[Supplementary] Procrustean Normal Distribution for Non-Rigid Structure from Motion

Minsik Lee, *Member, IEEE,* Jungchan Cho, *Member, IEEE,* and Songhwai Oh, *Member, IEEE*

1 Deriving the third condition of (2)

The third condition of (2) is the optimality condition for $\mathbf{R}_i$ when the other variables are fixed. Note that we can eliminate $\mathbf{R}_i$ in the last constraint of (1), i.e., the constraint is identical to $\left\| s_i\mathbf{X}_i\left(\mathbf{I}-\frac{1}{n_p}\mathbf{1}\mathbf{1}^T\right)\right\| = 1$, hence this constraint has nothing to do with $\mathbf{R}_i$. Note that $\mathbf{t}_i = -\frac{1}{n_p}s_i\mathbf{R}_i\mathbf{X}_i\mathbf{1}$ because of the second condition in (2), and let us denote $\mathbf{X}'_i \triangleq \mathbf{X}_i\left(\mathbf{I}-\frac{1}{n_p}\mathbf{1}\mathbf{1}^T\right)$, then

$$\left\|s_i\mathbf{R}_i\mathbf{X}_i + \mathbf{t}_i\mathbf{1}^T - \overline{\mathbf{X}}\right\|^2 = \left\|s_i\mathbf{R}_i\mathbf{X}'_i - \overline{\mathbf{X}}\right\|^2. \quad (29)$$

This problem is the orthogonal Procrustes problem [1], where the optimal $\mathbf{R}_i$ that minimizes (29) is given as $\mathbf{R}_i = \mathbf{V}_i\mathbf{U}_i^T$ from the SVD of $\mathbf{X}'_i\overline{\mathbf{X}}^T = \mathbf{U}_i\mathbf{\Lambda}_i\mathbf{V}_i^T$. From the polar decomposition of $\mathbf{X}'_i\overline{\mathbf{X}}^T = (\mathbf{U}_i\mathbf{V}_i^T)(\mathbf{V}_i\mathbf{\Lambda}_i\mathbf{V}_i^T)$, one can conclude that $\mathbf{R}_i$ is optimal *iff* it satisfies

$$\left(s_i\mathbf{R}_i\mathbf{X}_i + \mathbf{t}_i\mathbf{1}^T\right)\overline{\mathbf{X}}^T = s_i\mathbf{R}_i\mathbf{X}'_i\overline{\mathbf{X}}^T \in \mathbf{S}_+^{n_d}.$$

2 Proof of Proposition 1

First, we show that $(\mathbf{1}\otimes\mathbf{I})$ is orthogonal to the other terms. Note that

$$(\mathbf{1}\otimes\mathbf{I})^T\,\mathrm{vec}\big(\overline{\mathbf{Y}}\big) = \overline{\mathbf{Y}}\mathbf{1} = \mathbf{0},$$

because the centroid of $\overline{\mathbf{Y}}$ is at the origin. Similarly, because $\left[\mathbf{y}\right]_\times$ is a linear function of $\mathbf{y}$,

$$(\mathbf{1}\otimes\mathbf{I})^T\,\mathbf{K}\big(\overline{\mathbf{Y}}\big) = \sum\left[\overline{\mathbf{y}}_i\right]_\times = \left[\sum\overline{\mathbf{y}}_i\right]_\times = \mathbf{0}.$$

Finally, we show the relationship between $\mathrm{vec}\big(\overline{\mathbf{Y}}\big)$ and $\mathbf{K}\big(\overline{\mathbf{Y}}\big)$. Note that

$$\mathbf{K}\big(\overline{\mathbf{Y}}\big)^T\,\mathrm{vec}\big(\overline{\mathbf{Y}}\big) = \sum\left[\overline{\mathbf{y}}_i\right]_\times^T\overline{\mathbf{y}}_i = -\sum\left[\overline{\mathbf{y}}_i\right]_\times\overline{\mathbf{y}}_i. \quad (30)$$

Hence, this expression is equivalent to the sum of the exterior products of $\overline{\mathbf{y}}_i$ with themselves. Because the exterior product of a vector with itself is $\mathbf{0}$, i.e., $\mathbf{y}\wedge\mathbf{y} = \mathbf{0}$ and $\mathbf{y}\times\mathbf{y} = \mathbf{0}$ for $n_d = 3$, (30) is also $\mathbf{0}$. ■

3 Proof of Proposition 2

First of all, a linear transform of a normal distribution is also Gaussian, hence $\mathbf{Y}'$ is Gaussian. Moreover, the determinant $|\mathbf{\Sigma}_R|$ remains the same because the transform is orthogonal. Then, it is obvious that $\overline{\mathbf{Y}}' = E[\mathbf{Y}'] = \mathbf{R}E[\mathbf{Y}] = \mathbf{R}\overline{\mathbf{Y}}$ and

$$\begin{aligned}&E\left[\mathrm{vec}\big(\mathbf{Y}'-\overline{\mathbf{Y}}'\big)\,\mathrm{vec}\big(\mathbf{Y}'-\overline{\mathbf{Y}}'\big)^T\right]\\ &= (\mathbf{I}\otimes\mathbf{R})\,E\left[\mathrm{vec}\big(\mathbf{Y}-\overline{\mathbf{Y}}\big)\,\mathrm{vec}\big(\mathbf{Y}-\overline{\mathbf{Y}}\big)^T\right]\left(\mathbf{I}\otimes\mathbf{R}^T\right),\\ &= (\mathbf{I}\otimes\mathbf{R})\,\mathbf{\Sigma}\left(\mathbf{I}\otimes\mathbf{R}^T\right).\end{aligned}$$

The only thing left to prove is whether the constraints on $\mathbf{Y}'$ hold after the transformation. To show this, we substitute $\mathbf{Y} = \mathbf{R}^T\mathbf{Y}'$ in (7), i.e.,

$$\begin{aligned}&1 = \left\|\overline{\mathbf{Y}}\right\|^2 = \left\|\mathbf{R}\overline{\mathbf{Y}}\right\|^2 = \left\|\overline{\mathbf{Y}}'\right\|^2,\\ &1 = \mathrm{tr}\Big(\left(\mathbf{R}^T\mathbf{Y}'\right)\overline{\mathbf{Y}}^T\Big) = \mathrm{tr}\Big(\mathbf{Y}'\left(\mathbf{R}\overline{\mathbf{Y}}\right)^T\Big) = \mathrm{tr}\Big(\mathbf{Y}'\overline{\mathbf{Y}}'^T\Big),\\ &\mathbf{0} = \left(\mathbf{R}^T\mathbf{Y}'\right)\overline{\mathbf{Y}}^T - \overline{\mathbf{Y}}\left(\mathbf{R}^T\mathbf{Y}'\right)^T\\ &\quad = \mathbf{R}^T\left(\mathbf{Y}'\overline{\mathbf{Y}}'^T - \overline{\mathbf{Y}}'\mathbf{Y}'^T\right)\mathbf{R},\\ &\mathbf{0} = \left(\mathbf{R}^T\mathbf{Y}'\right)\mathbf{1} = \mathbf{R}^T\left(\mathbf{Y}'\mathbf{1}\right).\end{aligned}$$

Thus, the constraints for $\mathbf{Y}'$ holds after the transform.■

4 Proof of Proposition 3

First of all, the constraint $\mathbf{\Gamma}^T\mathbf{P}_N\big(\overline{\mathbf{Y}}\big) = \mathbf{0}$ does not depend on the scale of $\overline{\mathbf{Y}}$, because $\mathbf{P}_N\big(\overline{\mathbf{Y}}\big)$ is a linear function of $\overline{\mathbf{Y}}$. However, $\hat{\mathbf{R}}_i$ depends on the scale of $\overline{\mathbf{Y}}$, i.e., the first condition in (20) makes the scale of $\hat{\mathbf{R}}_i$s linearly decrease as the scale of $\overline{\mathbf{Y}}$ increases. Nevertheless, this does not affect J_2: The last term of (19) does not change due to the condition $\mathbf{\Gamma}^T\mathbf{C}'\mathbf{\Gamma} = \mathbf{I}$, even though the scale of $\mathbf{C}'$ decreases quadratically as the scale of $\hat{\mathbf{R}}_i$s decrease. However, this makes $\mathbf{\Gamma}$ increase linearly. Let us define $\hat{\mathbf{R}}'_i \triangleq \frac{1}{a}\hat{\mathbf{R}}_i$ and $\mathbf{\Gamma}' \triangleq a\mathbf{\Gamma}$, where a denote a scale. Then, J_2 becomes

$$\begin{aligned}
&2J_2\left(\{\hat{\mathbf{R}}_i'\}, \mathbf{\Gamma}'\right) \\
&= -\log\left|a^2\mathbf{\Gamma}^T\mathbf{\Gamma}\right| + \operatorname{tr}\left(\mathbf{\Gamma}^T\mathbf{C}'\mathbf{\Gamma}\right) - \frac{n_R}{3n_s}\sum\log\left|\frac{1}{a^2}\hat{\mathbf{R}}_i^T\hat{\mathbf{R}}_i\right| \\
&= -2n_R\log a + 2n_R\log a + 2J_2\left(\{\hat{\mathbf{R}}_i\}, \mathbf{\Gamma}\right) \\
&= 2J_2\left(\{\hat{\mathbf{R}}_i\}, \mathbf{\Gamma}\right). \qquad \blacksquare
\end{aligned}$$

5 Derivation of the gradient of J_2

Here, we show how to calculate each term in (23). $\partial J_2/\partial\mathbf{\Gamma}$ can be easily found by differentiating the cost function as

$$\frac{\partial J_2}{\partial\mathbf{\Gamma}} = \mathbf{\Gamma}^T\mathbf{C}' - \mathbf{\Gamma}^+.$$

However, finding $\partial\mathbf{\Gamma}/\partial\overline{\mathbf{Y}}$ is a bit tricky. By differentiating $\mathbf{\Gamma}^T\mathbf{P}_N(\overline{\mathbf{Y}}) = \mathbf{0}$, we have

$$\partial\mathbf{\Gamma}^T\mathbf{P}_N + \mathbf{\Gamma}^T\partial\mathbf{P}_N = \mathbf{0}. \tag{31}$$

Hence, $\partial\mathbf{\Gamma}/\partial\overline{y}_{jk}$ ($\overline{y}_{jk}$ is the (j,k)th element of $\overline{\mathbf{Y}}$) must be

$$\frac{\partial\mathbf{\Gamma}}{\partial\overline{y}_{jk}} = -\mathbf{P}_N^{+T}\mathbf{E}_{jk}^T\mathbf{\Gamma} + \mathbf{\Gamma}\mathbf{G}_{jk},$$

where $\mathbf{E}_{jk} = \partial\mathbf{P}_N/\partial\overline{y}_{jk}$ and $\mathbf{G}_{jk}$ is an appropriate matrix. The last term was added because the components of $\partial\mathbf{\Gamma}$ that are orthogonal to $\mathbf{P}_N$ are cancelled in (31), i.e., $\mathbf{P}_N^T\mathbf{\Gamma}\mathbf{G}_{jk} = \mathbf{0}$. Since $\mathbf{P}_N$ can be seen as a rearranged version of $\overline{\mathbf{Y}}$, $\mathbf{E}_{jk}$ is a sparse constant matrix that is filled with $\{1, 0, -1\}$, and can be expressed as

$$\mathbf{E}_{jk} = (\mathbf{e}_k \otimes \mathbf{I})\begin{bmatrix}\mathbf{E}_j & \mathbf{0}\end{bmatrix},$$

where $\mathbf{e}_k$ is the kth n_p-dimensional elementary (unit) vector, i.e., its element is one for the kth element and zero for the others. Here, $\mathbf{0}$ is an $n_d \times n_d$ zero matrix, and $\mathbf{E}_j$ is an $n_d \times (n_N - n_d)$ (=3×4) matrix that is described as

$$\mathbf{E}_1 = \begin{bmatrix}1 & 0 & 0 & 0\\ 0 & 0 & 0 & -1\\ 0 & 0 & 1 & 0\end{bmatrix}, \quad \mathbf{E}_2 = \begin{bmatrix}0 & 0 & 0 & 1\\ 1 & 0 & 0 & 0\\ 0 & -1 & 0 & 0\end{bmatrix},$$
$$\mathbf{E}_3 = \begin{bmatrix}0 & 0 & -1 & 0\\ 0 & 1 & 0 & 0\\ 1 & 0 & 0 & 0\end{bmatrix}.$$

Note that the last three columns of $\mathbf{E}_k$, and so those of $\mathbf{E}_{jk}$ or $\partial\mathbf{P}_N$, are all zeros because the corresponding elements of $\mathbf{P}_N$ are constants.

In practice, $\mathbf{G}_{jk}$ does not need to be evaluated, because

$$\begin{aligned}
&\left\langle\frac{\partial J_2}{\partial\mathbf{\Gamma}}, \frac{\partial\mathbf{\Gamma}}{\partial\overline{y}_{jk}}\right\rangle \\
&= \operatorname{tr}\left(\left(\mathbf{\Gamma}^T\mathbf{C}' - \mathbf{\Gamma}^+\right)\left(-\mathbf{P}_N^{+T}\mathbf{E}_{jk}^T\mathbf{\Gamma} + \mathbf{\Gamma}\mathbf{G}_{jk}\right)\right) \\
&= \operatorname{tr}\left(-\mathbf{\Gamma}^T\mathbf{C}'\mathbf{P}_N^{+T}\mathbf{E}_{jk}^T\mathbf{\Gamma} + \left(\mathbf{\Gamma}^T\mathbf{C}'\mathbf{\Gamma} - \mathbf{I}\right)\mathbf{G}_{jk}\right) \\
&= \operatorname{tr}\left(-\mathbf{\Gamma}^T\mathbf{C}'\mathbf{P}_N^{+T}\mathbf{E}_{jk}^T\mathbf{\Gamma}\right) = -\operatorname{tr}\left(\mathbf{E}_{jk}\mathbf{P}_N^+\mathbf{C}'\mathbf{\Sigma}^+\right),
\end{aligned}$$

and $\mathbf{G}_{jk}$ is eliminated in the above equation. Note that $\mathbf{P}_N^+\mathbf{C}'\mathbf{\Sigma}^+$ does not depends on j or k, and multiplying $\mathbf{E}_{jk}$ only changes the order and the signs of rows. Therefore, we can analytically find this part of the gradient by a few matrix multiplications.

The derivatives related to $\hat{\mathbf{R}}_i'$ can be similarly calculated. To calculate $\partial J_2/\partial\hat{\mathbf{R}}_i'$, $\operatorname{tr}\left(\mathbf{\Gamma}^T\mathbf{C}'\mathbf{\Gamma}\right)$ can be rearranged as

$$\begin{aligned}
\operatorname{tr}\left(\mathbf{\Gamma}^T\mathbf{C}'\mathbf{\Gamma}\right) &= \sum_i \operatorname{tr}\left(\left(\mathbf{I}\otimes\hat{\mathbf{R}}_i\right)\mathbf{C}_i'\left(\mathbf{I}\otimes\hat{\mathbf{R}}_i^T\right)\mathbf{\Gamma}\mathbf{\Gamma}^T\right) \\
&= \frac{1}{n_s}\sum_i\sum_{j,k}\operatorname{tr}\left(\hat{\mathbf{R}}_i\mathbf{C}_{i,k,j}'\hat{\mathbf{R}}_i^T\mathbf{\Xi}_{j,k}\right),
\end{aligned}$$

where $\mathbf{C}_{i,j,k}'$ and $\mathbf{\Xi}_{j,k}$ are the (j,k)th 3×3 block of $\mathbf{C}_i'$ and $\mathbf{\Gamma}\mathbf{\Gamma}^T$, respectively. Therefore,

$$\frac{\partial J_2}{\partial\hat{\mathbf{R}}_i} = \frac{1}{n_s}\left(-\frac{n_R}{3}\hat{\mathbf{R}}_i^{-1} + \sum_{j,k}\mathbf{C}_{i,k,j}'\hat{\mathbf{R}}_i^T\mathbf{\Xi}_{j,k}\right).$$

However, $\hat{\mathbf{R}}_i$ has a special structure satisfying the relation $\hat{\mathbf{R}}_i^T\hat{\mathbf{R}}_i = 2c_i\mathbf{I}$. Differentiating this constraint yields,

$$\partial\hat{\mathbf{R}}_i^T\hat{\mathbf{R}}_i + \hat{\mathbf{R}}_i^T\partial\hat{\mathbf{R}}_i = 2\partial c_i\mathbf{I}.$$

Thus, similar to (10), $\partial\hat{\mathbf{R}}_i$ can be expressed in terms of a four-dimensional vector $\hat{\mathbf{r}}_i = \begin{bmatrix}c_i & r_{ix} & r_{iy} & r_{iz}\end{bmatrix}^T$ as

$$\partial\hat{\mathbf{R}}_i\Big|_{\hat{\mathbf{R}}_i=\mathbf{I}} = \begin{bmatrix}\partial c_i & \partial r_{iz} & -\partial r_{iy}\\ -\partial r_{iz} & \partial c_i & \partial r_{ix}\\ \partial r_{iy} & -\partial r_{ix} & \partial c_i\end{bmatrix}.$$

This can be alternatively described in a vectorized form as $\operatorname{vec}\left(\partial\hat{\mathbf{R}}_i\right)\Big|_{\hat{\mathbf{R}}_i=\mathbf{I}} = \mathbf{L}\partial\hat{\mathbf{r}}_i$ with a $n_d^2 \times (n_N - n_d)$ (= 9×4) matrix $\mathbf{L}$;

$$\mathbf{L} = \begin{bmatrix}\mathbf{E}_1^T & \mathbf{E}_2^T & \mathbf{E}_3^T\end{bmatrix}^T.$$

Note that $\mathbf{L}$ satisfies the relation

$$\left(\mathbf{\Theta}^T\otimes\mathbf{I}\right)\mathbf{L} = \begin{bmatrix}\operatorname{vec}(\mathbf{\Theta}) & \mathbf{K}(\mathbf{\Theta})\end{bmatrix} = \mathbf{P}_N'(\mathbf{\Theta}),$$

where $\mathbf{P}_N'(\mathbf{\Theta})$ is the first four columns of $\mathbf{P}_N(\mathbf{\Theta})$, for any matrix $\mathbf{\Theta}$ consisting of three rows. Finally, we have the following derivative:

$$\begin{aligned}
\frac{\partial J_2}{\partial\hat{\mathbf{r}}_i}\Big|_{\hat{\mathbf{R}}_i=\mathbf{I}} &= \operatorname{vec}\left(\frac{\partial J_2}{\partial\hat{\mathbf{R}}_i}^T\right)^T\Bigg|_{\hat{\mathbf{R}}_i=\mathbf{I}}\mathbf{L} \\
&= \frac{1}{n_s}\operatorname{vec}(\mathbf{\Omega}_i)^T\mathbf{L},
\end{aligned}$$

for a 3×3 matrix $\mathbf{\Omega}_i \triangleq -\frac{n_R}{3}\mathbf{I} + \sum_{j,k}\mathbf{\Xi}_{k,j}\mathbf{C}_{i,j,k}'$.

Now, let us assume that $\mathbf{M}_i\overline{\mathbf{Y}}^T$ is full-rank so that $\hat{\mathbf{R}}_i\mathbf{M}_i\overline{\mathbf{Y}}^T$ is positive definite. Then, the PSD condition in (20) is equivalent to a symmetry condition, i.e.,

$$\hat{\mathbf{R}}_i\mathbf{M}_i\overline{\mathbf{Y}}^T = \overline{\mathbf{Y}}\mathbf{M}_i^T\hat{\mathbf{R}}_i^T,$$

for a neighborhood of current $\hat{\mathbf{R}}_i$. Then, the first two conditions in (20) can be expressed similarly as in (8), i.e.,

$$\mathbf{P}_N'^T\operatorname{vec}\left(\hat{\mathbf{R}}_i\mathbf{M}_i\right) = \begin{bmatrix}1 & 0 & 0 & 0\end{bmatrix}^T,$$

TABLE 1
Average reconstruction errors w/o noise

data	EM-PPCA	MP	EM-GPA		SPM	CSF2	EM-PND		
			v1	v2			[2]	init.	prop.
FRGC	.1469	.1395	.0828	.0842	.1094	.1926	**.0727**	.1035	.0731
walking	.1485	.2699	.1015	.0593	.0861	.0708	.0465	.0982	**.0407**
shark	.0688	.0874	.0590	.0455	.1670	.0551	**.0135**	.0809	.0272
face	.0209	.0331	.0203	.0218	.0233	.0209	.0165	.0190	**.0150**
yoga	.6100	.5924	.1095	.0377	.0224	.0225	.0140	.1201	**.0128**
stretch	.5392	.5915	.6717	.0383	.0288	.0219	.0156	.1085	**.0150**
pickup	.5149	.3465	.5331	.0356	.0356	.0607	.0372	.1447	**.0133**
drink	.1292	.2650	.1179	.0197	.0216	.0123	.0037	.0424	**.0031**
dance	.2325	.4062	.2671	.1381	.1472	.1350	.1834	.1793	**.1247**

TABLE 2
Average reconstruction errors w/ noise

data	EM-PPCA	MP	EM-GPA		SPM	CSF2	EM-PND		
			v1	v2			[2]	init.	prop.
FRGC	.1980	.1400	.1139	.1130	.1767	.2061	**.0889**	.1278	.0891
walking	.1368	.3231	.1205	.0988	.1050	.0966	.0770	.1238	**.0769**
shark	**.0486**	.1192	.0929	.0862	.1697	.1043	.0600	.1046	.0586
face	.0464	.0524	.0866	.1105	.1321	.0543	**.0403**	.0764	.0421
yoga	.5287	.6117	.1240	.0747	.0822	.0529	**.0409**	.1283	**.0409**
stretch	.5479	.5738	.6503	.0750	.0652	.0543	**.0444**	.1217	**.0444**
pickup	.5037	.3674	.5517	.0740	.0581	.0705	.0409	.1513	**.0405**
drink	.1768	.2678	.1341	.0571	.0407	.0365	**.0339**	.0704	**.0339**
dance	.2229	.4131	.2778	.1688	.1510	.1544	.1806	.1929	**.1356**

TABLE 3
Average camera motion errors (degree)

data	EM-PPCA	MP	EM-GPA		SPM	CSF2	EM-PND		
			v1	v2			[2]	init.	prop.
yoga	39.552	41.608	7.571	2.200	0.816	0.811	0.590	3.921	**0.498**
stretch	43.527	43.454	46.444	2.539	1.125	0.717	0.673	4.745	**0.593**
pickup	30.095	21.991	29.407	1.823	1.483	2.263	1.691	5.040	**0.449**
drink	7.097	15.957	6.773	1.264	1.132	0.449	0.145	1.818	**0.126**

or

$$\mathbf{P}_N'^T \left(\mathbf{M}_i^T \otimes \mathbf{I}\right) \mathrm{vec}\left(\hat{\mathbf{R}}_i\right) = \begin{bmatrix}1 & 0 & 0 & 0\end{bmatrix}^T,$$

in terms of $\mathrm{vec}\left(\hat{\mathbf{R}}_i\right)$. By differentiating this equation and using the relations $\hat{\mathbf{R}}_i = \mathbf{I}$ and $\mathrm{vec}\left(\partial\hat{\mathbf{R}}_i\right)\Big|_{\hat{\mathbf{R}}_i=\mathbf{I}} = \mathbf{L}\partial\hat{\mathbf{r}}_i$, we obtain

$$\partial\mathbf{P}_N'\left(\overline{\mathbf{Y}}\right)^T \mathbf{m}_i + \mathbf{P}_N'\left(\overline{\mathbf{Y}}\right)^T \mathbf{P}_N'(\mathbf{M}_i)\partial\hat{\mathbf{r}}_i = \mathbf{0}.$$

If we denote $\mathbf{\Psi}_i \triangleq \mathbf{P}_N'\left(\overline{\mathbf{Y}}\right)^T \mathbf{P}_N'(\mathbf{M}_i)$, we have

$$\left.\frac{\partial\hat{\mathbf{r}}_i}{\partial\overline{y}_{jk}}\right|_{\hat{\mathbf{R}}_i=\mathbf{I}} = -\mathbf{\Psi}_i^{-1}\breve{\mathbf{E}}_{jk}^T\mathbf{m}_i,$$

where $\breve{\mathbf{E}}_{jk} = (\mathbf{e}_k \otimes \mathbf{I})\mathbf{E}_j$ is the first four columns of $\mathbf{E}_{jk}$, which are non-zero.

Hence, we can calculate

$$\left\langle \frac{\partial J_2}{\partial\hat{\mathbf{r}}_i}, \frac{\partial\hat{\mathbf{r}}_i}{\partial\overline{y}_{jk}} \right\rangle = -\frac{1}{n_s}\,\mathrm{vec}(\mathbf{\Omega}_i)^T\,\mathbf{L}\mathbf{\Psi}_i^{-1}\breve{\mathbf{E}}_{jk}^T\mathbf{m}_i. \qquad (32)$$

Note that (32) is also found analytically. Finally, substituting all the derivatives to (23) yields (24).

6 Proof of Proposition 4

The condition $(\partial J_2/\partial\overline{\mathbf{Y}})\mathbf{1} = \mathbf{0}$ can be written for each element as

$$0 = -\sum_k \frac{\partial J_2}{\partial\overline{y}_{jk}} =$$

$$\sum_k \left(\mathrm{tr}\left(\mathbf{E}_{jk}\mathbf{P}_N^+\mathbf{C}'\mathbf{\Sigma}^+\right) + \frac{1}{n_s}\sum_i \mathrm{vec}(\mathbf{\Omega}_i)^T\,\mathbf{L}\mathbf{\Psi}_i^{-1}\breve{\mathbf{E}}_{jk}^T\mathbf{m}_i \right).$$

Since

$$\sum_k \mathbf{E}_{jk} = \sum_k (\mathbf{e}_k \otimes \mathbf{I})\mathbf{E}_j = (\mathbf{1} \otimes \mathbf{I})\mathbf{E}_j,$$

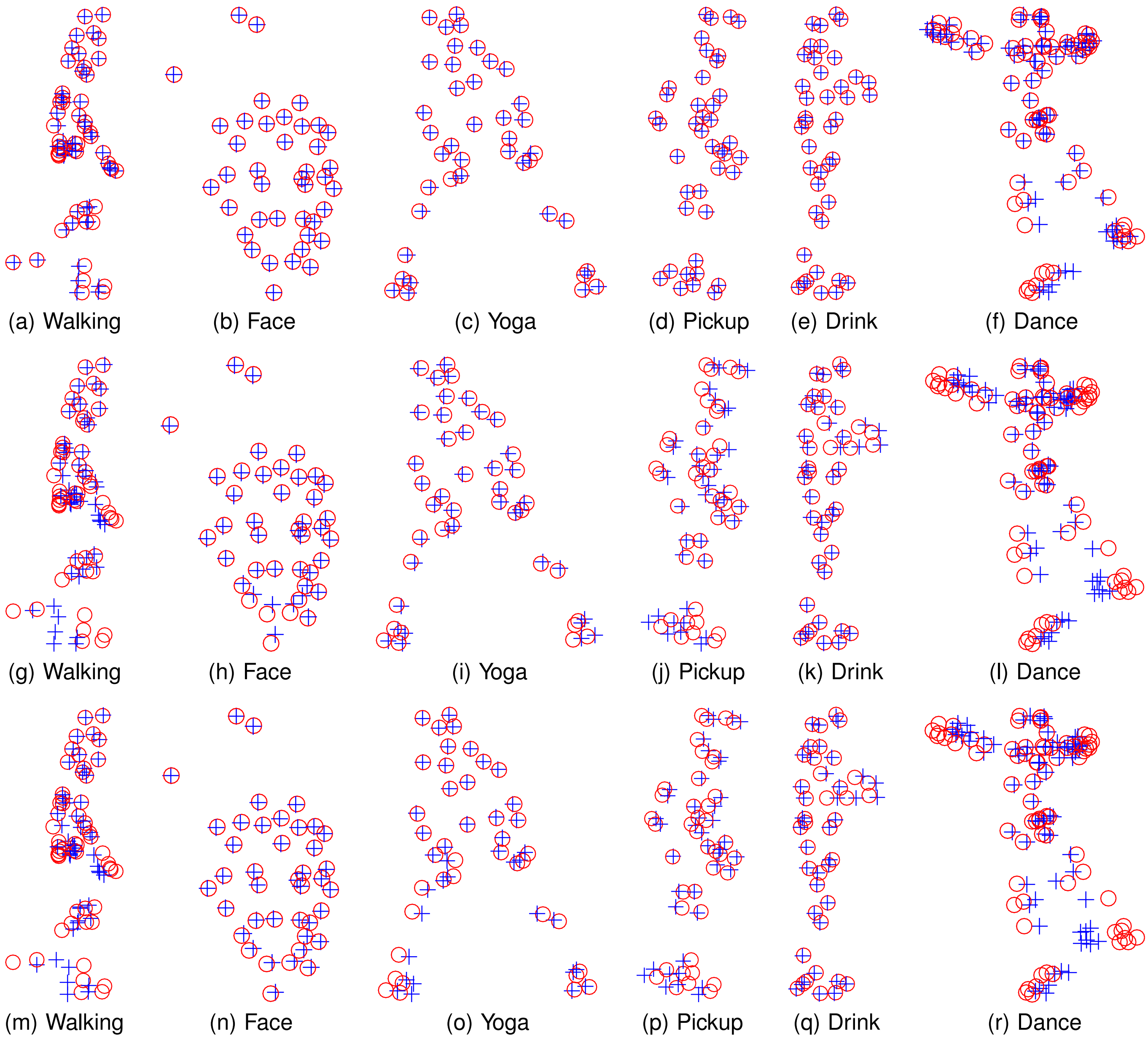


Fig. 1. Reconstructed results of benchmark datasets (top row: EM-PND, middle row: CSF2, bottom row: SPM, $\circ$: ground truth, $+$: reconstructed points).

$\sum_k \mathrm{tr}\big(\mathbf{E}_{jk}\mathbf{P}_N^{+}\mathbf{C}'\mathbf{\Sigma}^{+}\big)$ is zero, i.e.,

$$\mathrm{tr}\left(\left(\sum_k \mathbf{E}_{jk}\right)\mathbf{P}_N^{+}\mathbf{C}'\mathbf{\Sigma}^{+}\right) = \mathrm{tr}\big((\mathbf{1}\otimes\mathbf{I})\mathbf{E}_j\mathbf{P}_N^{+}\mathbf{C}'\mathbf{\Sigma}^{+}\big)$$
$$= \mathrm{tr}\big(\mathbf{E}_j\mathbf{P}_N^{+}\mathbf{C}'\mathbf{\Sigma}^{+}(\mathbf{1}\otimes\mathbf{I})\big) = \mathbf{0}.$$

Likewise,

$$\sum_k \breve{\mathbf{E}}_{jk} = (\mathbf{1}\otimes\mathbf{I})\breve{\mathbf{E}}_j$$

where $\breve{\mathbf{E}}_j$ is the first four columns of $\mathbf{E}_j$, and

$$\sum_k \mathrm{vec}(\mathbf{\Omega}_i)^T\,\mathbf{L}\mathbf{\Psi}_i^{-1}\breve{\mathbf{E}}_{jk}^T\mathbf{m}_i$$
$$= \sum_k \mathrm{vec}(\mathbf{\Omega}_i)^T\,\mathbf{L}\mathbf{\Psi}_i^{-1}\breve{\mathbf{E}}_j^T(\mathbf{1}^T\otimes\mathbf{I})\mathbf{m}_i = \mathbf{0}.$$

This concludes the proof. ■

7 Cost Function for Initial Rotations

Here, we explain the cost function for calculating the initial rotations in Section 3.5 of the main document. The shape-basis-based NRSfM approaches [3] usually assume that the 2D shape matrix $\mathbf{W}\in\mathbb{R}^{2n_s\times n_p}$ can be decomposed into $\mathbf{W}=\mathbf{R}(\mathbf{C}\otimes\mathbf{I})\mathbf{S}$, where $\mathbf{R}\in\mathbb{R}^{2n_s\times 3n_s}$ is a block diagonal matrix whose 2×3 diagonal blocks are Stiefel matrices, $\mathbf{C}\in\mathbb{R}^{n_s\times K}$ is the coefficient matrix, and $\mathbf{S}\in\mathbb{R}^{3K\times n_p}$ is the shape-basis matrix. Recent approaches suggest to decompose $\mathbf{W}=\mathbf{\Pi}\mathbf{S}'$ based on SVD, and find a matrix $\mathbf{G}\in\mathbb{R}^{3K\times 3}$ that makes the 2×3 submatrices of $\mathbf{\Pi}\mathbf{G}$ Stiefel matrices multiplied by some scalars.

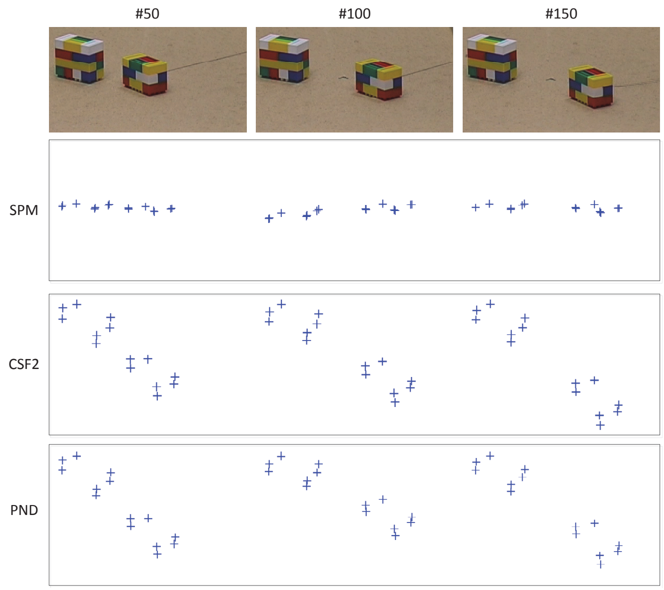


Fig. 2. Reconstruction of the cube sequence. (Blue crosses are the reconstructed points seen from a viewpoint different from the camera view.)

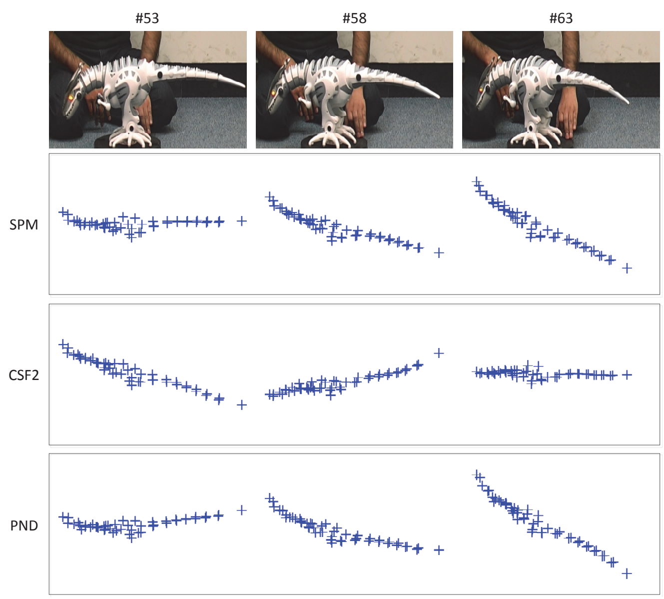


Fig. 3. Reconstruction of the dinosaur sequence.

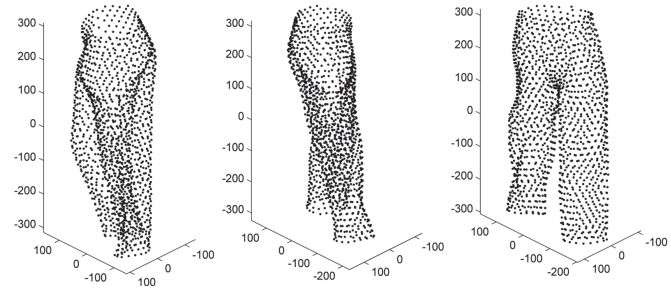
In [4], the shape deformation was assumed to have a dominant DC term, hence, they found $\mathbf{G}$ that makes the submatrices of $\mathbf{\Pi G}$ Stiefel matrices (without any scalar factors). The solution was found by minimizing the errors, i.e.,

$$\sum \|\mathbf{\Pi}_i \mathbf{G}\mathbf{G}^T \mathbf{\Pi}_i^T - \mathbf{I}\|^2,$$

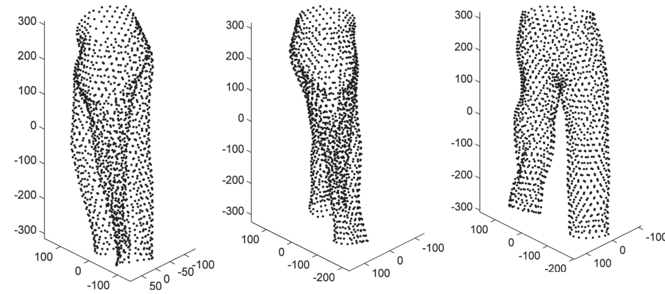
where $\mathbf{\Pi}_i$ is the ith pair of rows in $\mathbf{\Pi}$. If we do not have the DC term assumption, we can instead minimize

$$\sum \|\mathbf{\Pi}_i \mathbf{G}\mathbf{G}^T \mathbf{\Pi}_i^T - a_i\mathbf{I}\|^2,$$

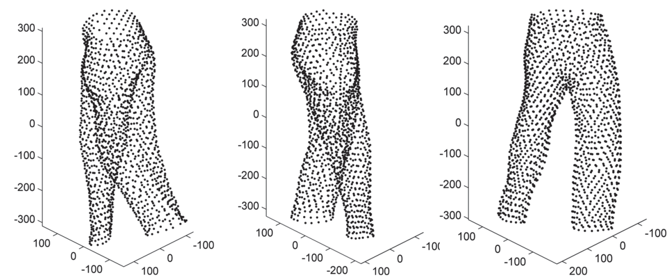
where a_i is a scalar variable. We can easily verify that



(a) Ground Truth



(b) EM-PND (Average reconstruction error: 0.0609)



(c) CSF2 (Average reconstruction error: 0.0997)

Fig. 4. Reconstruction of the pace sequence.

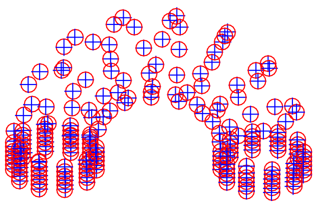
minimizing this cost function is identical to minimizing

$$\sum \left\| \begin{bmatrix} 1 & 0 & 0 & -1 \\ 0 & 1 & 1 & 0 \end{bmatrix} \mathrm{vec}\left(\mathbf{\Pi}_i \mathbf{G}\mathbf{G}^T \mathbf{\Pi}_i^T\right) \right\|^2,$$

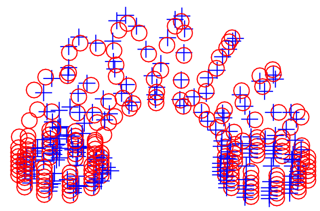
which was used for initializing the rotations of the proposed method. Note that this corresponds to the approach in [5].

8 Detailed results of experiments

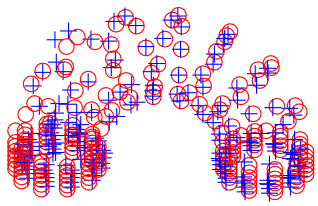
Tables 1 – 3 show the error including the early NRSfM schemes, EM-PPCA and MP. Here, we can see that EM-PPCA gives smallest error for the shark sequence with noise, which seems to be attributed due to the nature of the shark sequence that was artificially generated by superposing two basis shapes [6]. Because of this, EM-PPCA, which explicitly limits the number of shape bases in the reconstruction process, gives better performance for the shark data. However, this is hardly the case for the real-world applications. In the other cases, EM-PPCA and MP usually show very poor performance. The accuracy of the initial shape is not as good as the recent state of the art, but is better than those of the early NRSfM schemes, i.e., EM-PPCA and MP, so this



(a) EM-PND (Average error: 0.1349)



(b) CSF2 ($K = 4$, Average error: 0.1668)



(c) SPM ($K = 5$, Average error: 0.1621)

Fig. 5. Reconstruction examples of the synthetic spring toy sequence. ($\circ$: ground truth, $+$: reconstructed points)

can be effectively used for many NRSfM schemes for initialization. Some of the reconstruction examples for the benchmark sequences are shown in Figure 1. Here, EM-PND shows a better fit between the reconstructed points and the corresponding ground truth than CSF2 and SPM.

SPM, CSF2, and EM-PND have been tested for the real data sequences used in [4], which were the cube and the dinosaur sequences. Since there are no ground truth for these data sequences, Figures 2 and 3 are shown for qualitative evaluation. For the cube sequence, SPM gave flat reconstruction implying that the reconstruction was not successful. For the dinosaur sequence, the results of CSF2 show wrong directions. On the other hand, the reconstruction of EM-PND shows good result for each frame of the sequences.

Figure 4 shows some examples of the reconstructed frames of the pace sequence. Here, we can see that EM-PND gives better reconstruction results than CSF2. Although there are some minor differences between the reconstruction and the ground truth, the results are good enough for dense 3D recovery.

We have also tested a synthetic data whose deformation lies in a nonlinear manifold. Similar to the spring toy data in [7], we have generated a synthetic data of a spring toy, of which the deformation is nonlinear. The parameters of SPM and CSF2 have been tuned to yield the best performance. Figure 5 shows the average errors and the reconstructed examples. Here, we can see that EM-PND gives the best performance, even though it does not require any tuning parameter. Although a Gaussian distribution can only be defined on a linear manifold, the adaptive nature of EM-PND seems to be the cause of such better performance, according to the interpretation of Section 3.4 of the main document.

References

[1] J. C. Gower, "Generalized Procrustes Analysis," *Psychometrika*, vol. 40, no. 1, pp. 33–51, March 1975.

[2] M. Lee, J. Cho, C.-H. Choi, and S. Oh, "Procrustean normal distribution for non-rigid structure from motion," in *Proc. Computer Vision and Pattern Recognition*, June 2013.

[3] I. Matthews, J. Xiao, and S. Baker, "2D vs. 3D Deformable Face Models: Representational Power, Construction, and Real-Time Fitting," *Int'l J. Computer Vision*, vol. 75, no. 1, pp. 93–113, October 2007.

[4] I. Akhter, Y. Seikh, S. Khan, and T. Kanade, "Trajectory Space: A Dual Representation for Nonrigid Structure from Motion," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 33, no. 7, pp. 1442–1456, July 2011.

[5] Y. Dai, H. Li, and M. He, "A Simple Prior-free Method for Non-Rigid Structure-from-Motion Factorization," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, June 2012.

[6] L. Torresani, A. Hertzmann, and C. Bregler, "Nonrigid Structure-from-Motion: Estimating Shape and Motion with Hierarchical Priors," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 30, no. 5, pp. 878–892, May 2008.

[7] V. Rabaud and S. Belongie, "Re-thinking non-rigid structure from motion," in *Proc. Computer Vision and Pattern Recognition*, June 2008.