Revisiting Global Translation Estimation with Feature Tracks

Supplementary Material

This supplementary material consists of three sections. Section I provides a detailed derivation of some formulas mentioned in the main paper. Section II offers an in-depth discussion of the HETA method, encompassing implementation details for each step of Algorithm 1 in the main paper, along with a comprehensive understanding of the advantages of using hybrid constraints, explicit 3D points, and cross-product-form objective functions. Section III encompasses supplementary experiments and results not presented in the main paper due to space limitations. These include a comparison of running times for various objective functions and reconstruction results for the 1DSfM dataset using HETA and several state-of-the-art methods.To avoid conflicts with the index of formulas in the main text, the formula index in the supplementary material starts from (12).

I. Formula Derivations

I.1. Formula Derivation for Eq. (7)

For an orthogonal rotation matrix $\boldsymbol{R} \in \mathbb{R}^{3\times3}$, the cross product of two vectors $\boldsymbol{a}, \boldsymbol{b} \in \mathbb{R}^3$ satisfies:

$$\boldsymbol{R}(\boldsymbol{a} \times \boldsymbol{b}) = \boldsymbol{R}\boldsymbol{a} \times \boldsymbol{R}\boldsymbol{b}. \tag{12}$$

From Eq. (12), with known global rotations, the coplanarity constraint in epipolar geometry can be rewritten as:

$$\begin{aligned}
\boldsymbol{X}_{kj} \cdot (\boldsymbol{t}_{ij} \times \boldsymbol{R}_{ij}\boldsymbol{X}_{ki}) &= 0 \\
\Leftrightarrow (\boldsymbol{R}_{ij}\boldsymbol{X}_{ki} \times \boldsymbol{X}_{kj})^T \boldsymbol{t}_{ij} &= 0 \\
\Leftrightarrow (\boldsymbol{R}_{ij}\boldsymbol{X}_{ki} \times \boldsymbol{X}_{kj})^T (\boldsymbol{R}_j\boldsymbol{R}_j^T)\boldsymbol{t}_{ij} &= 0 \\
\Leftrightarrow (\boldsymbol{R}_j^T(\boldsymbol{R}_{ij}\boldsymbol{X}_{ki} \times \boldsymbol{X}_{kj}))^T \boldsymbol{R}_j^T\boldsymbol{t}_{ij} &= 0 \\
\Leftrightarrow (\boldsymbol{R}_j^T\boldsymbol{R}_{ij}\boldsymbol{X}_{ki} \times \boldsymbol{R}_j^T\boldsymbol{X}_{kj})^T \boldsymbol{v}_{ij} &= 0 \\
\Leftrightarrow (\boldsymbol{R}_i^T\boldsymbol{X}_{ki} \times \boldsymbol{R}_j^T\boldsymbol{X}_{kj})^T \boldsymbol{v}_{ij} &= 0 \\
\Leftrightarrow (\boldsymbol{f}_{ki} \times \boldsymbol{f}_{kj}) \cdot \boldsymbol{v}_{ij} &= 0.
\end{aligned} \tag{13}$$

I.2. Formula Derivation for Eq. (3)

From Fig. 2, since two adjacent triangles $\{P_k - t_l - t_r\}$ and $\{P_k - t_l - t_i\}$ have a common edge, according to the sine theorem, we have:

$$||\boldsymbol{P}_K - \boldsymbol{t}_l||_2 = \frac{||\boldsymbol{t}_l - \boldsymbol{t}_r||_2 \cdot \sin\theta_r}{\sin\alpha_{lr}} = \frac{||\boldsymbol{t}_l - \boldsymbol{t}_i||_2 \cdot \sin\theta_i}{\sin\alpha_{li}}. \tag{14}$$

From the second equation in Eq. (14), the ratio of two camera baselines is computed as shown in Eq. (3).

I.3. Formula Derivation for Eq. (4)

Compared to the raw LiGT constraint in [8], we rewrite it in the global coordinate system and give a geometric interpretation to it with the implicit 3D point $\boldsymbol{P}_k$. The rewritten constraint can be transformed into the raw LiGT constraint in the local camera coordinate system with the Eq. (12).

$$\begin{aligned}
||\boldsymbol{P}_k - \boldsymbol{t}_l||_2 &= \frac{||\boldsymbol{t}_l - \boldsymbol{t}_r||_2 \cdot \sin\theta_r}{\sin\alpha_{lr}} \\
&= \frac{||\boldsymbol{f}_{kr} \times (\boldsymbol{t}_l - \boldsymbol{t}_r)||_2}{||\boldsymbol{f}_{kl} \times \boldsymbol{f}_{kr}||_2} \\
&= \frac{||\boldsymbol{f}_{kl} \times \boldsymbol{f}_{kr}||_2 \cdot ||\boldsymbol{f}_{kr} \times (\boldsymbol{t}_l - \boldsymbol{t}_r)||_2}{||\boldsymbol{f}_{kl} \times \boldsymbol{f}_{kr}||_2 \cdot ||\boldsymbol{f}_{kl} \times \boldsymbol{f}_{kr}||_2} \\
&= \frac{(\boldsymbol{f}_{kl} \times \boldsymbol{f}_{kr}) \cdot (\boldsymbol{f}_{kr} \times (\boldsymbol{t}_l - \boldsymbol{t}_r))}{||\boldsymbol{f}_{kl} \times \boldsymbol{f}_{kr}||_2^2} \\
&= \frac{((\boldsymbol{f}_{kl} \times \boldsymbol{f}_{kr}) \times \boldsymbol{f}_{kr}) \cdot (\boldsymbol{t}_l - \boldsymbol{t}_r)}{||\boldsymbol{f}_{kl} \times \boldsymbol{f}_{kr}||_2^2}.
\end{aligned} \tag{15}$$

II. Detailed Discussion of HETA

II.1. Detail Pipeline of HETA

In this section, we provide comprehensive details for each step in the HETA pipeline. To begin, we outline the implementation of relative translation re-estimation. Since the re-estimation of each relative translation is independent, all relative translations can be efficiently estimated in parallel. The algorithm for re-estimating one relative translation is demonstrated below:

Algorithm 2 Relative translation re-estimation

```
Input: Global camera rotations $\boldsymbol{R}_i, \boldsymbol{R}_j$, feature matches set $M_{ij} = \{(X_{il}, X_{jr}), \cdots\}$ for image $i$ and image $j$, where $X_{il}, X_{jr} \in \mathbb{R}^3$ is the coordinate of feature points in local camera coordinate system and $l, r$ are respectively the index of the feature point in two images, and the loss width $\beta$.
Output: Re-estimated relative translation in the global coordinate system $\boldsymbol{v}_{ij}$, a set of filtered feature matches $M'_{ij}$.
Initialize an empty set for normal vectors $N_{ij} = \emptyset$ and an empty set for filtered feature matches $M'_{ij} = \emptyset$;
for $(X_{il}, X_{jr})$ in $M_{ij}$ do
    Calculate feature rays with known global rotations by Eq. (2);
    Calculate the unnormalized normal vector $\boldsymbol{n}_{lr}$ and the parallax angle $\alpha_{lr}$ through the cross product of two feature rays;
    if $\alpha > A$ then
        Insert $\boldsymbol{n}_{lr}$ into $N_{ij}$
    end if
end for
Estimate $\boldsymbol{v}_{ij}$ with all normal vectors in $N_{ij}$ by Eq. (9);
for $n_{lr}$ in $N_{ij}$ do
    if $\frac{\boldsymbol{v}_{ij} \cdot n_{lr}}{||\boldsymbol{v}_{ij} \cdot n_{lr}||_2} < \beta$ and $(X_{il}, X_{jr})$ complies with cheirality constraints then
        Insert $(X_{il}, X_{jr})$ into $M'_{ij}$
    end if
end for
```

Following the re-estimation of relative translations, the one-dimensional projection method in 1DSfM [54] is employed for filtering relative translation outliers with their connection relationship in the view graph. Subsequently, with the filtered feature matches from filtered relative translations, the comprehensive algorithm for constructing and selecting reliable feature tracks is outlined below:

Algorithm 3 Construct and select reliable Feature tracks

```
Input: Filtered feature matches $M'_{ij}$, the cover times for each images $N$.
Output: Set of selected feature tracks $P$.
Construct feature tracks $F = \{F_1, \cdots, F_M\}$ with $M'_{ij}$, where each feature track is a set of feature rays $F_k = \{f_{k1}, \cdots, f_{ki}, \cdots\}$;
Sort all feature tracks based on their maximum parallax angles in descending order to obtain $F_{order} = \{F_1, \cdots, F_M\}$
Initialize a set for each image $I = \{I_1, \cdots, I_i, \cdots\}$, where $I_i = 0, \forall i \in V$, $I_i$ is the times that image $i$ is covered;
for $F_k$ in $F_{order}$ do
    if IsEmpty($I$) then
        break
    end if
    if $F_k$ can cover some images in $I$ then
        Insert $F_k$ into $P$
        for $f_{ki}$ in $F_k$ do
            $I_i \leftarrow I_i + 1$
            if $I_i >= N$ then
                Delete $I_i$ from set $I$
            end if
        end for
    end if
end for
```

With the selected feature tracks P and filtered re-estimated relative translations, we construct a view track graph. Since the camera-to-camera and camera-to-point constraints are equivalent in a mathematical expression, for simplicity, we denote both the normalized relative translations and feature rays as $\boldsymbol{s}_{ij}, \forall ij \in E_v \bigcup E_p$ and denote the positions of estimated cameras and 3D points as $\boldsymbol{s}_i, \forall i \in V \bigcup P$. The algorithm for unbiased optimization is presented as below:

Algorithm 4 IRLS-BCD solver for unbiased optimization

```
Input: Initial camera and 3D points $\boldsymbol{s}_i, \forall i \in V \bigcup P$, normalized image observations $\boldsymbol{s}_{ij}, \forall ij \in E_v \bigcup E_p$
Output: Camera and 3D points $\boldsymbol{s}_i, \forall i \in V \bigcup P$.
Initialize $W_{ij}, \forall ij \in E_v \bigcup E_p$; Set $n = 0$;
while $n <$ IRLSIter AND not converged do
    $m = 0$;
    while $m <$ BCDIter do
        $\hat{\boldsymbol{s}}_{ij} \leftarrow \frac{\boldsymbol{s}_i - \boldsymbol{s}_j}{||\boldsymbol{s}_i - \boldsymbol{s}_j||_2}$;
        if $\boldsymbol{s}_{ij} \cdot \hat{\boldsymbol{s}}_{ij} \geq 0$ then
            $\boldsymbol{H}(\hat{\boldsymbol{s}}_{ij}) \leftarrow ||\boldsymbol{s}_{ij \times} \hat{\boldsymbol{s}}_{ij}||_2$
        else
            $\boldsymbol{H}(\hat{\boldsymbol{s}}_{ij}) \leftarrow ||\boldsymbol{s}_{ij}||_2$
        end if
        Update $\boldsymbol{s}_i$ : Solve the reweighted least squares objective function in Eq. (11) by Cholesky decomposition;
        $m = m + 1$;
    end while
    $W_{ij} \leftarrow \phi(\boldsymbol{H}(\hat{\boldsymbol{s}}_{ij}))$;
    $n = n + 1$;
end while
```

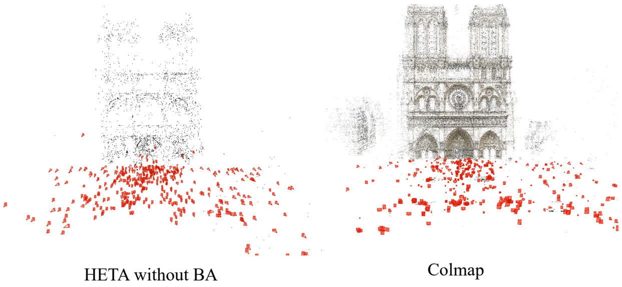


Figure 10. Comparison of results between the HETA method without BA and the ground truth from Colmap.

For the most cases when $\boldsymbol{s}_{ij} \cdot \hat{\boldsymbol{s}}_{ij} \geq 0$, we penalize the sine value of the error angles between known normalized vector $\boldsymbol{s}_{ij}$ and temporary estimated direction vector $\boldsymbol{H}(\hat{\boldsymbol{s}}_{ij})$ through the L_2 norm of $\boldsymbol{s}_{ij \times} \hat{\boldsymbol{s}}_{ij}$. However, when $\boldsymbol{s}_{ij} \cdot \hat{\boldsymbol{s}}_{ij} < 0$, we utilize the L_2 norm of normalized vector $\boldsymbol{s}_{ij}$, which equals 1 for a larger penalization. For the IRLS-BCD method, we set IRLSIter=30 & BCDIter=5 as used in [59] and utilize a Cauchy loss with scale $\beta = \sin 3°$. With a relatively accurate initial solution from L_1 norm optimization, this unbiased optimization can be approximately considered as minimizing the error angles of image observations. Especially for the feature rays, when the camera global rotations are relatively accurate, this unbiased optimization can achieve an effect similar to minimizing the reprojection error in bundle adjustment.

II.2. Output of HETA

After robust L_1 norm optimization and unbiased L_2 norm optimization in HETA method, both the positions of cameras and sparse 3D points for selected feature tracks are estimated. We take the Notre Dame data in the 1DSfM dataset as an example. As shown in Fig. 10, the estimated sparse point cloud represents the general structure of the scene, which shows the effectiveness of our method for estimating cameras and 3D points simultaneously.

II.3. Exploring the Use of Hybrid Constraints

There are three primary advantages to employing hybrid constraints. First, compared to the raw view graph with solely camera-to-camera constraints, the addition of constraints from 3D points ensures that nearly all cameras in the view track graph are part of a parallax rigid graph. This leads to the registration of more cameras with hybrid constraints. Second, while feature rays, serving as raw information in images, exhibit higher precision than estimated relative translations, they also have a higher outlier ratio than relative translations. In contrast, relative translations provide more direct and stringent constraints than feature rays. Therefore, a balance between accuracy and robustness is achieved by using all relative translations and selecting reliable feature tracks. Third, when the parallax angles of the

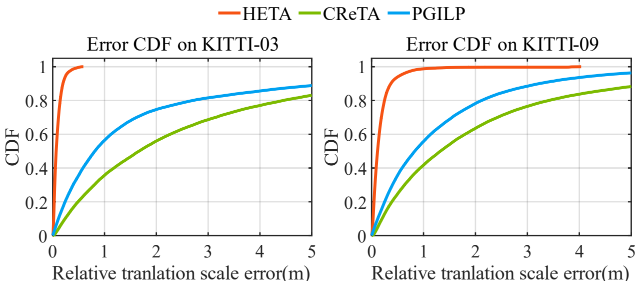


Figure 11. Cumulative distribution functions of relative translation scale errors among the HETA method, CReTA method, and PGILP method on a subset of the KITTI dataset. The HETA method utilizes hybrid constraints, the CReTA method employs pure camera-to-camera constraints, and the PGILP method relies on pure camera-to-point constraints.

feature rays are low, the 3D points become unstable. This can be addressed by selecting feature tracks with larger parallax angles. Similarly, when the camera motion trajectories are approximately collinear, the camera positions become unstable. This can be mitigated by incorporating constraints from 3D points rather than filtering relative translations.

We compare the scale error cumulative distribution function of relative translations estimated by three methods using different constraints, as shown in Fig. 11. We find that the accuracy of scales estimated by the HETA method using hybrid constraints is higher than the other two methods with pure constraints, supporting our conclusion.

II.4. Exploring the Use of Explicit Method

The representation of implicit 3D points for each feature track relies on two base cameras with the largest parallax angle, making it sensitive to the precision of the global rotations of these base cameras. On one hand, as two base cameras have the largest parallax angle, their relative rotation generally exhibits a larger error than two cameras with a smaller parallax angle. On the other hand, if the correspondence feature match of base cameras is an outlier, the represented 3D point also becomes an outlier, leading to the failure of constraints for the entire feature track.

In contrast, explicit 3D points for each feature track are estimated using all feature rays, which generally exhibit higher accuracy. Moreover, an unbiased angle-based objective function can be formulated with explicit 3D points. In summary, the HETA method with hybrid explicit constraints generally outperforms other methods.

II.5. The Necessity of Cross-product-form

When the angular errors of the relative translation direction are below 90°, the cross-product-form objective function exhibits better convergence than the scale-form by avoiding redundant variables. This advantage becomes more pronounced on large-scale datasets when incorporating camera-to-point constraints. As shown in Tab. 3, the Cross-HE method outperforms the Scale-HE method significantly as the number of images in the dataset increases. Although there are other types of non-linear distance metric methods mentioned in [54], the cross-product-form is utilized in HETA for better convergence.

	Raw view graph		1DSfM Filter		Re-estimation		Re-estimation + 1DSfM Filter	
data	N_o	$R_o(\%)$	N_o	$R_o(\%)$	N_o	$R_o(\%)$	N_o	$R_o(\%)$
00	1878	0.6	1071	0.4	610	0.2	373	0.1
01	1734	1.7	174	0.2	73	0.1	31	0.1
02	722	0.3	499	0.2	481	0.2	479	0.2
03	29	0.1	10	0.1	11	0.1	0	0
04	44	0.3	16	0.1	0	0	0	0
05	1244	0.7	501	0.3	12	0.1	9	0.1
06	1329	1.8	158	0.3	39	0.1	5	0.1
07	1868	2.3	620	0.1	66	0.1	7	0.1
08	446	0.2	110	0.1	6	0.1	3	0.1
09	310	0.4	249	0.3	199	0.3	195	0.3
10	23	0.1	5	0.1	1	0.1	1	0.1
ALM	387	1.4	244	0.9	173	0.7	104	0.5
ELS	433	4.3	175	1.9	81	1.1	39	0.7
GDM	376	2.9	289	2.4	186	1.6	137	1.5
MDR	137	6.3	122	5.7	67	3.4	49	2.7
MND	475	2.0	251	1.1	153	0.7	114	0.6
ND	461	1.3	301	0.9	182	0.5	134	0.4
NYC	218	2.7	167	2.1	80	1.1	53	1.0
PDP	374	2.9	278	2.3	145	1.2	104	1.1
PIC	3434	4.6	1820	2.8	994	1.5	463	1.3
ROF	1408	5.9	709	3.6	575	2.8	262	2.2
TFG	8738	5.0	4451	3.0	2872	1.9	1364	1.7
TOL	248	2.3	153	1.6	104	1.3	54	0.9
USQ	1967	14.0	912	8.0	419	4.7	204	3.2
VNC	931	2.7	525	1.7	384	1.2	200	0.8
YKM	366	4.0	248	3.0	150	1.9	82	1.5

Table 4. Changes in the relative translation outliers with angular errors above 90° in the view graph after using re-estimation and the 1DSfM filtering method. N_o and R_o respectively denote the number and the ratio of outliers.

II.6. The Availability of Cross-product-form

Even when the angular errors of the relative translation direction exceed 90°, the inequality constraints for the cross-product-form objective function can offer an incorrect feasible region. However, the ratio of these large angular errors is very low. Changes in the number and the ratio of these large outliers in the view graph after re-estimation and the 1DSfM filter are shown in Tab. 4. We observe a significant decrease in the number of these large relative translation outliers after relative translation re-estimation. Following the 1DSfM filter, the number of outliers decreases further. Particularly for the KITTI dataset, the outlier ratios are below 0.3%. Through experimental results, these outliers do not hinder HETA as the top-performing method. Therefore, the cross-product-form remains applicable.

II.7. ADMM Method for L_1 norm Optimization

To obtain a fast convergence, ADMM with iterative soft-thresholding algorithm (ISTA) [6] is utilized to estimate cameras and 3D points under L_1 norm. Eq. (10) is equivalent to a least absolute deviations problem:

$$\min_{x} |\boldsymbol{A}x|_1, \quad s.t. \quad \boldsymbol{B}x \geq b, \tag{16}$$

where x denotes the vector comprising all cameras and points to be optimized, $\boldsymbol{A}$ denotes the matrix formed by relative translations and feature rays in a cross-product metric, $\boldsymbol{B}$ denotes the matrix formed by the relative translations in a dot-product metric, b denotes the vector with all elements equal to 1. Eq. (16) can be written in ADMM form as:

$$\min_{x,z_1,z_2} |z_1|_1, \quad s.t. \quad z_1 = \boldsymbol{A}x, \quad z_2 = \boldsymbol{B}x - b \geq 0. \tag{17}$$

Let $\boldsymbol{M}$ denote $[\boldsymbol{A}^T, \boldsymbol{B}^T]^T$, let n denote $[0, b^T]^T$, and let z denote $[z_1^T, z_2^T]^T$ for convenience. With the parameter ρ, the augmented Lagrangian function is formulated below:

$$L_\rho(x, z, u) = |z_1|_1 + (\rho/2)||\boldsymbol{M}x - n - z + u||_2^2 - (\rho/2)||u||_2^2 \tag{18}$$

The ADMM update of Eq. (18) can be expressed as:

$$\begin{aligned} x^{k+1} &\leftarrow (\boldsymbol{M}^T\boldsymbol{M})^{-1}\boldsymbol{M}^T(n + z^k - u^k) \\ z_1^{k+1} &\leftarrow S_{1/\rho}(\boldsymbol{A}x^{k+1} + u^k) \\ z_2^{k+1} &\leftarrow max\{\boldsymbol{B}x^{k+1} - b + u^k, 0\} \\ u^{k+1} &\leftarrow u^k + \boldsymbol{M}x^{k+1} - z^{k+1} - n, \end{aligned} \tag{19}$$

where the soft thresholding operator $S_{1/\rho}(\cdot)$ is defined as:

$$S_{1/\rho}(a) = (a - 1/\rho)_+ - (-a - 1/\rho)_+ \tag{20}$$

II.8. Inequality Constraint v.s. Equality Constraint

Similar to [59], we conduct two experiments in a four-camera 2D case, and all their pairwise relative directions are observed with a random $-3° \sim 3°$ noise. For each experiment, we compare three methods, including RevisedLUD with equality constraints [59], HETA-L1 with equality constraints, and HETA-L1 with inequality constraints. As shown in above Fig. 12 (a), despite the long baselines of camera C4, its position estimations by all three methods are similar and do not exhibit significant gravitation towards cameras 1-3. Fig. 12 (b) further corroborates that there is no shrinkage effect on long baselines for all three methods. Moreover, for HETA, the optimization matrix of inequality constraints is more sparse than that of equality constraints.

III. Additional Results

III.1. Compared to BATA and Colmap

We compare HETA with global method BATA and incremental Colmap on three large-scale 1DSfM datasets and three large-scale KITTI datasets, as shown in Tab. 5. From these comparisons, it is clear that HETA outperforms both COLMAP and BATA on the KITTI datasets and demonstrates greater robustness and accuracy than BATA on the 1DSfM datasets. Additionally, both global methods exhibit greater efficiency compared to the incremental COLMAP.

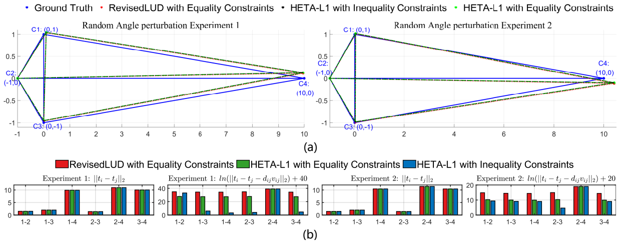


Figure 12. A toy example to show no shrinkage effect. (a) True and estimated camera locations of two random experiments (for better shape comparison, the four sets of estimated camera positions are aligned at $C2(-1, 0)$ and normalized to have the same scale of edge $C1 - C3$. (b) $||t_i - t_j||_2$ and $ln||t_i - tj - d_{ij}\boldsymbol{v}_{ij}||_2$ of two experiments plotted against edge $i - j$ between Ci and Cj.

Data	BATA[59]				HETA				COLMAP[44]			
Name	$\tilde{e}$	$\bar{e}$	N_c	T	$\tilde{e}$	$\bar{e}$	N_c	T	$\tilde{e}$	$\bar{e}$	N_c	T
KITTI-00	13.3	27.9	**9082**	23	**2.4**	7.3	**9082**	51	2.9	**4.6**	**9082**	9498
KITTI-02	25.7	28.5	**9322**	24	**4.6**	**6.5**	**9322**	51	13.3	29.9	**9322**	2415
KITTI-08	22.4	32.6	**8142**	21	**5.6**	**6.9**	**8142**	40	8.2	21.9	**8142**	4032
1DSfM-PIC	**0.1**	0.5	1775	4	**0.1**	**0.4**	1807	11	-	-	**1838**	422
1DSfM-ROF	**0.1**	**0.1**	893	5	**0.1**	**0.1**	907	12	-	-	**918**	184
1DSfM-TFG	1.0	2.5	3859	12	**0.7**	**2.4**	3951	25	-	-	**3989**	1242

Table 5. The notation $\tilde{e}$ and $\bar{e}$ respectively denotes the median and mean distance error, N_c is the number of registered images, and T shows the running times in minutes.

III.2. Running Time Comparison

We compared the running time on an Ubuntu 20.04.5 LTS platform, with 62 GB memory and 12th Gen Intel(R) Core(TM) i7-12700 CPU @ 2.10 GHz, 20 cores. The comparison of the running time for the L_1 norm optimization of various objective functions is shown in Tab. 7. While the efficiency of hybrid methods is slightly lower than methods using pure relative translations, their efficiency is significantly faster than methods using pure feature tracks. The comparison of different existing methods for the whole optimization is shown in Tab. 8.

III.3. Ablation Experiments of 1DSfM Dataset

The detailed results by method with various objective functions for the 1DSfM dataset are presented in Tab. 6. After unbiased optimization, we observe that the results from Scale-HE-L_2 and Cross-HE-L_2 are comparable. Since Cross-HE method outperforms Scale-HE method obviously, the cross-product-form objective function has wider applicability for both unordered dataset and Sequence dataset. Moreover, hybrid methods clearly outperform other pure methods. Reconstruction results for the 1DSfM dataset estimated by the HETA method are displayed in Fig. 13. Results estimated by other state-of-the-art methods are shown in Fig. 14.

1DSfM	Scale-PT		Cross-PT		Scale-PFI		Scale-HE-L_1		Scale-HE-L_2		Cross-HI		Cross-HE-L_1		Cross-HE-L_2	
Name	$\tilde{e}$	$\bar{e}$	$\tilde{e}$	$\bar{e}$	$\tilde{e}$	$\bar{e}$	$\tilde{e}$	$\bar{e}$	$\tilde{e}$	$\bar{e}$	$\tilde{e}$	$\bar{e}$	$\tilde{e}$	$\bar{e}$	$\tilde{e}$	$\bar{e}$
ALM	**0.5**	1.3	**0.5**	1.7	**0.5**	1.6	**0.5**	**1.2**	**0.5**	**1.2**	**0.5**	**1.2**	**0.5**	**1.2**	**0.5**	**1.2**
ELS	2.4	4.2	2.5	4.3	2.6	4.7	2.2	**3.7**	**2.1**	**3.7**	2.6	4.2	2.4	3.9	**2.1**	3.8
GDM	3.7	9.5	5.0	10.1	4.5	12.1	2.4	13.1	2.3	13.0	2.8	**9.6**	2.8	10.4	**2.2**	10.1
MDR	1.7	9.8	2.0	10.4	1.9	12.9	**1.4**	**9.1**	**1.4**	**9.1**	**1.4**	9.7	**1.4**	9.7	**1.4**	9.6
MND	**0.5**	1.1	0.6	1.1	0.6	1.2	**0.5**	**1.0**	**0.5**	**1.0**	0.6	1.1	**0.5**	**1.0**	**0.5**	**1.0**
ND	**0.3**	1.4	0.4	2.1	0.6	1.8	**0.3**	**0.9**	**0.3**	**0.9**	**0.3**	1.5	**0.3**	1.4	**0.3**	**0.9**
NYC	0.6	1.9	1.0	2.6	0.9	2.4	**0.5**	**1.6**	**0.5**	**1.5**	1.0	2.1	1.0	2.2	**0.5**	**1.5**
PDP	1.2	3.4	1.3	3.7	**1.1**	3.3	**1.1**	**2.9**	**1.1**	**2.9**	1.2	3.0	**1.1**	**2.9**	**1.1**	**2.9**
PIC	0.8	**1.7**	1.0	1.9	1.0	2.3	0.8	1.8	**0.7**	1.8	0.9	1.8	0.9	1.9	**0.7**	**1.7**
ROF	2.5	4.8	3.4	7.0	3.2	7.0	1.6	4.0	**1.2**	3.5	1.7	3.7	2.0	4.1	**1.2**	**3.3**
TFG	3.8	6.3	3.5	5.9	4.2	8.7	2.9	6.2	**2.6**	5.9	3.1	5.9	3.3	6.4	**2.6**	**5.8**
TOL	2.9	8.4	4.2	12.3	3.1	8.8	2.3	5.4	2.2	4.9	2.5	5.5	2.5	4.9	**2.1**	**4.5**
USQ	4.3	8.3	4.5	9.0	5.6	9.7	3.7	7.2	**3.4**	**7.0**	4.3	8.6	4.2	7.6	3.6	7.2
VNC	2.6	8.9	2.1	6.7	3.4	13.8	1.8	4.5	1.8	4.4	1.8	4.9	1.8	4.1	**1.7**	**4.0**
YKM	1.4	2.3	1.5	2.4	1.4	3.0	1.2	2.4	**1.1**	2.2	**1.1**	2.4	**1.1**	2.4	**1.1**	**2.1**

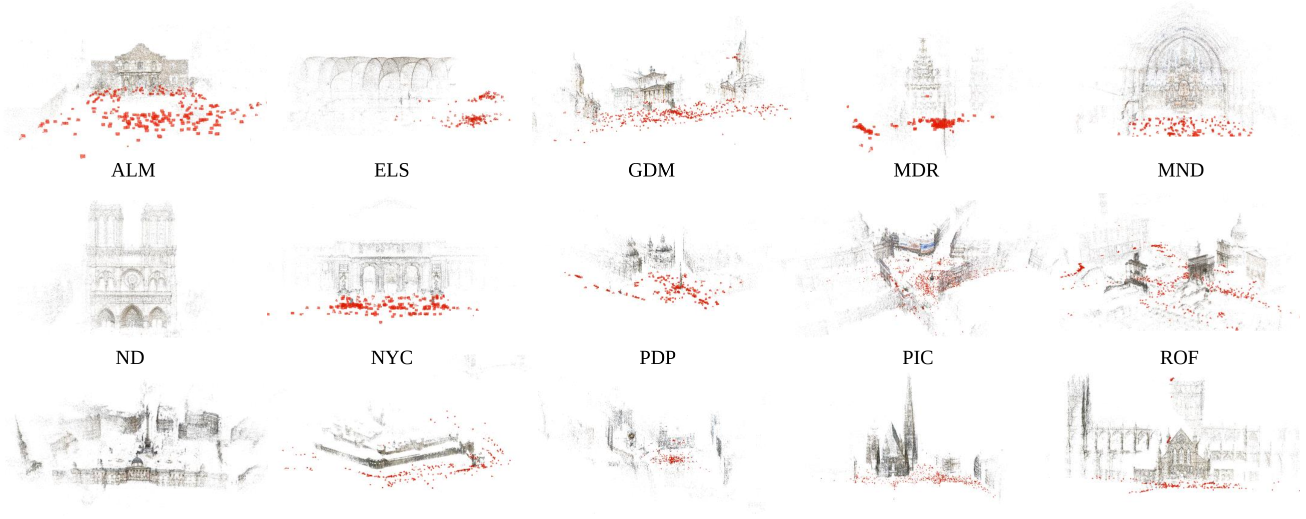
Table 6. Camera position errors produced by applying various objective functions on the 1DSfM dataset. The best results are shown in bold and the second-best results are underlined.

Data	00	01	02	03	04	05	06	07	08	09	10	ALM	ELS	GDM	MDR	MND	ND	NYC	PDP	PIC	ROF	TFG	TOL	USQ	VNC	YKM
Scale-PT	2.95	1.31	2.67	0.84	0.59	2.01	0.97	1.02	2.34	0.99	0.96	0.79	0.28	0.65	0.03	0.76	0.95	0.26	0.48	1.20	0.74	2.19	0.42	0.42	0.90	0.35
Cross-PT	2.68	0.80	2.46	0.86	0.60	1.89	0.91	0.97	2.16	0.94	0.93	0.77	0.23	0.56	0.10	0.73	0.91	0.24	0.44	1.20	0.71	2.02	0.34	0.52	0.93	0.35
Scale-PFI	83.42	32.36	95.47	30.25	6.07	56.50	18.85	26.02	99.81	33.03	29.19	14.63	6.26	10.97	3.20	11.25	17.15	8.46	4.97	24.21	23.72	44.29	14.58	8.54	22.57	11.68
Scale-HE	30.4	5.85	29.56	5.90	1.85	18.96	7.18	8.36	24.67	9.22	8.12	2.42	0.99	1.94	0.82	1.75	2.71	1.33	1.19	5.75	2.30	12.99	1.65	1.73	3.12	1.52
Cross-HI	22.59	4.67	22.12	4.54	1.66	14.04	5.49	6.40	18.87	6.94	6.23	2.16	1.03	1.73	0.75	1.60	2.29	1.27	1.12	4.16	2.50	8.80	1.60	1.56	2.47	1.37
Cross-HE	23.19	4.54	22.45	4.26	1.61	13.18	5.43	5.84	17.96	6.81	5.92	1.87	0.94	1.61	0.72	1.51	2.17	1.19	1.07	4.32	2.39	9.49	1.49	1.50	2.33	1.30

Table 7. Runtime comparison of L_1 norm optimization using various objective functions on KITTI and 1DSfM datasets in minutes.

Data	00	01	02	03	04	05	06	07	08	09	10	ALM	ELS	GDM	MDR	MND	ND	NYC	PDP	PIC	ROF	TFG	TOL	USQ	VNC	YKM
LUD	9.52	2.60	10.00	2.66	0.86	6.63	2.42	2.92	9.73	3.65	41.73	5.22	0.69	1.82	0.25	2.40	3.59	0.97	1.32	2.21	6.78	4.39	1.27	1.46	4.62	0.92
CReTA	17.68	4.19	18.04	3.69	1.25	10.56	20.18	4.03	16.82	4.79	4.16	5.67	1.24	2.44	0.85	2.94	4.77	1.75	1.50	3.66	7.97	11.01	2.07	1.85	5.68	1.61
LiGT	16.41	1.97	9.47	2.11	0.34	13.13	2.28	1.98	6.79	3.15	1.74	0.60	0.39	0.29	0.11	1.39	1.59	0.13	0.11	4.49	1.30	14.67	0.15	1.15	0.80	1.40
PGILP	89.76	32.36	102.42	31.67	6.29	60.93	20.18	27.64	106.48	35.44	31.09	16.74	6.09	11.12	3.12	11.86	17.76	8.40	5.20	25.13	26.40	42.74	14.16	8.85	26.01	11.20
1DSfM	32.14	2.31	31.59	4.42	0.52	14.33	4.88	6.16	22.52	7.83	4.86	1.22	0.54	0.65	0.10	1.44	1.80	0.63	0.83	4.62	2.17	3.87	0.87	1.51	2.25	0.70
HETA	51.05	11.21	51.19	9.32	2.77	28.93	10.67	12.45	40.03	14.23	12.72	8.21	1.90	5.06	1.17	4.56	7.26	2.93	2.33	11.15	12.74	23.996	4.02	3.91	9.82	3.33

Table 8. Runtime comparison of different methods on KITTI and 1DSfM datasets in minutes.



ALM ELS GDM MDR MND

ND NYC PDP PIC ROF

TFG TOL USQ VNC YKM

Figure 13. Reconstruction results for the 1DSfM dataset estimated by HETA method.

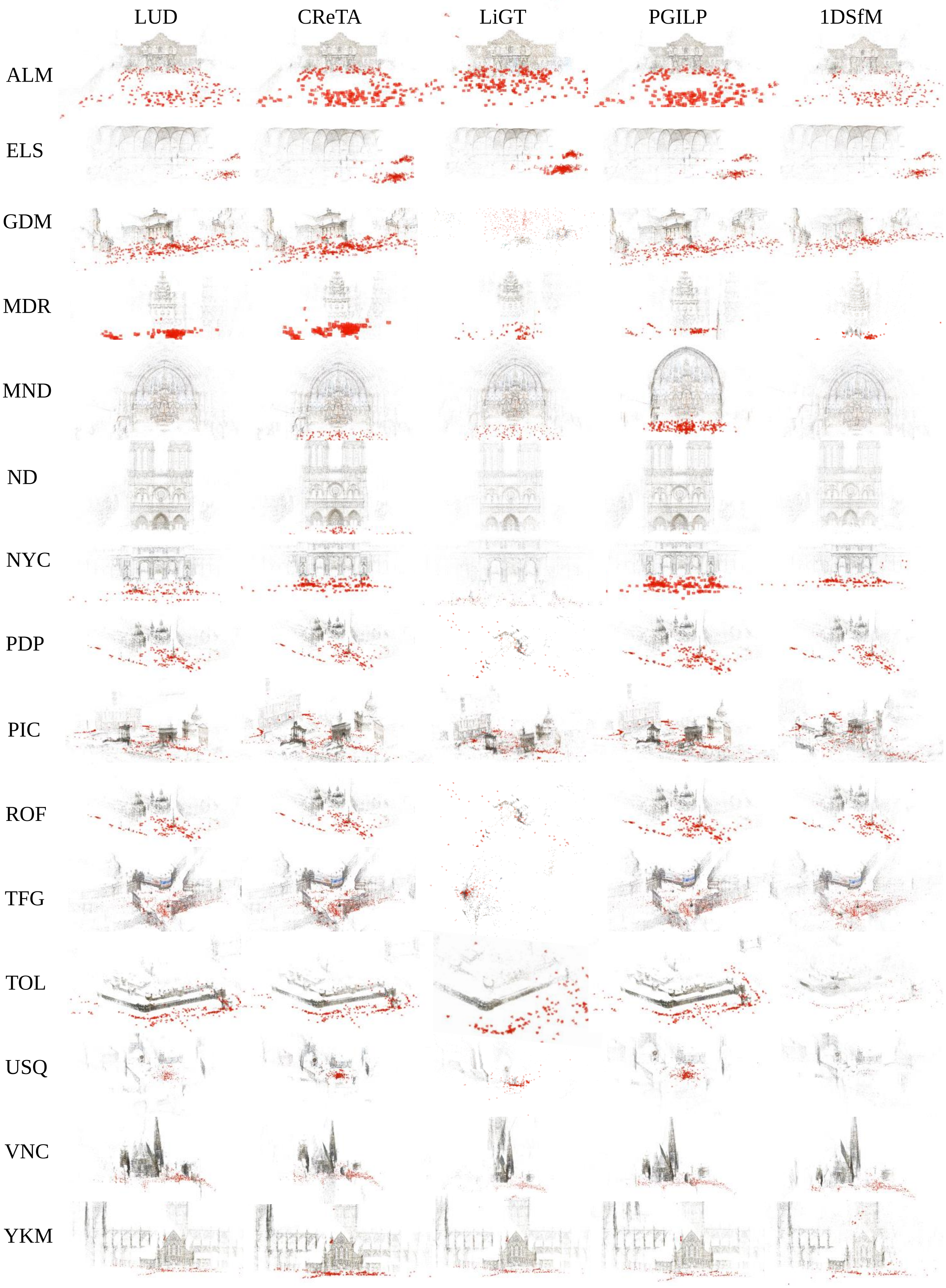


Figure 14. Reconstruction results for the 1DSfM dataset estimated by several state-of-the-art methods.