MANY EYES, ONE WORLD: FEED-FORWARD 3D RECONSTRUCTION FROM MIXED CAMERAS

Anonymous authors
Paper under double-blind review

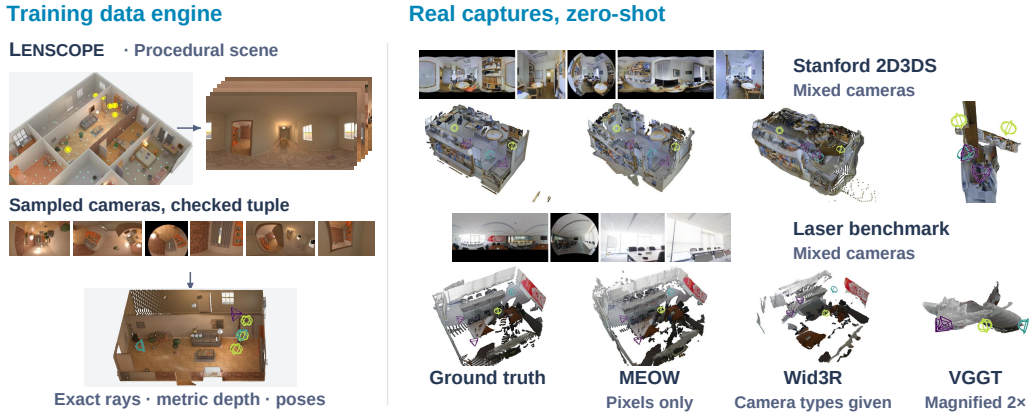

Figure 1: MEOW reconstructs a shared 3D scene from mixed-camera images in one forward pass without supplied camera calibration or camera-type labels. Left: LENSCOPE generates mixed-camera training views from procedural scenes, checks their covisibility, and provides exact rays, metric depth, and camera poses for supervision. Yellow markers indicate the selected panorama centres. Right: reconstructions of mixed-camera tuples constructed from Stanford 2D3DS and laser-scanned panoramic captures. MEOW receives images only, while Wid3R additionally receives camera-type labels. Each point cloud is independently similarity-aligned to the ground truth for visualisation; VGGT panels are magnified 2×.

ABSTRACT

Real-world capture is heterogeneous: perspective, fisheye, and 360° panoramic images can coexist within a single reconstruction task, yet most feed-forward 3D reconstruction models assume perspective imagery and a uniform input representation. Recent models handling several camera types are either informed of the camera type for each view or reconstruct one image pair at a time. No single-pass method reconstructs mixed-camera tuples containing full panoramas from images alone. We present MEOW, a feed-forward system that jointly reconstructs metric pointmaps and camera poses from one N-view tuple mixing perspective, fisheye and full-panorama images, in a single forward pass from images alone: no calibration, distortion parameters, camera-type labels or poses are supplied for any view. Our guiding design philosophy is to treat heterogeneous-camera reconstruction as a data-adaptation problem rather than an architectural redesign. MEOW retains a perspective-pretrained backbone and learns heterogeneous cameras entirely from a procedural data engine, which renders each scene across a continuous manifold of camera models with exact rays and depth, and certifies covisibility for every training tuple. Trained on synthetic tuples only, MEOW transfers zero-shot to real captures: on heterogeneous 2D3DS tuples it achieves 79.9 mAA@30 against 53.8 for Wid3R given the camera type of every view; on our laser-scanned mixed-camera benchmark it registers every four-view mixed tuple with 79.4 AUC@30. The data engine, benchmark, and complete evaluation pipeline will be released.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

1 INTRODUCTION

From surveillance installations to consumer panoramas and robotic rigs, real-world capture mixes perspective, fisheye, and panoramic cameras with different projection geometries, fields of view, and aspect ratios. Reconstructing these images together requires relating observations across camera types and recovering their shared 3D geometry, often without reliable calibration. Feed-forward reconstruction models, including DUSt3R (Wang et al., 2024), VGGT (Wang et al., 2025a), π^3 (Wang et al., 2026), and MapAnything (Keetha et al., 2026), provide a foundation for this task, but are developed primarily for perspective imagery. Their training distributions and input processing do not adequately cover heterogeneous projections and full panoramas, limiting their performance on mixed-camera tuples (Figure 1, Table 3).

Existing approaches address parts of this problem. Rectifying wide-angle images to perspective views requires calibration and can discard field of view. Wid3R (Jung et al., 2026) jointly processes multiple camera models but requires a camera-model token for each view. CAM3R (Guruprasad et al., 2026) removes this requirement, but processes image pairs; reconstructing a larger collection requires repeated pairwise inference followed by global alignment. Extending joint reconstruction to mixed-camera tuples without supplied camera information presents two challenges. First, diverse multi-view training tuples that combine perspective, fisheye, and full-panorama images with accurate geometric supervision are scarce. Second, a shared input pipeline must accommodate their different image layouts without discarding the coverage needed for reconstruction.

We present MEOW, which adapts the publicly available MapAnything checkpoint to jointly reconstruct metric pointmaps and camera poses from tuples mixing perspective, fisheye, and full-panorama images. Reconstruction takes one network forward pass, without externally supplied calibration, distortion parameters, camera-type labels, or poses. Our approach retains the perspective-pretrained backbone and uses synthetic data for all subsequent adaptation. LENSCOPE supplies geometrically supervised mixed-camera training tuples, while lightweight input adaptations preserve the coverage and layout information needed to process these views together.

LENSCOPE generates camera diversity from procedurally constructed indoor scenes (Section 3.2). At each sampled optical centre, we render a full panorama with exact rays, metric depth, and camera pose, then resample it into perspective, fisheye, or panoramic views with varying camera parameters. This construction varies field of view, distortion, and image layout while retaining geometric supervision. Camera sampling also changes which parts of the scene remain visible: two overlapping panoramic observations may produce narrow views with little shared content. We therefore recompute covisibility after sampling the cameras and retain tuples whose views form a connected covisibility graph. The resulting training stream combines diverse camera projections with overlapping observations that support joint reconstruction.

To process heterogeneous views together, we resize every image to a common tensor shape without cropping (Section 3.3). This preserves the complete image coverage, including all longitudes of a full panorama. Resizing can change aspect ratios, however, so the resulting tensor shape no longer identifies the source layout. An aspect-ratio embedding supplies this information to the network, while circular padding in the dense prediction head handles the longitude boundary of full panoramas. These panoramas are identified from their images at inference, without supplied camera-type labels. We also adapt geometric supervision to the different projections: solid-angle weighting accounts for unequal directional coverage per pixel, and a directional likelihood supervises predicted rays (Section 3.4).

We evaluate MEOW on mixed-camera tuples constructed from real panoramic captures, with perspective and fisheye views resampled from the original panoramas. On heterogeneous Stanford 2D3DS tuples, MEOW reaches 79.9 mAA@30, compared with 53.8 for Wid3R supplied with camera-type labels. On our laser-scanned benchmark, where all methods are evaluated without fine-tuning, MEOW reaches 79.4 AUC@30 on mixed tuples, compared with 29.3 for Wid3R. Controlled comparisons show that synthetic adaptation substantially improves the pretrained backbone, whereas changing its input resizing alone does not. Further ablations show that full-field-of-view resizing and the aspect-ratio embedding contribute to the adapted model's performance.

Our contributions are threefold:

- We adapt a perspective-pretrained reconstruction model to jointly recover metric pointmaps and camera poses from mixed-camera tuples containing full panoramas, in one forward pass and without externally supplied calibration, distortion parameters, camera-type labels, or poses.
- We develop LENSCOPE, a reproducible procedural data engine that generates heterogeneous camera views with exact geometric supervision and checks covisibility after camera sampling, enabling adaptation using synthetic tuples only.
- We introduce a laser-scanned benchmark with mixed-camera and single-camera tracks, and evaluate reconstruction methods without fine-tuning on it. We will release the data engine, benchmark, and evaluation pipeline.

2 RELATED WORK

Feed-forward reconstruction. Feed-forward reconstruction has moved from image pairs to large view collections. DUSt3R (Wang et al., 2024) predicts pointmaps from two images, and MASt3R (Leroy et al., 2024) adds matching. MUSt3R, Fast3R, and CUT3R (Cabon et al., 2025; Yang et al., 2025; Wang et al., 2025b) extend this line to many views, while VGGT and π^3 (Wang et al., 2025a; 2026) predict joint N-view geometry. MapAnything (Keetha et al., 2026) adds metric scale and optional geometric cues; Depth Anything 3 (Lin et al., 2026a) accommodates variable view counts. Yet these models remain centred on perspective images in a common tensor layout. Rig3R (Li et al., 2025) uses rig metadata when available and infers it otherwise, while Pow3R (Jang et al., 2025) accepts optional camera and depth priors. We build on the MapAnything backbone.

Non-pinhole cameras in feed-forward reconstruction. Existing methods differ chiefly in what camera information they require. Wid3R (Jung et al., 2026) needs a per-view camera token, which UCE (Chidlovskii et al., 2026) infers from the image. Rig3R (Li et al., 2025) can infer missing rig structure but targets perspective rigs; Fisheye3R (Duan et al., 2026) gates calibration tokens by camera type; and X-Lens (Zhou et al., 2026) takes per-pixel calibration and a camera type for every view and predicts metric depth but no poses. CAM3R (Guruprasad et al., 2026) needs no calibration, but reconstructs two views at a time and aligns them afterward. PanoVGGT, CasaMaestro, and Argus (Guo et al., 2026; Ji et al., 2026; Li et al., 2026c) handle panoramas alone, while RIGOR and HALO-SLAM (Huang et al., 2026; Xiong et al., 2026) use frozen foundation models for panoramic SLAM. Ray-aware alternatives use supplied rays for mixed pinhole/fisheye views (Zhang et al., 2026), adapt to one fisheye camera per sequence (Sinitsyn et al., 2026), or address novel-view synthesis and calibrated rig depth (Griffiths & Dansereau, 2026; Zhang et al., 2025). Among the systems in Table 1, MEOW alone combines mixed tuples containing full panoramas, no supplied calibration, distortion, or camera-type labels, one joint forward pass, and metric pointmaps with poses.

Monocular any-camera geometry and calibration. Monocular depth has expanded beyond perspective cameras (Piccinelli et al., 2025; Guo et al., 2025; Ganesan et al., 2026; Li et al., 2026b; Lin et al., 2026b; Gangopadhyay et al., 2025); OmniPoint (Ye et al., 2026) and PaGeR (Bozic et al., 2026) recover metric geometry from an arbitrary-camera image and a panorama, respectively. Another line estimates the camera model from one image (Veicht et al., 2024; Tirado-Garín & Civera,

Table 1: Input requirements and outputs of the closest systems, read from their papers and code (all cited in Section 2). ✓: supported; (✓): partial or qualitative by the authors' own account; –: not supported or not claimed. Only MEOW takes a mixed tuple with full panoramas from pixels alone and returns metric pointmaps and poses in one pass.

Method	Camera input at test time	Mixed tuple	Full 360°	N-view pass	Poses	Metric
Wid3R	Class per view	✓	✓	✓	✓	–
CAM3R	None	(✓) pairs	✓	–	✓	–
Fisheye3R	None	(✓)	(✓)	✓	✓	–
X-Lens	Intrinsics or rays, and class	✓	–	✓	–	✓
G-ray	Ray map per view	✓	–	✓	✓	–
PanoVGGT	Panoramas only	–	✓	✓	✓	–
MEOW (ours)	None	✓	✓	✓	✓	✓

2025; Bogdan et al., 2018; Lochman et al., 2021) or from sparse views (Li et al., 2026a). We let multiple views constrain one another, jointly recovering geometry and poses without supplied camera parameters.

Camera models and synthetic data. Our camera manifold draws on classical models (Kannala & Brandt, 2006; Geyer & Daniilidis, 2000; Khomutenko et al., 2016; Usenko et al., 2018; Mei & Rives, 2007). Synthesising cameras from panoramas (Bogdan et al., 2018; Guruprasad et al., 2026; Jung et al., 2026) and generating procedural multi-view data (Ma et al., 2026) are established practices. Existing resources span fixed indoor scenes (Structured3D, PanoCity, OmniRooms (Zheng et al., 2020; Guo et al., 2026; Song et al., 2026)), procedural scenes (PanoInfinigen (Bozic et al., 2026)), curated asset views (CM-EVS (Liu et al., 2026)), and real sequences with laser geometry (Holo360D (Ou et al., 2026)). We generate scenes reproducibly from seeds using procedural geometry and CC0 textures on a single GPU. Our engine samples a continuous camera manifold with photographic priors over principal point, roll, tilt, and optical blur, checks covisibility for each tuple, and outputs exact rays and depth for views with distinct optical centres.

3 METHOD

3.1 PROBLEM FORMULATION

Given N images captured by unknown and potentially different camera models, our goal is to reconstruct metric pointmaps and camera poses in a single forward pass. Writing $I_i \in \mathbb{R}^{H_i \times W_i \times 3}$ for the i-th image, MEOW is one network f_θ with

$$f_\theta\big(\{I_i\}_{i=1}^N\big) = \Big(\{\mathbf{r}_i,\, d_i,\, \mathbf{R}_i,\, \mathbf{t}_i\}_{i=1}^N,\; s\Big), \qquad \mathbf{X}_i(\mathbf{p}) = s\,\big(\mathbf{R}_i\, d_i(\mathbf{p})\, \mathbf{r}_i(\mathbf{p}) + \mathbf{t}_i\big), \tag{1}$$

where $\mathbf{r}_i(\mathbf{p}) \in \mathbb{S}^2$ is the unit ray of pixel $\mathbf{p}$ in the camera frame of view i, $d_i(\mathbf{p}) > 0$ the depth along that ray, $(\mathbf{R}_i, \mathbf{t}_i)$ the camera-to-world pose in the frame of the first view (OpenCV convention), $s > 0$ one metric scale for the tuple and $\mathbf{X}_i$ the resulting metric pointmap. At inference, no camera types, intrinsics, distortion parameters, or poses are externally supplied; the only auxiliary cues are each image's native shape (H_i, W_i) and a binary panorama flag π_i inferred from its pixels. We study whether high-quality synthetic data can lift a perspective-pretrained geometry model to this mixed-camera task. Starting from the publicly available MapAnything checkpoint, we perform all subsequent adaptation on synthetic mixed-camera tuples. We first present LENSCOPE, our synthetic data engine (Section 3.2); then describe how images retain their full field of view within a shared tensor shape (Section 3.3); and finally detail the model, losses and training procedure (Section 3.4).

3.2 LENSCOPE: THE CAMERA-MANIFOLD ENGINE

Figure 2 draws the pipeline: an offline stage renders and indexes every scene once, and an online stage assembles one tuple per training sample. Appendix B lists the parameters and gives the complete procedure as Algorithm 1. Scenes come from our own procedural indoor generator, in the spirit of ProcTHOR (Deitke et al., 2022), in two generations: 2,004 first-generation rooms and 499 denser and more realistic second-generation scenes. Panorama poses fill the free space of each scene, and every pose k is rendered once as an equirectangular panorama E_k with analytic unit rays, radial depth cast against the scene mesh and a validity mask. Panoramic sources allow fields of view up to 360° with exact rays and depth. Fisheye views that are warped from perspective frames, by contrast, cannot exceed the field of view of their source.

Covisibility. Two forms of one visibility test are used. Offline, between the panorama poses k, l of a scene, $\mathrm{cov}(k \to l)$ is estimated with 96 directions drawn uniformly on the sphere at pose k. Each direction is cast to the nearest solid, unambiguous surface point, which counts as covisible if the straight line from pose l reaches it unoccluded (within 5 cm). The stored value is $c_{kl} = \min\big(\mathrm{cov}(k \to l), \mathrm{cov}(l \to k)\big)$, an equal-solid-angle estimate that does not depend on the panorama grid. Online, on the sampled views, the test runs on the pixel grid of each view rendered at 96 pixels on the long side. A valid pixel $\mathbf{p}$ of view i is covisible in j when its 3D point, expressed in the frame of j as $\mathbf{Y} = \mathbf{R}_j^\top\big(\mathbf{X}_i(\mathbf{p}) - \mathbf{t}_j\big)$, lies in front of j, falls within 3.6° of the nearest ray

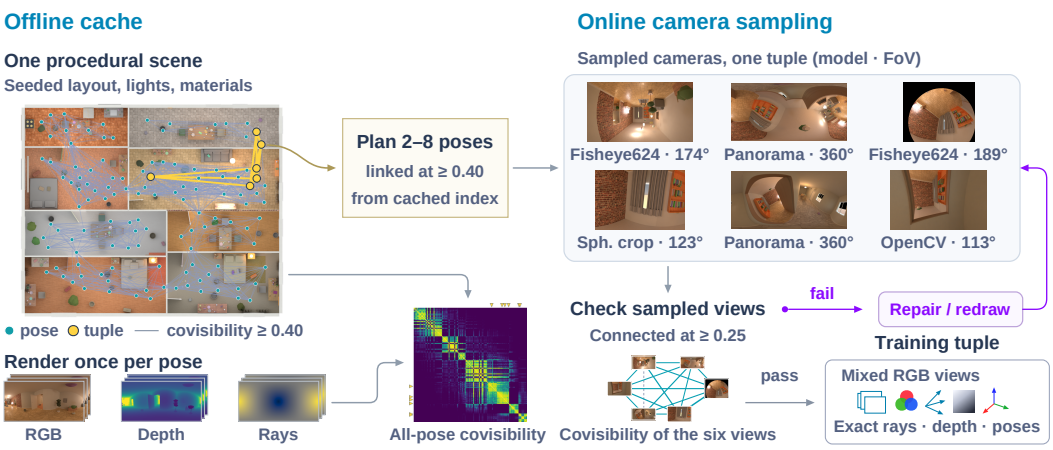

Figure 2: Offline, once per scene: a seed generates a procedural scene, panorama poses fill its free space, each pose is rendered with exact rays and depth; pairwise covisibility is computed by mutual visibility tests (lines: $c_{kl} \geq 0.40$). Online, per tuple: a connected walk plans 2–8 poses and each is resampled through a sampled camera. Covisibility is recomputed (edge width; dashed below 0.25) and only connected tuples are kept.

$\mathbf{q}^\star = \arg\max_{\mathbf{q}} \langle \mathbf{Y}/\|\mathbf{Y}\|, \mathbf{r}_j(\mathbf{q}) \rangle$ of a valid pixel, and is not occluded there:

$$\big|\, \|\mathbf{Y}\| - d_j(\mathbf{q}^\star) \big| \leq 0.10\,\mathrm{m} + 0.05\, d_j(\mathbf{q}^\star). \tag{2}$$

$\mathrm{cov}(i \to j)$ is the fraction of valid pixels of i that pass, and the online check uses $c_{ij} = \frac{1}{2}\big(\mathrm{cov}(i \to j) + \mathrm{cov}(j \to i)\big)$. The two directions differ by construction: a narrow view that lies inside the field of a panorama is covered entirely while the panorama is not. Each direction is normalised by its own sample count, so views of different resolution and field of view are comparable. Covisibility is recomputed online because a narrow or tilted camera sees a fraction of what its panorama saw.

Tuple sampling and camera manifold. A tuple of $K \in [2, 8]$ poses with distinct optical centres is drawn by a random walk on the offline matrix that adds a pose only if its covisibility with the current pose is at least 0.40 (relaxed to 0.20 when no chain of length K exists). With probability 0.55 the cameras of the tuple are aimed at a common surface point; otherwise each camera keeps the orientation of its panorama pose. Each pose is then resampled through a camera drawn from seven models with coupled field-of-view ranges (Table 5, Appendix B). Rectilinear OpenCV and pinhole cameras form the majority; Fisheye624, EUCM and Mei cover the fisheye family; spherical crops and full panoramas cover the sphere. The field of view follows a triangular density peaked at the common focal length of the model, and distortion coefficients are drawn from ranges valid for that field of view. Full panoramas receive a random $SO(3)$ content rotation. Principal-point shifts and roll follow priors set from calibration statistics and photographic collections (Clarke et al., 1998; Hold-Geoffroy et al., 2023). An optics layer (modulation transfer, vignetting, sensor noise) is decoupled from geometry, so blur is a weaker cue to the field of view.

Covisibility check and repair. The sampled views are rendered at low resolution, c_{ij} is recomputed with Eq. (2), and the tuple is kept only if the graph with edges $c_{ij} > 0.25$ is connected. A disconnected view is pulled toward its native orientation and field of view in three steps and the graph is re-tested; a tuple that still fails is discarded. Kept views are rendered once more at their target resolution. No real image of any camera type enters training after the public perspective pre-training. Scene realism matters: under the same configuration, second-generation scenes instead of first-generation renders lower Matterport3D (Chang et al., 2017) zero-shot accuracy and completeness error by 17.4% and 24.1%. Table 7 (Appendix D) contrasts this construction with the training data of the compared methods, read from their released code.

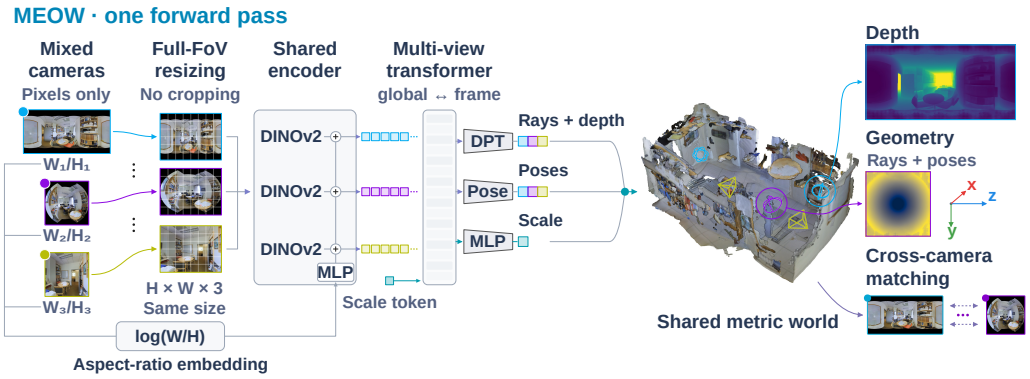

Figure 3: One forward pass. Views from different cameras are resized to one tensor shape with their full field of view; the native aspect of each (fixed to 2:1 for detected full panoramas) enters through an MLP added to its patch tokens. Sixteen alternating global and frame attention layers, a dense head per view (circular padding for panoramas), a pose head and a scale head give rays, depth, camera-to-world poses (OpenCV axes) and metric scale in one shared world. Inputs: a real tuple, zero-shot.

3.3 Mixed-camera inputs

Resizing without cropping. All views of a tuple enter the network at one tensor shape (h, w) drawn from ten buckets $\mathcal{B}$ whose sides are multiples of 14 and whose width-to-height ratios range from 0.49 to 3.08. View i is resized anisotropically, $\tilde{I}_i(u, v) = I_i(uW_i/w,\ vH_i/h)$, so that the tensor grid covers the entire native image and nothing is cropped (Figure 7, Appendix D). Ground-truth rays, depth and masks are resampled through the same map, and rays keep their directions. During training the bucket is drawn per batch independently of the content, so the deformation $\delta_i = (w/h)\,/\,(W_i/H_i)$ of each view is random. At inference the bucket closest to the mean aspect ratio of the tuple is used for all views.

Aspect-ratio embedding. Since the bucket is drawn independently of the content, the tensor shape no longer tells the network the native aspect of a view. A small MLP g with a zero-initialised last layer maps $a_i = \log(W_i/H_i)$ to a vector that is added to every patch token of view i, $\mathbf{F}_i \leftarrow \mathbf{F}_i + \mathbf{1}\, g(a_i)^\top$, following the global-representation encoder of MapAnything. The embedding holds 0.05% of the parameters.

Panorama flag, wrap and detection. A full equirectangular panorama is periodic in longitude. For views with $\pi_i = 1$ the dense head pads its token grid with three circular columns on each side, runs, and crops back, so no parameter is added, and the aspect input takes its definitional value $a_i = \log 2$ whatever the shape in which the panorama arrives; other views run the head unchanged. In training π_i is read from the ground-truth rays (azimuth span of the middle row above $350°$); at inference it comes from a detector. A full equirectangular image has two signatures that survive any aspect squeeze: its last and first columns are adjacent in content, and its top and bottom rows compress all longitudes into near-constant colour. The detector measures both on a 128-row copy of the image as ratios against its interior, with a guard against circular fisheyes with black corners; its thresholds were fixed on training renders only (Appendix C). On the two headline benchmarks it misses no panorama and flags 2 of 386 other views (Table 6, Appendix C).

3.4 Model, training and inference

Architecture. Figure 3 presents the overall system, whose reconstruction network retains the original MapAnything backbone without architectural modification. The first 24 layers of a DINOv2 (Oquab et al., 2024) ViT-G encoder, shared across views, turn each resized view into patch tokens $\mathbf{F}_i \in \mathbb{R}^{(hw/14^2)\times 1536}$, to which $g(a_i)$ is added. 16 transformer layers (width 1536, 24 heads) alternate global attention over the tokens of all views together with one learnable scale token, and frame attention within each view. One DPT (Ranftl et al., 2021) dense head, applied to every view

with the wrap of Section 3.3, predicts $\mathbf{r}_i$, d_i, a confidence and a validity logit. One pose head per view (convolutional pooling and an MLP) predicts $\mathbf{R}_i$ as a unit quaternion and $\mathbf{t}_i$, and a scale head on the scale token predicts s. Pointmaps follow from Eq. (1).

Losses. Ground truth carries an asterisk and m_i is the valid mask of view i. Two changes make the MapAnything objective aware of non-perspective geometry. First, every dense term of view i is weighted by the solid angle of its ground-truth ray field,

$$\omega_i(\mathbf{p}) = \frac{\big\|\partial_u \mathbf{r}_i^* \times \partial_v \mathbf{r}_i^*\big\|(\mathbf{p})}{\mathrm{mean}_{\mathbf{p}' \in m_i} \big\|\partial_u \mathbf{r}_i^* \times \partial_v \mathbf{r}_i^*\big\|(\mathbf{p}')}, \qquad \omega_i \in [0.05,\, 20], \tag{3}$$

normalised to mean one over the valid pixels of each view, so that the poles of a panorama and the rim of a fisheye, which the pixel grid over-represents, no longer dominate; on a pinhole view $\omega_i \approx 1$. Second, ray directions are scored by a von Mises–Fisher likelihood on $\mathbb{S}^2$ (Mardia & Jupp, 2000) with one learned concentration κ,

$$\mathcal{L}_{\mathrm{ray}} = \sum_i \sum_{\mathbf{p} \in m_i} \omega_i(\mathbf{p}) \Big[-\kappa \left\langle \mathbf{r}_i(\mathbf{p}), \mathbf{r}_i^*(\mathbf{p}) \right\rangle + \log \frac{4\pi \sinh \kappa}{\kappa} \Big], \tag{4}$$

whose normaliser keeps a finite optimum in κ for any non-zero residual. The remaining terms follow MapAnything. With $\rho(\mathbf{e}) = \frac{|\alpha-2|}{\alpha}\big[\big(\frac{\|\mathbf{e}\|^2/c^2}{|\alpha-2|} + 1\big)^{\alpha/2} - 1\big]$ the robust penalty of Barron (2019) ($\alpha = 0.5$, $c = 0.05$), applied in log space after predictions and targets are normalised by their mean distance to the origin,

$$\begin{aligned} \mathcal{L} = \sum_i \Big[& \rho\big(\mathbf{X}_i - \mathbf{X}_i^*\big) + 0.1 \left(\rho(\mathbf{P}_i - \mathbf{P}_i^*) + \rho(d_i - d_i^*) + \rho(\mathbf{q}_i - \mathbf{q}_i^*) + \rho(\mathbf{t}_i - \mathbf{t}_i^*)\right) \\ & + 0.3 \left(\mathcal{L}_{\mathrm{normal}} + \mathcal{L}_{\mathrm{grad}}\right)\Big] + 0.1\, \rho(s - s^*) + 0.1\, \mathcal{L}_{\mathrm{ray}} + 0.03\, \mathcal{L}_{\mathrm{mask}}, \end{aligned} \tag{5}$$

where dense terms are summed over valid pixels with the weights ω_i, $\mathbf{P}_i = d_i \mathbf{r}_i$ is the camera-frame pointmap, the pose terms are applied to the absolute poses and to all pairwise relative poses, $\mathcal{L}_{\mathrm{normal}}$ compares normals of the camera-frame pointmaps, $\mathcal{L}_{\mathrm{grad}}$ matches log-depth gradients, and $\mathcal{L}_{\mathrm{mask}}$ is a binary cross-entropy on the validity logit. The world-point term is confidence-weighted as in DUSt3R ($c_{\mathbf{p}}\rho - 0.2 \log c_{\mathbf{p}}$), and the 5% largest per-pixel residuals of the other dense point and depth terms are excluded.

Training and inference. Training starts from the public MapAnything checkpoint. Stage 1 fine-tunes the checkpoint on initial synthetic renders with the official pipeline for 35 epochs. Stage 2 turns on the full-field-of-view resizing, the aspect-ratio embedding and the online camera sampling of Section 3.2 (100 epochs). Stage 3 trains on second-generation scenes with the losses of Eqs. (3)–(5) and the panorama wrap, in two runs from Stage 2: one of 16 epochs and one of 859 epochs; the released weights interpolate the two (Appendix A). Batches hold tuples of 2–8 views at the ten aspect buckets with 48 images per GPU, on nodes of four or eight H200 GPUs; an epoch draws 40,000 tuples and takes 33 minutes on four H200s in Stage 2. For inference, the N images are resized to the bucket closest to their mean aspect ratio, a_i is read from their native shapes (set to $\log 2$ for detected full panoramas) and π_i from the detector, and one forward pass returns Eq. (1); no alignment, matching or optimisation follows.

4 Experiments

4.1 Protocols

Both real benchmarks start from real panoramic captures: the panoramas are the originals, and the perspective and fisheye views are resampled from them through the stated camera models, so every view carries exact rays and metric ground truth. Pose metrics are computed over ordered view pairs up to a 30° threshold: relative rotation and translation accuracy (RRA@30, RTA@30; sign-agnostic translation direction, Wang et al., 2023), their mean average accuracy (mAA@30) and area under the curve (AUC@30); ATE is the absolute trajectory error after one Sim(3) alignment per tuple (definitions in Appendix F). Every baseline runs with its official loader and checkpoint on the same tuples, ground truth and scorer; pointmap metrics and further protocol details are in Appendix F.

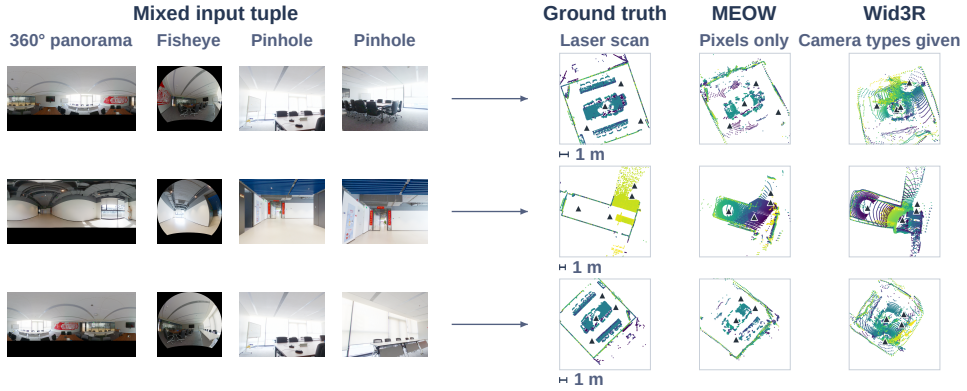

Figure 4: Laser-scanned mixed tuples, zero-shot. Each row shows the four inputs, the laser ground truth and the fused pointmaps of MEOW (pixels only) and Wid3R (camera types given), placed by the scorer's alignment. Top-down floor-to-wall slices coloured by height; triangles mark stations or predicted cameras.

4.2 Heterogeneous tuples

Table 2 reports the heterogeneous 2D3DS benchmark: areas 5a, 5b and 6 of Stanford 2D3DS (Armeni et al., 2017), 88 tuples of 3–24 views cycling through full panoramas, synthesised 90° perspective views and 180° equidistant fisheyes, all views in one forward pass. MEOW reaches 79.9 mAA@30; Wid3R, given the camera class of every view, reaches 53.8, and the difference lies in translation (RTA@30 96.0 against 86.3, ATE 0.62 against 0.83). The perspective models, which also receive pixels only, reach 10.4–19.5. Input resizing alone does not explain the gain: the public MapAnything weights score 16.2 with their own loader and 15.2 with our full-field-of-view resizing (Table 8, Appendix D).

4.3 A laser-scanned benchmark on which every method is zero-shot

We release a four-track benchmark built from our own registered BLK360 G2 laser scans of a multi-room office: 12 stations with survey-grade poses and pointmaps, tuples selected by exact covisibility computed from the scans, and regions the scanner cannot see masked out; no method has trained on it. Table 3 gives the results and Figure 4 shows three mixed tuples. On the four-view mixed track MEOW registers every tuple (RRA@30 and RTA@30 of 100) and reaches 79.4 AUC@30 against 19.0–32.9 for the open baselines, which collapse on the panorama track. Wid3R registers the rotations of the mixed tuples but not their translations (RTA@30 58.0, AUC@30 29.3), and its pointmaps are less complete.

4.4 Single-camera inputs and efficiency

On single-camera input the model behaves like its backbone on pinholes (85.7 against 86.8 AUC@30 on the laser pinhole track, Table 3) and, unlike any perspective model, registers panorama-only tuples: 74.8 AUC@30 on 2D3DS and 71.0 on the laser track against at most 7.9 (Table 9, Ap-

Table 2: Heterogeneous 2D3DS tuples (Armeni et al., 2017): 88 tuples of 3–24 views, one forward pass. Pose metrics are computed per tuple and averaged over the 88 tuples; ATE is after Sim(3) alignment. †No multi-view weights released.

Method	Camera input	RRA@30	RTA@30	mAA@30	ATE ↓
VGGT	None	32.5	47.2	10.4	1.65
π^3	None	45.5	59.4	19.5	1.24
MapAnything	None	52.4	53.5	16.2	1.48
CAM3R†	None	–	–	–	–
MEOW (ours)	None	95.0	**96.0**	**79.9**	**0.62**
Wid3R	Class per view	**96.6**	86.3	53.8	0.83

Table 3: Laser-scanned benchmark, 24 four-view tuples per track. AUC@30 is the mean over tuples; Acc and Comp are in metres after one alignment per tuple; NC is normal consistency.

		Mixed tuples (panorama + fisheye + two pinholes)						Single-camera tuples, AUC@30		
Method	Camera input	RRA@30	RTA@30	AUC@30	Acc ↓	Comp ↓	NC	Panorama	Pinhole	Fisheye
DUSt3R	None	62.5	67.4	32.9	0.164	0.863	0.758	0.5	96.0	73.8
MASt3R	None	66.7	66.0	31.8	0.182	0.508	0.737	2.7	96.2	56.2
VGGT	None	45.8	63.9	19.0	0.279	0.591	0.665	1.9	97.1	55.0
π^3	None	66.0	63.9	26.6	0.240	0.693	0.736	1.6	**97.9**	75.1
MapAnything	None	66.0	57.3	24.5	0.255	0.830	0.653	1.5	86.8	75.1
MEOW (ours)	None	**100**	**100**	**79.4**	**0.142**	**0.367**	**0.809**	71.0	85.7	73.2
Wid3R	Class per view	97.2	58.0	29.3	0.199	0.467	0.764	81.9	90.5	**87.3**
PanoVGGT	Panoramas	–	–	–	–	–	–	**90.0**	–	–

pendix D). Inference costs what MapAnything costs on the same views, 0.66 s for eight views on one RTX 5090 with image loading included, and 32 views fit in one backbone forward pass (Table 13, Appendix D).

4.5 Ablations

Table 4 changes one thing at a time. On the released weights, the public crop loader in place of the full-field-of-view resizing costs 3.4 mAA@30 on 2D3DS and 4.7 AUC@30 on the laser mixed track, and withholding the aspect-ratio embedding costs 23.1 and 11.3, with the largest drop on 16:9-squeezed panoramas, where the embedding carries the content aspect. Stage 2, where the embedding enters, was trained in two runs that differ only in it (Table 10, Appendix D): the run with the embedding leads by 3.2 mAA@30, 3.7 AUC@30 and 10.7 AUC@30 on 16:9 panoramas, trails by 1.7 on the laser panorama track, and is the one carried into Stage 3.

Table 4: Ablations. Top: the released weights with one input change at a time. Bottom: the two Stage-2 runs, identical except for the aspect-ratio embedding (Table 10). Columns: per-tuple mAA@30 on the 88 heterogeneous 2D3DS tuples; laser mixed-track AUC@30; AUC@30 on 2D3DS panorama tuples squeezed to 16:9.

	2D3DS mAA@30	Laser mixed AUC@30	16:9 panoramas AUC@30
Released weights (full model)	79.9	79.4	77.9
Crop loader instead of full-field-of-view resizing	76.5	74.7	–
Aspect-ratio embedding withheld	56.8	68.1	41.2
Stage-2 run with the embedding (carried into Stage 3)	63.6	65.4	63.3
Stage-2 run without the embedding	60.4	61.7	52.6

5 Conclusion

We presented MEOW, a feed-forward model that reconstructs metric pointmaps and camera poses from one tuple of perspective, fisheye and full-panorama images in a single forward pass, with no camera information supplied for any view. A perspective-pretrained backbone is adapted, with full-field-of-view resizing, on LENSCOPE tuples rendered through a continuous manifold of camera models with exact rays and depth and checked for covisibility. Adapted using synthetic tuples only, the model transfers zero-shot: 79.9 mAA@30 on heterogeneous 2D3DS tuples against 53.8 for Wid3R given the camera type of every view, and every tuple on the four-view mixed track of our laser-scanned benchmark registered at 79.4 AUC@30. The ablations show what it relies on: the full-field-of-view resizing and the aspect-ratio embedding at inference, and the embedding already at the stage where it enters training. The shared metric world also yields cross-camera correspondences without a matcher (Appendix E).

Limitations. The engine covers synthetic indoor scenes, and the released model is adapted on room-scale tuples of up to eight views; on centimetre-baseline perspective video (Replica (Straub et al., 2019), ADT (Pan et al., 2023)) it trails the perspective models (Table 12, Appendix D). The aligned metrics do not test the predicted metric scale, which is 12–25% short of the truth (Appendix F). Extending the training range to longer sequences and replaying real perspective video during adaptation are the next steps.

AI USE STATEMENT

The research ideas, the experimental design, the claims and the story of this paper are the authors'. Generative AI tools assisted with the manuscript and citation checks, LATEX editing, schematic artwork, and benchmarking code. The authors verified the code, re-derived every reported number from the raw logs, checked the citations, reviewed the final manuscript, and take responsibility for the final content. No generative model was used to produce training data.

REPRODUCIBILITY STATEMENT

All randomness in the engine, the training runs and the benchmark construction is seeded, and both scene generations regenerate from seeds with the scripts in the repository: a regeneration of three first-generation scenes reproduced the rays, depth and validity masks of the released packs exactly, with RGB differing only by renderer sampling noise (PSNR above 75 dB). The engine, the benchmark construction scripts with their convention tests, the evaluation protocols, the training configurations and the checkpoints will be released.

REFERENCES

Iro Armeni, Sasha Sax, Amir R. Zamir, and Silvio Savarese. Joint 2D-3D-semantic data for indoor scene understanding. *arXiv preprint arXiv:1702.01105*, 2017.

Jonathan T. Barron. A general and adaptive robust loss function. In *CVPR*, 2019.

Paul J. Besl and Neil D. McKay. A method for registration of 3-D shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):239–256, 1992.

Oleksandr Bogdan, Viktor Eckstein, François Rameau, and Jean-Charles Bazin. DeepCalib: A deep learning approach for automatic intrinsic calibration of wide field-of-view cameras. In *CVMP*, 2018.

Vukasin Bozic, Isidora Slavkovic, Dominik Narnhofer, Nando Metzger, Denis Rozumny, Konrad Schindler, and Nikolai Kalischek. Unified panoramic geometry estimation via multi-view foundation models. *arXiv preprint arXiv:2605.26368*, 2026.

Yohann Cabon, Lucas Stoffl, Leonid Antsfeld, Gabriela Csurka, Boris Chidlovskii, Jérôme Revaud, and Vincent Leroy. MUSt3R: Multi-view network for stereo 3D reconstruction. In *CVPR*, 2025. arXiv:2503.01661.

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. In *3DV*, 2017.

Boris Chidlovskii, Victor Domsa, Gabriela Csurka, and Levente Tamas. UCE: Universal camera embeddings for scalable 3D scene understanding. In *ECCV Workshops*, 2026.

T. A. Clarke, X. Wang, and J. G. Fryer. The principal point and CCD cameras. *The Photogrammetric Record*, 16(92):293–312, 1998. doi: 10.1111/0031-868X.00127.

Matt Deitke, Eli VanderBilt, Alvaro Herrasti, Luca Weihs, Jordi Salvador, Kiana Ehsani, Winson Han, Eric Kolve, Ali Farhadi, Aniruddha Kembhavi, and Roozbeh Mottaghi. ProcTHOR: Large-scale embodied AI using procedural generation. In *NeurIPS*, 2022.

Ruxiao Duan, Erin Hong, Dongxu Zhao, Eric Turner, Alex Wong, and Yunwen Zhou. Fisheye3R: Adapting unified 3D feed-forward foundation models to fisheye lenses. In *European Conference on Computer Vision (ECCV)*, 2026. arXiv:2603.28896.

Girish Chandar Ganesan, Yuliang Guo, Liu Ren, and Xiaoming Liu. UniDAC: Universal metric depth estimation for any camera. In *CVPR*, 2026. arXiv:2603.27105.

Suchisrit Gangopadhyay, Jung Hee Kim, Xien Chen, Patrick Rim, Hyoungseob Park, and Alex Wong. Extending foundational monocular depth estimators to fisheye cameras with calibration tokens. In *ICCV*, 2025. arXiv:2508.04928.

Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard Zemel, Wieland Brendel, Matthias Bethge, and Felix A. Wichmann. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2:665–673, 2020.

Christopher Geyer and Kostas Daniilidis. A unifying theory for central panoramic systems and practical implications. In *ECCV*, 2000.

Ryan Griffiths and Donald G. Dansereau. RoRE: Rotary ray embedding for generalised multi-modal scene understanding. In *ICLR*, 2026.

Yijing Guo, Mengjun Chao, Luo Wang, Tianyang Zhao, Haizhao Dai, Yingliang Zhang, Jingyi Yu, and Yujiao Shi. PanoVGGT: Feed-forward 3D reconstruction from panoramic imagery. In *CVPR*, 2026. arXiv:2603.17571.

Yuliang Guo, Sparsh Garg, S. Mahdi H. Miangoleh, Xinyu Huang, and Liu Ren. Depth any camera: Zero-shot metric depth estimation from any camera. In *CVPR*, 2025. arXiv:2501.02464.

Namitha Guruprasad, Abhay Yadav, Cheng Peng, and Rama Chellappa. CAM3R: Camera-agnostic model for 3D reconstruction. In *European Conference on Computer Vision (ECCV)*, 2026. arXiv:2603.22631.

Yannick Hold-Geoffroy, Dominique Piché-Meunier, Kalyan Sunkavalli, Jean-Charles Bazin, François Rameau, and Jean-François Lalonde. A perceptual measure for deep single image camera and lens calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(9): 10603–10614, 2023. doi: 10.1109/TPAMI.2023.3269641.

Tingjun Huang, Dmitry Rudshin, Mathieu Meyer, Pietro Bonazzi, Marc Pollefeys, and Emilia Szymańska. RIGOR: Rig-informed geometry for omnidirectional reconstruction. *arXiv preprint arXiv:2609.13504*, 2026.

Wonbong Jang, Philippe Weinzaepfel, Vincent Leroy, Lourdes Agapito, and Jérôme Revaud. Pow3R: Empowering unconstrained 3D reconstruction with camera and scene priors. In *CVPR*, 2025. arXiv:2503.17316.

Yuzhou Ji, Xiaotian Yang, and Zhipeng Zhang. CasaMaestro: Multi-view panoramas for house-scale 3D reconstruction. In *Computer Vision – ECCV 2026*, 2026. doi: 10.1007/978-3-032-37252-9_12.

Linyi Jin, Jianming Zhang, Yannick Hold-Geoffroy, Oliver Wang, Kevin Blackburn-Matzen, Matthew Sticha, and David F. Fouhey. Perspective fields for single image camera calibration. In *CVPR*, 2023.

Dongki Jung, Jaehoon Choi, Adil Qureshi, Somi Jeong, Dinesh Manocha, and Suyong Yeon. Wid3R: Wide field-of-view 3D reconstruction via camera model conditioning. In *European Conference on Computer Vision (ECCV)*, 2026. arXiv:2602.05321.

Juho Kannala and Sami S. Brandt. A generic camera model and calibration method for conventional, wide-angle, and fish-eye lenses. *IEEE TPAMI*, 28(8):1335–1340, 2006.

Nikhil Keetha, Norman Müller, Johannes Schönberger, Lorenzo Porzi, Yuchen Zhang, Tobias Fischer, Arno Knapitsch, Duncan Zauss, Ethan Weber, Nelson Antunes, Jonathon Luiten, Manuel Lopez-Antequera, Samuel Rota Bulò, Christian Richardt, Deva Ramanan, Sebastian Scherer, and Peter Kontschieder. MapAnything: Universal feed-forward metric 3D reconstruction. In *3DV*, 2026. arXiv:2509.13414.

Bogdan Khomutenko, Gaëtan Garcia, and Philippe Martinet. An enhanced unified camera model. *IEEE Robotics and Automation Letters*, 1(1):137–144, 2016.

Vincent Leroy, Yohann Cabon, and Jérôme Revaud. Grounding image matching in 3D with MASt3R. In *ECCV*, 2024.

Boying Li, Cheng Zhang, Weirong Chen, Guyuan Chen, Daniel Cremers, Jianfei Cai, Ian Reid, and Hamid Rezatofighi. CalibAnyView: Beyond single-view camera calibration in the wild. *arXiv preprint arXiv:2605.14615*, 2026a. URL https://arxiv.org/abs/2605.14615v2.

Haodong Li, Wangguangdong Zheng, Jing He, Yuhao Liu, Xin Lin, Xin Yang, Ying-Cong Chen, and Chunchao Guo. DA^2: Depth anything in any direction. In *International Conference on Learning Representations (ICLR)*, 2026b. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/786e39208b37bfcb0ad89413f155df99-Abstract-Conference.html.

Samuel Li, Pujith Kachana, Prajwal Chidananda, Saurabh Nair, Yasutaka Furukawa, and Matthew Brown. Rig3R: Rig-aware conditioning for learned 3D reconstruction. In *NeurIPS*, 2025. arXiv:2506.02265.

Xi Li, Linyuan Li, Yan Wu, Tong Rao, Kai Zhang, Xinchen Hui, and Cihui Pan. Argus: Metric panoramic 3D reconstruction for indoor scenes. In *Computer Vision – ECCV 2026*, 2026c. doi: 10.1007/978-3-032-37369-4_34.

Haotong Lin, Sili Chen, Jun Hao Liew, Donny Y. Chen, Zhenyu Li, Yang Zhao, Sida Peng, Hengkai Guo, Xiaowei Zhou, Guang Shi, Jiashi Feng, and Bingyi Kang. Depth anything 3: Recovering the visual space from any views. In *International Conference on Learning Representations (ICLR)*, 2026a. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/e4cd50120b6d7e8daff1749d6bbaa889-Abstract-Conference.html.

Xin Lin, Meixi Song, Dizhe Zhang, Wenxuan Lu, Haodong Li, Bo Du, Ming-Hsuan Yang, Truong Nguyen, and Lu Qi. Depth any panoramas: A foundation model for panoramic depth estimation. In *CVPR*, 2026b. arXiv:2512.16913.

Jiale Liu, Jungang Li, Jieming Yu, Xinglin Yu, Zihao Dongfang, Zongjian Ding, Kaifeng Ding, Yi Yang, Lidong Chen, Yang Zou, Shunwen Bai, Jiahuan Zhang, Haoran Huang, Shan Huang, Yudong Gao, and Mingjun Cheng. CM-EVS: Sparse panoramic RGB-D-pose data for complete scene coverage. *arXiv preprint arXiv:2605.15597*, 2026.

Yaroslava Lochman, Kostiantyn Liepieshov, Jianhui Chen, Michal Perdoch, Christopher Zach, and James Pritts. BabelCalib: A universal approach to calibrating central cameras. In *ICCV*, 2021.

Zeyu Ma, Alexander Raistrick, and Jia Deng. SimpleProc: Fully procedural synthetic data from simple rules for multi-view stereo. *arXiv preprint arXiv:2604.04925*, 2026.

Kanti V. Mardia and Peter E. Jupp. *Directional Statistics*. Wiley, 2000.

Christopher Mei and Patrick Rives. Single view point omnidirectional camera calibration from planar grids. In *ICRA*, 2007.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, et al. DINOv2: Learning robust visual features without supervision. *TMLR*, 2024.

Jing Ou, Zidong Cao, Yinrui Ren, Zhuoxiao Li, Jinjing Zhu, Tongyan Hua, Shuai Zhang, Hui Xiong, and Wufan Zhao. Holo360D: A large-scale real-world dataset with continuous trajectories for advancing panoramic 3D reconstruction and beyond. In *European Conference on Computer Vision (ECCV)*, 2026. URL https://arxiv.org/abs/2604.22482.

Xiaqing Pan, Nicholas Charron, Yongqian Yang, Scott Peters, Thomas Whelan, Chen Kong, Omkar Parkhi, Richard Newcombe, and Yuheng Carl Ren. Aria Digital Twin: A new benchmark dataset for egocentric 3D machine perception. In *ICCV*, 2023.

Photis Patonis. Methodology and tool development for mobile device cameras calibration and evaluation of the results. *Sensors*, 23, 2023.

Luigi Piccinelli, Christos Sakaridis, Mattia Segu, Yung-Hsu Yang, Siyuan Li, Wim Abbeloos, and Luc Van Gool. UniK3D: Universal camera monocular 3D estimation. In *CVPR*, 2025.

René Ranftl, Alexey Bochkovskiy, and Vladlen Koltun. Vision transformers for dense prediction. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021.

Enoc Sanz-Ablanedo, José Ramón Rodríguez-Pérez, Julia Armesto, and María Flor Álvarez Taboada. Geometric stability and lens decentering in compact digital cameras. *Sensors*, 10, 2010.

Daniil Sinitsyn, Nikita Araslanov, and Daniel Cremers. RayTun3R: Online camera adaptation in 3D foundation models. *arXiv preprint arXiv:2607.02711*, 2026.

Meixi Song, Dizhe Zhang, Hao Ren, Ruiyang Zhang, Bo Du, Ming-Hsuan Yang, and Lu Qi. UniSHARP: Universal sharp monocular view synthesis. *arXiv preprint arXiv:2606.07514*, 2026.

Julian Straub et al. The Replica dataset: A digital replica of indoor spaces. *arXiv preprint arXiv:1906.05797*, 2019.

Javier Tirado-Garín and Javier Civera. AnyCalib: On-manifold learning for model-agnostic single-view camera calibration. In *ICCV*, 2025. arXiv:2503.12701.

Shinji Umeyama. Least-squares estimation of transformation parameters between two point patterns. *IEEE TPAMI*, 13(4):376–380, 1991.

Vladyslav Usenko, Nikolaus Demmel, and Daniel Cremers. The double sphere camera model. In *3DV*, 2018.

Alexander Veicht, Paul-Edouard Sarlin, Philipp Lindenberger, and Marc Pollefeys. GeoCalib: Learning single-image calibration with geometric optimization. In *ECCV*, 2024.

Jianyuan Wang, Christian Rupprecht, and David Novotny. PoseDiffusion: Solving pose estimation via diffusion-aided bundle adjustment. In *ICCV*, 2023.

Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. VGGT: Visual geometry grounded transformer. In *CVPR*, 2025a.

Qianqian Wang, Yifei Zhang, Aleksander Holynski, Alexei A. Efros, and Angjoo Kanazawa. Continuous 3D perception model with persistent state. In *CVPR*, 2025b. arXiv:2501.12387.

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 *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025c.

Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jérôme Revaud. DUSt3R: Geometric 3D vision made easy. In *CVPR*, 2024.

Xintao Wang, Liangbin Xie, Chao Dong, and Ying Shan. Real-ESRGAN: Training real-world blind super-resolution with pure synthetic data. In *ICCV Workshops*, 2021.

Yifan Wang, Jianjun Zhou, Haoyi Zhu, Wenzheng Chang, Yang Zhou, Zizun Li, Junyi Chen, Jiangmiao Pang, Chunhua Shen, and Tong He. π^3: Permutation-equivariant visual geometry learning. In *ICLR*, 2026. arXiv:2507.13347.

Zhuang Xiong, Guohao Zhang, Chen Zhang, Zheyu Jiang, Yuchao Mei, Qingshan Xu, and Wenbing Tao. Look up and look back: Hidden attention and latent orientation in a frozen foundation model for panoramic SLAM. *arXiv preprint arXiv:2608.00925*, 2026.

Jianing Yang, Alexander Sax, Kevin J. Liang, Mikael Henaff, Hao Tang, Ang Cao, Joyce Chai, Franziska Meier, and Matt Feiszli. Fast3R: Towards 3D reconstruction of 1000+ images in one forward pass. In *CVPR*, 2025. arXiv:2501.13928.

Botao Ye, Marc Pollefeys, Ming-Hsuan Yang, and Abhijit Kundu. OmniPoint: Universal monocular metric pointcloud from any camera. In *Computer Vision – ECCV 2026*, 2026. doi: 10.1007/978-3-032-37602-2_32.

Shuo Zhang, Xin Su, Wei Wang, Jun Liu, Xinrui Zeng, Yongsen Chen, Chenjie Wang, Guibo Zhu, Jinqiao Wang, Bin Luo, and Liangpei Zhang. G-ray: Ray-level relative geometric position encoding in multi-view vision transformers under camera heterogeneity. *arXiv preprint arXiv:2609.15018*, 2026.

Zhiwei Zhang, Ruikai Xu, Weijian Zhang, Zhizhong Zhang, Xin Tan, Jingyu Gong, Yuan Xie, and Lizhuang Ma. PFDepth: Heterogeneous pinhole-fisheye joint depth estimation via distortion-aware gaussian-splatted volumetric fusion. In *ACM Multimedia*, 2025. arXiv:2509.26008.

Jia Zheng, Junfei Zhang, Jing Li, Rui Tang, Shenghua Gao, and Zihan Zhou. Structured3D: A large photo-realistic dataset for structured 3D modeling. In *ECCV*, 2020.

Heng Zhou, Shuhong Liu, Yonghao He, Bohao Zhang, Fa Fu, Chenhui Hou, Xianbao Hou, Lijun Han, Cong Yang, and Wei Sui. X-Lens: Real-time metric depth estimation with heterogeneous cameras. *arXiv preprint arXiv:2607.12993*, 2026.

A TRAINING RUNS AND EVALUATION CONVENTIONS

The released model descends from the public MapAnything checkpoint through domain adaptation on first-generation renders (35 epochs), Stage 2 with online camera sampling (100 epochs), and two Stage-3 runs on second-generation scenes: a 16-epoch run on an earlier 189-scene batch of the same generator (regenerable from its seeds), and a run on the 469-scene training split, trained in segments of at most 125 epochs, of which the checkpoint after 859 epochs is used; the released weights interpolate the two with weights 0.25 and 0.75; both parents are released with the interpolation script. Every stage uses AdamW with weight decay 0.05, a peak learning rate of 1.6×10^{-5} (8×10^{-7} for the encoder), 15 warm-up epochs and cosine decay to 1% of the peak, in bf16 autocast. Reported results use isolated checkpoint evaluation and explicit preprocessing and panorama-routing settings; scripts, logs and revision history are included with the release.

B LENSCOPE IN DETAIL

This appendix gives the parameters behind Section 3.2 and the complete procedure for one training tuple (Algorithm 1).

Renders. First-generation panoramas are rendered at 2160×1080; the 37,010 second-generation poses at 3072×1536 with 128 samples per pixel. In the first generation, narrow rectilinear views are re-rendered from the sharpest pinhole pack (18–40 pixels per degree) that covers the target and from the panorama (6 pixels per degree) otherwise; the second generation renders every view from its 8.5 pixel-per-degree panorama.

Camera models. Table 5 lists the seven models with their sampling weights and field-of-view ranges. The weights give a rectilinear majority with a deliberate share of fisheye and panoramic views.

Table 5: Camera models of the manifold with their sampling weights and field-of-view ranges. The field of view is drawn from a triangular density peaked at the common focal length of the model; full panoramas receive a random $SO(3)$ content rotation.

Camera model	Family	Sampling weight	Horizontal field of view
OpenCV	Rectilinear with distortion	0.40	48–120°
Pinhole	Rectilinear	0.15	48–95°
Fisheye624	Fisheye	0.13	120–210°
EUCM	Fisheye	0.07	90–180°
Mei	Fisheye	0.05	140–200°
Spherical crop	Spherical	0.05	110–300°
Full panorama	Spherical	0.15	360°

Photographic priors. Principal-point shifts: 90% Gaussian with $\sigma = 1.2\%$ of the image size truncated at $\pm 5\%$, of the order of the decentering reported for compact and phone cameras (Sanz-Ablanedo et al., 2010; Patonis, 2023), and 10% uniform to $\pm 22\%$ to cover zoom lenses and off-centre crops (Clarke et al., 1998; Jin et al., 2023). Roll: 90% a zero-centred two-component Cauchy mixture (scales $0.06°$ and $5.7°$, weights 1/3 and 2/3) following the roll model of Hold-Geoffroy et al. (2023), truncated at $\pm 30°$, 10% uniform; tilt: 90% Rayleigh with $\sigma = 4°$ truncated at $12°$, our choice for a near-level indoor prior, 10% uniform. Optics layer: modulation-transfer randomisation on half of the views (downsampling factor log-uniform in $[0.35, 1]$), lateral chromatic aberration, $\cos^4$ vignetting and sensor noise, a synthetic degradation pipeline in the manner of Wang et al. (2021); without it, sharpness would reveal the source asset and hence the field of view, a shortcut in the sense of Geirhos et al. (2020). The mutual information between field of view and sharpness, estimated on the first-generation source mix, falls from 0.557 to 0.251 bits.

Covisibility check and repair. The check renders the sampled views at 96 pixels on the long side with $2\times$ supersampling and recomputes c_{ij} with Eq. (2). A tuple whose graph is not connected at 0.25 is repaired by moving its disconnected views toward the native view in three steps (fractions

0.5, 0.8 and 1.0 of the way). A tuple that still fails is rejected and the scene redrawn once; after that the sample falls back to the unaugmented renders. Kept views are rendered once more at their target resolution with 2× supersampling.

Tuple-size schedule. The first epoch uses $K = 4$ only, and a 15-epoch cosine ramp opens the distribution to its target over $K = 2, \dots, 8$ (0.08, 0.12, 0.15, 0.18, 0.18, 0.15, 0.14). Each step holds $\lfloor 48/K \rfloor$ tuples per GPU, so every step sees up to 48 images.

Algorithm 1 One training tuple: from panorama renders to the network input

Require: scene with panorama renders $\{E_k\}$ (RGB, rays, depth, mask) and offline covisibility c_{kl}; tuple size K; bucket $(h, w) \in \mathcal{B}$
Ensure: input tensors $\{\tilde{I}_i, a_i, \pi_i\}_{i=1}^K$ and targets $\{\mathbf{r}_i, d_i, \mathbf{R}_i, \mathbf{t}_i, m_i\}_{i=1}^K$
$(k_1, \dots, k_K) \leftarrow$ ConnectedWalk$(c, K, \tau{=}0.40)$; relax to $\tau{=}0.20$ if no chain of length K exists ▷ distinct optical centres
draw the gaze mode: aimed at a common surface point with probability 0.55, native orientation otherwise
for $i = 1$ **to** K **do**
    draw the camera model μ_i, field of view θ_i, distortion, principal point, roll and tilt from Table 5 and the priors above
    $\mathbf{R}_i^c \leftarrow$ gaze rotation or native orientation; **if** μ_i is a full panorama **then** $\mathbf{R}_i^c \sim SO(3)$
    $V_i \leftarrow$ Resample$(E_{k_i}, \mu_i, \theta_i, \mathbf{R}_i^c, 96$ px$)$ ▷ RGB by ray lookup; rays, depth and mask analytically
end for
$c_{ij} \leftarrow$ Covis$(\{V_i\})$ by Eq. (2); $\mathcal{G} \leftarrow$ graph with edges $c_{ij} > 0.25$
for $f \in (0.5,\ 0.8,\ 1.0)$ **while** $\mathcal{G}$ is not connected **do**
    pull each view outside the largest component a fraction f toward its native orientation and field of view; re-render at 96 px; recompute $\mathcal{G}$
end for
if $\mathcal{G}$ is not connected **then return** REJECT ▷ redraw the scene once, then fall back to the native renders
end if
for $i = 1$ **to** K **do**
    $V_i \leftarrow$ Resample$(E_{k_i}, \mu_i, \theta_i, \mathbf{R}_i^c,$ target resolution, $2\times$ supersampling$)$; apply the optics and sensor layer to its RGB
    $(H_i, W_i) \leftarrow$ shape of V_i; $a_i \leftarrow \log(W_i/H_i)$; $\pi_i \leftarrow$ [azimuth span of the middle row $> 350°$]; **if** π_i **then** $a_i \leftarrow \log 2$
    $\tilde{I}_i \leftarrow$ RGB of V_i resized to (h, w); rays, depth and mask resampled through the same map (nearest for depth and mask)
end for
return $\{\tilde{I}_i, a_i, \pi_i\}_{i=1}^K$ and the targets

At inference only the last loop runs, on the real images, with the bucket chosen from the mean aspect ratio of the tuple and π_i from the detector of Appendix C.

C Panorama detector

The image is resized to 128 rows (width in proportion). With $\mathrm{MAE}(\cdot, \cdot)$ the mean absolute colour difference between two columns and $\bar{\sigma}$ the mean horizontal colour standard deviation of a band of rows,

$$\rho_{\text{seam}} = \frac{\mathrm{MAE}\big(I_{:,W},\, I_{:,1}\big)}{P_{90}\big\{\mathrm{MAE}(I_{:,k},\, I_{:,k+1})\big\}_{k<W}}, \qquad \rho_{\text{pole}} = \frac{\frac{1}{2}\big(\bar{\sigma}_{\text{top}} + \bar{\sigma}_{\text{bottom}}\big)}{\bar{\sigma}_{\text{middle}}}, \tag{6}$$

over the top and bottom $H/16$ rows and the middle $H/8$ rows. In a full panorama the last and first columns are adjacent in content, so ρ_{seam} is small, and the pole rows compress all longitudes into near-constant colour, so ρ_{pole} is small. The guard η is the fraction of rows in the central half whose 1%-wide left and right edge strips both hold mostly non-black pixels; it rejects circular fisheyes, whose black corners would satisfy both ratios. The flag is $\pi_i = [\eta \geq 0.5] \wedge [\rho_{\text{seam}} < 3] \wedge [\rho_{\text{pole}} < 0.55]$. Table 6 lists the error counts on the two headline sets; every other evaluation set is routed with the same detector.

Table 6: Full-panorama detector audit on the two headline sets with the released detector (two image cues plus the black-circle guard, thresholds fixed on training renders): panoramas missed and other views flagged as panoramas. Every other evaluation set is routed with the same detector; its per-set counts are part of the released audit.

Set	Views	Misses (panoramas not detected)	False alarms (other views flagged)
Heterogeneous 2D3DS tuples, with guard	503	0 / 189	2 / 314
Laser mixed track, with guard	96	0 / 24	0 / 72

Table 7: Training-data construction of the compared methods, from released code where available (X-Lens, a calibrated-rig depth method, omitted; test-time camera inputs are in Table 1). Of the compared methods, LENSCOPE draws each view's camera from a continuous manifold, checks covisibility after the cameras are drawn, and uses no real image after the public perspective pretraining.

	Wid3R	CAM3R	Fisheye3R	PanoVGGT	LENSCOPE (ours)
Training data	4 real + 5 synthetic sets	4 real sets + views derived from panoramas	3 real + 5 synthetic sets	2 real + 2 synthetic panorama sets	Synthetic only, two procedural generations
Camera models seen	Pinhole, Fisheye624, Mei, equirectangular panorama (ERP); EUCM, Fisheye624, OpenCV splatted from pinhole (25%)	Pinhole, fisheye (Aria; equidistant from panoramas), ERP	Pinhole; Kannala–Brandt fisheye distorted online from perspective frames; no 360°	ERP only	Continuous manifold: pinhole, OpenCV, Fisheye624, EUCM, Mei, spherical crops, full ERP
Camera identity within a tuple	One class per sample	Two-view pairs of mixed classes	Per-frame distortion, p=0.5	N/A	Drawn per view, 2–8 views, distinct optical centres
Covisibility control	Position prior, no overlap test	Offline baseline and angle thresholds	GT-depth overlap $\geq$ 25% before distortion	Room or trajectory grouping	Checked after the cameras are drawn (mutual projection, depth agreement); failures repaired or dropped

Table 8: Resizing and routing controls for the released checkpoint and the public MapAnything weights on the two mixed-camera benchmarks; 2D3DS pose metrics per tuple, averaged over tuples. The Stage-1 row is the checkpoint adapted on first-generation renders with the public loader and loss, before the full-field-of-view (full-FoV) resizing, the embedding and the camera manifold. The detector reproduces the dataset annotation on every tuple of the mixed laser track (189 of 189 panoramas found on 2D3DS, 2 of 314 other views flagged). On the single-camera laser tracks the annotation gives 71.0 / 86.6 / 73.2 AUC@30 (panorama / pinhole / fisheye) against the detector's 71.0 / 85.7 / 73.2; on the 16 panorama tuples inside the covisibility envelope of Appendix F the released model reaches 92.6.

		2D3DS tuples		Laser mixed track		
Method	Input resizing and panorama flag	mAA@30	ATE ↓	AUC@30	Acc ↓	Comp ↓
MEOW	Full-FoV resizing, detector flag	79.9	0.62	79.4	0.142	0.367
MEOW	Full-FoV resizing, annotation flag	79.8	0.62	79.4	0.142	0.367
MEOW	Public crop loader	76.5	0.75	74.7	0.168	0.356
Stage-1 checkpoint (renders only)	Public crop loader	68.1	0.99	69.0	0.120	0.322
MapAnything	Public crop loader	16.2	1.48	24.5	0.255	0.830
MapAnything	Full-FoV resizing, diagnostic	15.2	1.45	19.7	0.272	0.808

D ADDITIONAL TABLES AND FIGURES

Table 7 contrasts the training data of the compared methods with LENSCOPE, Table 8 lists the resizing and routing controls of the released model, Table 9 reports real panoramas, Table 10 the Stage-2 runs with and without the aspect-ratio embedding, Table 11 squeezed inputs of real panoramas, Table 12 the real perspective video sets, Table 13 the timing measurements, and Table 14 the 2D3DS results by tuple size. Figure 5 shows the camera models of LENSCOPE on one scene, Figure 6 the as-fed distributions of the training stream. Figure 7 shows the resizing on one tuple, and Figure 8 the laser-scanned benchmark's twelve stations with one of its tuples.

Table 9: Real panoramas. *2D3DS tuples*: same-room tuples of 2–13 panoramas, 20 cases, seven areas; Wid3R reaches RRA@30 94.3 and RTA@30 96.3, PanoVGGT 100 and 100. *Single panorama*: 40 panoramas of areas 5a and 5b, per-view Sim(3)-aligned Chamfer L1 in metres and median relative depth error. *Matterport3D (MP3D) tuples*: 18 scans, 8 panoramas per tuple, pointmap accuracy and completeness in metres. †PanoVGGT trains on 2D3DS and Matterport3D, so its 2D3DS numbers are in-domain; our model, Wid3R and the open baselines are zero-shot here. Wid3R's published 79.9 on 2D3DS uses a different room protocol and is not the number in this column.

		2D3DS tuples	2D3DS single panorama		MP3D tuples
Model	Checkpoint	AUC@30	Chamfer L1 ↓	Depth rel. ↓	Acc / Comp ↓
MEOW	Released	74.8	0.294	0.189	0.237 / 0.962
MEOW	90-epoch run	70.2	0.327	0.312	0.249 / **0.952**
Wid3R (camera type given) (Jung et al., 2026)	Released	92.6	0.090	0.044	–
PanoVGGT† (Guo et al., 2026)	Released	**99.9**	**0.063**	**0.028**	–
MapAnything (Keetha et al., 2026)	Public	6.7	0.853	0.518	**0.211** / 2.096
π^3 (Wang et al., 2026)	Public	7.9	–	–	–
VGGT (Wang et al., 2025a)	Public	5.1	–	–	–
DUSt3R (Wang et al., 2024)	Public	1.2	–	–	–
MASt3R (Leroy et al., 2024)	Public	2.1	–	–	–

Table 10: Stage-2 runs with and without the aspect-ratio embedding. Both start from the Stage-1 checkpoint and train the Stage-2 configuration (first-generation scenes, full-field-of-view resizing, online camera sampling, 100 epochs, seed 0); the only configuration difference is the embedding. Same protocol as the main tables; 2D3DS pose metrics per tuple.

	Heterogeneous 2D3DS tuples			Laser tracks, AUC@30	
Stage-2 run	RRA@30	RTA@30	mAA@30	Mixed	Panorama
With the embedding	91.7	93.3	63.6	65.4	45.0
Without the embedding	89.9	92.8	60.4	61.7	46.7

2D3DS panorama tuples, AUC@30: native 57.8 against 53.3; squeezed to 16:9, 63.3 against 52.6

Table 11: Squeezed input of real panoramas (2D3DS panorama tuples, AUC@30). The rows squeeze every panorama to the stated shape before it enters the pipeline. The panorama detector supplies the content aspect and panorama wrap in the released-model rows. At fixed 16:9 pixels, withholding the aspect-ratio embedding lowers the released model from 77.9 to 41.2. Dashes mark shapes that were not run for that checkpoint.

Input shape	Aspect input	90-epoch run	Earlier interpolation	Released
Native 2:1	Content aspect (2.0)	70.2	74.4	74.8
16:9 squeeze	Tensor aspect (1.78)	65.8	68.2	–
16:9 squeeze	Content aspect, from the detector (2.0)	75.2	78.2	77.9
16:9 squeeze	Embedding withheld	–	–	41.2
4:3 squeeze	Content aspect, from the detector (2.0)	71.1	–	75.6
1:1 squeeze	Content aspect, from the detector (2.0)	67.3	74.8	76.1

Table 12: Real perspective video with centimetre baselines, outside the room-scale tuples of the engine (AUC@30, same tuples, ground truth and scorer for every model). Rotation is solved by every model (RRA@30 of 99–100); the gap is translation direction under micro-baselines. Our row uses the released checkpoint. VGGT trains on Replica and ADT.

Set (tuples)	MEOW	MapAnything	VGGT	π^3	DUSt3R	MASt3R
Replica (48)	69.0	78.4	93.5	**98.5**	87.1	98.0
ADT pinhole (80)	39.4	60.8	53.7	80.4	49.3	**83.4**
ADT fisheye (80)	38.4	41.0	42.2	**46.1**	28.4	34.1

Table 13: Efficiency. Top: full pipelines per tuple on one RTX 5090, including image loading, first tuple excluded as warm-up. Bottom: the shared backbone alone at 518 px (random inputs pre-built on the GPU, 30 iterations after 8 warm-up iterations, CUDA-event timing). A six-point log-log fit gives exponents 1.09 (bf16) and 1.28 (fp32); the local slope rises with N, so this is an empirical scaling and not a complexity statement.

Per-tuple wall clock, I/O included (s)	MEOW	MapAnything	DUSt3R	MASt3R		
Replica, 8 views (RTX 5090)	0.66	0.66	5.54	8.36		
Laser mixed track, 4 views (RTX 5090)	0.48	–	3.65	4.83		
Backbone forward pass, N views (RTX 5090)	2	4	8	16	24	32
BF16 mean (ms)	67.5	126.9	266.8	583.0	965.8	1359.9
FP32 mean (ms)	129.2	267.7	610.9	1558.4	2867.0	4479.4
Peak memory BF16 (GB)	7.85	9.03	11.42	16.20	20.98	25.77

Table 14: Heterogeneous 2D3DS tuples by tuple size; the released model leads on 79 of the 88 tuples. Per-tuple mAA@30 and per-tuple accuracy at 15°, averaged over the tuples of each size range, for the released model and for Wid3R given the camera type of every view; the last column counts the tuples on which our per-tuple mAA@30 is higher. Training tuples have at most eight views.

		MEOW (pixels only)			Wid3R (camera type given)			
Views	Tuples	mAA@30	RRA@15	RTA@15	mAA@30	RRA@15	RTA@15	Ours ahead
3	36	**87.2**	**99.1**	**100.0**	53.5	91.7	64.8	35/36
4	13	**84.7**	**98.7**	**100.0**	56.2	87.2	72.4	13/13
5–8	28	**78.6**	**91.4**	**91.9**	52.6	85.9	66.2	25/28
9–14	7	**67.3**	**83.9**	**81.2**	56.7	83.3	70.1	6/7
15–24	4	29.1	46.3	49.3	**52.0**	**70.7**	**71.2**	0/4

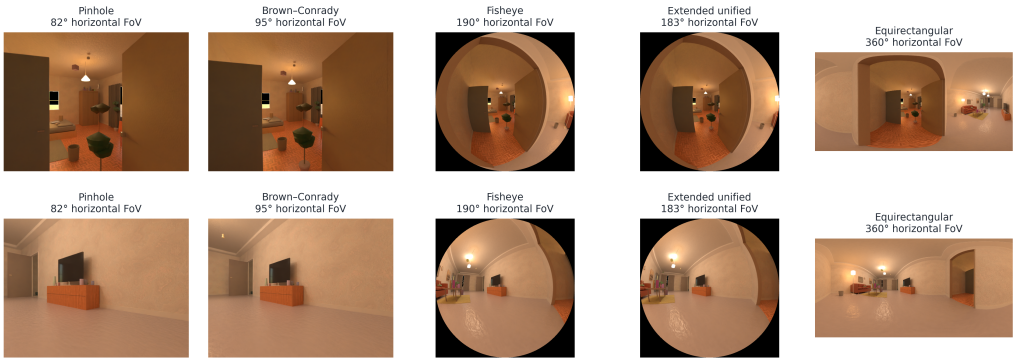

Figure 5: Same-scene camera-model gallery from LENSCOPE: two poses of one procedural scene (rows) rendered through five camera models (columns) at fixed illustrative fields of view; panel titles use the descriptive model names (Brown–Conrady for OpenCV, extended unified for EUCM). The gallery illustrates the camera models; training tuples are assembled separately under the covisibility check of Section 3.2.

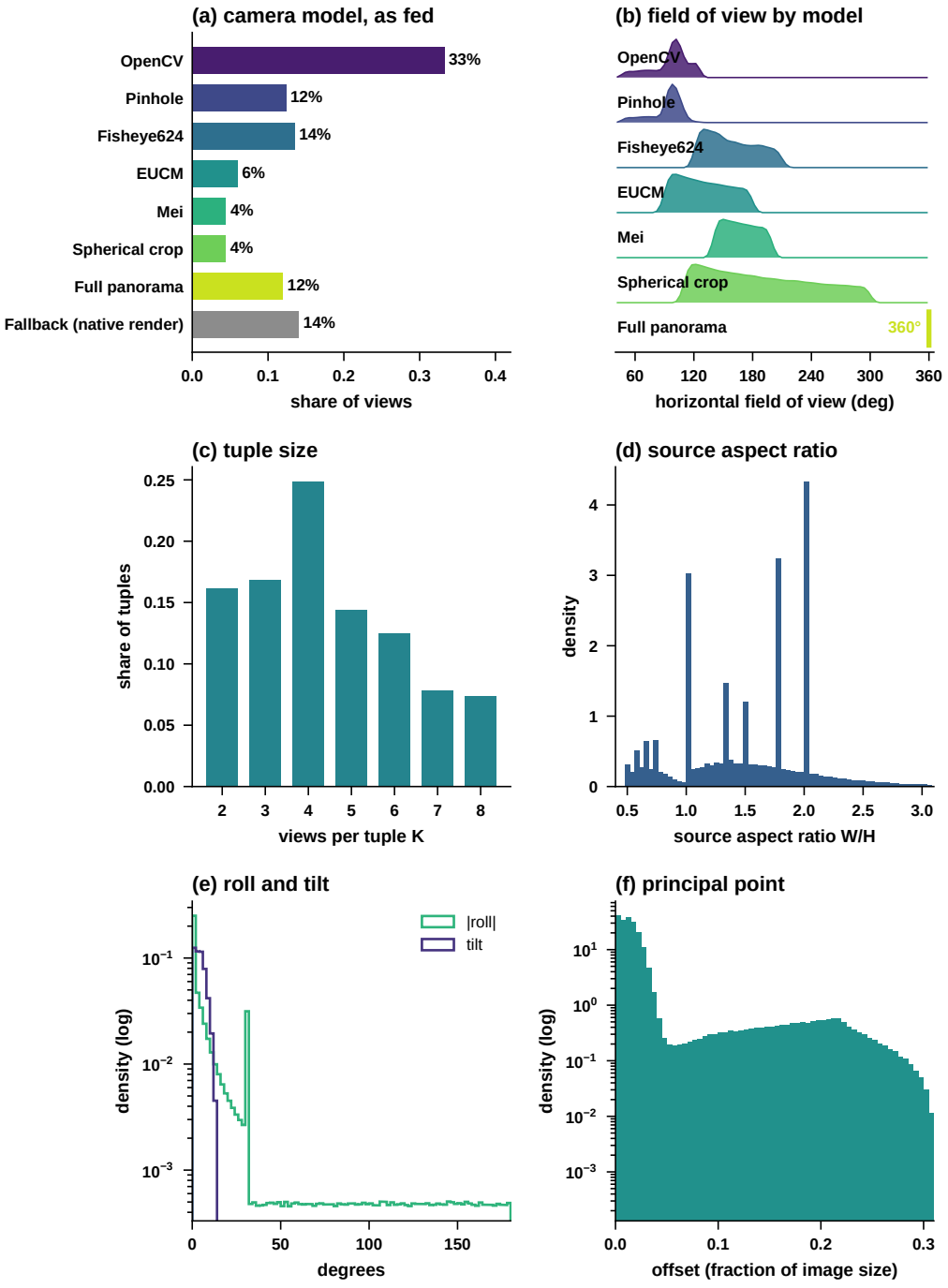

Figure 6: As-fed distributions of the training stream, read from the per-tuple sampling log of one data-parallel rank of Stage 2 (438,001 tuples): (a) camera-model share, (b) horizontal field of view, one density strip per camera model (full panoramas are 360°), (c) tuple size, (d) source aspect ratio, (e) roll magnitude and tilt, and (f) the principal-point offset magnitude; (e) and (f) use a log density so the 10% uniform tails are visible. The fallback bar in (a) counts views delivered as unaugmented renders (Appendix B).

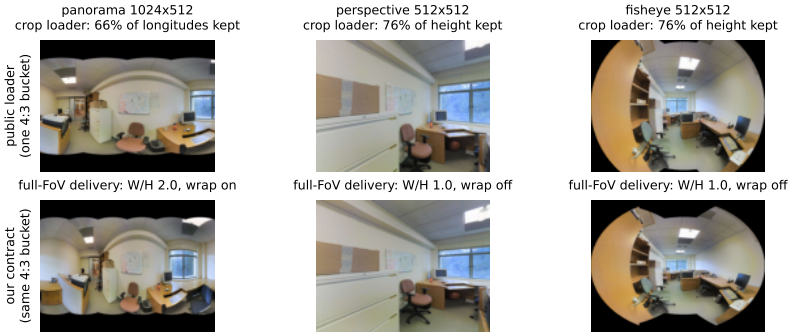

Figure 7: What the network receives for one heterogeneous tuple. The public loader picks one bucket from the mean aspect of the tuple, cover-scales and centre-crops every view, so the panorama loses a third of its longitudes and the square views lose a quarter of their height. Our resizing maps every view anisotropically to the same bucket, keeps all content, and passes the aspect ratio and the panorama flag.

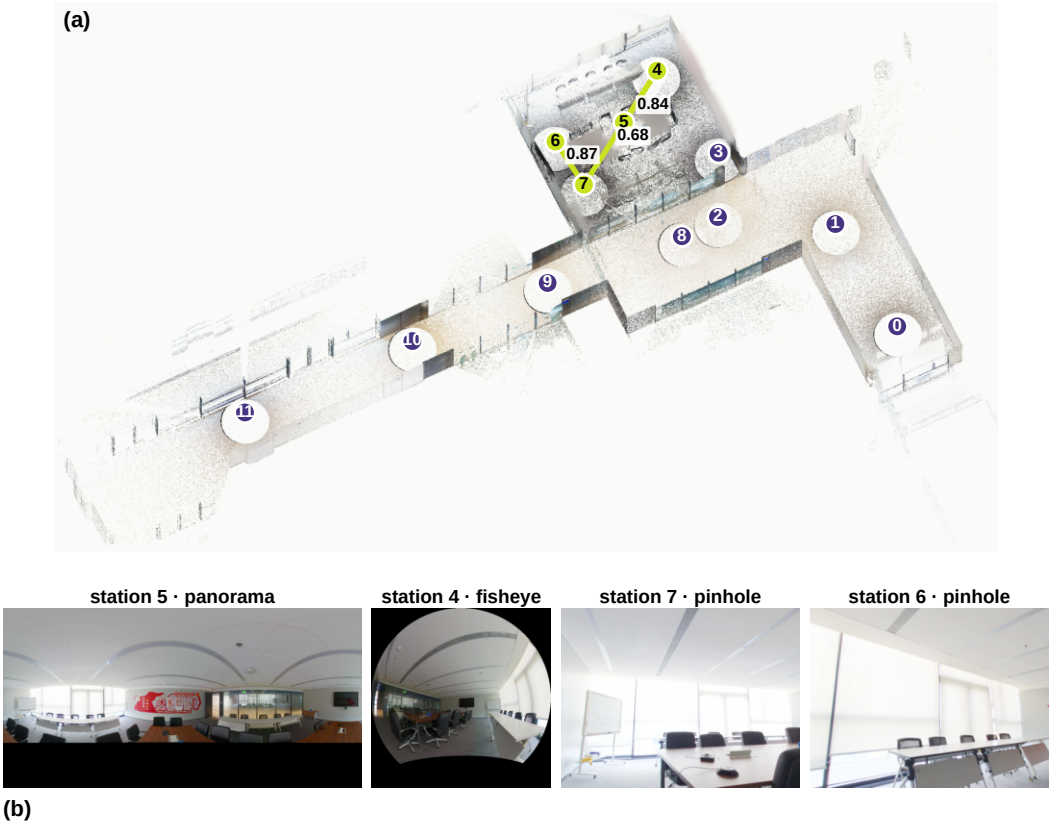

Figure 8: The laser-scanned benchmark. (a) The merged point cloud of the twelve registered scanner stations, numbered 0–11: a corridor chain and a meeting room. The highlighted stations and links are one mixed tuple of the benchmark (stations 5, 4, 7 and 6); the numbers on the links are the exact pairwise covisibility along the chain (0.84, 0.68 and 0.87). (b) The four views of that tuple as delivered to the models: a full panorama, a fisheye and two pinhole views resampled from the station scans.

E DOWNSTREAM PREVIEW: CROSS-CAMERA CORRESPONDENCES

A model that maps the same physical point to the same world coordinate regardless of the camera that observed it yields correspondences without a matcher: for a pixel of view A, its match in view B is the pixel whose predicted world point is the three-dimensional nearest neighbour of A's predicted point, and a match is kept when it is mutual. No matching head is trained and no descriptor is compared. Figure 9 shows such matches from the released model on real 2D3DS pairs across camera kinds and on pairs of a consumer panorama, a fisheye and a phone photograph of the same desk.



Figure 9: Cross-camera correspondences of the released model, obtained as mutual three-dimensional nearest neighbours between predicted world points. Rows one and two: real 2D3DS pairs with ground truth, panorama to fisheye and fisheye to perspective; yellow lines end within 3° of the true target, blue lines do not. Rows three and four: pairs without ground truth from a consumer panorama, a fisheye and a phone photograph of the same desk, coloured by the predicted three-dimensional agreement of the mutual match (brighter is closer).

F PROTOCOL DETAILS

Held-out engine scenes are scored per view by Sim(3)-aligned Chamfer L1 (Umeyama, 1991) and per tuple by fusing all views with their predicted poses, aligning the tuple to the ground truth once (Umeyama, then ICP (Besl & McKay, 1992)) and scoring pairwise relative poses. Pointmaps use accuracy, completeness and normal consistency after one alignment per tuple (Umeyama, the optimal point alignment of MoGe (Wang et al., 2025c), then ICP). Our model receives every view of a tuple at the common bucket by full-field-of-view resizing, with the aspect-ratio embedding and the image-detected panorama wrap. The open baselines (MapAnything, VGGT, π^3, DUSt3R, MASt3R) run with their official loaders and checkpoints on the same tuples, ground truth and scorer. Wid3R runs with its released weights and its official settings, which give the network the camera type of every view; its camera-frame convention was fixed once on a calibration split disjoint from the heterogeneous benchmark (2D3DS areas 1–4) and then frozen.

Pose metrics. Pose metrics are computed over ordered view pairs: RRA@30 and RTA@30 are the fractions of pairs whose relative rotation error and translation-direction error (sign-agnostic, Wang et al., 2023) are below 30°; mAA@30 averages, over thresholds from 1° to 30°, the fraction of pairs with both errors below the threshold, and AUC@30 integrates the same curve; ATE is the trajectory error after one Sim(3) alignment per tuple. On the 2D3DS benchmark the metrics are computed per tuple and averaged over tuples. All pose and pointmap metrics above are computed after a per-tuple Sim(3) alignment. The predicted metric scale is measured without alignment as the ratio of ground-truth to predicted camera-centre distances, median over the view pairs of a tuple and then over tuples: 1.14 on 2D3DS (74% of tuples within $\pm 20\%$, 91% within $\pm 30\%$) and 1.26–1.34 on the four laser tracks for the released model, against 0.63 and 0.70 for the public MapAnything weights on 2D3DS and the laser mixed track.

Laser-scanned benchmark. The scene is a multi-room office (a corridor chain and a meeting room) scanned from twelve stations with a Leica BLK360 G2 and registered in Cyclone; each station provides a 1536×3072 panorama with along-ray laser depth in the registered frame. The laser benchmark gates consecutive station pairs on the exact covisibility of the benchmark views themselves, symmetrised by the minimum of the two directions (depth agreement 10 cm plus 3%): 0.25 for the pinhole and fisheye tracks, 0.10 for the panorama track and 0.15 for mixed tuples; the covisibility envelope below uses the minimum of this statistic over the pairs of a tuple.

Covisibility envelope. Registration is governed by pairwise covisibility: tuples that fail to register have a median minimum pair covisibility of 0.033 and successful ones 0.424; below the training regime (about 0.05) no checkpoint in our study registers reliably, and the tuples of the mixed laser track lie inside this envelope.

Routing and controls. Headline numbers infer the binary full-panorama switch from the pixels; controls that instead use the dataset's panorama annotation are marked explicitly. Our self-run VGGT and π^3 rotation accuracies, pooled over pairs as in CAM3R's public evaluation code and paper, match the values it publishes for them (32.4 against 31.8 and 41.6 against 40.0).

Regenerating the data. The repository's `REPRODUCE.md` lists the commands that regenerate a first-generation scene (eight poses, five pinhole, two fisheye and one panoramic render at 64 samples per pixel) and a second-generation scene (panoramas at 3072×1536 and 128 samples per pixel), build shards and offline covisibility, draw one covisibility-checked tuple and run ten optimiser steps; the second-generation split (469/15/15, seed 20260729) is produced by `anycamgen/genesis/make_splits.py`, and the optional 935-file CC0 and public-domain asset bundle is inventoried with per-file provenance.