FloVerse: Floor Plan-Guided Multi-Modal Navigation

Supplementary Material

A. Floor Plan Construction

Algorithm 1: Floor Plan Extraction

Input: 3D scene in OBJ format
Output: floor plan $F \in \mathbb{R}^{H_f \times W_f}$

// Floor Height Estimation
$M_{\text{v}} \leftarrow \{ f \in M \mid \mathbf{n}(f) \text{ is nearly vertical} \}$
$H \leftarrow \{ z(f) \mid f \in M_{\text{v}} \}$
$C \leftarrow \textsc{Cluster}(H)$
$h_{\text{floor}} \leftarrow \textsc{Centroid}(\arg\max_{c \in C} |c|)$
// Horizontal Slicing for Floor Plan Extraction
$h\_cut \leftarrow \{ h_{\text{floor}} + 1.25 \}$
$s\_cut \leftarrow \textsc{Intersect}(M, h\_cut)$
$cts \leftarrow \textsc{TraceContours}(s\_cut)$
$\tilde{F}_0 \leftarrow \textsc{AssembleFloorPlan}(cts)$
// Post-Processing
$\tilde{F}_1 \leftarrow \textsc{Denoising}(\tilde{F}_0)$ // remove small artifacts
$\tilde{F}_2 \leftarrow \textsc{Morphology}(\tilde{F}_1;\ \text{dilate} \rightarrow \text{erode})$
$F \leftarrow \textsc{Denoising}(\tilde{F}_2)$ // smooth final boundaries
return F

Given the lack of floor plan annotations in many indoor datasets, we reconstruct 2D floor plans from 3D scene meshes using the pipeline described in Algorithm 1. We identify near-vertical faces by thresholding the angular deviation between face normals $\mathbf{n}(f)$ and the gravity direction, and cluster their centroid heights $z(f)$ to estimate the floor level. We then extract wall contours by intersecting the mesh with a horizontal plane at a fixed offset above the estimated floor.

The slicing offset is a critical design choice. Based on empirical analysis, we set the offset to $1.25\,\text{m}$. Smaller offsets (0.25–$0.5\,\text{m}$) intersect furniture and introduce significant high-frequency noise, while larger offsets (e.g., $2.0\,\text{m}$) fail to capture continuous wall structures, leading to fragmented contours. The $1.25\,\text{m}$ offset balances noise suppression and wall completeness, resulting in stable, coherent intersections.

Finally, we apply a morphological closing to consolidate fragmented wall segments, using 15 dilation and 5 erosion iterations with 13×1 and 1×13 kernels. We further suppress noise by removing small black connected components (< 100 pixels) and holes enclosed by components smaller than 400 pixels, yielding clean and consistent floor plan boundaries. The resulting floor plans are exported as PNG images, where wall structures are encoded in black. Moreover, **FloVerse-**1.6**K** provides precise correspondence between floor plan pixel coordinates and the associated real-world coordinates for each scene.

B. Navigable Map

To enable robust imitation learning, we require collision-free trajectories grounded in accurate and consistent navigable maps. Although HM3D and Gibson provide navigability annotations, their quality and formats are inconsistent. We therefore reconstruct a unified navigable map for each scene and perform careful manual refinement to ensure dataset-wide reliability and consistency.

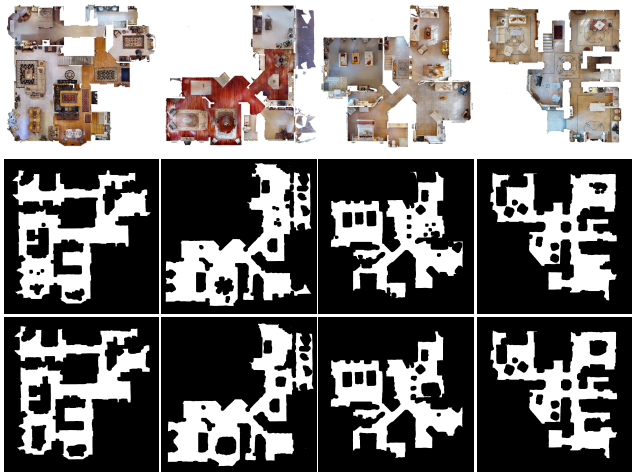

Figure A1. **High-Fidelity Navigable Maps in FloVerse-**1.6**K.** Top row: Top-down views for the scene. Second row: navigable maps before manual annotation. Bottom row: navigable maps after manual annotation.

We construct the navigable map by projecting 3D mesh faces within a selected height range onto the ground plane and marking occupied regions as non-navigable. Heights that are too low capture floor reconstruction noise, while excessive heights (e.g., door frames and overhanging structures) are mistakenly treated as obstacles, fragmenting free space. Based on empirical validation, we set the height range to 0.3–$2.0\,\text{m}$.

Finally, we manually verify and correct all generated maps to handle missing or spurious mesh artifacts arising from reconstruction errors. The resulting navigable maps are produced at a resolution of $1\,\text{cm}$/pixel, as shown in Fig. A1.

C. Object Goal Annotation

To enrich both the number of scenes with object-goal annotations and the diversity of object categories, we employ SpatialLM [20], a 3D large language model designed to process 3D point cloud data and generate structured 3D scene understanding outputs. As SpatialLM is designed to operate on single-layer point clouds, we first decompose each scene into hierarchical layers following the procedure in Sec. A. We then input each layer's point cloud into SpatialLM to obtain layer-wise object annotations. To prevent the outputs from containing incorrect detections, as shown in Fig. A3, we manually verify and correct the predictions to ensure that all object annotations are accurate.

To ensure the geometric validity of object placements, we perform an additional verification on all objects. Specifically, we re-

Table A1. Object categories in **FloVerse-**1.6**K**.

sink, blanket, bathroom cabinet, cabinet, chair, table, rack, tv, bathroom shelf, window, couch, stove, bed, toilet, support beam, shelving, wardrobe, sideboard, easy chair, kitchen lower cabinet, desk, bench, sofa set, stool, tray, board game, storage, screen, radiator, washing machine, ladder, dining_chair, sofa, washing_machine, dining_table, shower, bathroom_cabinet, nightstand, cupboard, bookcase, case, sign, box, sheet, treadmill, bag, rug, washer-dryer, dishwasher, pillow, blinds, mirror, bathtub, cooker, bathtub platform, refrigerator, plant, sink cabinet, mantle, dressing_table, towel, shower cabin, pot, kitchen cabinet lower, crate, pillar, hat, clothes, oven, shower bar, board, heater, antique clock, handbag, desk lamp, radio, curtain valence, kitchen appliance, counter, display cabinet, figure, flag, book rack, hunting trophy, cloth, wine_cabinet, plants, carpet, oven and stove, storage cabinet, vacuum cleaner, bath towel, cabinet clutter, record player, air conditioner, stovetop, paper, coffee_table, side_table, cardboard box, clutter, pouffe, laundry, wood, basket, brochure, note, office chair, canvas, banner, drawer sink table, duct, drawer, sofa seat, closet shelving, scarf, kitchen top, shower rail, bathrobe, bathroom towel, tv_cabinet, book, picture, footstool, monitor, window shade, countertop, bottle, laundry basket, shelf cubby, vase, microwave, flower vase, shower glass, clothes dryer, whiteboard, washbasin counter, kitchen table, bulletin board, island, stair, bedframe, dresser, plush toy, window shutter, sofa chair, cardboard, stack of papers, jacket, toy, guitar, flowerpot, closet, piano, bar_chair, worktop, shower tap, chimney, washbasin, sink table, candle, lounge chair, storage shelving, plate, freezer, boiler, folding table, trampoline, container, water tank, gas furnace, sleeping bag, electrical controller, patio chair, grill, rocking chair, stereo set, closet shelf, seat, lamp stand, gym equipment, dish rack, crib, cradle, bathroom accessory, tool, bucket, shower soap shelf, paper towel, spice rack, antique telephone, tub, light switch, jewelry box, potted_bonsai, coffee machine, bowl of fruit, glass, clock, dinner table, bar, hose, ottoman, display table, grass, shower-bath cabinet, decorative quilt, trashcan, magazine, stand, binder, amplifier, hutch, dressing table, kitchen extractor, entrance_cabinet, balustrade, telescope, sunbed, clothing stand, paper storage, dish cabinet, candle holder, kitchen shelf, elevator, office table, window glass, iron board, handrail, printer, laptop, tent, ornament, handle, soft chair, table stand, globe, decorative_cabinet, shoe_cabinet, aquarium, kitchen sink, backrest, stage, massage bed, exercise ladder, bed curtain, locker, pool, bed comforter, headboard, coat, cabinet table, parapet, exercise bike, photo, stack of stuff, desk cabinet, book cabinet, photo stand, shower stall, shade, water dispenser, archway, storage space, fire extinguisher, exhibition panel, solarium, pile of magazines, cosmetics, kitchen countertop items, statue, calendar, lampshade, hook, painting frame, baby changing station, purse, tile, artwork, arcade game, backsplash, fish tank, pantry, newspaper, cutting board, speaker, bunk bed, jar, decorative plant, floor-standing_lamp, telephone, dishrag, washing_cabinet, pool table, hanging clothes, umbrella, cart, barbecue, hanger, foosball game table, high shelf, range hood, robe, hammock, bar cabinet, clothes bag, violin case, round chair, folding stand, wreath, spa bench, tank

(a) Object Categories in the Training Set

lounge chair, bench, bed, aquarium, double armchair, cabinet, medical lamp, washbasin, chair, sauna oven, brochure, foot spa, pot, box, towel, relief, window, massage bed, seat, backrest, table, laptop, sign, basket, crate, desk, toilet, nightstand, bathroom_cabinet, picture, refrigerator, stair, dishwasher, rug, sink, couch, tv, wardrobe, display cabinet, bar, heater, oven and stove, blanket, kitchen top, pillar, dining_table, dining_chair, sideboard, cooker, sink cabinet, side_table, sofa, ladder, shower-bath cabinet, cardboard box, vacuum cleaner, board, statue, toy, carpet, stool, dressing_table, shower, stove, worktop, office chair, washing machine, clothes dryer, painting frame, handle, plants, floor-standing_lamp, oven, jacket, clothes, decorative quilt, countertop, bathtub, pouffe, decorative plate, closet, rocking chair, rack, shower tap, bathroom cabinet, stand, bathroom shelf, bath towel, kitchen table, bathroom towel, container, clutter, stovetop, case, hat, ornament, pillow, staircase handrail, parapet, mirror, window glass, kitchen lower cabinet, glass, kitchen appliance, tv_cabinet, coffee_table, elevator, tub, high shelf, stone support structure, electrical controller, vase, kitchen shelf, storage shelving, calendar, island, flower vase, flowerpot, bar_chair, gym equipment, balustrade, plush toy, decorative plant, radiator, shower cabin, display table, dinner table, telephone, fuse box, book, plant, window shade, microwave, drawer, photo mount, stack of papers, schedule, shade, shelving, cupboard, washing_machine, shoe_cabinet, mixer, wine cabinet, handbag

(b) Object Categories in the Validation Set

Figure A2. **Qualitative Results of ThreeDiff on FloVerse-**1.6**K.**

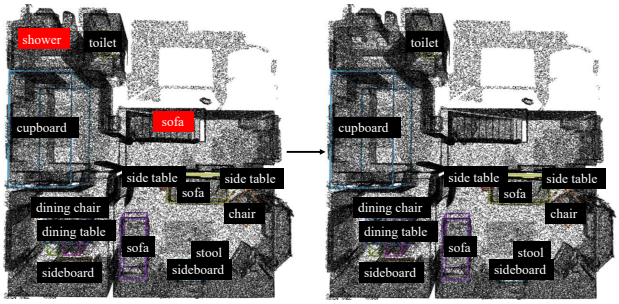

Figure A3. **The Manual Verification Process.** Left: detection results before manual verification, the red boxes indicate incorrect detections. Right: detection results after manual verification.

Table A2. **Additional Experiments.** Comparison between finetuned baselines and **ThreeDiff**.

Method	ImageNav		ObjectNav	
	SR	SPL	SR	SPL
ZSON (finetune) [19]	-	-	10.0	9.1
RL Monolithic (finetune) [28]	8.4	3.7	22.5	15.2
ThreeDiff	**28.9**	**22.4**	**28.6**	**22.3**

move two categories of invalid placements: (1) objects whose centers lie within navigable regions, since all objects should fall in non-traversable areas; (2) objects in overly confined or cluttered regions, where their placement prevents feasible observation viewpoints. The complete set of object categories included in our dataset is summarized in Tab. A1.

D. Training Data Construction

This section describes how we construct training samples for **ThreeDiff** from expert trajectory data. For each waypoint sequence, we sample a starting point every four waypoints, and each selected start point is then paired with the final waypoint of the trajectory, forming multiple start–end pairs. For each pair, we extract the past 8 waypoints as the historical context and the next 16 waypoints as the future supervision signal. Formally, each training sample consists of (S_h, P_{gt}, F, g), where the historical states $S_h = (s_{t-7}, s_{t-6}, ..., s_t,)$, and the ground-truth future positions $P_{gt} = (p_t, p_{t+1}, p_{t+2}, ..., p_{t+15})$. The goal specification g is defined as $g = (g_{point}, g_{image}, g_{object})$ for IO episodes, and $g = g_{point}$ otherwise.

E. Additional Experiments

We further finetune the ZSON [19] and RL Monolithic [15] models using the episodes in **FloVerse-**1.6**K**. We train ZSON and RL Monolithic for an additional 80M and 100M steps, respectively, and then evaluate them on our validation set. As shown in Tab. A2, ZSON exhibits a modest performance gain after finetuning, consistent with expectations, yet it remains notably inferior to **ThreeDiff**. In contrast, RL Monolithic experiences performance degradation, as illustrated in Tab. 6. This decline occurs because, unlike GoatBench, our evaluation protocol treats any episode with more than 15 collisions as a failure, thereby imposing a stricter success criterion.

F. More Qualitative Results

We provide additional navigation examples of **ThreeDiff**, as shown in the Fig. A2.

G. Inference speed.

On an NVIDIA 4090 GPU, **ThreeDiff** runs at 0.18s/step, with each stage taking about 0.09s.