M2H-MX: Multi-Task Semantic and Geometric Perception for Real-Time Monocular 3D Scene Graph Construction

U.V.B.L. Udugama[1], George Vosselman[1], and Francesco Nex[1]

***Abstract*— Reliable robot autonomy requires spatial representations that ground semantic understanding in metric geometry that remains stable under motion. 3D scene graphs offer such a structure, but existing systems typically rely on RGB-D or LiDAR sensing to obtain reliable geometry, limiting deployment on compact robotic platforms. This paper presents M2H-MX, a monocular dense perception front end that predicts metric depth and semantic labels from RGB frames. These predictions are integrated with IMU-assisted odometry in a real-time metric-semantic mapping pipeline for downstream 3D scene graph construction. M2H-MX combines DINOv3 feature adaptation, register-guided multi-scale decoding, and directed cross-task refinement so that global scene context, local boundaries, and geometric cues are fused before map integration. Unlike dense prediction studies that stop at per-frame metrics, we evaluate M2H-MX both as a benchmark predictor and as a deployed mapping component. On NYUDv2, M2H-MX improves semantic mIoU by 4.06 points and reduces depth RMSE by 9.4% over the strongest multi-task baseline considered. In ScanNet deployment, the M2H-MX Mono-Hydra stack reduces average ATE from 17.59 cm to 6.91 cm compared with monocular GO-SLAM, while sustaining a 25–30 Hz asynchronous perception-to-mapping loop at 640×480 input resolution on an RTX 4080 Super. A TensorRT FP16 deployment reaches 25 FPS on a Jetson Orin NX 16 GB at 192×256 input resolution. These results indicate that improving the monocular perception front end can strengthen the metric-semantic representation required for real-time 3D scene graph construction.**

Keywords— 3D scene graphs, dense multi-task learning, robot perception, real-time scene understanding

I. Introduction and Related Work

Humans describe tasks through places, objects, and relations, while robots sense the world through images, motion, and geometry. Actionable 3D scene graphs [1] help bridge this gap by converting metric maps into structured representations that connect rooms, objects, semantic regions, and spatial relations for planning, navigation, and human-robot interaction, as in Kimera [2] and Hydra [3]. This structure is only useful when the underlying map is metrically stable and semantically consistent: distorted depth can bend walls or shift objects, while unstable semantic labels can corrupt object and region nodes over time.

Most practical metric-semantic mapping and 3D scene graph systems obtain reliable geometry from RGB-D, stereo, or LiDAR sensing, including Kimera [5] and Hydra [3]. These sensors reduce depth ambiguity, but increase cost, power, calibration effort, and platform complexity. This is limiting for drones and compact robots, where payload, power, and mechanical simplicity are critical. Monocular RGB cameras are lighter and more widely deployable, but they move the burden of reliability to the perception front end: metric depth, semantic labels, camera tracking, and map fusion must remain consistent from a passive visual stream, with inertial measurements providing motion constraints rather than direct scene geometry. Monocular systems such as DROID-SLAM [6] and GO-SLAM [7] show strong tracking and reconstruction performance, while Mono-Hydra [4] extends monocular RGB-inertial perception toward scene graph construction. In all cases, the quality and stability of the visual front end strongly affect the final map.

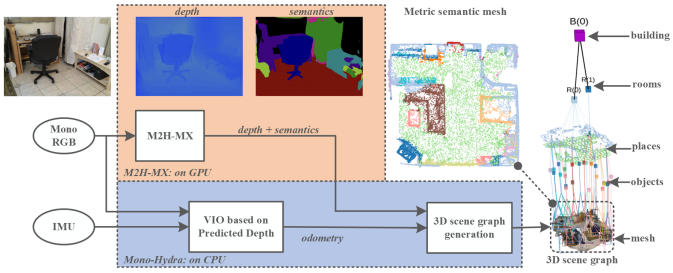


Fig. 1. System overview of M2H-MX as a GPU-based perception front end to the Mono-Hydra [4] monocular SLAM pipeline. From monocular RGB input, M2H-MX predicts dense depth and semantic labels. The predicted depth is used by the CPU-based RGB-inertial odometry and mapping backend, while semantic labels are fused into the metric-semantic map. The shown pipeline example uses ScanNet scene0000_00.

Dense multi-task learning is a natural way to strengthen monocular perception because semantics and geometry provide complementary constraints. PAD-Net [8], MTI-Net [9], InvPT [10], MTMamba [11], and MTMamba++ [12] show that joint prediction of depth, semantic segmentation, surface normals, and boundaries can improve dense scene understanding through cross-task feature sharing. More recently, DINOv3 [13] has provided stronger transferable features for dense prediction, while LoRA [14] and Mamba [15] make large pretrained features more practical for task-specific adaptation and sequence refinement. However, most dense prediction studies are still evaluated mainly with per-frame metrics such as mIoU and RMSE. These metrics do not directly show whether the predictions improve camera tracking, map fusion, or the metric-semantic representation from which a scene graph is built.

This paper addresses that gap with M2H-MX, a monocular dense perception front end for real-time RGB-inertial spatial perception. Given RGB frames, M2H-MX predicts metric depth and semantic labels for mapping, while using

[1]All authors are with the Department of Earth Observation Science, University of Twente, Enschede, 7522 NH, The Netherlands. {b.udugama, george.vosselman, f.nex}@utwente.nl

M2H-MX available at https://github.com/BavanthaU/m2h_mx.

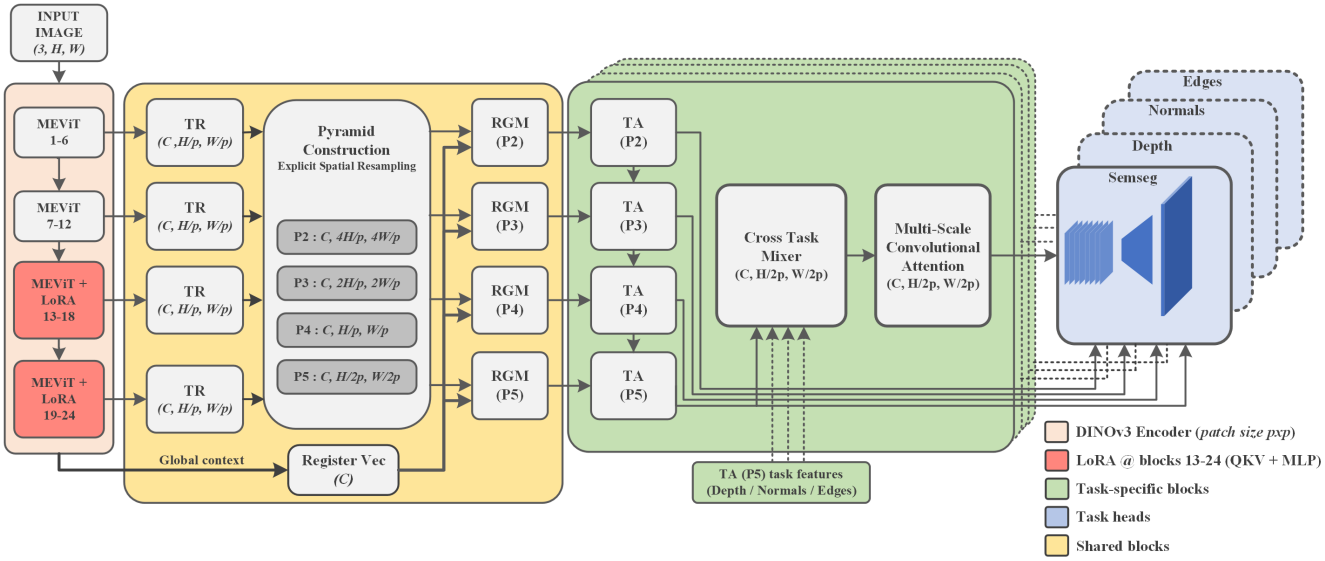


Fig. 2. Overview of M2H-MX. A monocular RGB image is processed by a DINOv3 encoder with Memory-Efficient Vision Transformer (MEViT) blocks and LoRA adaptation in the final transformer blocks. Token reassembly and pyramid construction form multi-scale features. Register-Gated Mamba (RGM) blocks inject global context from register tokens, while Task Adaptors (TA), Cross-Task Mixing (CTM), and Multi-Scale Convolutional Attention (MSCA) produce task-specific features for dense depth, semantic, normal, and edge prediction.

surface normals and edges as auxiliary geometric cues during training. Its design is motivated by map reliability: DINOv3 feature adaptation preserves strong visual features, register-guided multi-scale decoding injects global scene context, and directed cross-task refinement lets semantic and geometric cues reinforce each other while limiting negative transfer. We integrate M2H-MX with the RGB-inertial Mono-Hydra backend to isolate the role of the perception front end in metric-semantic mapping and downstream 3D scene graph construction [4]. Evaluating both dense prediction and deployed mapping, we show improved semantic and geometric accuracy, real-time operation, lower ScanNet trajectory error, and cleaner metric-semantic reconstructions. These results support our central claim: stronger monocular perception can improve the metric-semantic representation needed for reliable RGB-inertial 3D scene graph construction.

II. Methodology

A. M2H-MX Network

M2H-MX maps an RGB frame $I_i \in \mathbb{R}^{3\times H\times W}$ to dense predictions $\{\hat{Y}_i^q\}_{q\in\mathcal{K}}$, where $\mathcal{K} = \{d, s, n, e\}$ denotes depth, semantic segmentation, surface normals, and edges. Depth and semantics are used by the mapping system, while normals and edges provide auxiliary geometric supervision. As shown in Fig. 2, the model follows a compact perception path: DINOv3 feature extraction, token reassembly, pyramid construction, register-guided decoding, directed cross-task refinement, and dense prediction heads.

The encoder is kept mostly frozen to preserve foundation features, while LoRA [14] adapts the later transformer blocks to dense indoor prediction:

$$W_{\text{eff}} = W_0 + \lambda BA. \tag{1}$$

Here, W_0 is the frozen pretrained projection, A and B are trainable low rank matrices, and λ controls the adaptation scale. This provides task adaptation without fully retraining the DINOv3 [13] encoder.

DINOv3 produces token features, but dense mapping requires spatial multi-scale features. We therefore reassemble selected patch tokens into feature maps, aggregate them into a pyramid, and pool the final register tokens into a global context vector:

$$\begin{aligned} \tilde{F}^\ell &= \text{Conv}_{1\times1}^\ell\left(\text{TR}(H_{\text{patch}}^\ell)\right), \quad \ell \in \mathcal{L}, \\ \{p_2, p_3, p_4, p_5\} &= \Psi\left(\Phi(\{\tilde{F}^\ell\}_{\ell\in\mathcal{L}})\right), \\ r &= W_r\left(\frac{1}{R}\sum_{j=1}^{R} h_{\text{reg},j}^{\text{last}}\right). \end{aligned} \tag{2}$$

Here, $\mathcal{L}$ is the set of tapped encoder layers, H_{patch}^ℓ are patch tokens, $\text{TR}(\cdot)$ is token reassembly, $\Phi(\cdot)$ aggregates features, $\Psi(\cdot)$ builds pyramid levels p_2 to p_5, and r is the global register context obtained from R register tokens using projection W_r. The pyramid preserves local boundaries, while r carries room-level context useful for resolving monocular ambiguity.

The decoder updates the pyramid from coarse to fine scales using Register Gated Mamba blocks. At scale $k \in \{5, 4, 3, 2\}$,

$$\begin{aligned} q_k &= \text{Reshape}\left(p_k + \text{Up}(s_{k+1}; p_k)\right), \\ s_k &= \text{RGM}_k\left(q_k \odot \text{sigm}(A_k(r))\right), \quad s_6 = 0. \end{aligned} \tag{3}$$

Here, s_k is the decoded state, q_k is its sequence representation, $\text{Up}(\cdot)$ resamples the coarser state to the resolution of p_k, $A_k(\cdot)$ is a scale-specific projection, $\text{sigm}(\cdot)$ is the sigmoid gate, and $\odot$ denotes elementwise multiplication. This register gate conditions local decoding on global scene layout before map fusion.

Task adaptors convert s_k into task features. For each target task $q \in \mathcal{K}$, the Cross-Task Mixer injects only selected context tasks $\mathcal{C}_q$, followed by multi-scale convolutional attention:

$$\begin{aligned} m_j &= \Pi_j(h_j) \odot (1 + \mathrm{sigm}(G_j(\Pi_j(h_j)))), \\ u_q &= \mathrm{Conv}_{1\times1}\left([h_q, \{m_j\}_{j\in\mathcal{C}_q}]\right), \\ \tilde{h}_q &= u_q + A^q_{\mathrm{spatial}} \odot u_q. \end{aligned} \tag{4}$$

Here, h_q is the target task feature, h_j is a context task feature, m_j is the gated context feature, $\Pi_j(\cdot)$ aligns channels, $G_j(\cdot)$ predicts the context gate, and A^q_{spatial} is the MSCA attention map. This directed interaction lets geometry and semantics reinforce each other while reducing negative transfer.

The prediction heads produce dense outputs:

$$\begin{aligned} \hat{D}(u) &= \sum_{b=1}^{N_b} p_b(u)c_b + (W_o * \tilde{h}_d)(u), \\ \hat{S} &= \mathrm{SemHead}(\tilde{h}_s), \\ \hat{N} &= \mathrm{NormHead}(\tilde{h}_n), \\ \hat{E} &= \mathrm{EdgeHead}(\tilde{h}_e). \end{aligned} \tag{5}$$

Here, u denotes a pixel location, N_b is the number of depth bins, $p_b(u)$ is the predicted probability of depth bin b at u, and c_b is the corresponding bin center. The term $(W_o * \tilde{h}_d)(u)$ is the residual depth offset, and $\hat{E}$ denotes edge logits. At deployment, only $\hat{D}$ and $\hat{S}$ are passed to the mapping pipeline.

The model is trained with task losses, auxiliary supervision, geometric consistency, and uncertainty-based task balancing [16]:

$$\begin{aligned} L_q &= L_q^{\mathrm{main}} + \sum_{k=2}^{5} \alpha_{k,q} L_{k,q}^{\mathrm{aux}}, \\ L_{\mathrm{total}} &= \sum_{q\in\mathcal{K}} \left(\frac{1}{2\eta_q^2} L_q + \log \eta_q \right) + \lambda_{dn} L_{dn}(\hat{D}, \hat{N}) \\ &\quad + \lambda_{se} \|\mathrm{sigm}(\hat{E}) - \phi(\hat{S})\|_1. \end{aligned} \tag{6}$$

Here, η_q is the learned uncertainty for task q, $\alpha_{k,q}$ weights auxiliary supervision at scale k, L_{dn} encourages consistency between depth and normals, and $\phi(\hat{S})$ extracts semantic boundaries from semantic logits.

B. Perception to Mapping and Scene Graph Generation

M2H-MX provides dense inputs to the Mono-Hydra scene graph pipeline [4]. For each RGB frame, predicted depth $\hat{D}$ supports RGB-inertial odometry and metric reconstruction, while semantic labels from $\arg\max_c \hat{S}(u, c)$ are fused into the map. The backend integrates these frame-wise predictions over time to produce a metric-semantic mesh.

The scene graph is then extracted from the fused map: mesh vertices form the geometric layer, semantic regions support object node extraction, free-space samples form place nodes, and higher-level room and building nodes are inferred by the Mono-Hydra backend. Thus, M2H-MX does not directly predict the graph; it improves the depth and semantic evidence from which the mapping and graph layers are constructed.

TABLE I

DENSE PERCEPTION RESULTS ON NYUDV2 AND CITYSCAPES.

NYUDv2			Cityscapes		
Method	mIoU ↑	Depth RMSE ↓	Method	mIoU ↑	Disparity RMSE ↓
TaskPrompter [20]	55.30	0.5152	MTI-Net [9]	59.85	5.06
MQTransformer [21]	54.84	0.5325	InvPT [10]	71.78	4.67
MTMamba [11]	55.82	0.5066	TaskPrompter [20]	72.41	5.49
InvPT-B|MTPD-C [10], [22]	54.86	0.5150	MTMamba [11]	78.00	4.66
MLoRE [23]	55.96	0.5076	MTMamba++ [12]	<u>79.13</u>	<u>4.63</u>
MTMamba++ [12]	57.01	0.4818	**M2H-MX-L**	**82.28**	**3.89**
M2H [24]	61.54	0.4196			
M2H-MX-B	<u>61.80</u>	<u>0.4170</u>			
M2H-MX-L	**65.60**	**0.3800**			

III. EXPERIMENTS

A. Experimental Setup

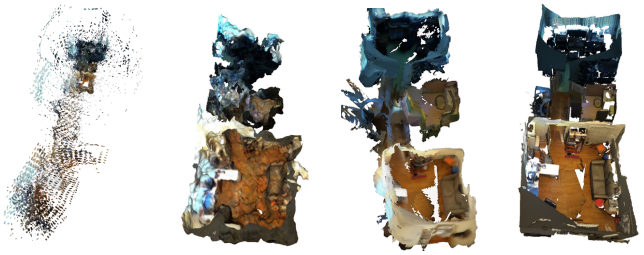
We evaluate M2H-MX at three levels: dense prediction quality, deployed monocular mapping, and architectural ablations. Dense perception is evaluated on NYUDv2 [17] and Cityscapes [18], while deployed mapping is evaluated on ScanNet [19]. Unless otherwise stated, experiments use M2H-MX-L with a DINOv3-ViT-L backbone, decoder width $C = 256$, Mamba state size 32, four register tokens, and 64 depth bins. M2H-MX-B uses DINOv3-ViT-B with the same decoder, heads, and training protocol. For M2H-MX-L, LoRA is applied to the final 12 backbone blocks ($r = 16$, $\alpha = 32$, dropout 0.05), with all other backbone parameters frozen. NYUDv2 uses all four heads, while Cityscapes and ScanNet use only depth and semantics, which are the outputs required by the spatial perception pipeline. ScanNet deployment uses a model trained on ScanNet25k, achieving 76.10 mIoU and 0.2210 depth RMSE. Models are trained on an NVIDIA A40 GPU. Benchmark inference and runtime evaluation use an Intel i7-14700K CPU, 32 GB DDR5 memory, and an NVIDIA RTX 4080 Super GPU, with perception running asynchronously on the GPU and mapping on the CPU. We also test real-time edge-device deployment on a Jetson Orin NX 16 GB with a ZED X One camera.

B. Dense Perception Benchmarks

Table I evaluates whether M2H-MX improves the dense semantic and geometric predictions used by the mapping pipeline. On NYUDv2, M2H-MX-L outperforms the strongest listed prior method, M2H, by $+4.06$ mIoU points and reduces depth RMSE by 9.4%. These gains indicate that M2H-MX improves the two cues most critical for mapping: semantically consistent labels and geometrically accurate depth. On Cityscapes, M2H-MX-L improves over the strongest listed state-of-the-art baseline, MTMamba++, by $+3.15$ mIoU points and reduces disparity RMSE by 16%, showing that the design also generalizes beyond indoor scenes.

C. Real-Time System Evaluation in SLAM

Dense prediction quality is useful for robot autonomy only if it improves the deployed spatial perception pipeline. Table II reports model complexity. The two-head M2H-MX-L deployment uses depth and semantics, the outputs required by the mapping system, and sustains 25–30 Hz



DROID-SLAM [6] GO-SLAM [7] M2H-MX GT

Fig. 3. Qualitative metric reconstruction on ScanNet scene0054_00. M2H-MX with Mono-Hydra produces cleaner geometry and more coherent spatial structure than the monocular baselines.

TABLE II

MODEL COMPLEXITY UNDER DIFFERENT ACTIVE-HEAD SETTINGS.

Method	#P (M)	GFLOPs
Reported baselines		
TaskPrompter [20]	373.00	416
MTMamba++ [12]	315.00	524
M2H [24]	81.00	488
M2H-MX variants (this work)		
M2H-MX-B (4 heads)	134.26	322.67
M2H-MX-L (2 heads)	332.03	371.76
M2H-MX-L (4 heads)	353.53	491.91

in the asynchronous perception-to-mapping loop at 640×480 input resolution on an NVIDIA RTX 4080 Super. For embedded deployment, the same model runs on a Jetson Orin NX 16 GB with TensorRT FP16 and CUDA Graphs at 192×256 input resolution, reaching 25.53 FPS with 39.02 ms mean GPU compute time. This shows that the proposed front end remains practical for real-time monocular deployment despite using foundation model features and cross-task refinement.

Table III reports average ATE on selected ScanNet sequences. All methods in Table III are reported on the same selected ScanNet sequences. RGB-D methods are included as reference points, while the main comparison is against monocular methods because M2H-MX uses only RGB input at the sensor level. Compared with the closest monocular baseline, GO-SLAM [7], the M2H-MX-based Mono-Hydra stack reduces average ATE from 17.59 cm to 6.91 cm. This suggests that improved depth and semantics stabilize tracking and metric-semantic mapping for 3D scene graph construction. Fig. 3 shows that M2H-MX also produces cleaner ScanNet scene0054_00 geometry than the monocular baselines.

D. Ablation Study: Feature Quality vs. Decoder Complexity

Because M2H-MX uses a strong foundation encoder, Table IV separates the effect of backbone choice from the proposed decoding and refinement blocks. The results show that performance does not come from the encoder alone: removing both CTM and MSCA reduces mIoU by 2.07 points and increases depth RMSE, while CTM-only and MSCA-only variants give limited gains. Removing RGM or the register feed also degrades both semantics and depth, showing that global register conditioning is important for coherent dense prediction.

TABLE III

AVERAGE ATE [CM] ON THE SAME SELECTED SCANNET SEQUENCES.

RGB-D method	ATE ↓	Monocular method	ATE ↓
iMAP [25]	56.21	DROID-SLAM (VO) [6]	63.61
NICE-SLAM [26]	13.05	DROID-SLAM [6]	52.60
DROID-SLAM (VO) [6]	11.59	GO-SLAM [7]	17.59
DROID-SLAM [6]	7.15	**Mono-Hydra + M2H-MX [4]**	**6.91**
GO-SLAM [7]	7.02		

TABLE IV

ABLATION RESULTS ON NYUDV2 RELATIVE TO M2H-MX-L.

Variant	mIoU ↑	RMSE ↓	Δ mIoU	Δ RMSE
M2H-MX-L	65.60	0.3800	–	–
Component ablations				
No CTM/MSCA	63.53	0.4619	-2.07	+0.0819
CTM only	63.55	0.4705	-2.05	+0.0905
MSCA only	63.53	0.4575	-2.07	+0.0775
w/o RGM	64.44	0.4346	-1.16	+0.0546
w/o reg. feed	64.22	0.4473	-1.38	+0.0673
Different backbones				
DINOv2-L	56.79	0.5131	-8.81	+0.1331
ConvNeXt-L	38.83	0.6940	-26.77	+0.3140

Backbone sensitivity. Table IV shows that encoder feature quality matters, but the gains are tied to the proposed decoder design rather than the encoder alone. Replacing DINOv3 with ConvNeXt causes a large degradation, indicating that M2H-MX is designed to exploit dense transformer features and register context rather than generic convolutional features. Replacing DINOv3 with DINOv2 still causes an 8.81-point mIoU drop and a 35.0% RMSE increase. This is consistent with DINOv3's Gram anchoring strategy, which encourages cleaner and more spatially coherent dense features than DINOv2 [13]. M2H-MX is designed to exploit these features through register-guided decoding and directed cross-task refinement, yielding a compact decoder that preserves dense prediction accuracy while remaining suitable for embedded deployment.

IV. CONCLUSION

This paper presented M2H-MX, a foundation-model-based dense multi-task perception front end for real-time monocular spatial understanding, metric-semantic mapping, and downstream 3D scene graph construction. The results show stronger dense prediction on NYUDv2 and Cityscapes, a 25–30 Hz deployed perception-to-mapping loop on an RTX 4080 Super, lower ScanNet trajectory error, and cleaner metric-semantic maps than existing monocular baselines. The ablation study further shows that the decoder is closely matched to DINOv3 features, particularly spatially coherent transformer representations and global register context. These findings suggest that improving the semantic and geometric front end can strengthen the monocular spatial representations needed for reliable robot autonomy. Future work will evaluate these scene graph representations for autonomous exploration, where object, region, and relation cues can support informative viewpoint selection, frontier reasoning, and semantically meaningful map expansion.

REFERENCES

[1] I. Armeni, Z.-Y. He, J. Gwak, A. R. Zamir, M. Fischer, J. Malik, and S. Savarese, "3d scene graph: A structure for unified semantics, 3d space, and camera," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 5664–5673.

[2] A. Rosinol, A. Violette, M. Abate, N. Hughes, Y. Chang, J. Shi, A. Gupta, and L. Carlone, "Kimera: From slam to spatial perception with 3d dynamic scene graphs," *The International Journal of Robotics Research*, vol. 40, no. 12-14, pp. 1510–1546, 2021.

[3] N. Hughes, Y. Chang, and L. Carlone, "Hydra: A real-time spatial perception system for 3d scene graph construction and optimization," *arXiv preprint arXiv:2201.13360*, 2022.

[4] U. Udugama, G. Vosselman, and F. Nex, "Mono-hydra real-time 3d scene graph construction from monocular camera input with imu," *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. X-1/W1-2023, pp. 439–445, 2023.

[5] A. Rosinol, M. Abate, Y. Chang, and L. Carlone, "Kimera: an open-source library for real-time metric-semantic localization and mapping," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 1689–1696.

[6] Z. Teed and J. Deng, "DROID-SLAM: Deep Visual SLAM for Monocular, Stereo, and RGB-D Cameras," *Advances in neural information processing systems*, 2021.

[7] Y. Zhang, F. Tosi, S. Mattoccia, and M. Poggi, "Go-slam: Global optimization for consistent 3d instant reconstruction," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2023.

[8] D. Xu, W. Ouyang, X. Wang, and N. Sebe, "Pad-net: Multi-tasks guided prediction-and-distillation network for simultaneous depth estimation and scene parsing," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 675–684.

[9] S. Vandenhende, S. Georgoulis, and L. Van Gool, "Mti-net: Multi-scale task interaction networks for multi-task learning," in *European conference on computer vision*. Springer, 2020, pp. 527–543.

[10] H. Ye and D. Xu, "Inverted pyramid multi-task transformer for dense scene understanding," in *European Conference on Computer Vision*. Springer, 2022, pp. 514–530.

[11] B. Lin, W. Jiang, P. Chen, Y. Zhang, S. Liu, and Y.-C. Chen, "Mtmamba: Enhancing multi-task dense scene understanding by mamba-based decoders," in *European Conference on Computer Vision*. Springer, 2024, pp. 314–330.

[12] B. Lin, W. Jiang, P. Chen, S. Liu, and Y.-C. Chen, "Mtmamba++: Enhancing multi-task dense scene understanding via mamba-based decoders," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025.

[13] O. Siméoni, H. V. Vo, M. Seitzer, F. Baldassarre, M. Oquab, C. Jose, V. Khalidov, M. Szafraniec, S. Yi, M. Ramamonjisoa, F. Massa, D. Haziza, L. Wehrstedt, J. Wang, T. Darcet, T. Moutakanni, L. Sentana, C. Roberts, A. Vedaldi, J. Tolan, J. Brandt, C. Couprie, J. Mairal, H. Jégou, P. Labatut, and P. Bojanowski, "DINOv3," 2025. [Online]. Available: https://arxiv.org/abs/2508.10104

[14] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen, "Lora: Low-rank adaptation of large language models," in *International Conference on Learning Representations*, 2022.

[15] A. Gu and T. Dao, "Mamba: Linear-time sequence modeling with selective state spaces," *arXiv preprint arXiv:2312.00752*, 2023.

[16] A. Kendall, Y. Gal, and R. 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*, 2018, pp. 7482–7491.

[17] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus, "Indoor segmentation and support inference from rgbd images," in *European Conference on Computer Vision*. Springer, 2012, pp. 746–760.

[18] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The cityscapes dataset for semantic urban scene understanding," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 3213–3223.

[19] A. Dai, A. X. Chang, M. Savva, M. Halber, T. Funkhouser, and M. Nießner, "Scannet: Richly-annotated 3d reconstructions of indoor scenes," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 5828–5839.

[20] H. Ye and D. Xu, "Taskprompter: Spatial-channel multi-task prompting for dense scene understanding," in *The Eleventh International Conference on Learning Representations (ICLR)*, 2023. [Online]. Available: https://openreview.net/forum?id=-CwPopPJda

[21] Y. Xu, X. Li, H. Yuan, Y. Yang, and L. Zhang, "Multi-task learning with multi-query transformer for dense prediction," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 34, no. 2, pp. 1228–1240, 2024.

[22] Y. Shang, D. Xu, G. Liu, R. R. Kompella, and Y. Yan, "Efficient multitask dense predictor via binarization," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024, pp. 15 899–15 908.

[23] Y. Yang, P.-T. Jiang, Q. Hou, H. Zhang, J. Chen, and B. Li, "Multi-task dense prediction via mixture of low-rank experts," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2024, pp. 27 927–27 937.

[24] U. Udugama, G. Vosselman, and F. Nex, "M2h: Multi-task learning with efficient window-based cross-task attention for monocular spatial perception," in *2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2025, pp. 8067–8072.

[25] E. Sucar, S. Liu, J. Ortiz, and A. Davison, "iMAP: Implicit mapping and positioning in real-time," in *Proceedings of the International Conference on Computer Vision (ICCV)*, 2021.

[26] 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," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.