LL3D Aligner Formulation

Zoox, Inc.

I. Introduction

The LL3D aligner aligns a new metaspin to the current voxel grid, so that points can be added to the correct voxels. More specifically, the points in the metaspin are aligned to *surfels* built from the voxel grid using ICP. The standard ICP iterates between a data association step and an optimization step. For the aligner, we replace the data association step by a voxel lookup and finds the closest point from the point to the surfel of the voxel it is in.

This document contains the exact mathematical formulation of the aligner, with derivations and proofs.

II. Cost Function Definition

Given the current voxel grid G, and set of points $\{\mathbf{p}_i\}$, we want to find a transform $T \in \mathrm{SE}(3)$ that minimizes the following cost function:

$$C(T) = \sum_{i=1}^{N} f(\mathbf{g}(\mathbf{p}_i, T), G) \tag{1}$$

where $\mathbf{g}$ is a vector function that applies the SE(3) transform T to a point $\mathbf{p}$, and f is a scalar function that given a point $\mathbf{p}$, gives a measure of its “distance” to the grid G.

To define precisely the grid G, we index its voxels by an integer triple $\mathbf{v} \in \mathbb{Z}^3$, some voxels have surfels, some do not (unoccupied, too few points etc.), let the set $H \subseteq \mathbb{Z}^3$ be the set of voxels having valid surfels, and $P : H \to \mathcal{P}$ be a function that maps a valid voxel to its surfel (here $\mathcal{P}$ is the set of all planes in $\mathbb{R}^3$, any representation will do). Then $G = (H, P)$.

Next we define the function f as follows:

$$f(\mathbf{p}, G) = \begin{cases} ||\mathbf{p} - \mathbf{r}(\mathbf{p}, P_{\mathbf{v}(\mathbf{p})})||^2 & \text{if } \mathbf{v}(\mathbf{p}) \in H \\ l^2 & \text{otherwise} \end{cases} \tag{2}$$

Here $\mathbf{v} : \mathbb{R}^3 \to \mathbb{Z}^3$ is a function that finds the voxel a point falls in, $\mathbf{r}$ is a function that given a point $\mathbf{p}$ and a plane P, returns the point on the plane that is closet to $\mathbf{p}$, and l is the diagonal length of a voxel. Because l^2 is an upper bound on any square distance from a point to a surfel (see Theorem 1), Equation 2 penalizes the case a point falls to an invalid voxel.

III. ICP Solver

We follow the standard ICP procedure of first fixing T and compute the closest points, then fixing the closest points and optimize for T. Let's rearrange the cost function to make the two steps explicit. First define a sequence of voxel indices $\{\mathbf{v}_i\}$ that is a function of T, each voxel index is given by $\mathbf{v}_i = \mathbf{v}(\mathbf{g}(\mathbf{p}_i, T))$, that is, the index of voxel the *transformed* point falls in. Then define the set of closest points $\{\mathbf{r}_i\}(T, \{\mathbf{v}_i\}, G)$, where each $\mathbf{r}_i$ is given by:

$$\mathbf{r}_i = \begin{cases} \mathbf{r}(\mathbf{g}(\mathbf{p}_i, T), P_{\mathbf{v}_i}) & \text{if } \mathbf{v}_i \in H \\ \mathbf{0} & \text{otherwise} \end{cases} \tag{3}$$

And define a set I that is a function of $\{\mathbf{v}_i\}$ and G:

$$I(\{\mathbf{v}_i\}, G) = \{i : \mathbf{v}_i \in H\} \tag{4}$$

Then the cost $C(T)$ can be written as

$$\begin{aligned} C(T) &= C(T, \{\mathbf{v}_i\}(T), \{\mathbf{r}_i\}(T, \{\mathbf{v}_i\}(T))) \\ &= \sum_{i \in I(\{\mathbf{v}_i\}(T))} ||\mathbf{g}(\mathbf{p}_i, T) - \mathbf{r}_i(T, \mathbf{v}_i(T))||^2 + \sum_{i \notin I(\{\mathbf{v}_i\}(T))} l^2 \end{aligned} \tag{5}$$

here we are only showing dependency on T for clarity (you can think of G as a constant).

Then ICP amounts to first fix $T = T^*$ to find $\{\mathbf{v}_i\}(T^*)$ and $\{\mathbf{r}_i\}(T^*, \{\mathbf{v}_i\}(T^*))$, then fix $\{\mathbf{v}_i\}(T^*)$ and $\{\mathbf{r}_i\}(T^*, \{\mathbf{v}_i\}(T^*))$, and optimize:

$$\begin{aligned} C^*(T) &= C(T, \{\mathbf{v}_i\}(T^*), \{\mathbf{r}_i\}(T^*, \{\mathbf{v}_i\}(T^*))) \\ &= \sum_{i \in I(\{\mathbf{v}_i\}(T^*))} ||\mathbf{g}(\mathbf{p}_i, T) - \mathbf{r}_i(T^*, \mathbf{v}_i(T^*))||^2 + \sum_{i \notin I(\{\mathbf{v}_i\}(T^*))} l^2 \end{aligned} \tag{6}$$

It turns out there is a closed-form solution to finding the *global* minimum of $C^*(T)$. If we parameterize the SE(3) transform by a unit quaternion and a translation vector $T = (\mathbf{q}, \mathbf{t})$, Algorithm 1 lists the steps required to compute the global minimum in closed-form, based on an eigenvalue decomposition. In the algorithm pseudo code, $I = I(\{\mathbf{v}_i\}(T^*))$ and $\mathbf{r}_i = \mathbf{r}_i(T^*, \mathbf{v}_i(T^*))$ to keep the presentation clear. The fact this algorithm indeed gives the global minimum is proved in Theorem 2. Note that Algorithm 1 is essentially the same as the algorithm described in Section III-C of the original ICP paper [1], except the small difference that the matrix M defined in Algorithm 1 is the *transpose* of the cross-covariance matrix Σ_{px} defined in [1].

IV. Regularized ICP

Once alignment has been run long enough, we may drift arbitrarily. Drifting in pitch and roll is particularly harmful because the ground plane may eventually drift off to be vertical and extend outside of the limited z-bounds of the LL3D volume. In this section we consider adding a regularization term to the standard formulation to constrain drift in pitch and roll.

The local_utm frame's z-axis is always opposite to the gravity direction. Hence, we can use local_utm frame's z-axis to constrain LL3D aligner's pitch and roll. In order to do this, we can compute the 6dof transformation local_utm_from_grid. Last column of the rotation matrix of local_utm_from_grid transformation gives us the drift in pitch and roll from the reference vertical direction.

In the following equations, let's denote grid frame as G, local_utm frame as L, cloud frame as C and rotation from source coordinate frame A to destination coordinate frame B as $\mathbf{R}_A^B$. So, the above constraint can be added as a regularization term:

$$\zeta(\mathbf{q}) = (\mathbf{R}_G^L(\mathbf{q})\hat{\mathbf{z}})^T \hat{\mathbf{z}} \tag{7}$$

Here, $\hat{\mathbf{z}} = [0, 0, 1]^T$. Goal here is to maximize $\zeta(\mathbf{q})$.

Note that the goal of the aligner is to compute the transformation: grid_from_cloud. So, $\zeta(\mathbf{q})$ becomes:

$$\zeta(\mathbf{q}) = (\mathbf{R}_C^L {\mathbf{R}_C^G}^T(\mathbf{q})\hat{\mathbf{z}})^T \hat{\mathbf{z}} = \hat{\mathbf{z}}^T \mathbf{R}_C^G(\mathbf{q}) {\mathbf{R}_C^L}^T \hat{\mathbf{z}} \tag{8}$$

Choosing

$$\hat{\mathbf{u}} = {\mathbf{R}_C^L}^T \hat{\mathbf{z}} = \mathbf{R}_L^C \hat{\mathbf{z}} \tag{9}$$

Algorithm 1 Closed-form global solution for $\hat{T} = \arg\min_T C^*(T)$

if $I = \varnothing$ **then**
  $\hat{T} \leftarrow T^*$
  return $\hat{T}$
end if
$\mathbf{M}_{rp} \leftarrow \frac{1}{|I|} \sum_{i \in I} \mathbf{r}_i \mathbf{p}_i^T$
$\bar{\mathbf{r}} \leftarrow \frac{1}{|I|} \sum_{i \in I} \mathbf{r}_i$
$\bar{\mathbf{p}} \leftarrow \frac{1}{|I|} \sum_{i \in I} \mathbf{p}_i$
$\mathbf{M} \leftarrow \mathbf{M}_{rp} - \bar{\mathbf{r}}\bar{\mathbf{p}}^T$
$\mathbf{Q} \leftarrow \begin{bmatrix} M_{11}+M_{22}+M_{33} & -M_{23}+M_{32} & M_{13}-M_{31} & -M_{12}+M_{21} \\ -M_{23}+M_{32} & M_{11}-M_{22}-M_{33} & M_{12}+M_{21} & M_{13}+M_{31} \\ M_{13}-M_{31} & M_{12}+M_{21} & -M_{11}+M_{22}-M_{33} & M_{23}+M_{32} \\ -M_{12}+M_{21} & M_{13}+M_{31} & M_{23}+M_{32} & -M_{11}-M_{22}+M_{33} \end{bmatrix}$
$\hat{\mathbf{q}} \leftarrow \text{EigenvectorOfLargestEigenvalue}(\mathbf{Q})$
$\hat{\mathbf{R}} \leftarrow \text{ToRotationMatrix}(\hat{\mathbf{q}})$
$\hat{\mathbf{t}} \leftarrow \bar{\mathbf{r}} - \hat{\mathbf{R}}\bar{\mathbf{p}}$
$\hat{T} \leftarrow (\hat{\mathbf{q}}, \hat{\mathbf{t}})$
return $\hat{T}$

and denoting

$$\mathbf{R}(\mathbf{q}) = \mathbf{R}_C^G(\mathbf{q}) \tag{10}$$

we have:

$$\begin{aligned} \zeta(\mathbf{q}) &= \hat{\mathbf{z}}^T(\mathbf{R}(\mathbf{q})\hat{\mathbf{u}}) \\ &= \text{tr}(\hat{\mathbf{z}}(\mathbf{R}(\mathbf{q})\hat{\mathbf{u}})^T) \\ &= \text{tr}(\hat{\mathbf{z}}\hat{\mathbf{u}}^T\mathbf{R}^T(\mathbf{q})) \end{aligned} \tag{11}$$

If $\hat{\mathbf{u}} = [u_1, u_2, u_3]$, then we have

$$\hat{\mathbf{z}}\hat{\mathbf{u}}^T = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ u_1 & u_2 & u_3 \end{bmatrix} \tag{12}$$

Maximizing the regularization term described in Equation 11 is equivalent to minimizing its negative. Let λ be the regularization weight which is independent of the number of points. As $C(T)$ has been defined as the sum of losses and not the average loss, we need to multiply the regularization term with N. So the overall cost function we optimize is:

$$\bar{C}(T) = \sum_{i=1}^{N} f(\mathbf{g}(\mathbf{p}_i, T), G) - \lambda N(\zeta(\mathbf{q}) - 1) \tag{13}$$

Note here that $\mathbf{q}$ is the quaternion representation of the rotation component of T. We've also added a -1 to $\zeta(\mathbf{q})$ to ensure the overall cost function is still ≥ 0.

Observing that Equation 11 and Equation 33, have very similar form. This suggests that the optimal transformation can be achieved by modifying the last row of $\mathbf{M}$ in Equation 34. Since the final expression in Equation 33 also has a negative sign and a $2|I|$ factor outside the trace, we can first modify λ to:

$$\lambda \leftarrow \lambda N/(2|I|) \tag{14}$$

and then update $\mathbf{M}$ to be:

$$\begin{aligned} M_{31} &\leftarrow M_{31} + \lambda u_1 \\ M_{32} &\leftarrow M_{32} + \lambda u_2 \\ M_{33} &\leftarrow M_{33} + \lambda u_3 \end{aligned} \tag{15}$$

APPENDIX

Theorem 1. *Given a cubic box B in $\mathbb{R}^3$, a plane P fitted to points in B in the least-square sense, and a point $\mathbf{p} \in B$, we have*

$$||\mathbf{p} - \mathbf{r}(\mathbf{p}, P)||^2 \leq l^2 \tag{16}$$

where $\mathbf{r}$ is a function returning the closest point on P to $\mathbf{p}$, and l is the diagonal length of B.

Proof. Because P is a least-square-fit plane of points in B, it goes through the mean of the points $\mathbf{m}$, and we have $\mathbf{m} \in B$ (mean preserves inequalities), then

$$||\mathbf{p} - \mathbf{r}(\mathbf{p}, P)||^2 = ((\mathbf{p} - \mathbf{m}) \cdot \hat{\mathbf{n}})^2 \leq ||\mathbf{p} - \mathbf{m}||^2 \leq l^2 \tag{17}$$

where $\hat{\mathbf{n}}$ is the unit normal of P. □

Lemma 1. *Given two sequences of vectors $\{\mathbf{a}_i\}$ and $\{\mathbf{b}_i\}$ of the same length $N > 0$, and each vector is D-dimensional (with $D > 0$), we have*

$$\frac{1}{N}\sum_{i=1}^{N} \mathbf{a}_i^T \mathbf{b}_i = tr(\frac{1}{N}\sum_{i=1}^{N} \mathbf{a}_i \mathbf{b}_i^T) \tag{18}$$

Proof.

$$\begin{aligned} \text{tr}(\frac{1}{N}\sum_{i=1}^{N} \mathbf{a}_i \mathbf{b}_i^T) &= \sum_{j=1}^{D} \frac{1}{N} \sum_{i=1}^{N} [\mathbf{a}_i]_j [\mathbf{b}_i]_j \\ &= \frac{1}{N} \sum_{i=1}^{N} \sum_{j=1}^{D} [\mathbf{a}_i]_j [\mathbf{b}_i]_j \\ &= \frac{1}{N} \sum_{i=1}^{N} \mathbf{a}_i^T \mathbf{b}_i \end{aligned} \tag{19}$$

□

Lemma 2. *Given a $D \times D$ ($D > 0$) real symmetric matrix $\mathbf{A}$, we have*

$$\forall (\mathbf{x} \in \mathbb{R}^D, ||\mathbf{x}|| = 1), \mathbf{x}^T \mathbf{A} \mathbf{x} \leq \mathbf{u}^T \mathbf{A} \mathbf{u} \tag{20}$$

where $\mathbf{u}$ is the eigenvector (normalized) corresponding to the largest eigenvalue of $\mathbf{A}$.

Proof. Since $\mathbf{A}$ is real symmetric, it has an eigenvalue decomposition

$$\mathbf{A} = \mathbf{W}\mathbf{\Lambda}\mathbf{W}^T \tag{21}$$

where $\mathbf{W}$ is orthogonal and $\mathbf{\Lambda}$ is the diagonal matrix with the eigenvalues along the diagonal. Then we have $\forall(\mathbf{x} \in \mathbb{R}^D, ||\mathbf{x}|| = 1)$:

$$\begin{aligned}\mathbf{x}^T\mathbf{A}\mathbf{x} &= \mathbf{x}^T\mathbf{W}\mathbf{\Lambda}\mathbf{W}^T\mathbf{x} \\ &= (\mathbf{W}^T\mathbf{x})^T\mathbf{\Lambda}(\mathbf{W}^T\mathbf{x}) \\ &= \mathbf{y}^T\mathbf{\Lambda}\mathbf{y} \\ &= \sum_{i=1}^{D}\lambda_i y_i^2\end{aligned} \tag{22}$$

We defined the variable $\mathbf{y} = \mathbf{W}^T\mathbf{x}$ in the above. Now because $||\mathbf{x}|| = 1$ and $\mathbf{W}$ is orthogonal, we have

$$\sum_{i=1}^{D} y_i^2 = \mathbf{y}^T\mathbf{y} = \mathbf{x}^T\mathbf{W}\mathbf{W}^T\mathbf{x} = \mathbf{x}^T\mathbf{x} = 1 \tag{23}$$

Let λ be an eigenvalue such that $\lambda = \max_i \lambda_i$, and $\mathbf{u}$ be its (normalized) eigenvector, then because $\forall i, \lambda_i \leq \lambda, y_i^2 \geq 0$

$$\mathbf{x}^T\mathbf{A}\mathbf{x} = \sum_{i=1}^{D}\lambda_i y_i^2 \leq \sum_{i=1}^{D}\lambda y_i^2 = \lambda\sum_{i=1}^{D} y_i^2 = \lambda \tag{24}$$

Also we have

$$\mathbf{u}^T\mathbf{A}\mathbf{u} = \mathbf{u}^T(\mathbf{A}\mathbf{u}) = \mathbf{u}^T(\lambda\mathbf{u}) = \lambda \tag{25}$$

This completes the proof. □

Theorem 2. *Algorithm 1 returns a $\hat{T}$ that is the global minimum of $C^*(T)$. In other words, $\hat{T} \in SE(3)$ and $\forall T \in SE(3), C^*(T) \geq C^*(\hat{T})$.*

Proof. If $I = \varnothing$, we have $C^*(T) = \sum_{i=1}^{N} l^2$, which is constant, hence any $\hat{T} \in$ SE(3) is a global minimum. Since the transform from the previous iteration T^* is in SE(3) by recursion, $\hat{T}$ is also in SE(3). Let us focus on the case $I \neq \varnothing$.

When $I \neq \varnothing$, by definition Algorithm 1 always returns a unit vector $\hat{\mathbf{q}}$, hence we have $\hat{T} \in$ SE(3). Now let's concentrate on proving the second half of the theorem statement, that is $\forall T \in$ SE(3)$, C^*(T) \geq C^*(\hat{T})$. First let's get rid of the translation component then we can focus on $\mathbf{q}$. Expanding $\mathbf{g}(\mathbf{p}_i, T) = \mathbf{R}(\mathbf{q})\mathbf{p}_i + \mathbf{t}$ into Equation 6, we have

$$C^*(T) = \sum_{i\in I}||\mathbf{R}(\mathbf{q})\mathbf{p}_i + \mathbf{t} - \mathbf{r}_i||^2 + \sum_{i\notin I} l^2 \tag{26}$$

Concentrating on the first term and define $\mathbf{b}_i = \mathbf{r}_i - \mathbf{R}(\mathbf{q})\mathbf{p}_i$, we have

$$\begin{aligned}\sum_{i\in I}||\mathbf{R}(\mathbf{q})\mathbf{p}_i + \mathbf{t} - \mathbf{r}_i||^2 &= \sum_{i\in I}||\mathbf{t} - \mathbf{b}_i||^2 \\ &= \sum_{i\in I}(\mathbf{t} - \mathbf{b}_i)^T(\mathbf{t} - \mathbf{b}_i) \\ &= \sum_{i\in I}(\mathbf{t}^T\mathbf{t} - 2\mathbf{b}_i^T\mathbf{t} + \mathbf{b}_i^T\mathbf{b}_i)\end{aligned} \tag{27}$$

Now using the usual notation to denote the mean and covariance matrix of $\{\mathbf{b}_i\}$, i.e $\bar{\mathbf{b}} = 1/|I|\sum_{i\in I}\mathbf{b}_i$ and $\mathbf{\Sigma} = 1/|I|\sum_{i\in I}(\mathbf{b}_i - \bar{\mathbf{b}})(\mathbf{b}_i - \bar{\mathbf{b}})^T$, by Lemma 1, we have

$$\begin{aligned}\text{tr}(\mathbf{\Sigma}) &= \text{tr}\left(\frac{1}{|I|}\sum_{i\in I}(\mathbf{b}_i - \bar{\mathbf{b}})(\mathbf{b}_i - \bar{\mathbf{b}})^T\right) \\ &= \frac{1}{|I|}\sum_{i\in I}(\mathbf{b}_i - \bar{\mathbf{b}})^T(\mathbf{b}_i - \bar{\mathbf{b}}) \\ &= \frac{1}{|I|}\sum_{i\in I}(\mathbf{b}_i^T\mathbf{b}_i - 2\bar{\mathbf{b}}^T\mathbf{b}_i + \bar{\mathbf{b}}^T\bar{\mathbf{b}}) \\ &= \frac{1}{|I|}\sum_{i\in I}\mathbf{b}_i^T\mathbf{b}_i - 2\bar{\mathbf{b}}^T\bar{\mathbf{b}} + \bar{\mathbf{b}}^T\bar{\mathbf{b}} \\ &= \frac{1}{|I|}\sum_{i\in I}\mathbf{b}_i^T\mathbf{b}_i - \bar{\mathbf{b}}^T\bar{\mathbf{b}}\end{aligned} \tag{28}$$

With this result, let us go back to Equation 27

$$\begin{aligned}\sum_{i\in I}||\mathbf{R}(\mathbf{q})\mathbf{p}_i + \mathbf{t} - \mathbf{r}_i||^2 &= \sum_{i\in I}(\mathbf{t}^T\mathbf{t} - 2\mathbf{b}_i^T\mathbf{t} + \mathbf{b}_i^T\mathbf{b}_i) \\ &= |I|(\mathbf{t}^T\mathbf{t} - 2\bar{\mathbf{b}}^T\mathbf{t} + \text{tr}(\mathbf{\Sigma}) + \bar{\mathbf{b}}^T\bar{\mathbf{b}}) \\ &= |I|((\mathbf{t} - \bar{\mathbf{b}})^T(\mathbf{t} - \bar{\mathbf{b}}) + \text{tr}(\mathbf{\Sigma})) \\ &\geq |I|\text{tr}(\mathbf{\Sigma})\end{aligned} \tag{29}$$

with equality at $\mathbf{t} = \bar{\mathbf{b}} = \bar{\mathbf{r}} - \mathbf{R}(\mathbf{q})\bar{\mathbf{p}}$. Hence we have

$$C^*(T) = C^*(\mathbf{q}, \mathbf{t}) \geq C^*(\mathbf{q}, \bar{\mathbf{r}} - \mathbf{R}(\mathbf{q})\bar{\mathbf{p}}) \tag{30}$$

Let us define a new cost function on only $\mathbf{q}$ to be

$$C_q^*(\mathbf{q}) = C^*(\mathbf{q}, \bar{\mathbf{r}} - \mathbf{R}(\mathbf{q})\bar{\mathbf{p}}) \tag{31}$$

Then we wish to show that $C_q^*(\mathbf{q}) \geq C_q^*(\hat{\mathbf{q}})$ for any $\mathbf{q} \in \mathbb{R}^4, ||\mathbf{q}|| = 1$. Let's first expand the expression for $C_q^*(\mathbf{q})$

$$\begin{aligned}C_q^*(\mathbf{q}) &= \sum_{i\in I}||\mathbf{R}(\mathbf{q})\mathbf{p}_i + \bar{\mathbf{r}} - \mathbf{R}(\mathbf{q})\bar{\mathbf{p}} - \mathbf{r}_i||^2 + \sum_{i\notin I} l^2 \\ &= \sum_{i\in I}||\mathbf{R}(\mathbf{q})(\mathbf{p}_i - \bar{\mathbf{p}}) - (\mathbf{r}_i - \bar{\mathbf{r}})||^2 + \sum_{i\notin I} l^2 \\ &= \sum_{i\in I}(\mathbf{R}(\mathbf{q})(\mathbf{p}_i - \bar{\mathbf{p}}) - (\mathbf{r}_i - \bar{\mathbf{r}}))^T(\mathbf{R}(\mathbf{q})(\mathbf{p}_i - \bar{\mathbf{p}}) - (\mathbf{r}_i - \bar{\mathbf{r}})) \\ &\quad + \sum_{i\notin I} l^2 \\ &= \sum_{i\in I}\Big((\mathbf{p}_i - \bar{\mathbf{p}})^T\mathbf{R}^T(\mathbf{q}) - (\mathbf{r}_i - \bar{\mathbf{r}})^T\Big) \\ &\quad (\mathbf{R}(\mathbf{q})(\mathbf{p}_i - \bar{\mathbf{p}}) - (\mathbf{r}_i - \bar{\mathbf{r}})) + \sum_{i\notin I} l^2 \\ &= \sum_{i\in I}\Big((\mathbf{p}_i - \bar{\mathbf{p}})^T\mathbf{R}^T(\mathbf{q})\mathbf{R}(\mathbf{q})(\mathbf{p}_i - \bar{\mathbf{p}}) \\ &\quad -2(\mathbf{r}_i - \bar{\mathbf{r}})^T\mathbf{R}(\mathbf{q})(\mathbf{p}_i - \bar{\mathbf{p}}) + (\mathbf{r}_i - \bar{\mathbf{r}})^T(\mathbf{r}_i - \bar{\mathbf{r}})\Big) \\ &\quad + \sum_{i\notin I} l^2 \\ &= \sum_{i\in I}\Big((\mathbf{p}_i - \bar{\mathbf{p}})^T(\mathbf{p}_i - \bar{\mathbf{p}}) \\ &\quad -2(\mathbf{r}_i - \bar{\mathbf{r}})^T\mathbf{R}(\mathbf{q})(\mathbf{p}_i - \bar{\mathbf{p}}) + (\mathbf{r}_i - \bar{\mathbf{r}})^T(\mathbf{r}_i - \bar{\mathbf{r}})\Big) \\ &\quad + \sum_{i\notin I} l^2 \\ &= \sum_{i\in I}(\mathbf{p}_i - \bar{\mathbf{p}})^T(\mathbf{p}_i - \bar{\mathbf{p}}) \\ &\quad - 2\sum_{i\in I}(\mathbf{r}_i - \bar{\mathbf{r}})^T\mathbf{R}(\mathbf{q})(\mathbf{p}_i - \bar{\mathbf{p}}) + \sum_{i\in I}(\mathbf{r}_i - \bar{\mathbf{r}})^T(\mathbf{r}_i - \bar{\mathbf{r}}) \\ &\quad + \sum_{i\notin I} l^2\end{aligned} \tag{32}$$

Let us now concentrate on the term involving $\mathbf{q}$, the other terms are all constant:

$$
\begin{aligned}
&-2\sum_{i\in I}(\mathbf{r}_i-\bar{\mathbf{r}})^T\mathbf{R}(\mathbf{q})(\mathbf{p}_i-\bar{\mathbf{p}}) \\
&\quad=-2|I|\left(\frac{1}{|I|}\sum_{i\in I}(\mathbf{r}_i-\bar{\mathbf{r}})^T\mathbf{R}(\mathbf{q})(\mathbf{p}_i-\bar{\mathbf{p}})\right) \\
&\quad=-2|I|\mathrm{tr}\left(\frac{1}{|I|}\sum_{i\in I}(\mathbf{r}_i-\bar{\mathbf{r}})(\mathbf{R}(\mathbf{q})(\mathbf{p}_i-\bar{\mathbf{p}}))^T\right) \\
&\quad=-2|I|\mathrm{tr}\left(\frac{1}{|I|}\sum_{i\in I}(\mathbf{r}_i-\bar{\mathbf{r}})(\mathbf{p}_i-\bar{\mathbf{p}})^T\mathbf{R}^T(\mathbf{q})\right) \\
&\quad=-2|I|\mathrm{tr}\left(\mathbf{M}\mathbf{R}^T(\mathbf{q})\right)
\end{aligned}
\tag{33}
$$

Here again we applied Lemma 1, and the fact that $\mathbf{M}=\mathbf{M}_{rp}-\bar{\mathbf{r}}\bar{\mathbf{p}}^T=\frac{1}{|I|}\sum_{i\in I}(\mathbf{r}_i-\bar{\mathbf{r}})(\mathbf{p}_i-\bar{\mathbf{p}})^T$. To see this, expand the sum:

$$
\begin{aligned}
\frac{1}{|I|}\sum_{i\in I}(\mathbf{r}_i-\bar{\mathbf{r}})(\mathbf{p}_i-\bar{\mathbf{p}})^T &= \frac{1}{|I|}\sum_{i\in I}\left(\mathbf{r}_i\mathbf{p}_i^T-\mathbf{r}_i\bar{\mathbf{p}}^T-\bar{\mathbf{r}}\mathbf{p}_i^T+\bar{\mathbf{r}}\bar{\mathbf{p}}^T\right) \\
&= \frac{1}{|I|}\sum_{i\in I}\mathbf{r}_i\mathbf{p}_i^T-\bar{\mathbf{r}}\bar{\mathbf{p}}^T-\bar{\mathbf{r}}\bar{\mathbf{p}}^T+\bar{\mathbf{r}}\bar{\mathbf{p}}^T \\
&= \mathbf{M}_{rp}-\bar{\mathbf{r}}\bar{\mathbf{p}}^T=\mathbf{M}
\end{aligned}
\tag{34}
$$

Using the standard conversion from a unit quaternion $\mathbf{q}=(w,\mathbf{v})$ to a rotation matrix

$$
\mathbf{R}(\mathbf{q})=\left(w^2-||\mathbf{v}||^2\right)\mathbf{I}+2\left(w[\mathbf{v}]_\times+\mathbf{v}\mathbf{v}^T\right) \tag{35}
$$

we have

$$
\begin{aligned}
&\mathbf{R}(\mathbf{q}) \\
&=\begin{bmatrix} w^2-v_1^2-v_2^2-v_3^2 & 0 & 0 \\ 0 & w^2-v_1^2-v_2^2-v_3^2 & 0 \\ 0 & 0 & w^2-v_1^2-v_2^2-v_3^2 \end{bmatrix} \\
&\quad+2\left(w\begin{bmatrix} 0 & -v_3 & v_2 \\ v_3 & 0 & -v_1 \\ -v_2 & v_1 & 0 \end{bmatrix}+\begin{bmatrix} v_1^2 & v_1v_2 & v_1v_3 \\ v_2v_1 & v_2^2 & v_2v_3 \\ v_3v_1 & v_3v_2 & v_3^2 \end{bmatrix}\right) \\
&=\begin{bmatrix} w^2-v_1^2-v_2^2-v_3^2 & 0 & 0 \\ 0 & w^2-v_1^2-v_2^2-v_3^2 & 0 \\ 0 & 0 & w^2-v_1^2-v_2^2-v_3^2 \end{bmatrix} \\
&\quad+\begin{bmatrix} 0 & -2wv_3 & 2wv_2 \\ 2wv_3 & 0 & -2wv_1 \\ -2wv_2 & 2wv_1 & 0 \end{bmatrix}+\begin{bmatrix} 2v_1^2 & 2v_1v_2 & 2v_1v_3 \\ 2v_2v_1 & 2v_2^2 & 2v_2v_3 \\ 2v_3v_1 & 2v_3v_2 & 2v_3^2 \end{bmatrix} \\
&=\begin{bmatrix} w^2+v_1^2-v_2^2-v_3^2 & 2(v_1v_2-wv_3) & 2(v_1v_3+wv_2) \\ 2(v_2v_1+wv_3) & w^2-v_1^2+v_2^2-v_3^2 & 2(v_2v_3-wv_1) \\ 2(v_3v_1-wv_2) & 2(v_3v_2+wv_1) & w^2-v_1^2-v_2^2+v_3^2 \end{bmatrix}
\end{aligned}
\tag{36}
$$

Substituting this result into $\mathrm{tr}(\mathbf{M}\mathbf{R}^T(\mathbf{q}))$, we get

$$
\begin{aligned}
&\mathrm{tr}(\mathbf{M}\mathbf{R}^T(\mathbf{q})) \\
&=\sum_{i=1}^3\sum_{j=1}^3 M_{ij}R_{ij}(\mathbf{q}) \\
&=M_{11}(w^2+v_1^2-v_2^2-v_3^2)+2M_{12}(v_1v_2-wv_3)+2M_{13}(v_1v_3+wv_2) \\
&\quad+2M_{21}(v_2v_1+wv_3)+M_{22}(w^2-v_1^2+v_2^2-v_3^2)+2M_{23}(v_2v_3-wv_1) \\
&\quad+2M_{31}(v_3v_1-wv_2)+2M_{32}(v_3v_2+wv_1)+M_{33}(w^2-v_1^2-v_2^2+v_3^2) \\
&=(M_{11}+M_{22}+M_{33})w^2+2(-M_{23}+M_{32})wv_1+2(M_{13}-M_{31})wv_2 \\
&\quad+2(-M_{12}+M_{21})wv_3+(M_{11}-M_{22}-M_{33})v_1^2+2(M_{12}+M_{21})v_1v_2 \\
&\quad+2(M_{13}+M_{31})v_1v_3+(-M_{11}+M_{22}-M_{33})v_2^2 \\
&\quad+2(M_{23}+M_{32})v_2v_3+(-M_{11}-M_{22}+M_{33})v_3^2 \\
&=\mathbf{q}^T\mathbf{Q}\mathbf{q}
\end{aligned}
\tag{37}
$$

where $\mathbf{Q}$ is the matrix defined in Algorithm 1. Now because $\mathbf{Q}$ is real symmetric, Lemma 2 applies, and we have

$$
\forall(\mathbf{q}\in\mathbb{R}^4,||\mathbf{q}||=1),\mathbf{q}^T\mathbf{Q}\mathbf{q}\le\hat{\mathbf{q}}^T\mathbf{Q}\hat{\mathbf{q}} \tag{38}
$$

This implies

$$
\forall(\mathbf{q}\in\mathbb{R}^4,||\mathbf{q}||=1),C_q^*(\mathbf{q})\ge C_q^*(\hat{\mathbf{q}}) \tag{39}
$$

Hence $\forall(T=(\mathbf{q},\mathbf{t})\in\mathrm{SE}(3))$,

$$
\begin{aligned}
C^*(T)=C^*(\mathbf{q},\mathbf{t}) &\ge C^*(\mathbf{q},\bar{\mathbf{r}}-\mathbf{R}(\mathbf{q})\bar{\mathbf{p}}) \\
&=C_q^*(\mathbf{q})\ge C_q^*(\hat{\mathbf{q}})=C^*(\hat{\mathbf{q}},\bar{\mathbf{r}}-\mathbf{R}(\hat{\mathbf{q}})\bar{\mathbf{p}}) \\
&=C^*(\hat{\mathbf{q}},\hat{\mathbf{t}})=C^*(\hat{T})
\end{aligned}
\tag{40}
$$

□

References

[1] P. J. Besl and N. D. McKay. A method for registration of 3-d shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):239–256, 1992.