[Supplementary] Organic Priors in Non-Rigid Structure from Motion

Suryansh Kumar[1] sukumar@vision.ee.ethz.ch
Luc Van Gool[1,2] vangool@vision.ee.ethz.ch

[1]ETH Zürich, [2]KU Leuven

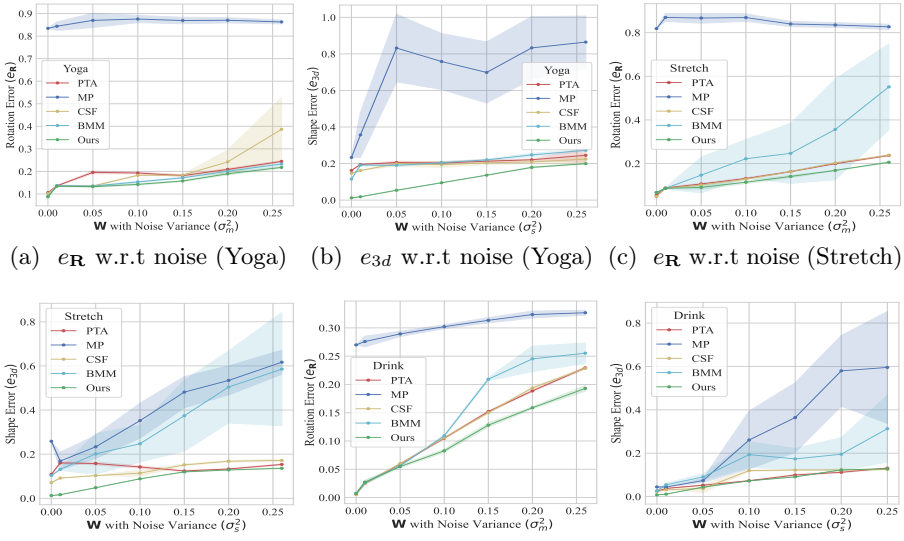
Abstract. This draft provides extra statistical results, comparison, ablations, mathematical derivations, remaining details, and discussions to further support the claims made in the main paper. Continuing with the results presented in the main paper, we first provide remaining results on the noisy sequence. Next, we supply a more comprehensive evaluation of our method on the NRSfM challenge dataset [17]. Additionally, we show the contribution of single rotation averaging to the overall accuracy of our approach on the CMU MoCap dataset [3]. Finally, we end the draft with timing details, more visual results, theory with discussion on the rank optimization methods, and future areas for research in this direction of work.

1 More Experiments and Analysis

a) Performance on the noisy trajectory. In the main paper, we provided the statistical comparison results for the noisy Pickup trajectory sequence. Results for the remaining Akther *et al.* [3] sequence *i.e.*, Yoga, Stretch and Drink sequence, with the noisy trajectories are shown in Fig.(1). We observed that our method handles the noise cases efficiently and generally outperform other competing methods.

b) Comprehensive results on the NRS*f*M benchmark dataset. Although the visual results are provided in the main paper, we added this qualitative 3D shape reconstruction results here for completeness. Fig.(2) show the visual outcome obtained on [17] dataset using our method, for the test frame. As demonstrated in Jensen *et al.* [17] work that camera trajectory type can have a noticeable difference in NRS*f*M algorithm's performance. Even though the overall average reconstruction error per subject is provided in the main paper, it may not reflect the true behavior of the algorithm. Consequently, we provide a more comprehensive statistical evaluation. Table (1) shows the quantitative results obtained using our method over different types of camera motion. All in all, Flyby and Circle camera trajectory gives better performance; however, results with Tricky camera trajectory type are a bit off than other categories. Nevertheless, our method's performance pattern is consistent with the observation made in the Jensen *et al.* [17] recent article on other state-of-the-art NRS*f*M methods.

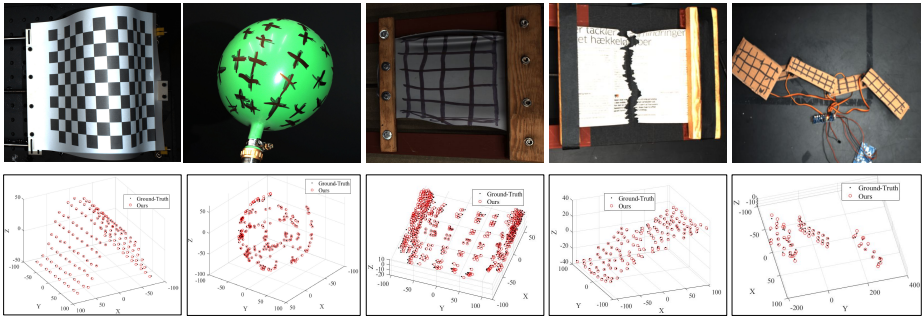
c) Performance Comparison with GT rotation. We analyzed the fidelity of our estimated rotation against the ground-truth rotation. For that, we first



(a) $e_{\mathbf{R}}$ w.r.t noise (Yoga) (b) e_{3d} w.r.t noise (Yoga) (c) $e_{\mathbf{R}}$ w.r.t noise (Stretch)

(d) e_{3d} w.r.t noise(Stretch) (e) $e_{\mathbf{R}}$ w.r.t noise (Drink) (f) e_{3d} w.r.t noise (Drink)

Fig. 1: **Top to bottom row:** Rotation and shape error on the noisy yoga, stretch and drink sequence trajectories, respectively. Consistently, our method show stable behaviour and provide satisfactory results. Mean and standard deviation is shown with bold and corresponding light shaded region, respectively.



PAPER BALLOON STRETCH TEARING ARTICULATED

Fig. 2: Qualitative Results on NRSfM challenge dataset [17]. **Top row.** Subject image. **Bottom row.** 3D reconstruction of the respective object shape.

Dataset ↓ / Camera Trajectory Type →	Circle	Flyby	Line	Semi-Circle	Tricky	Zigzag	**Avg. on Deformation Type (over rows)**
Articulated	11.5054	3.7178	12.9399	6.5405	21.9694	16.4351	**12.1847**
Balloon	1.3584	2.5427	10.8937	2.8792	18.5155	1.5760	**6.2942**
Paper	1.6968	4.3147	15.6295	8.5301	21.6723	1.3176	**8.8602**
Stretch	2.5105	3.4801	4.1066	2.7418	21.9850	3.3715	**6.3659**
Tearing	2.1395	2.5594	15.9419	3.3603	35.3746	6.1068	**10.9137**
Avg. on Trajectory Type (over columns)	**3.8421**	**3.3229**	**11.9023**	**4.8104**	**23.9034**	**5.7614**	

Table 1: Object 3D shape reconstruction error reported in millimeters (mm) over different camera trajectory type. The result corresponds to the test example supplied with the dataset [17]. The results indicate that our method can efficiently handle different kinds of shape deformation with distinct camera trajectory. Also, we can observe that our evaluation over the different camera trajectory aligns well the independent study conducted by Jensen *et al.* [17].

considered the 3D reconstruction accuracy (e_{3d}) of the pseudo-inverse inverse solution *i.e.*, $\mathbf{X}_{init} = \texttt{pinv}(\mathbf{R})\mathbf{W}$, using ground-truth (GT) rotation and our estimated rotation, respectively. Table (2) provides the statistical comparison for the same. It is easy to infer that the two results are almost equivalent. For us, such results were *surprising* as well as encouraging.

Next, we studied the performance our shape optimization proposed in the main paper with different rotation ($\mathbf{R}$) initialization *(i)* our rotation (Ours) *(ii)* with GT rotation (Ours$^\sharp$). Table (3) provide the statistical comparison for the same. The numbers suggest that our shape optimization performs better than other competing methods. Besides, using our estimated rotation we can have 3d reconstruction results which is as good as if ground-truth rotation were used, and the difference between the two is insignificant.

Dataset	Drink	Pickup	Yoga	Stretch
GT-PI	0.2195	0.2984	0.2739	0.2236
Ours-PI	0.2195	0.2985	0.2740	0.2238

Table 2: Pseudo inverse (PI) 3d reconstruction error results with GT (PI-GT) and ours rotation. Ours results are very close to pseudo-inverse solution result that can be recovered with ground-truth rotation.

Data	PTA [3]	CSF2 [11]	BMM [8]	Ours	Ours$^\sharp$
Yoga	0.0580	0.0215	0.0334	**0.0122**	**0.0119**
Pickup	0.0992	0.0814	0.0497	**0.0152**	**0.0139**
Stretch	0.0822	0.0442	0.0456	**0.0124**	**0.0121**

Table 3: Quantitative performance comparison using (e_{3d}) metric given ground-truth rotation. Ours, Ours$^\sharp$ show the statistics with our estimated rotation initialization and ground-truth rotation initialization results, respectively.

d) Effect of using our organic prior rotation compared to prior-free rotation. We conducted this extra experiment to show the contribution of our rotation estimation to the overall deforming shape 3D reconstruction. To effectively show the outcome, we conduct this experiment in two settings *(i)* We introduce Dai *et al.* rotation to our shape optimization *(ii)* We introduce our computed rotation to the proposed shape optimization. Table (4) show the result of this ablation. Clearly, our shape optimization provides outstanding results. Additionally, use our rotation approach, we can further enhance our results.

Dataset	Drink	Pickup	Yoga	Stretch	Dance	Face	Walking	Shark
BMM [8]	0.0266	0.1731	0.1150	0.1034	0.1864	0.0303	0.1298	0.2357
e_{3d} ([8] rotation)	0.0101	0.0164	0.0126	0.0126	0.1382	0.0152	0.0880	0.0563
e_{3d} (our rotation)	**0.0071**	**0.0152**	**0.0122**	**0.0124**	**0.1209**	**0.0145**	**0.0816**	**0.0550**

Table 4: Showing the contribution of our rotation averaging approach to overall performance on noise-free sequence. **Second row.** e_{3d} metric obtained using our shape optimization with Dai *et al.* method's rotation [8]. **Third row.** e_{3d} metric using our shape optimization results with our method's rotation. Indeed using intermediate rotation priors (organic prior) help improve the performance and works incredibly well on the noisy sequence, as shown in the main paper.

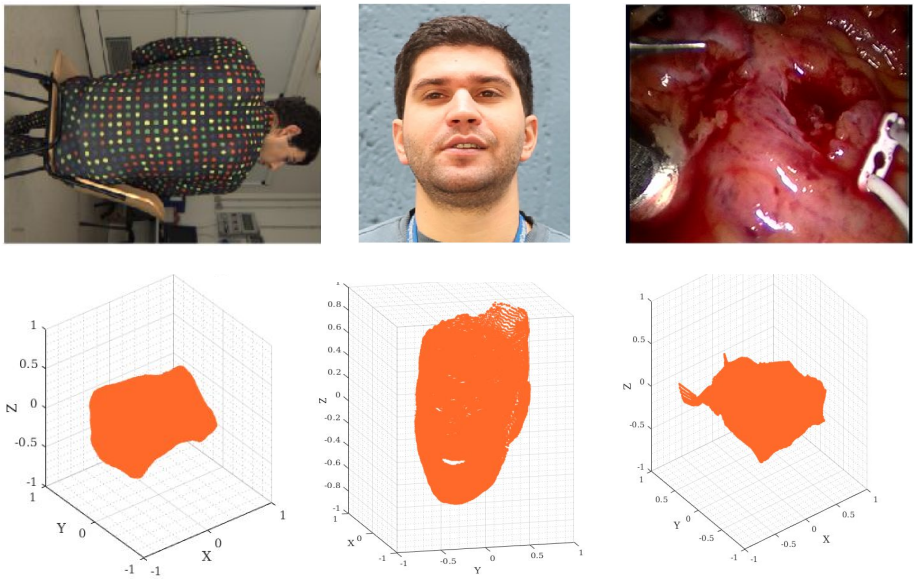
e) Timing Details. To provide a clear indication that we don't sacrifice much on the computation time as compared to BMM rotation [8] estimation, we evaluated the processing time of our algorithm. The study indicate that our method's computation time compares favorably with [8]. The reasons for that are *(i)* recovering column triplet and corresponding rotation can be run in parallel as each column triplet is processed independent of each other, and *(ii)* L_1 single rotation averaging algorithm is computationally fast [14], and with just 50 iterations of it for every frame gives good enough results. Table (5) provides the processing time comparison with [8] for both the parallel and sequential implementation of our method. Parallel implementation is used solely for the purpose of K parallel computation of $\mathbf{G}^k \in \mathbb{R}^{3K\times 3}$ and corresponding $\mathbf{R}^k \in \mathbb{R}^{2F\times 3}$. The supplied computation time is observed on MATLAB (R2020b)/C++ running on a desktop machine with Linux OS and 32GB RAM. For all of the CMU MoCap sequence [3, 32], our computation time is slightly more than the BMM [8]. Yet, there is an exception *for e.g.,* for the `shark` sequence our method gives better timing performance. That is because our shape optimization converges faster than BMM [8] shape optimization, at the same time, the value of K and number of frames for this example is relatively small.

Dataset	Drink	Pickup	Yoga	Stretch	Dance	Face	Walking	Shark
BMM [8]	89.7040 ± 0.12	50.0569 ± 0.35	25.3601 ± 0.24	34.0517 ± 0.56	61.6082 ± 0.22	16.3589 ± 0.26	26.9831 ± 0.29	52.6904 ± 0.35
Our Method (P)	123.1312 ± 0.39	67.3095 ± 0.58	45.2626 ± 0.42	50.3835 ± 0.49	84.7192 ± 0.21	20.1009 ± 0.78	33.5214 ± 0.22	5.7592 ± 0.31
Our Method (S)	536.9051 ± 0.63	460.0693 ± 0.19	143.8776 ± 0.52	265.8589 ± 0.13	146.7546 ± 0.99	24.2315 ± 0.81	45.0942 ± 0.75	8.3043 ± 0.49

Table 5: Timing details of our algorithm are compared to BMM original implementation [7]. The above statistics are the overall time taken by the algorithm for computing both the rotation and the shape. The above numbers are in seconds. Executing the code at different times can provide slightly different execution time; hence, we also put the corresponding expected variance up to 2 decimal places. Our Method (P) represent the processing time for the parallel implementation and Our Method (S) is for sequential implementation, respectively.

For comparison, we could have included other methods; yet we confine ourselves to the prior-free method as our approach is well aligned with it. Also, it indicates the computation overhead by using organic priors rather than no prior.

(f) Comparison with few other methods on CMU MoCap dataset. Based on the recommendation of the reviewer, we added some more statistical



(a) Back Sequence (b) Face Sequence (c) Heart Sequence

Fig. 3: Qualitative Results on Garg *et al.* [10] Real Video Sequence. **Top row**: Example Input Image Sequence. **Bottom row:** Reconstructed 3d shape of the corresponding frame in the video sequence.

comparison of our method with other methods. The table in the main paper is already too dense therefore, we put it in the supplementary material.

Dataset (noise-free)	Drink	Pickup	Yoga	Stretch	Dance
P-Bézier [1]	0.009	0.138	0.111	0.062	0.143
SEA [2]	0.011	0.235	0.158	0.084	0.229
Separable ST [31]	0.009	0.170	0.111	0.094	0.134
Ours	**0.007**	**0.015**	**0.012**	**0.012**	**0.120**

The statistical results show that our method despite being simple provides exemplary 3D reconstruction accuracy.

(g) Results on Real Dense Sequence. We used Garg *et al.* [10] dataset sequence for dense 3d reconstruction on real-world example images. Unfortunately, no ground-truth data is available for its statistical evaluation, and therefore, we show the qualitative results obtained on these sequences in Fig.(3).

2 Single Rotation Averaging for NRSfM

In this work, we use the theory of single rotation averaging for NRSfM problem. In particular L_1 single rotation averaging. To our knowledge, rotation averaging until now in the vision field is applied to rigid structure from motion [12,15]. Our work shows that rotation averaging is equally important to NRSfM problem.

Despite rotation averaging under the L_2 norm is widely studied and put to use [28], L_1 averaging averaging show markedly improved results [15]. It is well understood that L_1 mean (geometric median) is generally more robust to outliers than L_2 mean. In the context of NRSfM, we have absolute rotation prior $\mathbf{R}^k \in \mathbb{R}^{3F\times 3}$ for every column triplet $\mathbf{G}^k$. After registration to the reference rotation, some of them become turn out to be outliers; therefore, we must resort to the robust estimator for rotation averaging. Unfortunately, we don't have a closed-form solution to the L_1 single rotation averaging. Nonetheless, we have an effective and fast iterative Weiszfeld algorithm to solve L_1 single rotation averaging [14]. For rigorous details on theory and convergence analysis, refer to Hartley *et al.* rotation averaging tutorial [15].

2.1 Why single rotation averaging helps?

Averaging helps because we now have more evidence for per frame rotation that can help refine the solution. The first corrective triplet rotation solution gives one of the possible solutions, and it cannot guarantee the best plausible rotation that could be recovered from the matrix factorization theory to NRSfM, as shown recently by Kumar [19]. In our case, we have K rotation evidence per frame. We initialize the L_1 averaging with the *median* of the registered sample (Eq.6) than blindly average it with the first corrective triplet rotation prior as initialization (see **Algorithm 1** of the main paper). Such a strategy treats some prior rotations as outliers; hence provide us with the robust estimation of rotation [27]. Further, as quoted in [14] "The geometric median has a breakdown point of 0.5. That is, up to half of the sample data may be arbitrarily corrupted, and the median of the sample will still provide a robust estimation for the location of the uncorrupted data".

3 Shape optimization

In this section, we show the derivation of the closed form solution derivation of the shape variables *i.e.*, $\mathbf{X}$, $\mathbf{X}^\sharp$ used in ADMM optimization. The overall cost function for shape optimization presented in the main paper is as follows

$$\begin{aligned}\mathcal{L}_\rho(\mathbf{X}, \mathbf{X}^\sharp) &= \frac{1}{2}\|\mathbf{W} - \mathbf{R}\mathbf{X}\|_{\mathcal{F}}^2 + \mu\|\mathbf{X}^\sharp\|_{r=N,\theta} \\ &+ \frac{\rho}{2}\|\mathbf{X}^\sharp - \mathbf{\Phi}(\mathbf{X})\|_{\mathcal{F}}^2 + < \mathbf{Y}, \mathbf{X}^\sharp - \mathbf{\Phi}(\mathbf{X}) >\end{aligned} \tag{1}$$

(a) Closed form solution to X: Assuming $\mathbf{X}^\sharp$ as constant Eq.(1) can be written as

$$\mathbf{X} \approx \underset{\mathbf{X}}{\operatorname{argmin}} \frac{1}{2}\|\mathbf{W}-\mathbf{R}\mathbf{X}\|_{\mathcal{F}}^2 + \frac{\rho}{2}\|\mathbf{X}^\sharp - \mathbf{\Phi}(\mathbf{X})\|_{\mathcal{F}}^2 + < \mathbf{Y}, \mathbf{X}^\sharp - \mathbf{\Phi}(\mathbf{X}) >$$

Since $\mathbf{\Phi}(.)$ is a linear one-to-one function, we can re-write the above equation as

$$\approx \underset{\mathbf{X}}{\operatorname{argmin}} \frac{1}{2}\|\mathbf{W}-\mathbf{R}\mathbf{X}\|_{\mathcal{F}}^2 + \frac{\rho}{2}\|\mathbf{\Phi}^{-1}(\mathbf{X}^\sharp) - \mathbf{X}\|_{\mathcal{F}}^2 + < \mathbf{\Phi}^{-1}(\mathbf{Y}), \mathbf{\Phi}^{-1}(\mathbf{X}^\sharp) - \mathbf{X} >$$

$$\approx \underset{\mathbf{X}}{\operatorname{argmin}} \frac{1}{2}\|\mathbf{W}-\mathbf{R}\mathbf{X}\|_{\mathcal{F}}^2 + \frac{\rho}{2}\left[\|\mathbf{\Phi}^{-1}(\mathbf{X}^\sharp)\|_{\mathcal{F}}^2 + \|\mathbf{X}\|_{\mathcal{F}}^2 - 2Tr(\mathbf{\Phi}^{-1}(\mathbf{X}^\sharp)^T\mathbf{X})\right]$$

$$+ Tr(\mathbf{\Phi}^{-1}(\mathbf{Y}))^T(\mathbf{X}^\sharp - \mathbf{X})$$

Now, dropping the terms that do not involve $\mathbf{X}$ variable.

$$\approx \underset{\mathbf{X}}{\operatorname{argmin}} \frac{1}{2}\|\mathbf{W}-\mathbf{R}\mathbf{X}\|_{\mathcal{F}}^2 + \frac{\rho}{2}\left[\|\mathbf{X}\|_{\mathcal{F}}^2 - 2Tr(\mathbf{\Phi}^{-1}(\mathbf{X}^\sharp)^T\mathbf{X})\right] - Tr(\mathbf{\Phi}^{-1}(\mathbf{Y}))^T\mathbf{X}$$

Using the trace property, $Tr(\mathbf{A}^T\mathbf{B}) = Tr(\mathbf{B}^T\mathbf{A})$

(2)

$$\approx \underset{\mathbf{X}}{\operatorname{argmin}} \frac{1}{2}\|\mathbf{W}-\mathbf{R}\mathbf{X}\|_{\mathcal{F}}^2 + \frac{\rho}{2}\left[\|\mathbf{X}\|_{\mathcal{F}}^2 - 2Tr\mathbf{X}^T\left(\mathbf{\Phi}^{-1}(\mathbf{X}^\sharp) + \frac{\mathbf{\Phi}^{-1}(\mathbf{Y})}{\rho}\right)\right]$$

(3)

Adding the constant to get a better form for optimization

$$\mathcal{L}_\rho(\mathbf{X}, \mathbf{X}^\sharp) \approx \underset{\mathbf{X}}{\operatorname{argmin}} \frac{1}{2}\|\mathbf{W}-\mathbf{R}\mathbf{X}\|_{\mathcal{F}}^2 + \frac{\rho}{2}\left\|\mathbf{X} - \left(\mathbf{\Phi}^{-1}(\mathbf{X}^\sharp) + \frac{\mathbf{\Phi}^{-1}(\mathbf{Y})}{\rho}\right)\right\|_{\mathcal{F}}^2$$

To get the closed form solution for $\mathbf{X}$, we differentiate the above term $\mathcal{L}_\rho(\mathbf{X}, \mathbf{X}^\sharp)$ w.r.t $\mathbf{X}$ and equate it to zero *i.e.*, $\frac{\partial \mathcal{L}}{\partial \mathbf{X}} = 0$. By performing that, we get the following:

$$(\mathbf{R}^T\mathbf{R} + \rho\mathbf{I})\mathbf{X} = \mathbf{R}^T\mathbf{W} + \rho\left(\mathbf{\Phi}^{-1}(\mathbf{X}^\sharp) + \frac{\mathbf{\Phi}^{-1}(\mathbf{Y})}{\rho}\right) \tag{4}$$

Therefore,

$$\mathbf{X} = \left(\mathbf{R}^T\mathbf{R} + \rho\mathbf{I}\right)^{-1}\left[\mathbf{R}^T\mathbf{W} + \rho\left(\mathbf{\Phi}^{-1}(\mathbf{X}^\sharp) + \frac{\mathbf{\Phi}^{-1}(\mathbf{Y})}{\rho}\right)\right] \tag{5}$$

(b) Derivation of solution to $\mathbf{X}^\sharp$: Assuming $\mathbf{X}$ as constant Eq.(1) can be written as

$$\begin{aligned}
\mathbf{X}^\sharp &\approx \underset{\mathbf{X}^\sharp}{\operatorname{argmin}}\ \mu\|\mathbf{X}^\sharp\|_{r=N,\theta} + \frac{\rho}{2}\|\mathbf{X}^\sharp - \mathbf{\Phi}(\mathbf{X})\|_{\mathcal{F}}^2 + < \mathbf{Y}, \mathbf{X}^\sharp - \mathbf{\Phi}(\mathbf{X}) > \\
&\approx \underset{\mathbf{X}^\sharp}{\operatorname{argmin}}\ \mu\|\mathbf{X}^\sharp\|_{r=N,\theta} + \frac{\rho}{2}\left[\|\mathbf{X}^\sharp\|_{\mathcal{F}}^2 + \|\mathbf{\Phi}(\mathbf{X})\|_{\mathcal{F}}^2 - 2Tr(\mathbf{X}^\sharp)^T\mathbf{\Phi}(\mathbf{X})\right] + \\
&\quad Tr(\mathbf{Y}^T\mathbf{X}^\sharp) - Tr(\mathbf{Y}^T\mathbf{\Phi}(\mathbf{X})) \qquad (6)\\
&\approx \underset{\mathbf{X}^\sharp}{\operatorname{argmin}}\ \mu\|\mathbf{X}^\sharp\|_{r=N,\theta} + \frac{\rho}{2}\left[\|\mathbf{X}^\sharp\|_{\mathcal{F}}^2 - 2Tr(\mathbf{X}^\sharp)^T(\mathbf{\Phi}(\mathbf{X}) - \frac{\mathbf{Y}}{\rho})\right] \\
&\approx \underset{\mathbf{X}^\sharp}{\operatorname{argmin}}\ \mu\|\mathbf{X}^\sharp\|_{r=N,\theta} + \frac{\rho}{2}\left\|\mathbf{X}^\sharp - \left(\mathbf{\Phi} - \frac{\mathbf{Y}}{\rho}\right)\right\|_{\mathcal{F}}^2
\end{aligned}$$

For simplicity and clarity on the symbols for the derivation, assume $\mathbf{X}^\sharp = \mathbf{P}$ and $\left(\mathbf{\Phi} - \frac{\mathbf{Y}}{\rho}\right) = \mathbf{Q}$, and $\frac{\mu}{\rho} = \tau$. Substituting, the symbol in the above equation gives

$$\underset{\mathbf{P}}{\operatorname{argmin}}\ \tau\|\mathbf{P}\|_{r=N,\theta} + \frac{1}{2}\|\mathbf{P} - \mathbf{Q}\|_{\mathcal{F}}^2 \qquad (7)$$

Eq.(7) optimization has form similar to the one introduced by Oh *et al.* [29] on partial sum minimization of singular values. Yet, on the contrary, we have weight priors for the singular values greater than N for optimization[1].

Theorem 1. *If the optimization problem has the following form*

$$\underset{\mathbf{P}}{\operatorname{argmin}}\ \tau\|\mathbf{P}\|_{r=N} + \frac{1}{2}\|\mathbf{P} - \mathbf{Q}\|_{\mathcal{F}}^2 \qquad (8)$$

where, $\tau > 0$ and $\mathbf{P}$, $\mathbf{Q} \in \mathbb{R}^{m\times n}$ be real valued matrices which can be decomposed by Singular Value Decomposition (SVD). Then, the optimal solution can be expressed by the Partial Singular Value Thresholding (PSVT) operator defined as follows in [29]:

$$\mathcal{P}_{N,\tau}[\mathbf{Q}] = \mathbf{U}_Q(\mathbf{\Sigma}_{Q1} + \mathcal{S}_\tau[\mathbf{\Sigma}_{Q2}])\mathbf{V}_Q^T \qquad (9)$$

where, $\mathbf{\Sigma}_{Q1} = \mathbf{diag}(\sigma_1, \sigma_2, .., \sigma_N, .., 0)$ and $\mathbf{\Sigma}_{Q2} = \mathbf{diag}(0, ..\sigma_{N+1}, .., \max(m, n))$. Symbol $\mathcal{S}_\tau$ is the soft-thresholding operator defined as $\mathcal{S}_\tau(\sigma) = \text{sign}(\sigma)\max(|\sigma| - \tau, 0)$. $\mathbf{Q} = \mathbf{U}_Q(\mathbf{\Sigma}_{Q1} + \mathbf{\Sigma}_{Q2})\mathbf{V}_Q^T$

Proof: Let $r = \min(m, n)$ and $\mathbf{U}_P\mathbf{\Sigma}_P\mathbf{V}_P^T = \sum_{i=1}^{r}\sigma_i(\mathbf{P})u_iv_i^T$, where $\mathbf{\Sigma}_P = \texttt{diag}(\sigma(\mathbf{P}))$ with $\sigma(\mathbf{P}) = [\sigma_1(\mathbf{P}), ..., \sigma_r(\mathbf{P})]$ contains the non-increasing singular

[1] We are deriving here in the supplementary material for completeness. Readers may refer to [29] for more comprehensive discussion and related proofs.

values of $\mathbf{P}$. Using such algebraic relations, Eq.(8) optimization function w.r.t $\mathbf{P}$ variable can be re-written as follows:

$$\underset{\mathbf{P}}{\operatorname{argmin}}\ \tau \sum_{i=N+1}^{r} \sigma_i(\mathbf{P}) + \frac{1}{2}\|\mathbf{Q}\|_{\mathcal{F}}^2 + \frac{1}{2}\sum_{i=1}^{r}\big(-2\sigma_i(\mathbf{P})u_i^T\mathbf{Q}v_i + \sigma_i(\mathbf{P})^2\big) \tag{10}$$

Since the optimization is w.r.t $\mathbf{P}$, variable $\mathbf{Q}$ does not affect the optimization. This leads to the following minimization problem

$$\underset{\mathbf{P}}{\operatorname{argmin}}\ \tau \sum_{i=N+1}^{r} \sigma_i(\mathbf{P}) + \frac{1}{2}\sum_{i=1}^{r}\big(-2\sigma_i(\mathbf{P})u_i^T\mathbf{Q}v_i + \sigma_i(\mathbf{P})^2\big) \tag{11}$$

For our problem setup, using the von Neumann's lemma on the inner product of matrices [13,29] *i.e.*, $\langle \mathbf{P}, \mathbf{Q}\rangle \leq \langle \sigma(\mathbf{Q}), \sigma(\mathbf{Q})\rangle$, the upper bound of $u_i^T\mathbf{Q}v_i$ can be set as $\max\{u_i^T\mathbf{Q}v_i\}$, which is equal to $\sigma_i(\mathbf{Q})\ \forall\ i$ when $\mathbf{U}_P = \mathbf{U}_Q$ and $\mathbf{V}_P = \mathbf{V}_Q$. The lower envelope of Eq.(11) is obtained at $\mathbf{U}_P = \mathbf{U}_Q$ and $\mathbf{V}_P = \mathbf{V}_Q$. Consequently, Eq.(11) optimization depends only singular values of $\mathbf{P}$ *i.e.*, $\mathbf{\Sigma}_P$ leading to the following optimization function:

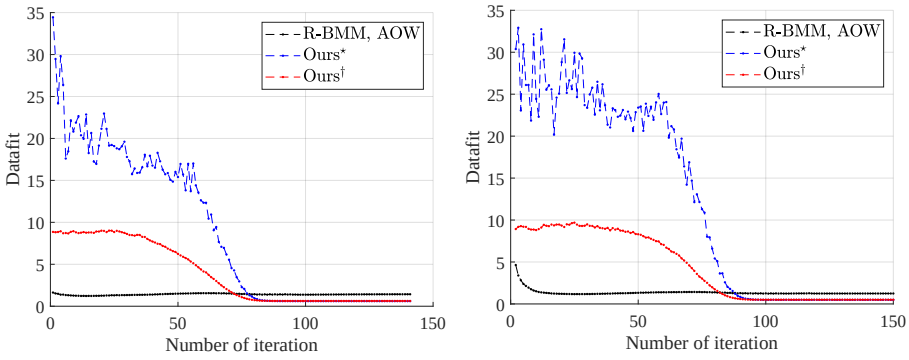
$$\begin{aligned}
&\underset{\mathbf{P}}{\operatorname{argmin}}\ \tau \sum_{i=N+1}^{r} \sigma_i(\mathbf{P}) + \frac{1}{2}\sum_{i=1}^{r}\big(-2\sigma_i(\mathbf{P})\sigma_i(\mathbf{Q}) + \sigma_i(\mathbf{P})^2\big) \\
&= \frac{1}{2}\left(\sum_{i=N+1}^{r}\big(2\tau\sigma_i(\mathbf{P}) - 2\sigma_i(\mathbf{P})\sigma_i(\mathbf{Q}) + \sigma_i(\mathbf{P})^2\big) + \sum_{i=1}^{N}\big(-2\sigma_i(\mathbf{P})\sigma_i(\mathbf{Q}) + \sigma_i(\mathbf{P})^2\big)\right)
\end{aligned} \tag{12}$$

As shown before [29] the minimum of Eq.(12) is obtained at $\hat{\mathbf{\Sigma}}_P = \texttt{diag}(\hat{\sigma}(\mathbf{P}))$ (first order optimality condition), where

$$\hat{\sigma}_i(\mathbf{P}) = \begin{cases} \sigma_i(\mathbf{Q}), & \text{if } i < N+1 \\ \max(\sigma_i(\mathbf{Q}) - \tau, 0) & \text{otherwise} \end{cases} \tag{13}$$

The above solution exactly corresponds to the PSVT operator as defined in Eq.(9) utilizing the von Neumann's lemma [29, 30] *i.e.*, $\mathbf{P}^* = \mathbf{U}_Q\hat{\mathbf{\Sigma}}_P\mathbf{V}_Q$. Re-writing, it using shrinkage operator $\mathbf{U}_Q(\mathbf{\Sigma}_{Q1} + \mathcal{S}_\tau[\mathbf{\Sigma}_{Q2}])\mathbf{V}_Q^T$.

For the NRS*f*M formulation, we have an additional weight prior θ due to the pseudo-inverse solution of $\mathbf{X}$, which we used for optimization. In our optimization (Eq.14 in the main paper), we substitute $\tau = \frac{\mu\theta}{\rho}$ and $N = 1$. The point to note is that by weighting the initial singular value prior for the shape matrix $\mathbf{X}^\sharp$ optimization, the criteria for the optimality on the weighted nuclear norm minimization do not change *i.e.*, the weights still satisfy the criteria $0 \leq w_1 \leq w_2 ... \leq w_n$ and $\sigma_1 \geq \sigma_2 .... \geq \sigma_n \geq 0$. The only difference is we do not penalise the first singular value of the shape during minimization. And therefore, it can be regarded as the special case of weighted nuclear norm minimization [6, 9, 13]. For



(a) Stretch Sequence (b) Pickup Sequence

Fig. 4: The shape optimization cost function value (Eq.(10) in the main paper) over iterations. Ours shape optimization gives better minima than the current state-of-the-art R-BMM [19] and AOW [16]. Ours* and Ours† show the results when [8] rotation and our proposed rotation averaging based estimated rotation is used as initialization, respectively. Clearly, our approach achieves better minima with stable convergence response, and faster w.r.t Ours*. (a)-(b) Results on stretch, pickup seq. [3] respectively.

our problem, we exploited the initial shape prior to encourage the proposed optimization to better meet the target rank constraint. Generally, for NRSfM case $N = 1$ works well.

Although Eq.(6) is a non-convex optimization problem, minimizing it using PSVT operator via ADMM —which has shown effectiveness for many non-convex computer vision problems [4, 33], gives highly satisfactory results. Fig.4 clearly show the our shape optimization provides better minima than weighted nuclear norm minimization methods *i.e.*, R-BMM [19], AOW [16]. In **Algorithm Table 1**, we provide the pseudo code of the proposed idea for better understanding of our approach.

4 Partial Sum Minimization of Singular Values: Intuition

In NRSfM, it is generally the case that the shape spans a low-rank space which can be less than the assumed K or $3K$ (depending on the previous works that dealt with the rank of the shape) [5] [34] [3] [8]. Therefore, using nuclear norm minimization to recover rank K shape solution can unfavorably affect the outcome of NRSfM factorization.

As demonstrated in several NRSfM factorization work, the shape may lie in a space that is much lower than K. Hence, outliers are included as inliers in the solution space for nuclear norm minimization can lead to questionable results after the optimization. Further, the nuclear norm minimization for the low-rank shape minimizes the rank, including the singular values within the space's true

Algorithm 1: Pseudo code of our approach

```
% Pseudo code of our algorithm.
%% Input
[W, K] = read_the_input_and_suitable_K('dataset_name');

% mean centralize the measurement matrix.
Wm = mean(W, 2);
W = W - Wm*ones(1, size(W, 2));

%% get the important intermediate information.
[F, P] = size(W); F = F/2;

%% Initialize the variable
epsilon_t = 1e-3; lambda = 1.1; xi=5e-3; gamma = 1e-6; rho = 1e-4;
delta = 0.05; rho_max = 1e10; epsilon = 1e-10;
Y = zeros(F, 3*P);

%% estimate the rotation matrix
[GK] = compute_the_correction_matrix(W, K);
[RK] = recover_rotation_from_correction(W, K, GK);
[R_avg] = perform_rotation_averaging(RK, delta, epsilon_t);
[R] = orthographic_rotation_per_frame(R_avg, F, P);

%% estimate the shape matrix
% get the initial shape prior and corresponding initial weights (
% in the main Eq.9 main).
X_init = pinv(R)*W;
[~, sigma, ~] = svd(Phi(X_init), 'econ');
c = xi*sqrt(sigma(1, 1));
[n, ~] = size(sigma);
Theta = eye(n, n).*repmat(c./(diag(sigma) + gamma), [1, n]);

% initialize some other variables
mu = 1.0;            % regularization parameter.
Xs = g(X_init);      % initialize the tranformed shape
I  = eye(3*F);
iter = 1;            % initialize the iteration value.
N = 1;               % for the first component.
Theta(1:N, 1:N) = zeros(N); % preserve the first component.
datafit_val = [];

% ADMM optimization
while(true)
    %solve for X
    X  = solveX(W, R, Xs, I, Y, rho);
    %solve for X^{\sharp}
    Xs = solveXs_preserve_component(X, Y, rho, mu, Theta, N);
    % update lagrange multiplier
    Y  = Y + rho*(Xs-g(X));
    % check for the gap
    gap = max(max(abs(Xs-g(X))));
    %check for convergence and break;
    if(gap<epsilon || rho>=rho_max)
        break;
    else
        rho = min(rho_max, lambda*rho);
    end
    %print the intermediate statistics
    fprintf('Iteration number = %d \n', iter);
    fprintf('The value of current rho = %f\n', rho);
    iter = iter + 1;
end

return X;
```

rank, which can really hurt the performance. Consequently, WNN minimization provides a good alternative for recovering better results [19] [16]. On the contrary, the proposed shape matrix optimization approach takes a more noticeable step that the first singular values of the pseudo-inverse shape solution is more often than not gives a very good prior on the shape, so why not preserve it.

Concretely, our shape optimization attempts to find a mid-way between the partial sum minimization of singular value and WNN in NRSfM for better shape recovery. The proposed method pushes the shape optimization towards a good solution by exploiting the organic shape prior so that it does not minimize the subspace variance within the given N. We demonstrated it *empirically* in the main paper that $N = 1$ suits the NRSfM case since we don't know the exact rank of the deforming shape *apriori* in NRSfM unlike photometric stereo ($N = 3$), rigid SfM ($N = 3$), and other low-level vision problems.

5 Discussion and critical observation with NRSfM

Similar to other available NRSfM algorithms, our method's performance varies with different camera trajectory types [17]. Finally, a large part of success in NRSfM is due to the development in the optimization methods assuming that image key-points correspondence is correct or reasonably accurate. So, our results are as good as image feature correspondences, yet image feature correspondence in itself is a challenging vision task.

One critical avenue of inspection with our method or other similar methods is the rank minimization approach to NRSfM [19, 21, 29, 35]. As mentioned above, ADMM based minimization may provide a sub-optimal solution for non-convex NRSfM formulation. Still, time and again, it has been observed practically that the rank minimization-based statistical approaches provides better results without theoretical global optimal guarantees [17, 21]. For instance, if we look into recent Jensen *et al.* [17] benchmark results in Table 5, the top method *i.e.*,"Multi-body" [21, 22] is a non-convex optimization algorithm with no theoretical global minimum guarantees for the shape optimization, yet it demonstrated state-of-the-art results for most of the object categories.

References

1. Agudo, A.: Piecewise bézier space: Recovering 3d dynamic motion from video. In: 2021 IEEE International Conference on Image Processing (ICIP). pp. 3268–3272. IEEE (2021)
2. Agudo, A., Moreno-Noguer, F.: A scalable, efficient, and accurate solution to non-rigid structure from motion. Computer Vision and Image Understanding **167**, 121–133 (2018)
3. Akhter, I., Sheikh, Y., Khan, S., Kanade, T.: Nonrigid structure from motion in trajectory space. In: Advances in neural information processing systems. pp. 41–48 (2008)
4. Boyd, S., Parikh, N., Chu, E., Peleato, B., Eckstein, J.: Distributed optimization and statistical learning via the alternating direction method of multipliers. Foundations and Trends® in Machine Learning **3**(1), 1–122 (2011)
5. Bregler, C., Hertzmann, A., Biermann, H.: Recovering non-rigid 3d shape from image streams. In: IEEE Conference on Computer Vision and Pattern Recognition. vol. 2, pp. 690–696. IEEE (2000)
6. Chen, K., Dong, H., Chan, K.S.: Reduced rank regression via adaptive nuclear norm penalization. Biometrika **100**(4), 901–920 (2013)
7. Dai, Y., Li, H., He, M.: A simple prior-free method for non-rigid structure-from-motion factorization. In: Computer Vision and Pattern Recognition (CVPR). pp. 2018–2025. IEEE (2012)
8. Dai, Y., Li, H., He, M.: A simple prior-free method for non-rigid structure-from-motion factorization. International Journal of Computer Vision **107**(2), 101–122 (2014)
9. Gaïffas, S., Lecué, G.: Weighted algorithms for compressed sensing and matrix completion. arXiv preprint arXiv:1107.1638 (2011)
10. Garg, R., Roussos, A., Agapito, L.: Dense variational reconstruction of non-rigid surfaces from monocular video. In: IEEE Conference on Computer Vision and Pattern Recognition. pp. 1272–1279 (2013)
11. Gotardo, P.F., Martinez, A.M.: Non-rigid structure from motion with complementary rank-3 spaces. In: Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on. pp. 3065–3072. IEEE (2011)
12. Govindu, V.M.: Robustness in motion averaging. In: Asian Conference on Computer Vision. pp. 457–466. Springer (2006)
13. Gu, S., Zhang, L., Zuo, W., Feng, X.: Weighted nuclear norm minimization with application to image denoising. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2862–2869 (2014)
14. Hartley, R., Aftab, K., Trumpf, J.: L1 rotation averaging using the weiszfeld algorithm. In: CVPR 2011. pp. 3041–3048. IEEE (2011)
15. Hartley, R., Trumpf, J., Dai, Y., Li, H.: Rotation averaging. International journal of computer vision **103**(3), 267–305 (2013)
16. Iglesias, J.P., Olsson, C., Valtonen Örnhag, M.: Accurate optimization of weighted nuclear norm for non-rigid structure from motion. In: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXVII 16. pp. 21–37. Springer (2020)
17. Jensen, S.H.N., Doest, M.E.B., Aanæs, H., Del Bue, A.: A benchmark and evaluation of non-rigid structure from motion. International Journal of Computer Vision (IJCV) **129**(4), 882–899 (2021)

18. Kumar, S.: Jumping manifolds: Geometry aware dense non-rigid structure from motion. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5346–5355 (2019)
19. Kumar, S.: Non-rigid structure from motion: Prior-free factorization method revisited. In: Winter Conference on Applications of Computer Vision (WACV 2020). pp. 51–60 (2020)
20. Kumar, S., Cherian, A., Dai, Y., Li, H.: Scalable dense non-rigid structure-from-motion: A grassmannian perspective. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 254–263 (2018)
21. Kumar, S., Dai, Y., H.Li: Spatio-temporal union of subspaces for multi-body non-rigid structure-from-motion. Pattern Recognition **71**, 428–443 (May 2017)
22. Kumar, S., Dai, Y., Li, H.: Multi-body non-rigid structure-from-motion. In: 3D Vision (3DV), 2016 Fourth International Conference on. pp. 148–156. IEEE (2016)
23. Kumar, S., Dai, Y., Li, H.: Monocular dense 3d reconstruction of a complex dynamic scene from two perspective frames. In: IEEE International Conference on Computer Vision. pp. 4649–4657 (Oct 2017)
24. Kumar, S., Dai, Y., Li, H.: Superpixel soup: Monocular dense 3d reconstruction of a complex dynamic scene. IEEE transactions on pattern analysis and machine intelligence **43**(5), 1705–1717 (2019)
25. Kumar, S., Ghorakavi, R.S., Dai, Y., Li, H.: Dense depth estimation of a complex dynamic scene without explicit 3d motion estimation. arXiv preprint arXiv:1902.03791 (2019)
26. Kumar, S., Van Gool, L., de Oliveira, C.E., Cherian, A., Dai, Y., Li, H.: Dense non-rigid structure from motion: A manifold viewpoint. arXiv preprint arXiv:2006.09197 (2020)
27. Lee, S.H., Civera, J.: Robust single rotation averaging. arXiv preprint arXiv:2004.00732 (2020)
28. Manton, J.H.: A globally convergent numerical algorithm for computing the centre of mass on compact lie groups. In: ICARCV 2004 8th Control, Automation, Robotics and Vision Conference, 2004. vol. 3, pp. 2211–2216. IEEE (2004)
29. Oh, T.H., Tai, Y.W., Bazin, J.C., Kim, H., Kweon, I.S.: Partial sum minimization of singular values in robust pca: Algorithm and applications. IEEE transactions on pattern analysis and machine intelligence **38**(4), 744–758 (2016)
30. de Sá, E.M.: Exposed faces and duality for symmetric and unitarily invariant norms. Linear Algebra and its Applications **197**, 429–450 (1994)
31. Simon, T., Valmadre, J., Matthews, I., Sheikh, Y.: Separable spatiotemporal priors for convex reconstruction of time-varying 3d point clouds. In: European Conference on Computer Vision. pp. 204–219. Springer (2014)
32. Torresani, L., Hertzmann, A., Bregler, C.: Nonrigid structure-from-motion: Estimating shape and motion with hierarchical priors. IEEE transactions on pattern analysis and machine intelligence **30**(5), 878–892 (2008)
33. Vidal, R., Ma, Y., Sastry, S.: Generalized principal component analysis (gpca). IEEE transactions on pattern analysis and machine intelligence **27**(12), 1945–1959 (2005)
34. Xiao, J., Chai, J.x., Kanade, T.: A closed-form solution to non-rigid shape and motion recovery. In: European conference on computer vision. pp. 573–587. Springer (2004)
35. Zhu, Y., Huang, D., De La Torre, F., Lucey, S.: Complex non-rigid motion 3d reconstruction by union of subspaces. In: IEEE Conference on Computer Vision and Pattern Recognition. pp. 1542–1549 (2014)