbf{J}_\nu$ as shown in (32) of Theorem 4 (with $j=\mathrm{p}$):

$$\mathbf{J}_\nu^{\mathrm{p}} = \frac{2}{N_\nu}\mathbf{u}_{\nu l}^T \mathbf{S_P}\left(\mathbf{T}_{\mathrm{p}} - \frac{1}{N_\nu}\mathbf{C}_\nu \mathbf{F}\right)\mathbf{C}_{f_{\nu\mathrm{p}}}\mathbf{T}_{\mathrm{p}}^T \mathbf{V}_{\nu l}^T \in \mathbb{R}^{1\times 6}.$$

where $\mathbf{u}_{\nu l}$ is the eigenvector associated to the l-th largest eigenvalue of the covariance matrix of the ν-th feature (or cost item), N_ν is the total number of points of the ν-th feature, $\mathbf{C}_\nu$ is the aggregation of all point clusters of the ν-th feature, and $\mathbf{C}_{f_{\nu\mathrm{p}}}$ is the point cluster contributed by the p-th pose to the ν-th feature.

The total Jacobian is hence

$$\mathbf{J} = \sum_{\nu=1}^{M_f} \mathbf{J}_\nu = \begin{bmatrix} \cdots & \mathbf{J}^{\mathrm{p}} & \cdots \end{bmatrix} \in \mathbb{R}^{1\times 6M_p}, \tag{206}$$

$$\begin{aligned} \mathbf{J}^{\mathrm{p}} &= \sum_{\nu=1}^{M_f} \mathbf{J}_\nu^{\mathrm{p}} \in \mathbb{R}^{1\times 6}, \\ &= \sum_{\nu=1}^{M_f}\left(\frac{2}{N_\nu}\mathbf{u}_{\nu l}^T \mathbf{S_P}\left(\mathbf{T}_{\mathrm{p}} - \frac{1}{N_\nu}\mathbf{C}_\nu \mathbf{F}\right)\mathbf{C}_{f_{\nu\mathrm{p}}}\mathbf{T}_{\mathrm{p}}^T \mathbf{V}_{\nu l}^T\right). \end{aligned} \tag{207}$$

Next, we calculate the partial derivative $\frac{\partial (\mathbf{J}^{\mathrm{p}})^T}{\partial \mathbf{c}_{f_{ij}}}$. Note that in the summation of (207), only the i-th summation term (i.e., $\nu=i$) is related to $\mathbf{C}_{f_{ij}}$ (hence $\mathbf{c}_{f_{ij}}$), hence,

$$\frac{\partial (\mathbf{J}^{\mathrm{p}})^T}{\partial \mathbf{c}_{f_{ij}}} = \frac{\partial}{\partial \mathbf{c}_{f_{ij}}}\left(\frac{2}{N_i}\mathbf{V}_{il}\mathbf{T}_{\mathrm{p}}\mathbf{C}_{f_{i\mathrm{p}}}(\mathbf{T}_{\mathrm{p}}^T - \frac{1}{N_i}\mathbf{F}\mathbf{C}_i)\mathbf{S}_{\mathbf{P}}^T \mathbf{u}_{il}\right). \tag{208}$$

Since $\mathbf{C}_i = \sum_{\nu=1}^{M_p} \mathbf{T}_\nu \mathbf{C}_{f_{i\nu}} \mathbf{T}_\nu^T$, $\mathbf{u}_{il}$ is the eigenvector associated to the l-th largest eigenvalue of matrix $\mathbf{A}(\mathbf{C}_i)$, and $\mathbf{V}_{il} = \begin{bmatrix} -\lfloor \mathbf{u}_{il} \rfloor & \mathbf{0}_{3\times 1} \\ \mathbf{0}_{3\times 3} & \mathbf{u}_{il} \end{bmatrix}$, the derivative with respect to $\mathbf{c}_{f_{ij}}$ on the right hand side of (208) consists of four terms respectively from $\mathbf{V}_{il}$, $\mathbf{C}_{f_{i\mathrm{p}}}$ (only when $\mathrm{p}=j$), $\mathbf{C}_i$, and $\mathbf{u}_{il}$. Combining with Lemma 3 and Lemma 4, we have

$$\begin{aligned} \mathbf{L}_{ij}^{\mathrm{p}} &\triangleq \frac{\partial (\mathbf{J}^{\mathrm{p}})^T}{\partial \mathbf{c}_{f_{ij}}} \in \mathbb{R}^{6\times 9} \qquad (209) \\ &= \frac{2}{N_i}\Big(\mathbf{g}_2(\mathbf{T}_{\mathrm{p}}\mathbf{C}_{f_{i\mathrm{p}}}(\mathbf{T}_{\mathrm{p}}^T - \frac{1}{N_i}\mathbf{F}\mathbf{C}_i)\mathbf{S}_{\mathbf{P}}^T\mathbf{u}_{il}) \\ &+ \mathbf{V}_{il}\mathbf{T}_{\mathrm{p}}\mathbf{C}_{f_{i\mathrm{p}}}(\mathbf{T}_{\mathrm{p}}^T - \frac{1}{N_i}\mathbf{F}\mathbf{C}_i)\mathbf{S}_{\mathbf{P}}^T\Big)\frac{\partial \mathbf{u}_{il}}{\partial \mathbf{c}_{f_{ij}}} \\ &- \frac{2}{N_i^2}\mathbf{V}_{il}\mathbf{T}_{\mathrm{p}}\mathbf{C}_{f_{i\mathrm{p}}}\mathbf{F}\mathbf{T}_j\mathbf{g}_1(\mathbf{T}_j^T\mathbf{S}_{\mathbf{P}}^T\mathbf{u}_{il}) \\ &+ \frac{2}{N_i}\mathbb{1}_{\mathrm{p}=j}\cdot\left(\mathbf{V}_{il}\mathbf{T}_{\mathrm{p}}\mathbf{g}_1\Big((\mathbf{T}_{\mathrm{p}}^T - \frac{1}{N_i}\mathbf{F}\mathbf{C}_i)\mathbf{S}_{\mathbf{P}}^T\mathbf{u}_{il}\Big)\right) \qquad (210) \end{aligned}$$

Only the term $\frac{\partial \mathbf{u}_{il}}{\partial \mathbf{c}_{f_{ij}}}$ is unknown in this formula. To compute it, we apply the second conclusion in Lemma 1 (i.e., (59)) to all components of $\mathbf{c}_{f_{ij}}$ and use the notation trick similar to (142):

$$\frac{\partial \mathbf{u}_{il}}{\partial \mathbf{c}_{f_{ij}}} = \sum_{k=1,k\neq l}^{3}\frac{1}{\lambda_{il}-\lambda_{ik}}\mathbf{u}_{ik}\frac{\partial\left(\mathbf{u}_{ik}^T\mathbf{A}(\mathbf{C}_i)\mathbf{u}_{il}\right)}{\partial \mathbf{c}_{f_{ij}}} \tag{211}$$

From (87) of Lemma 2,

$$\mathbf{u}_{ik}^T\mathbf{A}(\mathbf{C}_i)\mathbf{u}_{il} = \frac{1}{N_i}\mathbf{u}_{ik}^T\mathbf{S_P}\left(\sum_{\mu=1}^{M_p}\mathbf{T}_\mu\mathbf{C}_{f_{i\mu}}\mathbf{T}_\mu^T - \frac{1}{N_i}\sum_{\mu=1}^{M_p}\sum_{\nu=1}^{M_p}\mathbf{T}_\mu\mathbf{C}_{f_{i\mu}}\mathbf{F}\mathbf{C}_{f_{i\nu}}\mathbf{T}_\nu^T\right)\mathbf{S}_{\mathbf{P}}^T\mathbf{u}_{il}$$

Denote

$$\begin{aligned} \mathbf{G}_{kl}^{ij} &\triangleq \frac{\partial \mathbf{u}_{ik}^T\mathbf{A}(\mathbf{C}_{f_i})\mathbf{u}_{il}}{\partial \mathbf{c}_{f_{ij}}} = \frac{1}{N_i}\mathbf{u}_{ik}^T\mathbf{S_P}\mathbf{T}_j\mathbf{g}_1(\mathbf{T}_j^T\mathbf{S}_{\mathbf{P}}^T\mathbf{u}_{il}) \\ &- \frac{1}{N_i^2}\mathbf{u}_{ik}^T\mathbf{S_P}\sum_{\mu=1}^{M_p}\mathbf{T}_\mu\mathbf{C}_{f_{i\mu}}\mathbf{F}\mathbf{g}_1(\mathbf{T}_j^T\mathbf{S}_{\mathbf{P}}^T\mathbf{u}_{il}) \\ &- \frac{1}{N_i^2}\mathbf{u}_{ik}^T\mathbf{S_P}\mathbf{T}_j\mathbf{g}_1\left(\sum_{\nu=1}^{M_p}\mathbf{F}\mathbf{C}_{f_{i\nu}}\mathbf{T}_\nu^T\mathbf{S}_{\mathbf{P}}^T\mathbf{u}_{il}\right) \in \mathbb{R}^{1\times 9} \\ &= \frac{1}{N_i}\mathbf{u}_{ik}^T\mathbf{S_P}\Big(\mathbf{T}_j\mathbf{g}_1(\mathbf{T}_j^T\mathbf{S}_{\mathbf{P}}^T\mathbf{u}_{il}) - \frac{1}{N_i}\mathbf{C}_i\mathbf{F}\mathbf{g}_1(\mathbf{T}_j^T\mathbf{S}_{\mathbf{P}}^T\mathbf{u}_{il}) \\ &\qquad - \frac{1}{N_i}\mathbf{T}_j\mathbf{g}_1(\mathbf{F}\mathbf{C}_i\mathbf{S}_{\mathbf{P}}^T\mathbf{u}_{il})\Big) \end{aligned} \tag{212}$$

Substitute it into $\mathbf{L}_{ij}^{\mathrm{p}}$:

$$\begin{aligned} \mathbf{L}_{ij}^{\mathrm{p}} &= \frac{2}{N_i}\Big(\mathbf{g}_2(\mathbf{T}_{\mathrm{p}}\mathbf{C}_{f_{i\mathrm{p}}}(\mathbf{T}_{\mathrm{p}}^T - \frac{1}{N_i}\mathbf{F}\mathbf{C}_i)\mathbf{S}_{\mathbf{P}}^T\mathbf{u}_{il}) + \\ &\mathbf{V}_{il}\mathbf{T}_{\mathrm{p}}\mathbf{C}_{f_{i\mathrm{p}}}(\mathbf{T}_{\mathrm{p}}^T - \frac{1}{N}\mathbf{F}\mathbf{C}_i)\mathbf{S}_{\mathbf{P}}^T\Big)\left(\sum_{k=1,k\neq l}^{3}\frac{\mathbf{u}_{ik}\mathbf{G}_{kl}^{ij}}{\lambda_{il}-\lambda_{ik}}\right) \\ &- \frac{2}{N_i^2}\mathbf{V}_{il}\mathbf{T}_{\mathrm{p}}\mathbf{C}_{f_{i\mathrm{p}}}\mathbf{F}\mathbf{T}_j\mathbf{g}_1(\mathbf{T}_j^T\mathbf{S}_{\mathbf{P}}^T\mathbf{u}_{il}) \\ &+ \frac{2}{N_i}\mathbb{1}_{\mathrm{p}=j}\cdot\left(\mathbf{V}_{il}\mathbf{T}_{\mathrm{p}}\mathbf{g}_1\Big((\mathbf{T}_{\mathrm{p}}^T - \frac{1}{N_i}\mathbf{F}\mathbf{C}_i)\mathbf{S}_{\mathbf{P}}^T\mathbf{u}_{il}\Big)\right). \end{aligned} \tag{213}$$

Since $\mathbf{J} = \begin{bmatrix} \cdots & \mathbf{J}^{\mathrm{p}} & \cdots \end{bmatrix}$ and (202), we obtain

$$\frac{\partial \mathbf{J}^T}{\partial \mathbf{c}_{f_{ij}}} = \begin{bmatrix} \vdots \\ \frac{\partial (\mathbf{J}^{\mathrm{p}})^T}{\partial \mathbf{c}_{f_{ij}}} \\ \vdots \end{bmatrix} = \begin{bmatrix} \vdots \\ \mathbf{L}_{ij}^{\mathrm{p}} \\ \vdots \end{bmatrix} \triangleq \mathbf{L}_{ij} \in \mathbb{R}^{6M_p\times 9}, \tag{214}$$

$$\frac{\partial \mathbf{J}^T(\mathbf{T}^\star, \mathbf{C}_f)}{\partial \mathbf{C}_f}\delta\mathbf{C}_f = \sum_{i=1}^{M_f}\sum_{j=1}^{M_p}\mathbf{L}_{ij}\delta\mathbf{c}_{f_{ij}}. \tag{215}$$

Finally, from (52) of the main texts, we have

$$\delta\mathbf{T}^\star = \mathbf{H}^{-1}\left(\sum_{i=1}^{M_f}\sum_{j=1}^{M_p}\mathbf{L}_{ij}\delta\mathbf{c}_{f_{ij}}\right) \in \mathbb{R}^{6M_p}, \tag{216}$$

$$\boldsymbol{\Sigma}_{\delta\mathbf{T}^\star} = \mathbf{H}^{-1}\left(\sum_{i=1}^{M_f}\sum_{j=1}^{M_p}\mathbf{L}_{ij}\boldsymbol{\Sigma}_{\mathbf{c}_{f_{ij}}}\mathbf{L}_{ij}^T\right)\mathbf{H}^{-1}, \tag{217}$$

where

$$\mathbf{\Sigma}_{\mathbf{c}_{f_{ij}}} = \sum_{k=1}^{N_{ij}} \mathbf{B}_{f_{ijk}} \mathbf{\Sigma}_{\mathbf{p}_{f_{ijk}}} \mathbf{B}_{f_{ijk}}^T, \tag{218}$$

which is obtained from (200) and can be computed beforehand without enumerating each raw point in the run time.

IV. Time Complexity Analysis

In this section, we analyze the time complexity of the proposed BA solver in Algorithm 1 and compare it with other similar BA methods, including BALM [32] and Plane Adjustment [34]. The computation time of Algorithm 1 mainly consists of the evaluation of the Jacobian $\mathbf{J}$ and Hessian matrix $\mathbf{H}$ on Line 6, and the solving of the linear equation on Line 9. For the evaluation of the Jacobian $\mathbf{J}$, according to (42), it consists of M_f items, each item $\mathbf{J}_i$ requires to evaluate M_p block elements, and each block $\mathbf{g}_{ll}^j$ requires a constant computation time according to (34). Therefore, the time complexity for the evaluation of $\mathbf{J}$ is $O(M_f M_p)$. Similarly, for the Hessian matrix, according to (42), it consists of M_f items, each item $\mathbf{H}_i$ requires to evaluate M_p^2 block elements, and each block $\mathbf{H}^{ij}$ requires a constant computation time according to (35). Therefore, the time complexity for the evaluation of $\mathbf{H}$ is $O\left(M_f M_p^2\right)$. On Line 9, the linear equation has a dimension of $6M_p$, solving the lienar equation requires an inversion of the Hessian matrix which contributes a time complexity of $O(M_p^3)$. As a result, the overall time complexity of Algorithm 1 is $O\left(M_f M_p + M_f M_p^2 + M_p^3\right)$.

Our previous work BALM [32] also eliminated the feature parameters from the optimization, leading to an optimization similar to (9) whose dimension of Jacobian and Hessian are also $6M_p$. To solve the cost function, BALM [32] adopted a second order solver similar to Algorithm 1, where the linear solver on Line 9 has the same time complexity $O(M_p^3)$. However, when deriving the Jacobian and Hessian matrix of the cost function, the chain rule was used where the Jacobian and Hessian is the multiplication of the cost w.r.t. each point of a feature and the derivative of the point w.r.t. scan poses (see Section III. C of [32]). As a consequence, for each feature, the evaluation of Jacobian has complexity of $O(NM_p)$ and the Hessian has $O(N^2M_p^2)$, where N is the average number of points on each feature. Therefore, the time complexity including the evaluation of all features' Jacobian and Hessian matrices and the linear solver are $O(NM_fM_p + N^2M_fM_p^2 + M_p^3)$.

The plane adjustment method in [34] is a direct mimic of the visual bundle adjustment, which does not eliminate the feature parameters but optimizes them along with the poses in each iteration. The resultant linear equation at each iteration is in the form of $\mathbf{J}^T\mathbf{J}\delta\mathbf{x} = \mathbf{b}$, where $\mathbf{J}$ is the Jacobian of the cost function w.r.t. both the pose and feature parameters. Due to a reduced residual and Jacobian technique similar to our point cluster method, the evaluation of $\mathbf{J}$ does not need to enumerate each point of a feature, so the time complexity of computing $\mathbf{J}$ for all M_f features is $O(M_fM_p + M_f)$ by noticing the inherent sparsity. Let $\mathbf{H} = \mathbf{J}^T\mathbf{J} = \begin{bmatrix} \mathbf{H}_{11} & \mathbf{H}_{12} \\ \mathbf{H}_{12}^T & \mathbf{H}_{22} \end{bmatrix}, \delta\mathbf{x} = \begin{bmatrix} \delta\mathbf{T} \\ \delta\boldsymbol{\pi} \end{bmatrix}, \mathbf{b} = \begin{bmatrix} \mathbf{b}_1 \\ \mathbf{b}_2 \end{bmatrix}$, where $\delta\mathbf{T} \in \mathbb{R}^{6M_p}$ is the pose component and $\delta\boldsymbol{\pi} \in \mathbb{R}^{3M_f}$ is feature component. The plane adjustment in [34] further used a Schur complement technique similar to visual bundle adjustment, leading to the optimization of the pose vector only: $\left(\mathbf{H}_{11} - \mathbf{H}_{12}\mathbf{H}_{22}^{-1}\mathbf{H}_{21}\right)\delta\mathbf{T} = \mathbf{b}_1 - \mathbf{H}_{12}\mathbf{H}_{22}^{-1}\mathbf{b}_2$. Since $\mathbf{H}_{22}$ is block diagonal, its inverse has a time complexity of $O(M_f)$. As a consequence, constructing this linear equation would require computing $\mathbf{H}_{12}\mathbf{H}_{22}^{-1}\mathbf{H}_{21}$ and $\mathbf{H}_{12}\mathbf{H}_{22}^{-1}\mathbf{b}_2$, which have time complexity of $O(M_p^2M_f)$, and solving this linear equation has a complexity of $O(M_p^3)$. As a consequence, the overall time complexity is $O(M_f + M_fM_p + M_fM_p^2 + M_p^3)$.

In summary, BALM [32] has a complexity $O(NM_fM_p + N^2M_fM_p^2 + M_p^3)$, which is linear to M_f, the number of feature, quadratic to N, the number of point, and cubic to M_p, the number of pose. The plane adjustment [34] has a complexity $O(M_f + M_fM_p + M_fM_p^2 + M_p^3)$, which is linear to M_f, irrelevant to N, and cubic to M_p. Our proposed method has a complexity of $O\left(M_fM_p + M_fM_p^2 + M_p^3\right)$, which is similar to [34] (i.e., linear to M_f, irrelevant to N, and cubic to M_p) but has less operations.

Algorithm 1: LM optimization

Input: Initial poses $\mathbf{T}$;
Point cluster in the local frame $\mathbf{C}_{f_{ij}}$;
1 $\mu = 0.01,\ \nu = 2,\ j = 0$;
2 **repeat**
3 $\quad j = j + 1$;
4 $\quad \mathbf{J} = \mathbf{0}_{1\times 6M_p}, \mathbf{H} = \mathbf{0}_{6M_p \times 6M_p}$;
5 $\quad$ **foreach** $i \in \{1, \cdots, M_f\}$ **do**
6 $\quad\quad$ Compute $\mathbf{J}_i$ and $\mathbf{H}_i$ from (30) and (31);
7 $\quad\quad \mathbf{J} = \mathbf{J} + \mathbf{J}_i;\ \mathbf{H} = \mathbf{H} + \mathbf{H}_i$
8 $\quad$ **end**
9 $\quad$ Solve $(\mathbf{H} + \mu\mathbf{I})\Delta\mathbf{T} = -\mathbf{J}^T$;
10 $\quad \mathbf{T}' = \mathbf{T} \boxplus \Delta\mathbf{T}$;
11 $\quad$ Compute current cost $c = c(\mathbf{T})$ and the new cost $c' = c(\mathbf{T}')$ from (22);
12 $\quad \rho = (c - c')/(\frac{1}{2}\Delta\mathbf{T} \cdot (\mu\Delta\mathbf{T} - \mathbf{J}^T))$;
13 $\quad$ **if** $\rho > 0$ **then**
14 $\quad\quad \mathbf{T} = \mathbf{T}'$;
15 $\quad\quad \mu = \mu * \max\left(\frac{1}{3}, 1 - (2\rho - 1)^3\right);\ \nu = 2$;
16 $\quad$ **else**
17 $\quad\quad \mu = \mu * \nu;\ \nu = 2 * \nu$;
18 $\quad$ **end**
19 **until** $\parallel \Delta\mathbf{T} \parallel < \epsilon$ **or** $j \geq j_{max}$;
Output: Final optimized states $\mathbf{T}$;

References

[1] A. I. Mourikis, S. I. Roumeliotis *et al.*, "A multi-state constraint kalman filter for vision-aided inertial navigation." in *ICRA*, vol. 2, 2007, p. 6.

[2] S. Leutenegger, S. Lynen, M. Bosse, R. Siegwart, and P. Furgale, "Keyframe-based visual–inertial odometry using nonlinear optimization," *The International Journal of Robotics Research*, vol. 34, no. 3, pp. 314–334, 2015.

[3] T. Qin, P. Li, and S. Shen, "Vins-mono: A robust and versatile monocular visual-inertial state estimator," *IEEE Transactions on Robotics*, vol. 34, no. 4, pp. 1004–1020, 2018.

[4] C. Campos, R. Elvira, J. J. G. Rodríguez, J. M. Montiel, and J. D. Tardós, "Orb-slam3: An accurate open-source library for visual, visual–inertial, and multimap slam," *IEEE Transactions on Robotics*, vol. 37, no. 6, pp. 1874–1890, 2021.

[5] W. Xu, Y. Cai, D. He, J. Lin, and F. Zhang, "Fast-lio2: Fast direct lidar-inertial odometry," *IEEE Transactions on Robotics*, pp. 1–21, 2022.

[6] C. Forster, L. Carlone, F. Dellaert, and D. Scaramuzza, "On-manifold preintegration for real-time visual–inertial odometry," *IEEE Transactions on Robotics*, vol. 33, no. 1, pp. 1–21, 2016.

[7] W. Xu and F. Zhang, "Fast-lio: A fast, robust lidar-inertial odometry package by tightly-coupled iterated kalman filter," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 3317–3324, 2021.

[8] X. Liu, C. Yuan, and F. Zhang, "Targetless extrinsic calibration of multiple small fov lidars and cameras using adaptive voxelization," *IEEE Transactions on Instrumentation and Measurement*, vol. 71, pp. 1–12, 2022.

[9] X. Liu and F. Zhang, "Extrinsic calibration of multiple lidars of small fov in targetless environments," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 2036–2043, 2021.

[10] Z. Liu and F. Zhang, "Balm: Bundle adjustment for lidar mapping," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 3184–3191, 2021.

[11] X. Liu, Z. Liu, F. Kong, and F. Zhang, "Large-scale lidar consistent mapping using hierarchical lidar bundle adjustment," *IEEE Robotics and Automation Letters*, vol. 8, no. 3, pp. 1523–1530, 2023.

[12] Y. Pan, P. Xiao, Y. He, Z. Shao, and Z. Li, "Mulls: Versatile lidar slam via multi-metric linear least square," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, 2021, pp. 11 633–11 640.

[13] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[14] P. Dellenbach, J.-E. Deschaud, B. Jacquet, and F. Goulette, "Ct-icp: Real-time elastic lidar odometry with loop closure," *arXiv preprint arXiv:2109.12979*, 2021.

[15] G. Kim and A. Kim, "Scan context: Egocentric spatial descriptor for place recognition within 3d point cloud map," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 4802–4809.