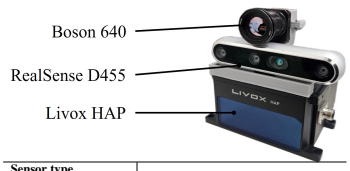
PTC-Depth: Pose-Refined Monocular Depth Estimation with Temporal Consistency

Supplementary Material

1. Dataset and Sensor Specifications

Our custom dataset is collected using multiple sensors mounted on a *Hunter SE* mobile platform, as shown in Fig. 1. The RGB camera is an *Intel RealSense D455*, capturing images at a resolution of 640×480 at 60 fps. The thermal camera is a *FLIR Boson 640*, providing LWIR data at a resolution of 640×512 at 60 fps. Depth ground truth is obtained using a *Livox HAP* LiDAR. In addition, wheel odometry from the platform is used to estimate the vehicle motion. The specifications of each sensor are summarized in Fig. 1.



Sensor type (Model)	Frame rate	Specification
RGB Camera (Intel RealSense D455)	60 fps	640×480 FOV: 90°(H)×65°(V) Global Shutter
Thermal Camera (FLIR Boson 640)	60 fps	640×512 FOV: 95°(H)×82°(V) LWIR (8–14 μm)
LiDAR (Livox HAP)	10 fps	Range: 150m FOV: 120°(H)×25°(V)

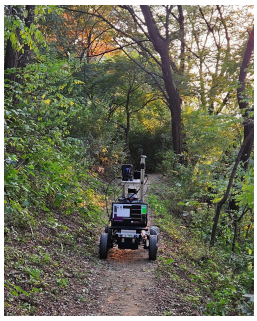


Figure 1. **Data collection platform.** (Left) Sensor module with FLIR Boson 640, Intel RealSense D455, and Livox HAP LiDAR. (Right) Hunter SE platform in a forest environment.

2. Overview

This supplementary document provides implementation details and additional analyses not included in the main paper. We first describe our custom dataset and sensor specifications (Section 1), followed by the procedures used for robust motion estimation (Section 3), Sampson residual computation and Bayesian variance modeling (Section 4), superpixel-wise scale refinement (Section 5), an ablation on odometry accuracy (Section 6), and runtime analysis (Section 7). Additional limitations are discussed in Section 8. The overall process is shown in Fig. 2.

3. Robust Motion Estimation

This section provides additional implementation details for the motion-estimation stage. The main paper introduces the motion-field formulation; here, we describe how correspondences are selected, how flow–motion consistency is evaluated, and how rotation-dominant frames are handled in practice.

3.1. Normalized motion-field residual

Let $\mathbf{f}(x)$ denote the observed optical flow at pixel x and $\dot{\mathbf{p}}(x)$ the motion-field prediction for parameters $(\boldsymbol{\Omega}, \boldsymbol{T})$. We first compute the pixel-domain discrepancy

$$r(x) = \|\mathbf{f}(x) - \dot{\mathbf{p}}(x)\|_2 . \tag{1}$$

Because the magnitude of $\mathbf{f}(x)$ varies across the image, we normalize this residual as

$$e(x) = \frac{r(x)}{\max(\|\mathbf{f}(x)\|_2,\ \tau)}. \tag{2}$$

Because $\|\mathbf{f}(x)\|_2$ can be very small in near-static regions, we set τ as a small lower bound (e.g., $\tau = 1$ pixel) to ensure numerical stability. This relative residual provides a scale-invariant measure of consistency and yields a stable inlier criterion across both large-motion and near-static regions.

3.2. Spatially and depth-balanced sampling

To avoid bias toward locally dense textures, we draw correspondences from a stratified distribution: (i) the image is divided into coarse spatial regions; (ii) candidate pixels are grouped into a few depth intervals; and (iii) each spatial–depth group contributes at most a fixed number of samples. All selected pixels must have finite flow, finite inverse depth, and non-negligible flow magnitude. This produces a well-conditioned set of correspondences supporting motion estimation across the full field of view.

3.3. Adaptive RANSAC with directional consistency

RANSAC evaluates hypotheses using the normalized residual $e(x)$. Two additional robustness mechanisms are employed.

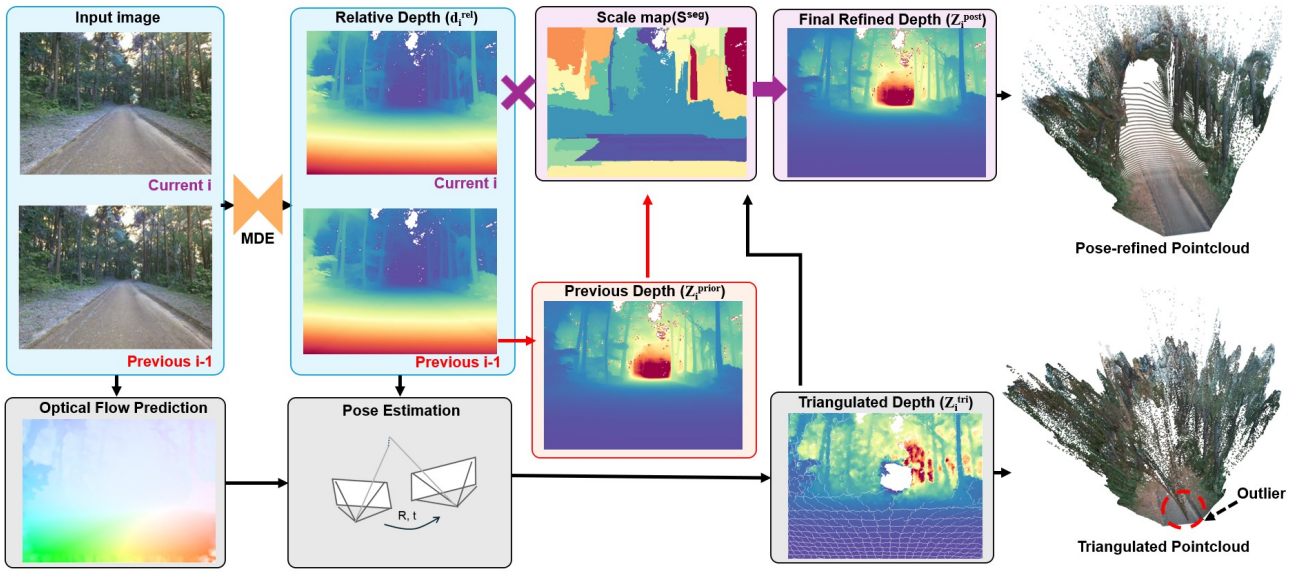


Figure 2. **Overall Process of Our Proposed Algorithm**

Directional consistency. For sufficiently large flows, we measure the angular deviation

$$\Delta\theta(x) = \arccos\left(\frac{\mathbf{f}(x)\cdot\dot{\mathbf{p}}(x)}{\|\mathbf{f}(x)\|_2\,\|\dot{\mathbf{p}}(x)\|_2}\right). \tag{3}$$

Correspondences with abnormally large deviations are rejected using a robust threshold derived from the distribution of $\Delta\theta(x)$. This removes flow outliers whose direction is inconsistent with rigid motion.

Adaptive residual threshold. The inlier threshold η for $e(x)$ is initialized from robust statistics of the residual distribution:

$$\eta_0 = \text{median}(e) + \lambda\,\text{MAD}(e), \tag{4}$$

where $\text{MAD}(e) = \text{median}(|e - \text{median}(e)|)$ and λ controls the tightness of the criterion. After each hypothesis, η is adjusted: if the inlier ratio falls below a target, η is relaxed; otherwise it is tightened. Hypotheses are scored by both inlier count and spatial coverage, favoring solutions supported across the image.

3.4. IRLS refinement

After RANSAC selects the best hypothesis, we refine the parameters $(\boldsymbol{\Omega}, \boldsymbol{T})$ using a small number of iteratively reweighted least squares (IRLS) iterations. Each inlier pixel x is assigned a Huber weight

$$w(x) = \begin{cases} 1, & e(x) \le \eta, \\ \eta\,/\,e(x), & e(x) > \eta, \end{cases} \tag{5}$$

where η is the inlier threshold from the RANSAC stage. At each iteration, the weighted linear system derived from the motion-field equation (Eq. 1 of the main paper) is solved to update $(\boldsymbol{\Omega}, \boldsymbol{T})$, and the weights are recomputed. This refinement stabilizes the estimated rotation and translation and suppresses residual outliers that survive RANSAC sampling.

3.5. Inlier validation and flow fusion

Given the final estimate $(\boldsymbol{\Omega}^*, \boldsymbol{T}^*)$, each pixel is validated using both $e(x)$ and $\Delta\theta(x)$. Pixels satisfying both criteria retain their observed flow; others are replaced with the motion-field prediction. This fused flow prevents a small set of erroneous flows from influencing triangulation or scale estimation.

4. Triangulation and Bayesian Fusion Details

This section provides the closed-form expressions for the Sampson residual map and the consistency score that are referenced in the main paper but omitted for brevity.

4.1. Sampson Residual Map

Given the estimated relative pose $(\boldsymbol{\Omega}, \boldsymbol{T})$, we construct the fundamental matrix

$$F = K^{-\top}[\boldsymbol{T}]_\times \boldsymbol{\Omega}\, K^{-1}, \tag{6}$$

where $[\boldsymbol{T}]_\times$ denotes the skew-symmetric matrix of $\boldsymbol{T}$ and K is the camera intrinsic matrix.

For each correspondence $(\mathbf{x}_{i-1}, \mathbf{x}_i)$ between frames

$i-1$ and i, the Sampson residual is computed as

$$\rho(x) = \frac{\left(\mathbf{x}_i^\top F\, \mathbf{x}_{i-1}\right)^2}{(F\,\mathbf{x}_{i-1})_1^2 + (F\,\mathbf{x}_{i-1})_2^2 + (F^\top \mathbf{x}_i)_1^2 + (F^\top \mathbf{x}_i)_2^2}, \tag{7}$$

where $(\cdot)_j$ selects the j-th component. Each $\rho(x)$ is assigned to its corresponding pixel in frame i, forming the *Sampson residual map*. This map is used both to derive the per-pixel observation variance V^{obs} and to inflate the prior variance V^{prior}, as described in the main paper.

4.2. Consistency Score

To prevent aggressive Kalman updates where triangulation and prior disagree, we compute a per-pixel consistency score $c(x) \in [0, 1]$. The relative discrepancy between the observed and prior scales is

$$\delta(x) = \frac{|S^{\text{obs}}(x) - S^{\text{prior}}(x)|}{S^{\text{obs}}(x)}, \tag{8}$$

and the frame-level tolerance is estimated as $\sigma_e = \text{MAD}(\delta)$, smoothed across time with an exponential moving average. The consistency score is then

$$c(x) = \exp\left(-\frac{\delta(x)^2}{2\,\sigma_e^2}\right). \tag{9}$$

Pixels with $c(x) \approx 1$ receive the full Kalman gain, while pixels with low $c(x)$ are down-weighted to suppress unreliable triangulation. This score is used to cap the raw Kalman gain κ_{raw} as described in the main paper.

5. Superpixel-Based Spatial Refinement

Although the Bayesian update described in the main paper provides a pixel-wise fusion of the triangulated depth and the warped prior, residual spatial inconsistencies remain due to imperfect optical flow, unstable triangulation, and amplified noise in distant regions. To mitigate these effects, we refine the posterior scale field S^{post} using superpixels that follow both the color and geometric structure of the scene.

5.1. Motivation

Neither triangulation nor prior warping is perfectly reliable. Triangulated depths become unstable when the stereo baseline is small, the motion is rotation-dominant, or when the flow correspondence lies near occlusion boundaries. Similarly, the optical flow used for warping may contain several-pixel errors in low-texture regions or around dynamic objects. These errors propagate into the posterior $S^{\text{post}}(x)$ and appear as irregular islands or thin artifacts after fusion.

Pixel-wise Bayesian fusion suppresses high-frequency noise but cannot fully remove spatially coherent distortions caused by flow mismatch or triangulation failures. A structural, surface-level correction stage is therefore necessary.

Figure 3. **Comparison of triangulated vs. refined point clouds.** (Left) Raw triangulated depth produces scattered points and surface discontinuities due to imperfect flow and unstable geometry. (Right) After our Bayesian update and superpixel refinement, the reconstructed surfaces become smoother and more consistent.

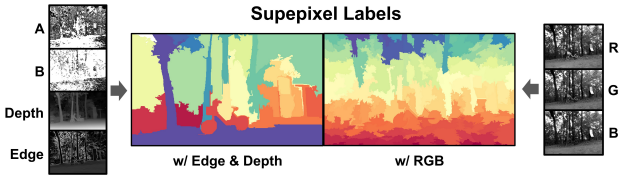


Figure 4. **LAB-based vs. RGB superpixel segmentation.** (Left) Our LAB+depth segmentation follows true object and geometric contours, making each label more consistent with the underlying 3D structure. (Right) RGB-based segmentation often misaligns with physical surface.

Figure 3 illustrates this issue: the raw triangulated point cloud (left) contains numerous spikes and scattered surface fragments, while our refined point cloud (right) is significantly smoother and geometrically more coherent.

5.2. LAB-based segmentation aligned to depth structure

We apply Felzenszwalb segmentation [**?**] using LAB color and relative depth features as input, rather than RGB alone, to better align superpixel boundaries with geometric structure. The LAB representation is chosen because the L channel is less sensitive to illumination changes, while the a and b channels provide stable chromatic cues. Combined with relative depth edges, the resulting superpixels adhere closely to object boundaries and planar surfaces.

Figure 4 compares segmentation results obtained from the original RGB image and from our LAB+depth representation. While RGB-based superpixels tend to bleed across surfaces, the LAB-based segmentation produces spatially coherent, semantically meaningful regions that align with depth discontinuities.

5.3. Label-wise scale refinement

Let $\{\Lambda_\ell\}$ denote the set of superpixel segments. For each segment Λ_ℓ, we refine the pixel-wise posterior scale by

$$\bar{s}_\ell = \text{median}\,\{\, S_k^{\text{post}} \mid k \in \Lambda_\ell \,\}.$$

This median-based representative scale is then applied to all pixels in the corresponding superpixel:

$$S_k^{\text{seg}} = \begin{cases} \bar{s}_\ell, & k \in \Lambda_\ell, \\ S_k^{\text{post}}, & \text{otherwise.} \end{cases}$$

Because all pixels in Λ_ℓ originate from the same physical surface, the above operation enforces geometric consistency across the region while automatically rejecting local artifacts caused by noisy flow or triangulation.

This refinement offers the following advantages:

- **Flow robustness:** A few-pixel flow mismatch no longer produces ripples or tearing across a surface because the label enforces a single representative scale.
- **Triangulation stability:** Noisy triangulated points within a label are suppressed when they are inconsistent with the dominant depth trend of the surface.
- **Structure preservation:** Since superpixels align with depth boundaries (Fig. 4), refinement does not blur across object edges.
- **Noise reduction in distant regions:** long-range triangulation noise is absorbed into the label median, producing smoother geometry as seen in Fig. 3.

The refined scale field S^{seg} is subsequently used to compute the final metric depth:

$$z_k^{\text{post}} = S_k^{\text{seg}} \cdot d_k^{\text{rel}}.$$

This stage acts as a structural regularizer that complements the Bayesian update and stabilizes the depth estimates across large spatial regions.

6. Ablation on Odometry Accuracy

To investigate the sensitivity of our method to odometry quality, we replace the estimated pose with ground-truth (GT) components and measure the effect on depth accuracy and temporal consistency. Three settings are compared: (1) fully estimated rotation and translation from optical flow, (2) GT rotation with estimated translation, and (3) full GT pose. Results are summarized in Table 1.

On TartanAir, where image–pose synchronization is perfect, using GT pose consistently improves both accuracy and temporal consistency. However, on MS2, using full GT pose actually degrades performance. This is because the thermal images and RTK-GPS measurements are not perfectly synchronized, and the resulting temporal misalignment introduces errors in triangulation that outweigh the benefit of more accurate pose. This result highlights that our flow-based pose estimation is more robust to synchronization issues than directly using external pose measurements.

	TartanAir (Synthetic RGB)			MS2 (Thermal)		
Pose setting	AbsRel↓	$\delta < 1.25$ ↑	TAE↓	AbsRel↓	$\delta < 1.25$ ↑	TAE↓
Estimated $\boldsymbol{\Omega}$, $\boldsymbol{T}$	0.218	0.714	4.65	0.513	0.603	5.75
GT $\boldsymbol{\Omega}$ only	0.183	0.755	4.45	0.254	0.624	5.91
GT $\boldsymbol{\Omega}$, $\boldsymbol{T}$	0.172	0.799	4.19	0.974	0.370	–

Table 1. **Ablation on odometry accuracy.** TartanAir: neighborhood (4.2K frames), MS2: 2021-08-13-21-18-04 (10K frames). On TartanAir, GT pose consistently improves accuracy thanks to perfect synchronization. On MS2, full GT pose degrades performance due to temporal misalignment between thermal images and RTK-GPS.

7. Runtime Analysis

We profile our method on a desktop PC equipped with an AMD Ryzen 9 9900X CPU. Table 2 summarizes the per-frame computational breakdown at KITTI resolution (376×1241).

	Seg+Flow	Motion	Scale	Tri+Fusion	**C++ Total**	**+Depth**
Time (ms)	43	25	15	6	**90**	**156**

Table 2. **Runtime breakdown at KITTI resolution (376×1241).** The full pipeline, including monocular depth estimation, achieves **6.4 FPS**.

The C++ core (segmentation, optical flow, motion estimation, scale recovery, triangulation, and Bayesian fusion) runs in approximately 90 ms per frame. Including the monocular relative depth model (DepthAnything v2), the total processing time is 156 ms per frame (6.4 FPS), which is suitable for real-time robotics applications.

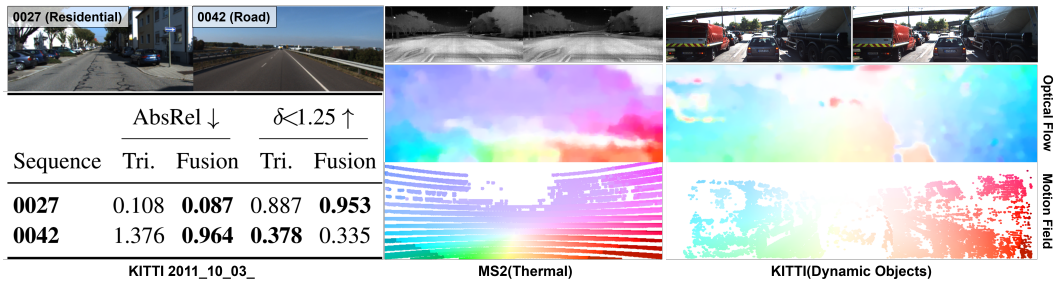
In terms of memory, the pipeline maintains approximately 100 MB of active and persistent buffers per megapixel of input resolution. The dominant allocations are per-pixel float32 maps for depth, scale, variance, and optical flow, along with persistent state buffers that carry the posterior estimate across frames.

8. Limitations

Although our system significantly improves the metric consistency of monocular depth, several limitations remain.

First, the method inherits the fundamental weakness of triangulation: when the camera motion is dominated by rotation or the parallax is too small, triangulated depths become unreliable. In such cases we fall back to flow–based warping, which stabilizes the scale but can still be affected by flow errors.

Second, our refinement relies on the structural continuity of the relative depth map. When the relative–depth predictor becomes unstable (e.g., in low-light scenes, overexposed regions, or very distant surfaces), the propagated scale may cause mild flickering.



	AbsRel ↓		δ<1.25 ↑	
Sequence	Tri.	Fusion	Tri.	Fusion
0027	0.108	**0.087**	0.887	**0.953**
0042	1.376	**0.964**	**0.378**	0.335

Figure 5. **Failure cases.** (Left) Forward-dominant motion in KITTI Seq 42 causes degenerate triangulation, resulting in unreliable depth observations. (Right) Scenes with dominant dynamic objects produce inconsistent optical flow, which degrades pose estimation and subsequent depth fusion.

Third, in thermal (LWIR) imaging, optical flow estimation can degrade significantly due to low texture, thermal crossover, or sensor bloom, even when FieldScale preprocessing is applied. In such cases, both flow-based scale estimation and pose recovery become unreliable, which can temporarily destabilize the scale tracking module.

Fourth, the approach assumes a predominantly rigid scene. If a large portion of the frame is covered by independently moving objects, the flow and triangulation constraints become inconsistent, and the fusion module may temporarily lose stability.

Finally, extremely distant regions (above 50–80 m) remain challenging because both triangulation and flow provide weak geometric cues. This limitation is shared across all monocular metric–depth approaches.