We thank the reviewers for their insightful feedback. We address their concerns below.

R1.Q1: Supervision Level. Existing single view reconstruction methods we compare to use synthetic renderings of 3D meshes for training. Therefore, camera poses and depth maps are available for free in this experimental setup. In effect, we propose to make use of this additional information to break down 3D reconstruction into simpler tasks. Similarly, in a scenario where ground truth 3D models are used in association with real photographs, well-established computer vision methods can be used to allow for recovering ground truth camera pose and depth map, as was done in the pix3D paper. There is therefore not much benefit in *not* taking advantage of them.

R1.Q2: Novelty. We acknowledge that we do not introduce any entirely new computational block. The novelty lies in how we assemble the blocks in a principled manner to break down 3D reconstruction into simpler subtasks. The hybrid shape decoder is also novel and allows for coarse-to-fine reconstruction.

R1.Q3: Depth Ablation Study. All reviewers noted that an ablation study on the influence of appending depth predictions to 3D feature grids was missing. In Tab. 1, we therefore report reconstruction accuracy on the subset of cars in the setting where ground truth camera poses are known, for different depth predictors: Using ground truth depth maps (*GT*), inferring them (*INF*), and removing them from the pipeline (*NO*). Removing depth information significantly degrades accuracy and using our inferred depth maps delivers an accuracy approaching that using the ground truth ones.

Depth	CD($\downarrow$)	EMD($\downarrow$)
GT	3.80	2.14
INF	3.83	2.16
NO	3.97	2.20

Table 1: **Depth ablation**

R1.Q4: Embedding. Our intuition is that relying on a 1D flat vector embedding forces most 3D reconstruction networks to semantically encode localization information, and the reviewer has a point that we do not demonstrate this explicitly. This is not easy to do because using 1D global embeddings would require a totally different network architecture. We would welcome any suggestion on how to do this properly.

R1.Q5: Clarity. We will clarify to make the paper self-contained, and improve Fig. 3.

R2.Q1a: Benchmark metrics. In Tab. 2 of Mesh-RCNN, shapes are normalized to bounding boxes of side length 10, and sampled using 10k points. In contrast and as detailed in the supplementary, we normalize to unit sphere, sample 2048 points and scale the final CD-L2 by a factor of 10^3 (like DISN). This strongly affects CD-L2 since it is dependent on scale and sampling density. Using the Mesh-RCNN setting on our dataset, our method yields an average CD-L2 of 0.197 to be compared with 0.250 for Mesh-RCNN.

R2.Q1b: Occupancy. We only sample points on the object's surface, without any points being generated inside. Therefore, in our occupancy maps, $occ = 1$ *only* for voxels intersecting the surface.

R2.Q2a: Back-projection. This is a misunderstanding. The features at a 2D location are back-projected everywhere along the camera ray, without reference to depth. We will clarify.

R2.Q2b: Depth Ablation. See R1.Q3 and Tab. 1.

R2.Q2c: Per-Voxel Point Sampling. It is essential to sample multiple points per voxels, since otherwise output shapes would be voxelized at a relatively coarse resolution (28^3). The effect of this refinement using folded patches is qualitatively shown shown in Fig. 4b of the main paper. Quantitatively, on our whole testing set, sampling points at the center of occupied voxels instead of locally folding patches yields an increase of ~6.3 in EMD.

R2.Q3: Synthetic vs. Real Scenes We share the reviewer's concern and would add that it applies to most current single view reconstruction deep learning methods, many of which only work on clean synthetic images. This is why we tested ours on the pix3D dataset with real images that feature more complex shadows, textures, and exposures. We will focus on the other issues the reviewer mentions in future work.

R3.Q1: Supervision Level. See R1.Q1.

R3.Q2: Failure to Predict Occupancy. In case occupancy is wrongly predicted, there is indeed no way for the local patch folders to recover the correct shape. For this reason, a strong emphasis is put on occupancy during training: during the first epoch the network is supervised using $\mathcal{L}_{BCE}$ only. Then the total loss is $\mathcal{L} = 100 * \mathcal{L}_{BCE} + \mathcal{L}_{CD}$.

R3.Q3: Ensuring Local Patches Contiguity. Patches sometimes do not perfectly align at voxels' borders. We will fix this in future work using either a regularizer, or different architecture, or downstream deterministic computations.

R3.Q4: Ablation Study. We provide additional ablation studies to support design choices: see R1.Q3 and Tab. 1 for an ablation of depth information. In addition, hard clamping the 3D feature grids using depth maps incurs an increase of ~0.16 in CD and ~0.07 in EMD compared to simply appending them to 3D feature grids for 3D convolutions.

R3.Q5: Pix3D Benchmark. We only tested on the chair subset of pix3D because it is the benchmark for single view reconstruction proposed in the original pix3D paper. Our method generalizes to other classes as shown in Fig. 1. For tables, we get a CD-L1 of 7.7 compared to 7.5 on chairs.

R3.Q6: Implementation Details. We will clarify the following points: **a)** Depth prediction, pose estimation and feature extraction subnetworks each have their own set of parameters. **b)** The 40 final features are split into 8 and 32 and then sent to the *occ* and *fold* branches with the intent to encourage disentanglement between occupancy and local patch deformation. Early experiments showed better results over feeding the entire set of features. **c)** The occupancy threshold τ is manually tuned at 0.35.

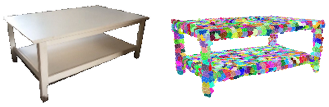

Figure 1: **Pix3D table: input image and 3D shape reconstructed by our method.**