EMERGENT MULTI-VIEW GEOMETRY THROUGH SELF-DISTILLATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Over a century ago, Henri Poincaré argued that a motionless observer cannot acquire the notion of space. Yet, most visual representation learning methods operate on individual images, while those that leverage multiple views rely on RGB reconstruction, entangling geometry with appearance. We propose Poincar3, a self-supervised method that learns representations from multiple views through self-distillation instead of RGB reconstruction. We combine masked patch and image-level distillation with a teacher that observes additional views, enabling training from scratch without explicit 3D supervision. Poincar3 outperforms both previous single and multi-view self-supervised approaches such as DINOv3, MuM, and Muskie on correspondence estimation, camera pose estimation, and 3D reconstruction. Using a lightweight Poincaré adapter, we also find that our learned features encode camera motion more accurately than existing self-supervised representations.

1 INTRODUCTION

Representation learning lies at the heart of modern computer vision. Self-supervised learning (SSL) has produced increasingly powerful single-image representations without requiring human annotations (Caron et al., 2021; Zhou et al., 2022; He et al., 2022; Oquab et al., 2024; Siméoni et al., 2026; Yang et al., 2026). For 3D vision, however, as Poincaré argued (Poincaré, 1905), a single image is inherently limited. Recent SSL approaches therefore exploit multiple images of the same scene to encourage representations to capture information shared across views. CroCo (Weinzaepfel et al., 2022; 2023), MuM (Nordström et al., 2026), and Muskie (Li et al., 2025) extend masked image modeling (He et al., 2022) to the multi-view setting, where information from one image can be used to predict content in another.

However, RGB reconstruction imposes unnecessary constraints on learned representations. Reconstructing RGB values across views requires encoding appearance alongside geometry, whereas downstream 3D tasks benefit from representations robust to appearance variations. This motivates a natural question: *can we learn to transfer information across views without RGB reconstruction?*

We address this question through self-distillation, which has proven effective for learning single-image representations (Caron et al., 2021; Oquab et al., 2024; Siméoni et al., 2026). Self-distillation, however, is notoriously difficult to train, and introducing a multi-view transformer only exacerbates

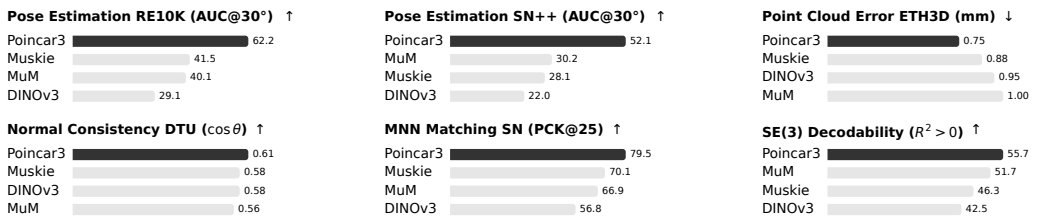


Figure 1: **Poincar3 main results.** Poincar3 outperforms state-of-the-art SSL baselines across multi-view geometric tasks, including pose estimation, point cloud estimation, and image matching.

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.



Figure 2: **Emerging matching capabilities without supervision.** Given query keypoints, we visualize the tracks formed by selecting the patch with the highest attention activation. Despite receiving neither correspondence labels nor explicit attention supervision, the model learns to identify patch correspondences across images, suggesting the emergence of 3D-aware representations from SSL.

this difficulty. Yet, we show that a multi-view teacher–student objective can be trained entirely from scratch, without any 3D supervision.

We find that stable and effective multi-view self-distillation relies on three key ingredients: introducing an image-level distillation objective, providing the teacher with access to additional views of the scene, and retaining full images rather than using local and global crops. Together, these ingredients give rise to Poincar3,[1] a multi-view self-supervised model that learns strong representations directly from unlabeled image sequences mined from the internet. We illustrate its state-of-the-art empirical performance in Fig. 1, and show that multi-view geometry emerges without labels, as evidenced by zero-shot correspondences in Fig. 2.

Across a broad range of 3D tasks, such as relative pose estimation, point cloud estimation, and correspondence estimation, Poincar3 substantially improves over state-of-the-art single- and multi-view SSL approaches, including DINOv3, MuM, and Muskie.

The main contributions are as follows:

(i) A new SSL objective based on multi-view self-distillation for learning visual representations from unlabeled image sequences, without RGB reconstruction or 3D supervision.

(ii) We show that the learned representations exhibit emergent multi-view geometric capabilities, including zero-shot multi-view correspondence estimation and lightweight $\mathrm{SE}(3)$ decodability via a Poincaré adapter, despite receiving no correspondence or 3D supervision.

(iii) An ablation study of the key components underlying our method.

2 Related work

Self-supervised Learning (SSL). SSL aims to learn powerful representations from unlabeled data. In vision, early works used hand-crafted pretext tasks (Doersch et al., 2015; Zhang et al., 2016; Gidaris et al., 2018). Later work focused on clustering (Caron et al., 2018; Asano et al., 2020; Caron et al., 2020; 2021) and contrastive learning (van den Oord et al., 2018; Misra & van der Maaten, 2020; Chen et al., 2020; He et al., 2020). A major advance was the exponential moving average (EMA) teacher–student framework (Tarvainen & Valpola, 2017) which has since become the dominant SSL paradigm (Grill et al., 2020; He et al., 2020; Caron et al., 2021; Zhou et al., 2022; Oquab et al., 2024; Darcet et al., 2025; Siméoni et al., 2026). The current state-of-the-art, DINOv3 (Siméoni et al., 2026), combines these advances with a masked image modeling objective based on iBOT (Zhou et al., 2022). Concurrently with iBOT, He et al. (2022) introduced another masked image modeling objective (MAE) that instead reconstructs RGB pixels. Although simple and effective for pretraining, MAE-style models produce frozen representations less suited for efficient linear extraction than DINO-style methods (Zhou et al., 2026).

[1] In reference to Poincaré's argument that understanding space requires motion. See also Chen et al. (2026).

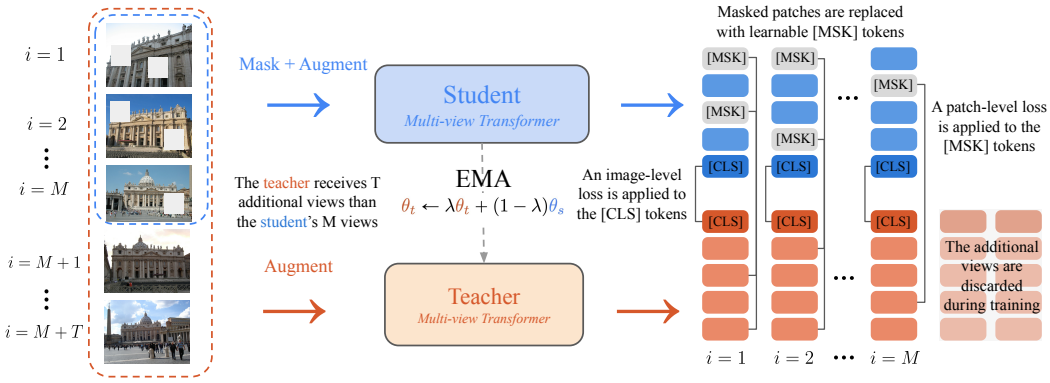


Figure 3: **Poincar3** learns visual features without labels via multi-view self-distillation. The teacher, an EMA of the student, processes a full image sequence, while the student processes a masked and photometrically augmented subset. After the multi-view transformer, student and teacher embeddings are aligned using an image-level loss on CLS tokens and a patch loss on masked tokens.

SSL in 3D Vision. To learn representations for 3D downstream tasks, CroCo (Weinzaepfel et al., 2022; 2023) extended the MAE objective by conditioning the reconstruction on an unmasked reference view of the same scene, while Alligat0R (Loiseau et al., 2025) and Gekko (Loiseau et al., 2026) focused on covisibility segmentation. MuM (Nordström et al., 2026) and Muskie (Li et al., 2025) concurrently generalized CroCo to reconstruct an arbitrary number of views from the same scene using a multi-view transformer. An et al. (2025) showed that these models can perform zero-shot correspondence estimation. In this paper, we show that self-distillation substantially improves it. Other approaches include adapting existing foundation models to 3D (Segre et al., 2026), using RGB-D guidance (Almukhamedov et al., 2026), novel view synthesis (Jin et al., 2025; Jiang et al., 2025; Mitchel et al., 2026; Zhao et al., 2026; Lucas et al., 2026), and video models (Kong et al., 2024; Assran et al., 2025; Mur-Labadia et al., 2026; Wang et al., 2026c). An important downstream application is *feed-forward reconstruction*, where a multi-view transformer directly predicts scene geometry from images. Recent models include (Wang et al., 2024; Leroy et al., 2024; Wang et al., 2025; 2026d; Keetha et al., 2026; Lin et al., 2026; Wang et al., 2026b). VGGT-Ω (Wang et al., 2026b), the strongest feed-forward reconstruction model to date, showed that post-training with an SSL objective enabled training on internet-scale videos, improving generalization. However, this approach first requires supervised pretraining with 3D annotations. In a similar spirit, SelfEvo (Huang et al., 2026) showed that VGGT can improve itself by using an EMA copy as a teacher, providing the teacher with additional views, and aligning teacher and student predictions. In contrast, we introduce a teacher–student objective that learns entirely from scratch without 3D annotations. It produces representations that outperform previous SSL baselines and are competitive with those obtained from supervised 3D training for zero-shot multi-view correspondence estimation.

3 Method

In this section, we begin by introducing the notation (Sec. 3.1) and subsequently propose multi-view self-distillation (Sec. 3.2). Thereafter, we outline the architecture (Sec. 3.3) and training (Sec. 3.4) of Poincar3. We illustrate Poincar3 in Fig. 3 and propose a pseudo-code implementation in Algo. 1.

3.1 Notations

Let $\mathcal{I} = \{\mathtt{I}_1, \mathtt{I}_2, \ldots, \mathtt{I}_M\}$ be a sequence of M images of the same scene. Our goal is to learn a set of dense patch features $\mathcal{Z} = \{\mathtt{Z}_1, \ldots, \mathtt{Z}_M\}$, where $\mathtt{Z}_i \in \mathbb{R}^{N \times C}$ contains a C-dimensional feature vector for each of the N patches in the ith image. These dense patch features should capture the underlying 3D structure of the scene, making them useful for downstream 3D vision tasks.

In practice, we parameterize our model by a backbone network f_θ that maps an image sequence to a set of dense patch features together with a global image representation $\mathtt{H}_i \in \mathbb{R}^C$ for each image: $(\mathcal{Z}, \mathcal{H}) = f_\theta(\mathcal{I})$, where $\mathcal{H} = \{\mathtt{H}_1, \ldots, \mathtt{H}_M\}$ is the global (`[CLS]`) representation for each image.

Algorithm 1: Multi-view self-distillation pseudo-code.

```
# fs, ft: student and teacher networks
# tps, tpt: student and teacher temperatures
# l: EMA momentum rate
# M, T: number of student and extra teacher views
ft.params = fs.params
for imgs in loader: # mini-batch of M+T frame seqs.
    sv = augment(imgs[:, :M])
    tv = augment(imgs)
    mask = sample_mask(sv) # patch mask [B, M, N]

    P_s, G_s, G_raw = fs(sv, mask=mask)
    with no_grad():
        P_t, G_t, _ = ft(tv, mask=None)

    # masked patch distillation (iBOT-style)
    P_t = sknopp(P_t[:, :M][mask].detach(), tpt)
    P_s = log_softmax(P_s[mask] / tps, dim=-1)
    L_patch = -(P_t * P_s).sum(-1).mean()

    # per-frame image objective (DINO-style)
    G_t = sknopp(G_t[:, :M].detach(), tpt)
    G_s = log_softmax(G_s / tps, dim=-1)
    L_global = -(G_t * G_s).sum(-1).mean()

    L_koleo = koleo(G_raw)

    loss = L_patch + 0.5*L_global + 0.1*L_koleo
    loss.backward()
    update(fs)  # AdamW
    ft.params = l*ft.params + (1-l)*fs.params
```

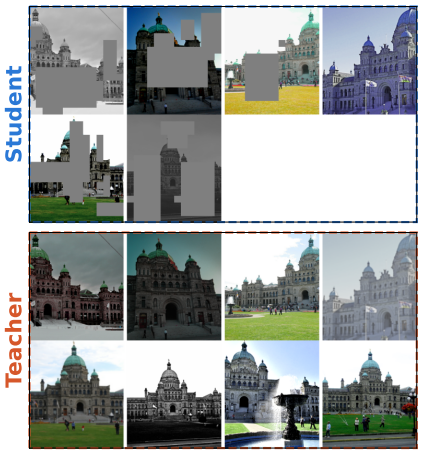


Figure 4: **Visualizing a training sequence.** The student and teacher get the same sequence (here $M = 6$) while the teacher gets additional frames (here $T = 2$), the student gets masked frames.

3.2 Multi-view Self-Distillation

Our goal is to obtain a multi-view self-supervised learning method that does not rely on RGB reconstruction. To do so, we adopt the teacher–student self-distillation framework. The student network is parameterized by θ_s and the teacher by θ_t. During training, the student learns to match the predictions of the teacher, while the teacher parameters are updated as an exponential moving average (EMA) of the student parameters $\theta_t \leftarrow \lambda\theta_t + (1-\lambda)\theta_s$, where $\lambda \in [0, 1)$ denotes the EMA decay.

The current state of the art for visual self-distillation is DINOv3 (Siméoni et al., 2026), built on the DINOv2 (Oquab et al., 2024) objective. The model operates on a single image by generating multiple augmented crops. The student processes masked global crops together with local crops, while the teacher observes only unmasked global crops. Both networks predict global and patch-level representations, and the student is trained to match the teacher using cross-entropy losses on the global representation and masked patches. Sinkhorn-Knopp (Sinkhorn & Knopp, 1967) and KoLeo (Sablayrolles et al., 2019) regularization are applied to prevent representational collapse.

Although highly effective for single-image representation learning, we show in Sec. 4.4 that naively extending this objective to multi-view image sequences leads to worse performance than RGB reconstruction. We therefore next consider the design of a multi-view self-distillation objective.

We operate on image sequences from the same scene where a multi-view transformer propagates information between frames. The backbone outputs are passed through a projection head (MLP + softmax) to transform the representation into a high-dimensional vector, as done in DINO, producing patch predictions $\mathtt{P}_i = g_p(\mathtt{Z}_i)$, and global predictions $\mathtt{G}_i = g_g(\mathtt{H}_i)$. For clarity, we denote the student outputs by $(\mathcal{P}^s, \mathcal{G}^s)$ and the teacher outputs by $(\mathcal{P}^t, \mathcal{G}^t)$.

Masked patch prediction. For every image $\mathtt{I}_i$, we randomly sample a binary patch mask $\mathbf{m}_i \in \{0, 1\}^N$, where $\mathbf{m}_i(u) = 1$ indicates that patch u of $\mathtt{I}_i$ is masked. We denote the corresponding set of masked patch indices by $\mathcal{M}_i$. The student receives the masked image $\tilde{\mathtt{I}}_i = (1 - \mathbf{m}_i) \odot \mathtt{I}_i$, where $\odot$ denotes masking at the patch level. The teacher always observes the full image.

Following iBOT (Zhou et al., 2022), the patch prediction loss is computed only over masked patches,

$$\mathcal{L}_{\text{patch}} = \frac{1}{\sum_{i=1}^{M} |\mathcal{M}_i|} \sum_{i=1}^{M} \sum_{u \in \mathcal{M}_i} \text{CE}\left(\mathcal{P}^t_{i,u}, \mathcal{P}^s_{i,u}\right), \tag{1}$$

where $|\mathcal{M}_i|$ denotes the number of masked patches, and $\text{CE}(\mathbf{p}, \mathbf{q}) = -\sum_k \mathbf{p}_k \log \mathbf{q}_k$ denotes the cross-entropy loss.

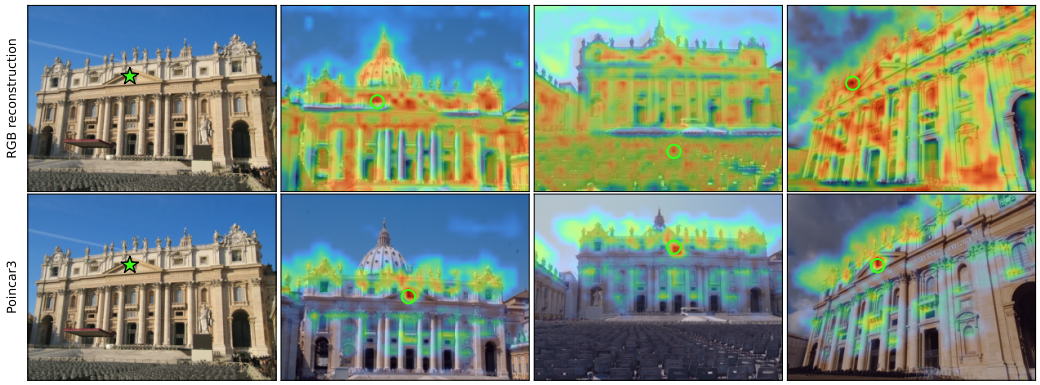


Figure 5: **Qualitative feature comparison.** Given a query patch (green star), we visualize its feature correlations across frames and mark the maximum response (green circle). RGB reconstruction produces diffuse responses, whereas our method yields localized responses at corresponding patches.

Image-level self-distillation. We additionally align the image-level representations using,

$$\mathcal{L}_{\text{global}} = \frac{1}{M} \sum_{i=1}^{M} \text{CE}\left(\mathcal{G}_i^t, \mathcal{G}_i^s\right), \tag{2}$$

which we find essential for guiding self-distillation towards a 3D-aware representation.

Teacher with additional views. We further propose that the teacher observe more of the scene than the student. If the student processes M views, the teacher also receives T extra frames from the same scene. These additional frames are excluded from all loss computations and only provide extra geometric context through the multi-view attention mechanism. During training, we sample T uniformly between 0 and 12. We illustrate an example sequence in Fig. 4.

Regularization and final objective. We apply Sinkhorn-Knopp and KoLeo regularization to prevent representational collapse. Our final training objective is,

$$\mathcal{L} = \mathcal{L}_{\text{patch}} + \alpha \mathcal{L}_{\text{global}} + \beta \mathcal{L}_{\text{KoLeo}}, \tag{3}$$

where $\alpha = 0.5$ and $\beta = 0.1$ in all experiments.

3.3 Network Architecture

We parameterize f_θ as a multi-view transformer inspired by VGGT-Ω (Wang et al., 2026b). The network consists of a ViT-L (Dosovitskiy et al., 2021) image encoder followed by a multi-view transformer decoder. The decoder propagates information across views using alternating frame-wise and global attention. Rotary positional embeddings (RoPE) (Su et al., 2023) are applied only within the frame-wise attention layers, while register attention (Wang et al., 2026b) is employed in a subset of the decoder blocks to enable efficient scaling to long sequences. Unless otherwise stated, the decoder comprises $L = 12$ alternating-attention layers with hidden dimension $C = 1024$. In total, our model comprises around 650M trainable parameters.

3.4 Training

Implementation Details. We pre-train Poincar3 for 400k steps using the AdamW (Loshchilov & Hutter, 2019) optimizer. Following VGGT and MuM, the input sequence length is sampled uniformly between 2 and 24 views while images are resized to 256×256. Given the sampled sequence length, we include as many training scenes as possible without exceeding a budget of 64 images per GPU. We use constant optimization hyper-parameters with learning rate 2×10^{-4}, weight decay 0.04, and EMA decay $\lambda = 0.999$. Training on 8 H200 GPUs takes three days. Further details on hyper-parameters are provided in Sec. D.1 in the Appendix. Code and weights will be released.

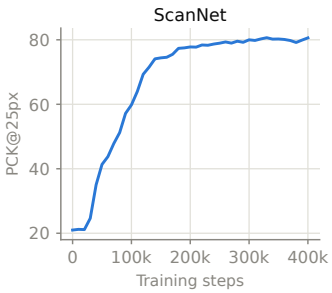


Figure 6: **Training dynamics.** Multi-view correspondence estimation attention performance.

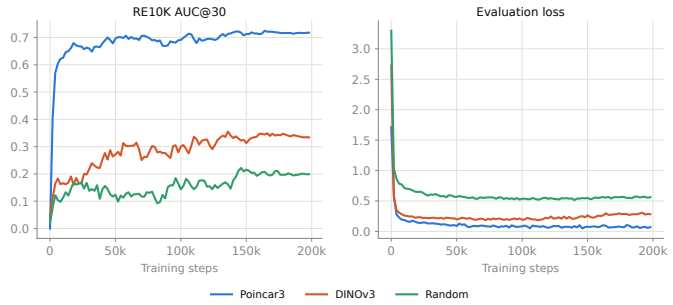


Figure 7: **Initialization for feed-forward reconstruction.** Comparing training from scratch (green), initializing the encoder from DINOv3 (red), and finetuning Poincar3 (blue).

Downstream performance improves steadily throughout training (see Figs. 6 and 15), while our features become increasingly consistent across views compared to RGB reconstruction (see Fig. 5). Interestingly, unlike DINOv3, we do not experience dense feature degradation during long training runs, which has been a topic of discussion lately (Dai et al., 2025; Siméoni et al., 2026).

Training Data. Our method is fully self-supervised and requires only unlabeled image sequences. We train on large-scale collections of internet videos such as SpatialVID (Wang et al., 2026a) and RealEstate10K (Zhou et al., 2018) and popular 3D annotated datasets such as MegaDepth (Li & Snavely, 2018) and ScanNet++ (Yeshwanth et al., 2023). For the former, sequences are constructed via temporal random sampling, whereas for the latter, frames are randomly sampled from the same scene to form a sequence. In Sec. 4.4, we show the strength of being able to train on data lacking 3D annotations. The full data distribution is provided in Tab. 7 in the Appendix.

4 Results

In this section, we compare Poincar3's visual features to those of state-of-the-art SSL models on a wide range of 3D vision tasks. We begin by evaluating feed-forward reconstruction (Sec. 4.1), both through finetuning and using frozen features. Thereafter, we consider the ability to extract accurate correspondences from multiple views zero-shot (Sec. 4.2) and decoding camera motion through a lightweight adapter (Sec. 4.3). Finally, we conduct extensive ablations that distinguish the contributions of each part of our design (Sec. 4.4). In the Appendix, we discuss our choice of baselines in Sec. B, additional experiments in Sec. C, and details on the evaluation in Sec. E.

4.1 Feed-forward 3D reconstruction

We begin by considering feed-forward reconstruction (Wang et al., 2024; 2025), where a multi-view transformer directly predicts scene geometry from a sequence of images. We consider three evaluation settings of increasing computational complexity (i) train only the pose and depth head, (ii) train a transformer, and (iii) finetune the full model with new heads. We train on a large collection of labeled 3D datasets using 4 H200 GPUs for each experiment and report the results in Tab. 1 together with training curves in Fig. 7. Poincar3 shows substantially improved performance, especially on camera pose estimation, compared to all existing models. Furthermore, initializing from the Poincar3 weights allows for rapid finetuning, achieving an AUC@30° of 65.0%+ on RE10K after only 10K steps with a low learning rate. Further details can be found in Sec. E in the Appendix.

4.2 Multi-view Correspondence Estimation

We study the 3D geometric consistency of features by zero-shot multi-view correspondence estimation. Correspondence estimation is at the heart of multiple-view geometry (Hartley & Zisserman, 2004) and has been shown to be closely linked to understanding 3D (El Banani et al., 2024; Starý et al., 2025; Chen et al., 2026). Concretely, we follow the evaluation protocol of El Banani et al.

Table 1: **Feed-forward reconstruction.** Reporting relative pose accuracy by AUC over 10 random frames and point cloud accuracy by median accuracy (acc.) in mm and normal consistency (NC) by the cosine of the angle between the normals. Training with 4 H200 GPUs for 3 days each.

	Multi-View Relative Pose						Point Cloud Estimation			
Method	RE10K		ScanNet++		MegaDepth		ETH3D		DTU	
Metric →	@3°	@30°	@3°	@30°	@3°	@30°	Acc.	NC	Acc.	NC
Train only heads on top of the frozen backbone										
DINOv3	0.0	18.9	0.0	9.4	1.5	55.5	1.18	0.59	10.76	0.54
Muskie	1.1	36.3	0.0	19.0	0.1	46.9	1.12	0.60	11.43	0.55
MuM	0.0	29.8	0.0	18.7	0.1	48.5	1.12	0.60	12.60	0.54
Poincar3	**2.4**	**51.9**	**0.1**	**48.7**	**3.7**	**67.5**	**0.82**	**0.65**	**10.60**	**0.59**
Train a transformer on top of the frozen backbone										
DINOv3	1.0	29.1	0.0	22.0	1.7	65.6	0.95	0.66	10.63	0.58
Muskie	1.8	41.5	0.0	28.1	1.7	58.9	0.88	0.67	11.95	0.58
MuM	1.4	40.1	0.0	30.2	1.9	60.8	1.00	0.60	12.26	0.56
Poincar3	**5.2**	**62.2**	**1.3**	**52.1**	**5.2**	**69.5**	**0.75**	**0.68**	**10.52**	**0.61**
Finetune the backbone[1]										
Random init.	0.0	17.4	0.0	11.3	0.0	35.9	1.24	0.60	12.34	0.56
DINOv3 init.	1.3	31.2	0.0	30.9	6.9	73.8	0.98	0.68	10.87	0.59
Poincar3	**8.3**	**68.2**	**4.5**	**67.0**	**7.4**	**74.6**	**0.81**	**0.80**	**6.73**	**0.63**

[1] MuM and Muskie are excluded because their architectures differ substantially, making direct finetuning comparisons less meaningful.

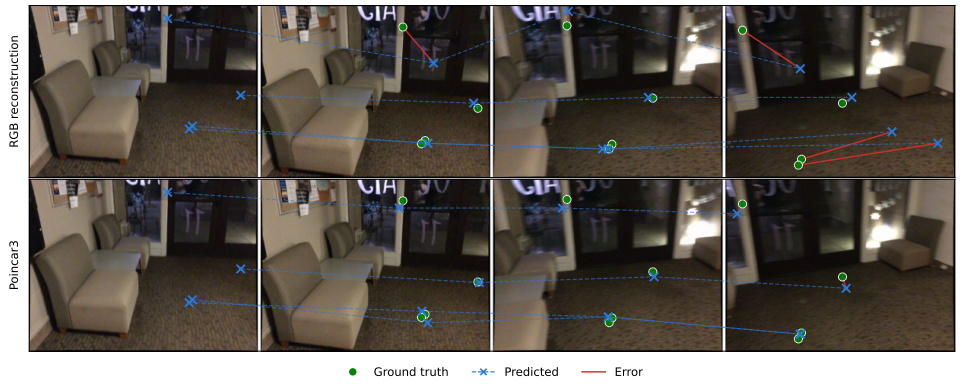


Figure 8: **Multi-view correspondence tracks.** We visualize the predicted tracks (blue), the ground-truth (green), and the error (red). Poincar3 gives accurate zero-shot multi-view geometry.

(2024) and sample sequences of 8 images where a set of patches are queried and tracks are produced by either nearest-neighbor matching in feature space or by using the maximum patch activation of the attention map. As shown in Fig. 14 (Appendix), the performance of baselines can vary substantially per layer, although Poincar3 consistently outperforms at each depth. For fair comparison, we sweep the layers and report the best performing one in Tab. 2. Aside from the quantitative evaluation, in Fig. 8 we qualitatively compare the predicted tracks with the ground-truth. We find that Poincar3 is substantially more accurate than other SSL methods and feed-forward reconstruction models trained with 3D annotations. The attention map is an even more powerful correspondence estimator, achieving a top-accuracy of 94.9. We also outperform all other methods on two-view correspondence estimation using a linear probe (Tab. 5, Appendix).

4.3 Emergent SE(3) Structure

Next, we follow Chen et al. (2026)[2] and evaluate whether the features capture the *geometry* of $\mathrm{SE}(3)$. Consider a static scene observed at poses $P_t \in \mathrm{SE}(3)$, and let $\mathcal{H}_t = f_\theta(\mathcal{I}_t) \in \mathbb{R}^C$ denote a model's

[2] Reimplemented based on the paper. No code is publicly available as of writing.

Table 2: **Multi-view correspondence estimation.** Zero-shot patch tracking across 8 views. While Poincar3 has strong representations at all layers, we compare to the **best** layer for fair comparison.

Method	ScanNet (Dai et al., 2017)				NAVI (Jampani et al., 2023)			
PCK@ →	5px	10px	25px	50px	5px	10px	25px	50px
Feature Nearest-Neighbor Matching								
Feed-forward Reconstruction								
VGGT-Ω	24.5	32.8	52.9	69.0	16.8	25.9	51.6	70.9
DA3	21.5	22.9	28.2	35.2	14.6	18.6	30.5	46.9
π^3	26.7	38.6	61.3	75.6	18.2	30.3	60.2	78.8
Self-supervised Models								
DINOv3	24.7	33.9	56.8	72.2	17.6	28.7	59.6	78.7
Muskie	27.7	42.4	70.1	83.1	18.7	32.7	64.7	80.5
MuM	27.0	40.5	66.9	79.9	17.9	30.2	60.2	74.9
Poincar3	**27.9**	**46.2**	**79.5**	**89.6**	**19.2**	**33.7**	**68.7**	**84.2**
Attention Matching								
Feed-forward Reconstruction								
VGGT-Ω	24.3	32.9	54.2	68.9	16.3	23.9	46.8	63.7
DA3	21.4	23.2	28.2	34.0	14.8	19.1	31.6	46.7
π^3	**27.0**	40.7	68.7	83.8	18.3	30.4	62.2	82.0
Self-supervised Models								
Muskie	24.6	34.7	62.9	81.5	15.5	22.3	46.2	67.9
MuM	25.6	37.2	64.7	77.2	17.4	28.5	56.6	68.7
Poincar3	**27.0**	**43.9**	**83.7**	**94.9**	**19.3**	**34.7**	**74.5**	**86.5**

Table 3: **Poincaré adapter.** We evaluate the frozen features using a lightweight camera motion adapter on 20 held-out scenes from ScanNet++. Inclusion of multi-view information is denoted by ✓.

Method	MV	Avg. R^2	$R^2 > 0$	$R^2 > 0.3$
DINOv3	✗	0.046±0.003	42.5±1.9	0.9±0.6
Muskie	✗	0.042±0.003	39.7±2.2	0.4±0.2
Muskie	✓	0.061±0.003	46.3±1.7	3.6±1.0
MuM	✗	0.059±0.002	43.1±1.8	4.5±0.6
MuM	✓	0.081±0.002	51.7±1.8	6.5±0.8
Poincar3	✗	0.053±0.003	42.4±2.3	2.0±0.7
Poincar3	✓	**0.098**±0.002	**55.7**±1.7	**7.1**±0.7

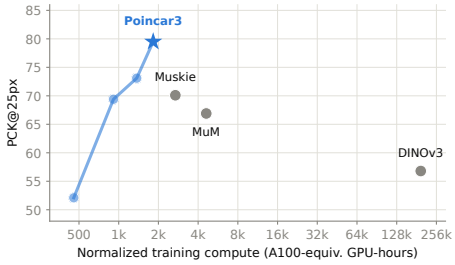


Figure 9: **Multi-view correspondence estimation accuracy versus training compute.** Poincar3 ties DINOv3's performance after just one day on 8×H200s.

feature for frame $\mathcal{I}_t$. The motion between frames t and $t+s$ is the $\mathfrak{se}(3)$ twist

$$\Delta P_{t,s} = \big(\mathrm{Log}(P_t^{-1} P_{t+s})\big)^{\vee} \in \mathbb{R}^6, \tag{4}$$

stacking three rotational and three translational velocities. We study if ΔP is a function of the corresponding feature displacement by fitting a trainable adapter φ_ϕ (an MLP) in a Siamese manner:

$$\Delta P_{t,s} \;\approx\; W\big(\varphi_\phi(\mathcal{H}_{t+s}) - \varphi_\phi(\mathcal{H}_t)\big) \tag{5}$$

The network φ_ϕ is called a *Poincaré adapter* as it aims to unroll the non-linear feature space to a homogeneous coordinate system, making changes in pose linear, revealing the $\mathrm{SE}(3)$ geometry.

We fit one such adapter per scene on 20 held-out ScanNet++ (Yeshwanth et al., 2023) scenes and report R^2 on the held-out pairs from each scene in Tab. 3. We qualitatively illustrate the results in Fig. 11. In the spirit of Poincaré's observation that spatial structure emerges through distinguishing changes of position from changes of state (Poincaré, 1905), the motion-informed features are consistently better than those obtained from one view, with Poincar3 achieving the strongest results.

Table 4: **Supervision ablation.** Multi-view correspondence estimation accuracy (PCK@25).

Method	Multi-view Matching	
Dataset →	ScanNet	NAVI
Same compute, data, and architecture		
I: RGB reconstruction (MuM obj.)	54.5	46.3
II: Baseline (DINOv2 obj.)	47.2	35.7
III: Multi-view DINOv2	49.7	35.9
IV: Multi-view iBOT	55.7	46.4
V: +Image-level objective	66.7	58.1
VI: +Teacher seeing more views	70.2	61.5
VII: +Increase scale (Poincar3)	**83.7**	**74.5**

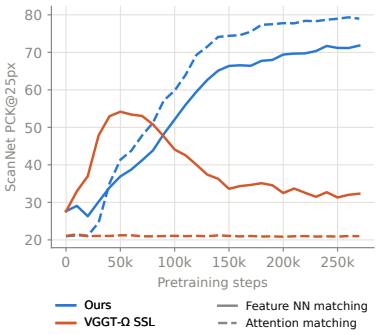


Figure 10: **VGGT-Ω SSL** comp.

4.4 Ablation Studies

Supervision. In Tab. 4, we compare different supervision objectives. For fair comparison, we keep the architecture fixed throughout and fix the compute budget to $4\times$H200 for a day. We establish two baselines: I) multi-view RGB reconstruction and II) single-view DINOv2. Building on the DINOv2 objective, we extend it to multiple views (III), remove the local crops (IV), add a global objective (V), feed the teacher additional views (VI), and scale the compute and model size (VII). Our objective (V) is clearly superior to the RGB reconstruction used in MuM and Muskie (I). Finally, in Fig. 10 we find the MSE loss from VGGT-Ω (SSL) does not work from scratch as self-supervision.

Data. Our setup enables us to leverage internet-scale data inaccessible to 3D reconstruction models requiring 3D labels. To quantify its benefit, we compare multi-view correspondence accuracy from attention maps when training on only 3D-labeled data versus all available data. Incorporating the additional data substantially improves multi-view geometry understanding, increasing PCK@50 from 87.5 to 94.9 on ScanNet and from 78.5 to 86.5 on NAVI.

5 Limitations

Our approach prioritizes geometric over semantic performance (see Tab. 6, Appendix). This is consistent with evidence that semantic and spatial information are processed by different neural systems in the human brain (Binder et al., 2009; Epstein et al., 2017). Furthermore, we do not have the computational resources to match the size of DINOv3 (we use around 100x less compute). Yet, we significantly outperform previous approaches (see Fig. 9).

We find that the representations degrade when hyper-parameters such as the EMA coefficient and weight decay deviate from those of DINOv3. We also find it crucial to follow prior work when adapting the learning rate to batch size (Goyal et al., 2018). This fragility has plagued self-distillation since its inception (Grill et al., 2020; Caron et al., 2021) and mitigating it constitutes interesting future work. Finally, while our training is self-supervised, frame selection within video sequences is currently handcrafted. Automating this selection is an interesting direction for future work.

6 Conclusions

We introduced Poincar3, a multi-view foundation model trained from scratch in a self-supervised manner. To remove the RGB reconstruction supervision used in MuM and Muskie, which entangles geometry with appearance, we proposed a DINO-like self-distillation objective. Self-distillation is notoriously challenging to train, and introducing a multi-view transformer further exacerbates this difficulty. We show that a multi-view teacher–student objective can nevertheless be trained from scratch, without supervision, by feeding the teacher additional frames, using a global stabilizing objective, and retaining full images rather than relying on local and global crops. Poincar3 outperforms MuM and Muskie on 3D vision tasks, and outperforms DINOv3 while using $100\times$ less compute.

AI USE STATEMENT

In this work, we did not use generative AI tools for any of the tasks with required disclosure. In particular, we have not used generative AI tools for helping develop theoretical models or conceptual frameworks, designing or providing feedback on research methodology or experiments, implementing methods, assisting with translation, cleaning and reformatting dataset, supporting qualitative and thematic data analysis, interpreting results, and formulating mathematical claims, providing critical ingredients for proving mathematical claims, assisting in the writing of proofs, proposing or refining hypotheses, are not applicable to this work. Additionally, we used generative AI tools for creating or modifying scientific figures or images, suggesting experimental parameters, creating or editing software code, and editing a research paper to improve readability. We have reviewed all AI-assisted work. We edited all LLM-generated writing suggestions and used it only as a source of inspiration for wording. The code is mainly human-written and verified by using external code-bases as evaluation of our methodology. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

Extensive efforts have been made to ensure reproducibility. Most importantly, the code and model weights will be released publicly. For training, there is extensive documentation of the data (Tab. 7), hyper-parameters (Sec. D.1), algorithm (Algo. 1), and compute (Fig. 9). We show the progression of training (Fig. 16) and evaluation (Fig. 6) metrics. The evaluation protocols follow prior work (such as El Banani et al. (2024) and Wang et al. (2025)) and are described in detail in Sec. E.

REFERENCES

Farkhat Almukhamedov, Sami Azirar, and Hermann Blum. Dinocular: Self-supervised visuospatial representations, 2026. URL https://arxiv.org/abs/2608.27226.

Honggyu An, Jin Hyeon Kim, Seonghoon Park, Jaewoo Jung, Jisang Han, Sunghwan Hong, and Seungryong Kim. Cross-view completion models are zero-shot correspondence estimators. In *CVPR*, 2025.

Eduardo Arnold, Jamie Wynn, Sara Vicente, Guillermo Garcia-Hernando, Aron Monszpart, Victor Prisacariu, Daniyar Turmukhambetov, and Eric Brachmann. Map-free visual relocalization: Metric pose relative to a single image. In *ECCV*, 2022.

Yuki M. Asano, Christian Rupprecht, and Andrea Vedaldi. Self-labelling via simultaneous clustering and representation learning. In *ICLR*, 2020.

Mahmoud Assran, Adrien Bardes, David Fan, Quentin Garrido, Russell Howes, Mojtaba Komeili, Matthew Muckley, Ammar Rizvi, Claire Roberts, Koustuv Sinha, Artem Zholus, Sergio Arnaud, Abha Gejji, Ada Martin, Francois Robert Hogan, Daniel Dugas, Piotr Bojanowski, Vasil Khalidov, Patrick Labatut, Francisco Massa, Marc Szafraniec, Kapil Krishnakumar, Yong Li, Xiaodong Ma, Sarath Chandar, Franziska Meier, Yann LeCun, Michael Rabbat, and Nicolas Ballas. V-jepa 2: Self-supervised video models enable understanding, prediction and planning. *arXiv preprint arXiv:2506.09985*, 2025.

Adrien Bardes, Jean Ponce, and Yann LeCun. Vicreg: Variance-invariance-covariance regularization for self-supervised learning. In *ICLR*, 2022.

Gilad Baruch, Zhuoyuan Chen, Afshin Dehghan, Tal Dimry, Yuri Feigin, Peter Fu, Thomas Gebauer, Brandon Joffe, Daniel Kurz, Arik Schwartz, and Elad Shulman. ARKitscenes - a diverse real-world dataset for 3d indoor scene understanding using mobile RGB-d data. In *NeurIPS*, 2021.

Jeffrey R. Binder, Rutvik H. Desai, William W. Graves, and Lisa L. Conant. Where is the semantic system? a critical review and meta-analysis of 120 functional neuroimaging studies. *Cerebral Cortex*, 19(12):2767–2796, December 2009. doi: 10.1093/cercor/bhp055.

Yohann Cabon, Naila Murray, and Martin Humenberger. Virtual kitti 2, 2020. URL https://arxiv.org/abs/2001.10773.

Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features. In *ECCV*, 2018.

Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. In *NeurIPS*, 2020.

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *ICCV*, 2021.

Caroline Chen, Sayna Ebrahimi, Fedor Kitashov, Ming-Hsuan Yang, Leonidas Guibas, Viorica Pătrăucean, and Maks Ovsjanikov. Seese3: Emergence of 3d space in vision features, 2026. URL https://arxiv.org/abs/2607.14228.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *Int. Conf. Machine Learning*, 2020.

Angela Dai, Angel X Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *CVPR*, 2017.

Siran Dai, Qianqian Xu, Peisong Wen, Yang Liu, and Qingming Huang. Exploring structural degradation in dense representations for self-supervised learning. In *NeurIPS*, 2025.

Timothée Darcet, Federico Baldassarre, Maxime Oquab, Julien Mairal, and Piotr Bojanowski. Cluster and predict latent patches for improved masked image modeling, 2025. URL https://arxiv.org/abs/2502.08769.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009.

Carl Doersch, Abhinav Gupta, and Alexei A. Efros. Unsupervised visual representation learning by context prediction. In *ICCV*, 2015.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021.

Johan Edstedt, Qiyu Sun, Georg Bökman, Mårten Wadenbäck, and Michael Felsberg. RoMa: Robust dense feature matching. In *CVPR*, 2024.

Mohamed El Banani, Amit Raj, Kevis-Kokitsi Maninis, Abhishek Kar, Yuanzhen Li, Michael Rubinstein, Deqing Sun, Leonidas Guibas, Justin Johnson, and Varun Jampani. Probing the 3d awareness of visual foundation models. In *CVPR*, 2024.

Russell A. Epstein, Eva Zita Patai, Joshua B. Julian, and Hugo J. Spiers. The cognitive map in humans: spatial navigation and beyond. *Nature Neuroscience*, 20:1504–1513, November 2017. doi: 10.1038/nn.4656.

Adrien Gaidon, Qiao Wang, Yohann Cabon, and Eleonora Vig. Virtual worlds as proxy for multi-object tracking analysis. In *CVPR*, 2016.

Spyros Gidaris, Praveer Singh, and Nikos Komodakis. Unsupervised representation learning by predicting image rotations. In *ICLR*, 2018.

Priya Goyal, Piotr Dollár, Ross Girshick, Pieter Noordhuis, Lukasz Wesolowski, Aapo Kyrola, Andrew Tulloch, Yangqing Jia, and Kaiming He. Accurate, large minibatch sgd: Training imagenet in 1 hour, 2018. URL https://arxiv.org/abs/1706.02677.

Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent-a new approach to self-supervised learning. In *NeurIPS*, 2020.

Richard Hartley and Andrew Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2 edition, 2004.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *CVPR*, 2020.

Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *CVPR*, 2022.

Nan Huang, Pengcheng Yu, Weijia Zeng, James M. Rehg, Angjoo Kanazawa, Haiwen Feng, and Qianqian Wang. Self-improving 4d perception via self-distillation, 2026. URL https://arxiv.org/abs/2604.08532.

Varun Jampani, Kevis-Kokitsi Maninis, Andreas Engelhardt, Arjun Karpur, Karen Truong, Kyle Sargent, Stefan Popov, Andre Araujo, Ricardo Martin-Brualla, Kaushal Patel, Daniel Vlasic, Vittorio Ferrari, Ameesh Makadia, Ce Liu, Yuanzhen Li, and Howard Zhou. Navi: Category-agnostic image collections with high-quality 3d shape and pose annotations. In *NeurIPS*, 2023.

Hanwen Jiang, Hao Tan, Peng Wang, Haian Jin, Yue Zhao, Sai Bi, Kai Zhang, Fujun Luan, Kalyan Sunkavalli, Qixing Huang, et al. Rayzer: A self-supervised large view synthesis model. In *ICCV*, 2025.

Haian Jin, Hanwen Jiang, Hao Tan, Kai Zhang, Sai Bi, Tianyuan Zhang, Fujun Luan, Noah Snavely, and Zexiang Xu. Lvsm: A large view synthesis model with minimal 3d inductive bias. In *ICLR*, 2025.

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.

Weijie Kong, Qi Tian, Zijian Zhang, Rox Min, Zuozhuo Dai, Jin Zhou, Jiangfeng Xiong, Xin Li, Bo Wu, Jianwei Zhang, Kathrina Wu, Qin Lin, Junkun Yuan, Yanxin Long, Aladdin Wang, Andong Wang, Changlin Li, Duojun Huang, Fang Yang, Hao Tan, Hongmei Wang, Jacob Song, Jiawang Bai, Jianbing Wu, Jinbao Xue, Joey Wang, Kai Wang, Mengyang Liu, Pengyu Li, Shuai Li, Weiyan Wang, Wenqing Yu, Xinchi Deng, Yang Li, Yi Chen, Yutao Cui, Yuanbo Peng, Zhentao Yu, Zhiyu He, Zhiyong Xu, Zixiang Zhou, Zunnan Xu, Yangyu Tao, Qinglin Lu, Songtao Liu, Dax Zhou, Hongfa Wang, Yong Yang, Di Wang, Yuhong Liu, Jie Jiang, and Caesar Zhong. Hunyuanvideo: A systematic framework for large video generative models, 2024. URL https://arxiv.org/abs/2412.03603.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *NeurIPS*, 2012.

Vincent Leroy, Yohann Cabon, and Jérôme Revaud. Grounding image matching in 3d with mast3r. In *ECCV*, 2024.

Wenyu Li, Sidun Liu, Peng Qiao, Yong Dou, and Tongrui Hu. Muskie: Multi-view masked image modeling for 3d vision pre-training, 2025. URL https://arxiv.org/abs/2511.18115.

Zhengqi Li and Noah Snavely. Megadepth: Learning single-view depth prediction from internet photos. In *CVPR*, 2018.

Haotong Lin, Sili Chen, Jun Hao Liew, Donny Y. Chen, Zhenyu Li, Guang Shi, Jiashi Feng, and Bingyi Kang. Depth anything 3: Recovering the visual space from any views. In *ICLR*, 2026.

Lu Ling, Yichen Sheng, Zhi Tu, Wentian Zhao, Cheng Xin, Kun Wan, Lantao Yu, Qianyu Guo, Zixun Yu, Yawen Lu, et al. Dl3dv-10k: A large-scale scene dataset for deep learning-based 3d vision. In *CVPR*, 2024.

Thibaut Loiseau, Guillaume Bourmaud, and Vincent Lepetit. Alligat0r: Pre-training through covisibility segmentation for relative camera pose regression. In *NeurIPS*, 2025.

Thibaut Loiseau, Vincent Lepetit, and Guillaume Bourmaud. Revisiting cross-view completion: Self-supervised pre-training via reconstruction error comparison. In *NeurIPS*, 2026.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2019.

Thomas Lucas, Maxime Pietrantoni, Philippe Weinzaepfel, Wonjune Cho, Bardienus Pieter Duisterhof, Vincent Leroy, and Jerome Revaud. Sparse auto-regressive modeling for scene generation from multi-view images. In *ECCV*, 2026.

Nikolaus Mayer, Eddy Ilg, Philip Hausser, Philipp Fischer, Daniel Cremers, Alexey Dosovitskiy, and Thomas Brox. A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation. In *CVPR*, 2016.

Ishan Misra and Laurens van der Maaten. Self-supervised learning of pretext-invariant representations. In *CVPR*, 2020.

Thomas W. Mitchel, Hyunwoo Ryu, and Vincent Sitzmann. True self-supervised novel view synthesis is transferable. In *ICLR*, 2026.

Lorenzo Mur-Labadia, Matthew Muckley, Amir Bar, Mahmoud Assran, Koustuv Sinha, Michael Rabbat, Yann LeCun, Nicolas Ballas, and Adrien Bardes. V-jepa 2.1: Unlocking dense features in video self-supervised learning. In *ECCV*, 2026.

David Nordström, Johan Edstedt, Fredrik Kahl, and Georg Bökman. MuM: Multi-view masked image modeling for 3d vision. In *CVPR*, 2026.

Maxime Oquab, Timothée Darcet, Theo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Russell Howes, Po-Yao Huang, Hu Xu, Vasu Sharma, Shang-Wen Li, Wojciech Galuba, Mike Rabbat, Mido Assran, Nicolas Ballas, Gabriel Synnaeve, Ishan Misra, Herve Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning robust visual features without supervision. In *Transactions on Machine Learning Research*, 2024.

Henri Poincaré. *La Valeur de la Science*. Flammarion, Paris, 1905.

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 *ICCV*, 2021.

Alexandre Sablayrolles, Matthijs Douze, Cordelia Schmid, and Hervé Jégou. Spreading vectors for similarity search. *ICLR*, 2019.

Paul-Edouard Sarlin, Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superglue: Learning feature matching with graph neural networks. In *CVPR*, 2020.

Leo Segre, Or Hirschorn, and Shai Avidan. Multi-view foundation models. In *ECCV*, 2026.

Oriane Siméoni, Huy V. Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, Michal Ramamonjisoa, Francisco Massa, Daniel Haziza, Luca Wehrstedt, Jianyuan Wang, Timothée Darcet, Théo Moutakanni, Leonel Sentana, Claire Roberts, Andrea Vedaldi, Jamie Tolan, John Brandt, Camille Couprie, Julien Mairal, Hervé Jégou, Patrick Labatut, and Piotr Bojanowski. Dinov3. *Transactions on Machine Learning Research*, 2026.

Richard Sinkhorn and Paul Knopp. Concerning nonnegative matrices and doubly stochastic matrices. *Pacific Journal of Mathematics*, 1967.

Michal Starý, Julien Gaubil, Ayush Tewari, and Vincent Sitzmann. Understanding Multi-View Transformers. In *ICCV 2025 E2E3D Workshop*, 2025.

Jianlin Su, Yu Lu, Shengfeng Pan, Ahmed Murtadha, Bo Wen, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding, 2023. URL `https://arxiv.org/abs/2104.09864`.

Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In *NeurIPS*, 2017.

Fabio Tosi, Yiyi Liao, Carolin Schmitt, and Andreas Geiger. Smd-nets: Stereo mixture density networks. In *CVPR*, 2021.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding, 2018. URL https://arxiv.org/abs/1807.03748.

Huy V. Vo, Vasil Khalidov, Timothe Darcet, Tho Moutakanni, Nikita Smetanin, Marc Szafraniec, Hugo Touvron, Camille Couprie, Maxime Oquab, Armand Joulin, Herv Jgou, Patrick Labatut, and Piotr Bojanowski. Automatic data curation for self-supervised learning: A clustering-based approach. *Transactions on Machine Learning Research*, 2024.

Khiem Vuong, Anurag Ghosh, Deva Ramanan, Srinivasa Narasimhan, and Shubham Tulsiani. Aerialmegadepth: Learning aerial-ground reconstruction and view synthesis. In *CVPR*, 2025.

Jiahao Wang, Yufeng Yuan, Rujie Zheng, Youtian Lin, Jian Gao, Lin-Zhuo Chen, Yajie Bao, Yi Zhang, Chang Zeng, Yanxi Zhou, et al. Spatialvid: A large-scale video dataset with spatial annotations. In *CVPR*, 2026a.

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

Jianyuan Wang, Minghao Chen, Shangzhan Zhang, Nikita Karaev, Johannes Schönberger, Patrick Labatut, Piotr Bojanowski, David Novotny, Andrea Vedaldi, and Christian Rupprecht. VGGT-Ω. In *CVPR*, 2026b.

Letian Wang, Chuhan Zhang, Rishabh Kabra, Jasper Uijlings, Steven Waslander, Andrew Zisserman, Joao Carreira, Kaiming He, Misha Andriluka, Eduard Gabriel Bazavan, Andrei Zanfir, and Cristian Sminchisescu. Video generation models are general-purpose vision learners. In *ECCV*, 2026c.

Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jerome Revaud. Dust3r: Geometric 3d vision made easy. In *CVPR*, 2024.

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 *IROS*, 2020.

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

Philippe Weinzaepfel, Vincent Leroy, Thomas Lucas, Romain Brégier, Yohann Cabon, Vaibhav Arora, Leonid Antsfeld, Boris Chidlovskii, Gabriela Csurka, and Jerome Revaud. CroCo: Self-Supervised Pre-training for 3D Vision Tasks by Cross-View Completion. In *NeurIPS*, 2022.

Philippe Weinzaepfel, Vaibhav Arora, Yohann Cabon, Thomas Lucas, Romain Brégier, Vincent Leroy, Gabriela Csurka, Leonid Antsfeld, Boris Chidlovskii, and Jérôme Revaud. Croco v2: Improved cross-view completion pre-training for stereo matching and optical flow. In *ICCV*, 2023.

Lihe Yang, Shang-Wen Li, Yang Li, Xinjie Lei, Dong Wang, Abdelrahman Mohamed, Saining Xie, Hengshuang Zhao, Kaiming He, and Hu Xu. In pursuit of pixel supervision for visual pre-training. In *CVPR*, 2026.

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. In *CVPR*, 2020.

Chandan Yeshwanth, Yueh-Cheng Liu, Matthias Nießner, and Angela Dai. Scannet++: A high-fidelity dataset of 3d indoor scenes. In *ICCV*, 2023.

Richard Zhang, Phillip Isola, and Alexei A Efros. Colorful image colorization. In *ECCV*, 2016.

Qitao Zhao, Hao Tan, Qianqian Wang, Sai Bi, Kai Zhang, Kalyan Sunkavalli, Shubham Tulsiani, and Hanwen Jiang. E-rayzer: Self-supervised 3d reconstruction as spatial visual pre-training. In *CVPR*, 2026.

Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *CVPR*, 2017.

Jinghao Zhou, Chen Wei, Huiyu Wang, Wei Shen, Cihang Xie, Alan Yuille, and Tao Kong. Image BERT pre-training with online tokenizer. In *ICLR*, 2022.

Tinghui Zhou, Richard Tucker, John Flynn, Graham Fyffe, and Noah Snavely. Stereo magnification: Learning view synthesis using multiplane images. *ACM Trans. Graph. (Proc. SIGGRAPH)*, 37, 2018. URL https://arxiv.org/abs/1805.09817.

Weichen Zhou, Yawen Zou, Chunzhi Gu, Ran Dong, Haoran Xie, and Chao Zhang. Understanding geometric representations in self-supervised vision transformers via subspace intervention. In *ECCV*, 2026.

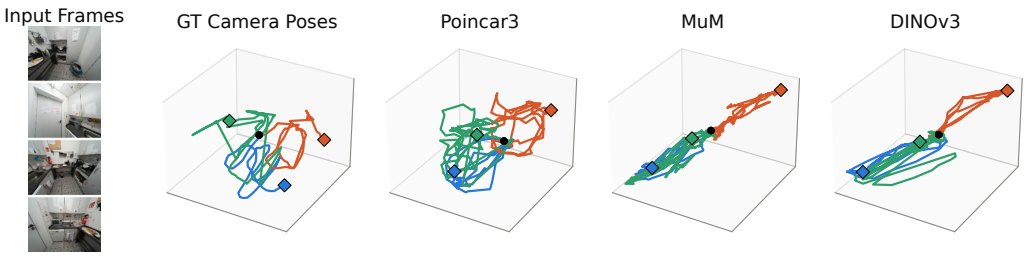


Figure 11: **Poincaré adapter.** We visualize the alignment between visual features and camera trajectories with a lightweight *Poincaré adapter*, following Fig. 1 of Chen et al. (2026). Applied to Poincar3, it unrolls the feature space toward the ground-truth camera trajectories.

A Things That Did Not Work

While not included in the ablation in the main paper (Tab. 4), we experimented with multiple SSL objectives before arriving at our formulation. We primarily sought a simpler objective, such as BYOL (Grill et al., 2020), which does not use centering or Sinkhorn-Knopp. While showing initial promise, the feature maps quickly degraded during training, resulting in unsatisfactory qualitative results. We attempted to mitigate this degradation through regularization, but were unable to obtain stable training. Neither centering nor Sinkhorn–Knopp normalization, nor more sophisticated regularization methods such as VICReg (Bardes et al., 2022), resolved the issue.

We also tried being more sloppy with our data curation. However, in contrast to RGB reconstruction, self-distillation appears to be more sensitive to low-quality data samples. This is likely why the DINO-team has spent significant effort in creating elaborate data curation pipelines (Vo et al., 2024).

B Baselines

In the paper, our core comparisons are to the features obtained by state-of-the-art SSL methods. Namely, DINOv3, MuM, and Muskie. We omit the comparison to CroCo, which was already shown to be inferior to MuM and Muskie in their respective papers. Furthermore, MuM showed a drastic improvement over VJEPA 2 on geometric understanding. At times, we compare to the features obtained by supervised 3D models such as VGGT-Ω, π^3, and DepthAnything3 (DA3). These constitute the state-of-the-art supervised methods and have garnered widespread interest from the 3D vision community. We include these as comparisons where they constitute a fair comparison, as in correspondence estimation, whereas in others, such as feed-forward reconstruction, their training objective is tailored for that specific task and are thus omitted. Lastly, we do not compare to GenCeption, as it is diffusion-based, or Alligat0r, as it requires covisibility labels.

C Additional Experiments

C.1 Qualitative

To understand what the model has learned, we visualize the PCA of the patch features after the decoder. We show the results for an image pair in Fig. 12. Furthermore, we visualize the alignment of the Poincaré adapter with the ground-truth camera trajectory in Fig. 11 and additional baselines for qualitative multi-view correspondence tracks in Fig. 13.

C.2 Two-view Matching

We now consider two-view matching. We employ a lightweight protocol by training linear probes and using zero-shot nearest-neighbor (NN) in feature space. This evaluation protocol follows from Edstedt et al. (2024) and Nordström et al. (2026). We report the accuracy in Tab. 5 on ScanNet-1500. We find that Poincar3 outperforms all other methods on both zero-shot matching and linear probing.

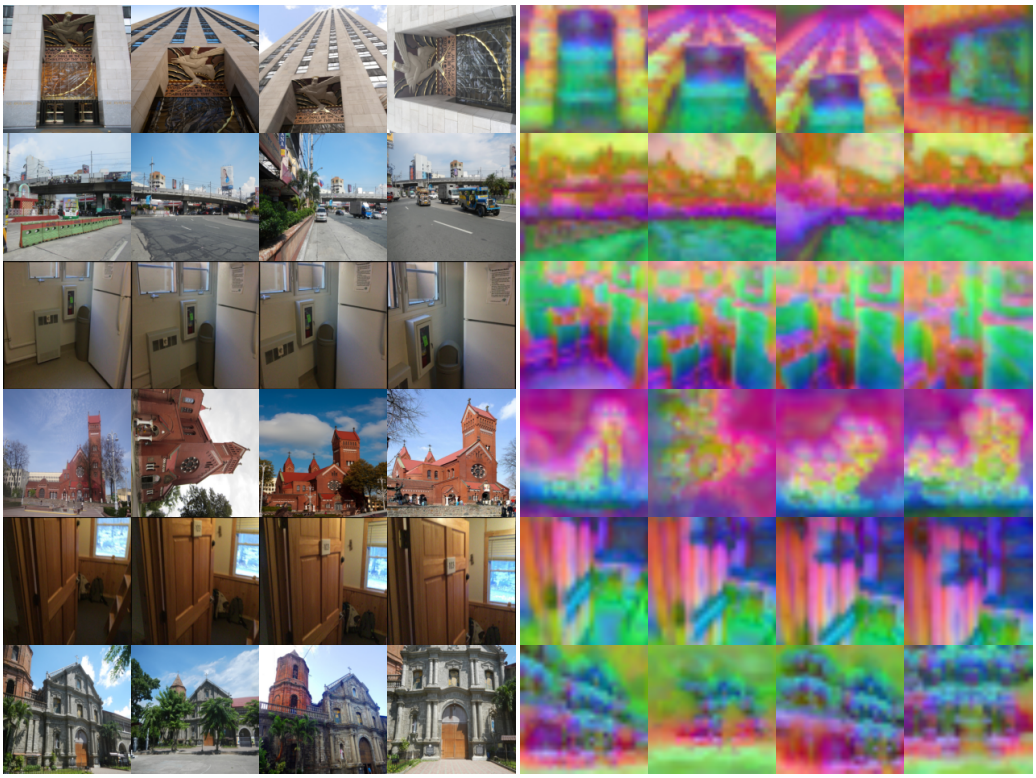


Figure 12: **PCA feature visualization.** We visualize the 3 principal components as RGB for different sequences from the same scene.

Table 5: **Lightweight two-view matching.** Matching robustness on ScanNet-1500 (Dai et al., 2017; Sarlin et al., 2020).

Method	Nearest-Neighbor			Linear Probe		
PCK →	8px	16px	32px	8px	16px	32px
Feed-forward Reconstruction						
VGGT-Ω	2.3	6.8	17.0	26.4	47.9	67.5
DA3	0.8	2.4	6.8	29.2	49.0	66.7
π^3	8.0	17.1	30.2	32.9	55.3	73.5
Self-supervised Models						
DINOv3	17.5	35.0	52.2	35.3	60.4	77.9
Muskie	6.6	16.2	32.1	34.8	54.5	69.8
MuM	15.7	31.9	51.1	40.7	63.6	78.9
Poincar3	**24.3**	**42.5**	**58.6**	**45.3**	**67.0**	**79.3**

C.3 Semantic Tasks

We compare the semantic performance in Tab. 6. Without being trained on ImageNet, Poincar3 outperforms other multi-view SSL models on image classification and semantic segmentation. However, removing the local crops and training on significantly more geometric data leads to significant performance degradation compared to DINOv3. For a fair comparison, we use the output of the final layer for all the models whereas in MuM, the semantic performance is evaluated after only the encoder. We follow the evaluation protocol of Darcet et al. (2025).

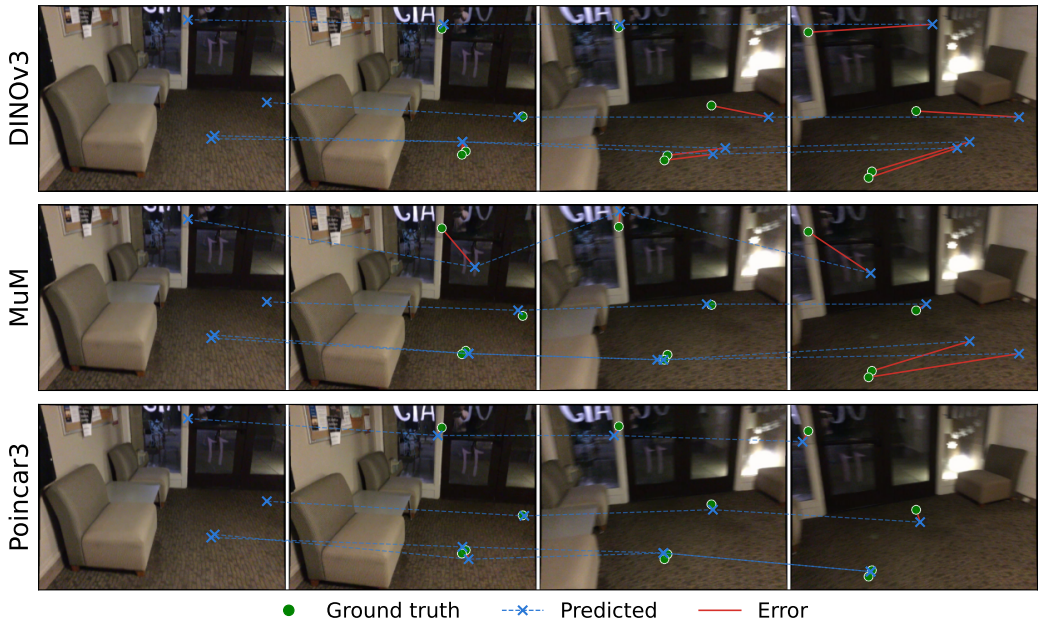


Figure 13: **Additional multi-view correspondence visualizations.** Including also DINOv3 as a baseline.

Table 6: **Semantic performance.** Image classification accuracy on ImageNet-1K and semantic segmentation (mIoU) on ADE20K.

Context	ImageNet-1K (Deng et al., 2009; Krizhevsky et al., 2012)	ADE20K (Zhou et al., 2017)
DINOv3	85.1	49.8
Muskie	27.5	6.4
MuM	27.5	4.6
Poincar3	**34.4**	**9.6**

C.4 Layer-wise Multi-View Correspondence Estimation

We investigate the geometric information at different depths for the models by multi-view correspondence estimation accuracy in Fig. 14. We find that Poincar3 consistently outperforms the baselines at all depths.

D Implementation Details

D.1 Hyper-parameters

Architecture. Poincar3 uses a ViT-L/16 backbone with DINOv3-style RoPE position embeddings, QK-norm, and LayerScale, initialized from scratch. The multi-view decoder has 12 blocks, alternating full cross-frame attention with a cheaper register-bottleneck attention at blocks $\{2, 6, 9\}$ (zero-indexed) using 16 register tokens.

Self-distillation objective. We use separate DINO/iBOT heads for the patch and global tokens, each with 65,536 prototypes, hidden dimension 2048, and bottleneck dimension 256. Teacher targets are normalized with Sinkhorn–Knopp (3 iterations) rather than centering, which we found more resistant to collapse. The student temperature is fixed at 0.1, while the teacher temperature warms up from 0.04 to 0.07 over the first 30k steps; the teacher EMA decay (λ) follows the same warmup schedule, from 0.994 to 0.999. The last layer of both heads is frozen for the first 2k steps.

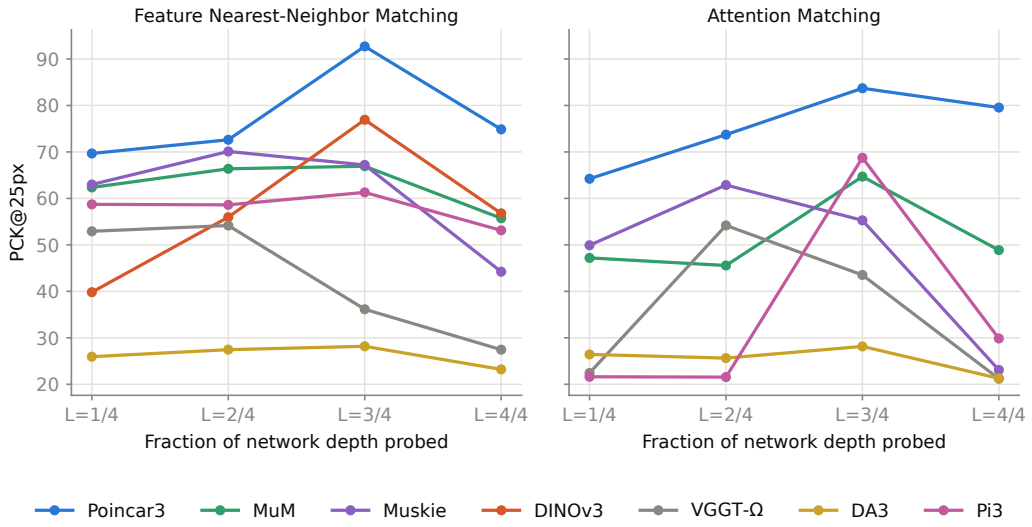


Figure 14: **Layer-wise performance.** We plot the multi-view correspondence accuracy at different depths. We find that Poincar3 encodes multi-view geometry throughout the network, while the performance of supervised methods collapses at later layers.

Multi-view sampling. Each training sequence draws a student frame count uniformly from $[2, 24]$ and, crucially, feeds the teacher $[0, 12]$ additional frames the student never sees, unmasked; the loss is computed only over the frames both networks observe. Patches are masked independently per frame with probability 0.5 and a mask ratio sampled from $[0.1, 0.5]$, following iBOT. The global objective compares teacher and student tokens frame-by-frame. Batches are built dynamically to a fixed budget of 64 frames per GPU. Images are resized to 256×256 with patch size 16. As we train on 8 H200 GPUs, the maximum number of frames can go into a batch is 512.

Optimization. We train with AdamW and a fixed learning rate of 2×10^{-4}, warmed up over 10k steps. Weight decay is 0.04 and gradients are clipped to norm 1.0. Training uses bf16 mixed precision.

D.2 Data

We detail our dataset mixture in Tab. 7. The majority of the scenes come from videos from the internet, providing a rich source of data that can only be mined through SSL. Note there is no overlap between the training scenes and the scenes used for evaluation in Sec. 4.

D.3 Training Dynamics

In Fig. 15, we visualize how six different evaluation metrics evolve during training and how performance is impacted by initializing the encoder from DINOv3. Furthermore, we provide plots of the stabilization metrics we tracked throughout training in Fig. 16. While the losses are uninformative in a teacher–student setup, it is encouraging to see that the teacher's features are not collapsing but rather becoming more discriminative as training progresses (as illustrated by the standard deviation and the effective rank).

E Evaluation

E.1 Feed-forward reconstruction

We broadly follow the training in Wang et al. (2026b), but make minor adjustments to fit with our limited compute budget. The protocol we use is outlined below.

Table 7: **Dataset mixture** for Poincar3. The top part contains unlabeled large-scale internet video datasets, while the bottom part contains 3D datasets with annotations. The weight is proportional to the probability of sampling from the respective dataset.

Datasets	Type / Source	Weight	№Scenes
SpatialVID (Wang et al., 2026a)	Outdoor / Video	1	176,749
DL3DV (Ling et al., 2024)	Mixed / Video	1	10,000
RealEstate10K (Zhou et al., 2018)	Indoor / Video	1	7,850
MegaDepth (Li & Snavely, 2018)	Outdoor / MVS	1	169
AerialMD (Vuong et al., 2025)	Aerial / MVS	1	124
BlendedMVS (Yao et al., 2020)	Aerial / Mesh	1	493
Hypersim (Roberts et al., 2021)	Indoor / Graphics	1	393
TartanAir v2 (Wang et al., 2020)	Outdoor / Graphics	1	46
Map-Free (Arnold et al., 2022)	Object-centric / MVS	1	397
ScanNet++ v2 (Yeshwanth et al., 2023)	Indoor / Mesh	1	856
FlyingThings3D (Mayer et al., 2016)	Outdoor / Graphics	0.5	2,239
ARKitScenes (Baruch et al., 2021)	Indoor / RGB-D	0.1	5,047
UnrealStereo4k (Tosi et al., 2021)	Outdoor / Graphics	0.01	8
Virtual KITTI 2 (Gaidon et al., 2016; Cabon et al., 2020)	Outdoor / Graphics	0.01	5
Total			**204,376**

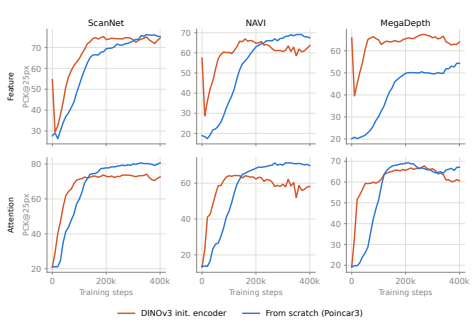


Figure 15: **Detailed evaluation metrics.** Multi-view correspondence estimation performance, from scratch vs. DINOv3 init.



Figure 16: **Detailed stability metrics.** We visualize how the loss terms evolve during training as well as stability metrics.

Training. Every feed-forward reconstruction model shares the same pair of prediction heads on top of a backbone: a *camera head*, a 4-layer self-attention trunk that mixes each frame's camera and register tokens across the whole sequence in one shot and regresses a 9D pose encoding per frame (translation, rotation quaternion, field of view), and a *depth head*, a DPT-style multi-scale fusion decoder (following VGGT/Depth-Anything) that reads dense tokens from four evenly-spaced depths of the backbone and regresses per-pixel depth and confidence independently per frame (no cross-frame mixing). Training data is a weighted mixture of posed, depth-annotated multi-view datasets (HyperSim, ScanNet++, MegaDepth, BlendedMVS, Map-free, MegaSynth, and SpatialVid). Each batch draws one sequence length (uniform in $[2, 24]$ frames) and one resolution bucket, following the same dynamic-batching scheme used for pretraining of Poincar3.

We compare three training protocols that trade off how much of the backbone is updated:

1. **Frozen backbone, heads only.** The backbone (encoder and, where present, its pretrained cross-view decoder) stays entirely frozen, and only the camera/depth heads are trained on top of its existing features, isolating how much 3D structure a backbone's pretraining objective already encodes.
2. **Frozen backbone + adapter.** The backbone stays frozen, but a small, freshly-initialized adapter (two layers of the same alternating frame/inter-frame self-attention used in our

own cross-view decoder) sits between the frozen per-frame features and the heads, giving every backbone an identical amount of new trainable cross-view capacity, which matters for backbones with no pretrained cross-view mixing at all (e.g. DINOv3, a plain per-frame ViT).

3. **Full finetune.** The entire backbone is updated jointly with the heads end to end, optionally regularized against representational drift with a feature-distillation term against a frozen VGGT-Ω teacher, matching both dense patch tokens and the camera token. From this experiment we do not include MuM or Muskie as it would constitute an unfair comparison as their architectures differ. For the DINOv3 comparison, we substitute the encoder for DINOv3 ViT-L/16. Muskie showed that this is superior to transplanting the DINO weights also into the multi-view transformer.

In all three protocols the training objective is $\mathcal{L} = \lambda_{\text{cam}}\mathcal{L}_{\text{cam}} + \lambda_{\text{depth}}\mathcal{L}_{\text{depth}}$, with $\lambda_{\text{cam}} = \lambda_{\text{depth}} = 1$. The camera loss is a flat L1 on the predicted vs. ground-truth pose encoding $p = (t, q, f)$ (translation, rotation quaternion, field-of-view):

$$\mathcal{L}_{\text{cam}} = \|\hat{p} - p\|_1. \tag{6}$$

The depth loss combines a confidence-weighted term, a plain regression term, and a multi-scale gradient term over pixels with valid ground-truth depth (after dropping the worst 2% by quantile):

$$\mathcal{L}_{\text{depth}} = \mathbb{E}\big[\gamma\,|\hat{d} - d|\,c - \alpha \log c\big] + \mathbb{E}\big[|\hat{d} - d|\big] + \mathcal{L}_{\text{grad}}, \tag{7}$$

where c is the predicted per-pixel confidence, γ=1, α=0.2 (the $-\alpha \log c$ term discourages trivially collapsing confidence to zero), and $\mathcal{L}_{\text{grad}}$ averages an L1 gradient-matching term over 4 progressively downsampled resolutions. Ground-truth depth and camera translations are rescaled per training batch, since absolute scale is unobservable from monocular images alone.

Evaluation. We evaluate the resulting checkpoints along two axes: relative camera pose and multi-view point-cloud geometry. For the former, we follow Wang et al. (2025), while for the latter we follow Wang et al. (2026d). For pose, a single joint forward pass over N=10 sampled frames of a test sequence predicts every frame's pose in one model-chosen reference frame (VGGT-style); since this absolute frame carries no ground-truth correspondence, we score every one of the $\binom{N}{2}$ frame pairs' *relative* pose instead – rotation error is the geodesic angle between predicted and ground-truth quaternions, translation error is the angle between the (sign-disambiguated) predicted and ground-truth translation directions – and report the area under the cumulative-accuracy curve of $\max(\text{rotation error}, \text{translation error})$, thresholded at 30° (AUC@30), averaged first within each test scene and then across scenes so scenes contributing more sampled sequences (e.g. RE10K) do not dominate. Test sets are MegaDepth-1500, RE10K, ScanNet-1500, and 50 held-out ScanNet++ scenes, each following its own reference protocol's image-loading convention.

For point-cloud quality, we unproject each backbone's predicted per-frame depth through its predicted camera parameters into a world-space point cloud, on 5-view sequences from DTU and ETH3D. Since predicted geometry is only defined up to an unknown similarity transform, we first coarsely align it to the ground-truth point cloud with Umeyama's method and then refine with rigid ICP. We report point-to-point accuracy and completion (mean/median nearest-neighbor distance predicted→ground-truth and vice versa) and normal consistency (mean cosine similarity between each point's PCA-estimated normal and its nearest neighbor's), averaged over sequences – reproducing VGGT's own multi-view reconstruction evaluation protocol.

E.2 Multi-view correspondence estimation

Our protocol follows El Banani et al. (2024). In particular, we evaluate zero-shot correspondence directly, without any task-specific finetuning, by feeding a cluster of 8 covisible views of one scene through a backbone in a single multi-view forward pass and tracking 100 query points sampled in the first view across the remaining 7. Ground-truth tracks come from reprojecting each query point through its ground-truth depth, pose, and intrinsics into every other view, with a depth-based occlusion test discarding views where the reprojected point is not actually visible; points never visible in any other view are dropped entirely. We evaluate on NAVI and ScanNet (both with metric-scale ground-truth depth) and MegaDepth (unscaled SfM depth).

We score two different ways of reading off a correspondence from the same forward pass. *Feature matching* takes the source query point's dense feature (bilinearly sampled at its exact sub-patch location) and finds its 1-nearest-neighbor by cosine similarity in each target view's dense feature map. *Attention matching* instead reads the correspondence directly off the backbone's own softmax cross-view attention: the source point's query vector attends over every view's keys, and the arg-max of that distribution restricted to a target view's patch tokens is the match – this asks whether the model's own attention agrees with correspondence, not just its output feature similarity. Both are read from features/attention at a specific network depth; since backbones differ widely in depth and in which blocks carry genuine cross-view mixing, we sweep four evenly-spaced relative depths (quarter-points of each backbone's own valid block range) and, following common practice, report the single best-performing depth per backbone and correspondence method.

Given a predicted match and its ground-truth target location, we report the fraction of visible query points landing within k pixels of the ground-truth match (PCK@k, $k \in \{5, 10, 25, 50\}$), plus the analogous 3D accuracy after unprojecting both points with ground-truth depth.

E.3 Emergent SE(3) Structure

We follow Chen et al. (2026). As the code is unpublished as of writing, we do our best to reimplement the core protocol. In particular, we fit a Poincaré adapter per scene on 20 held-out ScanNet++ (Yeshwanth et al., 2023) scenes, using a chronological $80/20$ split of each scene's frames and sweeping strides $s \in \{2, \dots, 60\}$, and measure R^2 on the held-out pairs. As R^2 is unbounded below, its raw mean is dominated by a few degenerate scenes; we therefore report the clipped mean $\overline{R^2} = \mathrm{mean}\left(\max(0, R^2)\right)$ together with the fraction of scenes exceeding $R^2 > 0$ and $R^2 > 0.3$. All values are the best over four relative encoder depths and are averaged over 10 adapter seeds ($\pm$ 95%-confidence interval where the scene set is fixed and shared across methods). Methods marked MV additionally receive an eight-frame context window with a frame spacing of 5, activating their cross-view attention; only the anchor frame's feature is retained.

E.4 Visualizing feature correlations

Here we detail how we perform a visualization such as Fig. 5. Given a cluster of covisible frames processed jointly in one multi-view forward pass, we mark a query patch in the first frame and, for every other frame, compute the cosine similarity between the query's dense feature and every patch of that frame (clipped below zero). Each row (backbone) is independently rescaled to its own 5^{th}-percentile–to-max range and rendered as a semi-transparent Turbo-colormap overlay (opacity 0.25–0.85, so weakly-correlated regions stay visible rather than fading to the bare image), with a circle marking the arg-max match. Rescaling per backbone sacrifices strict cross-row comparability of absolute intensity in exchange for making each backbone's correlation *dispersion* legible: a backbone that discriminates well shows a small, concentrated hot region, while one that does not washes warm over most of the frame.