Generative multiview inpainting for object removal in large indoor spaces

Joohyung Kim¹, Janghun Hyeon¹ and Nakju Doh²,³

Abstract
As interest in image-based rendering increases, the need for multiview inpainting is emerging. Despite of rapid progresses in single-image inpainting based on deep learning approaches, they have no constraint in obtaining color consistency over multiple inpainted images. We target object removal in large-scale indoor spaces and propose a novel pipeline of multiview inpainting to achieve color consistency and boundary consistency in multiple images. The first step of the pipeline is to create color prior information on masks by coloring point clouds from multiple images and projecting the colored point clouds onto the image planes. Next, a generative inpainting network accepts a masked image, a color prior image, imperfect guideline, and two different masks as inputs and yields the refined guideline and inpainted image as outputs. The color prior and guideline input ensure color and boundary consistencies across multiple images. We validate our pipeline on real indoor data sets quantitatively using consistency distance and similarity distance, metrics we defined for comparing results of multiview inpainting and qualitatively.

Keywords
Multiview inpainting, object removal, generative adversarial network, color consistency, boundary consistency

Introduction
The rendering of real indoor spaces is typically achieved via an image-based rendering (IBR) method¹–³ that allows a free-viewpoint exploration in a virtual world. Recent IBR applications¹ have reduced geometric complexity for the real-time rendering of large-scale indoor spaces. The key idea of the method¹ is to render only the architectural components without objects, which requires a consistent object removal over multiple images, as suggested in the literature.³,⁴

Multiview inpainting is a task to achieve image inpainting on multiple images while satisfying two conditions: color consistency and boundary consistency. For example, in Figure 1, images of C₂ and C₃ require image inpainting as ideal case. However, typical image inpainting methods might result in color or boundary inconsistency (cf. Figure 1(b)).

Philip and Drettakis⁵ proposed a method for multiview inpainting. Their proposed method begins with estimating the plane, projecting visible quadrangle parts of images into a common rectified plane, and then filling in the occluded parts using a conventional patch-based method.⁵ Although this method can ensure the color consistency up to the performance level achieved in traditional inpainting algorithm named PatchMatch,⁵ the boundary consistency is not secured as the planar assumption is not applicable in non-planar or complex environments.

¹ School of Electrical Engineering, Korea University, Seoul, Republic of Korea
² Institute of Convergence Science, Korea University, Seoul, Republic of Korea
³ TeeLabs Inc., Seoul, Republic of Korea

Corresponding author:
Nakju Doh, Institute of Convergence Science, Korea University, Seoul, Republic of Korea; TeeLabs Inc.
Email: nakju@korea.ac.kr
Recent advances in image inpainting are based on deep learning approaches, which have shown their efficiency and effectiveness on applying any form of mask. Recent studies further enhance the performance by utilizing guidelines. Although the guidelines themselves ensure the boundary consistency in case the user sets consistent guidelines over the multiple images, the color consistency over multiple images cannot be secured as such approaches consider only a single image.

In this study, we target object removal in the images covering large-scale indoor spaces. We propose a novel pipeline that fills in occluded regions of multiple images in such a way that both the color and the boundary consistencies are well preserved. The proposed pipeline extends the state-of-the-art GConv applied to a single-image inpainting problem.

First, for the color consistency, we create a color prior image, which is a pixel-wise color information observed from visible cameras (e.g. C₁ and C₄ in Figure 1(a)). This color prior is added along with other inputs of the inpainting network such as a masked image, a guideline input, and a mask similar to that of GConv, allowing the inpainted colors to be consistent in multiple inpainted images.

Second, for the boundary consistency, we extend GConv in such a way that imperfect guideline input can be refined within the network. Ideally, there should be no boundary inconsistency when the guidelines are perfectly set by users. However, because it is a laborious task to set the guidelines over multiple images, typical guidelines include various errors such as in the angle or location or missing information.

For an experiment validation, we utilized real indoor data sets enabling a real-time rendering of large-scale indoor spaces. For the color consistency, we developed new measures to evaluate intensity-free color consistency among multiple inpainted images, as well as the similarity between multiple inpainted images and images without occlusions. Given the metrics, the proposed method yields improvements of up to 46.4% and 30.6% over the previous single-image inpainting method EC and GConv, respectively. Regarding the boundary consistency, we conducted a qualitative comparison to determine whether the proposed pipeline predicts the enhanced guidelines.

This article is organized as follows: the second section briefly summarizes related works. The third section describes the proposed pipeline. The fourth section details the experiment results of multiview inpainting on real data sets. Finally, some concluding remarks are given in the fifth section. Our codes are available at https://github.com/kimjh069/generative-MVI.

**Related work**

**Indoor modeling and IBR**

Typical IBR methods require 3D map or model of the environment. Henry et al. use depth cameras integrating depth and color information for robust 3D mapping of indoor environments. The map, however, remains in point cloud rather than mesh, which makes hard for rendering novel viewpoints. Shao et al. introduce interactive approach for semantic modeling of indoor spaces. The approach retrieves 3D models from predefined database, which has limitations on reconstructing the real scene. More recent works focus on the architectural components for better representation of indoor spaces. Ikehata et al. devise structural grammar that divides the elements of indoor environments such as rooms, walls, and objects. The approach builds the indoor model with Manhattan assumption in a single-story buildings. Recent works overcome Manhattan assumption for indoor modeling or integrate the 3D model with IBR.

Hedman et al. proposed a real-time IBR method. However, this method has limitations in rendering larger than room-scale spaces. Turner et al. reconstruct an architectural mesh but do not remove objects in the images, which results in flattened object textures on the architectural mesh. In addition, recent studies have focused on architectural component modeling and objects removal in
images. However, the object removal in an image might cause a discordance of colors between multiple images. Recently, there are attempts to use learning-based methods for IBR. Hedman et al.\(^\text{16}\) use deep learning approach for blending novel views for IBR. It showed feasibility of using deep learning-based blending but needs further research to reduce blurriness and flicker. Thies et al.\(^\text{17}\) use a novel deep learning-based method for image synthesis and real-time rendering. However, the method needs training for every new object requiring more research for the generalization.

**Multiview inpainting**

To achieve color consistency through multiview inpainting, other studies\(^\text{3,4}\) warp multiple images onto a common rectified target image plane and optimize the objective function or process traditional patch-based algorithms. Li et al.\(^\text{18}\) apply RGB-D sensor, a device that augments the image with depth information, and employ exemplar-based image inpainting with sequential data. However, RGB-D sensors are inappropriate for large-scale indoor scanning or rendering owing to a short range of depth and an extremely large number of required images. Other approaches such as video inpainting\(^\text{19}\) for recovering consistent color using dense sequential data, however, are not applicable on sparse data or wide-baseline data used for IBR.\(^\text{1,9,3}\)

**Single-image inpainting**

Traditional inpainting algorithms\(^\text{5,20}\) use patch based or diffusion-based approaches with low-level features. Recent studies have focused on rapidly advanced deep learning-based approaches, such as a convolutional neural networks and generative adversarial networks (GANs),\(^\text{21}\) which are capable of using high-level features. Yu et al.\(^\text{8}\) employed a GAN and proposed a coarse-to-fine network as well as a contextual attention module (CAM). CAM attends to known patches of features directly, and explicitly borrows patches to fill in unknown regions of features. It has inspired numerous recent studies\(^\text{8,22}\) making use of CAM for image inpainting. Yu et al.\(^\text{8}\) proposed a gated convolution that enables a sketch channel input to propagate through a network along with semantic information.

An increasing number of recent studies\(^\text{7,23,8}\) have applied edge information (i.e. guideline) as an additional input for the purpose of enhancing the quality or achieving editable results. However, no studies have focused on combining the inputs of color and imperfect guidelines and a guideline refinement at the same time.

**Method**

In this section, we introduce the initial data and a multiview inpainting pipeline, which is shown in Figure 2, including a method for creating the color prior images and an image inpainting network. The color prior image is used as the additional input for the image inpainting network.

**Initial data**

Initial data consist of architectural mesh, images, and camera poses of each image of large-scale indoor spaces that were introduced in recent works.\(^\text{1,9}\) The data sets were acquired by scanning the indoor spaces employing a mobile robot system (cf. Figure 2). Global guidelines are obtained by automatic algorithm,\(^\text{9}\) which are overlaid with user guidelines. Masks were made manually for the purpose of thorough multiview inpainting.

**Color prior generation**

A color prior image is a synthetic image created by projecting a colored point cloud data (PCD) onto the image plane. We introduce a method for coloring the PCD with only architectural component colors and projecting the colored PCD onto the images. The PCD is initially sampled from the object-free architectural mesh. From the sampled PCD, we adopt the hidden point removal (HPR) algorithm,\(^\text{24}\) which is a simple and fast algorithm that approximates the visibility of the PCD for a given camera pose by flipping the point clouds and computing a convex hull onto it. To be more specific, for
each given camera position, \(c_j \in \mathbb{R}^3\), the HPR algorithm yields a binary vector \(v_j \in \{0, 1\}^{m \times 1}\) indicating the visibility of the point clouds. We make the relation matrix \(P^v \in \mathbb{R}^{m \times (3 + n)}\) as

\[
P^v = [P' V]
\]

where \(P \in \mathbb{R}^{m \times 3}\) is the 3D position of the point clouds, \(V = [v_1, v_2, \ldots, v_n]\), \(m\) is the number of points, and \(n\) is the number of camera poses.

The next step is coloring the PCD with architectural colors and creating color prior images. For each point, we make candidates of the proper color by projecting the point onto each image plane where the point is visible. We can refer to a row of the relation matrix, \(P'_i\), to find images in which the point is visible. If the point in a visible image plane corresponds to a nonmasked region (i.e. architectural component), the color becomes a candidate for the proper color, as shown in Figure 3(a). Among the accumulated candidates of color for a point, we simply take their median to determine the color of the point and achieve a robustness to the outliers and an efficient computation.

\[
P_{proj} = \frac{\mathcal{R} w^T \mathbf{P} + c^T \mathbf{T}}{\max(|\mathcal{R} w^T \mathbf{P} + c^T \mathbf{T}|)} = \begin{bmatrix} u \\ v \\ w \end{bmatrix}
\]

where \(\mathcal{R}_w\) and \(c^T\) are the rotation matrix and translation vector from the camera with respect to the world coordinates, respectively. In addition, \(\mathbf{P}\) is the point with respect to (w.r.t) world coordinates, and \(s_{img}\) is the image size.

The color prior image, however, depends on the sampling rate and the distance to a point from the camera pose. To be more precise, the sparse PCD originates from minimum sampling distance of \(r\) from a mesh. In addition, the points closer to an image plane appear even sparser in the image than those in farther (cf. Figure 4(a)).

To resolve the dependencies, we define point coverage area, \(C_{proj}\), in the image plane as shown in Figure 3(b). The distance between two points along the diagonal will be \(\sqrt{3}r\) in 3D space. Hence, we set the coverage radius of a point to \(\frac{3}{2}r\) allowing some overlap. The coverage space, \(C_{init}\), of a point becomes a sphere

\[
C_{init} = \left(\frac{\sqrt{3}}{2}r\right)^2 I_3
\]

where \(I_3\) refers to \(3 \times 3\) identity matrix. The Jacobian of \(P_{proj}\) w.r.t. each coordinate of a point is

\[
J = \frac{\partial p_{proj}}{\partial x} = \begin{bmatrix} \frac{\partial u}{\partial x} & \frac{\partial u}{\partial y} & \frac{\partial u}{\partial z} \\
\frac{\partial v}{\partial x} & \frac{\partial v}{\partial y} & \frac{\partial v}{\partial z} \\
\frac{\partial w}{\partial x} & \frac{\partial w}{\partial y} & \frac{\partial w}{\partial z} \end{bmatrix}
\]

\[\text{Figure 3. Creating the color prior image: (a) The color for a given point is computed from a set of images that are not occluded by objects; (b) a colored point with the coverage space is projected onto the image plane to create the color prior image.}\]

\[\text{Figure 4. Example of created color prior: (a) The colored point closer to the camera appears sparser than that farther away; (b) we resolve the sparsity issue by defining the coverage area; (c) the original image from the given viewpoint.}\]
Typical image inpainting networks\(^8,7\) take a masked image and a guideline image as inputs and output an inpainting result. By contrast, to preserve color consistency over multiple images, we train a network that is capable of receiving the color prior information as an additional input. In addition, the network is able to perform two different tasks concurrently such as the image inpainting and guideline refinement. In the following sections, we introduce the input data we generated to train the network, architecture, and loss functions to achieve our goal: the color consistent and guideline refined result. By contrast, to preserve color consistency over multiple images, we train a network that is capable of receiving the color prior information as an additional input. In addition, the network is able to perform two different tasks concurrently such as the image inpainting and guideline refinement. In the following sections, we introduce the input data we generated to train the network, architecture, and loss functions to achieve our goal: the color consistent inpainting and guideline refinement.

### Image inpainting

Typical image inpainting networks\(^8,7\) take a masked image and a guideline image as inputs and output an inpainting result. By contrast, to preserve color consistency over multiple images, we train a network that is capable of receiving the color prior information as an additional input. In addition, the network is able to perform two different tasks concurrently such as the image inpainting and guideline refinement. In the following sections, we introduce the input data we generated to train the network, architecture, and loss functions to achieve our goal: the color consistent inpainting and guideline refinement.

**Input data generation for training.** Given the ground truth image \(I_{gt}\) and guideline \(L_{gt}\), we need to generate training input data including the color prior \(I_{cp}\), imperfect guideline \(\hat{L}\), mask \(M\), and noncolor mask \(M_{cp}\) as in Figure 5. The ground truth guideline \(L_{gt}\) is generated by the pre-trained edge detector named as holistically nested edge detection,\(^25\) with a threshold of 0.6.

For the generation of color prior input \(I_{cp}\) and imperfect guideline input \(\hat{L}\), we tried to reproduce the data similar to a real case. The pixels in the color prior input \(I_{cp}\) are randomly sampled from two different randomly affine transformed and jittered ground truth images followed by Gaussian blur and random erase. The transformations are used to imitate various noises or errors that the color prior image of the real indoor data sets often contains such as sparsity, camera pose error, blurriness, occlusions by objects, and so on.

An imperfect guideline \(\hat{L}\) is generated through random affine transformed patches of the ground truth guideline \(L_{gt}\) and is then randomly erased. These transformations are used to imitate angle, location, or missing errors that can be often made by users.

The mask \(M\) is generated by a random free-form mask generation algorithm.\(^8\) A noncolor mask \(M_{cp}\) is another binary mask representing whether the color prior information exists in the mask

\[
M_{cp} = \begin{cases} 
0, & \text{where } I_{cp} + (1 - M) > 0 \\
1, & \text{otherwise} 
\end{cases}
\]  

No-guideline zone is another binary mask used for a loss function. The usage of no-guideline zone is explained later in detail. The codes of generating the input data for the training are also available at the link.\(^10\)

**Network architecture.** We modify the coarse-to-fine network used in GConv\(^8\) based on the key ideas from the recent work\(^22,26\) as shown in Figure 6. The coarse network receives an input tensor \(X\) with a size of \(9 \times H \times W\), where \(H\) and \(W\) are height and width of the input image, respectively. The nine input channels consist of a masked image \(\hat{I}\), mask \(M\), imperfect guideline \(L\), color prior \(I_{cp}\), and noncolor mask \(M_{cp}\).

The size of the output tensor of the coarse network is \(4 \times H/2 \times W/2\), where four channels consist of a coarse image output and coarse guideline correction. The refine network then receives the output of the coarse network and returns a tensor with a size of \(4 \times H \times W\), in which the four channels consist of a refined image output \(I_{pred}\) and refined guideline \(L_{pred}\).

The discriminator is similar to the SN-patchGAN structure in GConv\(^8\) except that instead of having a stride 1 convolution at the very first layer, our discriminator starts with a stride 2 convolution directly and uses Leaky ReLU as the activation function. We also adopt a spectral spin...
normalization technique recently proposed by Miyato et al.\textsuperscript{27} The discriminator takes eight channels of input consisting of the image, guideline, mask, and color prior.

**Loss functions.** The coarse-to-fine network is trained in an end-to-end manner over several joint losses consisting of an adversarial loss, $L_1$ loss, perceptual loss, style loss, focal loss, and two more proposing losses including anti-specificity loss and $L_{cp}$ loss.

Let $G$ and $D$ be the generator and discriminator, respectively. We use the hinge loss as the objective function

$$L_D = \mathbb{E}\left(\text{ReLU}(1 - D(X_{gt}))\right) + \mathbb{E}\left(\text{ReLU}(1 + D(Y_{gen}))\right)$$

(8)

$$L_{GAN} = -\mathbb{E}(D(G(X), M, I_{cp}))$$

(9)

where $X_{gt}$ consists of the ground truth image $I_{gt}$ and guideline $L_{gt}$, mask $M$, and color prior $I_{cp}$, whereas $Y_{gen}$ consists of the output image and guideline from generator, as well as the mask and color of input data. The $L_1$ loss is computed as follows

$$L_{l_1}(I_{gt}, I_{pred}, M) = \frac{\| (I_{gt} - I_{pred}) \odot M \|_1}{n_M}$$

$$+ \frac{\| (I_{gt} - I_{pred}) \odot (1 - M) \|_1}{n_{1-M}}$$

(10)

where $n_M$ is the number of elements in the binary mask $M$, $I_{gt}$ is the ground truth image, $I_{pred}$ is the output image of the generator, and $\odot$ is the Hadamard product. We normalize each term based on the mask size for proper scaling.

We also include the perceptual loss $L_{perc}$ and style loss $L_{style}$ proposed in works.\textsuperscript{28,29} We observed that the perceptual loss and the style loss help enhancing image details and reducing artifacts as addressed in the work.\textsuperscript{30} Here, $L_{perc}$ penalizes the distance between two feature maps of a pre-trained network forcing the generator to make perceptually similar output to the ground truth image. This is computed as follows

$$L_{perc} = \mathbb{E}\left[ \sum_{i} \frac{1}{N_i} \| \phi_i(I_{gt}) - \phi_i(I_{pred}) \|_1 \right]$$

(11)

where $\phi_i$ is the feature map of $i$th layer of the pre-trained VGG-16 with batch normalization applied on the ImageNet data set.\textsuperscript{31} We use layers of pool1, pool3, pool4, and pool5. These layers are also used to compute style loss, which penalizes the difference between covariances of the feature maps. Here, $L_{style}$ is computed as follows

$$L_{style} = \sum_{i} \mathbb{E}_{I_{gt}} \left[ \frac{\| G^{\phi}_i(I_{gt}) - G^{\phi}_i(I_{pred}) \|_1}{N_{HW_i}} \right]$$

(12)

where $G^{\phi}_i$ is a $C_i \times C_i$ gram matrix made from each feature map $\phi_i$ of shape $C_i \times H_i \times W_i$ and $N_{HW_i} = H_i \times W_i$.

The guideline refinement is trained using the focal loss $L_{foc}$ proposed by Lin et al.\textsuperscript{32} It regularizes the network not only to restore the ground truth guideline from randomly distorted and erased input guideline but also to sharpen the image edges in the inpainted image. It is computed as follows

$$L_{foc} = -\mathbb{E} \left[ (I_{gt} - I_{pred})^2 \odot H(I_{gt}, I_{pred}) \right]$$

(13)

where $H$ is the binary cross entropy.

As can be seen in Figure 7, we found that the black region in the color prior affects the prediction of the guideline and results in black holes. This is because the network regards a black region in the color prior as the proper color and predicts a borderline around the region. However, a black region is where no prior color information exists and should be filled in appropriately.

We resolve this problem by proposing two new losses, anti-specificity loss $L_{as}$ and another $l_1$ loss for the color prior $L_{l_1, cp}$. Here, $L_{as}$ penalizes false-positive predicted guideline that belongs to the no-guideline zone, $N$ (as in Figure 5)

$$N = (1 - L_{gt}) \odot \text{Dilate}(\text{Canny}(M_{gt}))$$

(14)

It is where guideline prediction should be negative that belongs to the negative ground truth guideline $1 - L_{gt}$.
overlapping with thick edges surrounding the black region $\text{M}_{cp}$, $\text{Dilate}(\text{Canny}(\text{M}_{cp}))$. The anti-specificity loss $L_{as}$ is computed as follows

$$L_{as}(L_{pred}, N) = \frac{\text{FP}}{\text{TN} + \text{FP}} = \frac{\mathbb{E}[L_{pred} \cap N]}{\mathbb{E}[N] + \varepsilon} \quad (15)$$

where $\text{FP} = \mathbb{E}[L_{pred} \cap N]$ is the mean of the false-positive guideline prediction, $\text{TN} = \mathbb{E}[1 - (1 - L_{pred}) \cap N]$ is mean of true-negative guideline prediction, and $\varepsilon$ is a small value to avoid dividing by zero. The code of $L_{as}$ is also available at the link.\textsuperscript{10} $L_{i, cp}$ is the same as in equation (10) except for the mask $\text{M}_{cp}$.

An adversarial loss, a perceptual loss, and a style loss are used to regularize the entire generator, whereas the others are additionally used to regularize both the coarse network and the entire generator. The overall loss function for our generative network $L_{G}$ is defined as follows

$$L_{G} = L_{color} + L_{guideline} \quad (16)$$

$$L_{color} = \lambda_g L_{GAN} + \lambda_s L_{style} + \lambda_p L_{perc} + \lambda_{l_{i}} L_{l_{i}} \quad (17)$$

$$L_{guideline} = \lambda_{l_{i, cp}} L_{l_{i, cp}} + \lambda_f L_{f_{oc}} + \lambda_{as} L_{as} \quad (18)$$

In short, $L_{color}$ is a typical loss similar to ones used in other works.\textsuperscript{8,7} By contrast, $L_{guideline}$ is introduced to resolve the black hole problem and to perform two different tasks in a single model concurrently which are the inpainting task, $L_{color}$, and the guideline refinement task, $L_{f_{oc}}$. We choose $\lambda_g = 0.1$, $\lambda_s = 200$, $\lambda_p = 0.08$, $\lambda_{l_{i}} = 0.5$, $\lambda_{l_{i, cp}} = 0.7$, $\lambda_f = 1$, and $\lambda_{as} = 0.1$.

**Experiments and results**

We evaluate our multiview inpainting pipeline on real indoor spaces introduced in recent works.\textsuperscript{1,9} The properties of each space are summarized in Table 1. The degree of light difference, occlusions, or environmental complexity of a space is indicated qualitatively by “+” sign, where more “+” signs represent harder conditions for multiview inpainting (Experiencing real-time rendering using our multiview inpainting results is available at www.teevr.net:2019/generative-mvi.\textsuperscript{33})

Among the different methods, the best performance of $\text{avg}(d_i)$ and $\text{avg}(d_s)$ are indicated in boldface.

![Figure 7](image)

**Table 1.** Properties, statistics, and results on real data sets.

| Data set | Space 1 | Space 2 | Space 3 |
|----------|---------|---------|---------|
| Size ($m^2$) | 80 | 1930 | 2804 |
| Camera poses | 5 | 29 | 59 |
| Num. images | 30 | 174 | 354 |
| Light difference | + | + | + |
| Occlusions | ++ | + | ++ |
| Env. complexity | ++ | ++ | ++ |
| r (cm) | 0.75 | 1.25 | 2.75 |
| Evaluation points | 253,798 | 844,441 | 501,873 |
| $\text{avg}(n)$ | 2.65 | 3.25 | 4.67 |
| $\text{avg}(m)$ | 2.16 | 5.92 | 8.52 |
| EC - $\text{avg}(d_i)$ | 0.0140 | 0.0238 | 0.0236 |
| GConv - $\text{avg}(d_i)$ | 0.0108 | 0.0222 | 0.0200 |
| OURs w/o cp - $\text{avg}(d_s)$ | 0.0099 | 0.0209 | 0.0171 |
| OURs w/ cp - $\text{avg}(d_s)$ | 0.0075 | 0.0184 | 0.0166 |
| EC - $\text{avg}(d_s)$ | 0.0170 | 0.0262 | 0.0269 |
| GConv - $\text{avg}(d_s)$ | 0.0152 | 0.0266 | 0.0233 |
| OURs w/o cp - $\text{avg}(d_s)$ | 0.0145 | 0.0249 | 0.0210 |
| OURs w/ cp - $\text{avg}(d_s)$ | 0.0110 | 0.0226 | 0.0200 |

For comparison, we selected recent two state-of-the-art papers that utilize the edge information, EC\textsuperscript{7} and GConv.\textsuperscript{8} EC consists of two separate generative networks, one predicting the image edge and the other completing the inpainting with the use of the edge prediction. By contrast, GConv uses a user-sketch input for the image inpainting but does not predict the edge. Our model uses imperfect guideline as well as the color prior as input and refines the guideline and completes the image inpainting concurrently. We used $512 \times 512$ images of the Places2\textsuperscript{34} data set for training our model, which is the same resolution as the indoor data sets. EC was pre-trained over the Places2 data set using $256 \times 256$ images.
Color evaluation

Before beginning color evaluation for multiview inpainting, pre-processing is necessary to alleviate luminance dependency due to a lot of light sources affecting differently on color intensity w.r.t. camera poses. To resolve the problem, all images are normalized for conversion into the intensity-free \( rg \) chromaticity space as

\[
\begin{align*}
    r &= \frac{R}{R + G + B}, \\
    g &= \frac{G}{R + G + B}
\end{align*}
\]

where \( r \) and \( g \) are bounded between 0 and 1.

Following data arrangement is also processed for pixel-wise evaluation. For a given 3D point \( p_i \) from the sampled PCD, two sets of images are defined, \( A_i \) and \( B_i \). \( A_i \) is a set of images that \( p_i \) is occluded by object while \( B_i \) is a set of images that the point is directly visible (i.e. sets of inpainted images vs. architectural component visible images). Then pixel-wise \( rg \) chromaticity matrices, \( A_i \) and \( B_i \), can be constructed by utilizing \( p^v \) and \( p_{proj} \) in equation (1) and (2) as

\[
\begin{align*}
    A_i &= \begin{bmatrix} r_1 & g_1 \\ \vdots & \vdots \\ r_n & g_n \end{bmatrix}, \\
    B_i &= \begin{bmatrix} r'_1 & g'_1 \\ \vdots & \vdots \\ r'_m & g'_m \end{bmatrix}
\end{align*}
\]

where \( r_j \) and \( g_j \) are the \( rg \) chromaticity of the pixel corresponding to the point \( p_j \) in \( j \)th inpainted image belonging to the set \( A_i \), and similarly \( r'_k \) and \( g'_k \) are the \( rg \) chromaticity of that in \( k \)th visible image belonging to the set \( B_i \). In addition, \( n_i \) and \( m_i \) refer to the number of elements in the set \( A_i \) and \( B_i \), respectively, where \( n_i \) is larger than one and \( m_i \) is larger than zero. The number of point clouds to
evaluate that satisfy the conditions are subset of the overall sampled PCD, which is shown in 7th row of Table 1. The average of the number of elements $n_i$ and $m_i$ over the points set is presented in the following rows of the Table 1.

For the evaluation, we compare two factors, consistency over $n_i$ images ($d_c$) and similarity between the sets $A_i$ and $B_i$ ($d_s$) as visualized in Figure 8. First, consistency distance $d_c$ is computed as follows

$$d_c = \sqrt{\max_{i} \left( \sigma \left( \frac{1}{n_i - 1} \left( \bar{A}_i^T \bar{A}_i \right) \right) \right)}$$

(22)

where $\bar{A}_i$ is $A_i$ mean over $n_i$-elements and $\sigma$ refers to singular values. We decide consistency distance as the square root of the largest singular value of sample covariance of $A_i$.

**Figure 11.** A sample result of missing guidelines. Line patterns on the left side pillar in the input guideline image are not drawn but are visible in the refined guideline output. For visualization, the output guideline corresponding to the mask is colored in blue and emphasized.

**Figure 12.** Quantitative comparisons of consistency distance for different models: (a), (b), and (c) are the results of the Space 1, 2, and 3, respectively. The dashed vertical line represents the average value of $n_i$'s of the given space. Our approach shows lower value on every $n$ compared to the other methods. It means our approach outperforms the others on the perspective of consistency distance regardless of the number of inpainted images of the multiview inpainting task.

**Figure 13.** Quantitative comparisons of similarity distance for different models: (a), (b), and (c) are the results of the Space 1, 2, and 3, respectively. The dashed vertical line represents the average value of $n_i$'s of the given space. Our approach shows lower value on every $n$ compared to the other methods. It means our approach outperforms the others on the perspective of similarity distance regardless of the number of inpainted images of the multiview inpainting task.
Second, similarity distance $d_s$ is the Euclidean distance between the means of the matrices $A_i$ and $B_i$, which is computed as

$$d_s = ||\bar{A}_i - \bar{B}_i||_2 \quad (23)$$

Finally, since $d_c$ and $d_s$ are point-wise measures, they are averaged over all points. Evaluations were conducted on EC, GConv, OURs in the absence of the color prior to verify the effects of the color prior, and OURs using the color prior. The results are presented in the Table 1.

**Figure 14.** Qualitative comparisons on real data sets. Each pair of rows is Space 1, 2, and 3 of the real data set, respectively.
Improvements of up to 46.4% and 30.6% in the consistency distance and also up to 35.6% and 27.3% in the similarity distance were achieved compared to EC and GConv, respectively. The qualitative comparisons are shown in Figure 9.

Boundary evaluation
We evaluate boundary consistency qualitatively. As can be seen in Figure 10, We observed that EC is good at finding finer edges because it was trained using a Canny edge detector followed by nonmaximum suppression. However, it often fails to generate proper edge on the area where the architectural boundary should exist in a real data set (cf. Figure 10(b) and (c)). By contrast, both GConv and our approach achieve better results in terms of the architectural boundary because of the explicit guideline inputs. Furthermore, with the approach of guideline refinement in the network, Figure 11 shows that our model is able to find the missing guidelines in the images.

Ablation studies
For the quantitative evaluation of multiview inpainting on the real indoor data sets,1,9 we defined the point-wise consistency metrics, consistency distance and similarity distance, which are averaged over all points. However, averaging over the number of all points might blur the point of multiview inpainting, which is color-consistent inpainting regardless of the number of inpainted images.

This time, we further analyze the impact of the number of inpainted images on the color consistency. For this, each point, \( p_i \), is classified according to the number of times, \( n_i \), that it is used for inpainting. Then, the consistency distance and similarity distance of each classified group of points are averaged over the number of the group’s points. In specific, \( n_i \) is the number of rows in the matrix \( A_i \) in equation 20. Finally, we have the average values of each metric according to the number of inpainted images and make plots of them as Figures 12 and 13.

Figure 12 shows the impact of \( n_i \) on consistency distance. Consistency distance tends to increase as the number \( n_i \) increases. Our approach consistently shows lower value on every \( n_i \) compared to other single-image inpainting methods. Likewise, Figure 13 also shows similar results and that our approach outperforms other methods on every \( n_i \) in similarity distance. Both results suggest that our pipeline using the color prior is more effective on the multiview inpainting task regardless of the number of inpainted images compared to the single-image inpainting approaches. Figure 14 shows more qualitative comparisons on the real data sets.

Conclusions
We proposed a multiview inpainting pipeline, that is, creating color prior images and an image inpainting network. Our pipeline aims at achieving both color and boundary consistency for multiview inpainting by introducing a color prior and guideline input, respectively. While training the network, we introduced new losses to resolve the black hole problem and achieve two goals, inpainting and guideline refinement, concurrently. We evaluated the results of our pipeline using a consistency distance and a similarity distance, which indicate that the method outperforms other methods applied to multiview inpainting.

We plan to extend the proposed multiview image inpainting with a better pipeline to further stabilize the light effects during the color prior generation step. There are also other promising applications and future works using our pipeline. Since the inpainting is affected by the color prior images, user-interactive inpainting can be possible. Users may remodel their virtual spaces as the inpainting can be modified by user input. Similarly, our method can be used for image synthesis in indoor or outdoor of target space if the point clouds and poses of images are given. Also, our pipeline may be used as a preprocessing and extended to recent deep learning-based rendering approaches.

Declaration of conflicting interests
The author(s) declared the following potential conflicts of interest with respect to the research, authorship, and/or publication of this article: Korea University and TeeLabs Inc.

Funding
The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This research was supported by the Technology Innovation Program (10073166) funded By the Ministry of Trade, Industry & Energy (MOTIE, Korea)

ORCID iDs
Joohyung Kim https://orcid.org/0000-0003-4641-1406
Janghun Hyeon https://orcid.org/0000-0001-5737-9179

References
1. Doh N, Choi H, Jang B, et al. Teevr: spatial template-based acquisition, modeling, and rendering of large-scale indoor spaces. In: ACM SIGGRAPH Emerging Technologies (ed M Rose), Los Angeles, CA, 28 July to 1 August 2019, pp. 1–1. New York: ACM.
2. Hedman P, Ritschel T, Drettakis G, et al. Scalable inside-out image-based rendering. ACM Trans Graphic 2016; 35(6): 1–11.
3. Philip J and Drettakis G. Plane-based multi-view inpainting for image-based rendering in large scenes. In: Proceedings of the ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games (ed RC Anthony), Quebec, Canada, 15–18 May 2018, pp. 1–11. New York: ACM.
4. Thonat T, Shechtman E, Paris S, et al. Multi-view inpainting for image-based scene editing and rendering. In: 2016 Fourth International Conference on 3D Vision (ed S Savarese), Stanford University, CA, 25–28 October 2016, pp. 351–359. New York, USA: IEEE.
5. Barnes C, Shechtman E, Finkelstein A, et al. Patchmatch: a randomized correspondence algorithm for structural image editing. *ACM Trans Graphic* 2009; 28(3): 24.

6. Yu J, Lin Z, Yang J, et al. Generative image inpainting with contextual attention. In: *Proceedings of the IEEE conference on computer vision and pattern recognition* (eds M Brown, B Morse and S Peleg), Salt Lake City, UT, 18–22 June 2018, pp. 5505–5514. New York, USA: IEEE.

7. Nazeri K, Ng E, Joseph T, et al. Edgeconnect: structure guided image inpainting using edge prediction. In: *Proceedings of the IEEE International Conference on Computer Vision Workshops* (eds K Mu Lee, D Forsyth, M Pollefeys, et al.), Seoul, Korea, 27 October to 2 November 2019, pp. 1–9. New York, USA: IEEE.

8. Yu J, Lin Z, Yang J, et al. Free-form image inpainting with gated convolution. In: *Proceedings of the IEEE International Conference on Computer Vision* (eds KM Lee, D Forsyth, M Pollefeys, et al.), Seoul, Korea, 27 October to 2 November 2019, pp. 4471–4480. New York, USA: IEEE.

9. Hyeon J, Choi H, Kim J, et al. Automatic spatial template generation for realistic 3D modeling of large-scale indoor spaces. In: *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems* (ed D Sun), Macau, China, 4–8 November 2019, pp. 4221–4228. New York, USA: IEEE.

10. Available at: https://github.com/kimjh069/generative-MVI (accessed 8 February 2020).

11. Henry P, Krainin M, Herbst E, et al. RGB-D mapping: using Kinect-style depth cameras for dense 3D modeling of indoor environments. *Int J Rob Res* 2012; 31(5): 647–663.

12. Shao T, Xu W, Zhou K, et al. An interactive approach to semantic modeling of indoor scenes with an RGBD camera. *ACM Trans Graphic* 2012; 31(6): 1–11.

13. Ikehata S, Yang H, and Furukawa Y. Structured indoor modeling. In: *Proceedings of the IEEE International Conference on Computer Vision* (eds R Bajcsy, G Hager and Yi Ma), Santiago, Chile, 11–18 December 2015, pp. 1323–1331. New York, USA: IEEE.

14. Lim G, Oh Y, Kim D, et al. Modeling of architectural components for large-scale indoor spaces from point cloud measurements. *IEEE Rob Auton Lett* 2020; 5(3): 3830–3837.

15. Turner E, Cheng P, and Zakhor A. Fast, automated, scalable generation of textured 3D models of indoor environments. *IEEE J Sel Top Signal Process* 2014; 9(3): 409–421.

16. Hedman P, Philip J, Price T, et al. Deep blending for free-viewpoint image-based rendering. *ACM Trans Graphic* 2018; 37(6): 1–15.

17. Thies J, Zollhöfer M, and Nießner M. Deferred neural rendering: image synthesis using neural textures. *ACM Trans Graphic* 2019; 38(4): 1–12.

18. Li F, Ricardez GAG, Takamatsu J, et al. Multi-view inpainting for RGB-D sequence. In: *2018 International conference on 3D Vision* (ed A Fusiello), Verona, Italy, 5–8 September 2018, pp. 464–473. New York, USA: IEEE.

19. Kim D, Woo S, Lee JY, et al. Deep video inpainting. In: *Proceedings of the IEEE conference on computer vision and pattern recognition* (eds L Davis, P Torr and SC Zhu), Long Beach, CA, 16–20 June 2019, pp. 5792–5801. New York, USA: IEEE.

20. Bertalmio M, Sapiro G, Caselles V, et al. Image inpainting. In: *Proceedings of the 27th annual conference on computer graphics and interactive techniques* (eds JR Brown and K Akeley). New Orleans, LA, 23–28 July 2000, pp. 417–424. New York: ACM.

21. Goodfellow I, Pouget-Abadie J, Mirza M, et al. Generative adversarial nets. In: *Advances in neural information processing systems* (eds Z Ghahramani and M Welling), Montreal, Canada, 8–13 December 2014, pp. 2672–2680. Cambridge, MA, USA: MIT Press.

22. Zeng Y, Fu J, Chao H, et al. Learning pyramid-context encoder network for high-quality image inpainting. In: *Proceedings of the IEEE conference on computer vision and pattern recognition* (eds L Davis, P Torr and SC Zhu), Long Beach, CA, 16–20 June 2019, pp. 1486–1494. New York, USA: IEEE.

23. Xiong W, Yu J, Lin Z, et al. Foreground-aware image inpainting. In: *Proceedings of the IEEE conference on computer vision and pattern recognition* (eds P Torr Davis and SC Zhu), Long Beach, CA, 16–20 June 2019, pp. 5840–5848. New York, USA: IEEE.

24. Katz S, Tal A, and Basri R. Direct visibility of point sets. *ACM Trans Graphic* 2007; 26(3): 24–es.

25. Xie S and Tu Z. Holistically-nested edge detection. In: *Proceedings of the IEEE international conference on computer vision* (eds R Bajcsy, G Hager and Y Ma), Santiago, Chile, 11–18 December 2015, pp. 1395–1403. New York, USA: IEEE.

26. Zhang H, Goodfellow I, Metaxas D, et al. Self-attention generative adversarial networks. In: *International conference on machine learning* (ed E Xing), Long Beach, CA, 9–15 June 2019, pp. 7354–7363. New York, USA: IMLS.

27. Miyato T, Kataoka T, Koyama M, et al. Spectral normalization for generative adversarial networks. arXiv preprint arXiv:1802.05957. 2018.

28. Gatys LA, Ecker AS, and Bethge M. Image style transfer using convolutional neural networks. In: *Proceedings of the IEEE conference on computer vision and pattern recognition* (eds R Bajcsy, FF Li and T Tuytelaars), Las Vegas, NV, USA, 26 June to 1 July 2016, pp. 2414–2423. New York, USA: IEEE.

29. Johnson J, Alahi A, and Fei-Fei L. Perceptual losses for real-time style transfer and super-resolution. In: *European conference on computer vision* (eds T Gevers and A Smeulders), Amsterdam, The Netherlands, 8–16 October 2016, pp. 694–711. Germany: Springer.

30. Sajjadi MS, Scholkopf B, and Hirsch M. Enhancenet: single image super-resolution through automated texture synthesis. In: *Proceedings of the IEEE international conference on computer vision*, Venice, Italy, 22–29 October 2017, pp. 4491–4500. New York, USA: IEEE.

31. Russakovsky O, Deng J, Su H, et al. Imagenet large scale visual recognition challenge. *Int J Comput Vision* 2015; 115(3): 211–252.

32. Lin TY, Goyal P, Girshick R, et al. Focal loss for dense object detection. In: *Proceedings of the IEEE international
conference on computer vision (eds K Ikeuchi, G Medioni and M Pelillo), Venice, Italy, 22–29 October 2017, pp. 2980–2988. New York, USA: IEEE.

33. Available at: www.teevr.net:2019/generative-mvi (accessed 21 August 2020).

34. Zhou B, Lapedriza A, Khosla A, et al. Places: a 10 million image database for scene recognition. *IEEE Trans Pattern Anal Mach Intell* 2017; 40(6): 1452–1464.

35. Hunt R. *The reproduction of colour*. 6th ed. Hoboken: John Wiley, 2004.