2023 IEEE International Conference on Robotics and Automation (ICRA 2023)
May 29 - June 2, 2023. London, UK

iMODE:Real-Time Incremental Monocular Dense Mapping Using Neural Field

Hidenobu Matsuki, Edgar Sucar, Tristan Laidow, Kentaro Wada, Raluca Scona and Andrew J. Davison

***Abstract*— We present a novel real-time dense and semantic neural field mapping system that uses only monocular images as input. Our scene representation is a dense continuous radiance field represented by a Multi-Layer Perceptron (MLP), trained from scratch in real-time. We build on high-performance sparse visual SLAM and use camera poses and sparse keypoint depths as supervision alongside RGB keyframes. Since no prior training is required, our system flexibly fits to arbitrary scale and structure at runtime, and works even with strong specular reflections. We demonstrate reconstruction over a range of scenes from small indoor to large outdoor spaces. We also show that the method can straightforwardly benefit from additional inputs such as learned depth priors or semantic labels for more precise and advanced mapping.**

I. Introduction

Intelligent embodied devices require a visual SLAM system to build a 3D representation of the environment incrementally and in real time that is accurate and robust enough for localisation and safe navigation, and sufficiently informative for scene understanding tasks such as object recognition and physical interaction. These requirements mean that the 3D representation needs to be compact but able to represent rich scene geometry and semantics. Decades of research showed that keypoint-based SLAM methods show the best performance in terms of accuracy and efficiency, but the sparse map representation tells little about geometry and semantics and it is not useful for advanced robot navigation tasks. To this end, we propose a novel monocular dense mapping system that combines the accuracy and robustness of sparse landmark-based visual SLAM systems with dense photometric alignment on a neural field to incrementally obtain a dense 3D reconstruction from a monocular camera in real time. Inspired by iMAP [1], we represent the geometry and appearance of the 3D scene with a neural field. Recent advances in neural field representations have shown that a small MLP can be optimised, from scratch, to accurately represent the geometry and appearance of a 3D scene [2], [3]. Additionally, the smoothness priors inherent in the structure of the MLP allow these networks to make watertight reconstructions from partial input data without any pre-training. iMAP showed how these neural fields could be used in an accurate, real-time, dense SLAM system. However, it relied on a depth camera, limiting it to certain depth ranges and operating conditions. In our system, the incoming stream of RGB images is fed into an off-the-shelf sparse SLAM system that creates and tracks against keyframe images. A set of sparse but accurate depth estimates are generated for these keyframes and further semi-densified by using an online multi view semi-dense mapping module. In a parallel thread during live operation, the weights of the MLP are optimised to minimise the photometric error between the observed and rendered keyframe images, as well as the geometric distance between the rendered depth and the semi-dense depth estimates. Our system can run at more than 30fps on a single desktop CPU/GPU system.

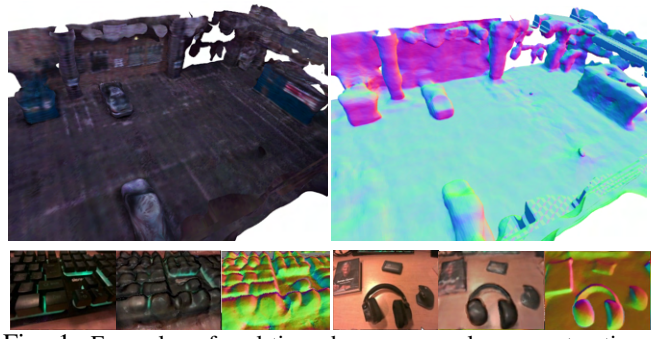

Fig. 1: Examples of real-time dense monocular reconstructions. **Top**: 3D mesh and its normal map of a large-scale parking lot. **Bottom**: Small object reconstruction examples. We show keyframe image and the rendering of shaded colour and normal from the network.

As our system only uses monocular images and no pre-training, we are capable of reconstructing scenes from vastly different domains and scales. We perform exhaustive quantitative evaluations of our system on a number of standard benchmark datasets and show that our method achieves comparable performance to state-of-the-art learning-based methods; however, while these learning-based methods only perform well when tested on domains close to their training data, our performance remains consistent across all datasets. We also show a number of qualitative results of our system on a wide variety of scenes including small complex objects, room-scale indoor environments, and large outdoor scenes. Furthermore, we demonstrate that our approach is very general, modular and can easily be extended to include semantic information, multi-modal sensor fusion, and dense depth priors. Example reconstructions generated by our method are provided in Fig. 1. In summary, the key contributions of this paper are:

- The first real-time purely monocular mapping system using a neural field representation of the scene.
- Dense reconstructions of scenes from arbitrary domains without any additional sensor or prior training.

Research presented in this paper has been supported by Dyson Technology Ltd. Dyson Robotics Laboratory, Imperial College London, United Kingdom h.matsuki20@imperial.ac.uk . R. Scona is with the Advanced Technology Division of Ocado Technology. This work was done when R.Scona was with the Dyson Robotics Labratory

- A demonstration of how our system can be flexibly extended to include semantic information, dense depth priors and multi-modal sensor fusion.

II. Related Work

SLAM with a single camera is attractive due to minimal hardware requirements and flexibility. There has been steady progress since early real-time methods [4], [5] which built sparse feature maps to enable accurate camera tracking. Similar methods are now widespread in industry, and are also well represented by open source systems such as ORB-SLAM [6] and DSO [7]. A SLAM system becomes even more useful when its output is not just localisation but also detailed scene reconstruction. Rising processing power in particular made it feasible to develop real-time systems which built dense scene models [8], [9]. However, real-time fully dense scene reconstruction from only monocular input is still both computationally demanding and somewhat fragile particularly in low texture areas. Therefore, most real-time visual SLAM systems that are close to being useful for applications have required additional sensing, particular from RGB-D cameras [10], [11], [12], [13]. Note though that depth cameras have their own limitations, especially depth measurement range with both minimum and maximum bounds. Deep learning offers new tools and promise for monocular SLAM, with the possibility to move away from hand-designed regularlisers and densely reconstruct scenes in a data-driven manner. Several ways to use neural network representations in SLAM have been proposed, and the main approach has been to pre-learn a representation from a dataset with some ability to use code optimisation or similar at runtime to fit this to a live scenario [14], [15], [16], [17]. It has often been found, however, that it is difficult to get reliable and general high performance from such methods apart from in domains close to the training data. A different way to use neural networks in scene reconstruction is to train a network from scratch to represent one specific scene, and to take advantage of priors which are inherent in the structure of the network itself rather than datasets. This approach has become well known in very high performance offline methods for scene and radiance reconstruction [18], [3], [2], where it was shown that a MLP can represent scenes accurately using neural coordinate fields. While these methods were originally designed for offline use, there has been much recent interest in speeding them up both for training and rendering. iMAP [1] was the first system which showed that a neural field representation could be trained from scratch in real-time as the only scene model in a real-time SLAM system by making use of RGB-D camera input. Unlike other voxel-based Neural Field methods [19] [20], iMAP uses only a single MLP to represent the scene, which has significant advantages in terms of low memory footprint and design simplicity. We show that similar performance can be obtained without depth input by using a sparse monocular system to offer sparse depth measurements and camera tracking. Besides the reduction in hardware complexity required, our monocular system has the benefit of being able to work across a large range of scales inaccessible to depth cameras.

III. System Overview

Our system runs three loosely-coupled processes concurrently. The first is localisation, which uses an off-the-shelf sparse landmark-based SLAM system to provide real-time camera pose estimates. The second process uses these poses to run semi-dense mapping for selected keyframes. Both localisation and semi-dense mapping systems run on the CPU. The camera poses and semi-dense maps are passed to the dense reconstruction system that runs on a GPU and optimises the weights of an MLP representing a 3D neural field. Both colour and depth rendering losses are minimised to create a watertight reconstruction of the scene geometry and appearance. Each of these sub-systems is discussed in detail below; see the overview in Fig. 2.

a) Localisation Process: We use the sparse landmark-based SLAM system ORB-SLAM [6] for localisation, which is highly accurate and robust, and comfortably runs at frame rate on a CPU. We use a subset of the keyframes selected by ORB-SLAM in our semi-dense mapping and dense reconstruction processes.

b) Semi-Dense Mapping Process: As shown in iMAP, the accuracy and convergence speed of a neural-field-based reconstruction system can be dramatically improved by directly supervising the geometry via depth rendering during the online training process. Unlike iMAP, however, our method does not use a depth camera, so any depth measurements used must come from monocular multi-view stereo (MVS) methods. DS-NeRF [21] showed that sparse depth supervision speeds up the convergence with fewer frames, thus we further increase supervision signals using the CPU-based real-time monocular semi-dense mapping system proposed in [22]. This system exploits high-gradient regions to produce semi-dense depth maps for a set of keyframes and camera poses obtained from the localisation procedure. This process runs in a separate thread and it does not delay the main localisation system.

c) Dense Reconstruction Process: In the dense reconstruction process, the weights of the MLP representing the neural field are optimised against a subset of the keyframe images and corresponding semi-dense depth maps produced by the first two processes. To keep the system running at real time, we limit the optimisation window to a fixed size of 7 representative keyframes. To balance adding new information to the map without losing information from previously viewed regions, the representative keyframes consist of the two most recently added keyframes and 5 older keyframes selected at random. This windowed optimisation is run for 20 iterations each time a new keyframe is received. Details of the network architecture, loss formulation and ray sampling procedure are provided in the next section.

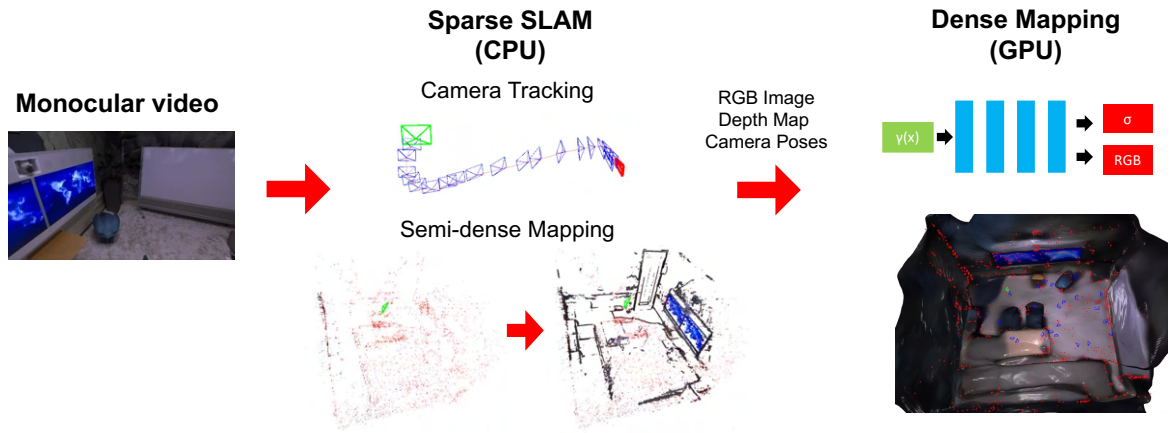

Fig. 2: System overview: from a monocular input stream sparse SLAM is used for camera localisation, a semi dense point cloud is then obtained in high gradient regions, which is used for obtaining a full dense reconstruction through training an MLP neural field representation.

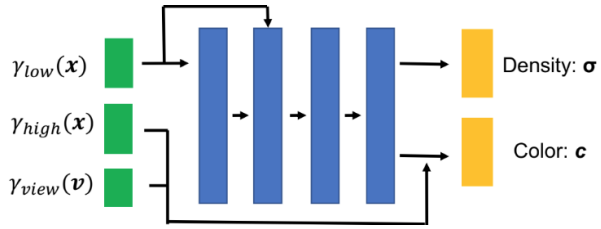

Fig. 3: Neural field represented through a 4-layer MLP, low frequency embedding is used for geometry while a high frequency one along with the viewing direction are used to reconstruct high detailed texture with view-dependant effects.

IV. 3D Scene Representation

A. Network Architecture

The neural field representing the scene is modeled by an MLP, F_θ, with weights θ. F_θ consists of 4 hidden layers of size 256 and a colour and a density head. The 3D coordinates of a query point, $\mathbf{p}$, are mapped to the frequency domain using the positional encoding functions γ_{low} and γ_{high} (mapping to higher and lower sets of frequencies, respectively), and then passed as input to F_θ along with the viewing direction, $\mathbf{v}$. F_θ then predicts the density, σ, and colour, $\mathbf{c}$, corresponding to the query point and viewing direction:

$$F_\theta(\gamma(\mathbf{p}), \mathbf{v}) = (\mathbf{c}, \sigma) \tag{1}$$

Our network architecture has two notable distinctions to the one in iMAP: (i) *View Dependency*: Because our system relies more heavily on photometric consistency we model view dependant effects such as specularities by concatenating the viewing direction to the colour head of the network (following the design of NeRF). (ii) *Frequency Separation*: As natural scenes tend to present higher frequencies in texture than in geometry, we introduce this bias into our network by feeding a lower frequency embedding as the initial input (avoiding jaggy artifacts in geometry), and a higher frequency embedding only to the colour head. A diagram of our proposed network architecture is provided in Fig. 3. Fig. 4 shows the influence of frequency seperation by comparing the rendering quality after running multi-view optimisation.

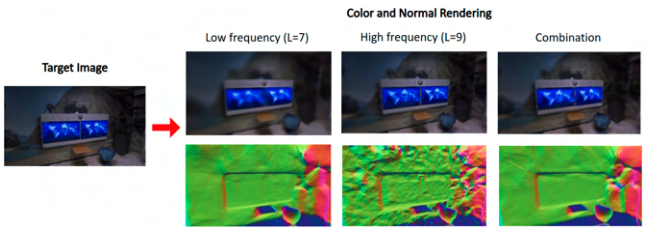

Fig. 4: Positional encoding frequency separation. By using low frequency for geometry input and high frequency for colour head separately, the MLP can output both noise-less surface reconstruction and detailed colour. L denotes the frequency of axis-aligned positional encoding [2].

B. Neural Field Optimisation

The MLP is optimised with respect to the semi-dense depth estimates and colour of the keyframes by volume renderings computed by combining the queried network values along the 3D ray of pixel $[u, v]$:

$$\hat{D}[u,v] = \sum_{i=1}^{N} w_i d_i,\ \hat{I}[u,v] = \sum_{i=1}^{N} w_i c_i$$
$$\hat{D}_{var}[u,v] = \sum_{i=1}^{N} w_i (\hat{D}[u,v] - d_i), \tag{2}$$

where $\hat{D}$ and $\hat{D}_{var}$ are the rendered depth and depth variance respectively, and $\hat{I}$ is the rendered colour image. The weighting $w_i = o_i \prod_{j=1}^{i-1}(1-o_j)$ is the ray-termination probability of the ith sample along the ray at depth d_i, where $o_i = 1 - exp(-\rho_i \delta_i)$ is the occupancy activation function with the inter-sample distance $\delta_i = d_{i+1} - d_i$. We minimise the colour and depth consistency loss for a small number of pixels from selected keyframes. For M pixel samples and W keyframes, the photometric loss, L_p, is the sum of the difference between rendered and measured colour values, and the geometric loss, L_g, is the difference between the depth rendering and the semi-dense depth estimates, normalised by $\hat{D}_{var}$:

$$L_p = \frac{1}{M}\sum_{i=1}^{W}\sum_{s_i}|I_i[u,v]-\hat{I}_i[u,v]|, L_g = \frac{1}{M}\sum_{i=1}^{W}\sum_{s_i}\frac{|D_i[u,v]-\hat{D}_i[u,v]|}{\sqrt{\hat{D}_{var}[u,v]}} \quad (3)$$

We also apply a surface smoothness loss to the predicted normals as proposed in UNISURF [23]. Given $\mathbf{n}(\mathbf{x_t})$, the predicted unit normal vector at the ray-termination point, $\mathbf{x_t}$, we enforce the nearby normals to be similar:

$$L_n = \frac{1}{M}\sum_{i=1}^{W}\sum_{(u,v)\in s_i}||\mathbf{n}(\mathbf{x_t})-\mathbf{n}(\mathbf{x_t}+\varepsilon)||, \quad (4)$$

where ε is a small random perturbation in 3D.

We minimise the weighted loss sum with constants λ_p and λ_n (5.0 for both):

$$\theta\frac{1}{M}\sum_{i=1}^{W}\sum_{(u,v)\in s_i} L_g + \lambda_p L_p + \lambda_n L_n \quad (5)$$

We sample 200 points from pixels where semi-dense depth values are available (mainly corners or edges) for full supervision and another 200 points from randomly selected pixels for colour supervision.

C. View-based Adaptive Ray Sampling

We use an adaptive interval based on the scale of estimated pixel depth to efficiently allocate points around the surface. Let $d_{\min}$ and $d_{\max}$ be the per-frame minimum and maximum sampling distances along the ray, $\mathbf{D_0}$ be the depth map of initial frame in ORB-SLAM, and $\mathbf{D}$ be the depth map of the target frame. We change the ray sampling distance from the initial interval value $(d_{\max,0}, d_{\min,0})$ based on the scale of the view, calculated by the median value of sparse depth map:

$$(d_{\max}, d_{\min}) = \frac{\text{Median}(\mathbf{D})}{\text{Median}(\mathbf{D_0})}(d_{\max,0}, d_{\min,0}). \quad (6)$$

Fig. 5 shows the effect of adaptive ray sampling, which helps to recover fine detail by providing denser training signals around the surface. Without it we observe spotted artifacts due to wide sampling steps.

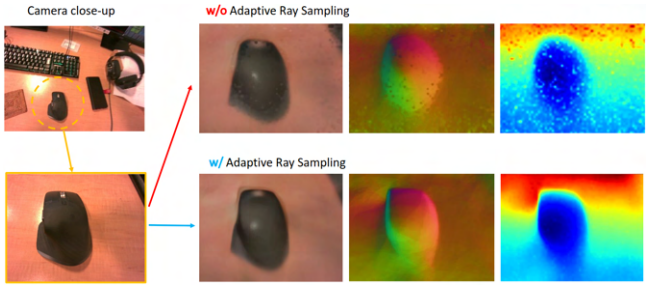

Fig. 5: Adaptive ray sampling. From left to right, the middle images show the rendered colour, normals and depth with and without adaptive sampling.

V. Experiments

We evaluate our system in a variety of small scale indoor and large scale outdoor scenes. Our monocular system can reconstruct dense geometry in these domains without any prior training. We also show that it can flexibly make use of prior information (such as domain-specific network depth prediction or multi-modal sensor inputs) to improve performance. We also demonstrate the dense semantic label fusion. All experiments use a desktop system with an Intel Core i9 12900K 3.50GHz CPU and an NVIDIA Geforce RTX 3090 GPU. As in iMAP, the meshes and view renderings we show are for visualisation and evaluation purposes and do not form part of our system. Sparse SLAM and the semi-dense mapper are implemented in C++ and the dense mapping in Python (PyTorch). The communication is done via ROS. The input image stream is 15Hz for benchmark datasets and 30Hz for our demos. For low frequency encoding, we use axis-aligned encoding with $L_{\text{low}} = 7$ or 8 for our demos and Gaussian embedding [24] with $\sigma = 35$ and size 123 for dataset evaluation. We use axis-aligned encoding with $L_{\text{high}} = 9$ and $L_{\text{view}} = 6$ for higher frequency and view direction encoding respectively. We do not use a view direction on the Replica dataset because the dataset does not contain specular reflection.

A. Dense Scene Reconstruction: Quantitative Results

We evaluate our system on a 2D depth benchmark and a 3D reconstruction benchmark. Our method only uses monocular color images without any depth sensor and learned prior. To this end, the most appropriate baselines are other methods that also restrict their inputs in this way such as REMODE [25] and Plane Sweep Stereo [26], which reconstruct scene geometry as 2D depth maps. Therefore we firstly show the 2D depth map evaluation for the fairest comparison to the baselines. We also compare our method to learning-based approaches in 2D depth evaluation to show our method has a strong domain-agnostic performance. Furthermore, we conduct a 3D scene evaluation of our scene reconstruction performance and show how our monocular reconstruction performs compared to iMAP which uses full dense depth sensor input in indoor scenes.

a) **Depth Evaluation:** We show the evaluation of the 2D depth map quality on public benchmark datasets [27], [28], [29] (Table I). Following [30], we measure the accuracy and density of the depth map by computing the percentage of depth values with an error less than 10% of the ground truth depth. We compare against two real-time monocular motion stereo algorithms with pure geometric formulation (Plane Sweep Stereo [26] and REMODE [25]), a learning-based multi-view dense reconstruction method (NeuralRecon [31]) and a single-view depth completion method (SparseToDense [32]). To generate pseudo-ground-truth depth maps for the EuRoC dataset, we first generate a global mesh model using the provided stereo camera data and ground truth camera poses and render it to each view. For Plane Sweep Stereo and REMODE, we use 5 neighboring ORB-SLAM keyframes per image for dense stereo. For NeuralRecon and SparseToDense, we use the public models pre-trained on ScanNet [33] and the NYU dataset [34] respectively. We provide ground truth camera poses to NeuralRecon to run its own keyframe selection. As Fig. 6 shows our method can smoothly reconstruct scenes thanks to the inherent prior in the MLP and multi-view consistent single global scene

model. Learning-based methods, in particular NeuralRecon can vary a lot in performance depending on the test dataset. This is because NeuralRecon is trained on small room-scale data (ScanNet) and learns a strong prior regarding the object domain or scene scale. This cannot easily be applied to different scene domains such as the large scale hall in the EuRoC dataset.

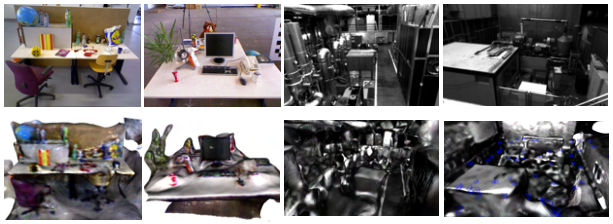

Fig. 6: Example images (top) and reconstructed 3D meshes (bottom) on TUM and EuRoC dataset. Meshes are phong-shaded to highlight the geometry.

	TUM RGB-D			EuRoC MH		Replica				Avg.
	fr1	fr2	fr3	MH01	MH02	o0	o1	r0	r1	
Ours	80.7	**85.7**	61.9	**63.6**	**66.6**	**87.4**	**80.6**	**88.1**	**91.1**	**78.4**
Plane Sweep Stereo [26]	39.3	41.5	38.7	36.4	30.5	60.4	45.8	52	47.8	43.6
REMODE [25]	25.62	24.28	7.83	13.3	11.1	13.5	14.1	22.8	20.1	16.9
NeuralRecon [31]	**84.7**	79.3	**76.2**	1.24	2.09	18.4	26.4	0.99	2.1	32.4
SparseToDense [32]	24.4	32.3	32.0	35.9	35.8	52.6	42.3	56.7	52.9	40.5

TABLE I: Depth map evaluation.

b) **3D Scene Reconstruction:** We evaluate 3D reconstruction quality against the ground truth meshes of the Replica dataset [29]. As in [1], we sample 200,000 points from both ground truth and reconstructed meshes and evaluate the accuracy, completion and completion ratio. We first run Sim(3) alignment between the monocular SLAM trajectory and groundtruth trajectory and then apply the Sim(3) matrix to the SLAM mesh for alignment. As Table II shows, our monocular system achieves a completion ratio similar to RGB-D methods with less than 20cm error. While our purely monocular method cannot outperform the RGB-D based method in indoor scenes where the depth sensor works, our method works in a much wider variety of scenes where the depth sensor cannot provide measurements. Figure 7 shows 3D meshes from our system.

		r0	r1	r2	o0	o1	o2	o3	o4	avg
iMAP (RGBD)	Acc. [cm] ↓	3.58	3.69	4.68	5.87	3.71	4.81	4.27	4.83	4.43
	Comp. [cm] ↓	5.06	4.87	5.51	6.11	5.26	5.65	5.45	6.59	5.56
	Comp. Ratio [<5cm %] ↑	83.9	83.4	75.5	77.7	79.6	77.2	77.3	77.6	79.0
Ours (Monocular)	Acc. [cm] ↓	7.4	6.4	9.3	6.6	11.8	11.4	9.4	8.0	8.78
	Comp. [cm] ↓	13.5	10.1	19.2	9.7	17	14.5	11.8	15.4	13.9
	Comp. Ratio [<5cm %] ↑	38.7	46.1	36.1	49.3	30.1	29.8	36.0	31.0	37.1
	Comp. Ratio [<20cm %] ↑	79.3	85.8	73.1	88.7	75	74.7	81.0	74.8	79.1

TABLE II: Reconstruction results for 8 indoor Replica scenes. r and o stand for room and office sequences.

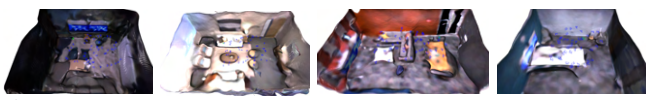

Fig. 7: 3D reconstruction results on the Replica dataset: our system generates dense and watertight mesh models in real-time from monocular video.

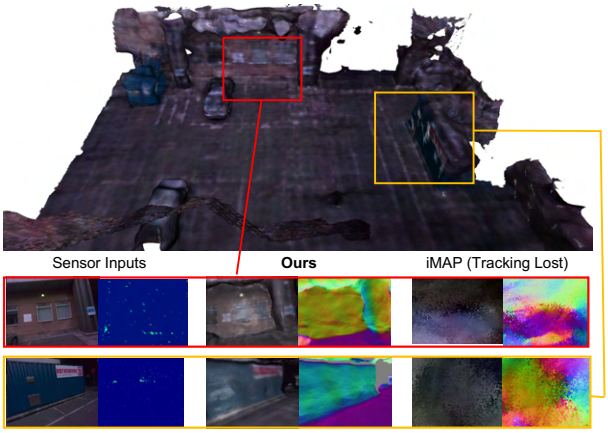

Fig. 8: Large-scale reconstruction: The proposed method can reconstruct outdoor scenes measuring tens of metres while iMAP, which relies on depth camera input, fails.

Fig. 9: Small-scale reconstruction: While commodity depth sensors cannot measure surfaces closer than 10cm and iMAP completely fails at this scale, our purely monocular system can reconstruct each key of the mechanical keyboard.

B. Dense Scene Reconstruction: Qualitative Results

We further demonstrate the uniqueness of our method by using three different types of recorded scenes. Unlike other RGB-D SLAM and Learning-based SLAM, we show that our method is not bound by the measurement range of the sensor or the domain of the training data, but works in a wide range of real-world scenes. Please also check the video submission to see the real-time incremental reconstruction.

a) **Scene reconstruction at different scales:** As Fig. 8 shows, we can reconstruct a dense 3D model in real-time even in a large-scale outdoor parking lot, while iMAP does not work at all in this sequence because a commodity ToF sensor fails to measure the depth in this scene due to the limited sensor measurement range. Our method can also reconstruct small, detailed objects, which are also outside the measurement range of depth sensors. Fig. 9 is an example of mechanical keyboard reconstruction. While the depth from the RGB-D camera has huge holes and an RGB-D based SLAM system (iMAP) completely fails, our purely monocular system is still able to track and reconstruct the detail of the each individual key.

b) **Specular Reflection:** Including the viewing-direction vector allows to model specularities which break the inter-frame photometric consistency. Fig. 10 demonstrates how our method can handle a strong specular reflection on a desk and accurately reconstruct the dense geometry, while a system that does not include the viewing direction

corrupts the geometry in an attempt to account for the lack of colour consistency.

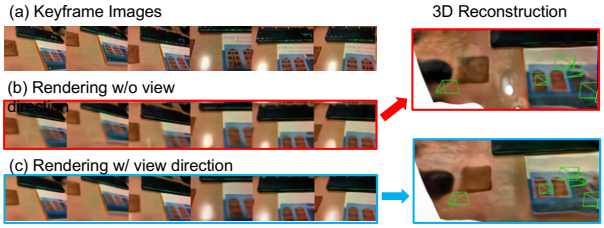

Fig. 10: View dependent effect: The use of view-direction vector allows our system to deal with strong specular reflection.

c) **Analysis of Supervision Signals:** We investigate the effect of semi-dense mapping and different loss terms by running our system on the Replica dataset. As Fig. 11 shows, semi-dense depth maps provide more geometric constraints, speeding up convergence and increasing reconstruction accuracy. We also analyzed the effect of depth and colour supervision independently and found that depth supervision enables fast coarse geometry convergence and colour supervision helps recover the detail where a depth signal from MVS is not available.

C. Multi-modal Data Fusion

While we discussed the domain-agnostic performance of our system in the previous sections, our method can easily make full use of an available domain-specific prior.

1) Predicted Dense Depth Fusion: We show that our system can use and improve upon a dense depth prior provided by a CNN by simply adding more supervisory signals to geometric loss function. We trained an off-the-shelf depth-completion network [32] on ScanNet and evaluated it on sequences from the test split. We run RGB-D ORB-SLAM to get a metric sparse depth map and camera poses in real-time and then apply depth completion for each keyframe to obtain a dense depth prior. Note that we do not use dense sensor depth information for the supervision. We sample depth from pixels with sparse SLAM measurements and the dense CNN-based predictions equally. As Table. III and Fig. 12 shows, the learned priors have inaccurate depth values but our method can fuse them to coherent 3D models. The reconstruction can be further improved using the uncertainty of a dense depth prior as proposed in [35], [36].

2) Semantic Label Fusion: Our monocular mapping system allows for the fusion and propagation of partially observed semantic labels. Following [37], [38], we add a semantic prediction head to the MLP and supervise it by comparing the rendered semantic labels to the semantic labels of the keyframes. For the semantic view synthesis task, we provide ground truth semantic labels to keyframes and evaluate novel view label synthesis performance after the SLAM process finishes. For the semantic label denoising task, we provide labels with pixel-wise random noise and evaluate the denoising effect by multi-view label fusion. Table IV and Fig 13 shows results on the 3 sequences from Replica (office0, room0, room1). The method is able to denoise partial input label data, achieving 10x more accuracy as measured by mIoU.

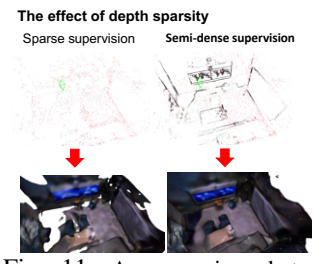

Fig. 11: A comparison between different supervision signals. (left) Depth sparsity analysis. (right) Loss function analysis. The combination of semi-dense depth and colour supervision generates the best reconstruction.

	0084_00	0100_00	0356_02	0406_02	0535_00
Ours + SparseToDense [32]	**0.279**	**0.061**	**0.207**	**0.068**	**0.21**
Ours (Sparse input only)	1.08	0.23	0.689	0.22	0.457
SparseToDense [32]	1.19	0.065	1.17	0.151	0.829

TABLE III: Depth RMSE in metres on ScanNet dataset.

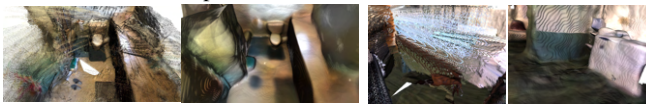

Fig. 12: The use of dense depth prior prediction. (A, C) Back projection of depth map predicted by depth-completion network. (B, D) 3D mesh reconstruction by our system using the dense depth prior.

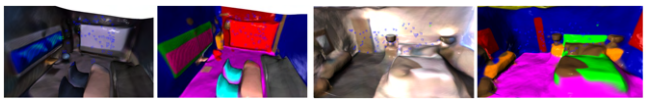

Fig. 13: Semantic label fusion examples on Replica dataset.

3) Sensor Fusion: We also show the proposed method can use multi-modal sensor inputs commonly used in robotics applications such as stereo-inertial data. The sparse SLAM module [39] works as an interface to the multi-modal sensor inputs and sends more accurate camera poses and dense depth maps to dense mapping process. Fig. 14 is a qualitative result on EuRoC dataset.

VI. Conclusions

We presented the first real-time dense monocular mapping system using a neural field. Our system requires no prior training or active sensors and can estimate any scene at runtime. Our evaluation shows the flexibility of our method to achieve complete and watertight reconstructions over multiple domains, including small objects and large outdoor scenes. Our system is also capable of harnessing extra domain-specific prior information that may be available. In the future, we hope to further improve the accuracy of the method over multiple scales, using adaptive positional encoding. We also aim to experiment with tight integration

(a) Label Synthesis

mIoU	Avg Acc	Total Acc
0.776	0.863	0.938

(b) Label Denoising

Noise Ratio	Label type	mIoU	Avg Acc	Total Acc
50%	Input	0.182	0.286	0.538
	Denoised	0.676	0.77	0.918
90%	Input	0.048	0.132	0.169
	Denoised	0.556	0.640	0.896

TABLE IV: Quantitative evaluation of semantic label fusion.

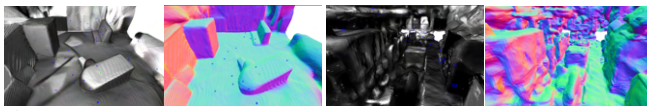

Fig. 14: 3D reconstruction by using stereo-inertial data on EuRoC dataset.

of the pose tracking module.

REFERENCES

[1] E. Sucar, S. Liu, J. Ortiz, and A. J. Davison, "imap: Implicit mapping and positioning in real-time," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 6229–6238.

[2] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, "Nerf: Representing scenes as neural radiance fields for view synthesis," in *European conference on computer vision*. Springer, 2020, pp. 405–421.

[3] L. Yariv, J. Gu, Y. Kasten, and Y. Lipman, "Volume rendering of neural implicit surfaces," *Advances in Neural Information Processing Systems*, vol. 34, 2021.

[4] A. J. Davison, N. D. Molton, I. Reid, and O. Stasse, "MonoSLAM: Real-Time Single Camera SLAM," *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 29, no. 6, pp. 1052–1067, 2007.

[5] G. Klein and D. W. Murray, "Parallel Tracking and Mapping for Small AR Workspaces," in *Proceedings of the International Symposium on Mixed and Augmented Reality (ISMAR)*, 2007.

[6] R. Mur-Artal and J. D. Tardós, "Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras," *IEEE transactions on robotics*, vol. 33, no. 5, pp. 1255–1262, 2017.

[7] J. Engel, V. Koltun, and D. Cremers, "Direct sparse odometry," *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 2017.

[8] R. A. Newcombe, S. Lovegrove, and A. J. Davison, "DTAM: Dense Tracking and Mapping in Real-Time," in *Proceedings of the International Conference on Computer Vision (ICCV)*, 2011.

[9] J. Engel, T. Schoeps, and D. Cremers, "LSD-SLAM: Large-scale direct monocular SLAM," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2014.

[10] T. Whelan, S. Leutenegger, R. F. Salas-Moreno, B. Glocker, and A. J. Davison, "ElasticFusion: Dense SLAM without a pose graph," in *Proceedings of Robotics: Science and Systems (RSS)*, 2015.

[11] T. Schöps, T. Sattler, and M. Pollefeys, "BAD SLAM: Bundle adjusted direct RGB-D SLAM," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[12] A. Dai, M. Nießner, M. Zollhöfer, S. Izadi, and C. Theobalt, "BundleFusion: Real-time Globally Consistent 3D Reconstruction using On-the-fly Surface Re-integration," *ACM Transactions on Graphics (TOG)*, vol. 36, no. 3, pp. 24:1–24:18, 2017.

[13] C. Kerl, J. Sturm, and D. Cremers, "Dense visual SLAM for RGB-D cameras," in *Proceedings of the IEEE/RSJ Conference on Intelligent Robots and Systems (IROS)*, 2013.

[14] M. Bloesch, J. Czarnowski, R. Clark, S. Leutenegger, and A. J. Davison, "CodeSLAM — learning a compact, optimisable representation for dense visual SLAM," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[15] E. Sucar, K. Wada, and A. J. Davison, "NodeSLAM: Neural object descriptors for multi-view shape reconstruction," in *Proceedings of the International Conference on 3D Vision (3DV)*, 2020.

[16] H. Matsuki, R. Scona, J. Czarnowski, and A. J. Davison, "CodeMapping: Real-time dense mapping for sparse SLAM using compact scene representations," *IEEE Robotics and Automation Letters*, vol. 6, no. 4, pp. 7105–7112, 2021.

[17] K. Tateno, F. Tombari, I. Laina, and N. Navab, "CNN-SLAM: Real-time dense monocular slam with learned depth prediction," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[18] M. Niemeyer, L. Mescheder, M. Oechsle, and A. Geiger, "Differentiable volumetric rendering: Learning implicit 3d representations without 3d supervision," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 3504–3515.

[19] T. Müller, A. Evans, C. Schied, and A. Keller, "Instant neural graphics primitives with a multiresolution hash encoding," *ACM Trans. Graph.*, vol. 41, no. 4, pp. 102:1–102:15, Jul. 2022. [Online]. Available: https://doi.org/10.1145/3528223.3530127

[20] Z. Zhu, S. Peng, V. Larsson, W. Xu, H. Bao, Z. Cui, M. R. Oswald, and M. Pollefeys, "Nice-slam: Neural implicit scalable encoding for slam," *arXiv*, 2021.

[21] K. Deng, A. Liu, J.-Y. Zhu, and D. Ramanan, "Depth-supervised nerf: Fewer views and faster training for free," *arXiv preprint arXiv:2107.02791*, 2021.

[22] R. Mur-Artal and J. D. Tardós, "Probabilistic semi-dense mapping from highly accurate feature-based monocular slam." in *Robotics: Science and Systems*, vol. 2015. Rome, 2015.

[23] M. Oechsle, S. Peng, and A. Geiger, "Unisurf: Unifying neural implicit surfaces and radiance fields for multi-view reconstruction," *arXiv preprint arXiv:2104.10078*, 2021.

[24] M. Tancik, P. Srinivasan, B. Mildenhall, S. Fridovich-Keil, N. Raghavan, U. Singhal, R. Ramamoorthi, J. Barron, and R. Ng, "Fourier features let networks learn high frequency functions in low dimensional domains," *Advances in Neural Information Processing Systems*, vol. 33, pp. 7537–7547, 2020.

[25] M. Pizzoli, C. Forster, and D. Scaramuzza, "Remode: Probabilistic, monocular dense reconstruction in real time," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2014, pp. 2609–2616.

[26] D. Gallup, J.-M. Frahm, P. Mordohai, Q. Yang, and M. Pollefeys, "Real-time plane-sweeping stereo with multiple sweeping directions," in *2007 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2007, pp. 1–8.

[27] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers, "A Benchmark for the Evaluation of RGB-D SLAM Systems," in *Proceedings of the IEEE/RSJ Conference on Intelligent Robots and Systems (IROS)*, 2012.

[28] M. Burri, J. Nikolic, P. Gohl, T. Schneider, J. Rehder, S. Omari, M. W. Achtelik, and R. Siegwart, "The EuRoC Micro Aerial Vehicle Datasets," *International Journal of Robotics Research (IJRR)*, vol. 35, no. 10, pp. 1157–1163, September 2016.

[29] J. Straub, T. Whelan, L. Ma, Y. Chen, E. Wijmans, S. Green, J. J. Engel, R. Mur-Artal, C. Ren, S. Verma *et al.*, "The replica dataset: A digital replica of indoor spaces," *arXiv preprint arXiv:1906.05797*, 2019.

[30] K. Tateno, F. Tombari, I. Laina, and N. Navab, "Cnn-slam: Real-time dense monocular slam with learned depth prediction," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 6243–6252.

[31] J. Sun, Y. Xie, L. Chen, X. Zhou, and H. Bao, "Neuralrecon: Real-time coherent 3d reconstruction from monocular video," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 15 598–15 607.

[32] F. Ma and S. Karaman, "Sparse-to-dense: Depth prediction from sparse depth samples and a single image," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2018.

[33] A. Dai, A. X. Chang, M. Savva, M. Halber, T. Funkhouser, and M. Nießner, "ScanNet: Richly-annotated 3d reconstructions of indoor scene," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[34] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus, "Indoor segmentation and support inference from RGBD images," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2012.

[35] B. Roessle, J. T. Barron, B. Mildenhall, P. P. Srinivasan, and M. Nießner, "Dense depth priors for neural radiance fields from sparse input views," *arXiv preprint arXiv:2112.03288*, 2021.

[36] Y. Wei, S. Liu, Y. Rao, W. Zhao, J. Lu, and J. Zhou, "Nerfingmvs: Guided optimization of neural radiance fields for indoor multi-view stereo," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 5610–5619.

[37] S. Zhi, T. Laidlow, S. Leutenegger, and A. J. Davison, "In-place scene labelling and understanding with implicit scene representation," in *Proceedings of the International Conference on Computer Vision (ICCV)*, 2021.

[38] S. Zhi, E. Sucar, A. Mouton, I. Haughton, T. Laidlow, and A. J. Davison, "iLabel: Interactive neural scene labelling," 2021.

[39] C. Campos, R. Elvira, J. J. G. Rodríguez, J. M. Montiel, and J. D. Tardós, "Orb-slam3: An accurate open-source library for visual, visual–inertial, and multimap slam," *IEEE Transactions on Robotics*, vol. 37, no. 6, pp. 1874–1890, 2021.