Alligat0R: Pre-Training through Covisibility Segmentation for Relative Camera Pose Regression

Thibaut Loiseau[1] **Guillaume Bourmaud**[2] **Vincent Lepetit**[1]
[1] LIGM, Ecole des Ponts, Univ. Gustave Eiffel, CNRS, France
[2] Univ. Bordeaux, CNRS, Bordeaux INP, IMS, UMR 5218, France
{thibaut.loiseau,vincent.lepetit}@enpc.fr guillaume.bourmaud@u-bordeaux.fr

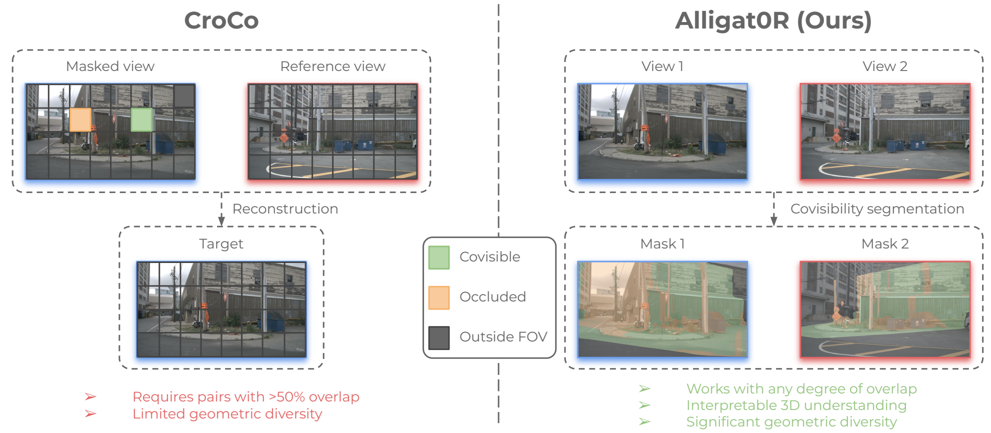
CroCo
Masked view
Reference view
Reconstruction
Target
➢ Requires pairs with >50% overlap
➢ Limited geometric diversity
Covisible
Occluded
Outside FOV
Alligat0R (Ours)
View 1
View 2
Covisibility segmentation
Mask 1
Mask 2
➢ Works with any degree of overlap
➢ Interpretable 3D understanding
➢ Significant geometric diversity

Figure 1: **We introduce Alligat0R, a novel pretraining method for binocular vision.** Alligat0R explicitly segments pixels as covisible, occluded, or outside field-of-view, overcoming the fundamental limitation of CroCo [47, 48] which attempts to reconstruct potentially non-covisible regions.

Abstract

Pre-training techniques have greatly advanced computer vision, with CroCo's cross-view completion approach yielding impressive results in tasks like 3D reconstruction and pose regression. However, cross-view completion is ill-posed in non-covisible regions, limiting its effectiveness. We introduce Alligat0R, a novel pre-training approach that replaces cross-view learning with a covisibility segmentation task. Our method predicts whether each pixel in one image is covisible in the second image, occluded, or outside the field of view, making the pre-training effective in both covisible and non-covisible regions, and provides interpretable predictions. To support this, we present Cub3, a large-scale dataset with 5M image pairs and dense covisibility annotations derived from the nuScenes and ScanNet datasets. Cub3 includes diverse scenarios with varying degrees of overlap. The experiments show that our novel pre-training method Alligat0R significantly outperforms CroCo in relative pose regression. Alligat0R and Cub3 will be made publicly available.

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

1 Introduction

Pre-training techniques have revolutionized computer vision by enabling large models to learn rich representations [5, 29, 7, 19, 18, 2, 55]. In 3D computer vision, CroCo [47, 48] pioneered cross-view completion as a pretext task, where one view is partially masked and reconstructed using visible portions along with a second reference view of the same scene (see Figure 1 left). This approach allows to learn powerful 3D features and has made possible impressive results on downstream tasks such as 3D reconstruction [45, 14, 49, 42], 3D pose regression [9], camera calibration [26], 4D reconstruction [52, 23, 44] and Gaussian splatting [35, 51].

Despite its success, the cross-view completion pretext task has a fundamental limitation: pretraining is only effective in covisible regions, as the task is ill-posed in non-covisible regions (see Figure 1). As a consequence, Croco [47, 48] relies on pairs with at least 50% of covisible regions.

In this paper, we present Alligat0R, a novel pre-training approach that offers an alternative to the cross-view completion objective through a covisibility segmentation task. Instead of reconstructing masked regions, our method explicitly predicts whether each pixel in one image is: (1) covisible in the second image, (2) occluded, or (3) outside the field of view (FOV). This formulation offers several advantages: it makes the pre-training effective in both covisible and non-covisible regions, it provides interpretable predictions that reveal the model's geometric understanding, and it aligns more directly with the correspondence reasoning required in downstream binocular vision tasks.

To enable our approach, we introduce Cub3, a large-scale dataset comprising two sub datasets, each of 2.5 million image pairs with dense covisibility annotations derived from nuScenes [4] and ScanNet [8] datasets. Cub3 includes challenging scenarios with varying degrees of overlap between views, providing a more diverse training signal than previous approaches.

One might notice that Alligat0R is not strictly self-supervised, unlike some earlier pre-training methods, as it relies on covisibility annotations. However, the same can be said for CroCo, which requires filtering image pairs to retain only those with at least 50% overlap. The creation of our covisibility annotations for nuScenes is more complex than for ScanNet, as the ground truth poses and depth maps are not available, however the process is fully automated. This process is very similar to the one used in RUBIK [28] and relies on registered video sequences and depth predictions.

Our contributions can be summarized as follows:

1. We introduce covisibility segmentation as a novel pre-training objective for binocular vision tasks, replacing the cross-completion-based approach of prior work while maintaining the same network architecture.
2. We create and release Cub3, a large-scale dataset with dense covisibility annotations derived from nuScenes and ScanNet datasets.
3. We show that Alligat0R provides an effective alternative to CroCo pre-training when fine-tuned on the relative pose regression task, particularly for challenging scenarios with limited overlap between views.
4. We demonstrate the effectiveness of our pre-training and provide insights into the model's cross-view reasoning capabilities through interpretable segmentation outputs.

Our experiments demonstrate that explicitly learning to understand covisibility relationships between image pairs leads to more robust and transferable features for relative pose regression compared to reconstruction-based approaches.

2 Related Work

Pretraining on Image Pairs. To the best of our knowledge, CroCo [47] pioneered the extension of masked image modeling [20] to image pairs, introducing cross-view completion. CroCo v2 [48] applied this cross-view completion framework to large amounts of data. P-Match [56] introduced a variation where both images are partially masked to pre-train an image matching model. Another variant, masked appearance transfer, was proposed in [53] for object tracking, with a similar approach in [36]. Let us highlight that the foundational models DUSt3R [45] and MASt3R [26] are built on CroCo. While all these works rely on cross-view completion, we depart from this framework and propose a novel pretraining objective based on covisibility segmentation.

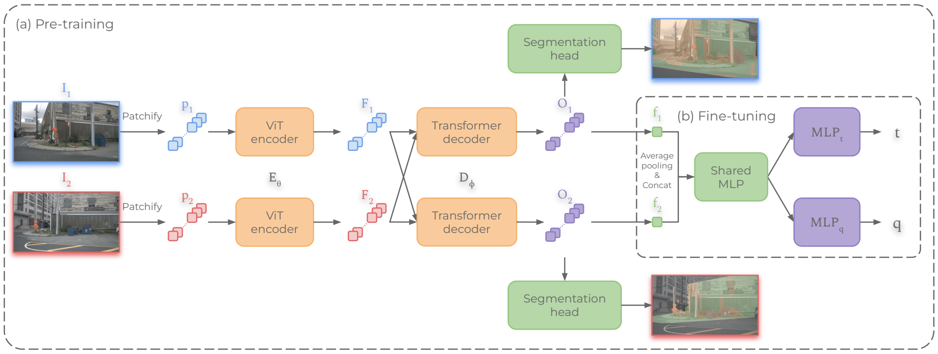

Figure 2: **Overview of Alligat0R.** (a) During pre-training, we use the same architecture as CroCo but replace the reconstruction objective with a covisibility segmentation task, without masking, where each pixel in one view is classified as covisible, occluded, or outside FOV with respect to the other view. (b) For fine-tuning on the relative pose regression task, we pool features from both views, process them through a shared MLP, and use separate heads for predicting translation and rotation.

Considering covisibility across image pairs. Ground-truth covisibility masks are widely used in image matching methods [31, 12, 27, 54, 37, 46, 13, 6, 43, 40, 11, 15, 17, 25, 16, 22, 39, 32] to compute overlaps and select training pairs. Several approaches [32, 13, 41, 11, 16, 3] also leverage these masks to learn *matchability* scores, which are used at test time to select correspondences. Additionally, [21] jointly learns covisibility and relative planar pose between 360° panoramas. To the best of our knowledge, our work is the first to learn visual representations from covisibility segmentation.

Fine-tuning for Relative Pose Regression. Reloc3R [9] recently fine-tuned the foundational model DUSt3R [45], which is based on CroCo, on large-scale data for relative pose regression. In our experiments, we adopt a similar strategy to evaluate our novel pretraining method against CroCo. However, we fine-tune Alligat0R for metric relative translation, whereas Reloc3R predicts only the direction of the relative translation.

3 Method

In this section, we describe our novel pre-training approach, Alligat0R, which replaces the cross-view completion objective used in CroCo [47, 48] by a segmentation task. We first outline the overall architecture, which remains largely similar to CroCo, and then detail our covisibility segmentation pre-training objective. Finally, we explain how our model can be fine-tuned for downstream tasks such as relative pose regression.

3.1 Architecture Overview

Our architecture closely follows the design of CroCo, which consists of an encoder and a decoder. The encoder is a ViT [10] that processes the input images by dividing them into non-overlapping patches and encoding them into feature representations. The decoder is another transformer that combines information from both views, using cross-attention layers, to make predictions.

We adopt a symmetric architecture for our forward pass, where both images are processed in the same way without any masking. This differs from CroCo, which uses an asymmetric approach that heavily masks one image while leaving the other unmasked. Our symmetric design is not only more efficient but also better aligned with downstream binocular vision tasks, which typically process unmasked images.

As shown in Figure 2 (a), given two images $\mathtt{I}_1$ and $\mathtt{I}_2$ of the same scene taken from different viewpoints, we first divide both into non-overlapping patches, denoted as tokens $\mathtt{P}_1 = \{\mathtt{P}_1^1, \ldots, \mathtt{P}_1^{N_1}\}$ and $\mathtt{P}_2 = \{\mathtt{P}_2^1, \ldots, \mathtt{P}_2^{N_2}\}$.

The encoder E_θ processes the tokens from both images independently:

$$\mathtt{F}_1 = E_\theta(\mathtt{P}_1), \quad \mathtt{F}_2 = E_\theta(\mathtt{P}_2)\ , \tag{1}$$

and the decoder D_ϕ then takes these features and processes them to enable cross-view reasoning:

$$\mathtt{O}_1 = D_\phi(\mathtt{F}_1, \mathtt{F}_2), \quad \mathtt{O}_2 = D_\phi(\mathtt{F}_2, \mathtt{F}_1)\ , \tag{2}$$

where $\mathtt{O}_1$ and $\mathtt{O}_2$ represent the output features from the decoder. The decoder contains cross-attention mechanisms that allow information exchange between features from both views.

3.2 Covisibility Segmentation Pre-Training

A key difference between our approach and CroCo lies in the pre-training objective. Instead of reconstructing masked pixels, we formulate the pre-training task as a covisibility segmentation problem. Our goal is to predict for each pixel in each image whether it is:

1. **Covisible:** the pixel corresponds to a 3D point that is also visible in the other image.
2. **Occluded:** the pixel corresponds to a 3D point that is occluded in the other image.
3. **Outside FOV:** the pixel corresponds to a 3D point that is outside the field of view in the other image.

Formally, for each patch token, the decoder produces a feature vector that is processed by a fully-connected layer to output probabilities for each of the three classes, for each pixel:

$$\hat{\mathtt{Y}}_{ikj} = \text{softmax}(\mathtt{W} \cdot \mathtt{O}_{ik} + \mathbf{b})_j\ , \tag{3}$$

where $\hat{\mathtt{Y}}_{ikj} \in \mathbb{R}^3$ represents the predicted probabilities for the three covisibility classes for pixel j in patch k of image i, and $\mathtt{W}$ and $\mathbf{b}$ are learnable parameters.

During pre-training, we optimize the network using a cross-entropy loss in each image:

$$\mathcal{L}_{\text{ce}_i} = -\frac{1}{N_i}\sum_{j=1}^{N_i} \ln(\hat{\mathtt{Y}}_{ij,c_{ij}})\ , \tag{4}$$

where $\hat{\mathtt{Y}}_{ij,c_{ij}}$ is the predicted probability for the ground-truth class c_{ij} in pixel j of image i, and N_i is the number of pixels in each image. The total cross-entropy loss is the sum over both images: $\mathcal{L}_{\text{ce}} = \mathcal{L}_{\text{ce}_1} + \mathcal{L}_{\text{ce}_2}$.

This formulation offers several advantages over the cross-view-completion-based approach. First, it makes the pre-training effective in both covisible and non-covisible regions, as we explicitly model cases where pixels have no covisible region in the other view. Second, it provides interpretable outputs that directly reveal the model's understanding of scene geometry. Third, it better aligns with downstream tasks such as pose regression, where both images are fully visible.

3.3 Fine-Tuning for Relative Pose Regression

After pre-training, we fine-tune our model for the task of relative pose regression. We add a pose regression head on top of the pre-trained encoder-decoder architecture, while keeping the original covisibility segmentation head. This design allows the model to leverage the geometric understanding acquired during pre-training.

The pose regression head consists of several components as shown in Figure 2 (b). First, we apply global average pooling to the decoder outputs for both images:

$$\mathbf{f}_1 = \text{GlobalAvgPool}(\mathtt{O}_1), \quad \mathbf{f}_2 = \text{GlobalAvgPool}(\mathtt{O}_2)\ . \tag{5}$$

The pooled features are concatenated and processed through a shared MLP:

$$\mathbf{f}_{\text{shared}} = \text{MLP}([\mathbf{f}_1, \mathbf{f}_2])\ . \tag{6}$$

We then use separate heads for predicting the metric relative translation vector $\hat{\mathbf{t}} \in \mathbb{R}^3$ and the relative rotation represented as a quaternion $\hat{\mathbf{q}} \in \mathbb{R}^4$:

$$\hat{\mathbf{t}} = \text{MLP}_{\mathbf{t}}(\mathbf{f}_{\text{shared}}), \quad \hat{\mathbf{q}} = \text{MLP}_{\mathbf{q}}(\mathbf{f}_{\text{shared}})\ . \tag{7}$$

We use unit quaternions with L2 normalization, treating them as 4D vectors in the homoscedastic loss function.

Our fine-tuning stage consists of two phases:

1. First, we freeze the pre-trained encoder, decoder and covisibility segmentation head, and only train the pose regression head. During this phase, we use a homoscedastic loss [24] combining MSE losses for translation and quaternion predictions:

$$\mathcal{L}_{\text{pose}} = \frac{1}{2\sigma_{\mathbf{t}}^2}\|\mathbf{t} - \hat{\mathbf{t}}\|^2 + \frac{1}{2\sigma_{\mathbf{q}}^2}\|\mathbf{q} - \hat{\mathbf{q}}\|^2 + \log\sigma_{\mathbf{t}} + \log\sigma_{\mathbf{q}} \,, \tag{8}$$

   where $\mathbf{t}$ and $\mathbf{q}$ are the ground-truth metric translation and quaternion respectively, and $\sigma_{\mathbf{t}}$ and $\sigma_{\mathbf{q}}$ are learnable parameters that automatically balance the two loss terms.

2. In the second phase, we unfreeze the backbone and the covisibility segmentation head, and train the full network with a joint loss that combines the pose regression loss and the covisibility segmentation loss:

$$\mathcal{L}_{\text{joint}} = \mathcal{L}_{\text{pose}} + \frac{1}{2\sigma_{\text{seg}}^2}\mathcal{L}_{\text{ce}} + \log\sigma_{\text{seg}} \,, \tag{9}$$

   where σ_{seg} is a learnable parameter.

This training strategy ensures that the model maintains its interpretable covisibility segmentation capabilities while being optimized for the pose regression task.

4 Cub3: A Large-Scale Covisibility Dataset

To enable our covisibility segmentation approach, we introduce Cub3, a large-scale dataset comprising two sub datasets each of 2.5 million image pairs with dense covisibility annotations derived from both the autonomous driving nuScenes dataset [4], and the indoor ScanNet [8] dataset. Cub3 will be made publicly available.

4.1 Dataset Construction

nuScenes – We leverage the covisibility estimation pipeline introduced in RUBIK [28] to generate pixel-level covisibility annotations for image pairs from nuScenes. In brief, this pipeline uses monocular metric depth predictions from UniDepth [30] and surface normals from Depth Anything V2 [50], combined with camera poses from COLMAP [33] reconstructions, to automatically classify each pixel as either covisible, occluded, or outside FOV with respect to another image. We apply this pipeline to all possible image pairs within each scene, resulting in approximately 34 million annotated pairs.

ScanNet – The pipeline is similar but simpler for ScanNet, as we rely on the ground truth depths and camera poses to get all covisibility annotations.

For our experiments, we created two dataset variants, each containing 5M image pairs (2.5M from nuScenes and 2.5M from ScanNet):

- **Cub3-50:** Image pairs with at least 50% overlap, similar to the criterion used in CroCo [48]. This dataset provides a direct comparison point with existing cross-view completion approaches.
- **Cub3-all:** Image pairs with at least 5% overlap. This dataset, contains challenging image pairs to test the robustness of methods to handle low-overlap scenarios that are common in real-world applications.

Figure 3 shows examples or our covisibility annotations on image pairs from Cub3, illustrating how our process classifies pixels into the three categories across varying levels of difficulty.

It is also worth noting that our annotation process, especially on nuScenes, inherits some limitations from the underlying monocular depth estimation models. The depth predictions may occasionally struggle with challenging scenarios such as reflective surfaces, transparent objects, or regions with complex geometry or far-away objects. Consequently, some annotations in our dataset, particularly the distinction between covisible and occluded pixels, may contain noise. Despite these imperfections, our experiments show that the scale and diversity of Cub3 enable effective training of robust covisibility segmentation models.

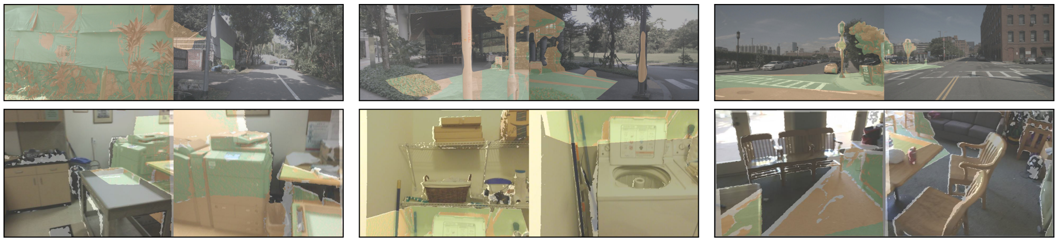

Figure 3: Covisibility annotation examples from Cub3 for nuScenes (top) and ScanNet (bottom). For each image pair, we show the corresponding covisibility maps with color-coding for covisible, occluded, and outside FOV regions. Note how our annotation process handles varying degrees of overlap and challenging viewpoint changes. Let us highlight that some annotations, particularly the distinction between covisible and occluded pixels, may contain noise, especially for nuScenes, and we demonstrate in the experiments that Alligat0R is highly robust to this noise.

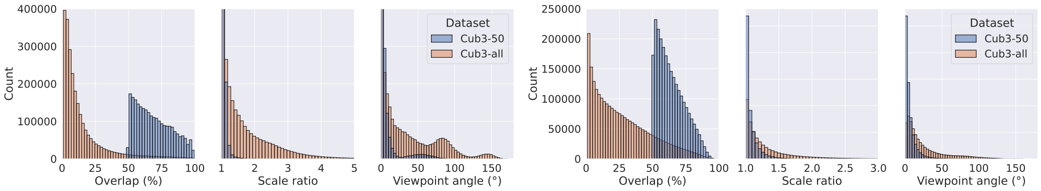

Figure 4: Distributions of overlap, scale ratio, and viewpoint angle in Cub3-all and Cub3-50 for nuScenes (left) and ScanNet (right).

4.2 Dataset Statistics

Figure 4 illustrates the distribution of our datasets across three geometric criteria introduced in RUBIK [28]: overlap percentage, scale ratio, and viewpoint angle. The Cub3-all dataset's distribution is heavily skewed towards very challenging pairs, which are particularly difficult for cross-completion methods like CroCo. In contrast, our covisibility segmentation approach is designed to benefit from such challenging cases effectively.

Given our computational resources, Cub3 is currently limited to either autonomous driving scenarios from nuScenes, or indoor scenes from ScanNet. But such scenarios have very important applications, and given its size, variety, and challenges, Cub3 is already sufficient to validate Alligat0R—very much like the initial version of CroCo [47] was demonstrated as a "proof-of-concept":

- **Scale:** With 5M annotated pairs, Cub3 provides sufficient data to train and validate our approach on both outdoor and indoor scenarios.
- **Geometric variety:** It covers a wide range of overlaps, scales and viewpoint angles within the driving and indoor contexts.
- **Challenging scenarios:** It includes pairs with minimal overlap that test the limits of CroCo, as was done in the original paper.

Our experiments on the RUBIK benchmark, which was specifically designed to evaluate challenging autonomous driving scenarios, as well as results on ScanNet1500 test set demonstrate the effectiveness of our approach. The promising results we obtain motivate future work to extend Cub3 to more diverse environments beyond urban driving scenes and indoor scenes. However, even within this specific domain, our method shows significant improvements over CroCo, particularly in geometrically challenging configurations.

5 Experiments

5.1 Implementation Details

We implement Alligat0R using PyTorch and conduct all experiments on NVIDIA A100 GPUs. The model architecture follows the design described in Section 3.1. The input images are classically resized to have a maximum dimension of 512 pixels while maintaining their aspect ratio. More information is provided in the supplementary material.

5.2 Main Results

Table 1: Results on RUBIK [28] and ScanNet1500 [8] for metric relative pose regression. For all experiments, fine-tuning is performed using Cub3-all.

			RUBIK			**ScanNet1500**		
Pre-training	Pre-Training Set	Backbone	5° / 0.5m	5° / 2m	10° / 5m	10° / 0.25m	10° / 0.5m	10° / 1m
CroCo	Cub3-50	❄	4.4	21.8	48.1	48.3	64.1	69.3
CroCo	Cub3-all	❄	2.4	8.9	25.2	8.7	18.3	24.9
Alligat0R	Cub3-50	❄	9.3	32.4	58.4	47.5	59.9	63.4
Alligat0R	Cub3-all	❄	24.0	55.3	**82.3**	78.2	90.1	92.5
CroCo	Cub3-50	🔥	12.4	38.3	66.7	75.7	87.4	91.5
CroCo	Cub3-all	🔥	12.5	38.6	66.2	64.1	79.1	85.0
Alligat0R	Cub3-50	🔥	21.2	53.3	78.1	82.8	91.7	93.9
Alligat0R	Cub3-all	🔥	**24.6**	**60.3**	81.9	**85.5**	**92.5**	**95.1**
From scratch	Cub3-all	🔥	10.8	29.1	52.3	37.0	51.8	57.2

Metric Relative Pose Regression Performance. We evaluate our proposed Alligat0R pre-training approach on the metric relative pose regression task and compare it with CroCo pre-training using the same architecture and fine-tuning protocol. Table 1 presents the results on the RUBIK [28] benchmark, which is specifically designed to evaluate performance across different geometric challenges in autonomous driving scenarios, and on the indoor ScanNet1500 [8] benchmark.

As shown in Table 1, Alligat0R consistently outperforms CroCo across different training and evaluation configurations. Notably, when pre-trained on Cub3-all and fine-tuned with the backbone frozen, Alligat0R achieves significantly higher performance and reaches state-of-the-art on the RUBIK benchmark (55.3% success rate at 5°/2m and 82.3% at 10°/5m) compared to the CroCo model pre-trained on the same dataset (which only achieves 8.9% and 25.2% respectively in the frozen backbone configuration). This substantial improvement demonstrates the effectiveness of our covisibility segmentation pre-training approach, especially for challenging scenarios with varying degrees of overlap.

Impact of Training Data Distribution. A key observation is the contrasting behavior of Alligat0R and CroCo when trained on different data distributions. While CroCo performs similarly or better when trained on easier pairs (Cub3-50) compared to harder pairs (Cub3-all), Alligat0R shows a different trend. Our method systematically improves when trained on the full distribution (Cub3-all). This demonstrates that Alligat0R can effectively learn from challenging examples, while CroCo struggles to benefit from them.

Figure 5 further illustrates this advantage by breaking down performance across different overlap percentages, scale ratios and viewpoint angles. While both methods perform well on high-overlap pairs (80-100%), Alligat0R trained on Cub3-all maintains strong performance even as overlap decreases, whereas CroCo's accuracy drops dramatically for pairs with less than 40% overlap. This demonstrates Alligat0R's ability to generalize across a wide range of geometric configurations, which is crucial for real-world applications where overlap cannot be guaranteed.

Fine-tuning Strategy Comparison. We also compare different fine-tuning strategies, specifically the impact of freezing versus unfreezing the backbone during fine-tuning. One can see that Alligat0R trained on Cub3-all is already quite high with a frozen backbone and always outperforms CroCo even when its backbone is unfrozen. This suggests that our pre-training strategy is better aligned with the downstream pose regression task.

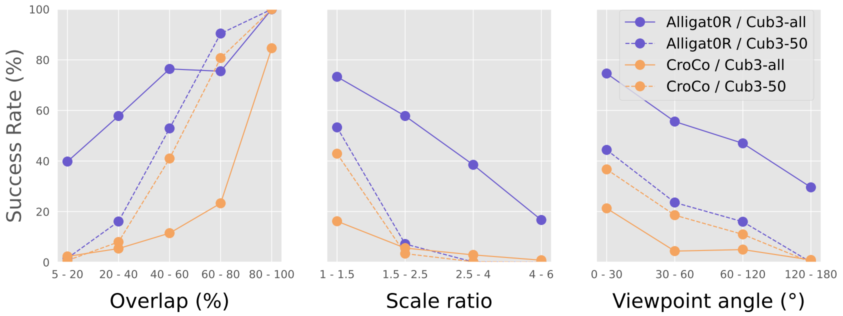

Figure 5: Performance of CroCo and Alligat0R across different geometric challenges on RUBIK. Results show accuracy at the 5°/2m threshold for models trained on different datasets (Cub3-50 or Cub3-all) and for frozen backbones. Alligat0R trained on Cub3-all consistently outperforms other configurations, particularly for challenging cases with low overlap, large scale differences, and extreme viewpoint changes.

Comparison with Training from Scratch. To isolate the impact of our pre-training approach, we compare the performance of a model trained from scratch for relative pose regression versus those initialized with Alligat0R or CroCo pre-training. The bottom row of Table 1 shows that training from scratch achieves notably lower performance, highlighting the value of the representations learned during pre-training.

Robustness to Label Noise. To address concerns about sensitivity to covisibility segmentation errors, we conducted a sensitivity analysis by training Alligat0R on ScanNet with 20% random label noise injected during pre-training. Surprisingly, this actually improved performance across all thresholds (+2.2%, +2.4%, +0.8% respectively), as shown in Table 2. This counterintuitive result suggests that label noise acts as a form of regularization similar to label smoothing [38], encouraging learning of more generalizable features. Importantly, this demonstrates that our approach remains robust even when covisibility annotations are imperfect.

Table 2: Sensitivity analysis: Impact of label noise on Alligat0R performance on ScanNet1500. Results show that 20% random label noise during pre-training actually improves performance, demonstrating robustness to annotation errors.

		ScanNet1500		
Method	Noise	10° / 0.25m	10° / 0.5m	10° / 1m
Alligat0R	0%	85.5	92.5	95.1
Alligat0R	20%	**87.7**	**94.9**	**95.9**
CroCo	N/A	75.7	87.4	91.5

Table 3: **RUBIK Benchmark.** The results are reported from [28]. Success rate (in %) for each method across individual geometric criterion bins. Best and second-best values for each column are shown in **bold** and underlined respectively.

	Overlap (%)					Scale Ratio				Viewpoint Angle (°)				Whole	Time
	80–100	60–80	40–60	20–40	5–20	1.0–1.5	1.5–2.5	2.5–4.0	4.0–6.0	0–30	30–60	60–120	120–180	Dataset	(ms)
Detector-free methods															
LoFTR [37]	87.2	88.4	47.2	17.5	5.0	51.6	10.1	2.3	0.6	43.2	27.9	15.1	0.0	24.9	185
ELoFTR [46]	56.4	90.3	50.8	22.1	6.3	51.2	15.6	4.4	0.7	42.2	30.8	18.2	0.1	26.6	124
ASpanFormer [6]	72.2	72.3	44.5	21.9	7.4	46.0	14.9	6.9	1.6	42.5	27.2	16.0	0.1	24.8	108
RoMa [13]	67.0	**98.3**	84.5	52.7	20.2	71.2	43.2	26.6	8.3	57.5	56.2	44.1	3.0	47.3	614
DUSt3R [45]	81.8	97.4	**90.8**	58.4	30.4	73.3	57.9	40.1	9.9	67.4	55.3	50.0	35.2	54.8	257
MASt3R [26]	52.0	97.5	89.6	61.0	28.4	71.2	52.3	42.5	13.8	53.5	**65.6**	**54.5**	14.1	53.6	173
Relative pose regression															
Alligat0R (Ours)	**100.0**	89.7	80.2	**61.5**	**44.3**	**78.4**	**61.3**	**43.9**	**23.2**	**77.5**	62.7	51.3	**37.3**	**60.3**	**57**

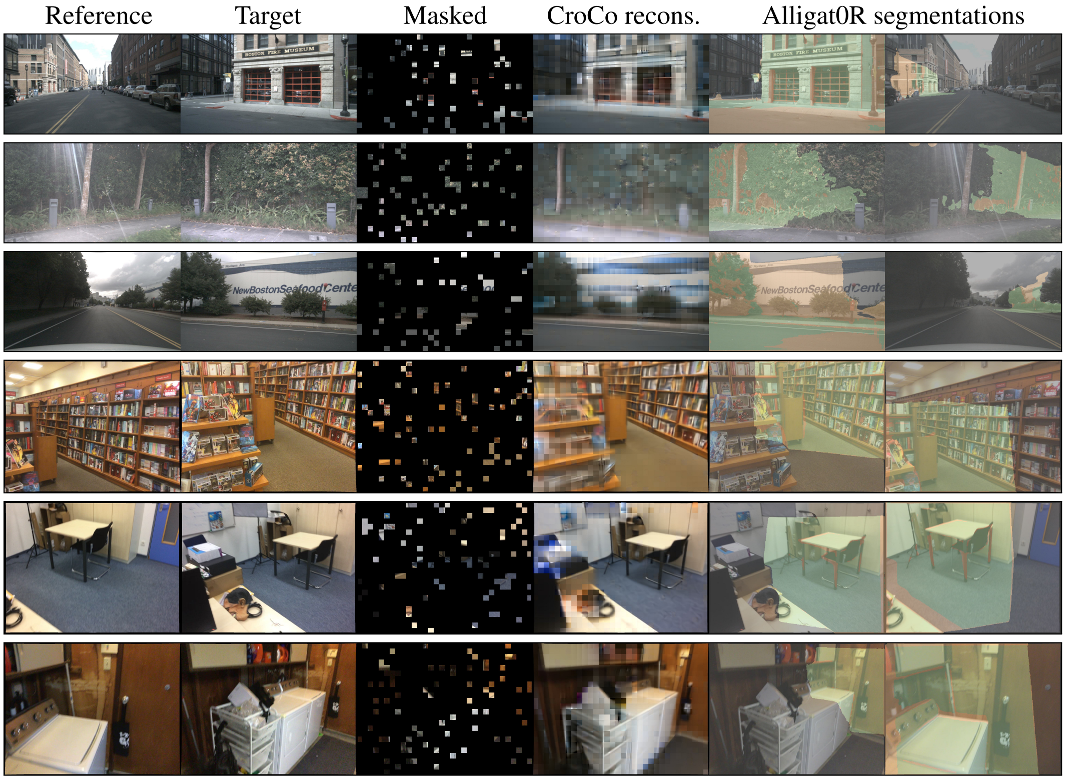

Figure 6: **Qualitative comparison.** CroCo's reconstructions are blurred in masked non-covisible regions, while Alligat0R often correctly identifies covisible, occluded, outside FOV regions across varying degrees of overlap and viewpoint changes.

5.3 Qualitative Analysis

Covisibility and Reconstruction Visualization. Figure 6 shows qualitative examples of the covisibility segmentation and CroCo reconstructions produced by our models on pairs from Cub3. Alligat0R successfully identifies covisible regions, occluded areas, and regions outside FOV across varying degrees of overlap and viewpoint changes. These visualizations provide insights into the geometric understanding acquired by our model. Conversely, while Croco succeeds in reconstructing masked covisible regions, the reconstructions are blurred in masked non-covisible regions as cross-completion is ill-posed in such areas.

5.4 Comparison with state-of-the-art on RUBIK

In Table 3, we evaluate the performance of Alligat0R fine-tuned for metric relative pose estimation against state-of-the-art methods on RUBIK [28]. Let us highlight that:

- Alligat0R was trained on nuScenes (on scenes that are not part of RUBIK),
- whereas none of the state-of-the-art methods used nuScenes as training set.

This gives an advantage to Alligat0R over the other methods. However, RUBIK allows the evaluation over large ranges of three geometric criteria—overlap, scale ratio and viewpoint angle—which we found interesting. Despite this advantage, we argue that Alligat0R's performance remains remarkable, as pretraining with CroCo, in contrast, leads to a significant performance drop (see Table 1).

With an overall success rate of 60.3%, Alligat0R ranks first. Notably, it significantly outperforms all other methods on the hardest bin across the three geometric criteria, which confirms the advantage of its pretraining on difficult image pairs. While Alligat0R's training set was highly skewed toward challenging pairs (see Figure 4), one might expect a performance drop on easier bins. However, this does not happen: for instance, Alligat0R still ranks first for small scale ratios and small viewpoint angles. Finally, Alligat0R is among the fastest methods, as it directly regresses the relative pose rather than producing intermediate correspondences.

5.5 Out-of-Domain Generalization

To demonstrate the generalization capabilities of our method beyond the training domains, we evaluate Alligat0R on zero-shot correspondence estimation using the ETH3D dataset [34]. We use the correlations from decoder features to estimate correspondences and measure performance using Average EndPoint Error (AEPE) as described in [1].

Zero-shot Correspondence Estimation. As detailed in the supplementary material, Alligat0R consistently outperforms CroCo when trained on the same datasets, and even surpasses CroCo v2 which was pre-trained on 5 datasets. This demonstrates that our covisibility segmentation pre-training learns more generalizable features for dense correspondence tasks.

Comparison with Official CroCo v2. We also compare against the official CroCo v2 model, fine-tuned on our Cub3-all dataset for pose regression. As shown in Table 4, Alligat0R significantly outperforms CroCo v2 across all metrics, demonstrating the effectiveness of our covisibility segmentation approach even when compared to a model pre-trained on multiple datasets.

Table 4: Comparison with official CroCo v2 fine-tuned on Cub3-all dataset for pose regression on RUBIK benchmark. Alligat0R significantly outperforms CroCo v2 across all metrics.

	RUBIK		
Method	5° / 0.5m	5° / 2m	10° / 5m
Alligat0R (Frozen)	**24.0**	**55.3**	**82.3**
CroCo v2 (Frozen)	5.8	15.8	34.1
Alligat0R (Unfrozen)	**24.6**	**60.3**	**81.9**
CroCo v2 (Unfrozen)	14.6	44.5	73.1

6 Limitations

While our approach demonstrates significant improvements over CroCo, it has several limitations:

- First, our current evaluation is limited to urban driving scenes from nuScenes and indoor scenes from ScanNet. The performance of Alligat0R on more diverse environments, such as natural scenes or extreme weather conditions, remains to be investigated.
- Second, our pre-training method requires ground truth depth maps and poses to produce dense covisibility annotations. While we investigated a fully automated pseudo ground truth generation for nuScenes, such a pre-processing is computationally intensive. This dependency on 3D data might limit the scalability of our approach to scenarios where such data is not readily available.
- Finally, while we show promising results on metric relative pose regression, the effectiveness of our pre-training approach for other binocular vision tasks, such as stereo matching or optical flow estimation, remains to be explored.

7 Conclusion and Future Work

We introduced Alligat0R, a novel pre-training approach that replaces cross-view completion with covisibility segmentation. Unlike CroCo's masked reconstruction, our approach explicitly models covisibility relationships, enabling more effective learning from challenging geometric configurations.

We demonstrated significant improvements over CroCo on metric relative pose regression and zero-shot correspondence estimation, particularly on challenging scenes with limited overlap. Our approach's robustness to geometrically challenging examples and superior generalization capabilities highlight the effectiveness of covisibility segmentation for binocular vision tasks.

We created Cub3, a large-scale dataset of 5M image pairs with dense covisibility annotations from nuScenes and ScanNet. Our work demonstrates that explicitly modeling covisibility provides an effective alternative to binocular vision tasks compared to reconstruction-based methods.

Acknowledgments

This project has received funding from the Bosch Research Foundation (Bosch Forschungsstiftung), the European Union (ERC Advanced Grant explorer Funding ID #101097259), and was granted access to the HPC resources of IDRIS under the allocation 2024-AD011014905R1 made by GENCI.

References

[1] 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 *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 1103–1115, 2025.

[2] Hangbo Bao, Li Dong, Songhao Piao, and Furu Wei. BEit: BERT pre-training of image transformers. In *International Conference on Learning Representations*, 2022.

[3] Guillaume Bono, Leonid Antsfeld, Boris Chidlovskii, Philippe Weinzaepfel, and Christian Wolf. End-to-end (instance)-image goal navigation through correspondence as an emergent phenomenon. *arXiv preprint arXiv:2309.16634*, 2023.

[4] Holger Caesar, Varun Bankiti, Alex H Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A multimodal dataset for autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11621–11631, 2020.

[5] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9650–9660, 2021.

[6] Hongkai Chen, Zixin Luo, Lei Zhou, Yurun Tian, Mingmin Zhen, Tian Fang, David Mckinnon, Yanghai Tsin, and Long Quan. Aspanformer: Detector-free image matching with adaptive span transformer. In *European Conference on Computer Vision*, pages 20–36. Springer, 2022.

[7] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PmLR, 2020.

[8] Angela Dai, Angel X Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5828–5839, 2017.

[9] Siyan Dong, Shuzhe Wang, Shaohui Liu, Lulu Cai, Qingnan Fan, Juho Kannala, and Yanchao Yang. Reloc3r: Large-scale training of relative camera pose regression for generalizable, fast, and accurate visual localization. *arXiv preprint arXiv:2412.08376*, 2024.

[10] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[11] Johan Edstedt, Ioannis Athanasiadis, Mårten Wadenbäck, and Michael Felsberg. Dkm: Dense kernelized feature matching for geometry estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17765–17775, 2023.

[12] Johan Edstedt, Georg Bökman, and Zhenjun Zhao. Dedode v2: Analyzing and improving the dedode keypoint detector. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4245–4253, 2024.

[13] Johan Edstedt, Qiyu Sun, Georg Bökman, Mårten Wadenbäck, and Michael Felsberg. Roma: Robust dense feature matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19790–19800, 2024.

[14] Sven Elflein, Qunjie Zhou, Sérgio Agostinho, and Laura Leal-Taixé. Light3r-sfm: Towards feed-forward structure-from-motion. *arXiv preprint arXiv:2501.14914*, 2025.

[15] Miao Fan, Mingrui Chen, Chen Hu, and Shuchang Zhou. Occˆ 2net: Robust image matching based on 3d occupancy estimation for occluded regions. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9652–9662, 2023.

[16] Hugo Germain, Vincent Lepetit, and Guillaume Bourmaud. Visual correspondence hallucination. In *International Conference on Learning Representations*, 2022.

[17] Pierre Gleize, Weiyao Wang, and Matt Feiszli. Silk: Simple learned keypoints. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 22499–22508, 2023.

[18] 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. *Advances in neural information processing systems*, 33:21271–21284, 2020.

[19] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16000–16009, 2022.

[20] Vlad Hondru, Florinel Alin Croitoru, Shervin Minaee, Radu Tudor Ionescu, and Nicu Sebe. Masked image modeling: A survey. *arXiv preprint arXiv:2408.06687*, 2024.

[21] Will Hutchcroft, Yuguang Li, Ivaylo Boyadzhiev, Zhiqiang Wan, Haiyan Wang, and Sing Bing Kang. Covispose: Co-visibility pose transformer for wide-baseline relative pose estimation in 360° indoor panoramas. In *European Conference on Computer Vision*, pages 615–633. Springer, 2022.

[22] Wei Jiang, Eduard Trulls, Jan Hosang, Andrea Tagliasacchi, and Kwang Moo Yi. Cotr: Correspondence transformer for matching across images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6207–6217, 2021.

[23] Linyi Jin, Richard Tucker, Zhengqi Li, David Fouhey, Noah Snavely, and Aleksander Holynski. Stereo4d: Learning how things move in 3d from internet stereo videos. *arXiv preprint arXiv:2412.09621*, 2024.

[24] Alex Kendall, Yarin Gal, and Roberto Cipolla. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7482–7491, 2018.

[25] Shinjeong Kim, Marc Pollefeys, and Daniel Barath. Learning to make keypoints sub-pixel accurate. In *European Conference on Computer Vision*, pages 413–431. Springer, 2025.

[26] Vincent Leroy, Yohann Cabon, and Jérôme Revaud. Grounding image matching in 3d with mast3r. *arXiv preprint arXiv:2406.09756*, 2024.

[27] Philipp Lindenberger, Paul-Edouard Sarlin, and Marc Pollefeys. Lightglue: Local feature matching at light speed. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 17627–17638, 2023.

[28] Thibaut Loiseau and Guillaume Bourmaud. Rubik: A structured benchmark for image matching across geometric challenges. *arXiv preprint arXiv:2502.19955*, 2025.

[29] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *Transactions on Machine Learning Research Journal*, pages 1–31, 2024.

[30] Luigi Piccinelli, Yung-Hsu Yang, Christos Sakaridis, Mattia Segu, Siyuan Li, Luc Van Gool, and Fisher Yu. Unidepth: Universal monocular metric depth estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10106–10116, 2024.

[31] Guilherme Potje, Felipe Cadar, André Araujo, Renato Martins, and Erickson R Nascimento. Xfeat: Accelerated features for lightweight image matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2682–2691, 2024.

[32] Paul-Edouard Sarlin, Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superglue: Learning feature matching with graph neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4938–4947, 2020.

[33] Johannes L Schonberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4104–4113, 2016.

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

[35] Brandon Smart, Chuanxia Zheng, Iro Laina, and Victor Adrian Prisacariu. Splatt3r: Zero-shot gaussian splatting from uncalibrated image pairs. *arXiv preprint arXiv:2408.13912*, 2024.

[36] Zikai Song, Run Luo, Junqing Yu, Yi-Ping Phoebe Chen, and Wei Yang. Compact transformer tracker with correlative masked modeling. In *Proceedings of the AAAI conference on artificial intelligence*, pages 2321–2329, 2023.

[37] Jiaming Sun, Zehong Shen, Yuang Wang, Hujun Bao, and Xiaowei Zhou. Loftr: Detector-free local feature matching with transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8922–8931, 2021.

[38] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826, 2016.

[39] Dongli Tan, Jiang-Jiang Liu, Xingyu Chen, Chao Chen, Ruixin Zhang, Yunhang Shen, Shouhong Ding, and Rongrong Ji. Eco-tr: Efficient correspondences finding via coarse-to-fine refinement. In *European Conference on Computer Vision*, pages 317–334. Springer, 2022.

[40] Shitao Tang, Jiahui Zhang, Siyu Zhu, and Ping Tan. Quadtree attention for vision transformers. In *International Conference on Learning Representations*, 2022.

[41] Prune Truong, Martin Danelljan, Radu Timofte, and Luc Van Gool. Pdc-net+: Enhanced probabilistic dense correspondence network. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(8):10247–10266, 2023.

[42] Hengyi Wang and Lourdes Agapito. 3d reconstruction with spatial memory. *arXiv preprint arXiv:2408.16061*, 2024.

[43] Qing Wang, Jiaming Zhang, Kailun Yang, Kunyu Peng, and Rainer Stiefelhagen. Matchformer: Interleaving attention in transformers for feature matching. In *Proceedings of the Asian Conference on Computer Vision*, pages 2746–2762, 2022.

[44] Qianqian Wang, Yifei Zhang, Aleksander Holynski, Alexei A Efros, and Angjoo Kanazawa. Continuous 3d perception model with persistent state. *arXiv preprint arXiv:2501.12387*, 2025.

[45] Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jerome Revaud. Dust3r: Geometric 3d vision made easy. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20697–20709, 2024.

[46] Yifan Wang, Xingyi He, Sida Peng, Dongli Tan, and Xiaowei Zhou. Efficient loftr: Semi-dense local feature matching with sparse-like speed. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21666–21675, 2024.

[47] Philippe Weinzaepfel, Vincent Leroy, Thomas Lucas, Romain Brégier, Yohann Cabon, Vaibhav Arora, Leonid Antsfeld, Boris Chidlovskii, Gabriela Csurka, and Jérôme Revaud. Croco: Self-supervised pre-training for 3d vision tasks by cross-view completion. *Advances in Neural Information Processing Systems*, 35:3502–3516, 2022.

[48] Philippe Weinzaepfel, Thomas Lucas, Vincent Leroy, Yohann Cabon, Vaibhav Arora, Romain Brégier, Gabriela Csurka, Leonid Antsfeld, Boris Chidlovskii, and Jérôme Revaud. Croco v2: Improved cross-view completion pre-training for stereo matching and optical flow. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 17969–17980, 2023.

[49] 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. *arXiv preprint arXiv:2501.13928*, 2025.

[50] Lihe Yang, Bingyi Kang, Zilong Huang, Zhen Zhao, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything v2. *arXiv preprint arXiv:2406.09414*, 2024.

[51] Botao Ye, Sifei Liu, Haofei Xu, Xueting Li, Marc Pollefeys, Ming-Hsuan Yang, and Songyou Peng. No pose, no problem: Surprisingly simple 3d gaussian splats from sparse unposed images. *arXiv preprint arXiv:2410.24207*, 2024.

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

[53] Haojie Zhao, Dong Wang, and Huchuan Lu. Representation learning for visual object tracking by masked appearance transfer. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 18696–18705, 2023.

[54] Xiaoming Zhao, Xingming Wu, Weihai Chen, Peter CY Chen, Qingsong Xu, and Zhengguo Li. Aliked: A lighter keypoint and descriptor extraction network via deformable transformation. *IEEE Transactions on Instrumentation and Measurement*, 72:1–16, 2023.

[55] Jinghao Zhou, Chen Wei, Huiyu Wang, Wei Shen, Cihang Xie, Alan Yuille, and Tao Kong. Image BERT pre-training with online tokenizer. In *International Conference on Learning Representations*, 2022.

[56] Shengjie Zhu and Xiaoming Liu. Pmatch: Paired masked image modeling for dense geometric matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21909–21918, 2023.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The paper clearly states its main contribution - a novel pretraining method for binocular vision that explicitly segments pixels as covisible, occluded, or outside field-of-view, overcoming limitations of previous methods like CroCo. The claims are supported by experimental results in Section 5.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The paper acknowledges limitations in Section 6, discussing the scope of the method and potential areas for improvement.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [NA]

Justification: The paper is primarily focused on empirical results and does not contain theoretical proofs or theorems.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: Section 5 (Experiments) provides detailed information about the experimental setup, including dataset details, training procedures, and evaluation metrics. Further details will also be in the supplementary material, and code and data will be made publicly available.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in

some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: The paper explicitly mentions code and data will be released in the abstract.

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: Section 5 provides comprehensive details about the experimental setup, including training parameters, data splits, and evaluation protocols.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: Error bars are not reported in the paper because it would require additional compute resources.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The paper provides detailed information about the experimental setup, including training parameters, data splits, and evaluation protocols, as well as the compute resources needed to reproduce the experiments.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research focuses on computer vision and pose estimation, with no apparent ethical concerns or violations of the NeurIPS Code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [No]

    Justification: The paper focuses on technical contributions to computer vision and does not explicitly discuss societal impacts.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: The paper does not involve high-risk models or datasets that would require specific safeguards.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: The paper properly cites and credits all referenced works, including datasets like nuScenes and ScanNet, and methods like CroCo. We will add the licences of used assets in the supplementary material.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.

- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: The paper explicitly mentions the release of new assets (code, datasets, or models).

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: The paper does not involve any crowdsourcing or human subject research.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: The paper does not involve any human subject research requiring IRB approval.

    Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: [NA]

    Justification: The paper does not use LLMs as a component of its core methodology.

    Guidelines:

    - The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.

A Supplementary Material

This supplementary material provides additional details about our Alligat0R model, including training curves, implementation details, and visualizations that complement the main paper.

A.1 Pre-training Learning Curves

Figure 7 shows the learning curves for the pre-training phase of both CroCo and Alligat0R on the Cub3-50 and Cub3-all datasets. Alligat0R's loss converges smoothly on both datasets, indicating stable training.

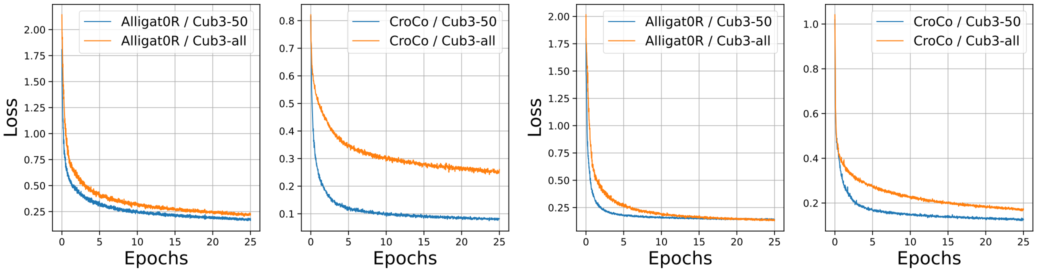

Figure 7: Learning curves during pre-training for CroCo (reconstruction loss) and Alligat0R (segmentation loss) on Cub3-50 and Cub3-all datasets for nuScenes (left) and ScanNet (right). Alligat0R shows stable convergence on both datasets.

A.2 Fine-tuning Learning Curves

Figure 8 presents the fine-tuning curves for the pose regression task. The plots show the full loss during training (Eq. 9 in the main paper for Alligat0R, Eq. 8 for CroCo). Alligat0R fine-tuned on Cub3-all shows faster convergence and reaches higher accuracy than the same configuration for CroCo, highlighting the transferability of features learned through covisibility segmentation on difficult pairs. The loss increase at 5 epochs corresponds to unfreezing the backbone. For Alligat0R, at 5 epochs we also add the segmentation loss to maintain interpretability.

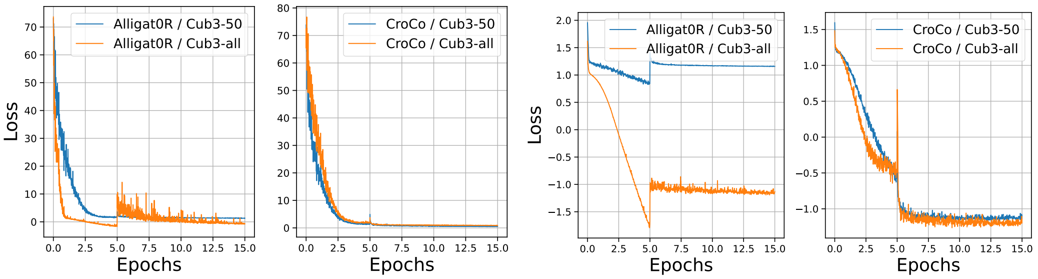

Figure 8: Learning curves during fine-tuning for pose regression for nuScenes (left) and ScanNet (right). Alligat0R pre-trained on Cub3-all converges faster and achieves higher success rates than other methods, demonstrating the effectiveness of our covisibility segmentation pre-training approach.

A.3 Detailed Performance Analysis

Figure 9 provides a comprehensive breakdown of performance across different geometric criteria on RUBIK. This visualization emphasizes Alligat0R's strong performance across all difficulty ranges, particularly in challenging scenarios where traditional methods struggle.

A.4 Implementation Details

We use a ViT-based encoder and transformer decoder backbone similar to CroCo, with 24 layers for the encoder and 12 for the decoder. For pre-training, we use the AdamW optimizer with a learning

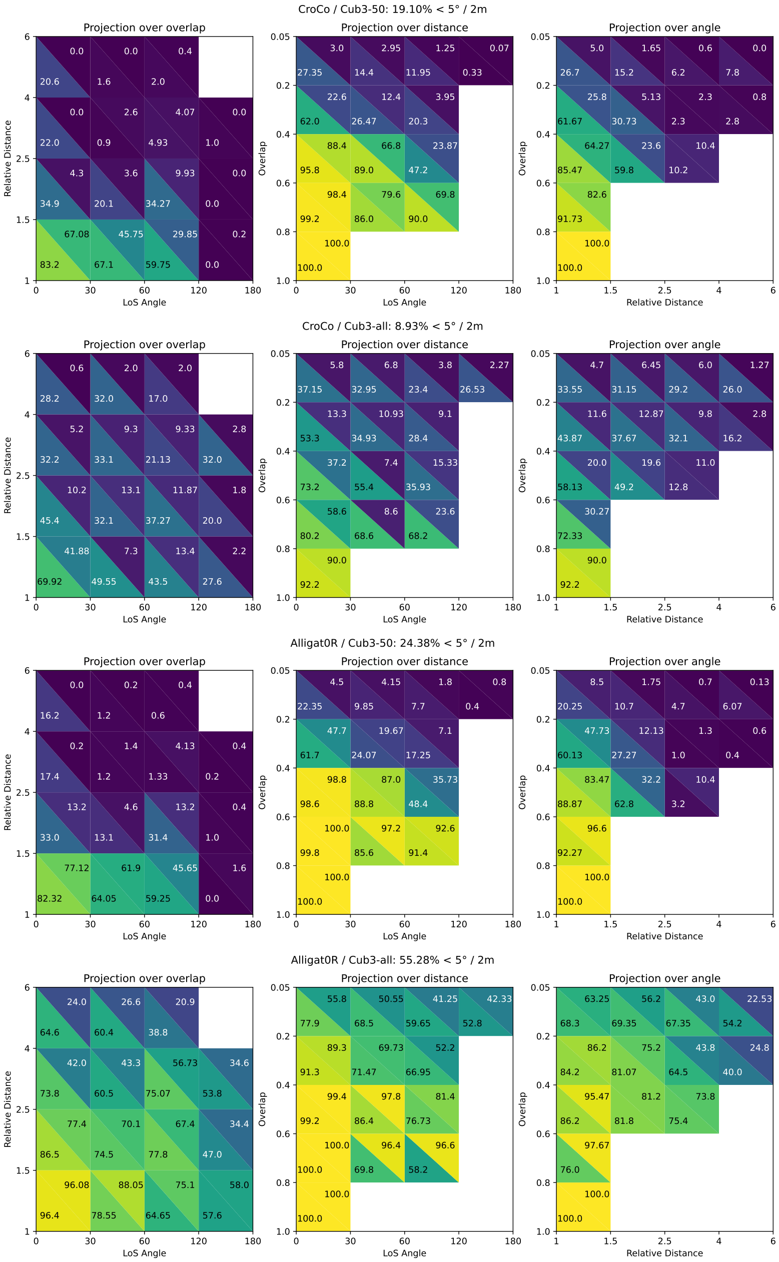

Figure 9: **Detailed breakdown of performance across different geometric criteria.** Success rate either for R@5° or t@2m (bottom-left and top-right of each triangle, respectively), when projecting results onto individual geometric criteria of RUBIK. For each method, we show three plots corresponding to the projection over overlap (left), scale ratio (middle), and viewpoint angle (right).

rate of 1.5e-4, weight decay of 0.05, and a batch size of 32 per GPU. We employ a cosine learning rate schedule with 2 epochs of warmup and train for 25 epochs on our Cub3 datasets.

For fine-tuning on relative pose regression, we follow the two-phase approach described in Section 3.3 of the main paper:

- Phase 1: Freeze the backbone and train only the pose regression head for 5 epochs with a learning rate of 1e-4

- Phase 2: Unfreeze the entire network and jointly train with both the pose regression loss and covisibility segmentation loss for an additional 10 epochs with a learning rate of 5e-5

The comprehensive architecture of Alligat0R is illustrated in Figure 2 of the main paper, showing both the pre-training and fine-tuning phases. During pre-training, the model learns to segment pixels in each view as covisible, occluded, or outside the field of view with respect to the other view. During fine-tuning, we introduce a pose regression head while maintaining the segmentation capability to leverage the geometric understanding acquired during pre-training.

A.5 Data Splits and Sampling Protocol

To ensure proper evaluation and avoid data leakage, we carefully separate training and test data:

Data Splits:

- **RUBIK benchmark:** Uses test scenes from nuScenes, while our Cub3 dataset uses only the training split of nuScenes, with completely disjoint scenes.
- **ScanNet1500 benchmark:** Derived from the standard ScanNet test split, whereas Cub3 uses scenes from the ScanNet training split exclusively.

Sampling Protocol: We pre-filter all possible pairs from all training scenes with at least 5% overlap for the "all" version of Cub3, and at least 50% for the "50" version. We then sample from all those pre-filtered pairs to extract the desired number of samples and covisibility masks. The RUBIK benchmark creation follows the protocol described in [28] and is extracted from test scenes, while ScanNet1500 was created from test scenes of ScanNet as first introduced in [32].

A.6 Additional Ablation Studies

A.6.1 Impact of the Number of Classes

We implemented two variants of Alligat0R to investigate the importance of our pre-training with the three classes (covisible, occluded, outside FOV). We tried pre-training with only two classes, either covisible or not (in this case occluded and outside FOV are merged), or inside FOV or not (in this case, covisible and occluded are merged) on Cub3-all for nuScenes. The results on the RUBIK benchmark are presented in Table 5.

Table 5: Results on RUBIK for **metric** relative pose regression when pre-training with only two classes. For all experiments, pre-training and fine-tuning is performed using Cub3-all.

	RUBIK		
Classes	5° / 0.5m	5° / 2m	10° / 5m
Covisible or not	22.5	55.7	80.0
Inside FOV or not	**24.6**	59.6	**81.9**
All 3 classes	**24.6**	**60.3**	**81.9**

While the performance improvement with three classes is modest, we believe that the model's knowledge about occluded regions could be beneficial for other tasks. As noted in the main paper, the annotated maps may contain noise between covisible and occluded zones due to reliance on monocular depth predictions.

A.6.2 Non-metric Relative Pose Regression

We investigated whether our pre-training is useful for non-metric relative pose regression (regressing only the angle for translation) and compared our results with CroCo pre-training on the ScanNet1500 benchmark. For this experiment, we changed our pose regression head using the one from Reloc3r, along with its loss function. The results are shown in Table 6.

Alligat0R significantly outperforms CroCo on this task, demonstrating the versatility of our pre-training approach.

Table 6: Results on ScanNet1500 for **non-metric** relative pose regression. For Alligat0R, pre-training and fine-tuning is performed using Cub3-all, whereas for CroCo, pre-training is performed using Cub3-50 and fine-tuning using Cub3-all.

	ScanNet1500		
Pre-Training	AUC@5	AUC@10	AUC@20
CroCo	13.2	34.1	57.1
Alligat0R	**20.5**	**43.9**	**66.2**

A.7 Zero-shot Correspondence Estimation on ETH3D

To demonstrate the generalization capabilities of our method beyond the training domains, we evaluate Alligat0R on zero-shot correspondence estimation using the ETH3D dataset [34]. We use the correlations from decoder features to estimate correspondences and measure performance using Average EndPoint Error (AEPE) as described in [1].

Table 7: Zero-shot correspondence estimation on ETH3D dataset using AEPE. Alligat0R consistently outperforms CroCo variants, demonstrating better generalization to out-of-domain dense matching tasks.

Method	Training Data	AEPE (↓)
Alligat0R	nuScenes-all	**43.82**
CroCo	nuScenes-all	77.61
Alligat0R	nuScenes-50	**44.12**
CroCo	nuScenes-50	92.98
Alligat0R	ScanNet-all	**36.07**
CroCo	ScanNet-all	56.70
Alligat0R	ScanNet-50	**38.45**
CroCo	ScanNet-50	86.60
CroCo v2	5 datasets	51.55

The results in Table 7 show that Alligat0R consistently outperforms CroCo when trained on the same datasets, and even surpasses CroCo v2 which was pre-trained on 5 datasets. This demonstrates that our covisibility segmentation pre-training learns more generalizable features for dense correspondence tasks.

A.8 Additional Qualitative Results

We provide additional visualizations from our nuScenes and ScanNet datasets in Figures 10 and 11, along with more predictions from Alligat0R and CroCo in Figure 12.

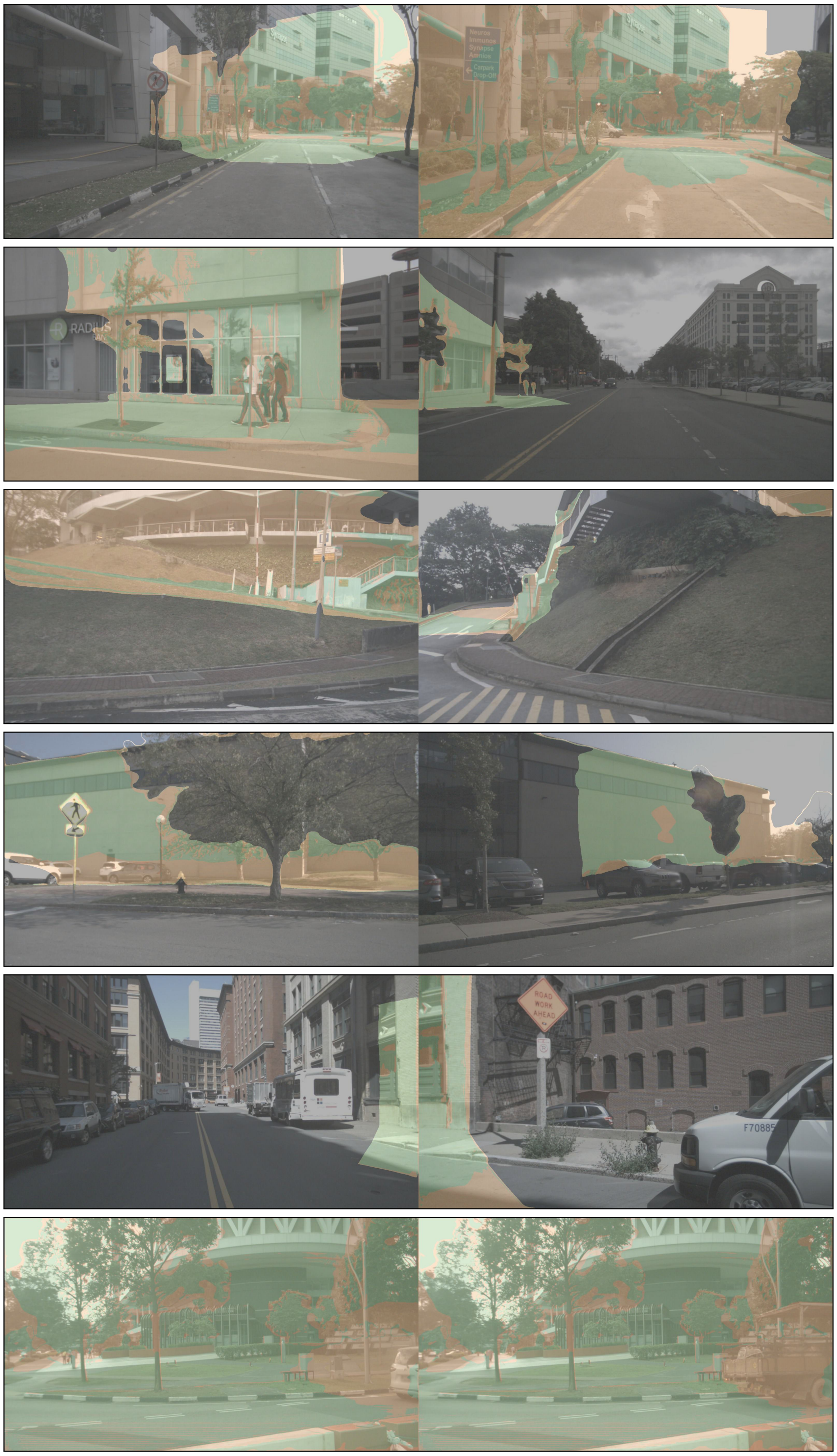

Figure 10: Covisibility annotation examples from Cub3 for nuScenes. For each image pair, we show the corresponding covisibility maps with color-coding for covisible, occluded, and outside FOV regions. Note how our annotation process handles varying degrees of overlap and challenging viewpoint changes. Let us highlight that some annotations, particularly the distinction between covisible and occluded pixels, may contain noise, especially for nuScenes, and we demonstrate in the experiments that Alligat0R is highly robust to this noise.

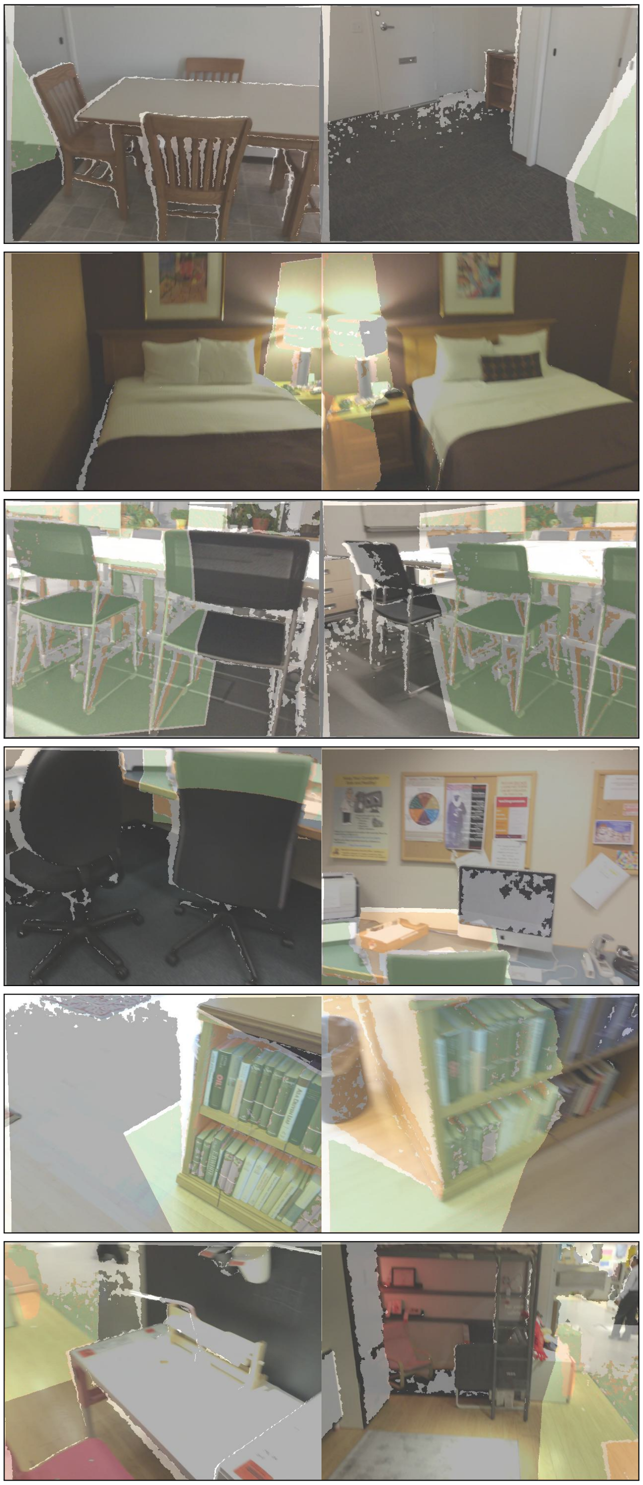

Figure 11: Covisibility annotation examples from Cub3 for ScanNet. For each image pair, we show the corresponding covisibility maps with color-coding for covisible, occluded, and outside FOV regions. Note how our annotation process handles varying degrees of overlap and challenging viewpoint changes.

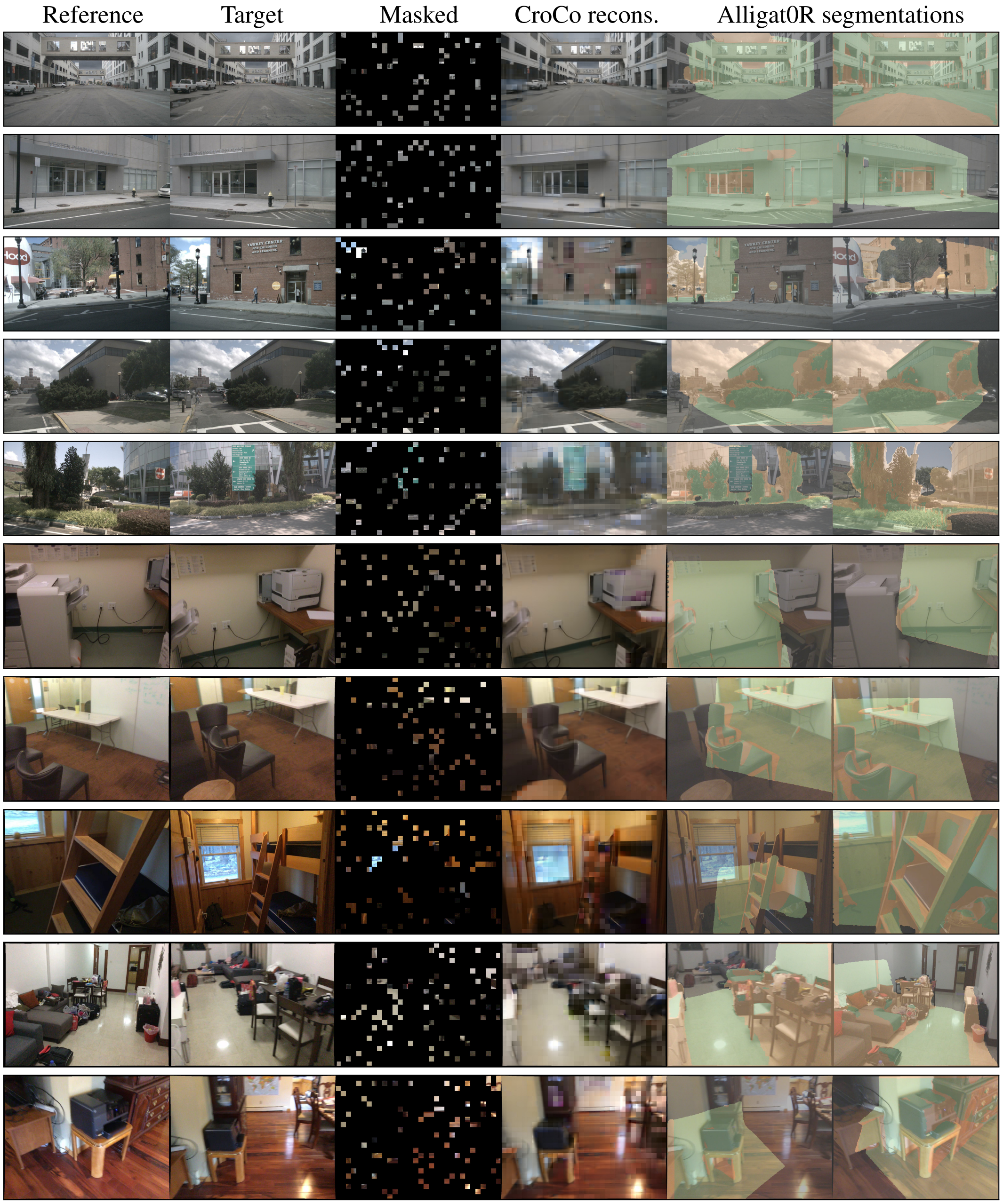

Figure 12: **Qualitative comparison.** CroCo's reconstructions are blurred in masked non-covisible regions, while Alligat0R often correctly identifies covisible, occluded, outside FOV regions across varying degrees of overlap and viewpoint changes.