DAGE: Dual-Stream Architecture for Efficient and Fine-Grained Geometry Estimation

Supplementary Material

Table 1. **Training datasets**.

Dataset Name	Scene type	Metric	Real	Dynamic
ARKitScenes [2]	Indoor	Yes	Real	Static
ScanNet [11]	Indoor	Yes	Real	Static
ScanNet++ [51]	Indoor	Yes	Real	Static
TartanAir [46]	Mixed	Yes	Synthetic	Dynamic
Waymo [37]	Outdoor	Yes	Real	Dynamic
BlendedMVS [50]	Mixed	No	Synthetic	Static
HyperSim [31]	Indoor	Yes	Synthetic	Static
MVS Synth [19]	Outdoor	No	Synthetic	Static
GTA-Sfm [42]	Outdoor	No	Synthetic	Static
MegaDepth [25]	Outdoor	No	Real	Static
CO3Dv2 [30]†	Object-centric	No	Real	Static
WildRGBD [48]†	Object-centric	Yes	Real	Static
VirtualKITTI2 [6]	Outdoor	Yes	Synthetic	Dynamic
Matterport3D [7]	Indoor	Yes	Real	Static
BEDLAM [3]†	Mixed	Yes	Synthetic	Dynamic
Dynamic Replica [22]	Indoor	Yes	Synthetic	Dynamic
PointOdyssey [53]†	Mixed	Yes	Synthetic	Dynamic
Spring [28]	Mixed	Yes	Synthetic	Dynamic

† *Only a subset of each dataset is used.*

1. More Training Details

1.1. Training datasets

We train on 18 datasets spanning indoor, outdoor, and object-centric scenes, covering both static and dynamic settings. The full list appears in Tab. 1. Following [43], we filter scenes with ambiguous annotations in PointOdyssey [53], and remove scenes with panorama backgrounds and zoom-in/out effect in BEDLAM [3]. For object-centric datasets [30, 48], we only subsample 40 scenes for each object category.

1.2. Implementation Details

Architecture. For the high-resolution (HR) stream, we initialize the model with the 24-layer ViT from MoGe2 [45] and keep these weights frozen throughout training. For the low-resolution (LR) stream, our training corpus (Sec. 1.1) is considerably smaller than those used by recent feed-forward visual-geometry models [23, 41, 47]. Consequently, rather than training a global video transformer from scratch, we start from the Pi3 [47] checkpoint, which comprises 36 attention layers with alternating frame-wise and global attention. The adapter contains five blocks; each block consists of one cross-attention layer, one self-attention layer, and an MLP. We train the adapter from scratch and *zero-initialize* its final projection to avoid destabilizing the frozen HR features at the start of training. For dense geometry, the pointmap head is implemented as a stack of residual convolutional blocks with transposed convolutions that progressively upsample from patch resolution $(h_{hr} \times w_{hr} = H/14 \times W/14)$ to the original image resolution $(H \times W)$. Camera poses and the scene-wise metric scale factor are predicted with a two-layer MLP. For distillation during training, we use a 2-layer MLP and *pixel shuffle* to project $\mathcal{F}^{\text{lr}}$ to the teacher spatial resolution.

Training loss. We set the weight for each loss as follow: $\lambda_{\text{pm}} = 1.0, \lambda_{\text{cam}} = 0.1, \lambda_{\text{trans}} = 100.0, \lambda_{\text{rot}} = 1.0, \lambda_{\text{scale}} = 1.0, \lambda_{\text{normal}} = 1.0, \lambda_{\text{gradient}} = 0.1, \lambda_{\text{distill}} = 0.5$.

Optimization. We train our model in two stages. *Stage 1* targets low–medium resolutions with longer clips; *Stage 2* fine-tunes on high-resolution inputs with short clips. Both stages use AdamW [26] optimizer with a OneCycleLR schedule. In Stage 1 (30,000 steps), we set a base LR of 1×10^{-4} for the adapter and dense heads, and 1×10^{-5} ($10\times$ lower) for the global transformer initialized from Pi3. In Stage 2 (10,000 steps), we freeze the global transformer and fine-tune only the adapter and heads at 1×10^{-5}. To keep training efficient, we use FlashAttention [12, 13], `bfloat16` mixed precision, gradient checkpointing, and gradient accumulation. With this setup, training takes roughly five days on 16×A100-80GB GPUs.

Augmentation and sampling. We extend the MoGe [44] augmentation pipeline to the multi-view setting and adopt stage-specific regimes. *Stage 1 (long sequence):* we sample 2–24 frames per clip and constrain the total pixels to $[1.0 \times 10^5, 2.55 \times 10^5]$, thereby enabling a large per-GPU batch of 48 images; we apply distillation only in this stage. *Stage 2 (high resolution):* we sample just 2–4 frames per clip, set the total pixels to $[2.7 \times 10^5, 9.0 \times 10^5]$ (roughly 518×518–952×952 for 14-px patches), vary the aspect ratio within $[0.5, 2.0]$, and use 24 images per GPU.

2. More Evaluation Details

This section details the datasets and metrics used in our experiments.

2.1. Video geometry estimation.

Datasets. Following GeometryCrafter [49], we configure each test dataset as follows:

- **GMU Kitchens** [16]: We use all scenarios, extract 110 frames per sequence with a stride of 2, and downsample the 1920p videos and depth maps to 960×512.
- **Monkaa** [27]: We select 9 scenes and truncate each sequence to 110 frames at the native resolution of 960×512.
- **Sintel** [5]: We use all training sequences (21–50 frames) and crop from 1024×436 to 896×448.
- **ScanNet** [11]: We evaluate 100 test scenes with 90 frames per video (stride 3), and center-crop each frame to 640×512.
- **KITTI** [15]: We use all sequences from the depth-annotated validation split; for longer videos we keep the first 110 frames (yielding 13 videos with 67–110 frames), and center-crop to 768×384.
- **Diode** [39]: We use all 771 validation images at the default resolution of 1024×768.

In addition, we prepare two high-resolution evaluation sets:

- **UrbanSyn** [17]: We sample ten clips of 100 frames each from the original 7000-frame sequences and keep the resolution at 2048×1024.
- **Unreal4K** [38]: We use all nine scenes, keep the first 100 frames per scene, and downsample to 1920×1080.

Metrics. For the pointmap estimation, we report the mean relative point error $\mathrm{Rel}^p \downarrow = \|\hat{\mathbf{p}} - \mathbf{p}\|_2 / \|\mathbf{p}\|_2$ and the inlier ratio $\delta^p \uparrow$, where a point is an inlier if $\|\hat{\mathbf{p}} - \mathbf{p}\|_2 / \min(\|\mathbf{p}\|_2, \|\hat{\mathbf{p}}\|_2) < \tau$ (with τ=0.25), averaged over valid pixels. Similarly, we leverage $\mathrm{Rel}^d \downarrow$ and $\delta^d \uparrow$ for depth estimation.

2.2. Video sharpness depth.

Datasets. We evaluate depth–boundary sharpness on four synthetic datasets—Monkaa [27], Sintel [5], UrbanSyn [17], and Unreal4K [38].

Metrics. We use the F1↑ edge metric from DepthPro [4]. For each pair of neighboring pixels, we mark an occluding contour when the depth ratio exceeds a *predefined threshold*. Applying this to both prediction and ground truth yields two contour maps. *Precision* is the fraction of predicted contour pairs that are also contours in the ground truth, and *recall* is the fraction of ground-truth contour pairs recovered by the prediction. The F1 score is the harmonic mean of precision and recall. We report the F1 averaged over multiple thresholds. This metric requires no ground-truth edge maps and is easily computed wherever dense depth annotations are available (e.g., synthetic data).

To further assess boundary sharpness, we adopt the Depth Boundary Error (DBE) from iBims [24] and use its pseudo variant (PDBE) for datasets without depth–edge annotations (following [29]). Concretely, we run Canny edge detection on both predicted and ground-truth depth maps to obtain edge sets, then compute the iBims accuracy and completeness terms. The accuracy term penalizes predicted edges that are far from any ground-truth edge, while the completeness term penalizes ground-truth edges not recovered by the prediction. Finally, we report the *chamfer distance* $\mathcal{C}_{\mathrm{PDBE}} \downarrow$, which is the average of accuracy and completeness.

2.3. Multi-view reconstruction.

Datasets. We evaluate 3D pointmap reconstruction on 7-Scenes [33] and NRGBD [1] under both sparse and dense view protocols. For sparse views, we sample keyframes every 200 frames on 7-Scenes and every 500 on NRGBD; for dense views, the strides are 40 and 100, respectively.

Metrics. We employ the *Accuracy* (Acc↓): mean nearest-neighbor distance from each predicted point to the ground truth, *Completion* (Comp↓): mean nearest-neighbor distance from each ground-truth point to the reconstruction, and *Normal Consistency* (NC↑): mean absolute dot product of ground truth and predicted normals (computed on the fly using `Open3D` library).

2.4. Camera pose estimation.

Datasets. We evaluate on Sintel [5], TUM-Dynamics [35], and ScanNet [11]. For Sintel, we follow [9, 52], excluding static scenes and those with perfectly straight camera motion, leaving 14 sequences. For TUM-Dynamics and ScanNet, we use the first 90 frames with a temporal stride of 3.

Metrics. Following [43, 47, 52], we report Absolute Trajectory Error (ATE↓) and Relative Pose Error for translation and rotation ($\mathrm{RPE}_T \downarrow$ / $\mathrm{RPE}_R \downarrow$). Predicted trajectories are first aligned to ground truth with a single $\mathrm{Sim}(3)$ transform (global scale, rotation, translation). ATE is the root-mean-square discrepancy between aligned and ground-truth camera positions over the entire sequence. RPE_T is the translation error over a certain distance, and RPE_R is the rotation error over a certain degree; both are averaged over all pose pairs.

3. More Results

3.1. Video geometry estimation

We evaluate video geometry estimation under four other settings. First, for *scale-invariant* video pointmaps, we align predictions to ground truth with a *single* per-video scale and report results in Tab. 2. Second, for video *depth*, we follow standard practice and report both *affine-invariant* results—per-frame scale + shift alignment—in Tab. 3, and *scale-invariant* results—single per-video scale—in Tab. 4. Finally, we assess *metric-scale* video pointmaps with **no** alignment (direct comparison in the dataset's metric units); see Tab. 5. For the metric setting, we compare against methods capable of predicting metric geometry, including CUT3R [43] and MapAnything [23].

Table 2. **Scale-invariant video pointmap evaluation**. Results are aligned with the ground truth by optimizing a shared scale factor across the entire video. We mark best and second-best.

Method	**GMU** [16]		**Monkaa** [27]		**Sintel** [5]		**ScanNet** [11]		**KITTI** [15]		**UrbanSyn** [17]		**Unreal4K** [38]		**Diode** [39]		**Rank** ↓
	Rel^p ↓	δ^p ↑	Rel^p ↓	δ^p ↑	Rel^p ↓	δ^p ↑	Rel^p ↓	δ^p ↑	Rel^p ↓	δ^p ↑	Rel^p ↓	δ^p ↑	Rel^p ↓	δ^p ↑	Rel^p ↓	δ^p ↑	
DepthPro [4]	10.5	92.7	27.9	51.2	55.0	37.5	9.3	95.0	11.7	93.6	22.5	61.1	96.1	1.2	30.3	58.1	7.6
MoGe [44]	21.4	69.0	27.7	58.3	29.5	59.8	13.4	88.2	8.6	95.6	13.4	89.9	34.0	55.7	30.3	53.4	6.9
MoGe2 [45]	19.7	72.1	30.8	51.1	34.3	47.7	12.7	89.4	11.7	96.9	12.3	91.7	30.1	62.3	29.5	55.4	7.0
MoGe2 [45]†	7.1	94.6	25.8	60.2	33.1	52.1	7.8	97.5	10.5	98.4	6.5	97.2	8.9	92.1	15.8	84.1	3.9
CUT3R [43]	8.2	93.6	34.9	45.9	42.9	35.8	6.5	98.0	16.0	88.1	57.9	14.0	17.5	78.3	17.2	81.6	6.9
VGGT [41]	5.6	93.8	16.0	80.4	26.7	65.8	3.1	99.0	8.4	97.3	18.5	75.0	8.7	96.5	13.6	80.2	3.6
Pi3 [47]	5.4	94.2	12.6	90.2	29.6	62.5	2.4	99.4	9.2	90.8	10.7	93.8	17.2	75.4	9.0	96.1	3.1
GeoCrafter [49]	8.4	94.5	20.7	73.9	30.2	57.8	8.9	96.4	6.4	98.8	11.3	95.3	21.0	73.5	13.0	92.8	4.1
DAGE (ours)	5.0	94.2	11.3	88.1	26.6	66.2	2.4	99.5	7.3	99.0	7.9	96.6	9.2	92.9	10.0	94.4	**1.7**

Table 3. **Affine-invariant video depthmap evaluation**. Results are aligned with the ground truth by optimizing a shared scale and shift factor across the entire video. We mark best and second-best.

Method	**GMU** [16]		**Monkaa** [27]		**Sintel** [5]		**ScanNet** [11]		**KITTI** [15]		**UrbanSyn** [17]		**Unreal4K** [38]		**Diode** [39]		**Rank** ↓
	Rel^d ↓	δ^d ↑	Rel^d ↓	δ^d ↑	Rel^d ↓	δ^d ↑	Rel^d ↓	δ^d ↑	Rel^d ↓	δ^d ↑	Rel^d ↓	δ^d ↑	Rel^d ↓	δ^d ↑	Rel^d ↓	δ^d ↑	
DepthPro [4]	8.8	93.0	23.3	55.8	36.1	49.5	8.1	94.6	11.7	93.6	51.3	38.3	105.0	20.0	31.0	58.8	8.0
MoGe [44]	19.9	66.5	19.9	63.6	26.5	60.0	12.5	85.9	7.6	94.2	15.2	82.1	38.7	46.2	31.3	48.2	7.6
MoGe2 [45]	19.0	68.9	20.8	60.8	26.4	59.9	12.1	86.9	6.7	96.7	13.8	85.4	33.6	52.8	29.5	50.4	6.8
MoGe2 [45]†	6.6	93.8	18.0	68.1	25.0	63.8	6.4	96.8	5.2	98.3	7.7	96.4	12.0	88.3	14.7	80.7	3.8
CUT3R [43]	7.3	92.9	28.0	49.9	31.9	50.5	5.4	97.4	10.2	89.1	48.6	36.7	15.4	79.7	16.0	78.4	6.8
VGGT [41]	5.2	93.2	12.3	80.5	22.2	70.4	2.7	98.7	4.7	97.2	13.9	84.5	8.2	94.3	12.4	85.2	3.5
Pi3 [47]	4.9	93.5	8.2	91.4	20.2	71.7	2.0	99.3	3.0	99.1	16.0	78.8	18.3	78.6	8.6	92.9	2.7
GeoCrafter [49]	7.7	94.1	13.4	79.3	21.4	70.6	7.3	96.1	5.0	98.5	12.2	90.3	20.7	72.2	9.1	93.4	3.9
DAGE (ours)	4.8	93.5	9.5	87.2	19.5	74.4	2.1	99.4	3.2	98.8	7.7	95.8	12.1	88.1	8.7	92.5	**1.9**

We additionally evaluate feed-forward visual-geometry approaches at each dataset's native resolution (540p–2K). As reported in Tab. 6, performance degrades steadily with increasing resolution; at the highest, far beyond training scales (e.g. Urbansyn and Unreak4k datasets), most methods collapse except ours.

3.2. Single-image geometry estimation

Following [44, 45], we evaluate the single-image geometry estimation on eight different datasets, including NYUv2 [34], KITTI [15], ETH3D [32], iBims-1 [24], GSO [14], Sintel [5], DDAD [18], DIODE [39], HAMMER [21]. The results are summaried in Tab. 7, validating that our dual-stream design preserves single-image quality compared to single-image based methods like DepthPro [4], MoGE [44, 45].

3.3. Camera pose estimation

We additionally report the predicted camera poses on RealEstate10K and CO3Dv2 datasets. We report the Relative Rotation Accuracy (RRA) and Relative Translation Accuracy (RTA) at a given threshold, and the Area Under the Curve (AUC) of the min(RRA,RTA) threshold curve. Tab. 8 shows that DAGE remains competitive with Pi3 [47] and VGGT [41], even while operating at a lower resolution.

3.4. More ablation studies

Low-resolution stream architecture. We perform an ablation study of the global module in our LR stream. Specifically, in addition to the global transformer with alternative frame/global attention, we ablate with two other design: (1) transformer-based recurrent network [43] and (2) temporal Mamba network [10]. Results in Tab. 9a show that the alternating global-attention transformer consistently outperforms both variants, reflecting stronger multi-view aggre-

Table 4. **Scale-invariant video depthmap evaluation**. Results are aligned with the ground truth by optimizing a shared scale factor across the entire video. We mark best and second-best.

Method	**GMU** [16]		**Monkaa** [27]		**Sintel** [5]		**ScanNet** [11]		**KITTI** [15]		**UrbanSyn** [17]		**Unreal4K** [38]		**Diode** [39]		**Rank** ↓
	Rel^d ↓	δ^d ↑	Rel^d ↓	δ^d ↑	Rel^d ↓	δ^d ↑	Rel^d ↓	δ^d ↑	Rel^d ↓	δ^d ↑	Rel^d ↓	δ^d ↑	Rel^d ↓	δ^d ↑	Rel^d ↓	δ^d ↑	
DepthPro [4]	9.4	92.1	26.7	45.9	53.6	35.3	8.8	92.9	8.2	92.5	22.2	40.2	96.0	1.1	29.3	56.4	7.9
MoGe [44]	20.7	64.7	25.5	54.8	31.4	48.9	13.3	85.0	7.7	94.1	13.1	86.3	34.7	49.8	29.8	48.2	7.6
MoGe2 [45]	19.5	67.1	27.1	51.6	31.2	47.7	12.0	86.5	7.2	96.7	12.0	88.8	30.7	56.6	28.5	50.7	6.9
MoGe2 [45]†	6.7	93.8	21.9	60.4	30.1	51.9	7.1	95.9	5.6	98.3	6.0	96.8	8.7	91.0	14.7	80.7	3.7
CUT3R [43]	7.9	92.6	33.0	38.3	37.3	42.4	5.8	97.0	11.3	86.8	22.2	63.1	16.8	79.6	15.6	80.0	6.7
VGGT [41]	5.2	93.0	14.4	77.3	25.3	62.1	2.8	98.6	5.3	96.7	18.3	73.3	8.2	96.1	13.4	79.2	3.6
Pi3 [47]	4.9	93.4	10.8	88.9	28.4	60.6	2.1	99.3	3.1	99.1	9.5	92.5	16.6	75.0	8.7	95.5	2.3
GeoCrafter [49]	8.1	93.8	18.1	71.1	27.1	58.7	7.9	95.5	5.1	98.4	11.0	92.4	21.1	70.9	10.0	92.4	4.1
DAGE (ours)	4.7	93.4	11.5	85.5	25.6	64.8	2.2	99.4	3.3	98.8	7.9	95.9	8.7	90.3	9.9	94.0	**1.9**

Table 5. **Metric video pointmap evaluation**. Predicted pointmaps are directly compared with ground truth.

Method	**GMU** [16]		**ScanNet** [11]		**KITTI** [15]		**UrbanSyn** [17]		**Diode** [39]	
	Rel^p ↓	δ^p ↑	Rel^p ↓	δ^p ↑	Rel^p ↓	δ^p ↑	Rel^p ↓	δ^p ↑	Rel^p ↓	δ^p ↑
CUT3R [43]	13.5	90.7	9.1	95.2	34.2	14.6	15.4	84.4	31.6	47.2
MapAny [23]	22.6	63.4	35.2	28.5	29.1	26.3	28.5	37.3	33.8	31.8
DAGE (ours)	7.5	95.3	2.5	99.5	12.0	98.3	8.3	96.5	12.9	87.5

gation and more reliable cross-view consistency.

RoPE design in the adapter. We ablate rotary positional encodings (RoPE) in the adapter in Tabs. 9b and 9c. For self-attention (Tab. 9b), standard RoPE [36] is ineffective at high resolutions (e.g., UrbanSyn dataset), whereas interpolated RoPE improves performance. For cross-attention (Tab. 9c), adding RoPE alongside our alignment ("snapping") further boosts results.

3.5. More qualitative results

Interactive viewer (highly recommended). Our webpage (github.com/dage-site) contains the interactive 3D viewer of reconstructed pointmaps and predicted videodepths from real-world scenarios.

Fig. 1 shows qualitative 3D pointmap reconstructions on in-the-wild scenes spanning static/dynamic motion, indoor/outdoor settings, and object-centric versus scene-level compositions.

Figs. 2, 3, 4 compare our video depth to recent state-of-the-art methods [41, 47, 49], highlighting sharper boundaries and stronger temporal stability.

Fig. 5 visualizes depth-edge maps—the contours obtained by thresholding neighboring-pixel depth changes. Compared to baselines [41, 47, 49], our results capture thin structures and small or distant objects more reliably.

Fig. 6 compares 3D pointmaps from DAGE to an *aligned-MoGe2* baseline. In Tab. 6 (Sec. 4.6), we define **Setting A**: run MoGe2 [45] per frame and *post hoc* align each predicted pointmap to a globally consistent pointmap from Pi3 [47]. This simple alignment recovers fine detail and enforces a shared scale, but—as the figure shows—still produces layering/stitching artifacts because depth is estimated independently per frame without strong cross-view coupling.

Fig. 7 visualizes 3D pointmaps reconstructed from 2K inputs. DAGE runs substantially faster—especially on longer clips—while producing more plausible, multi-view–consistent reconstructions. In contrast, global-attention baselines [41, 47] either run out of memory or degrade at this resolution.

4. High-resolution inference analysis of visual-geometry models

We analyze how pretrained feed-forward visual-geometry models [41, 47] behave when evaluated well beyond their training resolution (up to 2K on the long side).

Single-image stress test. We resize single-image inputs to several resolutions (e.g., 540p, 1080p, and 2K) and run the public checkpoints of VGGT [41] and Pi3 [47] without any architectural changes. We visualize depth maps and corresponding 3D pointmaps (VGGT in Fig. 8a, Pi3 in Fig. 8b). At $\sim$ 540p, both methods produce plausible geometry.

Table 6. **Affine-invariant video pointmap evaluation at native resolution.** Predictions are aligned to ground truth by optimizing a single scale and shift across the entire video.

Method	**GMU** [16] (960 × 512)		**Monkaa** [27] (960 × 512)		**Sintel** [5] (896 × 448)		**ScanNet** [11] (640 × 512)		**KITTI** [15] (768 × 384)		**UrbanSyn** [17] (2048 × 1024)		**Unreal4K** [38] (1920 × 1080)		**Diode** [39] (1024 × 768)	
	$\text{Rel}^d \downarrow$	$\delta^d \uparrow$	$\text{Rel}^d \downarrow$	$\delta^d \uparrow$	$\text{Rel}^d \downarrow$	$\delta^d \uparrow$	$\text{Rel}^d \downarrow$	$\delta^d \uparrow$	$\text{Rel}^d \downarrow$	$\delta^d \uparrow$	$\text{Rel}^d \downarrow$	$\delta^d \uparrow$	$\text{Rel}^d \downarrow$	$\delta^d \uparrow$	$\text{Rel}^d \downarrow$	$\delta^d \uparrow$
CUT3R [43]	22.0	67.9	30.9	51.0	38.9	40.9	7.0	97.9	13.2	88.7	56.7	13.9	71.6	5.6	31.1	52.6
VGGT [41]	15.9	91.4	17.7	81.6	28.7	63.8	4.5	99.1	7.8	97.5	OOM	OOM	OOM	OOM	20.5	76.3
Pi3 [47]	6.2	92.2	12.6	88.9	21.7	72.9	2.2	99.5	5.9	97.5	55.9	14.7	54.2	17.1	13.9	87.2
DAGE (ours)	4.9	94.2	10.1	91.0	21.5	75.6	2.1	99.5	5.9	99.0	8.8	96.0	11.9	89.1	9.7	94.4

Table 7. **Single-image geometry evaluation**. Results are aligned with the ground truth by optimizing a scale and shift factor for each image. We mark best and second-best.

Method	**NYUv2** [34]		**KITTI** [15]		**ETH3D** [32]		**iBims-1** [24]		**GSO** [14]		**Sintel** [5]		**DDAD** [18]		**DIODE** [39]		**HAMMER** [21]		**Rank** ↓
	$\text{Rel}^d \downarrow$	$\delta^d \uparrow$	$\text{Rel}^d \downarrow$	$\delta^d \uparrow$	$\text{Rel}^d \downarrow$	$\delta^d \uparrow$	$\text{Rel}^d \downarrow$	$\delta^d \uparrow$	$\text{Rel}^d \downarrow$	$\delta^d \uparrow$	$\text{Rel}^d \downarrow$	$\delta^d \uparrow$	$\text{Rel}^d \downarrow$	$\delta^d \uparrow$	$\text{Rel}^d \downarrow$	$\delta^d \uparrow$	$\text{Rel}^d \downarrow$	$\delta^d \uparrow$	
DepthPro [4]	4.36	97.9	9.15	90.7	7.73	94.0	4.34	97.4	3.16	99.7	19.6	74.5	14.4	81.2	6.28	93.7	5.31	98.8	3.8
MoGe [44]	3.68	98.3	4.86	97.2	3.57	99.0	3.61	97.3	1.14	100	16.8	77.8	10.5	91.4	4.37	96.4	3.88	98.1	1.8
MoGe2 [45]	3.33	98.4	6.47	96.4	3.89	98.7	3.65	98.5	1.16	100	17.4	77.0	10.1	90.3	5.13	94.9	4.19	99.1	1.9
DAGE (ours)	3.34	98.4	7.52	94.7	3.49	98.0	3.70	97.8	1.26	99.9	18.9	74.8	10.7	89.2	4.97	94.6	3.43	98.6	2.5

When the resolution is increased to $\sim$ 1080p, predictions exhibit shape distortions; at $\sim$ 2K, outputs often collapse into fragmented or globally inconsistent pointmaps. These failures are consistent across scenes.

Global-attention behavior (3-view input). To probe failure modes under high resolution, we evaluate VGGT with *triplets* of views (3 frames), not single images. We fix the number of views to three and vary only the spatial resolution. Following prior observations that global layers perform exhaustive correspondence search [40], we visualize post-softmax maps for a few query tokens in view 1 and overlay their responses in the other views (Figs. 9–11). At $\sim$ 540p, the maps are compact and centered on true correspondences. As resolution increases, attention becomes diffuse and multi-modal, drifting toward semantically similar yet geometrically incorrect regions; by 2K it degenerates into high-entropy responses with no clear matches.

Likely causes. (i) *Positional extrapolation:* standard rotary/absolute positional parameterizations learned at $\sim$ 540 px do not extrapolate reliably to much larger token grids, skewing query–key phases and degrading similarity scores [8]. (ii) *Entropy growth:* increasing resolution raises token count without increasing the effective receptive field, making correspondence sparser per token and increasing attention entropy [20]. (iii) *Distribution shift:* training rarely exposes models to high-frequency, high-resolution statistics; the learned global matcher thus overfits to lower-res aliasing patterns.

From our experiments, we find that naively scaling input resolution is unreliable for current global-attention pipelines: at 1K–2K, pretrained models often exhibit correspondence collapse—diffuse attention and distorted depth/pointmaps—likely due to positional-encoding extrapolation and distribution shifts. Therefore, in our proposed DAGE, we amortize global aggregation at low resolution and fuse it into a per-frame high-resolution path; this preserves detail at 2K while keeping memory and runtime practical. Furthermore, to stabilize high-res inference, we adopt resolution-aware positional encodings (interpolated RoPE), explicit cross-scale alignment (snapping HR token coordinates to the LR grid for cross-attention), and multi-scale training that includes high-res regimes.

Table 8. **Pose Estimation** on RealEstate10K and Co3Dv2

Method	RealEstate10K			Co3Dv2		
	RRA@30 ↑	RTA@30 ↑	AUC@30 ↑	RRA@30 ↑	RTA@30 ↑	AUC@30 ↑
VGGT (518px)	99.97	93.13	77.62	98.64	97.62	91.28
Pi3 (518px)	99.99	95.62	85.90	98.49	97.53	91.39
DAGE (252px)	99.98	95.22	83.12	98.74	97.71	90.71

Table 9. **Ablations.** (a) LR-stream architectures. (b,c) Positional encodings.

(a) **Ablation on different architectures of the LR stream**.

Method	**GMU** [16]		**Monkaa** [27]		**Sintel** [5]		**ScanNet** [11]		**KITTI** [15]		**UrbanSyn** [17]		**Unreal4K** [38]		**Diode** [39]	
	$\text{Rel}^p \downarrow$	$\delta^p \uparrow$	$\text{Rel}^p \downarrow$	$\delta^p \uparrow$	$\text{Rel}^p \downarrow$	$\delta^p \uparrow$	$\text{Rel}^p \downarrow$	$\delta^p \uparrow$	$\text{Rel}^p \downarrow$	$\delta^p \uparrow$	$\text{Rel}^p \downarrow$	$\delta^p \uparrow$	$\text{Rel}^p \downarrow$	$\delta^p \uparrow$	$\text{Rel}^p \downarrow$	$\delta^p \uparrow$
MoGe2	19.6	72.4	25.0	57.0	29.8	58.4	12.4	89.4	9.0	97.2	13.4	90.0	32.9	59.1	31.0	54.2
Mamba	8.4	93.5	17.8	77.1	27.7	63.5	7.0	97.1	5.9	98.1	10.1	91.0	24.0	60.0	25.1	66.5
Trans. RNN	6.7	94.4	22.2	68.8	27.9	64.5	4.9	98.8	7.6	98.2	9.3	93.9	15.7	80.0	17.3	81.6
Global Trans.	4.9	94.2	10.1	91.0	21.5	75.6	2.1	99.5	5.9	99.0	8.8	96.0	11.9	89.1	9.7	94.4

(b) **Effect of RoPE in the self-attention**.

Positional Embedding	**Monkaa** [27]		**UrbanSyn** [17]	
	$\text{Rel}^p \downarrow$	$\delta^p \uparrow$	$\text{Rel}^p \downarrow$	$\delta^p \uparrow$
None	11.0	89.9	10.1	93.9
RoPE	**9.7**	**92.1**	10.3	93.5
Interp. RoPE (ours)	10.1	91.0	**8.8**	**96.0**

(c) **Effect of RoPE in the cross-attention**.

Positional Embedding	**Monkaa** [27]		**UrbanSyn** [17]	
	$\text{Rel}^p \downarrow$	$\delta^p \uparrow$	$\text{Rel}^p \downarrow$	$\delta^p \uparrow$
None	10.7	**91.1**	9.6	95.1
"Snap" RoPE (ours)	**10.1**	91.0	**8.8**	**96.0**

Figure 1. **Visualization of 3D pointmap reconstruction** on *in-the-wild* scenarios.

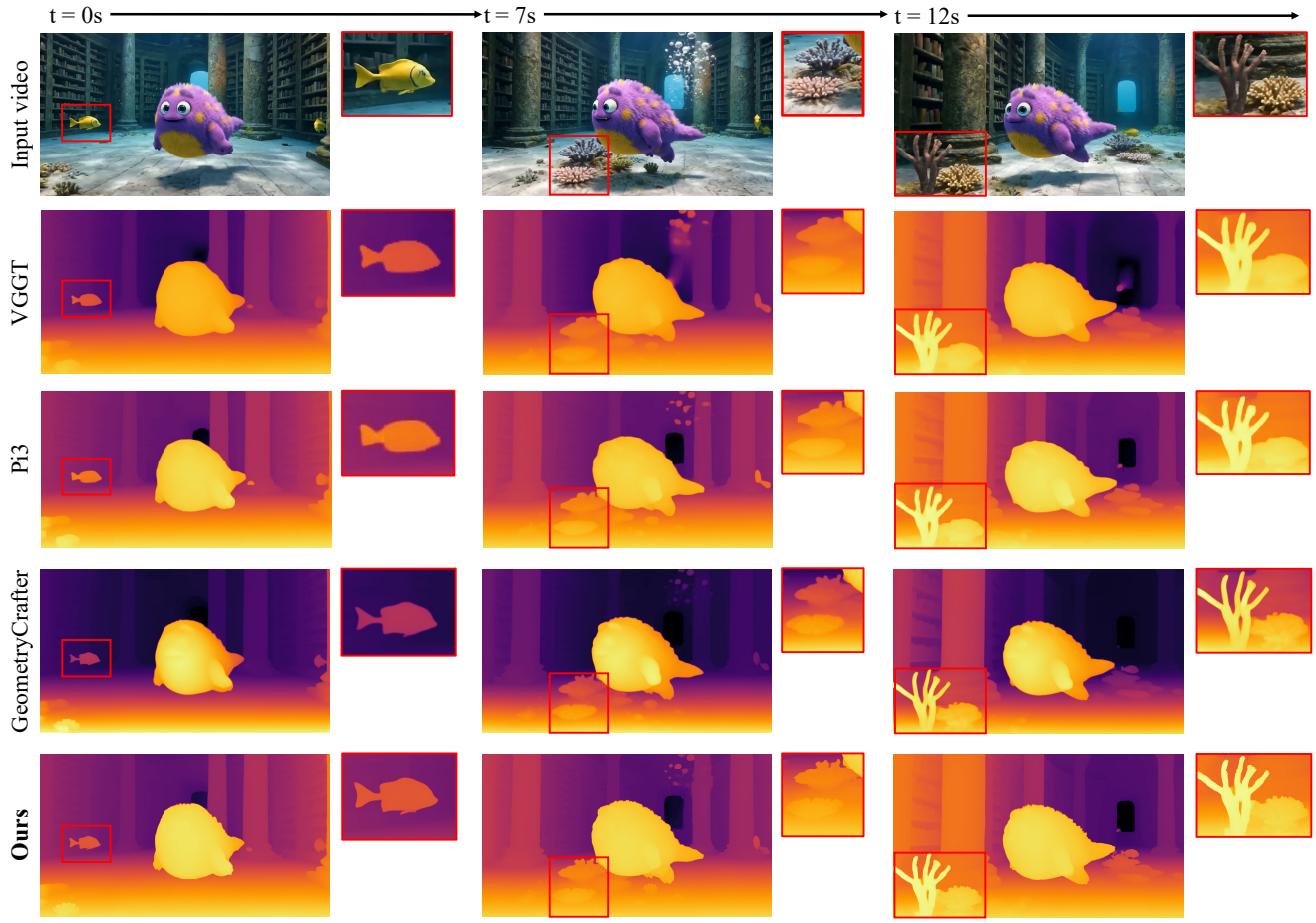


Figure 2. **Visualization of video depth estimation**. We compare our video depth prediction with VGGT [41], Pi3 [47], and GeoemtryCrafter [49]. DAGE demonstrates more sharp and fine-grained predictions.

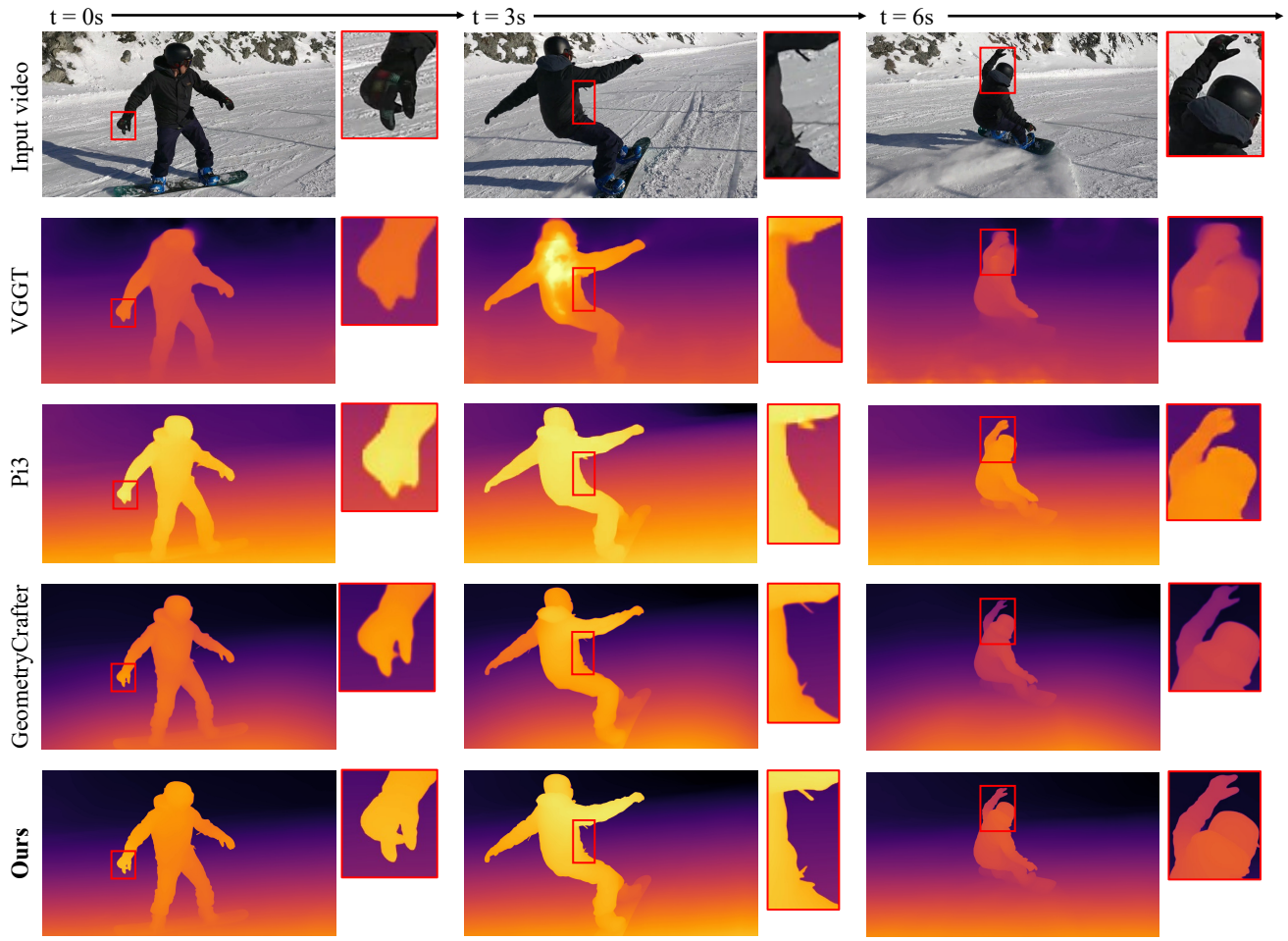


Figure 3. **Visualization of video depth estimation**. We compare our video depth prediction with VGGT [41], Pi3 [47], and GeoemtryCrafter [49]. DAGE demonstrates more sharp and fine-grained predictions.

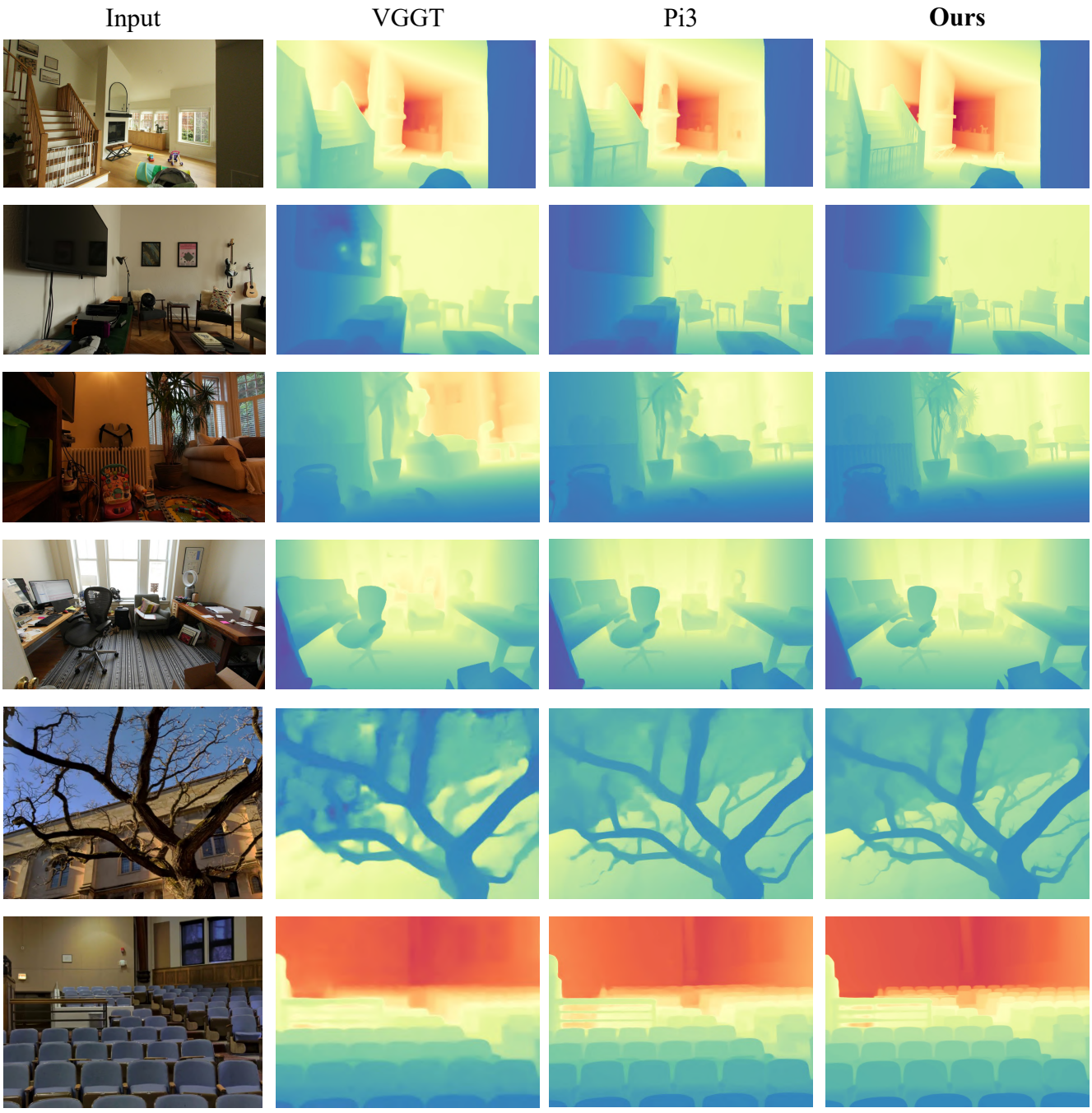


Figure 4. **Visualization of depth estimation** on static scenes.

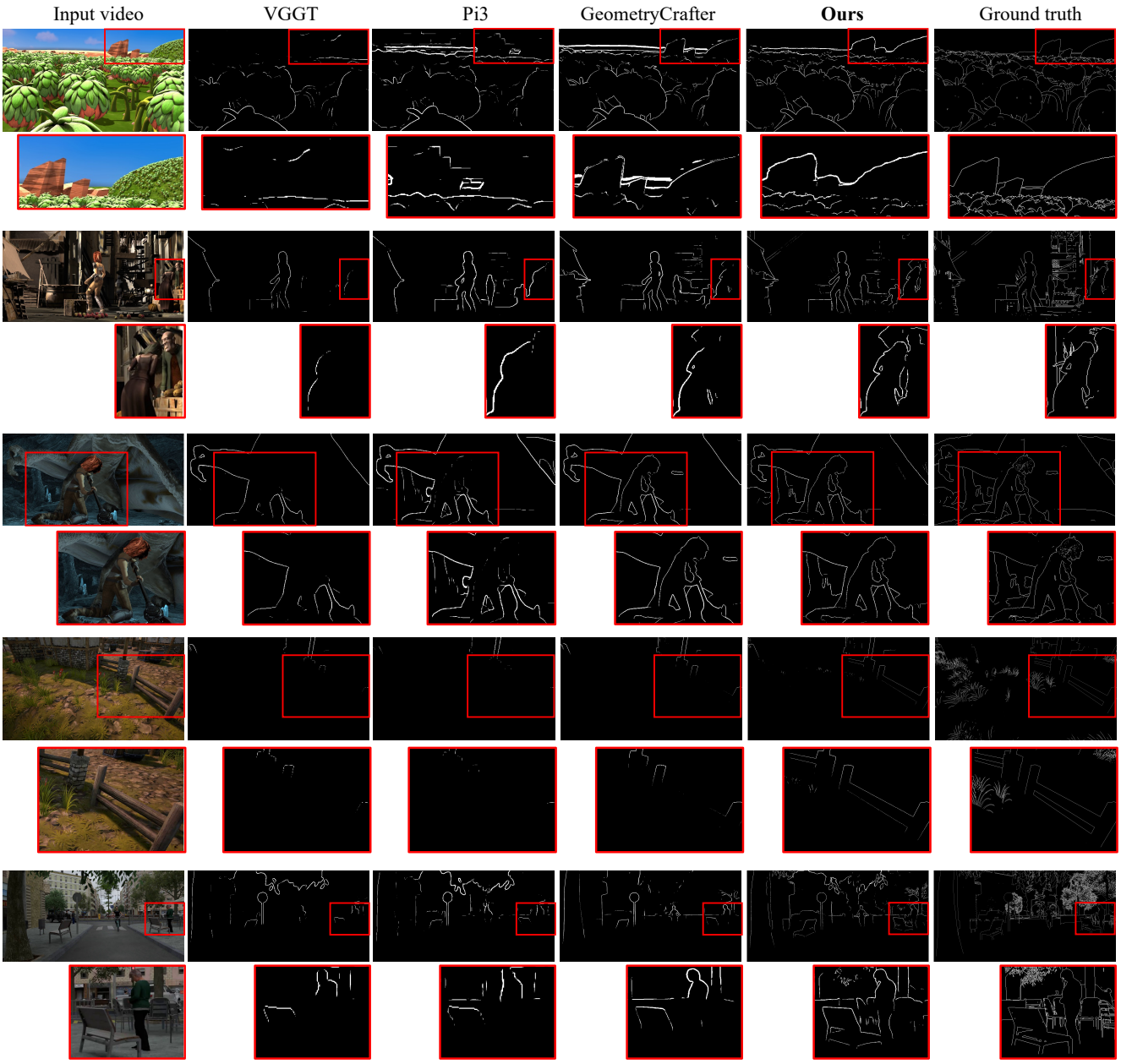


Figure 5. **Visualization of predicted depth edge maps**, which are defined by a depth ratio between neighboring pixels above a threshold. We zoom-in the edge map details in the red bounding boxes.

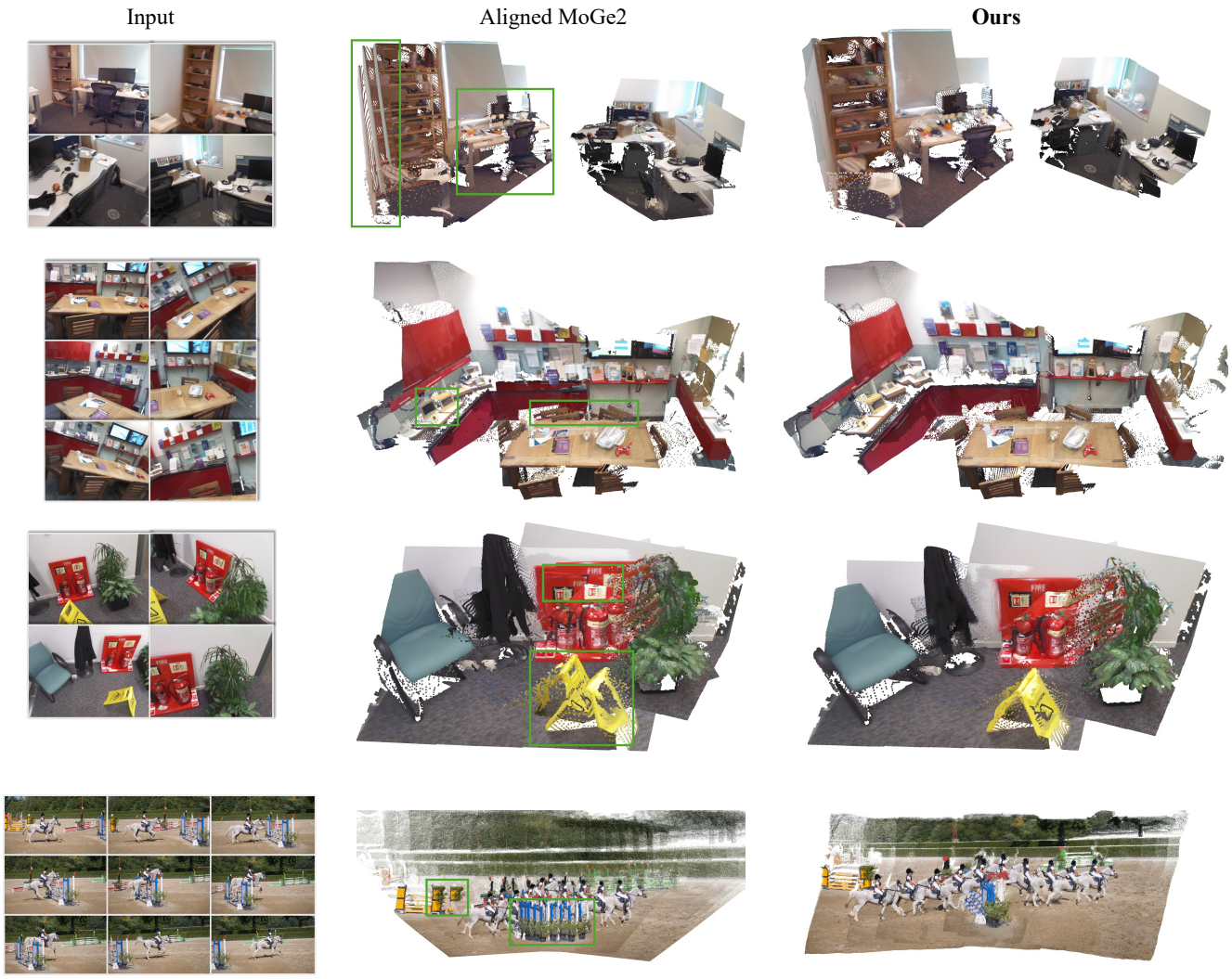


Figure 6. **Predicted 3D pointmaps** of the aligned MoGe2 baseline and our method. The aligned MoGe2 baseline exhibits layering artifacts (green boxes) due to the lack of strong multi-view binding.

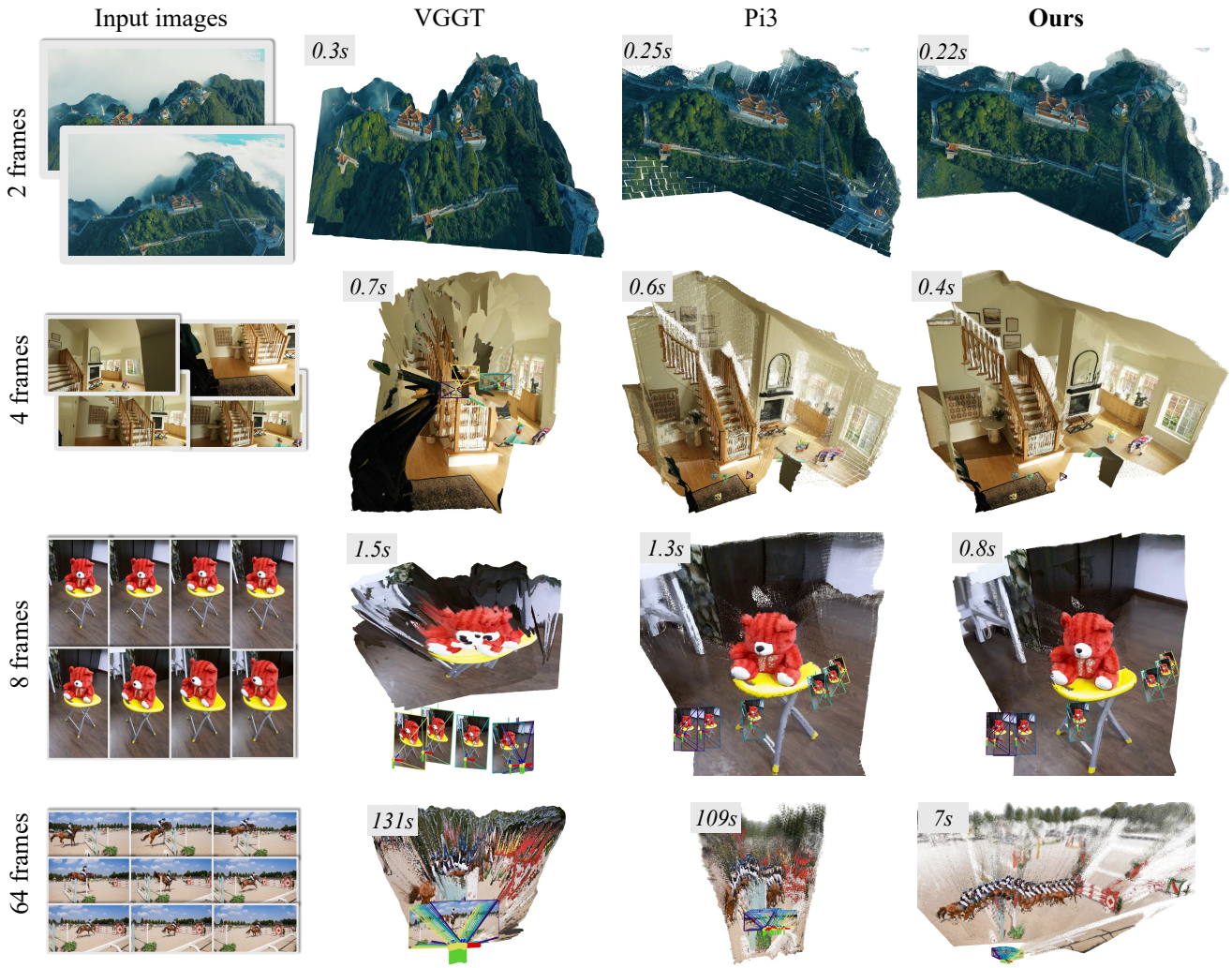
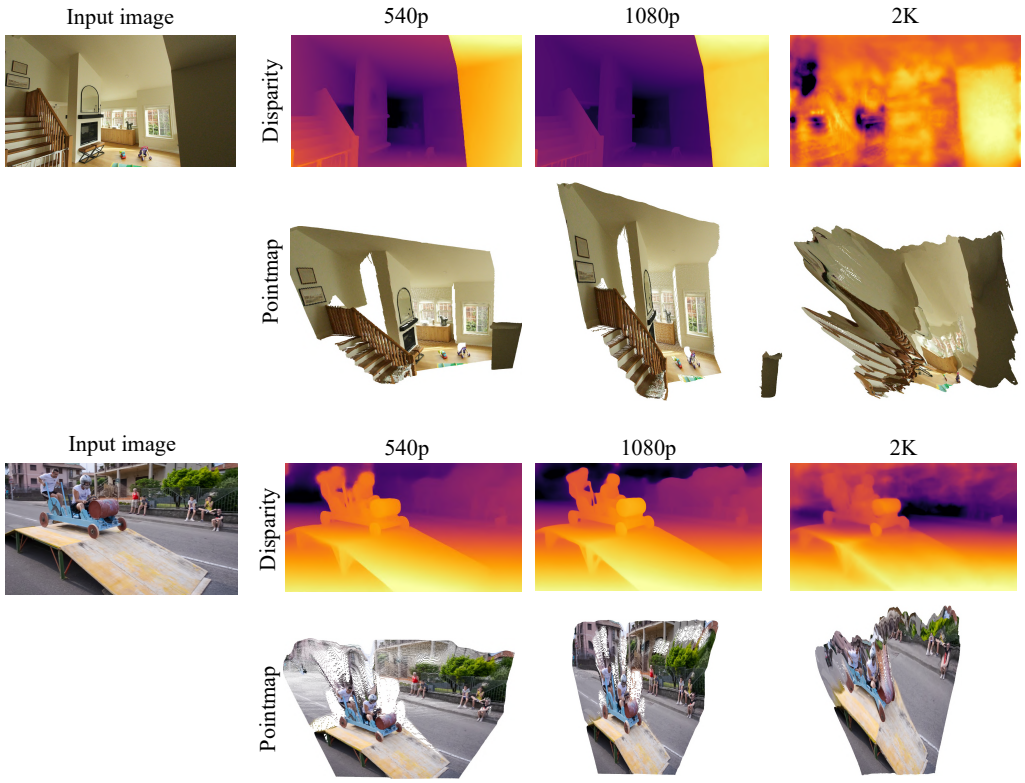
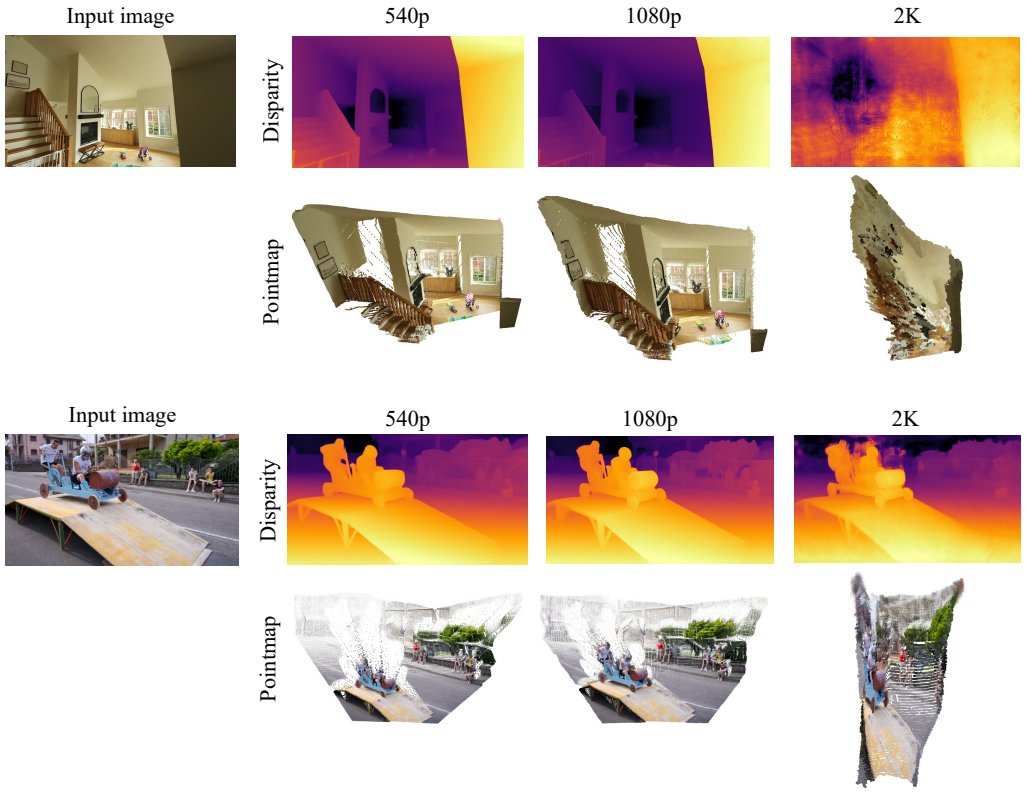


Figure 7. **Visualization of 3D reconstruction** with high-resolution inputs.



(a) **High-resolution single-image inference** of VGGT [41]



(b) **High-resolution single-image inference** of Pi3 [47]

Figure 8. Qualitative results for high-resolution single-image inference: (a) VGGT [41] and (b) Pi3 [47].

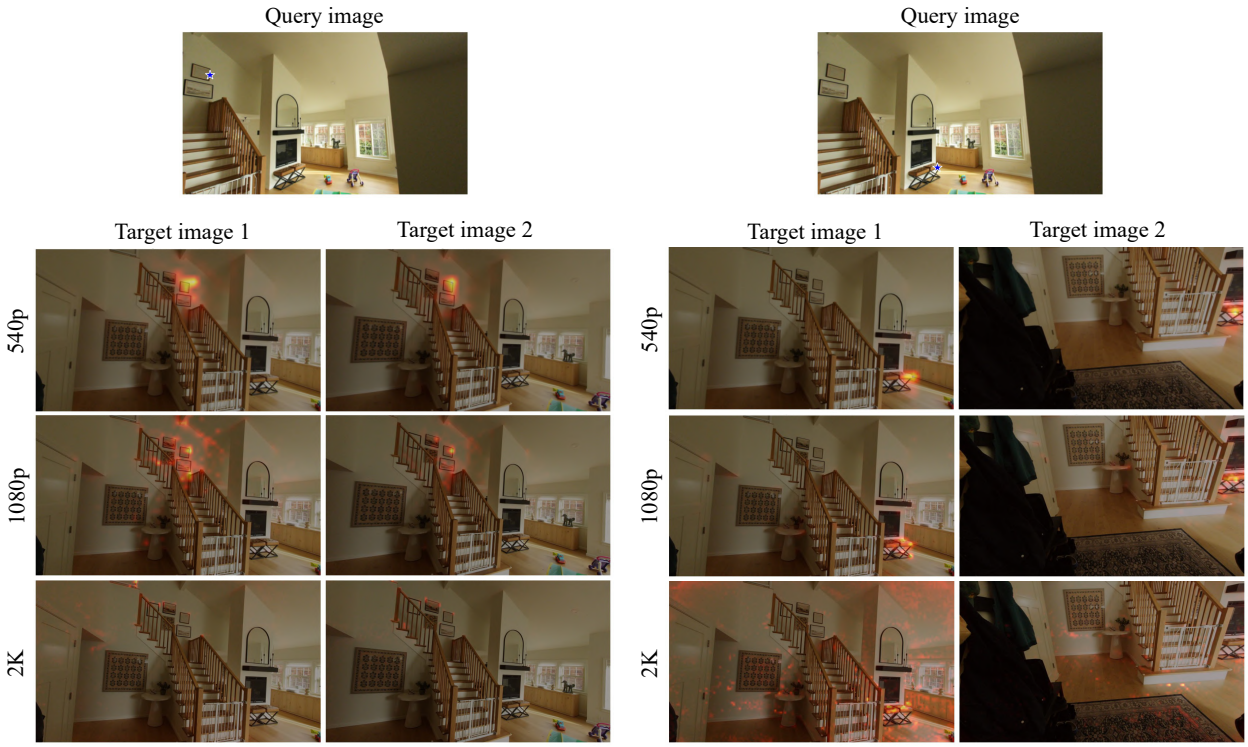


Figure 9. **Attention map** of the 15th global-attention layer of VGGT [41] at different input resolutions. The query token in the first image is marked with a blue star.

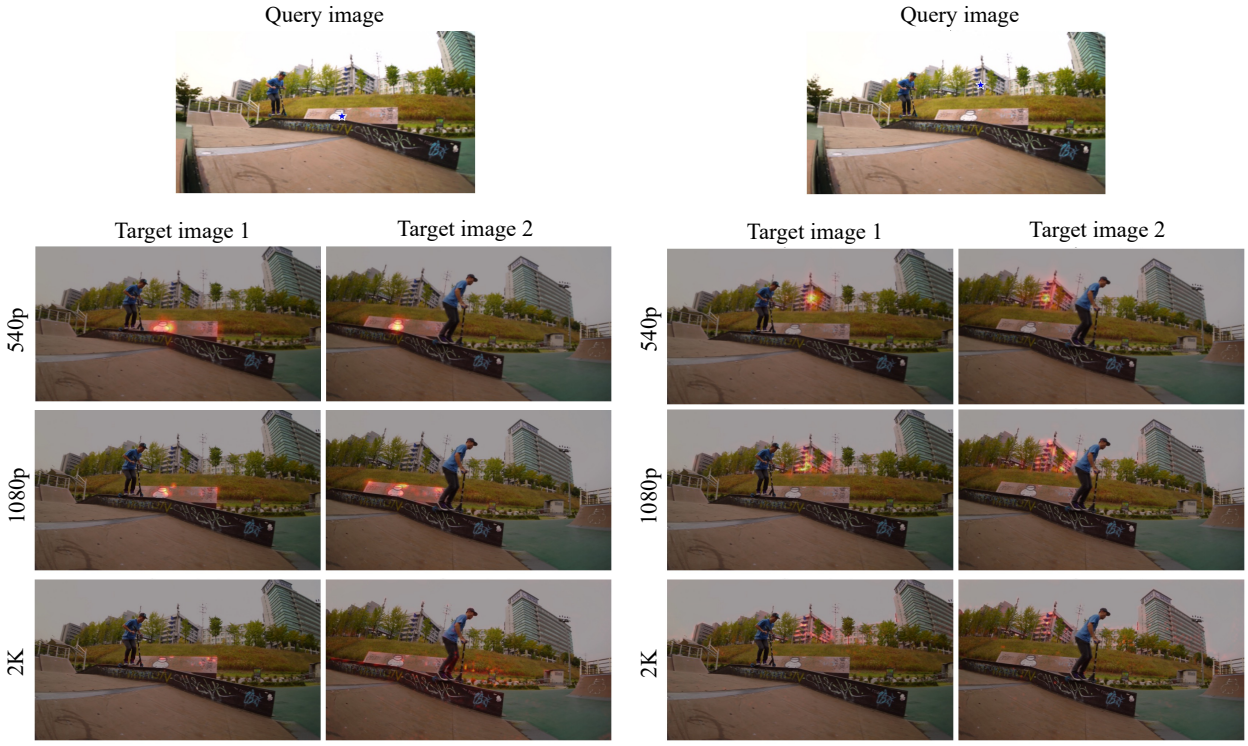


Figure 10. **Attention map** of the 15th global-attention layer of VGGT [41] at different input resolutions. The query token in the first image is marked with a blue star.

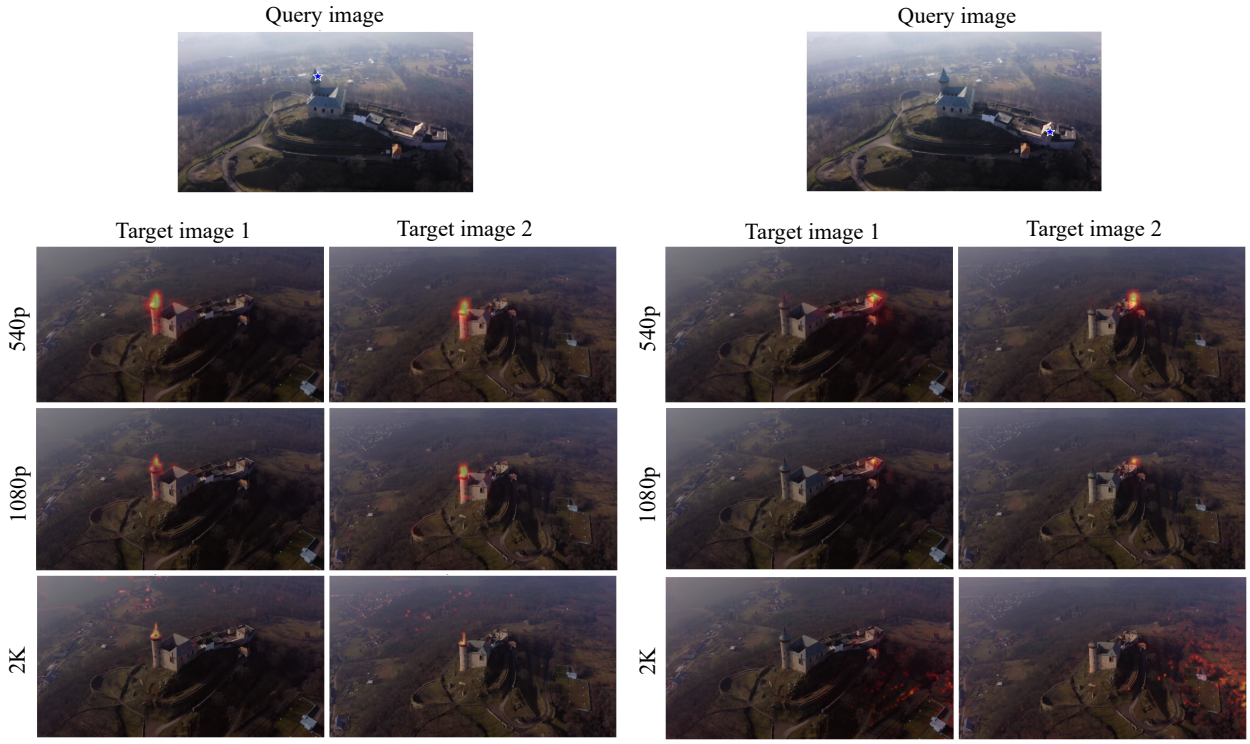


Figure 11. **Attention map** of the 15th global-attention layer of VGGT [41] at different input resolutions. The query token in the first image is marked with a blue star.

References

[1] Dejan Azinović, Ricardo Martin-Brualla, Dan B Goldman, Matthias Nießner, and Justus Thies. Neural rgb-d surface reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6290–6301, 2022. 2

[2] Gilad Baruch, Zhuoyuan Chen, Afshin Dehghan, Tal Dimry, Yuri Feigin, Peter Fu, Thomas Gebauer, Brandon Joffe, Daniel Kurz, Arik Schwartz, et al. Arkitscenes: A diverse real-world dataset for 3d indoor scene understanding using mobile rgb-d data. *arXiv preprint arXiv:2111.08897*, 2021. 1

[3] Michael J Black, Priyanka Patel, Joachim Tesch, and Jinlong Yang. Bedlam: A synthetic dataset of bodies exhibiting detailed lifelike animated motion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8726–8737, 2023. 1

[4] Aleksei Bochkovskii, AmaÃ**c**Gl Delaunoy, Hugo Germain, Marcel Santos, Yichao Zhou, Stephan R Richter, and Vladlen Koltun. Depth pro: Sharp monocular metric depth in less than a second. *arXiv preprint arXiv:2410.02073*, 2024. 2, 3, 4, 5

[5] Daniel J. Butler, Jonas Wulff, Garrett B. Stanley, and Michael J. Black. A naturalistic open source movie for optical flow evaluation. In *European Conference on Computer Vision*, 2012. 2, 3, 4, 5, 6

[6] Yohann Cabon, Naila Murray, and Martin Humenberger. Virtual kitti 2. *arXiv preprint arXiv:2001.10773*, 2020. 1

[7] Angel Chang, Angela Dai, Thomas Funkhouser, Maciej Halber, Matthias Niessner, Manolis Savva, Shuran Song, Andy Zeng, and Yinda Zhang. Matterport3d: Learning from rgb-d data in indoor environments. *arXiv preprint arXiv:1709.06158*, 2017. 1

[8] Shouyuan Chen, Sherman Wong, Liangjian Chen, and Yuandong Tian. Extending context window of large language models via positional interpolation. *ArXiv*, abs/2306.15595, 2023. 5

[9] Weirong Chen, Le Chen, Rui Wang, and Marc Pollefeys. Leap-vo: Long-term effective any point tracking for visual odometry. *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 19844–19853, 2024. 2

[10] Gene Chou, Wenqi Xian, Guandao Yang, Mohamed Abdelfattah, Bharath Hariharan, Noah Snavely, Ning Yu, and Paul Debevec. Flashdepth: Real-time streaming video depth estimation at 2k resolution. *arXiv preprint arXiv:2504.07093*, 2025. 3

[11] Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas A. Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2432–2443, 2017. 1, 2, 3, 4, 5, 6

[12] Tri Dao. Flashattention-2: Faster attention with better parallelism and work partitioning. *arXiv preprint arXiv:2307.08691*, 2023. 1

[13] Tri Dao, Dan Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. Flashattention: Fast and memory-efficient exact attention with io-awareness. *Advances in neural information processing systems*, 35:16344–16359, 2022. 1

[14] Laura Downs, Anthony Francis, Nate Koenig, Brandon Kinman, Ryan Hickman, Krista Reymann, Thomas B McHugh, and Vincent Vanhoucke. Google scanned objects: A high-quality dataset of 3d scanned household items. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 2553–2560. Ieee, 2022. 3, 5

[15] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The kitti dataset. *The International Journal of Robotics Research*, 32:1231 – 1237, 2013. 2, 3, 4, 5, 6

[16] Georgios Georgakis, Md. Alimoor Reza, Arsalan Mousavian, Phi Hung Le, and Jana Kosecka. Multiview rgb-d dataset for object instance detection. *2016 Fourth International Conference on 3D Vision (3DV)*, pages 426–434, 2016. 2, 3, 4, 5, 6

[17] Jos'e L. G'omez, Manuel Silva, Antonio Seoane, Agnes Borr'as, Mario Noriega, Germ'an Ros, Jose A. Iglesias-Guitian, and Antonio M. L'opez. All for one, and one for all: Urbansyn dataset, the third musketeer of synthetic driving scenes. *ArXiv*, abs/2312.12176, 2023. 2, 3, 4, 5, 6

[18] Vitor Campanholo Guizilini, Rares Ambrus, Sudeep Pillai, and Adrien Gaidon. 3d packing for self-supervised monocular depth estimation. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2482–2491, 2019. 3, 5

[19] Po-Han Huang, Kevin Matzen, Johannes Kopf, Narendra Ahuja, and Jia-Bin Huang. Deepmvs: Learning multi-view stereopsis. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2821–2830, 2018. 1

[20] Zhiyu Jin, Xuli Shen, Bin Li, and Xiangyang Xue. Training-free diffusion model adaptation for variable-sized text-to-image synthesis. *Advances in Neural Information Processing Systems*, 36:70847–70860, 2023. 5

[21] HyunJun Jung, Patrick Ruhkamp, Guangyao Zhai, Nikolas Brasch, Yitong Li, Yannick Verdie, Jifei Song, Yiren Zhou, Anil Armagan, Slobodan Ilic, et al. On the importance of accurate geometry data for dense 3d vision tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 780–791, 2023. 3, 5

[22] Nikita Karaev, Ignacio Rocco, Benjamin Graham, Natalia Neverova, Andrea Vedaldi, and Christian Rupprecht. Dynamicstereo: Consistent dynamic depth from stereo videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13229–13239, 2023. 1

[23] Nikhil Keetha, Norman Müller, Johannes Schönberger, Lorenzo Porzi, Yuchen Zhang, Tobias Fischer, Arno Knapitsch, Duncan Zauss, Ethan Weber, Nelson Antunes, Jonathon Luiten, Manuel López-Antequera, Samuel Rota Bulò, Christian Richardt, Deva Ramanan, Sebastian Scherer, and Peter Kontschieder. Mapanything: Universal feed-forward metric 3d reconstruction. *ArXiv*, abs/2509.13414, 2025. 1, 2, 4

[24] Tobias Koch, Lukas Liebel, Friedrich Fraundorfer, and Marco Korner. Evaluation of cnn-based single-image depth

estimation methods. In *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*, pages 0–0, 2018. 2, 3, 5

[25] Zhengqi Li and Noah Snavely. Megadepth: Learning single-view depth prediction from internet photos. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2041–2050, 2018. 1

[26] Ilya Loshchilov, Frank Hutter, et al. Fixing weight decay regularization in adam. *arXiv preprint arXiv:1711.05101*, 5 (5):5, 2017. 1

[27] Nikolaus Mayer, Eddy Ilg, Philip Häusser, Philipp Fischer, Daniel Cremers, Alexey Dosovitskiy, and Thomas Brox. A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4040–4048, 2015. 2, 3, 4, 5, 6

[28] Lukas Mehl, Jenny Schmalfuss, Azin Jahedi, Yaroslava Nalivayko, and Andrés Bruhn. Spring: A high-resolution high-detail dataset and benchmark for scene flow, optical flow and stereo. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4981–4991, 2023. 1

[29] Duc-Hai Pham, Tung Do, Phong Nguyen, Binh-Son Hua, Khoi Nguyen, and Rang Nguyen. Sharpdepth: Sharpening metric depth predictions using diffusion distillation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 17060–17069, 2025. 2

[30] Jeremy Reizenstein, Roman Shapovalov, Philipp Henzler, Luca Sbordone, Patrick Labatut, and David Novotny. Common objects in 3d: Large-scale learning and evaluation of real-life 3d category reconstruction. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10901–10911, 2021. 1

[31] Mike Roberts, Jason Ramapuram, Anurag Ranjan, Atulit Kumar, Miguel Angel Bautista, Nathan Paczan, Russ Webb, and Joshua M Susskind. Hypersim: A photorealistic synthetic dataset for holistic indoor scene understanding. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10912–10922, 2021. 1

[32] Thomas Schops, Johannes L Schonberger, Silvano Galliani, Torsten Sattler, Konrad Schindler, Marc Pollefeys, and Andreas Geiger. A multi-view stereo benchmark with high-resolution images and multi-camera videos. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3260–3269, 2017. 3, 5

[33] Jamie Shotton, Ben Glocker, Christopher Zach, Shahram Izadi, Antonio Criminisi, and Andrew Fitzgibbon. Scene coordinate regression forests for camera relocalization in rgb-d images. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2930–2937, 2013. 2

[34] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from rgbd images. In *European conference on computer vision*, pages 746–760. Springer, 2012. 3, 5

[35] Jürgen Sturm, Nikolas Engelhard, Felix Endres, Wolfram Burgard, and Daniel Cremers. A benchmark for the evaluation of rgb-d slam systems. *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 573–580, 2012. 2

[36] Jianlin Su, Yu Lu, Shengfeng Pan, Bo Wen, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *ArXiv*, abs/2104.09864, 2021. 4

[37] Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, et al. Scalability in perception for autonomous driving: Waymo open dataset. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2446–2454, 2020. 1

[38] Fabio Tosi, Yiyi Liao, Carolin Schmitt, and Andreas Geiger. Smd-nets: Stereo mixture density networks. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8938–8948, 2021. 2, 3, 4, 5, 6

[39] Igor Vasiljevic, Nicholas I. Kolkin, Shanyi Zhang, Ruotian Luo, Haochen Wang, Falcon Z. Dai, Andrea F. Daniele, Mohammadreza Mostajabi, Steven Basart, Matthew R. Walter, and Gregory Shakhnarovich. Diode: A dense indoor and outdoor depth dataset. *ArXiv*, abs/1908.00463, 2019. 2, 3, 4, 5, 6

[40] Chung-Shien Brian Wang, Christian Schmidt, Jens Piekenbrinck, and Bastian Leibe. Faster vggt with block-sparse global attention. *arXiv preprint arXiv:2509.07120*, 2025. 5

[41] Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. Vggt: Visual geometry grounded transformer. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 5294–5306, 2025. 1, 3, 4, 5, 8, 9, 14, 15, 16

[42] Kaixuan Wang and Shaojie Shen. Flow-motion and depth network for monocular stereo and beyond. *IEEE Robotics and Automation Letters*, 5(2):3307–3314, 2020. 1

[43] Qianqian Wang, Yifei Zhang, Aleksander Holynski, Alexei A Efros, and Angjoo Kanazawa. Continuous 3d perception model with persistent state. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 10510–10522, 2025. 1, 2, 3, 4, 5

[44] Ruicheng Wang, Sicheng Xu, Cassie Dai, Jianfeng Xiang, Yu Deng, Xin Tong, and Jiaolong Yang. Moge: Unlocking accurate monocular geometry estimation for open-domain images with optimal training supervision. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 5261–5271, 2025. 1, 3, 4, 5

[45] Ruicheng Wang, Sicheng Xu, Yue Dong, Yu Deng, Jianfeng Xiang, Zelong Lv, Guangzhong Sun, Xin Tong, and Jiaolong Yang. Moge-2: Accurate monocular geometry with metric scale and sharp details. *arXiv preprint arXiv:2507.02546*, 2025. 1, 3, 4, 5

[46] Wenshan Wang, Delong Zhu, Xiangwei Wang, Yaoyu Hu, Yuheng Qiu, Chen Wang, Yafei Hu, Ashish Kapoor, and Sebastian Scherer. Tartanair: A dataset to push the limits of visual slam. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4909–4916. IEEE, 2020. 1

[47] Yifan Wang, Jianjun Zhou, Haoyi Zhu, Wenzheng Chang, Yang Zhou, Zizun Li, Junyi Chen, Jiangmiao Pang, Chunhua Shen, and Tong He. Scalable permutation-equivariant visual

geometry learning. *arXiv preprint arXiv:2507.13347*, 2025. 1, 2, 3, 4, 5, 8, 9, 14

[48] Hongchi Xia, Yang Fu, Sifei Liu, and Xiaolong Wang. Rgbd objects in the wild: Scaling real-world 3d object learning from rgb-d videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22378–22389, 2024. 1

[49] Tian-Xing Xu, Xiangjun Gao, Wenbo Hu, Xiaoyu Li, Song-Hai Zhang, and Ying Shan. Geometrycrafter: Consistent geometry estimation for open-world videos with diffusion priors. *arXiv preprint arXiv:2504.01016*, 2025. 1, 3, 4, 8, 9

[50] Yao Yao, Zixin Luo, Shiwei Li, Jingyang Zhang, Yufan Ren, Lei Zhou, Tian Fang, and Long Quan. Blendedmvs: A large-scale dataset for generalized multi-view stereo networks. *Computer Vision and Pattern Recognition (CVPR)*, 2020. 1

[51] Chandan Yeshwanth, Yueh-Cheng Liu, Matthias Nießner, and Angela Dai. Scannet++: A high-fidelity dataset of 3d indoor scenes. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12–22, 2023. 1

[52] Junyi Zhang, Charles Herrmann, Junhwa Hur, Varun Jampani, Trevor Darrell, Forrester Cole, Deqing Sun, and Ming-Hsuan Yang. Monst3r: A simple approach for estimating geometry in the presence of motion. *arXiv preprint arXiv:2410.03825*, 2024. 2

[53] Yang Zheng, Adam W Harley, Bokui Shen, Gordon Wetzstein, and Leonidas J Guibas. Pointodyssey: A large-scale synthetic dataset for long-term point tracking. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 19855–19865, 2023. 1