Supplementary Material for K-Best Transformation Synchronization

Yifan Sun * Jiacheng Zhuo*
UT Austin
Austin, TX 78712
{yifansun12,jzhuo}@utexas.edu

Arnav Mohan
Liberal Arts and Science Academy
Austin, TX 78745
arnavmohan@gmail.com

Qixing Huang
UT Austin
Austin, TX 78712
huangqx@cs.utexas.edu

A. Proofs for Exact Recovery of K-best solution

In this section, we present a formal proof of the exact recovery condition of K-best solution.

A.1. Randomized Exact Recovery Conditions

We begin with reviewing the random noise model for generating the K-best solution. A graph $G = (\mathcal{I}, \mathcal{E})$ is a random graph $G = (\mathcal{I}, p_e, p_g)$ if $\mathcal{E}$ meets the follow two conditions:

- Independently and identically for any vertices i and j, $\Pr[(i,j) \in \mathcal{E}] = p_e$;
- For any edge $(i,j) \in \mathcal{E}$ and $1 \le k \le K$, independently with probability $\frac{p_g}{K}$, $\mathcal{P}_{\mathcal{S}}(T_{ij}^{in} T_{i1}^{\star}) = \mathcal{P}_{\mathcal{S}}(T_{jk}^{\star})$.

As the underlying object possesses a symmetry group of size K, the second condition implies that with equal probability an edge (i,j) picks one of the K possible ground-truth transformations from Σ_i to Σ_j, and with probability $1 - p_g$ it is initially incorrectly estimated.

The key idea of the proof is to divide our algorithm into an exploration stage and a final stage. In the exploration stage, our algorithm seeks to reach all input objects from the root object. The final stage assumes all input objects have been visited and our algorithm proceeds until termination. In particular, we will show that under the proposed exact recovery conditions the majority of the propagated transformations are correct after the exploration stage. Then the final stage iteratively corrects the remaining incorrect propagated transformations.

Let us introduce a few notations. Based on the iteration $exploretime(I_i)$ when an object I_i is explored for the first time, e.g., $\boldsymbol{v}_i^{(t)} \neq \mathbf{0}$ if and only if $t \ge exploretime(I_i)$, we can divide $\mathcal{I}$ into subgroups $\Gamma_0, \Gamma_1, \Gamma_2, ..., \Gamma_D$, where

$$\Gamma_j = \{I | I \in \mathcal{I}, exploretime(I) = j\},$$

and $D = D(G)$ is defined as the diameter of G, which is the maximum of shortest paths from every pair of vertices. Note that Γ_j can be viewed as the set of vertices of height j on the breadth-first-search tree(BFS-Tree) of G rooted by the first vertex I_1.

We define the correct rate function $Corr^{(j)}(\mathcal{I}')$ on a subset of vertices $\mathcal{I}'$ to be

$$Corr^{(l)}(\mathcal{I}') = \frac{\sum\limits_{I_i \in \mathcal{I}'} \text{\# of correct estimations in } \mathcal{T}_i^{(l)}}{\sum\limits_{I_i \in \mathcal{I}'} \text{\# of estimations in } \mathcal{T}_i^{(l)}}.$$

To avoid comprehensive notations, we will replace some of the constant variables in the analysis of our main article by a constant number. The definition of random graph in the proof of exact recovery conditions is also slightly different in definition for simplification of the proof without affecting its correctness, because δ is considered only a small perturbation of p_e. Since we will apply Chernoff bound several times, we always use η to denote the small constant in the standard expression of Chernoff bound independently every time.

Exploration stage. The exploration stage consists of the first D iterations of transformation propagation and clustering in our algorithm, where still some of the objects remain unexplored. We show that under certain conditions on parameters p_e and p_g of random graphs, the graph can meet the requirement of convergence in the final stage, despite that $Corr^{(t)}(\bigcup_{j \le t} \Gamma_j)$ may decrease during the exploration stage for $0 \le t \le D$.

Lemma A.1. *(Lemma 3 and Corollary 8 in [5]) For a random graph $G(\mathcal{I}, p_e, p_g)$ with $|\mathcal{I}| = n$, if $p_e = n^{\frac{1}{d-\epsilon}-1}$ for constants $d > 3$ and $0 < \epsilon < 1$, then with probability higher than $1 - e^{-n}$ the diameter of G is $D = d$. Moreover, with probability higher than $1 - n^{-11}$ for every $1 \le i \le d-1$,*

$$\frac{3}{4}(p_e n)^j < |\Gamma_j| < \frac{5}{4}(p_e n)^j$$

when n is sufficiently large.

*Both authors contributed equally

This lemma implies that almost surely the following properties hold as G satisfies the conditions stated above:

- **Constant Diameter**: The diameter of the graph is upper-bounded by a constant;
- **Top Domination**: The size of the highest level on the BFS-Tree, Γ_d, dominates the sum of all the other levels, which is only of order $n^{\frac{d-1}{d-\epsilon}}$;
- **Weak Downward Connection**: For any vertex I_x, if $I_x \in \Gamma_j, 1 \leq j \leq d$, it is possible that only one neighbor of I_x is in Γ_{j-1}, and this one is guaranteed by the definition of Γ. Moreover, if $j < d$ then most of I_x's neighbors are in Γ_{j+1}, otherwise if $j = d$ they are in Γ_d.

Lemma A.2. ***(Convergence rate of exploration stage)*** *If $\mathcal{E}$ satisfies the condition of a random graph $G(\mathcal{I}, p_e, p_g)$, where $p_e = n^{\frac{1}{d-\epsilon}-1}$ for constants $d > 3$ and $0 < \epsilon < 1$, and moreover, p_g satisfies*

$$p_g > 0.75^{\frac{1}{d}},$$

then almost surely for every Γ_j, $Corr^{(j)}(\Gamma_i) \geq 0.75$.

Proof. We first give a proof of a tighter upper-bound on $Corr^{(j)}(\Gamma_j)$ by induction. We show that with high probability $Corr^{(j)}(\Gamma_j) \geq (1-\eta)p_g Corr^{(j-1)}(\Gamma_{j-1})$ for every $1 \leq j \leq d$, where $0 < \eta < 1$ is a small constant.

- $j = 0$: There is only one vertex, I_1, in Γ_0, and by definition it is always perfectly aligned with the coordinate system of itself after the identity transformation, $Corr^{(0)}(\Gamma_0) = 1$;
- $j = t$: We randomly pick a vertex I_x from Γ_t and one of its neighbor I_y in Γ_{t-1}.
  - Case 1: I_y is the only vertex in Γ_{t-1} connected to I_x. In this case K-sparse$(\overline{\boldsymbol{v}}_x^{(t)}) = \overline{\boldsymbol{v}}_x^{(t)}$. If the edge between I_x and I_y is correctly estimated initially then $Corr^{(t)}(I_x) = Corr^{(t-1)}(I_y)$, otherwise $Corr^{(t)}(I_x)$ may decrease to 0. By the definition of random graph, I_y can be any vertex in Γ_{t-1} with equal probability, and the expectation of $Corr^{(t)}(I_x)$ can be derived as:

    $$\mathbb{E}[Corr^{(t)}(I_x)] = p_g Corr^{(t-1)}(\Gamma_{t-1});$$

  - Case 2: There are multiple vertices connected to I_x in Γ_{t-1}. Let the number of correct and incorrect pose estimations of I_y after iteration $t-1$ be $r = r(I_y)$ and $w = w(I_y)$, e.g., $Corr^{(t-1)}(I_y) = \frac{r}{w+r}$. If edge (x, y) is correct, the expected number of votes a correct estimation $T_i \in \mathcal{T}_i^{(t)}$ of I_x may receive from I_y is $\frac{r}{K}$ since the edge is uniformly randomly chosen from K possible correct ones. Meanwhile, the expected number of votes an incorrect estimation $T_i' \in \mathcal{T}^{(t)}$ of I_x receives has a tight upper-bound $\frac{w}{K}$ because for any incorrect estimation $T_j \in \mathcal{T}_j^{(t-1)}$ of I_y, among the K possible correct choices of edge at most one of them can map T_j to T_i'. Fix I_x and sum up all choices of I_y, we derive the expected numbers of votes a correct estimation of I_x receives and an incorrect estimation receives:

    $$\mathbb{E}_x[corr] = \sum_{I_y} \frac{r(I_y)}{K} p_g$$

    $$\mathbb{E}_x[incorr] = \sum_{I_y} \frac{w(I_y)}{K} + \frac{r(I_y)}{K}(1-p_g).$$

    As the number of neighbors I_y increases, the ratio between these two expectations will always concentrate to

    $$\frac{p_g Corr^{(t-1)}(\Gamma_{t-1})}{1 - p_g Corr^{(t-1)}(\Gamma_{t-1})},$$

    which means as long as $p_g Corr^{(t-1)}(\Gamma_{t-1}) > 0.5$, the more neighbors I_x have in Γ_{t-1}, the more likely there will be a gap between the number of votes a correct estimation receives and the number of votes an incorrect estimation receives and we will discuss this gap in detail in the proof of lemma A.4. Thus more correct estimates will survive after K-sparse operation;

In combination of the two cases above, we have

$$\mathbb{E}[Corr^{(t)}(I_x)] \geq p_g Corr^{(t-1)}(\Gamma_{t-1}).$$

To minimize $\mathbb{E}[Corr^{(t)}(\Gamma_t)]$, any vertex I_x in Γ_t should have only one neighbor in Γ_{t-1}, which is reasonable under the weak downward connection property, and we reach a lower bound on $\mathbb{E}[Corr^{(t)}(\Gamma_t)]$:

$$\inf \mathbb{E}[Corr^{(t)}(\Gamma_t)] = p_g Corr^{(t-1)}(\Gamma_{t-1}).$$

Apply Chernoff bound under this case, we have

$$\Pr\left[Corr^{(t)}(\Gamma_t) \leq (1-\eta)p_g Corr^{(t-1)}(\Gamma_{t-1})\right]$$
$$\leq \exp\left(-\frac{1}{2}\eta^2 |\Gamma_t| p_g Corr^{(t-1)}(\Gamma_{t-1})\right),$$

where η is any constant between 0 and 1. From lemma A.1 and the top domination property, with high probability every $\Gamma_j, 1 \leq j \leq d$ has size of at least the order of $n^{\frac{1}{d-\epsilon}}$, and thus with high probability $Corr^{(t)}(\Gamma_t) > (1-\eta)p_g Corr^{(t-1)}(\Gamma_{t-1})$.

By applying union bound on every $j > 0$, we have that with high probability (e.g., of order at least $1-\exp(-n^{\frac{1}{d}})$), $Corr^{(j)}(\Gamma_j) > (1-\eta)p_g Corr^{(j-1)}(\Gamma_{j-1})$ holds true for every $1 \le j \le d$.

We the substitute $p_g > 0.75^{\frac{1}{d}}$ in, let η satisfy $(1-\eta)p_g \ge 0.75^{\frac{1}{d}}$:

$$Corr^{(j)}(\Gamma_j) > p_g^d(1-\eta)^d Corr^{(0)}(\Gamma_0) \ge 0.75,$$

and this finishes the proof. □

Final stage. The final stage consists of all the iterations after iteration D until the transformation propagation and clustering step finishes. We will show in the following part that under certain conditions with high probability a random graph $G(\mathcal{I}, p_e, p_g)$ converges to a perfect recovery (e.g., all the K underlying ground-truths are recovered for any object other than I_1).

We start with a more specifically defined weak-downward-connection property:

Lemma A.3. ***(Weak downward connection)*** *If $\mathcal{E}$ satisfies the condition of a random graph $G(\mathcal{I}, p_e, p_g)$, where $p_e = n^{\frac{1}{d-\epsilon}-1}$ for constants $d > 3$ and $0 < \epsilon < 1$, then with high probability for any $I_x \in \mathcal{I}$, the following properties always hold:*

1. *I_x has more than $0.9n^{\frac{1}{d-\epsilon}}$ neighbors;*
2. *if $I_x \in \Gamma_j$ where $j > 0$, then no more than 0.1% of its neighbors are in Γ_{j-1};*
3. *if $I_x \in \Gamma_j$ where $j < n$, then no more than 0.1% of its neighbors are in Γ_j.*

Proof.
1. For any $y \neq x$, the probability that I_y is a neighbor of I_x is p_e. We simply apply Chernoff bound on all possible I_y:

$$\Pr[I_x \text{ has less than } 0.9n^{\frac{1}{d-\epsilon}} \text{ neighbors}] \le \exp\left(-0.005n^{(\frac{1}{d-\epsilon})}\right),$$

and then apply union bound on I_x:

$$\Pr[\text{All vertices have more than } 0.9n^{\frac{1}{d-\epsilon}} \text{ neighbors}] \ge 1 - n\exp\left(-0.005n^{(\frac{1}{d-\epsilon})}\right);$$

2. Now we know that I_x has more than $0.9n^{\frac{1}{d-\epsilon}}$ neighbors, and with high probability there are no more than $\frac{5}{4}n^{\frac{j}{d-\epsilon}}$ vertices in Γ_{j-1}. Besides the "compulsory" neighbor from the definition of Γ, when we randomly choose another neighbor I_y of I_x, the probability that $I_y \in \Gamma_{j-1}$ is

$$\Pr[I_y \in \Gamma_{j-1}] = \frac{|\Gamma_{j-1}| - 1}{\sum_{i=j-1}^{d} |\Gamma_i|} \le \frac{|\Gamma_{j-1}|}{|\Gamma_j|} \le \frac{5}{3}n^{-\frac{1}{d-\epsilon}}.$$

Therefore, the expected number of neighbors $I_y \notin \Gamma_{j-1}$ is of constant order, and

$$\Pr[\frac{|\{I_y \in \Gamma_{j-1}\}|}{|\{I_y\}|} > \eta] \le \exp\left(-0.45\eta^2 n^{\frac{1}{d-\epsilon}}\right)$$

for positive constant η arbitrarily small. Again by applying union bound, we have the second claim proved.

3. We prove this claim with a similar argument as in the proof of the second one. Instead of evaluating $\Pr[I_y \in \Gamma_{j-1}]$, we consider $\Pr[I_y \in \Gamma_j]$:

$$\Pr[I_y \in \Gamma_j] \le \frac{|\Gamma_j|}{|\Gamma_{j+1}|} \le \frac{5}{3}n^{-\frac{1}{d-\epsilon}}$$

and the remaining proof is straightforward. □

Lemma A.4. *If $\mathcal{E}$ satisfies the condition of a random graph $G(\mathcal{I}, p_e, p_g)$, where $p_e = n^{\frac{1}{d-\epsilon}-1}$ for constants $d > 3$ and $0 < \epsilon < 1$, then with high probability all vertices in Γ_d are perfectly recovered after iteration $d+1$, and will always be perfectly recovered in every iteration after.*

Proof. From lemma A.2 we know that when the exploration stage finishes after iteration d, we end up with $Corr^{(d)}(\Gamma_d) \ge 0.75$ and also for any $I_x \in \Gamma_d$, $\mathbb{E}[Corr^{(d)}(I_x)] \ge 0.75$. Also from lemma A.3 we know a dominant majority of I_x's neighbors are in Γ_d. First we only consider neighbors I_y of I_x in Γ_d. We denote this neighborhood subset as $\mathcal{N}_d(I_x)$. In iteration $d+1$, every $\mathcal{T}_y^{(d)}$ propagates $|\mathcal{T}_y^{(d)}|$ candidates to $\overline{\mathcal{T}}_x^{(d+1)}$, thus $\overline{\mathcal{T}}_x^{(d+1)}$ receives

$$\sum_{(y,x)\in\mathcal{E}} \mathcal{T}_y^{(d)} \ge \mathcal{N}_d(I_x) \ge 0.999n^{\frac{1}{d-\epsilon}}$$

votes in total. For a fixed $T \in \mathcal{T}_x^{gt}$, the probability that a vote is for T is

$$\frac{p_g}{K^\star} Corr^{(d)}(\mathcal{N}_d(I_x)).$$

Since $N_d(I_x)$ are randomly chosen from Γ_d, we can replace $Corr^{(d)}(\mathcal{N}_d(I_x))$ by $Corr^{(d)}(\Gamma_d)$ when computing the expectation. Again the total number of votes T receives will concentrate near

$$\frac{p_g}{K^\star} Corr^{(d)}(\Gamma_d) \sum_{(y,x)\in\mathcal{E}} \mathcal{T}_y^{(d)}$$

with probability $1-\exp(-O(1)n^{\frac{1}{d-\epsilon}})$.

Similarly, the total number of votes a non-ground-truth transformation T' of I_x receives will concentrate below

$$\left(\frac{1}{K^\star}-\frac{p_g}{K^\star}Corr^{(d)}(\Gamma_d)\right)\sum_{(y,x)\in\mathcal{E}}\mathcal{T}_y^{(d)}$$

with probability $1-\exp(-O(1)n^{\frac{1}{d-\epsilon}})$.

We have now quantified the gap between the number of votes a ground-truth transformation receives and a non-ground-truth transformation receives. Apply union bound on all the $|\Gamma_d|K$ estimations of vertices in Γ_d, with probability at least $1-|\Gamma_d|K^{\frac{1}{d-\epsilon}}\exp(-O(1)n^{\frac{1}{d-\epsilon}})$, $Corr^{(d+1)}(\Gamma_d)=1$, which is still high.

It is trivial to show that Γ_d will remain perfectly recovered after iteration $d+1$ because $Corr^{(l)}(\Gamma_d)=1>0.75$ for every $l>d$.

□

Lemma A.5. *If $\mathcal{E}$ satisfies the condition of a random graph $G(\mathcal{I},p_e,p_g)$, where $p_e=n^{\frac{1}{d-\epsilon}-1}$ for constants $d>3$ and $0<\epsilon<1$, then with high probability all vertices in Γ_j remain perfectly recovered after iteration $2d+1-j$, for any $0\le j\le d$.*

Proof. A proof by induction is straightforward as lemma A.4 gives the proof of the boundary case where $j=d$. Now we assume all vertices in $\Gamma_{(j+1)}$ are perfectly recovered after iteration $j+1$. For any I_x in Γ_j, lemma A.3 implies that most of its neighbors are perfectly recovered because they are in Γ_{j+1}. We simply apply the concentration bounds as in the proof of lemma A.4. We denote a polynomial of $n,K^\star,d$ as $poly(n,K^\star,d)$, and after running the transformation propagation and clustering algorithm for $O(d)$ iterations, it ends up with a exact recovery of all vertices with high probability $1-poly(n,K^\star,d)\exp(-O(1)n^{\frac{1}{d-\epsilon}})$. □

To sum up, we combine lemma A.1, A.2, A.3, A.4, A.5 and state part 1) of our main theorem here, which is slightly different from the one in our main article only in the notation of constants:

Theorem A.1. *If the input data follows the random noisy model described above, and suppose $p_e=n^{\frac{1}{d-\epsilon}-1}$ for constants $d>3$ and $0<\epsilon<1$, and moreover, p_g satisfies*

$$p_g>0.75^{\frac{1}{d}}$$

. Then almost surely, after $l=O(d)$ iterations, $\mathcal{T}_i^{(l)}$ recovers all the $K^\star$ underlying ground-truth transformations of partial object I_i for every $2\le i\le n$.

A.2. Exact Recovery of $K^\star$

We divide this section into two parts. In the first part, we provide a concise proof about our approach on the exact recovery of $K^\star$. In the second part, we provide an empirical study on the robustness of the proposed approach.

Proof of the exact recovery of $K^\star$. Under the noise model described in the main paper, we have that when $K\le K^\star$, $\text{score}(\mathcal{T}^{out}(K),\mathcal{T}^{in}_{\text{pair}})\in[\frac{n^2p_ep_g}{K^\star}-O(n),\frac{n^2p_ep_g}{K^\star}+O(n)]$ with overwhelming probability. When $K\ge K^\star+1$, $\text{score}(\mathcal{T}^{out}(K),\mathcal{T}^{in}_{\text{pair}})\in[\frac{n^2p_ep_g}{K}-O(n),\frac{n^2p_ep_g}{K}+O(n)]$ with overwhelming probability. Thus, it is easy to see that the largest gap happens at $K=K^\star$, which ends the proof.

	Ours	Base 1)	Base 2)
ShapeNetCoreSym	95%	65%	75%
ScanNetSym	90%	70%	70%

Table 1: Exact recovery rate.

Robustness of our approach. We compared our approach with two alternative approaches described in the main paper, i.e., 1) run the propagation operation without clustering for several iterations and then determine K from the propagated transformations, and 2) run the propagation-and-clustering procedure with a sufficient large $K'>K$ and then determine K from the final output. As shown in the insect table to the right, our approach is more accurate than these two alternative approaches for recovering the underlying symmetries.

B. Proof of Theorem 4.2

The proof of Theorem 4.2 comes from the stability of the leading eigenspace of a matrix that possesses a spectral-gap under random perturbation of matrices. To begin with, we consider the K-order cyclic group $\mathcal{C}_K=\{C_{K,i}\in\mathbb{R}^{K\times K},1\le i\le K\}$. For example, when $K=3$,

$$C_{3,1}=\begin{pmatrix}1&0&0\\0&1&0\\0&0&1\end{pmatrix},\ C_{3,2}=\begin{pmatrix}0&1&0\\0&0&1\\1&0&0\end{pmatrix},$$
$$C_{3,3}=\begin{pmatrix}0&0&1\\1&0&0\\0&1&0\end{pmatrix}$$

Now let us consider a symmetric block-wise random matrix $X\in\mathbb{R}^{nK\times nK}$. The diagonal blocks are identify, i.e.,

$$X_{ii}=I_k,\quad 1\le i\le n.$$

Each off-diagonal block follows the following noisy model independently:

$$X_{ij}=\begin{cases}0 & w.p.\ 1-p_ep_g\\ C_{K,1} & w.p.\ p_ep_g(\frac{1}{K}+\delta)\\ C_{K,i} & w.p.\ p_ep_g(\frac{1}{K}-\frac{\delta}{K-1})\text{ for each }i\end{cases}\quad(1)$$

	ShapeNetCoreSym (K-Best)						ScanNetSym (K-Best)						ShapeNetCore (Single-Best)						ScanNet (Single-Best)					
	Rotation			Trans.			Rotation			Trans.			Rotation			Trans.			Rotation			Trans.		
	3°	30°	Mean	0.05	0.25	Mean	3°	30°	Mean	0.05	0.25	Mean	3°	30°	Mean	0.05	0.25	Mean	3°	30°	Mean	0.05	0.25	Mean
Input[14]	41.4	76.1	24.5	34.1	61.2	0.36	32.5	51.1	41.2	31.9	51.1	0.69	42.1	62.1	39.1	24.8	52.8	0.65	19.8	38.0	69.9	12.5	29.3	1.35
MRF-SFM[9]	51.5	84.6	12.1	45.1	78.1	0.19	32.1	62.3	31.6	36.5	64.3	0.42	47.8	81.1	13.7	31.2	55.9	0.48	33.6	55.9	36.3	22.7	35.1	0.81
IRLS[6, 11]	42.5	78.3	25.2	32.3	65.2	0.40	34.2	53.1	35.5	33.2	55.1	0.65	51.1	79.1	20.5	31.0	61.2	0.45	31.1	51.0	45.4	24.5	36.7	0.78
RobustAlign[7]	42.7	79.2	23.2	33.4	67.3	0.37	34.4	52.2	34.5	34.2	57.2	0.64	**52.3**	78.5	22.5	31.2	**63.3**	0.39	33.8	50.5	43.4	24.7	38.4	0.70
Our approach	**67.2**	**91.3**	**6.7**	**49.3**	**80.1**	**0.17**	**41.1**	**84.4**	**19.6**	**38.5**	**67.3**	**0.29**	51.8	**87.1**	**9.7**	**36.2**	58.9	**0.24**	**39.6**	**63.9**	**21.3**	**26.7**	**39.4**	**0.56**
K^{gt}	69.9	95.1	5.8	53.3	86.1	0.15	44.5	87.3	16.9	40.2	70.1	0.26	55.3	89.8	15.2	39.4	61.9	0.21	40.6	66.1	19.1	28.7	42.1	0.52
3×binsize	59.2	88.3	8.9	42.5	78.4	0.21	35.1	80.1	22.6	32.7	65.4	0.32	47.9	84.9	11.5	33.4	56.4	0.27	35.3	60.8	22.9	25.2	38.1	0.59
2×binsize	65.1	90.2	6.5	48.1	79.3	0.18	39.6	83.3	20.1	36.8	65.7	0.30	50.7	86.2	9.9	34.7	57.3	0.25	38.1	62.7	22.9	24.9	38.1	0.57
0.5×binsize	70.2	90.4	6.5	48.8	79.2	0.16	44.3	82.3	18.9	39.1	66.4	0.30	52.4	86.7	9.9	38.1	58.2	0.24	40.1	64.9	20.5	28.6	40.4	0.58
Vary-root	±4.7	±3.1	±0.3	±4.5	±2.7	±0.02	±3.2	±2.5	±0.5	±5.1	±2.8	±0.02	±3.8	±3.1	±0.3	±4.1	±2.7	±0.01	±3.3	±2.9	±0.4	±3.5	±2.2	±0.02
Bernard et al	30.7	60.8	21.8	38.3	65.4	0.29	27.1	58.4	30.3	28.2	57.3	0.41	52.5	79.4	13.8	23.4	54.2	0.42	25.8	49.3	35.1	15.9	34.4	0.81
Arrigoni et al	28.2	63.1	20.1	40.1	73.2	0.26	32.3	65.1	26.3	29.2	62.6	0.37	44.2	70.2	16.8	23.2	59.1	0.44	29.3	56.7	31.1	18.1	36.3	0.76
Birdal et al	34.2	64.1	19.2	43.1	77.1	0.23	27.6	60.7	29.3	30.1	58.5	0.38	43.7	68.9	17.1	19.7	58.1	0.45	30.2	57.5	29.5	20.5	39.2	0.69
SE-Sync	28.1	57.9	23.4	41.1	72.3	0.25	28.7	57.5	33.3	23.7	55.7	0.43	42.8	67.8	17.9	25.4	62.5	0.38	29.4	59.1	28.4	22.1	39.3	0.67
Torsello et al	24.2	55.2	26.1	39.2	67.8	0.29	26.5	53.1	35.3	21.3	53.9	0.48	39.5	66.1	21.9	22.9	58.4	0.45	27.8	54.9	32.4	17.5	38.3	0.73

Table 2: Benchmark evaluation on our approach and baseline approaches. The columns labeled as $3°$, $30°$, 0.05 and 0.25 tell the portion of the algorithm output that is within these error bracket. The columns labeled as 'mean' tell the mean rotation error or the translation error. Our approach outperforms the three methods that we are comparing with.

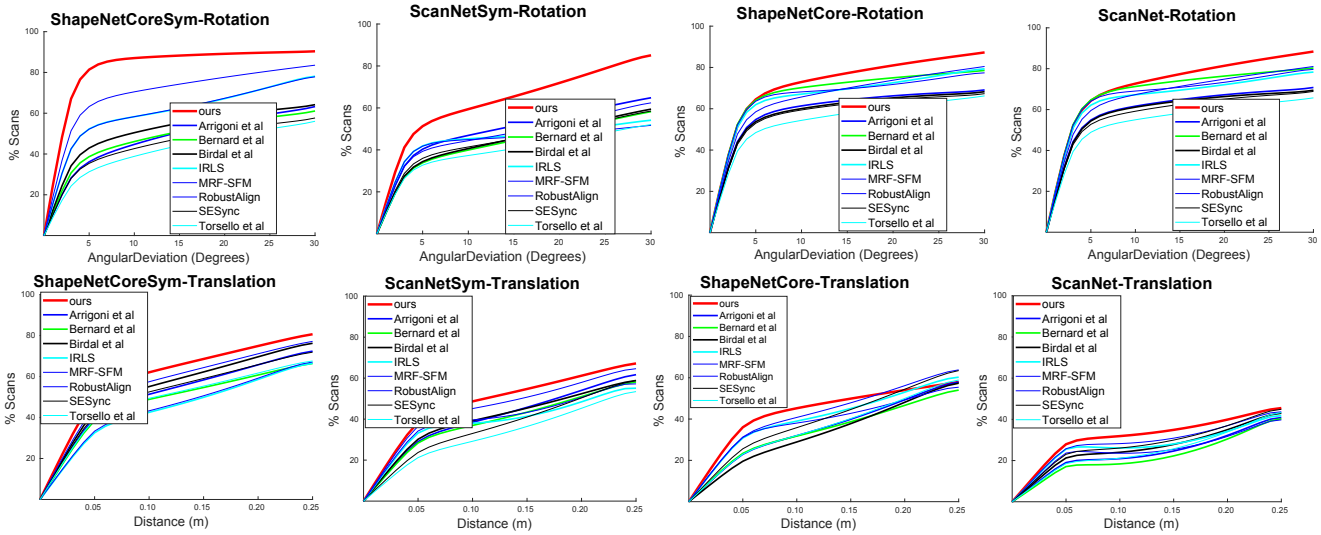

Figure 1: Cumulative distribution functions of our approach and baseline approaches on the four benchmark datasets: ShapeNetCorSym, ScanNetSym, ShapeNetCore and ScanNet.

where w.p. stands for with probability.

It should be noted that matrix X essentially redefines the data matrix that is used to compute the single-best solution. Specifically, the rows and columns correspond the the K possible solutions of each object, and matrix X essentially encode the association between the K best solutions and the input maps.

Now let us calculate the expectation of each block, which gives

$$E[X_{ij}] = 0(1-p_e p_g) + (\frac{1}{K} - \frac{\delta}{K-1})\mathbf{1}\mathbf{1}^T + \frac{K}{K-1}\delta I_k$$
$$= p_e p_g (\frac{1}{K} - \frac{\delta}{K-1})\mathbf{1}\mathbf{1}^T + \frac{K\delta p_e p_g}{K-1} I_k.$$

It follows that the top K eigenvectors of $E[X]$ is given by $\mathbf{1}_n \otimes I_k$, and the top $K+1$ eigenvalues are

$$p_e p_g (\frac{1}{K} + \delta)n, p_e p_g \frac{K\delta}{K-1} n, \cdots, p_e p_g \frac{K\delta}{K-1} n, 1, \cdots, 1, ...$$

It is clear that there is a spectral-gap between the k-th and $k+1$-th eigenvalues of $E[X]$. Now consider X is a perturbation from $E[X]$, since for each block

$$\|X_{ij} - E[X_{ij}]\| \leq 1.$$

It follows that the leading eigenspace spanned by the top K eigenvectors of X is well-approximately by $\mathbf{1}_n \otimes I_k$. Specifically, from Lemma 4.1 of[2], we have the following stability proposition

Lemma B.1. *Let $U \in \mathbb{R}^{nK \times K}$ denotes the leading K eigenvectors of X. With e_i we denote canonical basis vector, then there exist universal constants C_1 and C_2 and a unitary matrix $R \in O(K)$ such that*

$$\max_{1 \leq i \leq n} \|(\boldsymbol{e}_i^T \otimes I_k)(U - \mathbf{1}_n \otimes R)\| \leq \frac{1}{\sqrt{n}} C_1 \frac{\sqrt{n \log(n)}}{\frac{K\delta}{K-1} n p_e p_g}. \tag{2}$$

with probability at least $1 - O(\frac{1}{n_2^C})$.

Note that LemmaB.1 essentially extends the L^∞ stability results of random matrix perturbations to the setting of block matrices.

Since LemmaB.1 provides a universal L^∞ bound, it follows that when projecting $\boldsymbol{e}_1$ onto the space spanned by U, i.e., $UU^T\boldsymbol{e}_1$, each block of $UU^T\boldsymbol{e}_1$ is bounded from $\frac{1}{n}(\mathbf{1}_n\mathbf{1}_n^T) \otimes \boldsymbol{e}_1$ element-wise by $2 \leq \frac{1}{\sqrt{n}} C_1 \frac{\sqrt{n \log(n)}}{\frac{K\delta}{K-1} n p_e p_g}$. In other words, we applying the element-wise to obtain the single-best solution, under the condition that $\frac{\sqrt{n \log(n)}}{\frac{K\delta}{K-1} n p_e p_g} = o(1)$, then the solution obtained by the leading eigen-vector agrees with the underlying ground-truth. This ends the proof.

C. Additional Quantitative Evaluations

Table 2 and Figure 1 present additional quantitative comparisons between our approach and five additional baseline approaches: Bernard et al.[3], Arrigoni et al.[1], Birdal et al.[4], SE-Sync[12], and Torsello et al.[13]. The evaluation protocol remains the same, namely, on ShapeNetCoreSym and ScanNetSym, we compare against the K-best solutions, and on ShapeNetCore and ScanNet, we compare against the single-best solution. Note that Birdal et al.[4] output a probability distribution over the space of synchronizations. We compare against the most-provable solution (c.f.[4]).

These additional baseline approaches are based on convex relaxations, spectral relaxations and MAP-based inferences. As a result, their performance appears to be at best similar to the three baseline approach considered in the main paper, namely, MRF-SFM[10], IRLS[6, 11] and RobustAlign[8].

References

[1] F. Arrigoni, B. Rossi, and A. Fusiello. Spectral synchronization of multiple views in se (3). *SIAM Journal on Imaging Sciences*, 9(4):1963–1990, 2016. 6

[2] C. Bajaj, T. Gao, Z. He, Q. Huang, and Z. Liang. Smac: simultaneous mapping and clustering using spectral decompositions. In *International Conference on Machine Learning*, pages 334–343, 2018. 5

[3] F. Bernard, J. Thunberg, P. Gemmar, F. Hertel, A. Husch, and J. M. Gonçalves. A solution for multi-alignment by transformation synchronisation. In *CVPR*, pages 2161–2169. IEEE Computer Society, 2015. 6

[4] T. Birdal, U. Simsekli, M. O. Eken, and S. Ilic. Bayesian pose graph optimization via bingham distributions and tempered geodesic MCMC. In *NeurIPS*, pages 306–317, 2018. 6

[5] B. Bollobás. The diameter of random graphs. *Transactions of the American Mathematical Society*, 267(1):41–52, 1981. 1

[6] A. Chatterjee and V. Madhav Govindu. Efficient and robust large-scale rotation averaging. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 521–528, 2013. 5, 6

[7] S. Choi, Q.-Y. Zhou, and V. Koltun. Robust reconstruction of indoor scenes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5556–5565, 2015. 5

[8] S. Choi, Q.-Y. Zhou, and V. Koltun. Robust reconstruction of indoor scenes. In *CVPR*, pages 5556–5565. IEEE Computer Society, 2015. 6

[9] D. Crandall, A. Owens, N. Snavely, and D. Huttenlocher. Discrete-continuous optimization for large-scale structure from motion. In *CVPR 2011*, pages 3001–3008. IEEE, 2011. 5

[10] D. Crandall, A. Owens, N. Snavely, and D. Huttenlocher. SfM with MRFs: Discrete-continuous optimization for large-scale structure from motion. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 35(12):2841–2853, December 2013. 6

[11] X. Huang, Z. Liang, C. Bajaj, and Q. Huang. Translation synchronization via truncated least squares. In *Advances in neural information processing systems*, pages 1459–1468, 2017. 5, 6

[12] D. M. Rosen, L. Carlone, A. S. Bandeira, and J. J. Leonard. Se-sync: A certifiably correct algorithm for synchronization over the special euclidean group. *The International Journal of Robotics Research*, 38(2-3):95–125, 2019. 6

[13] A. Torsello, E. Rodolà, and A. Albarelli. Multiview registration via graph diffusion of dual quaternions. In *The 24th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2011, Colorado Springs, CO, USA, 20-25 June 2011*, pages 2441–2448, 2011. 6

[14] Q. Zhou, J. Park, and V. Koltun. Open3d: A modern library for 3d data processing. *CoRR*, abs/1801.09847, 2018. 5