VEOcc++: Learning Persistent Voxel Feature Maps for Embodied Online Semantic Occupancy Mapping and Offline Refinement

Anonymous authors
Paper under double-blind review

Abstract

Semantic occupancy mapping supports embodied perception during exploration and detailed scene understanding afterward, giving rise to two complementary requirements: online incremental mapping for maintaining an up-to-date global map under causal constraints, and offline global map refinement for improving the completed scene representation with additional computation. We present VEOcc++, a unified voxel-centric framework that connects both stages through a persistent sparse voxel feature map. During exploration, the map expands with newly observed regions and integrates aligned local features through reliability-guided updates, while context-aware decoding produces the current semantic occupancy map. After exploration, cached local features are consolidated as complementary evidence and combined with the completed online representation for multi-scale spatial refinement. On EmbodiedOcc-ScanNet, VEOcc++ achieves 67.12% IoU and 58.40% mIoU, establishing new state-of-the-art performance for online incremental mapping. With the local predictor fixed, our online mapping pipeline outperforms VEOcc fusion by 3.70 IoU and 3.74 mIoU points, demonstrating substantial gains from the proposed global mapping design. Offline global map refinement further improves the completed map to 67.79% IoU and 59.37% mIoU using only the observations acquired during exploration. Zero-shot deployment on a mobile robot further demonstrates transferability to unseen environments, onboard feasibility, and scalability to large-scale indoor mapping.

1 Introduction

3D scene understanding (Savva et al., 2019; Gu et al., 2024; Rana et al., 2023; Huang et al., 2023) is a fundamental capability for embodied agents. 3D semantic occupancy maps provide an effective representation by jointly modeling spatial occupancy and semantic categories in a fine-grained volumetric space (McCormac et al., 2017; Yang et al., 2017; Song et al., 2017; Cao & de Charette, 2022). Practical embodied applications place different demands on these maps during and after exploration. During exploration, observations are sequentially integrated under causal constraints to maintain an up-to-date global map for perception and decision making. After exploration, the accumulated map and retained evidence can be processed with additional computation to improve the final scene representation. As shown in Fig. 1, we therefore study *online incremental mapping* and *offline global map refinement* as complementary stages of embodied semantic occupancy mapping.

Recent studies have extended semantic occupancy prediction from individual observations to sequential mapping. Gaussian-centric approaches progressively update scene-level representations (Wu et al., 2025; Wang et al., 2025; Zhang et al., 2025; Qian et al., 2026; Zhou et al., 2026a;b), whereas VEOcc (Wang et al., 2026) directly maintains an expanding global voxel map. VEOcc performs long-term accumulation at the semantic prediction level, without retaining the intermediate voxel features for subsequent integration and spatial decoding. Beyond the choice of accumulated representation, incorporating incoming observations and refining a completed map involve different processing objectives. Sequential updates prioritize timely integration, while refinement can devote dedicated computation to broader spatial context across the accumulated scene. These considerations motivate a persistent feature representation that supports both online accumulation and subsequent global map refinement.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

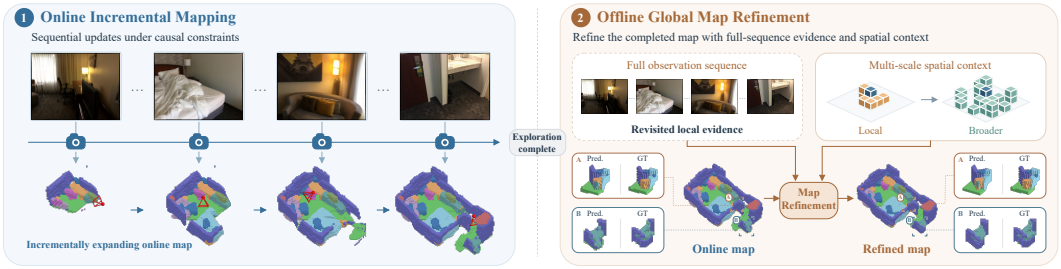

Figure 1: **Overview of the two-stage mapping setting in VEOcc++.** Online incremental mapping sequentially integrates observations under causal constraints, while offline global map refinement exploits complementary local evidence and broader spatial context over the completed map. Zoomed examples highlight representative improvements after refinement.

To this end, we present VEOcc++, a unified voxel-centric framework that connects both stages through a shared Persistent Voxel Feature Map. This sparse map expands with exploration while retaining feature states independently of their semantic readout. Reliability-Guided Incremental Update learns to combine aligned local features with historical voxel states, while Context-Aware Decoding incorporates neighboring voxel context to produce the current occupancy map. After exploration, Local Evidence Aggregation consolidates cached local features as complementary evidence. Global Spatial Refinement then combines this evidence with the completed feature map and its online semantic probabilities for multi-scale spatial reasoning. The resulting refinement improves the final map without acquiring additional observations or expanding the established map support.

Experiments on EmbodiedOcc-ScanNet demonstrate that the online stage alone outperforms all compared methods, achieving 67.12% IoU and 58.40% mIoU. With the local predictor held fixed, our online mapping pipeline improves IoU and mIoU by 3.70 and 3.74 percentage points over VEOcc fusion, respectively, demonstrating gains beyond improvements to the local predictor. Offline global map refinement further raises the results to 67.79% IoU and 59.37% mIoU. Beyond benchmark evaluation, zero-shot deployment on a mobile robot demonstrates transferability to unseen real-world environments, onboard feasibility, and scalability to large-scale indoor mapping.

Our main contributions are summarized as follows:

- We develop a voxel-centric framework built upon a Persistent Voxel Feature Map that connects online incremental mapping with offline global map refinement. Reliability-guided updates and context-aware decoding enable feature-level scene accumulation and dynamic map expansion over time.
- We introduce offline global map refinement that combines cached local evidence with the completed online representation for multi-scale spatial reasoning, improving occupancy and semantic predictions without additional observations or changes to the established map support during refinement.
- Experiments demonstrate state-of-the-art online performance, and further improvements after refinement. Zero-shot robotic deployment demonstrates transferability, onboard feasibility, and scalability to large scale indoor environments.

2 Related Work

2.1 Embodied Semantic Occupancy Mapping

Embodied semantic occupancy mapping extends frame-wise occupancy prediction to sequential scene construction during exploration. EmbodiedOcc (Wu et al., 2025) introduces this setting with a Gaussian-centric global representation, followed by methods that improve geometry modeling, temporal fusion, and representation efficiency (Wang et al., 2025; Zhang et al., 2025; Qian et al., 2026; Zhou et al., 2026a;b). Recent works further explore persistent Gaussian feature memory and long-horizon semantic mapping (Qian et al., 2026; Zhu et al., 2026), while FreeOcc (Jiang et al., 2026) studies training-free open-vocabulary mapping. In contrast, VEOcc (Wang et al., 2026) adopts a voxel-centric formulation with an incrementally expanding global map. VEOcc++ follows this di-

rection but performs long-term accumulation in a persistent voxel feature space and further connects online incremental mapping with offline global map refinement.

2.2 Voxel-Centric Persistent Scene Memory for Sequential 3D Perception

Voxel-centric persistent representations have been explored for sequential 3D perception. NeuralRecon (Sun et al., 2021) and NeuralFusion (Weder et al., 2021) maintain latent volumetric states for incremental reconstruction and fusion, while ST-Occ (Leng et al., 2025), StreamOcc (Moon et al., 2026), and LMPOcc (Yuan et al., 2026) extend related ideas to occupancy prediction through spatiotemporal memory, recurrent voxel feature propagation, and global occupancy priors. VEOcc (Wang et al., 2026) adopts an incrementally expanding voxel map for embodied semantic occupancy, but performs long-term integration at the prediction level. OccFiner (Shi et al., 2025) focuses on offboard occupancy refinement over completed prediction sequences, rather than causal online map maintenance. In contrast, VEOcc++ maintains a persistent sparse voxel feature map as the shared scene representation for causal online mapping and post-exploration refinement, preserving latent scene information beyond semantic readout throughout both stages.

3 Methodology

3.1 Problem Formulation

Given a monocular image $\mathbf{I}_t \in \mathbb{R}^{H\times W\times 3}$ and geometric information $\mathbf{G}_t$, including camera intrinsics, pose, and local voxel geometry, VEOcc++ produces local and global semantic occupancy predictions through

$$\begin{aligned} \mathbf{Y}_t^{\text{local}} &= \mathcal{F}_{\text{local}}(\mathbf{I}_t, \mathbf{G}_t), \\ (\mathcal{O}_t, \mathbf{Y}_t^{\text{on}}) &= \mathcal{F}_{\text{online}}(\mathcal{O}_{t-1}, \mathbf{I}_t, \mathbf{G}_t), \\ \mathbf{Y}_T^{\text{off}} &= \mathcal{F}_{\text{offline}}\left(\mathcal{O}_T, \{(\mathbf{I}_t, \mathbf{G}_t)\}_{t=1}^{T}\right). \end{aligned} \tag{1}$$

Here, $\mathbf{Y}_t^{\text{local}}$ is the frame-wise local prediction, and $\mathcal{O}_t$ is the persistent global scene state from which the online map $\mathbf{Y}_t^{\text{on}}$ is decoded. Online incremental mapping processes observations sequentially under causal constraints and expands the map support as new regions are observed. After exploration ends at timestep T, offline global map refinement uses the completed state and evidence retained from the same sequence to produce $\mathbf{Y}_T^{\text{off}}$ on the established support.

3.2 Method Overview

As illustrated in Fig. 2, a Persistent Voxel Feature Map connects the online and offline stages. The local predictor (Sec. 3.3) extracts voxel features from each frame. Online incremental mapping (Sec. 3.4) aligns and integrates these features into the persistent map, from which semantic occupancy is decoded. Offline global map refinement (Sec. 3.5) subsequently combines the completed online representation with cached local evidence for multi-scale spatial refinement.

3.3 Local Occupancy Prediction

We adopt the local predictor of VEOcc (Wang et al., 2026) without architectural modifications. Given $\mathbf{I}_t$ and $\mathbf{G}_t$, it extracts image features and incorporates a monocular depth prior (Yang et al., 2024) to produce a local voxel feature volume $\mathbf{F}_t$ and the local occupancy prediction $\mathbf{Y}_t^{\text{local}}$. The features before the occupancy head are retained for online accumulation and offline evidence aggregation.

3.4 Online Incremental Mapping

Persistent Voxel Feature Map. The global scene state $\mathcal{O}_t$ comprises a persistent sparse voxel feature map $\mathcal{M}_t = \{(k, \mathbf{h}_t(k)) \mid k \in \mathcal{K}_t\}$ and auxiliary voxel-wise reliability statistics. Here, $\mathcal{K}_t$ contains the stored voxel coordinates in the world frame, and $\mathbf{h}_t(k) \in \mathbb{R}^C$ is the feature associated with voxel k. The map retains accumulated scene features independently of their semantic readout, allowing the same representation to support subsequent online updates and offline refinement. Its support grows with exploration without requiring a predefined global scene boundary.

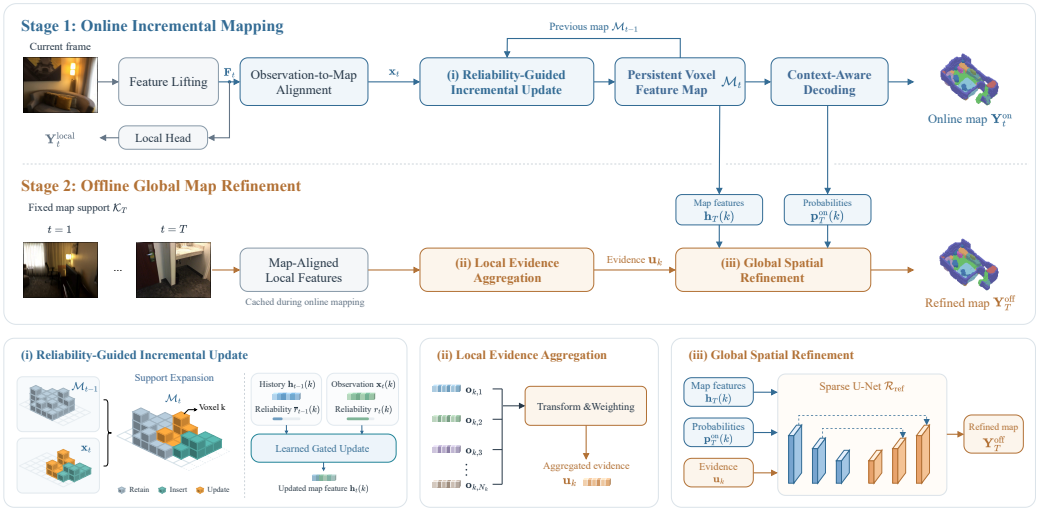

Figure 2: **Overall framework of VEOcc++.** During online incremental mapping, local voxel features are aligned with the global map and accumulated in a Persistent Voxel Feature Map through reliability-guided updates, while Context-Aware Decoding produces the current occupancy map. During offline global map refinement, Local Evidence Aggregation consolidates cached local features as complementary evidence, which is combined with the completed online representation for Global Spatial Refinement.

Observation-to-Map Alignment. For each incoming frame, we select geometrically visible local voxels and apply a lightweight pose-conditioned adaptation to their features. Let $\widehat{\mathbf{F}}_t$ denote the adapted local features, $\mathcal{S}_t$ the selected voxel set, and $\mathbf{w}_{t,i}$ the world coordinate of voxel $i \in \mathcal{S}_t$. With global voxel size δ, quantization gives the active map coordinates $\mathcal{A}_t = \{\lfloor \mathbf{w}_{t,i}/\delta \rfloor \mid i \in \mathcal{S}_t\}$.

To account for misalignment between the local and global voxel lattices, we sample features at each active global voxel center $\mathbf{q}_k = \delta(k + \frac{1}{2})$:

$$\mathbf{x}_t(k) = \text{TriSample}\left(\widehat{\mathbf{F}}_t, \mathbf{q}_k; \mathbf{G}_t\right), \qquad k \in \mathcal{A}_t. \tag{2}$$

The sampling operator uses $\mathbf{G}_t$ to express the world-coordinate query in the local voxel lattice before trilinear interpolation. The resulting $\mathbf{x}_t(k)$ is the aligned observation feature. Pose-conditioned adaptation is detailed in Appendix A.2.

Reliability-Guided Incremental Update. Given $\mathcal{A}_t$, we expand the map support and update its feature states as

$$\mathcal{K}_t = \mathcal{K}_{t-1} \cup \mathcal{A}_t, \qquad \mathbf{h}_t(k) = \begin{cases} \mathbf{x}_t(k), & k \in \mathcal{A}_t \setminus \mathcal{K}_{t-1}, \\ \mathcal{U}(\mathbf{h}_{t-1}(k), \mathbf{x}_t(k)), & k \in \mathcal{A}_t \cap \mathcal{K}_{t-1}, \\ \mathbf{h}_{t-1}(k), & k \in \mathcal{K}_{t-1} \setminus \mathcal{A}_t. \end{cases} \tag{3}$$

New voxels are initialized from the aligned features, while voxels outside the current observation retain their previous states. For re-observed voxels, $\mathcal{U}$ adapts the incoming feature and determines its contribution to the persistent state. Following VEOcc (Wang et al., 2026), we estimate the current reliability $r_t(k)$ from depth and image-boundary cues and maintain the historical average reliability $\bar{r}_{t-1}(k)$. Together with semantic uncertainty $E_t(k)$ from the local prediction and relations between the historical and current features, these quantities form an update descriptor $\boldsymbol{\xi}_t(k)$. Its complete construction is provided in Appendix A.3.

The descriptor conditions a corrected candidate feature and a channel-wise update gate:

$$\begin{aligned} \widetilde{\mathbf{x}}_t(k) &= \mathbf{x}_t(k) + \Delta_c(\boldsymbol{\xi}_t(k)), \\ \mathbf{g}_t(k) &= \sigma\left(\text{logit}\left(\frac{r_t(k)}{r_t(k) + \bar{r}_{t-1}(k) + \epsilon}\right) + \Delta_g(\boldsymbol{\xi}_t(k))\right). \end{aligned} \tag{4}$$

Here, Δ_c and Δ_g are learned residual functions. The candidate branch adjusts the incoming feature, while the gate balances its contribution against the historical state. The reliability ratio anchors this

balance to observation quality, and Δ_g learns descriptor-dependent corrections. The resulting update is

$$\mathcal{U}(\mathbf{h}_{t-1}(k), \mathbf{x}_t(k)) = (1 - \mathbf{g}_t(k)) \odot \mathbf{h}_{t-1}(k) + \mathbf{g}_t(k) \odot \widetilde{\mathbf{x}}_t(k). \tag{5}$$

Context-Aware Decoding. Feature updates are voxel-wise, whereas occupancy prediction also benefits from interactions among neighboring map voxels. A lightweight sparse convolutional head $\mathcal{H}_{\text{map}}$ therefore decodes persistent features over the entire current support $\mathcal{K}_t$ into semantic logits $\mathbf{L}_t^{\text{on}}$, yielding semantic probabilities $\mathbf{P}_t^{\text{on}}$ and the online occupancy map $\mathbf{Y}_t^{\text{on}}$, with the former retained for subsequent offline refinement.

3.5 Offline Global Map Refinement

After exploration, we refine the completed online representation using cached evidence from the same image sequence. The spatial support $\mathcal{K}_T$ remains fixed, while occupancy and semantic predictions are revised through multi-scale spatial processing. This offline refinement consists of Local Evidence Aggregation and Global Spatial Refinement, described below.

Local Evidence Aggregation. Local voxel features and their geometry are cached during online processing. After exploration, we retrieve these features and align them to $\mathcal{K}_T$ without rerunning the local predictor. For each mapped voxel k, the aligned features form $\mathcal{V}_k = \{\mathbf{o}_{k,j}\}_{j=1}^{N_k}$, where $\mathbf{o}_{k,j} \in \mathbb{R}^C$ and N_k is the number of valid features. Each feature undergoes a residual transformation and receives a scalar weight normalized over the observations associated with the same voxel:

$$\mathbf{v}_{k,j} = \mathbf{o}_{k,j} + \psi(\mathbf{o}_{k,j}), \qquad w_{k,j} = \text{Softmax}_j(\rho(\mathbf{v}_{k,j})), \qquad \mathbf{u}_k = \sum_{j=1}^{N_k} w_{k,j}\mathbf{v}_{k,j}. \tag{6}$$

Here, ψ and ρ are lightweight MLPs. The aggregated feature $\mathbf{u}_k$ supplies complementary local evidence for global refinement.

Global Spatial Refinement. For each mapped voxel, we concatenate the persistent feature, online semantic probability, and aggregated local evidence as $\mathbf{z}_k = [\mathbf{h}_T(k), \mathbf{p}_T^{\text{on}}(k), \mathbf{u}_k]$, where $\mathbf{p}_T^{\text{on}}(k) = \text{Softmax}(\mathbf{L}_T^{\text{on}}(k))$. These inputs provide the accumulated scene features, their current semantic interpretation, and complementary frame-wise evidence, respectively. A sparse U-Net combines them across spatial scales to predict

$$\mathbf{L}_T^{\text{off}} = \mathcal{R}_{\text{ref}}\left(\{\mathbf{z}_k\}_{k \in \mathcal{K}_T}\right). \tag{7}$$

Compared with the lightweight online decoder, this multi-scale processing incorporates broader spatial context into the prediction of each mapped voxel. The refined features are directly decoded into final semantic logits $\mathbf{L}_T^{\text{off}}$. Semantic classification then produces $\mathbf{Y}_T^{\text{off}}$, updating occupancy and category labels while preserving the spatial support.

4 Experiments

4.1 Experimental Setup

Datasets and Metrics. We evaluate VEOcc++ on two benchmarks: Occ-ScanNet (Yu et al., 2025) for local occupancy prediction and EmbodiedOcc-ScanNet (Wu et al., 2025) for online incremental mapping and offline global map refinement. Occ-ScanNet provides voxelized annotations in a $60 \times 60 \times 36$ grid covering a $4.8\,\text{m} \times 4.8\,\text{m} \times 2.88\,\text{m}$ local volume, while EmbodiedOcc-ScanNet extends the annotations to complete scenes for sequential global mapping evaluation. We report occupancy IoU and semantic mIoU, together with runtime and memory usage for efficiency evaluation.

Implementation Details. VEOcc++ is trained in three stages. We first train the VEOcc local occupancy predictor with a larger learning rate while keeping its architecture unchanged. We then freeze the local predictor and train the online incremental mapping modules. Finally, both the local predictor and online mapper are fixed, and only the refinement modules are optimized on completed scene maps. Training is conducted on four NVIDIA RTX 3090 GPUs and takes approximately 28 hours in total. Additional training and implementation details are provided in Appendix B.

Table 1: **Local Occupancy Prediction** results of embodied occupancy mapping methods on Occ-ScanNet. The best results are highlighted in **bold**, and the second-best results are underlined.

Method	IoU	mIoU	ceiling	floor	wall	window	chair	bed	sofa	table	tvs	furniture	objects
EmbodiedOcc (Wu et al., 2025)	53.95	45.48	40.90	50.80	41.90	33.00	41.20	55.20	61.90	43.80	35.40	53.50	42.90
EmbodiedOcc++ (Wang et al., 2025)	54.90	46.20	36.40	53.10	41.80	34.40	42.90	57.30	64.10	45.20	34.80	54.20	44.10
RoboOcc (Zhang et al., 2025)	56.48	47.67	45.36	53.49	44.35	34.81	43.38	56.93	63.35	46.35	36.12	55.48	44.78
TGSFormer (Qian et al., 2026)	64.42	54.73	54.20	61.30	50.50	41.30	49.50	64.50	69.30	52.40	47.10	63.00	48.80
GPOcc (VGGT) (Zhou et al., 2026a)	63.14	56.19	51.67	59.93	**52.07**	46.44	51.35	64.45	69.47	54.30	**51.76**	63.29	53.36
GPOcc++ (VGGT) (Zhou et al., 2026b)	63.72	57.41	**56.62**	61.40	50.63	**48.13**	**53.12**	66.34	71.05	**56.48**	48.95	64.48	**54.22**
VEOcc (Wang et al., 2026)	64.55	55.49	53.20	64.00	49.00	42.90	50.80	65.50	70.50	53.70	48.30	62.60	50.00
VEOcc++(Ours)	**66.98**	**58.17**	55.70	**67.30**	51.60	46.00	53.00	**67.40**	**72.70**	56.10	51.20	**65.40**	53.30

Table 2: **Global Occupancy Mapping** results on EmbodiedOcc-ScanNet. The best results are highlighted in **bold**, and the second-best results are underlined. † denotes VEOcc using the same local predictor as VEOcc++.

Method	IoU	mIoU	ceiling	floor	wall	window	chair	bed	sofa	table	tvs	furniture	objects
EmbodiedOcc (Wu et al., 2025)	51.52	42.53	22.70	44.60	37.40	38.00	50.10	56.70	59.70	35.40	38.40	52.00	32.90
EmbodiedOcc++ (Wang et al., 2025)	52.20	43.60	27.90	43.90	38.70	40.60	49.00	57.90	59.20	36.80	37.80	53.50	34.10
RoboOcc (Zhang et al., 2025)	53.30	44.05	21.94	44.57	39.54	38.48	51.28	57.04	63.09	36.70	43.05	54.42	34.38
TGSFormer (Qian et al., 2026)	54.42	45.29	31.00	39.90	39.90	32.70	52.00	66.10	53.00	47.30	38.70	59.60	37.90
GPOcc (VGGT) (Zhou et al., 2026a)	61.41	55.39	42.61	51.35	51.49	48.72	54.32	67.91	70.73	52.94	54.75	64.76	49.67
GPOcc++ (VGGT) (Zhou et al., 2026b)	62.50	57.41	**48.00**	52.14	**54.70**	**53.39**	54.71	68.82	71.03	54.27	**55.23**	66.81	52.42
VEOcc (Wang et al., 2026)	62.21	53.00	29.90	62.60	45.60	44.60	54.10	68.10	69.70	54.30	45.90	61.60	46.40
VEOcc† (Wang et al., 2026)	63.42	54.66	33.20	63.40	47.00	46.60	55.40	69.00	71.60	55.90	47.10	63.40	48.60
VEOcc++ (Ours, Online)	67.12	58.40	43.09	66.87	51.95	49.11	57.87	70.49	73.99	60.01	48.44	67.28	53.34
VEOcc++ (Ours, Offline)	**67.79**	**59.37**	42.77	**68.76**	52.12	49.51	**58.68**	**70.74**	**74.65**	**61.48**	52.37	**68.01**	**54.01**

4.2 Results on Public Benchmarks

Local Occupancy Prediction. We first evaluate frame-wise occupancy prediction on Occ-ScanNet. Increasing the learning rate improves the unchanged local predictor to 66.98% IoU and 58.17% mIoU, achieving the best performance among the compared embodied occupancy mapping methods in Table 1. Further analysis of the learning rate is provided in Appendix C.1. The resulting local features provide a strong basis for subsequent online mapping and offline refinement.

Global Occupancy Mapping. **(i) Online Mapping Performance.** As shown in Table 2, VEOcc++ establishes new state-of-the-art performance on EmbodiedOcc-ScanNet, achieving 67.12% IoU and 58.40% mIoU in the online setting. It surpasses GPOcc++ by 4.62 IoU and 0.99 mIoU points, despite GPOcc++ adopting a substantially larger VGGT backbone with large-scale pretraining. The strong performance obtained before any offline refinement highlights the effectiveness of our online incremental mapping design. **(ii) Mapping Gains with a Fixed Local Predictor.** To isolate the contribution of global mapping, we additionally report VEOcc†, which uses the same local predictor as VEOcc++ while retaining the analytic prediction-level fusion of VEOcc. This stronger local predictor improves VEOcc from 62.21/53.00 to 63.42/54.66 IoU/mIoU. With the local predictor fixed, VEOcc++ further reaches 67.12/58.40, yielding gains of 3.70 IoU and 3.74 mIoU points. Improvements are observed across all 11 semantic categories, demonstrating a substantial advantage of the proposed online mapping pipeline over the VEOcc formulation. **(iii) Offline Refinement.** Starting from the completed online map, offline global map refinement further improves performance to 67.79% IoU and 59.37% mIoU. Under the same observations and fixed spatial support, our offline refinement yields gains of 0.67 IoU and 0.97 mIoU points over the online result, with improvements across nearly all semantic categories. These consistent gains demonstrate that the proposed refinement effectively exploits previously acquired evidence to enhance global map quality without additional observations or spatial expansion.

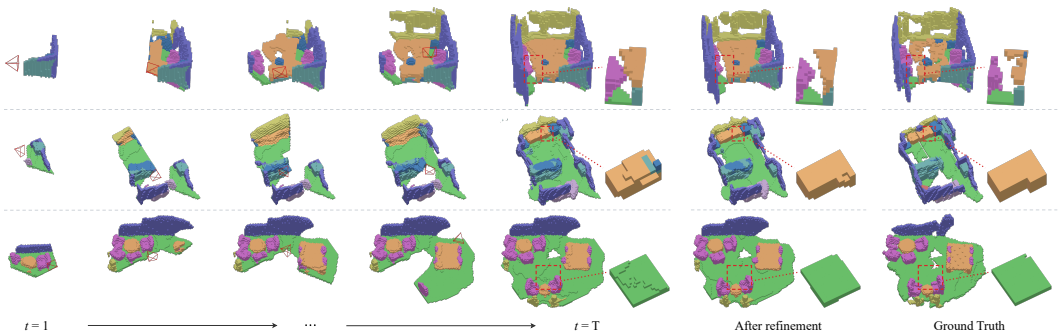

Figure 3: **Progressive online incremental mapping and offline map refinement.** Each row shows the map evolution from $t = 1$ to $t = T$, followed by the refined result and ground truth. Red dashed boxes indicate the enlarged regions.

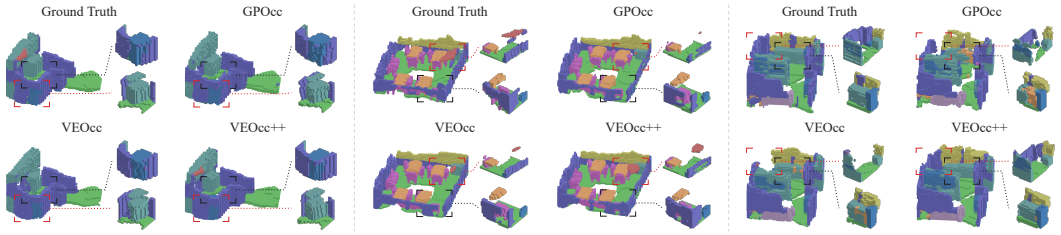

Figure 4: **Qualitative comparison of online incremental mapping.** Highlighted regions are enlarged for detailed comparison.

Qualitative Results. Fig. 3 illustrates progressive map construction during exploration and subsequent offline refinement. The online map expands as observations arrive, while offline global map refinement reduces residual geometric and semantic artifacts in the highlighted regions. Fig. 4 further compares online incremental mapping with GPOcc and VEOcc on representative EmbodiedOcc-ScanNet scenes. VEOcc++ produces more complete object structures and cleaner semantic boundaries, complementing the quantitative global mapping results.

Table 3: **Efficiency and precision comparison** on EmbodiedOcc-ScanNet. Runtime and GPU memory are measured on a single RTX 3090.

Setting	Online s/frame	Online s/scene	Offline s/scene	Online GPU GiB	Offline GPU GiB	Map B/voxel	Online IoU	Online mIoU	Offline IoU	Offline mIoU
FP32	0.152	4.57	0.319	1.753	3.169	424	67.12	58.40	67.79	59.37
Mixed FP16	0.120	3.60	0.224	1.558	3.252	232	67.08	58.36	67.77	59.35

Efficiency Analysis. We evaluate VEOcc++ in full precision on a single RTX 3090. As shown in Table 3, the online incremental mapping pipeline averages 152.44 ms/frame, corresponding to approximately 6.6 FPS, while offline global map refinement averages 319.03 ms per 30-frame scene. This refinement cost is only about 7.0% of the cumulative online processing time for the same scene. The persistent map requires 424 bytes per voxel, with moderate GPU memory peaks in both stages, providing a practical full-precision deployment baseline. We further evaluate a mixed-FP16 deployment configuration. As shown in Table 3, this configuration reduces the overall deployment cost with negligible accuracy degradation and is therefore adopted for the subsequent real-world experiments. The slightly higher offline GPU memory peak mainly reflects the additional integer indexing overhead of the Native sparse-convolution backend, which outweighs the memory savings from half-precision features.

4.3 Zero-Shot Real-World Robotic Experiments

Robotic Platform. We evaluate zero-shot transfer and onboard feasibility on an AgileX UMR mobile robot equipped with an Intel RealSense D435i, a Livox MID-360 LiDAR with IMU for localization, and an NVIDIA Jetson Orin 16GB for computation (Fig. 6).

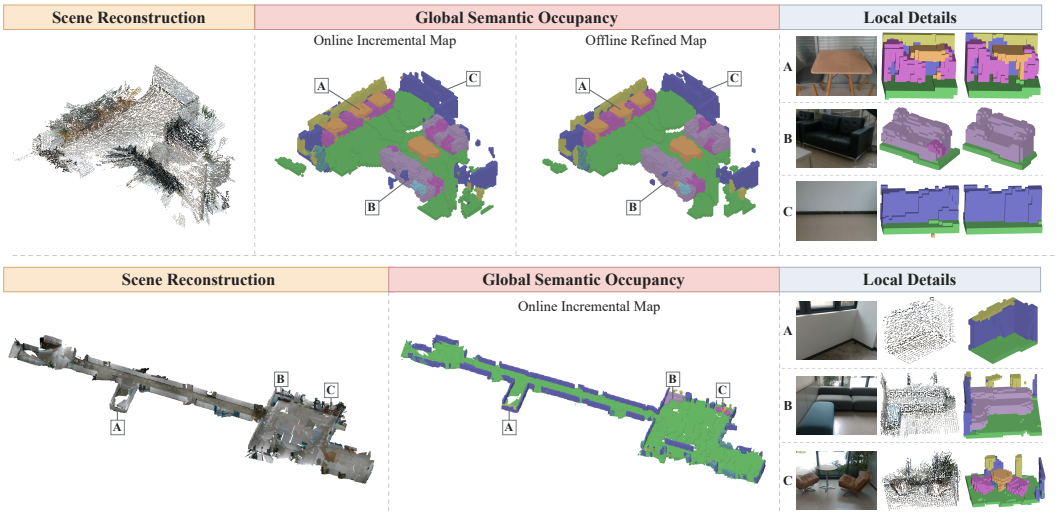

Figure 5: **Zero-shot real-world mapping results.** The upper and lower rows show the Reading Room and Laboratory Floor, respectively. Scene reconstructions are provided only for spatial reference and are not used by the occupancy predictor. The global maps demonstrate online incremental mapping at different scene scales, while offline refinement further improves the Reading Room result. A–C highlight representative local regions.

Occupancy prediction uses monocular RGB images, while camera poses are provided by the LiDAR–inertial localization system. The benchmark-trained model is deployed without scene-specific fine-tuning or adaptation. All real-world experiments use mixed FP16 together with TensorRT-accelerated monocular depth estimation.



Figure 6: Real-world robotic platform.

Real-World Mapping Scenarios. We consider two unseen indoor environments to examine deployment across different scene scales: a furnished Reading Room and a complete Laboratory Floor. As summarized in Table 4, their trajectories span 31.31 m and 235.15 m, respectively, covering compact room-scale exploration and extended floor-scale mapping. Motion-gated keyframe selection with a minimum interval of one second yields 91 and 351 processed frames, respectively.

Qualitative Results. Fig. 5 shows the maps produced without scene-specific adaptation. In the Reading Room, online incremental mapping captures the room layout and recognizable furniture structures. Offline global map refinement further reduces surface fragmentation in the highlighted object and wall regions while retaining the overall layout. On the Laboratory Floor, the online map represents connected corridor and room regions over the longer trajectory, with identifiable structural and object-level semantics in the enlarged views. Additional real-world comparisons with VEOcc across multiple mapping sequences are provided in Appendix C.3.

Deployment Efficiency. We measure runtime and memory during sequential replay of the recorded sequences on the onboard device. Table 4 evaluates the complete pipeline in both room-scale and floor-scale environments. In both settings, online mapping takes less than one second per keyframe on average, including local-feature cache writes, with average latencies of 0.630 and 0.764 s/frame for the Reading Room and Laboratory Floor, respectively. The corresponding persistent maps occupy 35.07 and 215.69 MiB, while complete offline refinement takes 4.29 and 25.07 s. In both environments, the offline refinement cost remains below 10% of the cumulative online processing time. Even for the full-floor sequence, peak process RSS remains 5.136 GiB during online mapping and 7.806 GiB during refinement. These results show that VEOcc++ supports both compact room-scale and long-horizon floor-scale mapping on the same embedded platform, demonstrating its practical efficiency and scalability for onboard embodied scene understanding. A finer-grained analysis of latency, process memory, and persistent-map growth throughout the Laboratory Floor sequence is provided in Appendix C.2.

Table 4: **Real-world deployment results** in two indoor environments. Map extent denotes the horizontal bounding box of the final stored voxel map. RSS reports the peak process memory usage in each stage, while Local Cache denotes the cached local features stored for offline refinement.

Scene	Travel Dist. (m)	Map Extent (m)	Key Frames	Online Avg. (s/frame)	Online Time (s/scene)	Offline Refine (s)	Online RSS (GiB)	Offline RSS (GiB)	Feature Map (MiB)	Local Cache (MiB)
Reading Room	31.31	10.0×9.0	91	0.630	57.32	4.29	4.172	4.340	35.07	151.55
Laboratory Floor	235.15	57.3×15.6	351	0.764	268.08	25.07	5.136	7.806	215.69	576.50

Table 5: **Online mapping ablation.** PFM: persistent feature map; CA: candidate adaptation; RG: reliability-guided gating; CAD: context-aware decoding.

Variant	PFM	CA	RG	CAD	IoU	mIoU
VEOcc fusion	Analytic fusion			✓	63.42	54.66
No memory	–	–	–	✓	61.33	52.03
Mean update	✓	–	–	✓	66.25	57.49
Gate only	✓	–	✓	✓	66.86	58.16
Candidate only	✓	✓	–	✓	66.77	57.85
Pointwise decode	✓	✓	✓	–	66.51	57.81
Full	✓	✓	✓	✓	**67.12**	**58.40**

Table 6: **Offline refinement ablation.** PF: persistent feature; LF: local features; LA: learned aggregation; GSR: global spatial refinement.

Variant	PF	LF	LA	GSR	IoU	mIoU
Online map	–	–	–	–	67.12	58.40
Probability only	–	–	–	✓	67.36	58.62
Prob. + feature	✓	–	–	✓	67.54	59.03
Prob. + local	–	✓	✓	✓	67.56	58.94
Mean fusion	✓	✓	–	✓	67.59	59.05
Pointwise refine	✓	✓	✓	–	67.21	58.68
Full	✓	✓	✓	✓	**67.79**	**59.37**

4.4 Ablation Studies

Online Incremental Mapping. Table 5 analyzes the main components of online incremental mapping under the same local predictor. Removing persistent memory causes a substantial drop to 61.33/52.03 IoU/mIoU, while replacing analytic prediction-level fusion with a Persistent Voxel Feature Map and cumulative mean update improves performance from 63.42/54.66 to 66.25/57.49. Building on this representation, reliability-guided gating and candidate adaptation provide complementary improvements when combined in the full Reliability-Guided Incremental Update, reaching 67.12/58.40. Replacing Context-Aware Decoding with pointwise prediction reduces performance to 66.51/57.81, further showing the benefit of spatial context when decoding the feature map.

Offline Global Map Refinement. Table 6 examines the contributions of the three input sources and spatial refinement. Applying Global Spatial Refinement to online semantic probabilities alone improves the completed map from 67.12/58.40 to 67.36/58.62 IoU/mIoU. Adding either persistent map features or learned local evidence further raises performance to 67.54/59.03 and 67.56/58.94, respectively, showing that both provide useful information beyond the online semantic prediction. Combining persistent features with mean-aggregated local evidence reaches 67.59/59.05, while learned Local Evidence Aggregation yields the best result of 67.79/59.37. Replacing Global Spatial Refinement with pointwise prediction reduces performance to 67.21/58.68. Overall, persistent map features and local evidence provide complementary cues, and spatial refinement is important for effectively exploiting them over the completed map.

5 Conclusion

We presented VEOcc++, a unified voxel-centric framework that connects online incremental mapping and offline global map refinement through a Persistent Voxel Feature Map. During exploration, reliability-guided feature updates and context-aware decoding maintain an incrementally expanding global map; after exploration, cached local evidence and spatial refinement further improve the completed representation. VEOcc++ establishes state-of-the-art online mapping performance on EmbodiedOcc-ScanNet, while offline refinement further improves the final map using the observations already acquired during exploration. Zero-shot robotic experiments additionally demonstrate transferability to unseen environments, onboard feasibility, and floor-scale indoor mapping. VEOcc++ currently relies on externally provided camera poses, and its persistent map and cached local evidence grow with the explored scene and sequence length. Future work will investigate robustness to pose uncertainty and more compact long-term scene representations.

AI USE STATEMENT

Generative AI tools were used for language editing and document organization. The authors verified the technical details, citations, experimental results, and claims, and take responsibility for the final content.

REPRODUCIBILITY STATEMENT

The paper specifies the task definition, state schema, evaluation datasets, metrics, training protocol, and baseline configuration to support reproduction of the reported results.

REFERENCES

Anh-Quan Cao and Raoul de Charette. MonoScene: Monocular 3D semantic scene completion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3991–4001, 2022.

Qiao Gu, Ali Kuwajerwala, Sacha Morin, Krishna Murthy Jatavallabhula, Bipasha Sen, Aditya Agarwal, Corban Rivera, William Paul, Kirsty Ellis, Rama Chellappa, Chuang Gan, Celso Miguel de Melo, Joshua B. Tenenbaum, Antonio Torralba, Florian Shkurti, and Liam Paull. ConceptGraphs: Open-vocabulary 3D scene graphs for perception and planning. In *2024 IEEE International Conference on Robotics and Automation*, pp. 5021–5028, 2024.

Wenlong Huang, Chen Wang, Ruohan Zhang, Yunzhu Li, Jiajun Wu, and Li Fei-Fei. VoxPoser: Composable 3D value maps for robotic manipulation with language models. In *Proceedings of the 7th Conference on Robot Learning*, volume 229 of *Proceedings of Machine Learning Research*, pp. 540–562. PMLR, 2023.

Zeyu Jiang, Changqing Zhou, Xingxing Zuo, and Changhao Chen. FreeOcc: Training-free embodied open-vocabulary occupancy prediction. In *Proceedings of Robotics: Science and Systems*, Sydney, Australia, July 2026. doi: 10.15607/RSS.2026.XXII.166.

Ziyang Leng, Jiawei Yang, Wenlong Yi, and Bolei Zhou. Occupancy learning with spatiotemporal memory. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 26569–26578, 2025.

John McCormac, Ankur Handa, Andrew J. Davison, and Stefan Leutenegger. SemanticFusion: Dense 3D semantic mapping with convolutional neural networks. In *2017 IEEE International Conference on Robotics and Automation*, pp. 4628–4635, 2017. doi: 10.1109/ICRA.2017.7989538.

Seokha Moon, Janghyun Baek, Yujin Jeong, Daewon Chae, Giseop Kim, Jungbeom Lee, Jinkyu Kim, and Sunwook Choi. Streaming dense voxel representations for 3D occupancy prediction. In *European Conference on Computer Vision*, pp. 659–676, 2026. doi: 10.1007/978-3-032-37356-4_36.

Rui Qian, Haozhi Cao, Tianchen Deng, Tianxin Hu, Weixiang Guo, Shenghai Yuan, and Lihua Xie. TGSFormer: Scalable temporal Gaussian splatting for embodied semantic scene completion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11881–11890, 2026.

Krishan Rana, Jesse Haviland, Sourav Garg, Jad Abou-Chakra, Ian Reid, and Niko Suenderhauf. SayPlan: Grounding large language models using 3D scene graphs for scalable robot task planning. In *Proceedings of the 7th Conference on Robot Learning*, volume 229 of *Proceedings of Machine Learning Research*, pp. 23–72. PMLR, 2023.

Manolis Savva, Abhishek Kadian, Oleksandr Maksymets, Yili Zhao, Erik Wijmans, Bhavana Jain, Julian Straub, Jia Liu, Vladlen Koltun, Jitendra Malik, Devi Parikh, and Dhruv Batra. Habitat: A platform for embodied AI research. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9339–9347, 2019.

Hao Shi, Song Wang, Jiaming Zhang, Xiaoting Yin, Guangming Wang, Jianke Zhu, Kailun Yang, and Kaiwei Wang. Offboard occupancy refinement with hybrid propagation for autonomous driving. *IEEE Transactions on Intelligent Transportation Systems*, 26(10):16001–16016, 2025. doi: 10.1109/TITS.2025.3572200.

Shuran Song, Fisher Yu, Andy Zeng, Angel X. Chang, Manolis Savva, and Thomas Funkhouser. Semantic scene completion from a single depth image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 190–198, 2017. doi: 10.1109/CVPR.2017.28.

Jiaming Sun, Yiming Xie, Linghao Chen, Xiaowei Zhou, and Hujun Bao. NeuralRecon: Real-time coherent 3D reconstruction from monocular video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15598–15607, 2021.

Hao Wang, Xiaobao Wei, Xiaoan Zhang, Jianing Li, Chengyu Bai, Ying Li, Ming Lu, Wenzhao Zheng, and Shanghang Zhang. EmbodiedOcc++: Boosting embodied 3D occupancy prediction with plane regularization and uncertainty sampler. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pp. 925–934, 2025.

Ruoyu Wang, Yong Liu, Jiahan Li, Sheng Tao, Yuhang Lin, and Yukai Ma. VEOcc: Voxel-centric online semantic occupancy prediction for embodied scene understanding. *arXiv preprint arXiv:2605.25059*, 2026.

Silvan Weder, Johannes L. Schonberger, Marc Pollefeys, and Martin R. Oswald. NeuralFusion: Online depth fusion in latent space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3162–3172, 2021.

Yuqi Wu, Wenzhao Zheng, Sicheng Zuo, Yuanhui Huang, Jie Zhou, and Jiwen Lu. EmbodiedOcc: Embodied 3D occupancy prediction for vision-based online scene understanding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 26360–26370, 2025.

Lihe Yang, Bingyi Kang, Zilong Huang, Zhen Zhao, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth Anything V2. In *Advances in Neural Information Processing Systems*, volume 37, pp. 21875–21911, 2024.

Shichao Yang, Yulan Huang, and Sebastian Scherer. Semantic 3D occupancy mapping through efficient high order CRFs. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 590–597, 2017. doi: 10.1109/IROS.2017.8202212.

Hongxiao Yu, Yuqi Wang, Yuntao Chen, and Zhaoxiang Zhang. Monocular occupancy prediction for scalable indoor scenes. In *Computer Vision – ECCV 2024*, volume 15088 of *Lecture Notes in Computer Science*, pp. 38–54. Springer, 2025. doi: 10.1007/978-3-031-73404-5_3.

Shanshuai Yuan, Julong Wei, Muer Tie, Xiangyun Ren, Zhongxue Gan, and Wenchao Ding. Collaborative learning of local 3D occupancy prediction and versatile global occupancy mapping. In *IEEE International Conference on Robotics and Automation*, 2026.

Zhang Zhang, Qiang Zhang, Wei Cui, Shuai Shi, Yijie Guo, Gang Han, Wen Zhao, Hengle Ren, Renjing Xu, and Jian Tang. RoboOcc: Enhancing the geometric and semantic scene understanding for robots. *arXiv preprint arXiv:2504.14604*, 2025.

Changqing Zhou, Yueru Luo, and Changhao Chen. Generalizing visual geometry priors to sparse Gaussian occupancy prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 28578–28587, 2026a.

Changqing Zhou, Yueru Luo, Yulan Guo, Bing Wang, Jie Qin, and Changhao Chen. GPOcc++: Unified sparse Gaussian occupancy prediction with visual geometry priors. *arXiv preprint arXiv:2607.13481*, 2026b.

Hu Zhu, Bohan Li, Xianda Guo, Hongsi Liu, Baorui Peng, Mingqi Yuan, Xin Jin, Wenjun Zeng, and Chang Wen Chen. GEM-Occ: From visual geometry evidence to embodied semantic occupancy memory. *arXiv preprint arXiv:2607.05543*, 2026.

Appendix

Appendix Contents

A Additional Method Details **12**

A.1 Local Predictor and Voxel Feature Extraction 12

A.2 Observation Alignment and Reliability Estimation 12

A.3 Persistent Map Update and Online Decoding 13

B Training Details **13**

B.1 Local Predictor Training 13

B.2 Online Incremental Mapping 14

B.3 Offline Global Map Refinement 14

C Additional Experimental Analysis **15**

C.1 Learning-Rate Analysis for Local Prediction 15

C.2 Online Scaling in Real-World Deployment 15

C.3 Real-World Online Mapping Comparison 16

A Additional Method Details

A.1 Local Predictor and Voxel Feature Extraction

The local occupancy predictor follows VEOcc (Wang et al., 2026) without architectural modification. Image features are extracted by a 2D backbone, while Depth Anything V2 (Yang et al., 2024) provides a monocular depth prior from the same RGB input. The depth prior conditions the LSS-based lifting process, projecting image features into the local voxel space without requiring an external depth sensor. A 3D encoder then produces the voxel representation $\mathbf{F}_t$, from which a lightweight occupancy head predicts frame-wise semantic occupancy. Importantly, the subsequent mapping stages operate on $\mathbf{F}_t$ before the final occupancy head, thereby preserving richer feature information for cross-frame accumulation.

A.2 Observation Alignment and Reliability Estimation

The pose-conditioned adaptation in Sec. 3.4 is implemented as a residual transformation conditioned on the normalized local voxel coordinate and a 6D camera rotation encoding:

$$\widehat{\mathbf{f}}_{t,i} = \mathbf{f}_{t,i} + \phi\big([\mathbf{f}_{t,i}, \mathbf{a}_{t,i}, \mathbf{e}_t^{\mathrm{rot}}]\big)\,, \tag{8}$$

where $\phi(\cdot)$ is a lightweight MLP. Its output layer is zero-initialized, such that the module starts from an identity transformation.

For reliability estimation, we follow the Reliability-Aware Confidence Modulation (RCM) of VEOcc (Wang et al., 2026). For each local voxel observation i, the confidence is computed from its camera-space depth and image-boundary proximity:

$$c_{t,i} = \mathrm{clip}\left(\exp(-\alpha z_{t,i})\exp(-\beta b_{t,i}), c_{\min}, 1\right), \tag{9}$$

where $z_{t,i}$ denotes camera-space depth and $b_{t,i}$ measures proximity to the image boundary. For the observations $\mathcal{G}_t(k)$ associated with global voxel k, the voxel-wise reliability is computed as

$$r_t(k) = \frac{\sum_{i\in\mathcal{G}_t(k)} c_{t,i}^2}{\sum_{i\in\mathcal{G}_t(k)} c_{t,i}}. \tag{10}$$

Table 7: **Training configuration and time of the three stages.**

Stage	Per-GPU Batch	Total Batch	Epochs	Training Time (h)
Local Predictor	4 images	16 images	10	24.8
Online Incremental Mapping	1 trajectory (30 frames)	4 trajectories	3	1.3
Offline Global Map Refinement	1 scene map	4 scene maps	30	1.9

Table 8: **Local learning-rate study on Occ-ScanNet.** All results are measured at epoch 10 using the same architecture and remaining training configuration. The best results are marked in **bold**.

Learning rate	Multiplier	IoU	mIoU
4×10^{-4} (VEOcc; Wang et al., 2026)	$1\times$	64.55	55.49
8×10^{-4}	$2\times$	66.11	57.35
1.6×10^{-3}	$4\times$	66.90	58.08
2×10^{-3} (VEOcc++, Ours)	$5\times$	**66.98**	**58.17**

This reliability is computed independently from the aligned voxel feature $\mathbf{x}_t(k)$ and is used to condition the subsequent feature update.

A.3 Persistent Map Update and Online Decoding

For each mapped voxel, we additionally maintain the historical reliability $\bar{r}_t(k)$ and observation count $n_t(k)$. For a repeated observation, they are updated as

$$\bar{r}_t(k) = \frac{n_{t-1}(k)\bar{r}_{t-1}(k) + r_t(k)}{n_{t-1}(k) + 1}, \qquad n_t(k) = n_{t-1}(k) + 1. \tag{11}$$

The observation count is used only to maintain the running reliability and is not provided to the learned updater.

For a re-observed voxel, we use the shorthand $\mathbf{h} = \mathbf{h}_{t-1}(k)$ and $\mathbf{x} = \mathbf{x}_t(k)$. The standardized feature distance is

$$D(\mathbf{h}, \mathbf{x}) = \sqrt{\text{mean}\left[(\mathbf{h}^n - \mathbf{x}^n)^2\right] + \epsilon}, \tag{12}$$

where $\mathbf{h}^n$ and $\mathbf{x}^n$ denote the standardized historical and current features.

We characterize the relation between the historical and current features as

$$\mathbf{f}_t^{\text{rel}}(k) = \left[\mathbf{h}, \mathbf{x}, \mathbf{h} - \mathbf{x}, \mathbf{h} \odot \mathbf{x}, \cos(\mathbf{h}, \mathbf{x}), D(\mathbf{h}, \mathbf{x})\right]. \tag{13}$$

Together with semantic uncertainty $E_t(k)$ derived from the local prediction, the current reliability $r_t(k)$, and the historical average reliability $\bar{r}_{t-1}(k)$, the complete update descriptor is

$$\boldsymbol{\xi}_t(k) = \left[\mathbf{f}_t^{\text{rel}}(k), E_t(k), r_t(k), \bar{r}_{t-1}(k)\right]. \tag{14}$$

The descriptor jointly conditions the candidate adaptation and reliability-guided gating branches. The two branches share a lightweight feature encoder, and their output layers are zero-initialized such that the initial update reduces to reliability-ratio interpolation.

The online decoder is implemented by a submanifold sparse convolution followed by a pointwise sparse classifier.

B Training Details

VEOcc++ is trained in three sequential stages, with modules optimized in earlier stages frozen during subsequent training. All stages are trained on four NVIDIA RTX 3090 GPUs. Table 7 summarizes the batch configuration, training schedule, and training time for each stage.

B.1 Local Predictor Training

We train the local predictor for 10 epochs using AdamW with a base learning rate of 2×10^{-3} and a 0.1 learning-rate multiplier for the image backbone. Following VEOcc (Wang et al., 2026), the

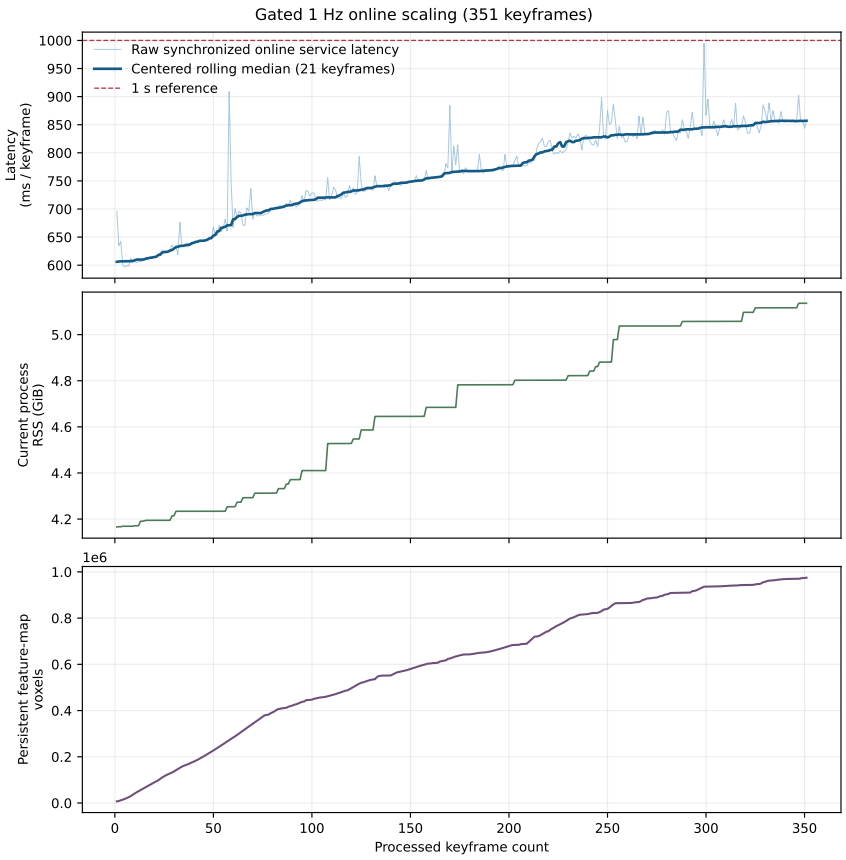

Figure 7: **Online deployment scaling on the Laboratory Floor sequence.** We track runtime and memory as the global map grows over 351 motion-gated keyframes. The panels show per-keyframe online service latency, current process RSS, and Persistent Voxel Feature Map size, respectively. The centered rolling median over 21 keyframes highlights the latency trend, while the dashed line indicates the 1 s reference.

local training objective is

$$\mathcal{L}_{\text{local}} = 100\mathcal{L}_{\text{focal}} + \mathcal{L}_{\text{Lovasz}} + \mathcal{L}_{\text{sem}} + \mathcal{L}_{\text{geo}}. \tag{15}$$

The monocular depth prior is not additionally supervised during this stage.

B.2 Online Incremental Mapping

We freeze the local predictor and optimize the pose-conditioned feature adaptation, Reliability-Guided Incremental Update, and Context-Aware Decoding modules for 3 epochs. AdamW is used with a learning rate of 2×10^{-4} and a linear warm-up over 603 iterations. The online training objective is

$$\mathcal{L}_{\text{online}} = \mathcal{L}_{\text{CE}} + \mathcal{L}_{\text{Lovasz}} + \mathcal{L}_{\text{sem}} + \mathcal{L}_{\text{geo}}, \tag{16}$$

where the cross-entropy term uses inverse-log class-frequency weighting.

Training follows the causal mapping process. The persistent map state is propagated across frames within each trajectory, while gradients through the historical state are detached after each optimization step. Therefore, the model is optimized with single-step truncated temporal back-propagation rather than back-propagating through the complete sequence.

B.3 Offline Global Map Refinement

Finally, we freeze both the local predictor and online incremental mapping modules and jointly optimize Local Evidence Aggregation and Global Spatial Refinement for 30 epochs. AdamW is

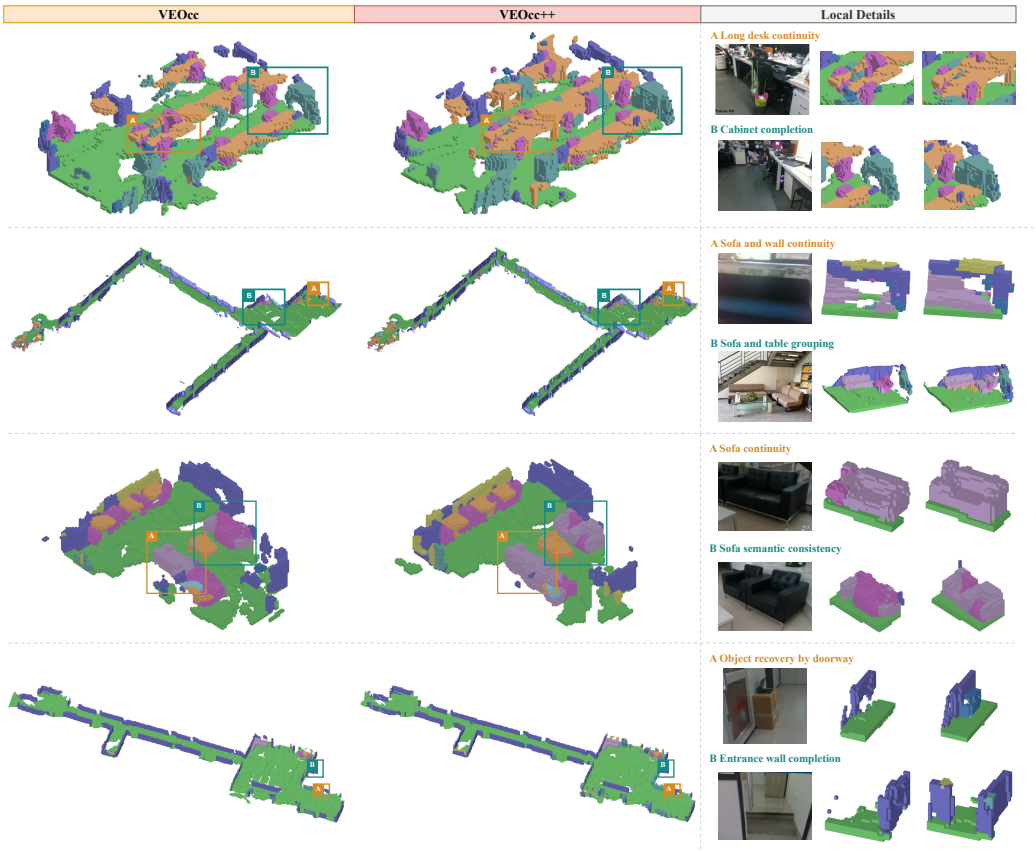

Figure 8: **Qualitative comparison of online incremental mapping on real-world robotic sequences.** The first two rows show self-collected sequences previously used in VEOcc, while the last two rows correspond to the two zero-shot deployment sequences reported in the main paper of VEOcc++. For each sequence, we compare the online maps produced by VEOcc and VEOcc++, and show representative local regions in the right column. VEOcc++ yields more complete structures, better continuity of extended objects and surfaces, and improved semantic consistency in the highlighted regions.

used with a peak learning rate of 10^{-3}, one epoch of warm-up, and cosine decay to 10^{-6}. The offline refinement stage uses the same combination of cross-entropy, Lovasz, semantic-scaling, and geometric-scaling losses as the online stage.

C Additional Experimental Analysis

C.1 Learning-Rate Analysis for Local Prediction

We further analyze the effect of the learning rate on the unchanged VEOcc local predictor. As shown in Table 8, increasing the learning rate consistently improves local occupancy prediction, indicating that the original architecture retains additional optimization potential. Based on this analysis, we adopt the selected learning-rate setting for all subsequent experiments while keeping the local prediction architecture unchanged.

C.2 Online Scaling in Real-World Deployment

Fig. 7 provides a finer-grained view of the online deployment behavior on the Laboratory Floor sequence. As exploration proceeds, the Persistent Voxel Feature Map grows to nearly one million voxels, while process memory increases gradually with the accumulated scene state. Online latency also rises with map growth but remains below the 1 s reference for all processed keyframes in this

sequence. These results complement the aggregate statistics in Table 4 by showing how runtime and memory evolve throughout long-horizon floor-scale mapping.

C.3 Real-World Online Mapping Comparison

Fig. 8 presents additional qualitative comparison of online incremental mapping in real-world robotic scenes. The first two rows show self-collected sequences previously used in VEOcc, and the last two rows show the two zero-shot deployment sequences reported in the main paper of VEOcc++. In all four cases, VEOcc++ produces cleaner and more spatially consistent occupancy maps. The highlighted local regions further show improved continuity of elongated structures, more complete recovery of cabinets and walls, and stronger semantic consistency for furniture and small objects. These results provide additional evidence that the proposed feature-level mapping pipeline improves real-world online occupancy mapping beyond analytic prediction-level fusion.