Seeing Through Clutter
Structured 3D Scene Reconstruction via Iterative Object Removal

Supplementary Material

Supplementary Material for "Seeing Through Clutter: Structured 3D Scene Reconstruction via Iterative Object Removal"

6. VLM orchestrator

We provide the prompt we use to query the VLM (GPT-4o) and prompt to remove objects (Flux Kontext).

6.1. Amodal selection

Amodal Selection Prompt

Instructions
1. You will be given an image of a scene. First, describe what the scene is.
2. Examine the scene and identify the object that is closest to the camera. This object should be distinct and fully visible, meaning it is not occluded by any other objects (unless part of it is outside the camera's frame). If it is occluding any objects, note which objects it is occluding. If there are no objects in the scene, state that the scene is empty.
3. If the scene is not empty and you have identified the closest fully visible object, explain why it might be there in relation to the full scene. Explain if there are any objects adjacent to the closest fully visible object and explain how they are behind the closest fully visible object. Use your explanations to re-evaluate what the closest fully visible object is.
4. Once you have determined what the closest fully visible object is, check if there are any small, distinct objects placed on top of or inside the fully visible object. These are referred to as secondary objects and should not include integral parts of the visible object itself. List these objects if applicable.
5. Once you've identified the objects, compile your list in the format:
- {VISIBLE_OBJECT: [Object]}
- {SECONDARY_OBJECTS: [Object 1, Object 2, ..., Object N]}
Examples:
- For an empty scene: {VISIBLE_OBJECT: []}, {SECONDARY_OBJECTS: []}
- For a scene with a table as the closest fully visible object and a book on top: {VISIBLE_OBJECT: [Table]}, {SECONDARY_OBJECTS: [Book]}
Guidelines
1. If there are multiple objects that seem to be equally distant from the camera and fully visible, pick one of them to be selected for {VISIBLE_OBJECT: []}.
2. In your final list of objects, exclude objects that could be considered background elements, such as floors, walls, windows, doors, grass, rugs, or other structures. If the visible object is one of these background elements, pick the next closest fully visible object if there is one. If there isn't, state so.
3. When describing objects, try to keep the names of these objects between 1–3 words. Avoid unnecessary adjectives.
4. When compiling your lists, use the singular form of objects, even if there are multiple. Only include the name of the object in your compiled lists, NOT the explanations.
5. If there are no secondary objects on or inside the fully visible object, format the list as: {VISIBLE_OBJECT: [Object]}, {SECONDARY_OBJECTS: []}
6. If you are unsure if there are any objects, leave the lists empty: {VISIBLE_OBJECT: []}, {SECONDARY_OBJECTS: []}
Prompt
Using the provided instructions and image of a scene, identify the object closest to the camera that is fully visible and not occluded by any other objects. Then, determine if there are any secondary objects on or inside it. Compile your findings in the formats {VISIBLE_OBJECT: []} and {SECONDARY_OBJECTS: []}.

Amodal Selection Output

OUTPUT:
Scene description:
A cozy living room with brown leather seating, a central wooden coffee table on a patterned rug, a large TV, windows with outdoor greenery, and several plants and wall art.
Closest fully visible object:
The small leather ottoman in the foreground, centered near the bottom.
Secondary objects on or inside it:
None visible.
- {VISIBLE_OBJECT: [Ottoman]}
- {SECONDARY_OBJECTS: []}

6.2. Inpainting

Inpainting Prompt

Prompt: Remove the {OBJ_NAME} inside of the red highlight.

Algorithm 2: LEAST_SQUARES

Input: $X_B \in \mathbb{R}^{N\times 3}$: points from rotated render (space B)
$X_A \in \mathbb{R}^{N\times 3}$: points from depth (space A)
Output: T: Sim(3) transform $[sR \mid t]$ mapping $B \to A$

$\mu_B \leftarrow \frac{1}{N}\sum X_B, \quad \mu_A \leftarrow \frac{1}{N}\sum X_A;$
$\tilde{X}_B \leftarrow X_B - \mu_B, \quad \tilde{X}_A \leftarrow X_A - \mu_A;$
$\Sigma \leftarrow \frac{1}{N}\tilde{X}_A^\top \tilde{X}_B;$
$U, D, V^\top \leftarrow \mathrm{SVD}(\Sigma);$
$S_{fix} \leftarrow \mathrm{diag}(1, 1, \mathrm{sign}(\det(UV^\top)));$
$R \leftarrow US_{fix}V^\top;$
$s \leftarrow \frac{\mathrm{trace}(S_{fix}D)}{\mathrm{Var}(\tilde{X}_B)};$
$t \leftarrow \mu_A - sR\mu_B;$
Form $T \in \mathbb{R}^{4\times 4}$ with $T_{0:3,0:3} = sR$, $T_{0:3,3} = t$;
return T;

Output:

7. Object Fitting Algorithms

Algorithm 3: ICP

Input: P_B: source cloud from rotated render
P_A: target cloud from segmented depth
v: voxel size, r: NN radius, ρ: keep ratio, T_{max}: max iters
Output: T: Sim(3) transform $[sR \mid t]$ mapping $B \to A$

$P_B \leftarrow \mathrm{VoxelDownsample}(P_B, v);$
$P_A \leftarrow \mathrm{VoxelDownsample}(P_A, v);$
Initialize scale s, rotation $R = I$, translation t by centroid alignment;
for $iter \leftarrow 1$ **to** T_{max} **do**
  $P_B^{now} \leftarrow sRP_B + t;$
  $Corr \leftarrow \mathrm{NearestNeighbor}(P_B^{now}, P_A, r);$
  if $|Corr| < N_{min}$ **then**
    break
  Compute residuals $\{\|p_B^{now} - p_A\|\}_{(p_B,p_A)\in Corr}$;
  Keep top $K = \max(8, \lfloor \rho|Corr| \rfloor)$ pairs;
  $X \leftarrow$ source subset, $Y \leftarrow$ target subset;
  $(s, R, t) \leftarrow$ **SIM3_LEAST_SQUARES**(X, Y);
  if *change in RMS* $< \delta$ **then**
    break
Form T from s, R, t and return;

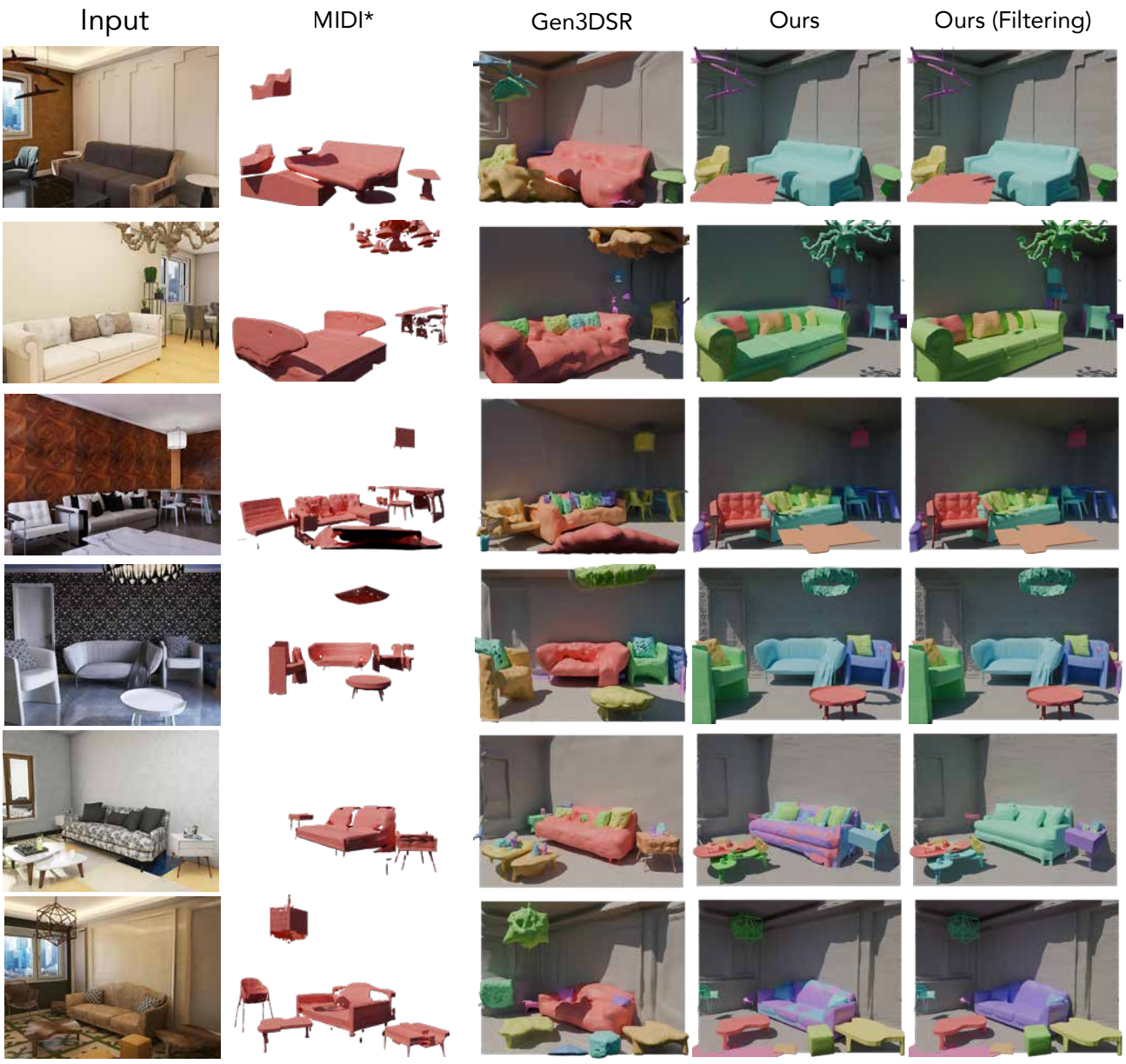

Figure 8. Qualitative comparison between MIDI*, Gen3DSR, ours, ours (w/ object filtering) on 3D front scenes.

Figure 9. A gallery of qualitative results on in-the-wild indoor and outdoor scenes.