ReMov3r: Real-Time Monocular Video to 3D reconstruction

Yash Jangir, Karan Mirakhor, Tanya Choudhary
Carnegie Mellon University

Abstract

In this project, we propose ReMov3r, an end-to-end architecture for dense 3D scene reconstruction from monocular RGB video with known camera intrinsics. Our method integrates three key innovations: explicit multi-modal feature fusion using cross-attention between derived geometric and visual features, a hierarchical state representation that separates local temporal coherence from global spatial consistency, and a confidence-aware depth refinement process. By leveraging pretrained depth and image encoders, ReMov3r generates semantically-conditioned feature embeddings and refines depth estimates for accurate pointmap prediction. An adaptive keyframe selection strategy ensures computational efficiency while maintaining reconstruction quality. Experiments show that ReMov3r delivers robust, scalable, and accurate 3D reconstruction performance on par with existing methods. We believe that with future work—particularly scaling up training and incorporating improved loss functions—ReMov3r has the potential to surpass state-of-the-art approaches.

1 Introduction

3D scene reconstruction from monocular RGB input represents a fundamental challenge in computer vision with applications spanning robotics, augmented reality, and autonomous navigation. Despite significant progress in this domain, existing methods face persistent challenges in maintaining spatio-temporal consistency, handling geometric ambiguities, and achieving real-time performance without sacrificing reconstruction quality. These challenges become particularly acute when working with uncalibrated cameras, dynamic scenes, or extended video sequences where traditional optimization-heavy approaches may fail or become computationally prohibitive.

Recent transformer-based architectures have demonstrated promising results in dense monocular 3D reconstruction. DUSt3R[1] introduced a unified framework for regressing dense per-pixel 3D pointmaps from RGB images without requiring camera intrinsics at inference time. However, it lacks temporal modeling capabilities and generates pointmaps in local coordinate systems that require costly post-processing alignment. CUT3R[2] extended this work by introducing a state-recurrent transformer architecture for continuous 3D perception, maintaining a persistent latent state for online prediction of dense pointmaps and camera parameters in a shared world coordinate system. While effective, these approaches still struggle with balancing computational efficiency and reconstruction accuracy, particularly in unconstrained real-world settings.

Alternative approaches have explored different extensions to address these limitations. Align3R[3] explicitly incorporates monocular depth priors to enhance geometric reasoning, but requires significant computational overhead for global optimization. SLAM3R[4] prioritizes real-time, calibration-free reconstruction through a two-tier pipeline with Image-to-Points and Local-to-World modules, offering improved efficiency but with potential trade-offs in reconstruction detail. Other specialized extensions like MASt3R[5], MonST3R[6], and Spann3R[7] have contributed specific innovations but lack comprehensive system-level integration across the efficiency-quality spectrum.

We propose ReMov3r, a novel approach that synergistically combines three key innovations to address these challenges. First, we introduce explicit multi-modal feature fusion that combines geometric and visual features through cross-attention, creating Semantically-conditioned Feature (SCF) embeddings that capture both spatial relationships and semantic context. Second, our hierarchical state representation separates local temporal coherence from global spatial consistency, enabling efficient tracking of camera motion and scene structure across varying temporal scales. Third, we implement a confidence-aware depth refinement process that optimizes initial depth predictions through learned error correction, enhancing the accuracy of geometric reconstructions even in challenging scenarios.

ReMov3r distinguishes itself from prior work through its comprehensive approach to spatio-temporal consistency. Our adaptive keyframe selection strategy balances reconstruction accuracy with computational efficiency, while the dual-level state representation enables both short-term coherence between consecutive frames and long-term consistency across the entire sequence. By incorporating pretrained monocular depth estimation models like Depth Anything v2[8]and visual feature extractors such as DINOv2[9], our architecture leverages powerful foundation models while addressing their inherent limitations through our refinement pipeline.

2 Related Work

The task of dense monocular 3D reconstruction has evolved significantly through the development of transformer-based architectures that regress dense geometry without explicit depth sensors or multi-view calibration. A foundational model in this direction is DUSt3R [1], which introduced a unified framework for monocular and binocular reconstruction by regressing dense per-pixel 3D pointmaps in a learned coordinate frame. Using a Vision Transformer (ViT) [10] encoder and a transformer decoder, DUSt3R learns 2D-3D correspondences from RGB images without requiring camera intrinsics at inference. It supports downstream recovery of camera parameters via optimization over predicted geometry. However, DUSt3R is limited to static scenes and generates pointmaps in local coordinate systems, which must be globally aligned through a costly post-processing step. It also lacks temporal modeling or memory, which restricts its scalability in long-term or video-based settings.

To address these limitations, several extensions have emerged. CUT3R [2] builds on DUSt3R by introducing a state-recurrent transformer architecture designed for continuous 3D perception. It maintains a persistent latent state that encodes the evolving scene as new frames arrive. Each incoming image both updates and queries this state via dual transformer decoders, enabling online prediction of dense pointmaps and camera parameters in a shared world coordinate system. CUT3R supports both dense reconstruction and scene completion: by querying the state with virtual raymaps, it can infer unobserved or occluded geometry, making it particularly effective for handling dynamic or partially observed scenes. Unlike optimization-heavy methods like DUSt3R and Align3R [3], CUT3R is fully feed-forward and online, achieving real-time performance without requiring camera intrinsics or post-hoc alignment.

Align3R takes a complementary path by explicitly injecting monocular depth priors into DUSt3R's geometric reasoning pipeline. It uses pretrained monocular depth estimators (such as Depth Anything V2 [8]) to generate per-frame depth maps, which are unprojected into 3D pointmaps and encoded using a dedicated ViT. These encoded features are fused into DUSt3R's decoder via zero convolutions, preserving model stability while enhancing geometric accuracy. This conditioning resolves ambiguities in textureless or occluded regions and anchors DUSt3R's predictions at metric scale. Align3R is particularly effective for dynamic scenes, where depth priors help stabilize frame-to-frame variation. After prediction, a global optimization step refines depth and camera pose estimates across frames. Though accurate, this step incurs significant computational overhead and limits real-time viability.

SLAM3R [4] addresses this challenge directly by prioritizing real-time, calibration-free 3D scene reconstruction. It introduces a novel two-tier pipeline consisting of an Image-to-Points (I2P) module and a Local-to-World (L2W) module. The I2P module processes sliding windows of RGB frames to regress dense 3D pointmaps in a local frame, selecting a keyframe (typically the middle frame) to serve as a reference coordinate system. Using a multi-branch ViT-based encoder-decoder architecture, SLAM3R encodes each frame independently and fuses features via cross-attention and aggregation layers, enabling dense multi-view reasoning without solving for camera poses.

The L2W module then incrementally registers these local reconstructions into a shared global scene model. Unlike traditional SLAM [11] or learned-SLAM [12] or SfM [13] [14] methods, SLAM3R does not recover or optimize for extrinsic transformations. Instead, it maintains a reservoir of previously reconstructed "scene frames", from which it retrieves the top-K most relevant keyframes using a learned similarity metric. Both appearance and geometry are embedded into the retrieval process using ViT-encoded tokens and patch embeddings. The new keyframe's features are aligned against retrieved scene frames via transformer-based decoding, and its pointmap is fused into the global point cloud through deformation and confidence-weighted integration. This alignment process is entirely feed-forward and

does not rely on rigid-body assumptions, making SLAM3R highly robust to drift while maintaining 20+ FPS throughput.

By avoiding camera parameter estimation and relying solely on learned visual-geometric correspondences, SLAM3R is uniquely suited for robotics, augmented reality, and any scenario where calibration data is unavailable or dynamic scenes are common. Its fully end-to-end, optimization-free architecture makes it significantly faster than methods like Align3R and more scalable than memory-heavy models like Spann3R [7].

Other works in the 3R family contribute specific innovations but do not offer the same level of system-level integration. MASt3R [5] improves matching precision via a Dense Feature Head, Fast Reciprocal Matching, and coarse-to-fine refinement, enabling high-resolution reconstructions and improved robustness. **MonST3R** [6] adapts DUSt3R to dynamic scenes by regressing global pointmaps per timestep, handling motion without explicit priors. **Spann3R**, similar to CUT3R, introduces a spatial memory and a geometry-aware attention mechanism to maintain scene consistency in online settings. While these methods offer valuable insights, SLAM3R distinguishes itself by offering the best trade-off between efficiency, scalability, and reconstruction quality in calibration-free real-time environments.

3 Methodology

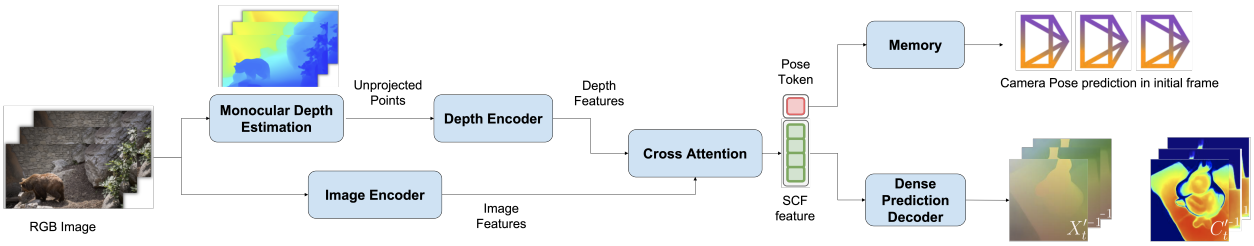

Figure 1: ReMov3r fuses depth and image features from monocular RGB video via cross-attention to form semantically-conditioned embeddings, refines depth and pointmaps with confidence prediction, and maintains hierarchical local and global states for efficient, consistent 3D reconstruction using adaptive keyframe selection and pose estimation.

We propose an end-to-end architecture for 3D scene reconstruction from monocular RGB video sequences with known camera intrinsics, featuring three key ideas: (i) Explicit Multi-modal Feature Fusion, (ii) Hierarchical State Representation for efficient and effective spatio-temporal consistency and (iii) Refining Depth Estimation using a confidence-aware refinement process.

3.1 Model Architecture

Our method, illustrated in fig. 1, is built on three interconnected components: (1) early fusion of geometric and visual features via cross-attention to produce Semantically-Conditioned Feature (SCF) embeddings, (2) a hierarchical state representation that disentangles local temporal coherence from global spatial consistency, and (3) a confidence-aware depth refinement module that enhances initial depth estimates through learned error correction. These elements work together to enable accurate and efficient 3D reconstruction, supported by adaptive keyframe selection and a hierarchical attention framework.

3.1.1 Explicit Multimodal Feature Fusion

Geometric initialization employs a pretrained monocular depth prediction model Depth Anything v2[8] to generate frame-wise pointmaps through unprojection of depth estimates into 3D space. The pointmaps are processed using a ViT-based[10] depth encoder module to create depth features. Simultaneously, the visual stream extracts global image-level features using DINOv2[9]. The fusion occurs at the point cloud level through cross-attention such that 3D pointmaps are cross-attended by their corresponding visual features. We call these contexualized pointmaps as Semantically-Conditioned Feature (SCF) embeddings that capture both geometric relationships and semantic context,

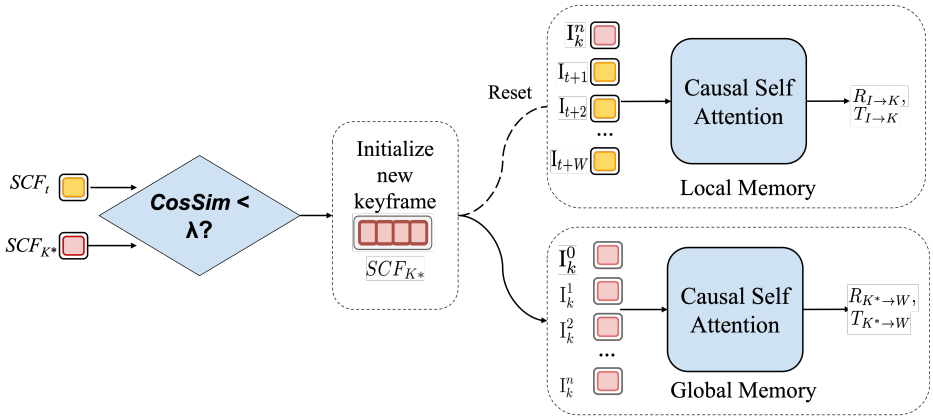

Figure 2: Memory Module with hierarchical state representation

enabling the model to learn spatial relationships conditioned on visual semantics. For global pose estimation we also add a pose token corresponding to each frame to fusion block to create pose embedding. This token is self-attended by depth features and cross-attended by visual features, capturing the overall semantics of the frame.

3.1.2 Refining Depth Estimation

While leveraging existing depth prediction models for initialization, our architecture uses a refinement stage to address inherent monocular depth estimation inaccuracies. The SCF embeddings serve as input to a regression network that predicts both refined depthmaps and pointmaps, as well as per-point confidence scores associated with both of them. The refinement process employs confidence-aligned loss for depth and pointmap. We create groundtruth pointmaps by unprojecting groundtruth depth using camera intrinsics. This optimization encourages the network to produce accurate depth estimates while automatically learning which predictions require higher confidence weighting.

3.1.3 Hierarchical State Representation Learning:

The Hierarchical state representation handles spatio-temporal consistency at different scales, as shown in Fig. 2.

I. Adaptive Keyframe Strategy: Keyframe selection employs a divergence metric based on cosine similarity between pose embeddings of the current frame (I) and the keyframe (K). A new keyframe is created when the metric exceeds a given threshold, balancing reconstruction accuracy with computational efficiency. This adaptive approach prevents information redundancy while maintaining sufficient overlap for reliable pose estimation.

II. Local State (L): Tracking short-term temporal coherence within a window between consecutive keyframes, **L** employs causal self-attention followed by an MLP for state management. The readout head computes camera transformations (R_I^K, t_I^K) from current frame I to keyframe K. This formulation allows for continuous adaptation to scene changes while maintaining local consistency. During training, for batch computation of attention on the entire sequence across all windows we pad frames in each window to a constant maximum window length.

III. Global State (G): Providing long-term spatial consistency across the entire sequence, G operates at keyframe granularity. When a new keyframe $(K*)$ is selected, global readout computes the transformation (R_{K*}^W, t_{K*}^W) from the new keyframe's camera coordinate frame $(K*)$ to the initial frame's camera coordinate frame (W) through causal self-attention between pose embedding of G and all keys frames upto $K*$. This updates uses the global state embedding to ensure progressive integration of local observations into the global context. For parallel computation we pad all the scenes to a maximum length, and during inference as new keyframe comes in, the padding is updated with the latest keyframe.

In summary, our approach sits conceptually between the memory-driven design of CUT3R and the feed-forward,

windowed architecture of SLAM3R. CUT3R maintains a persistent global memory throughout the entire sequence, updating a scene representation incrementally with each frame. While this enables continuous state tracking, it also causes feature dilution over time—early-frame information is gradually averaged out, degrading geometric detail and limiting responsiveness to scene changes. In contrast, SLAM3R avoids any persistent memory and instead processes short sliding-window clips, selecting keyframes and predicting dense 3D pointmaps via direct correspondence without estimating camera poses. Although efficient, this design lacks structural constraints and limits cross-window consistency.

3.2 Training Objective

3D Confidence-aligned Regression Loss: Following Cut3r, we apply confidence-aware pointmap and depth prediction loss since we are predicting pointmaps in the frame local to each camera and predicting a Rotation and Translation to the target (initial) camera frame. The groundtruth pointmaps are computed by unprojecting depth using camera intrinsics.

$$\mathcal{L}_{\text{conf}} = \sum_{(\hat{\mathbf{x}},c)\in(\hat{\mathcal{X}},\mathcal{C})} \left(c \cdot \left\| \frac{\hat{\mathbf{x}}}{\hat{s}} - \frac{\mathbf{x}}{s} \right\|_2 - \alpha \log c \right), \tag{1}$$

where $\hat{s}$ and s are scale normalization factors for $\hat{\mathbf{x}}$ and $\mathbf{x}$, respectively. Similar to Cut3r, when the ground-truth pointmaps are metric, we set $\hat{s} := s$ to enable the model to learn metric-scale pointmaps.

Pose Regression Loss: We parameterize the pose $\hat{\mathbf{P}}_t$ as quaternion $\hat{\mathbf{q}}_t$ for rotation and translation $\hat{\boldsymbol{\tau}}_t$, and minimize the L2 norm between the prediction and ground truth.

$$\mathcal{L}_{\text{pose}} = \sum_{t=1}^{N} \left(\|\hat{\mathbf{q}}_t - \mathbf{q}_t\|_2 + \left\| \frac{\hat{\boldsymbol{\tau}}_t}{\hat{s}} - \frac{\boldsymbol{\tau}_t}{s} \right\|_2 \right). \tag{2}$$

4 Experiments

4.1 Training Data

Given the broad evaluation scope of CUT3R—trained on a diverse mix of synthetic and real-world datasets spanning object-centric, scene-level, dynamic, and static settings—our goal was to adopt a more focused and computationally feasible dataset selection. While CUT3R demonstrates zero-shot generalization on varied benchmarks such as MPI Sintel [15], Bonn [16], and KITTI [17], training across such heterogeneous datasets is resource-intensive and impractical within the scope of this project. Consequently, we selected **ScanNetV2** and **7-Scenes** as our primary datasets due to their complementary characteristics and suitability for our method's evaluation.

The **ScanNet** dataset [18] is a large-scale collection of RGB-D video sequences captured across 1,513 diverse indoor environments, including offices, apartments, and public spaces. Each sequence in ScanNet is accompanied by accurate camera pose annotations, dense surface reconstructions in the form of mesh models, and instance-level semantic segmentations. The dataset contains over 2.5 million RGB-D frames, making it one of the most comprehensive resources for learning-based 3D scene understanding. The scale and diversity of ScanNet enable models to learn robust scene priors, such as common object shapes and spatial layouts, which are critical for generalization to unseen environments.

In contrast, the **7-Scenes** dataset [19] consists of RGB-D video sequences recorded in seven small indoor environments, such as offices and meeting rooms, using a Kinect sensor. Each frame is annotated with a high-precision camera pose estimated via KinectFusion, alongside dense 3D models of the scenes. While the scale of 7-Scenes is significantly smaller than ScanNet, its precise pose annotations and controlled environment make it a standard benchmark for evaluating camera relocalization, tracking, and temporal consistency in 3D reconstruction systems. The dataset is particularly valuable for assessing the real-time performance and robustness of monocular reconstruction methods, as it provides challenging sequences with significant viewpoint variation and occlusions.

Together, ScanNet and 7-Scenes offer complementary strengths for the development and evaluation of monocular video to 3D reconstruction approaches. ScanNet is primarily used for training due to its large-scale, richly annotated

data, enabling supervised learning of detailed 3D representations. 7-Scenes, on the other hand, is typically employed for benchmarking and cross-dataset generalization tests, providing rigorous evaluation of pose accuracy and reconstruction quality in real-world scenarios. The combination of these datasets allows for the development of models that are both accurate in 3D geometry prediction and robust under diverse deployment conditions.

4.2 Training Details

For training our model, we utilized a distributed setup comprising 8 NVIDIA A100 GPUs (each with 40GB VRAM) and implemented PyTorch's `DistributedDataParallel` framework to ensure efficient parallelization and optimal resource utilization. The training pipeline incorporated pretrained DINOv2 (Vision Transformer, ViT-L/14) features for semantic understanding and depth embeddings derived from Align3r's pretrained depth encoder for geometric consistency. All backbone layers except the final two were frozen. Our training approach encompassed the entire pipeline shown in the fig. 1, including the monocular depth estimation, concept fusion, self-attention mechanism, decoder for point map, depth and confidence prediction, and both local and global state components for pose prediction. During training, batches were sampled as contiguous sequences to preserve temporal and spatial coherence, for estimating the camera pose in sequential manner. Each batch was structured as a tensor of shape `[batch size, sequence length, ...]`, where the sequence length corresponds to consecutive frames from the same scene. This design enables the network to exploit geometric consistency and temporal relationships across multiple views, which is crucial for accurate 3D scene understanding and reconstruction.

Table 1: Training Hyperparameters

Parameter	**Value**
Batch Size	8 per GPU (64 global)
Sequence Length	16
Image size	224×224
Learning Rate	1e-4 with exponential decay
Learning Rate Schedule	Exponential decay ($\gamma = 0.995$)
Gradient Clipping	5.0 max norm
Weight Decay	0.01
Optimizer	AdamW
Training Duration	1000 epochs
Precision	Mixed precision (fp16/fp32)

4.3 Metrics

4.4 Depth Metrics

For depth evaluation, we adopt established metrics to assess accuracy and temporal consistency. The primary metrics include:

- **Absolute Relative Error (AbsRel)** $\downarrow$: Measures proportional deviation between predicted and true depth values, calculated as $\frac{1}{N}\sum_{i=1}^{N}\frac{|d_i-\hat{d}_i|}{d_i}$.
- **Threshold Accuracy** ($\delta < 1.25$) $\uparrow$: Percentage of pixels where predicted-to-groundtruth depth ratio falls within 125% threshold.

4.5 Pose Metrics

For pose estimation evaluation, we utilize three well-established metrics:

- **Absolute Translation Error (ATE)** $\downarrow$: Euclidean distance between estimated and groundtruth trajectories after SE(3) alignment.
- **Relative Translation Error (RTE)** $\downarrow$: Average positional discrepancy over consecutive pose pairs.

- **Relative Rotation Error (RRE)** ↓: Angular difference in rotation components between adjacent poses.

This metric selection enables comprehensive assessment of both absolute trajectory accuracy and local pose consistency, with arrow notations (↓ / ↑) indicating whether lower or higher values represent better performance

4.6 Results

The evaluation of our method was conducted on the test sets of the ScanNet and 7Scenes datasets. For both datasets, the data was partitioned such that 80% was used for training and the remaining 20% was reserved for evaluation.

4.6.1 Quantitative

Table 2: Quantitative comparison of depth estimation performance on the test set of ScanNet and 7Scenes datasets. We report the depth accuracy under metric scale alignment using Absolute Relative Error (Abs Rel) and the percentage of predicted depths within a threshold ($\delta < 1.25$). Lower is better for error metrics; higher is better for $\delta < 1.25$

Alignment	Method	Scannet		7scenes	
		Abs Rel ↓	δ<1.25 ↑	Abs Rel ↓	δ<1.25 ↑
Metric Scale	Cut3r [2]	0.351	48.59	0.111	87.03
	Slam3r [4]	**0.148**	**85.82**	**0.067**	**91.34**
	Ours	0.169	74.55	0.164	75.35

Table 3: Quantitative comparison of pose estimation performance on the ScanNet and 7Scenes datasets. We report the trajectory estimation metrics: Absolute Trajectory Error (ATE), Relative Trajectory Error (RTE), and Relative Rotation Error (RRE). Lower is better for error metrics.

Method	Scannet			7scenes		
	ATE ↓	RTE ↓	RRE ↓	ATE ↓	RTE ↓	RRE ↓
Cut3r [2]	0.099	0.022	0.600	0.568	0.024	0.882
Slam3r [4]	0.066	0.014	**0.515**	0.084	0.012	**0.764**
Ours	**0.049**	**0.011**	1.359	**0.039**	**0.009**	1.612

Depth Estimation Discussion The results in the Tab 2 demonstrate that Slam3r achieves the best overall depth estimation performance across both datasets. On ScanNet, Slam3r obtains the lowest Absolute Relative Error (Abs Rel) of 0.148 and the highest percentage of predicted depths within the $\delta < 1.25$ threshold at 85.82%. This trend continues on the 7Scenes dataset, where Slam3r again outperforms the other methods with an Abs Rel of 0.067 and a $\delta < 1.25$ accuracy of 91.34%. In comparison, the "Ours" method shows a notable improvement over Cut3r, achieving a lower Abs Rel and higher $\delta < 1.25$ on ScanNet (0.169 and 74.55%, respectively), but it lags behind Slam3r. On 7Scenes, "Ours" records an Abs Rel of 0.164 and $\delta < 1.25$ of 75.35%, which, while competitive, is still inferior to Slam3r's results. Cut3r consistently yields the highest error and lowest accuracy among the three methods.

Pose Estimation Discussion Tab 3 presents the trajectory estimation results. Here, the "Ours" method demonstrates superior performance in terms of Absolute Trajectory Error (ATE) and Relative Trajectory Error (RTE) on both datasets. Specifically, on ScanNet, "Ours" achieves the lowest ATE (0.049) and RTE (0.011), indicating the most accurate overall and relative trajectory estimation. On 7Scenes, "Ours" also secures the best results for ATE (0.039) and RTE (0.009). However, in terms of Relative Rotation Error (RRE), Slam3r performs best, with RRE values of 0.515 on ScanNet and 0.764 on 7Scenes, while "Ours" records higher RREs of 1.359 and 1.612, respectively. Cut3r performs worst across all pose estimation metrics.

Analysis The experimental evaluation of the three SLAM approaches-Cut3R, SLAM3R, and our proposed method-demonstrates clear performance distinctions attributable to their underlying architectural designs. Cut3R, which utilizes a single shared state embedding vector for both depth and pose estimation, exhibits the highest errors in both

depth and pose. This degradation is primarily due to the accumulation of errors in the state representation, particularly in complex or extended sequences. SLAM3R addresses some of these limitations through a hierarchical state memory architecture that integrates local frame-to-frame registration with global keyframe optimization. This design leads to notable reductions in both depth error and relative rotation error (RRE), indicating enhanced robustness and improved overall performance compared to Cut3R. Our proposed method further advances trajectory estimation accuracy by employing an attention-based Local-Global state representation combined with adaptive keyframe selection. This approach achieves superior results in absolute trajectory error (ATE) and relative trajectory error (RTE), reflecting more precise trajectory reconstruction. However, depth consistency and relative rotation error (RRE) remains less optimal. The primary factors contributing to this limitation include (i) restricted training epochs due to GPU and dataset constraints, (ii) the absence of a frame-wise pose smoothness loss during training, and (iii) the lack of a separate scale-shift prediction module rather than direct metric depth regression. Overall, these results underscore the inherent trade-offs between trajectory accuracy and depth fidelity in learned monocular SLAM systems for 3D reconstruction. While architectural innovations such as hierarchical memory and attention-based state representations can substantially improve trajectory estimation, achieving high-fidelity depth reconstruction remains challenging under current training and modeling constraints. In terms of computational cost, our approach leverages a single pose CLS token to represent the Local-Global state, resulting in significantly lower resource requirements. In contrast, SLAM3R's I2P and L2W modules incur much higher computational costs due to their dense pointmap prediction and fusion strategies.

4.6.2 Qualitative

We visualize our predicted depth maps and corresponding point clouds for test scenes from the ScanNet (Fig. 3) and 7-Scenes (Fig. 4) datasets. These qualitative results align with our quantitative evaluations: ReMov3r produces geometrically consistent and semantically meaningful reconstructions across a range of environments. Compared to CUT3r, our outputs exhibit fewer artifacts and more reliable surface estimations, particularly in areas with challenging geometry or sparse textures. While SLAM3r produces sharper reconstructions in static, well-structured scenes—benefiting from strong geometric priors—ReMov3r shows greater robustness in cluttered environments.

We note that our current visual performance is partially limited by training constraints: specifically, the absence of infinite value masking in the training data and the use of a suboptimal smoothing loss. We believe that with scaled training and the integration of these improvements, ReMov3r can not only close the remaining visual quality gap with SLAM3r but also surpass it in both robustness and generalization.

5 Limitations

Several limitations in our current approach constrain its overall performance and generalizability. First, the number of training epochs is restricted by available GPU resources and dataset size, which limits the model's capacity to fully learn complex scene representations. Second, the absence of a frame-wise pose smoothness loss during training means that the model is not explicitly penalized for abrupt changes in camera pose between consecutive frames, potentially resulting in jerky motion and suboptimal rotation estimates. Third, the method directly regresses metric depth without a dedicated scale-shift prediction module, which can hinder the accuracy of depth reconstruction. Additionally, the current implementation does not mask depth and pointmap losses for pixels where ground-truth depth is undefined, making the model susceptible to noisy or incomplete sensor data. Finally, evaluation has been primarily conducted on indoor, relatively static environments, limiting the demonstrated generalizability of the approach to more complex real-world scenarios, such as dynamic scenes or outdoor environments with varying lighting and larger spatial scales.

6 Future Work

To address the aforementioned limitations, several avenues for future research are proposed. Inspired by Monst3r [6], we plan to introduce a loss function that penalizes significant variations in camera rotation and translation between consecutive timesteps, thereby encouraging smoother camera motion and improving rotation estimation. We also intend to incorporate masking strategies for depth and pointmap losses to exclude pixels with undefined ground-truth depth, which should enhance robustness to noisy sensor data. To improve generalizability, we aim to extend training and evaluation to more diverse datasets that include dynamic scenes (e.g., TUM RGB-D dynamic subset,

VirtualKITTI [20] with moving objects) and outdoor environments (e.g., KITTI Raw [17], Waymo Open [21], Oxford RobotCar [22] [23]). This will allow us to assess the model's robustness to challenges such as motion blur, dynamic occlusions, changing lighting conditions, and increased scene depth. Furthermore, we will explore explicit modeling of scene dynamics or introduce motion segmentation to decouple moving foreground objects from the static background during training. Data augmentation strategies simulating outdoor conditions and dynamic movements, as well as reassessment of network architecture and loss terms (e.g., photometric consistency, depth smoothness), will also be investigated. Broadening the training set and refining the modeling approach in these ways are expected to enhance both the robustness and scalability of the system, facilitating its application to a wider range of real-world scenarios, including autonomous driving, augmented reality, and mobile robotics.

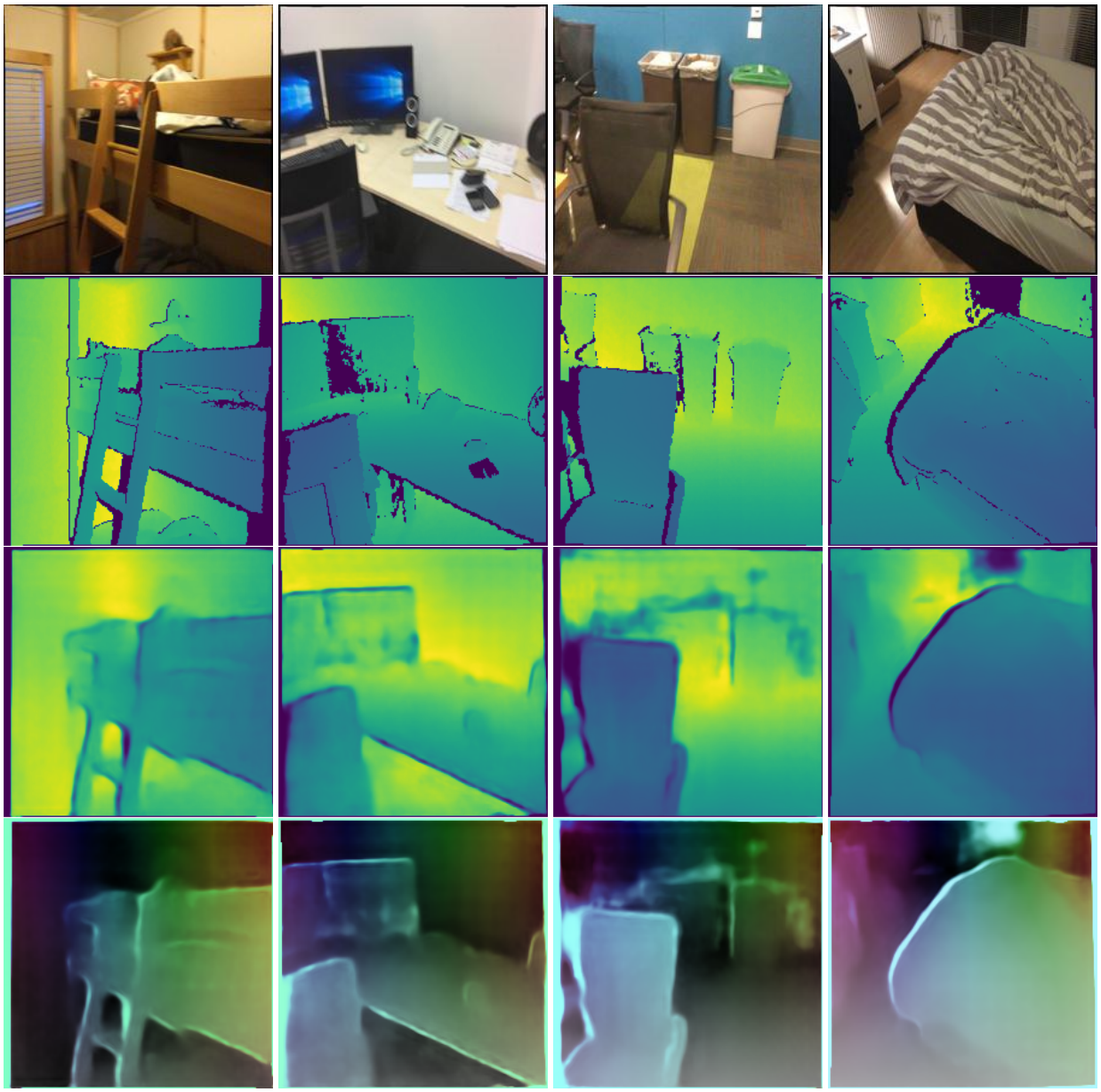

Figure 3: Visualization of groundtruth and our results for depth and pointmap of ScanNet dataset. *Rows first to fourth correspond to RGB images, groundtruth depth, estimated depth and estimated pointmap respectively.*

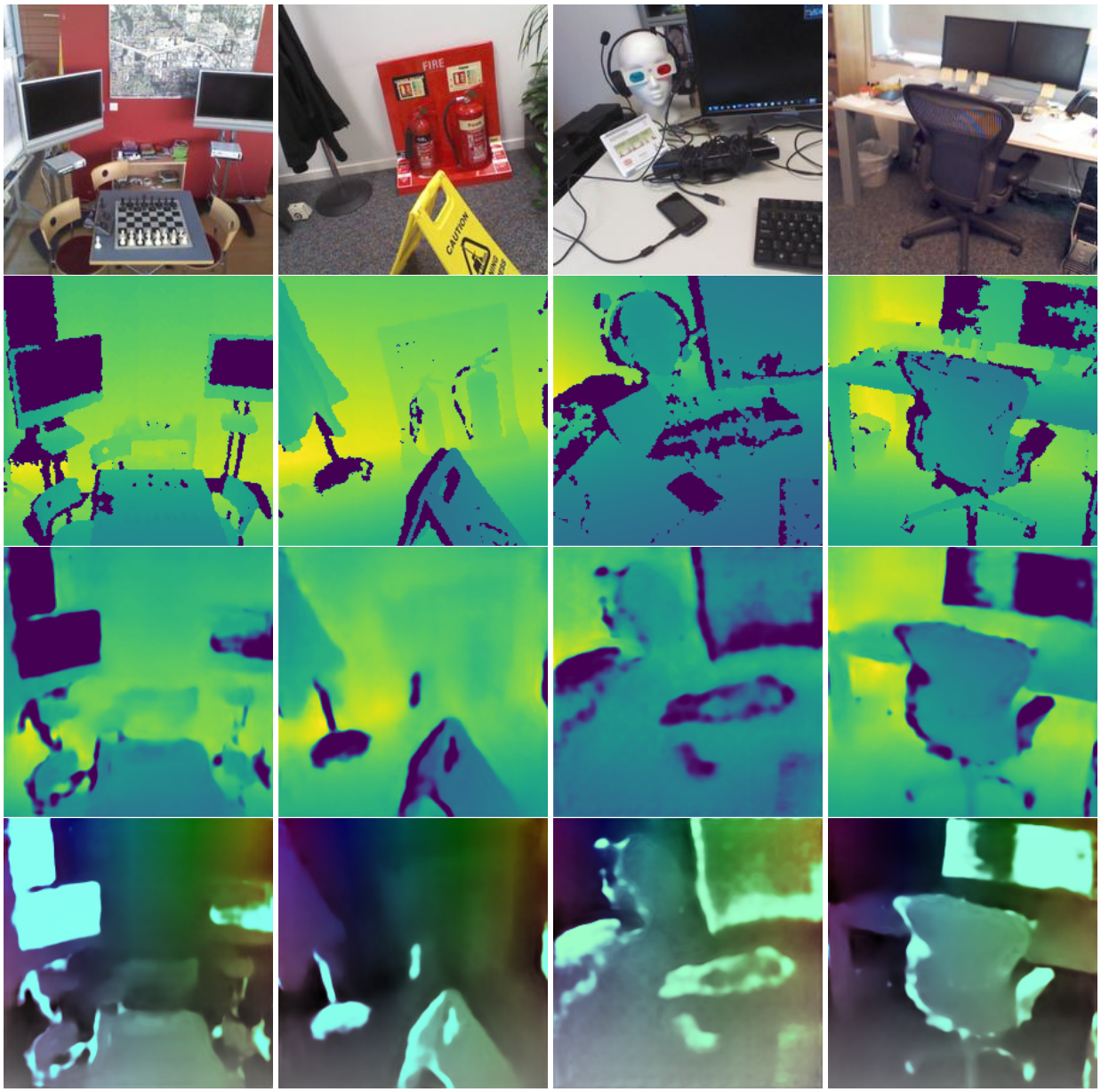

Figure 4: Visualization of groundtruth and our results for depth and pointmap of 7-Scenes dataset. *Rows first to fourth correspond to RGB images, groundtruth depth, estimated depth and estimated pointmap respectively.*

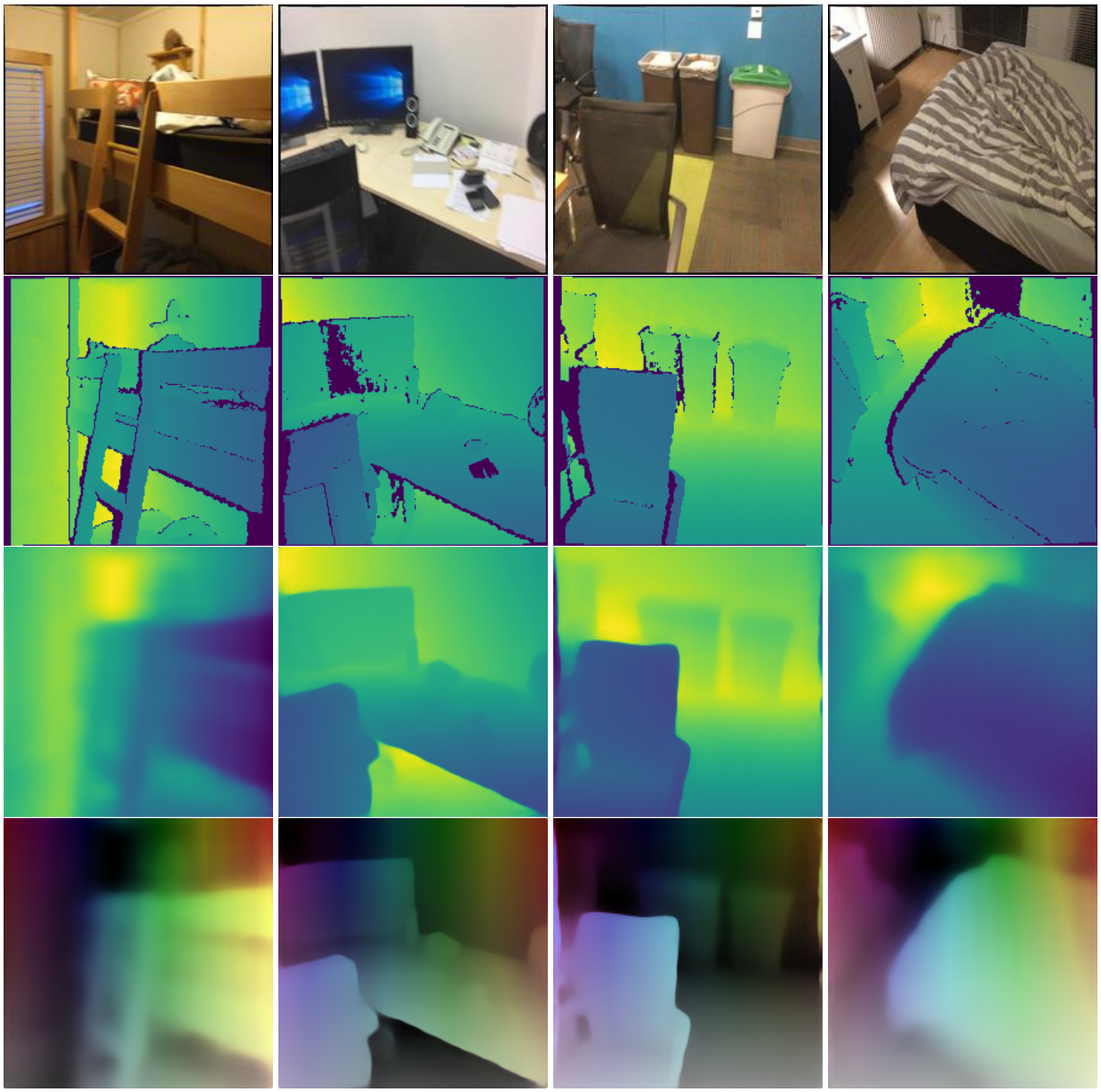

Figure 5: Visualization of groundtruth and Cut3R results for depth and pointmap of ScanNet dataset. *Rows first to fourth correspond to RGB images, groundtruth depth, estimated depth and estimated pointmap respectively.*

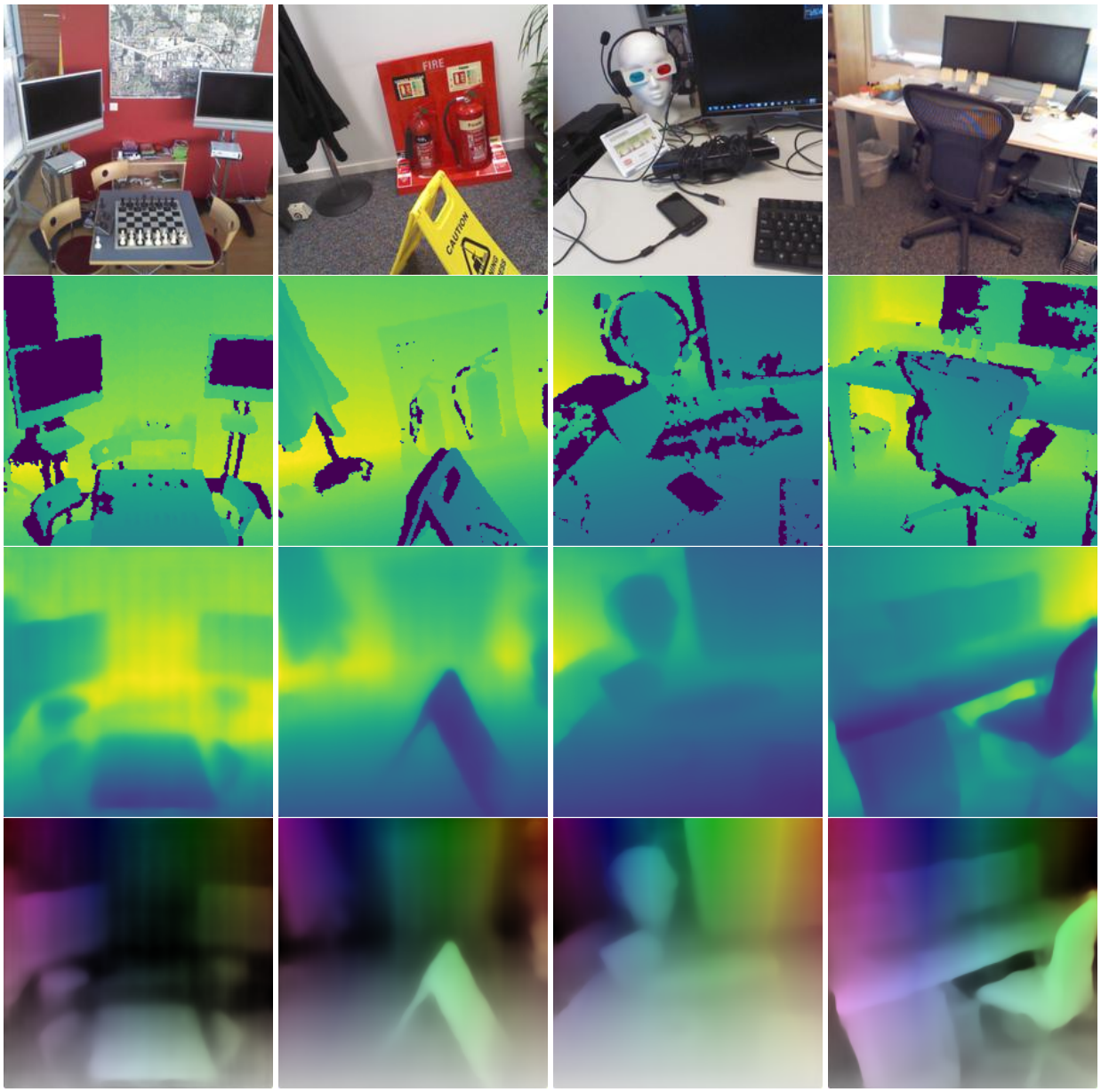

Figure 6: Visualization of groundtruth and Cut3R results for depth and pointmap of 7-Scenes dataset. *Rows first to fourth correspond to RGB images, groundtruth depth, estimated depth and estimated pointmap respectively.*

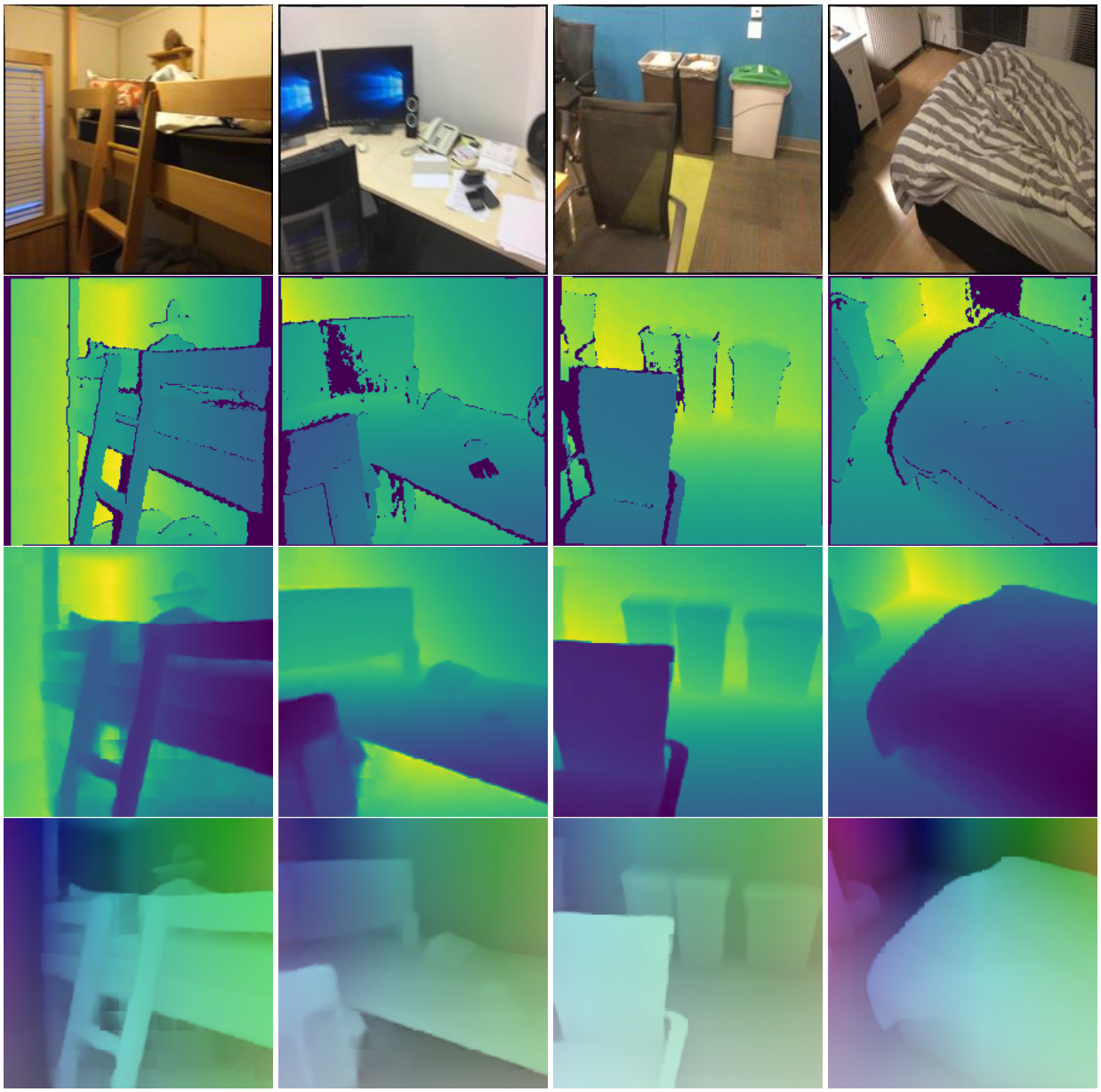

Figure 7: Visualization of groundtruth and SLAMM3R results for depth and pointmap of ScanNet dataset. *Rows first to fourth correspond to RGB images, groundtruth depth, estimated depth and estimated pointmap respectively.*

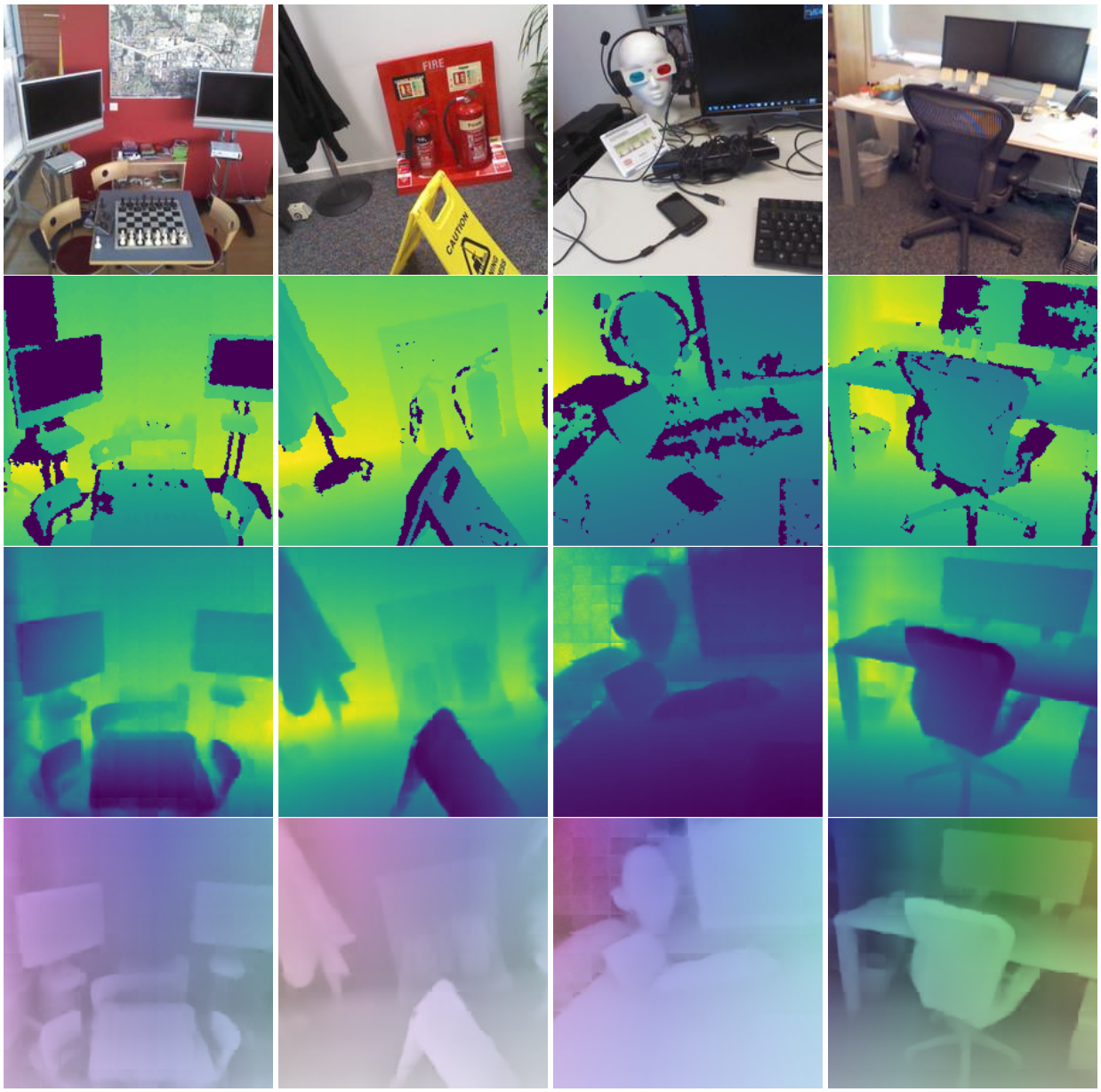

Figure 8: Visualization of groundtruth and SLAMM3R results for depth and pointmap of 7-Scenes dataset. *Rows first to fourth correspond to RGB images, groundtruth depth, estimated depth and estimated pointmap respectively.*

References

[1] Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jerome Revaud. Dust3r: Geometric 3d vision made easy, 2024.

[2] Qianqian Wang, Yifei Zhang, Aleksander Holynski, Alexei A. Efros, and Angjoo Kanazawa. Continuous 3d perception model with persistent state, 2025.

[3] Jiahao Lu, Tianyu Huang, Peng Li, Zhiyang Dou, Cheng Lin, Zhiming Cui, Zhen Dong, Sai-Kit Yeung, Wenping Wang, and Yuan Liu. Align3r: Aligned monocular depth estimation for dynamic videos, 2024.

[4] Yuzheng Liu, Siyan Dong, Shuzhe Wang, Yingda Yin, Yanchao Yang, Qingnan Fan, and Baoquan Chen. Slam3r: Real-time dense scene reconstruction from monocular rgb videos, 2025.

[5] Vincent Leroy, Yohann Cabon, and Jérôme Revaud. Grounding image matching in 3d with mast3r, 2024.

[6] 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, 2024.

[7] Hengyi Wang and Lourdes Agapito. 3d reconstruction with spatial memory, 2024.

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

[9] 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. *arXiv preprint arXiv:2304.07193*, 2023.

[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] Cesar Cadena, Luca Carlone, Henry Carrillo, Yasir Latif, Davide Scaramuzza, Jose Neira, Ian Reid, and John J. Leonard. Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age. *IEEE Transactions on Robotics*, 32(6):1309–1332, December 2016.

[12] Nikhil Keetha, Jay Karhade, Krishna Murthy Jatavallabhula, Gengshan Yang, Sebastian Scherer, Deva Ramanan, and Jonathon Luiten. Splatam: Splat, track and map 3d gaussians for dense rgb-d slam, 2024.

[13] Philipp Lindenberger, Paul-Edouard Sarlin, Viktor Larsson, and Marc Pollefeys. Pixel-perfect structure-from-motion with featuremetric refinement, 2021.

[14] Shaohui Liu, Yifan Yu, Rémi Pautrat, Marc Pollefeys, and Viktor Larsson. 3d line mapping revisited, 2023.

[15] D. J. Butler, J. Wulff, G. B. Stanley, and M. J. Black. A naturalistic open source movie for optical flow evaluation. In A. Fitzgibbon et al. (Eds.), editor, *European Conf. on Computer Vision (ECCV)*, Part IV, LNCS 7577, pages 611–625. Springer-Verlag, October 2012.

[16] E. Palazzolo, J. Behley, P. Lottes, P. Giguère, and C. Stachniss. ReFusion: 3D Reconstruction in Dynamic Environments for RGB-D Cameras Exploiting Residuals. In *IROS*, 2019.

[17] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The kitti dataset. *International Journal of Robotics Research (IJRR)*, 2013.

[18] Angela Dai, Angel X Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[19] Jamie Shotton, Ben Glocker, Christopher Zach, Shahram Izadi, Antonio Criminisi, and Andrew Fitzgibbon. Scene coordinate regression forests for camera relocalization in rgb-d images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013.

[20] Yohann Cabon, Naila Murray, and Martin Humenberger. Virtual kitti 2, 2020.

[21] Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, Vijay Vasudevan, Wei Han, Jiquan Ngiam, Hang Zhao, Aleksei Timofeev, Scott Ettinger, Maxim Krivokon, Amy Gao, Aditya Joshi, Sheng Zhao, Shuyang Cheng, Yu Zhang, Jonathon Shlens, Zhifeng Chen, and Dragomir Anguelov. Scalability in perception for autonomous driving: Waymo open dataset, 2020.

[22] Dan Barnes, Matthew Gadd, Paul Murcutt, Paul Newman, and Ingmar Posner. The oxford radar robotcar dataset: A radar extension to the oxford robotcar dataset. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, Paris, 2020.

[23] Will Maddern, Geoff Pascoe, Chris Linegar, and Paul Newman. 1 Year, 1000km: The Oxford RobotCar Dataset. *The International Journal of Robotics Research (IJRR)*, 36(1):3–15, 2017.