We thank reviewers for their insightful comments. Please find below our answers to the questions.

R1: Describe PerspectiveNet in more clear steps. Describe \( g, K, \) non-holes. Thank you, we will add a clear overview of the algorithm as suggested and expand ln.86-94 with more concrete descriptions.

R1 & R2: Move point-tracer from supplementary. We agree and we will migrate the paragraph to the paper.

R1 & R3: BiGAN predictions noisy. Incorrectly trained? During preliminary experiments, we observed “red flags” related to GANs, suggesting autoencoders are more suitable: (1) Training a state-of-the-art MSGGAN [Karnewar et al.]; MSGGAN ... on SceneNet lead to unrealistic blurry results (fig. I). (2) Insufficient coverage of the image distribution, whose evidence was an inability to recover latent codes that lead to a correct reconstruction of arbitrary held-out images.

R1: Hyperparam opt? Grid search over 3 weights \( \{10^{-i}\}_{i=3}^{2} \) for each of 3 losses on 100-scene subset of the train set.

R1: Show more supplementary. As suggested, we will expand the supplementary with more qualitative results.

R1: Why optimizing only non-holes of \( \bar{v} \) (Eq. 3)? \( \bar{v} \) is a point cloud renderer and can contain holes. Minimizing \( h(\bar{v}, \bar{u}) \) over holes \( \bar{u} \) would make \( \bar{v} \) attain an unrealistic color of a hole (black by default) which is not desirable.

R1: The approach requires depth as input. We have now implemented a method requiring ground truth (GT) depth solely at train time. We replaced the GT reference view depth with an output of a depth predictor [Laina et al.: Deeper ...] trained on the ScanNet train set. Again, PerspectiveNet outperforms other baselines (Table II).

R2: Not mentioning depth as a required input. We will update the text accordingly to avoid misleading readers.

R2: Evaluation only on 1 dataset. As suggested, we have now conducted evaluation on Matterport3D and SceneNet (same train/test protocol as for ScanNet). Note that SceneNet is synthetic and composed of ShapeNet objects and, hence, is more suitable for our scene-centric setting than the object-centric ShapeNet. Tables (Ib) and (Ic) contain results of our experiments. Similar to Tab. 1 in paper, PerspectiveNet outperforms other approaches. Unfortunately, due to limited amount of time, we could not finish all 3DConvNet experiments (we will include them in camera-ready).

R2: Test GQN on real data? We have now trained&tested GQN on ScanNet. GQN failed to learn and attained poor quantitative results - Ours/GQN: \( \ell_{1RGB}^{RG} = 67.77/165.70, \) PSNR=13.79/6.96, LPIPS=0.434/0.687, \( \ell_{1}^{D} = 0.109/0.513. \) The failure to learn probably occurs due to a greater complexity of ScanNet compared to GQNs’ simplified synthetic scenes.

R2: Is 3D ConvNet a contribution? The 3D ConvNet was designed as a baseline and we compare with.

R2: Range/units of depth \( d_u \).? The depth is always expressed in meters. Range is roughly \([0, 7]\) meters.

R2: Which layers for residuals? \( \Delta \phi^i \) were added after every “upsample&add” layer of FPN (four \( \Delta \phi^i \) in total).

R2: L209: Are the 8 views used for testing? We have now trained&tested GQN on ScanNet. GQN failed to learn and attained poor quantitative results - Ours/GQN: \( \ell_{1RGB}^{RG} = 67.77/165.70, \) PSNR=13.79/6.96, LPIPS=0.434/0.687, \( \ell_{1}^{D} = 0.109/0.513. \) The failure to learn probably occurs due to a greater complexity of ScanNet compared to GQNs’ simplified synthetic scenes.

R2: View clustering? Given \( N \) cameras, we KMeans-clustered the set of corresponding descriptors \( \{ \vec{v}(g) \}_{i=1}^{N} \).

R2: Loss weights? Train/test split? \( w(\ell_{\text{style}}, \ell_{\text{cons}}, \ell_{R}) = (0.1, 0.01, 0.1) \). Using official train/test split of ScanNet.

R2: Explain perception of improvements in LPIPS / PSNR / \( \ell_{1} \). PSNR and \( \ell_{1RGB}^{RG} \) are sensitive to low-frequency image details while LPIPS better assesses image realism. Hence, the +8/-1% improvement of PerspNet over PerspNet w/o. opt in LPIPS / PSNR means that, while the local color distributions are roughly correct in both cases, adding the scene-consistent optimizer brings better image realism and an image-to-image consistent inpainting.

R2: Performance analysis. While PerspectiveNet brings better image quality, it is fair to admit that this comes at the cost of sub-real-time execution times (~20s per scene).

R3: Discuss differences with [Meshery et al.]. We agree that there are similarities with the work of Meshery et al. [a] and we will cite this paper in Sec. 2. However, our work differs substantially in: (1) The task: While we focus on precise reconstruction of geometry and appearance of a scene given a limited amount of information in form of an image with large undefined regions, [a] is a form of stylization that aims at capturing a complete distribution of possible appearance variations of a, mostly hole-free, image. (2) Available data: [a] uses 1000s of reference images to reconstruct a scene that is later re-rendered. We use only 4 reference views, leading to large holes in new views and significantly harder inpainting problem. Furthermore, [a] requires semantic segmentation of the scene.

Finally, please note that [a] uses a BiGAN approach which we compare with in our work and outperform it significantly.

Table I: Additional results on test sets of Matterport3D, SceneNet, ScanNet (will be included in camera-ready).

<table>
<thead>
<tr>
<th>Dataset</th>
<th>(a) ScanNet w/o test-time GT depth</th>
<th>(b) SceneNet</th>
<th>(c) Matterport3D</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>( \ell_{\text{PSNR}} )</td>
<td>( \ell_{\text{LPIPS}} )</td>
<td>( \ell_{\text{PSNR}} )</td>
</tr>
<tr>
<td>PerspectiveNet</td>
<td>93.819</td>
<td>11.193</td>
<td>0.515</td>
</tr>
<tr>
<td>PerspectiveNet w/o opt</td>
<td>94.333</td>
<td>11.224</td>
<td>0.537</td>
</tr>
<tr>
<td>PartialConv</td>
<td>96.742</td>
<td>10.948</td>
<td>0.516</td>
</tr>
<tr>
<td>3DConvNet</td>
<td>95.942</td>
<td>12.614</td>
<td>0.614</td>
</tr>
<tr>
<td>BiGAN</td>
<td>156.958</td>
<td>7.194</td>
<td>0.715</td>
</tr>
</tbody>
</table>