

Towards Full-to-Empty Room Generation with Structure-Aware Feature Encoding and Soft Semantic Region-Adaptive Normalization

Vasileios Gkitsas, Nikolaos Zioulis, Vladimiro Sterzentsenko, Alexandros Doumanoglou, Dimitrios Zarpalas
Centre for Research and Technology Hellas, Thessaloniki, Greece

Keywords: Deep Learning, Omnidirectional Vision, Image-to-Image translation, Depth Estimation

Abstract: The task of transforming a furnished room image into a background-only is extremely challenging since it requires making large changes regarding the scene context while still preserving the overall layout and style. In order to acquire photo-realistic and structural consistent background, existing deep learning methods either employ image inpainting approaches or incorporate the learning of the scene layout as an individual task and leverage it later in a not fully differentiable semantic region-adaptive normalization module. To tackle these drawbacks, we treat scene layout generation as a feature linear transformation problem and propose a simple yet effective adjusted fully differentiable soft semantic region-adaptive normalization module (softSEAN) block. We showcase the applicability in diminished reality and depth estimation tasks, where our approach besides the advantages of mitigating training complexity and non-differentiability issues, surpasses the compared methods both quantitatively and qualitatively. Our softSEAN block can be used as a drop-in module for existing discriminative and generative models. Implementation is available on [vcl3d.github.io/PanoDR/](https://github.com/vcl3d/PanoDR/).

1 Introduction

In recent years, deep learning has witnessed an unprecedented pace of improvement, most notably concerning the generation of high-dimensional content. Contemporary approaches that leverage generative adversarial networks (Goodfellow et al., 2014) have shown impressive achievements in generating realistic images after sampling from distribution as well as various applications including image inpainting and image-to-image translation.

This paper focuses on translating fully-furnished rooms into empty ones. Specifically, the task aims to hallucinate the occluded regions of an input image, thus after translation, yields an image from the same distribution but with a different context. In addition, the growing interest in AR/VR applications has increased the need for assisting applications in improving the user’s experience. Concerning interior redecoration applications, diminishing objects from a scene is of paramount importance, a task that can be approached by explicitly translating the existing scene to a background-only scene.

Moreover, 360° devices get popularized, with multiple panorama datasets (Armeni et al., 2017; Chang et al., 2017; Zheng et al., 2020a) being available to facilitate the contemporary deep-learning-based methods. The wide field-of-view, provided by 360° cameras further motivates the development of

image synthesis approaches in the 360° domain since it provides enough surrounding context information.

Concerning occluded areas generation, the surrounding context must be rich to aid the generation process. In a sense, synthesizing occluded areas of a scene can be approached by image inpainting. Nonetheless, while image inpainting manages to generate plausible images that adhere to the distribution of the target image, it neglects to preserve the fidelity of the occluded structure. This demandingness makes the task lean toward an image-to-image translation problem. On the other hand, image-to-image translation methods aim to translate an image from one domain to another, with one of its applications being the transformation of semantic labels to real images. In that direction, owing to the need for both preserving the structure of the concealed region and the generation of plausible structures, (Gkitsas et al., 2021) manage to approach the problem using a hybrid approach. However, this approach heavily depends on the necessity of a pre-trained dense layout network to predict the three dominant semantic classes of the scene (floor, wall, ceiling). This demand not only makes it hard to re-train or fine-tune the model on new datasets but also makes the training procedure complex and time-consuming. Further, the style modulation derived from (Zhu et al., 2020) does not take into account the hole-filling nature in the case of diminishing tasks. Specifically, the styles

used for modulating the layer activations are not selected by considering the neighborhood of the hole but rather explicitly derive styles from the whole image. Finally, the single-stage approach is vulnerable to artifacts, principally in challenging scenarios.

In this work, we investigate empty room generation from fully-furnished ones, as a probe to comprehend the capability of neural networks to generate occluded regions of a scene. This is achieved by employing a two-stage, coarse-to-fine architecture. To assist our network with the inferred scene structure, we overcome the semantic segmentation network requirement (Gkitsas et al., 2021) by exploiting the feature space of the coarse network and train our model end-to-end.

In summary, our contributions are:

- We propose a full-to-empty room generation model that learns simultaneously to infer the scene dense layout, showing the benefits of end-to-end training for both training simplicity and model performance.
- Using an adapted semantic region adaptive normalization layer, we prove that we do not have to resort to non-differentiable semantic maps to modulate the layer activations.

2 Related Work

Image-to-image Translation: Image-to-image translation approaches aim at translating a given source image to a corresponding image of a target domain. Over the last years, these approaches have gained increased attention, due to their applicability in a wide range of computer vision applications. Isola et al. (Isola et al., 2017) first introduced the use of conditional GANs for tasks as translating semantic labels to images. In view of the recent works of conditional adversarial networks, SPADE (Park et al., 2019) introduces the spatially-adaptive normalization layer in order to propel the semantic information provided by semantic masks in the deeper layers of the network. The modulation for the activations in normalization layers is accomplished via a spatial adaptive learned transformation. Accordingly, SEAN residual block (Zhu et al., 2020) follows the same path while tackling the two shortcomings of SPADE. First, the use of only one style code to control the style of the whole generated image, and second, the absence of style contribution in the deeper layers of the network. Both drawbacks are alleviated by incorporating for each semantic class its corresponding style and thus using this style information via spatially varying normalization



Figure 1: Modeling the full-to-empty room generation task for spherical panoramas using our proposed approach. From top to bottom: The input panorama, fully-furnished, next the compared method (Gkitsas et al., 2021) generated background panorama and the one predicted from our method, and finally the ground-truth. It is easily observable that our method produces more realistic results based on the ground truth empty panorama. By making the semantic region-adaptive normalization layer fully differentiable, the style modulation can effectively retain the style of the scene background even in extremely challenging cases.

parameters.

Multi-Task Learning: Deep learning multi-task methods aim at improving learned representation via simultaneously utilizing multiple learning-based tasks. Such approaches have been applied in several applications (Liu et al., 2015; Jaderberg et al., 2016). An important application in this context is semantic segmentation. The incorporation of semantic segmentation task has been studied to perform detection or instance segmentation (Gidaris and Komodakis, 2015; Chen et al., 2015; Pinheiro et al., 2016). Recently introduced, (Xu and Zheng, 2021), employs multi-task learning by leveraging the capability of generative networks to encode image semantics in its internal feature maps. Using a pre-trained GAN for generating an image from a latent vector, a simple linear transformation in the feature space is sufficient to provide the semantic segmentation map. To supervise the model, the ground truth semantic mask is obtained by a pre-trained semantic segmentation network while the standard cross-entropy loss is used as a loss function.

Image Inpainting: Traditional image inpainting methods fill the missing content by either searching the most similar patches in the background (Barnes

et al., 2009) or propagating neighboring structures (Sun et al., 2005). However, these approaches struggle when large regions are to be filled or content is unique and not present in the rest of the image. On the other hand, modern approaches (Iizuka et al., 2017; Yu et al., 2018; Yu et al., 2019) leverage the recent advantages in deep learning to fill the missing regions by learning from a large corpus of data. More recently, some approaches assist the generation process by first estimating structural information such as edges (Nazari et al., 2019) and edge-preserved smooth structures (Ren et al., 2019).

Diminished Reality: Diminished Reality (DR) is the process of removing objects that are perceivable in our visual system. In order to diminish an object from a perceived view, background information is required. This prerequisites for the viewer to observe the occluded region from a different viewpoint or in advance (Mori et al., 2017). Nonetheless, this is not feasible in cases where real-time requirements or technical substantial inability occur. In those cases, the occluded areas can be inferred by image inpainting approaches. (Gkitsas et al., 2021) introduced a hybrid image inpainting, and image-to-image translation method to approach the DR problem. More specifically, first, the dense layout of the scene was inferred by a pre-trained semantic segmentation network. Next, the occluded regions were synthesized by a single-stage generator. Additionally, SEAN residual blocks (Zhu et al., 2020) were employed to modulate the normalized activations using the dense layout and the style codes from the input image.

3 Approach

In this section we present our approach, which is depicted in Fig. 2 for modeling the translation from full to empty rooms.

First, we introduce the architecture of our generator, which is composed of a coarse and a refine network. Second, we propose a method for inferring the scene dense layout mask by leveraging the coarse network features. Next, we present an approach for overcoming the demand for a non-differentiable semantic mask as input in the SEAN residual block. Last but not least, regarding the diminishing task, we present a simple yet efficient method for enforcing styles close to holes to dominate in the generated occluded areas.

3.1 Coarse Network

Apart from the challenges posed by the need for hallucinating occluded areas, inferring such a region with-

out first obtaining a coarse estimation is a challenging task. Albeit there exist one stage approaches for image inpainting (Zheng et al., 2019; Li et al., 2020) and diminished reality (Gkitsas et al., 2021), the quality of the generated content is prone to generate artifacts, especially when large holes occur. To mitigate this ambiguity we follow a two-stage coarse-to-fine architecture for our generator. Firstly, the input image, of size 256×512 , is down-sampled to resolution 128×256 . Furthermore, we follow a slim architecture to reduce the parameters. The scope of the coarse network is to produce a coarse prediction, \mathcal{I}_c , that will be fed as input to the refine network in order to facilitate the hallucination of the occluded areas. In addition, the dense layout map of the scene \mathcal{M}_s is generated from the feature space of the decoder. Formally, given an input, furnished image, \mathcal{I}_f , with foreground objects to be removed, masked, we desire to learn a mapping for the coarse network, G^c , such that

$$\{\mathcal{I}_c, \mathcal{M}_s\} = G^c(\mathcal{I}_f)$$

3.2 Linear Transformation on Coarse Network Feature Space

To provide the dense layout map of the scene, \mathcal{M}_s in the fine stage without necessitating a pre-trained model (Gkitsas et al., 2021), we follow a nuanced approach. More specifically, assuming the coarse generator G^c is composed of an encoder E^c and a decoder D^c , then D^c , in the i^{th} layer, comprises $x_i^c \in \mathbb{R}^{h_i \times w_i \times c_i}$ feature maps. We denote with h_i, w_i, c_i , the height, width and channels of the i^{th} layer, respectively. Additionally, we denote as $\mathcal{X} \in \mathbb{R}^{n \times h \times w}$ the upsampled feature maps, to the output image resolution, of the D^c , concatenated along the depth axis.

Given an input furnished image \mathcal{I}_f to E^c , we seek to estimate a coarse prediction of the occluded background \mathcal{I}_c , alongside the dense layout of the scene. Our intuition for obtaining the latter stems from the hypothesis that the feature maps \mathcal{X} encode the semantics of the three abundant classes of an indoor scene (ceiling, wall, floor) and can be inferred by applying a linear transformation on \mathcal{X} .

Therefore, we aim to learn a mapping F such that:

$$\mathcal{M}_s = F(\mathcal{X})$$

Inspired by (Xu and Zheng, 2021), this mapping F can be a linear transformation, defined as:

$$\mathcal{M}_s = \sum_{i=1}^{N-1} T_i(u_i x_i^c) = T \cdot \mathcal{X}$$

where u_i is the upsampling operation for the i^{th} layer, and $T \in \mathbb{R}^{m \times n}$.

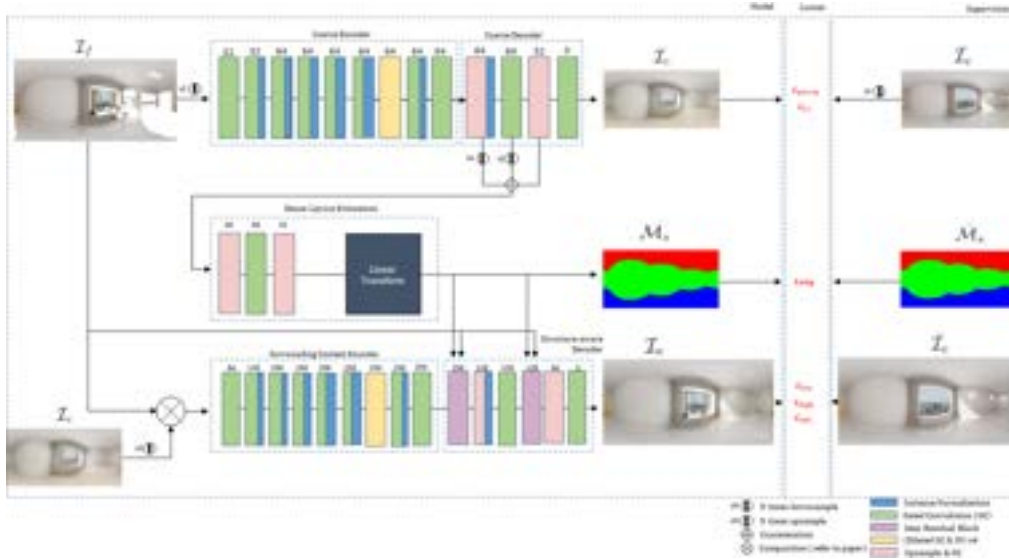


Figure 2: The architecture of the proposed method, along with the flow of data, supervision, and losses used in each stage of the end-to-end train. First, the input panorama is masked in the foreground area yielding \mathcal{I}_f and fed to the coarse network. Next, after predicting the coarse background image \mathcal{I}_c , the dense layout of the scene \mathcal{M}_s is estimated by first applying upsampling and concatenation in its decoder feature space and afterward applying a linear transformation. Following, \mathcal{I}_c is composited with \mathcal{I}_f and fed to the refine network. After obtaining the encoded representation from the surrounding context encoder, \mathcal{M}_s is used along with style code obtained from \mathcal{I}_f for modulating the structure-aware decoder activations via SSEAN block. The generated background image, \mathcal{I}_c is supervised by employing a low-level loss, a high-level loss as well as an adversarial loss. The layout estimation is supervised by the standard cross-entropy loss. The architectures of the discriminator and SSEAN block are omitted for brevity.

3.3 Refine Network

After obtaining the coarse estimation \mathcal{I}_c , we seek to eliminate generated artifacts from the coarse output via the refine stage. The architecture of the refine network G' is derived from (Gkitsas et al., 2021). In order to exploit the obtained \mathcal{M}_s from the coarse stage, we modify the blocks that leverage the dense layout map for modulating the normalized activations of the decoder.

3.4 Soft Semantic Region-Adaptive Normalization

In the recently introduced semantic region-adaptive normalization (SEAN) layer (Zhu et al., 2020), the generation process is assisted by conditioning its output on one style code per semantic region. The original formulation in (Zhu et al., 2020) assumes one hot encoding for the semantic labels maps, which is typical in ground-truth annotated datasets. However, this demand hinders its applicability in end-to-end trainable models, due to the fact that the operations involved are not fully differentiable when the semantic label maps are provided by a predictive model, since in that case, the one-hot encoding is superseded

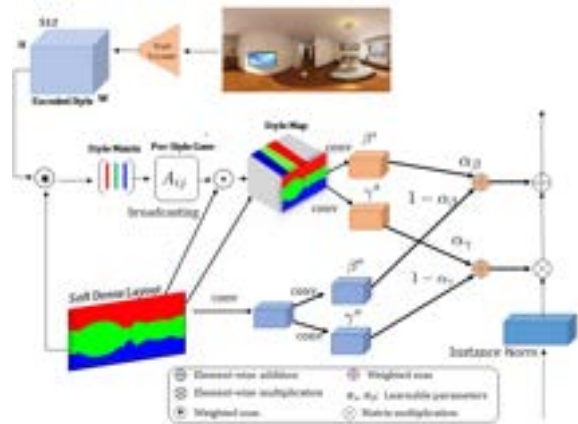


Figure 3: Our Soft Semantic Region-Adaptive Normalization module which is built upon SEAN. First, the soft layout semantic mask is applied with a weighted sum on the encoded style, and afterward with a matrix multiplication with the styles obtained for each semantic region. For more info please refer to section 3.4.

by a probability distribution among possible semantic labels. In order to overcome this shortcoming, we make two adjustments in (Zhu et al., 2020), the first related to the style encoder and the second related to the SEAN block.

In SEAN, the purpose of the style encoder is to

encode one style code per semantic label. This is represented by a style matrix $\mathbf{ST} \in \mathbb{R}^{C \times D}$ where C denotes the total number of possible semantic labels and D the dimensionality of the style code. \mathbf{ST} is computed via region-wise average pooling, from an intermediate matrix $\tilde{\mathbf{ST}} \in \mathbb{R}^{D \times H \times W}$ based on the semantic segmentation map $\mathbf{M} \in \mathbb{R}^{C \times H \times W}$, essentially averaging the style codes at all spatial locations for a given class. To make the process fully differentiable in cases where a pixel belongs to a class with a given probability, not necessarily strictly either zero or one (the previously mentioned one-hot encoding assumption), instead of using the one-hot encoded \mathbf{M} , we employ \mathcal{M}_s . Let $\tilde{\mathbf{s}}_{i,j} \in \mathbb{R}^D$ denote the style code at the spatial location (i, j) in $\tilde{\mathbf{ST}}$. Further, let $p_{i,j,c} \in [0, 1]$ denote the probability that the pixel at spatial location (i, j) , belongs to class c as defined by the value of the respective cell in \mathcal{M}_s . Then, the style code $\mathbf{s}_c \in \mathbb{R}^D$ for the class c is computed by the weighted sum:

$$\mathbf{s}_c = \frac{1}{\sum_{i,j} p_{i,j,c}} \sum_{i,j} p_{i,j,c} \tilde{\mathbf{s}}_{i,j} \quad (1)$$

and the matrix \mathbf{ST} is constructed by stacking $\{\mathbf{s}_c, \forall c\}$.

Regarding the SEAN block, we replace the broadcasting operation that produces the stylemap $\mathbf{SM} \in \mathbb{R}^{D \times H \times W}$ in a similar fashion. In particular, let $\mathbf{s}'_c \in \mathbb{R}^D$ denote the style code for class c which is computed after the 1×1 convolution with the respective style code in \mathbf{ST} . In this context, the purpose of the broadcast operation is to fill the stylemap’s spatial locations at each pixel (i, j) with the corresponding style code of the pixel’s semantic label. The original broadcasting operation in SEAN does not take into account probability distributions other than one-hot. We make this operation soft and fully differentiable by assigning to each pixel of the stylemap the sum of all $\mathbf{s}'_c \forall c$ weighted by the pixel’s probability to belong to class c . More formally, let $\mathbf{sm}_{i,j}$ denote the style code in style map spatial location (i, j) . Then:

$$\mathbf{sm}_{i,j} = \sum_c p_{i,j,c} \mathbf{s}'_c \quad (2)$$

In both previously mentioned modifications in order to reduce the effect of mixing style codes belonging to different semantic labels, we pre-process the semantic label map \mathcal{M}_s (and consequently all $p_{i,j,c}$) via a sharpening operation powered by a softmax transformation parameterized by sharpening constant K :

$$p'_{i,j,c} = \frac{e^{p_{i,j,c}/K}}{\sum_c e^{p_{i,j,c}/K}} \quad (3)$$

In our experiments we empirically set $K = 0.1$. To that end, we polarize $p_{i,j,c}$ towards the extreme values of 0 and 1.

With those two modifications, we make SEAN fully differentiable and compatible with input segmentation masks following arbitrary probability distributions across semantic labels. Other than that, as depicted in Fig.3, we keep the rest of the SEAN pipeline intact.

3.5 Supervision

In order to obtain the background image for a furnished on, we combine several losses to obtain :

$$\mathcal{L} = \mathcal{L}_{low} + \mathcal{L}_{high} + \mathcal{L}_{adv} + \mathcal{L}_{seg}. \quad (4)$$

A low level reconstruction loss \mathcal{L}_{low} , a high level synthesis loss \mathcal{L}_{high} , an adaptive adversarial loss \mathcal{L}_{adv} and a layout estimation loss \mathcal{L}_{seg} .

Low-level Reconstruction Loss. This pixel-based loss focuses on the reconstruction of low frequency components of the predicted image \mathcal{I}_e :

$$\mathcal{L}_{low} = \lambda_{L1} \frac{1}{|\Omega|} \|\mathbf{vec}(\mathcal{A} \odot |\tilde{\mathcal{I}}_e - \mathcal{I}_e|)\|_1 + \lambda_{TV} \left(\frac{1}{|\Omega_{\nabla_x}|} \|\mathbf{vec}(\mathcal{A} \odot |\nabla_x \tilde{\mathcal{I}}_e|)\|_1 + \frac{1}{|\Omega_{\nabla_y}|} \|\mathbf{vec}(\mathcal{A} \odot |\nabla_y \tilde{\mathcal{I}}_e|)\|_1 \right) \quad (5)$$

where $|\Omega|, |\Omega_{\nabla_x}|, |\Omega_{\nabla_y}|$ are the total number of pixels in $\tilde{\mathcal{I}}_e$ and in the respective gradient images in x, y directions. $\mathcal{A} \in \mathbb{R}^{W \times H}$ is the spherical attention mask used in (Zioulis et al., 2019) that accounts for equirectangular distortion, while the \mathbf{vec} operator treats its matrix argument as a flattened vector, and $\|\cdot\|_1$, denotes the L1-norm. Finally, \odot denotes the Hadamard product.

Apart from the spherically weighted L1 loss, a total variation smoothness prior is used for the diminished area specifically to counter the high frequency artifacts usually seen in the early training stages of generative models.

High-level Synthesis Loss. Apart from encouraging \mathcal{I}_e and $\tilde{\mathcal{I}}_e$ to have the same representation at the pixel level with \mathcal{L}_{low} , we additionally employ a data-driven loss \mathcal{L}_{high} . This enforces them to have a similar representation in the feature space as computed by a CNN model Φ , which in our case, is a pre-trained VGG-19 (Simonyan and Zisserman, 2014). Let $\Phi_j(\mathcal{I}) \in \mathbb{R}^{C_j \times H_j \times W_j}$ be the tensor of activations of the j -th layer of the network Φ with C_j channels, for the given image \mathcal{I} , and $|\Omega_{\Phi_j}|$ the total number of elements of the tensor.

Then the loss is formulated as a combination of the perceptual and style losses:

$$\mathcal{L}^{high} = \lambda_{perc} \mathcal{L}_{perc} + \lambda_{style} \mathcal{L}_{style} \quad (6)$$

$$\mathcal{L}_{perc} = \sum_j \frac{1}{|\Omega_{\Phi_j}|} \|\mathbf{vec}(\Phi_j(\bar{\mathcal{I}}_e) - \Phi_j(\mathcal{I}_e))\|_1 \quad (7)$$

$$\mathcal{L}_{style} = \sum_j \frac{1}{C_j} \|\mathbf{vec}(\mathcal{G}(\Phi_j(\bar{\mathcal{I}}_e)) - \mathcal{G}(\Phi_j(\mathcal{I}_e)))\|_1 \quad (8)$$

$\mathcal{L}_{perc}, \mathcal{L}_{style}$ are the perceptual and style (Gatys et al., 2016; Johnson et al., 2016) losses, and $\mathcal{G}(\mathbf{M}) = \mathbf{MM}^T, \mathbf{M} \in \mathbb{R}^{C_j \times (H_j \cdot W_j)}$ is the Gram matrix function. Both losses are derived in a high dimensional data-driven feature space, with the former (perceptual) operating on a global level, and the latter (style) operating on global and local levels.

Adaptive Adversarial Loss. To adaptively improve the quality of the generated background images \mathcal{I}_e we additionally employ a discriminator-based loss that is learned during training. Since we use a PatchGAN discriminator, we formulate our combined adversarial loss as a combination of a hinge loss on the final real/fake predictions (Lim and Ye, 2017), and a feature matching loss using the discriminator’s intermediate features:

$$\mathcal{L}_{adv} = \lambda_D \mathcal{L}_D + \lambda_{FM} \mathcal{L}_{FM} \quad (9)$$

$$\mathcal{L}_D = \frac{1}{|\Omega_d|} \left(\|\mathbf{vec}(r(1 - \mathbf{d}_e))\|_1 + \|\mathbf{vec}(r(1 + \mathbf{d}_\hat{e}))\|_1 \right) \quad (10)$$

$$\mathcal{L}_{FM} = \sum_i \frac{1}{|\Omega_d^i|} \|\mathbf{vec}(\mathbf{d}_e^i - \mathbf{d}_\hat{e}^i)\|_1, \quad (11)$$

where \mathbf{d}_e and $\mathbf{d}_\hat{e}$ are the discriminator outputs for the real and predicted background images, Ω_d is the element domain of the discriminator’s output, $|\Omega_d|$ the total count of its elements, while i denotes intermediate discriminator feature maps and $|\Omega_d^i|$ their spatial element count. Finally, r stands for the ReLU activation. The spatial discriminator hinge loss and the feature matching loss are weighted by their respective weights. Feature matching enforces the generator to minimize the statistical difference between the features of the ground truth images and the generated images, which helps further stabilize the training and improve the quality of the generated content.

Layout Estimation loss: To supervise the dense layout estimation, we use the focal loss (Lin et al., 2017) which is proven to penalize the network better than the standard cross entropy loss on hard negative examples:

$$\mathcal{L}_{seg} = a(1 - p_t)^\gamma \mathcal{L}_{ce}(\bar{\mathcal{M}}_s, \mathcal{M}_s) \quad (12)$$

With p_t and \mathcal{L}_{ce} we denote the probabilities of the target class and the standard cross entropy loss, respectively. For our experiments we set $\alpha = 0.25$ and $\gamma = 2$.

4 Results

Implementation details. We implement our model using PyTorch (Paszke et al., 2017) with all experiments conducted on a Nvidia GeForce RTX 3090 GPU. Our generative models are optimized using Adam (Kingma and Ba, 2014), with $b_1 = 0.5$ and $b_2 = 0.999$, a learning rate of 0.0002 and a batch size of 6. The input and output panorama resolutions are 256×128 for the coarse network and 512×256 for the refine. The weights the models are initialized from a zero-centered Normal distribution with $\sigma = 0.02$. We empirically set $\lambda_{L1} = 4.0$, $\lambda_{TV} = 1.0$, $\lambda_{perc} = 0.15$, $\lambda_{style} = 40.0$, $\lambda_D = 0.2$ and $\lambda_{FM} = 20.0$.

Experiments. We compare our proposed method against the set of the current state-of-the-art methods of PanoDR (Gkitsas et al., 2021), RFR (Li et al., 2020), PICNet (Zheng et al., 2019). Moreover, to highlight the effectiveness of our method, we compare it against PanoDR-e2e. PanoDR-e2e is considered as the official work, but trained end-to-end, using our adapted SEAN residual block. All training configurations use the same adaptation of the Structured3D (Zheng et al., 2020b) dataset as in (Gkitsas et al., 2021), with fixed seeds and using the official train/test splits of (Zheng et al., 2020b). The mentioned adaptation enables the applicability of the original dataset for diminished reality applications. For more details please refer to the original work of (Gkitsas et al., 2021).

Quantitative comparisons. Table 1 shows the performance of each method on the standard set of metrics, Mean Absolute Error (MAE), Peak to Signal-to-Noise Ratio (PSNR), Structural Similarity Index (SSIM), Fréchet inception distance (FID), and Learned Perceptual Image Patch Similarity (LPIPS), with the three first expressing similarity in a spatially local manner (pixel-wise or in small patches) and the latter two in a perceptual-global manner. More precisely, LPIPS compares the features extracted by a pre-trained VGG-16 model, rather than the images themselves, with the rationale being that the extracted features can be more expressive in local regions of the image. On the same page, FID compares the high-dimensional feature distributions of the predicted and ground-truth images, which assesses how close these distributions are across the dataset. Given the nature of our task, we aim to maximize the performance of

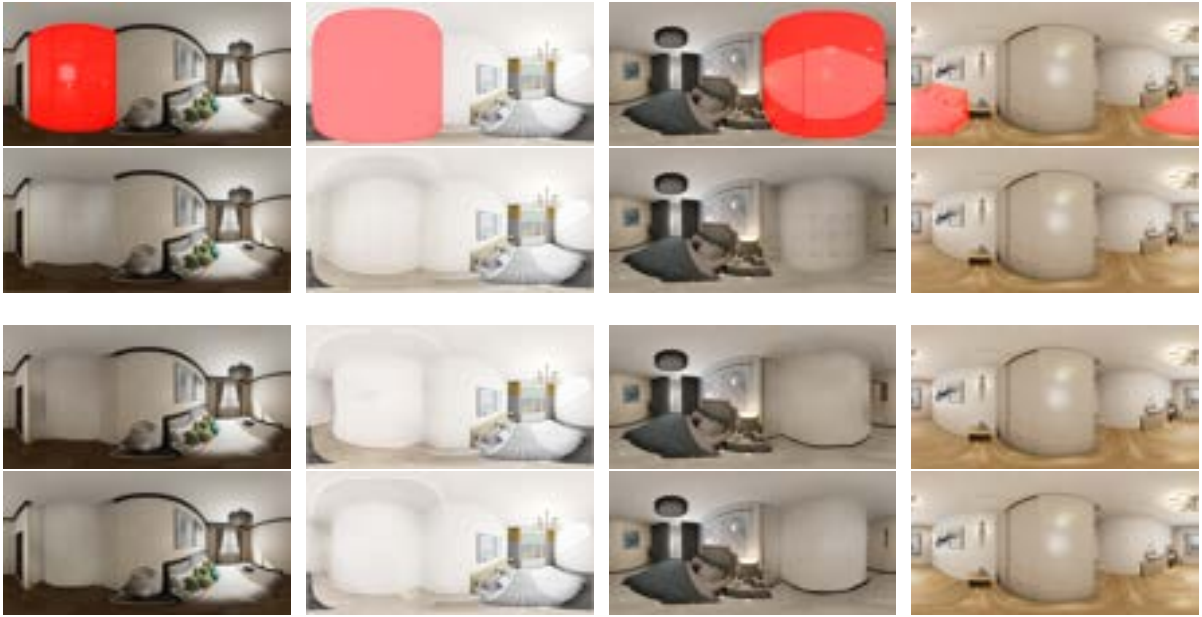


Figure 4: Qualitative comparison on diminished reality application from scenes in our test set. From top to bottom: Input image with the diminished area masked with transparent red, PICNet, PanoDR, and ours.

the perceptual metrics, since preserving the the structure of the room is more visually appealing, rather than some minor photometric inconsistencies. Additionally, to assess the boundary preservation for the generated image, we follow the mIoU estimation for the introduced in (Gkitsas et al., 2021), by applying a pre-trained semantic segmentation network on the generated image and comparing it with the ground truth.

Diminished reality application. Regarding the diminished reality application, given in Table 1, it is apparent that our method not only surpasses the compared methods in terms of perceptual metrics (LPIPS, FID) but also exhibits equivalent performance concerning boundary preservation. More specifically, the model performance in terms of FID and LPIPS increases by 2.1% and 16.6% over the baseline, respectively whilst PSNR, SSIM, mIoU and MAE do not exhibit significant variations. This performance gain is attributed to the adapted SEAN residual block, which manages to handle in a better manner the style modulation of the scene.

Qualitative comparisons In order to further assess the quality of the model performance, we take a closer look at the qualitative comparisons. About the full-to-empty room generation, it can be observed in Fig.1 that our method better preserves the overall hue of the scene than the compared method. In such challenging scenes where the objects that are to be diminished cover a large part of the scene, it is crucial the generated image preserves both structure and the

hue that depict the scene. For instance, in the first row of Fig. 1 the floor of the scene is almost covered by the bed and furniture of the room. PanoDR misses generating floor with the realistic visual result, while exhibits severe flaws with regard to the structure of the scene. Similarly, in the second row, PanoDR not only misses capturing the hue of the floor but also its output is blurry at the lower side of the scene. On the other hand, our method generates content with a hue not vastly different than of the ground truth.

Concerning the diminishing application, in Fig.4 we compare our method with PICNet, RFR, and PanoDR. The compared methods are prone to generating blurry images and in some cases, artifacts. In contrast, our approach achieves visually appealing results given the challenging nature of that cases. For example, one can see in the second column that the texture of the generated image is compatible with that of the surrounding context. In addition, albeit the object to be diminished covers almost the 40% of the scene, its output is free of artifacts and blurry regions.

Monocular Depth Estimation task

To further evaluate the softSEAN block, we conduct another experiment on a dense regression task, monocular depth estimation using the Structured3D dataset. We adapt UNet architecture (Ronneberger et al., 2015) and derive the direct supervision from (Zioulis et al., 2019). Similarly, we employ a linear transformation, L_t to obtain the soft semantic mask of the input scene. Following a different line compared with the diminished reality task, we use the features

Table 1: Quantitative comparison on diminished reality application. Six metrics are used, FID, LPIPS, PSNR, SSIM, MAE and structural preservation (mIoU) on the Structured3D test set (\downarrow means lower is better, and \uparrow means higher is better).

Method	FID \downarrow	LPIPS \downarrow	PSNR \uparrow	SSIM \uparrow	MAE \downarrow	mIoU \uparrow
RFR (Li et al., 2020)	7.2474	0.0510	31.0114	0.9528	0.0067	0.8583
PICNet (Zheng et al., 2019)	6.7063	0.0533	32.3072	0.9557	0.0070	0.8502
PanoDR (Gkitsas et al., 2021)	6.8374	0.0398	33.6611	0.9620	0.0058	0.8768
PanoDR(e2e)	7.2052	0.0357	33.6681	0.9622	0.0060	0.8488
Ours	6.6915	0.0320	33.6576	0.9624	0.0058	0.8789

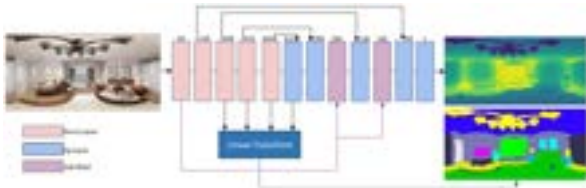


Figure 5: The architecture of the UNet adapted model using softSEAN block for depth estimation task. First, the input image is fed to the model, next, a linear transformation is applied to obtain the soft semantic segmentation of the scene, which is used to modulate the activation layers in the decoder.

from the second layer of the encoder to modulate the layers in the decoder. The main goal is to enhance the predictions for local regions through integrating the semantic map, composed of 41 classes. The network architecture is depicted in Fig.5. Quantitative results are presented in Table 2, for which we used typical metrics. The results enhance previous findings in that the softSEAN residual block along with the exploitation of the linear transformation block enforces the network to better predictions. Fig.6 clearly illustrates the benefits of our method, in terms of both local and global regions of the predicted depth map. For instance, in the second row, the baseline model misses predicting the correct depth due to texture transfer (highlighted on the left) while fails at capturing local objects' depth (highlighted on the right).

5 Conclusion

In this work, we propose an approach for indoor spherical panoramas, in which an empty room is generated from a full-furnished one. The core idea of our method lies in using a two-stage coarse-to-fine network. First, a lightweight network is utilized to estimate a coarse prediction of the background while encodes in its features and generating the dense layout of the occluded regions of the scene. Subsequently, to leverage the latter for modulating layers

activations, we adjust the SEAN block in a way that maintains the differentiability of the dense layout. Interestingly, we demonstrate that our method shows consistent improvement over the baselines regarding the diminished reality application while overcomes the barriers of previous methods and is trainable in an end-to-end manner. Further, we believe that the key insight of this work can be applied to room re-decoration and interior design applications. Last but not least, we validate the effectiveness of the soft SEAN residual block via applying it in the depth estimation task, showcasing its efficacy for different computer vision tasks.

Table 2: Results of omnidirectional depth estimation. The first row represents the baseline UNet architecture, the second the adapted model with linear transformation (Lt) and the latter the architecture that encompasses both linear transformation and softSEAN.

Method	RMSE \downarrow	RMSE(log) \downarrow	δ_1 \uparrow	δ_2 \uparrow	δ_3 \uparrow
Baseline	0.4635	0.1738	0.9144	0.9613	0.9695
w Lt	0.4533	0.1684	0.9244	0.9675	0.9759
w Lt+softSEAN	0.3820	0.1585	0.9573	0.9768	0.9809

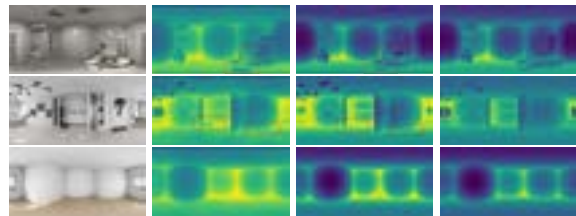


Figure 6: Qualitative results on omnidirectional dense depth estimation on samples of Structured3D. From left to right: input image, baseline, ours (Lt+softSEAN), ground truth.

Acknowledgements. This work was supported by the EC funded H2020 project ATLANTIS [GA 951900].

REFERENCES

- Armeni, I., Sax, S., Zamir, A. R., and Savarese, S. (2017). *Joint 2d-3d-semantic data for indoor scene understanding*. 1
- Barnes, C., Shechtman, E., Finkelstein, A., and Goldman, D. B. (2009). *PatchMatch: A randomized correspondence algorithm for structural image editing*, volume 28. 3
- Chang, A., Dai, A., Funkhouser, T., Halber, M., Niebner, M., Savva, M., Song, S., Zeng, A., and Zhang, Y. (2017). Matterport3d: Learning from rgb-d data in indoor environments. In *2017 International Conference on 3D Vision (3DV)*, pages 667–676. IEEE Computer Society. 1
- Chen, X., Kundu, K., Zhu, Y., Berneshawi, A. G., Ma, H., Fidler, S., and Urtasun, R. (2015). 3d object proposals for accurate object class detection. In *Advances in Neural Information Processing Systems*, pages 424–432. Citeseer. 2
- Gatys, L. A., Ecker, A. S., and Bethge, M. (2016). Image style transfer using convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2414–2423. 6
- Gidaris, S. and Komodakis, N. (2015). Object detection via a multi-region and semantic segmentation-aware cnn model. In *Proceedings of the IEEE international conference on computer vision*, pages 1134–1142. 2
- Gkitsas, V., Sterzentsenko, V., Zioulis, N., Albanis, G., and Zarpalas, D. (2021). Panodr: Spherical panorama diminished reality for indoor scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3716–3726. 1, 2, 3, 4, 6, 7, 8
- Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014). *Generative adversarial networks*. 1
- Iizuka, S., Simo-Serra, E., and Ishikawa, H. (2017). *Globally and locally consistent image completion*, volume 36. ACM New York, NY, USA. 3
- Isola, P., Zhu, J.-Y., Zhou, T., and Efros, A. A. (2017). Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1125–1134. 2
- Jaderberg, M., Mnih, V., Czarnecki, W. M., Schaul, T., Leibo, J. Z., Silver, D., and Kavukcuoglu, K. (2016). *Reinforcement learning with unsupervised auxiliary tasks*. 2
- Johnson, J., Alahi, A., and Fei-Fei, L. (2016). Perceptual losses for real-time style transfer and super-resolution. In *European conference on computer vision*, pages 694–711. Springer. 6
- Kingma, D. P. and Ba, J. (2014). *Adam: A method for stochastic optimization*. 6
- Li, J., Wang, N., Zhang, L., Du, B., and Tao, D. (2020). Recurrent feature reasoning for image inpainting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7760–7768. 3, 6, 8
- Lim, J. H. and Ye, J. C. (2017). *Geometric gan*. 6
- Lin, T.-Y., Goyal, P., Girshick, R., He, K., and Dollár, P. (2017). Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988. 6
- Liu, X., Gao, J., He, X., Deng, L., Duh, K., and Wang, Y.-Y. (2015). *Representation learning using multi-task deep neural networks for semantic classification and information retrieval*. 2
- Mori, S., Ikeda, S., and Saito, H. (2017). *A survey of diminished reality: Techniques for visually concealing, eliminating, and seeing through real objects*, volume 9. SpringerOpen. 3
- Nazeri, K., Ng, E., Joseph, T., Qureshi, F., and Ebrahimi, M. (2019). Edgeconnect: Structure guided image inpainting using edge prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, pages 0–0. 3
- Park, T., Liu, M.-Y., Wang, T.-C., and Zhu, J.-Y. (2019). Semantic image synthesis with spatially-adaptive normalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2337–2346. 2
- Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L., and Lerer, A. (2017). *Automatic differentiation in pytorch*. 6
- Pinheiro, P. O., Lin, T.-Y., Collobert, R., and Dollár, P. (2016). Learning to refine object segments. In *European conference on computer vision*, pages 75–91. Springer. 2
- Ren, Y., Yu, X., Zhang, R., Li, T. H., Liu, S., and Li, G. (2019). Structureflow: Image inpainting via structure-aware appearance flow. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 181–190. 3
- Ronneberger, O., Fischer, P., and Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer. 7
- Simonyan, K. and Zisserman, A. (2014). *Very deep convolutional networks for large-scale image recognition*. 5
- Sun, J., Yuan, L., Jia, J., and Shum, H.-Y. (2005). Image completion with structure propagation. In *ACM SIGGRAPH 2005 Papers*, pages 861–868. 3
- Xu, J. and Zheng, C. (2021). Linear semantics in generative adversarial networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9351–9360. 2, 3
- Yu, J., Lin, Z., Yang, J., Shen, X., Lu, X., and Huang, T. S. (2018). Generative image inpainting with contextual attention. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5505–5514. 3
- Yu, J., Lin, Z., Yang, J., Shen, X., Lu, X., and Huang, T. S. (2019). Free-form image inpainting with gated convolution. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4471–4480. 3

- Zheng, C., Cham, T.-J., and Cai, J. (2019). Pluralistic image completion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1438–1447. [3](#), [6](#), [8](#)
- Zheng, J., Zhang, J., Li, J., Tang, R., Gao, S., and Zhou, Z. (2020a). Structured3d: A large photo-realistic dataset for structured 3d modeling. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part IX 16*, pages 519–535. Springer. [1](#)
- Zheng, J., Zhang, J., Li, J., Tang, R., Gao, S., and Zhou, Z. (2020b). Structured3d: A large photo-realistic dataset for structured 3d modeling. In *Proceedings of The European Conference on Computer Vision (ECCV)*. [6](#)
- Zhu, P., Abdal, R., Qin, Y., and Wonka, P. (2020). Sean: Image synthesis with semantic region-adaptive normalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5104–5113. [1](#), [2](#), [3](#), [4](#)
- Zioulis, N., Karakottas, A., Zarpalas, D., Alvarez, F., and Daras, P. (2019). Spherical view synthesis for self-supervised 360° depth estimation. In *2019 International Conference on 3D Vision (3DV)*, pages 690–699. IEEE. [5](#), [7](#)