Image-Adaptive Hint Generation via Vision Transformer for Outpainting

Daehyeon Kong*[1], Kyeongbo Kong*[2], Kyunghun Kim*[1], Sung-Jun Min*[1], and Suk-Ju Kang[1]
[1]Sogang University, {kdh4672, godgang, sjmin9868, sjkang}@sogang.ac.kr
[2]Pukyong National University, kbkong@pknu.ac.kr

Abstract

Image outpainting has recently received considerable attention because it can be useful in tasks such as image retargeting and panorama image generation. In general, the problem of extending an image beyond its given boundaries is still ill-posed. Conventional methods predominantly attempt image outpainting by using complex network structures. Some recent studies have tried to decrease the problem complexity through the conversion techniques from outpainting to inpainting. Although these methodologies work well in simple cases, their performance reduces considerably for asymmetrical images. This paper proposes a novel hint-based outpainting methodology that can adaptively select the most plausible patches as hints from a given image to reduce the difficulty of outpainting. To estimate high-quality hints, inspired by patch-based image inpainting methods, we utilize Vision Transformer that considers self-attention for each patch. The estimated hints are attached on both boundaries of the input image and the inside missing regions are predicted by using an inpainting network. After finishing the prediction, the output image is obtained by removing the hints. Experiments show that our image-adaptive hint framework, when employed in representative inpainting networks, can consistently improve its performance compared to the other conversion techniques from outpainting to inpainting on SUN and Beach benchmark datasets.

1. Introduction

Can you easily imagine what the outside of an image looks like? For example, given an image of the sea, we can imagine the areas surrounding the beach or the waves by considering the connectivity and the content of the image. In the image completion field, image outpainting involves drawing the outer area of a given image. It can enable various content creation applications such as image editing, panorama image generation, 3D game graphics, and virtual reality. In addition, image outpainting is also used in image retargeting [3, 4], which resizes the image to fit various display aspect ratios.

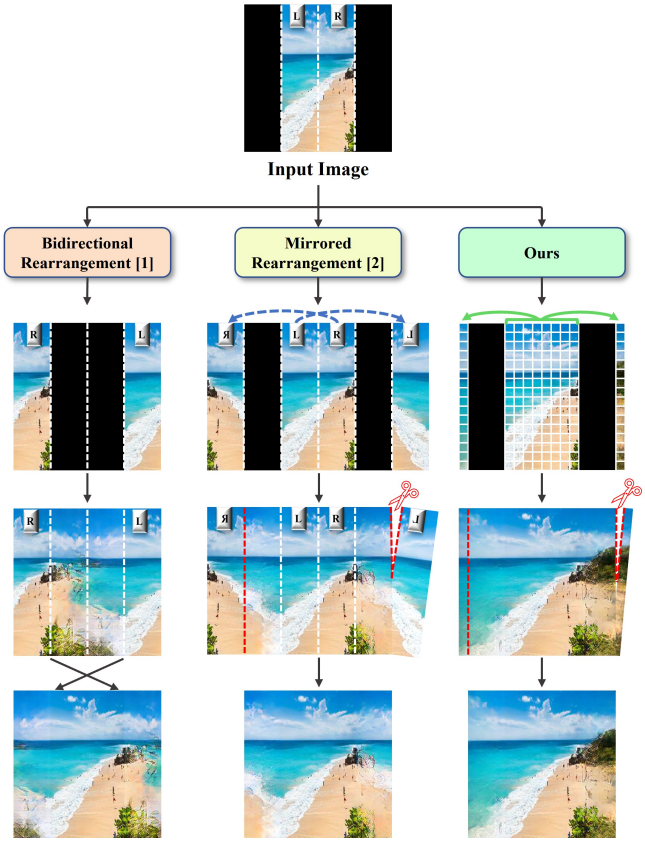

Figure 1. Methodology comparison for conversion techniques from outpainting to inpainting. Input image is referred as an image with missing regions in both left and right sides of ground-truth image. Bidirectional Rearrangement (BR) [1] and Mirrored Rearrangement (MR) [2] predict missing regions by using left and right swapped input image and mirror flipped input image, respectively. The proposed method generates the image-adaptive hints based on a Vision Transformer, and attaches the hints on both boundaries of the input image to predict a high-quality image structurally.

Generally, image outpainting and image inpainting are similar in that each generates unknown areas. However,

*equal contribution

since image inpainting can utilize the surrounding context [5], it is considered to be much easier than the outpainting task. Recent studies have tried to solve the problem by converting the outpainting problem into an image inpainting problem. In Fig. 1, Kim *et al.* [1] first rearranged the input image bidirectionally and predicted the interior of an image via inpainting network and then rearranged the result again. Akimoto *et al.* [2] mirrored the horizontal quarter patches inside the input image and added them to both sides of the image. After predicting missing regions by inpainting network, both redundant sides are eliminated, leaving the output image. However, these methods have limitations because they assume that both ends of the image should be similar or symmetrically matched. For example, the structure of the beach was distorted through the Bidirectional Rearrangement (BR) and Mirrored Rearrangement (MR) methods (Fig. 1). Consequently, they have difficulties with complex scenes.

This paper proposes a hint generation methodology that can expand the image without making such assumptions. The performance of outpainting can be substantially improved if the patches within the image are selected judiciously. Therefore, we adopt a Vision Transformer [6], which has been a spotlight in recent vision studies. The overall process of the proposed hint generation is simple (Fig. 1; rightmost). An input image is divided into patches and passed through the Vision Transformer to generate the hint patches. Then, the hints are attached on both ends of the input image, and the missing regions are predicted by an inpainting network. Finally, the output image can be obtained by removing the hints. Using the hints, we can predict high-quality results with structural completeness and prevent repeated patterns of boundary region. Our contributions are summarized as follows:

- We propose a novel hint-based framework converting from outpainting to inpainting, thereby allowing to apply any type of inpainting models without changing their architectures.
- The proposed hint generation module based on Vision Transformer can produce high-quality hints far from the inside image.
- Our framework consistently improves upon even recent state-of-the-art image completion methods.

2. Related Work

2.1. Image Inpainting

Image inpainting is an image restoration task where the goal is to fill in missing regions within the image while making the entire image visually realistic.

Patch-Based Image Inpainting Method Classical patch-based image inpainting methods [7, 8, 9] regard inpainting as a task of finding the patches for each masked region with manually designed constraints. These methods are inherently limited by the ability of generating novel content. They tend to fail to preserve a reasonable global structure when the masked region is large.

Deep Image Inpainting Method Recently, deep learning-based methods are being studied, which are divided into two categories, single-stage approaches and two-stage approaches. Single-stage approaches [10, 11, 12] adopt an encoder–decoder network with multiple losses to recover the corrupted region directly. These networks are trained to jointly capture the structure and texture information in a single pass. The two-stage approaches reconstruct structural information in the first stage, prior to the second stage for that synthesizes detailed textures [13, 14, 15, 16].

2.2. Image Outpainting

Image outpainting is more challenging than image inpainting because it entails creating new contents rather than filling in partial regions, requiring a more in-depth understanding of scenes.

Patch-Based Image Outpainting Method Classical image outpainting methods are patch-based methods that expand an image by completing the surrounding area of the target image. These methods [17, 18, 19] search the database for images with matching boundaries. As they use real image patches, these methods seem to work well for images with simple and repeating patterns. However, if the database image does not contain a patch that matches the target image, the result does not match the context of the target image. Consequently, when generating a large image, the pixels are merely repeated and the result is not optimal.

Learning-based Image Outpainting Method Sabini *et al.* [20] first attempted to tackle the image outpainting problem using a Generative Adversarial Network (GAN). Since then, several methods [21, 22, 23] have attempted to enhance the output image quality. These methods use a single image as input and apply GAN models to fill in plausible extrapolations. They typically use an encoder–decoder structure and adversarial loss as a starting point. They are trained on diverse datasets.

Outpainting-to-Inpainting Conversion Previous methods using deep learning attempted to solve image outpainting problem as a conditional image-to-image translation, and proposed a new complex network configuration [22, 24]. However, given the lack of information about the adjacent pixels to reference when creating an image, the result is blurry or inconsistent. Recently, some methods [1, 2] attempted to solve the problem by converting the image outpainting problem to an image inpainting problem. These methods do not require a new network for image outpainting; they simply use the latest image inpainting models. As in Fig. 1, Kim *et al.* [1] proposed a bidirectional border region rearrangement method to increase the adjacent infor-

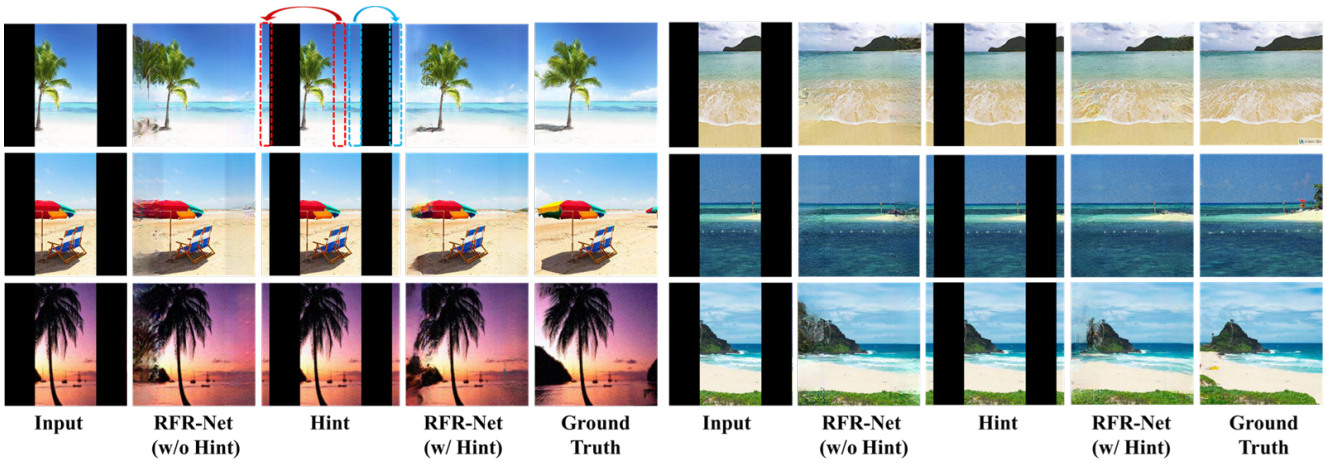

Figure 2. **Feasibility test for hint.** The missing regions are depicted in black. We compared a series of column patches from an input image with the outermost column patch of the ground truth by using mean absolute error. We then chose the hints from the input image that are most similar to the outermost column patch of the ground truth (red- and blue-dotted boxes). Upper examples show that appropriate hints from an input image help to produce high-quality results in terms of structure and detail.

mation. This is easier to handle than outpainting because it takes advantage of the similarity at both ends of the natural image dataset. Akimoto *et al.* [2] proposed a method to include more references to adjacent information by mirroring the input image next to the masked region. These attempts are very interesting and involve simple tweaks to exiting inpainting methods, but they have some limitations in that they require datasets where both ends of the image have to be similar or symmetric.

2.3. Vision Transformer

The breakthrough in transformer networks in natural language processing has also attracted considerable interest in the field of computer vision. Transformer models have been successfully used for several tasks such as image recognition [6, 25], object detection [26, 27], image super-resolution [28], and image generation [29]. A key difference from convolutional networks that inherently incorporate inductive biases of locality is that a transformer makes no assumptions about how the data is structured. This makes the transformer universal and flexible. Generally, Vision Transformer architectures are based on a self-attention mechanism that learns the relationships between patches of an image. In this work, we utilize this patch-based self-attention mechanism which is similar to patch-based image inpainting methods to predict hints of image completion task.

3. Proposed Method

In this section, we first describe the general problem settings for image outpainting. Then, we empirically demonstrate the advantage of hint-based image outpainting via a feasibility test, and justify the need for a hint in the outpainting task.

3.1. Image Outpainting Problem Set-up

The goal of image outpainting is to generate images outside the images when the given information is only the inside of the images. Generally, the outpainting task predicts both sides of images as in Fig. 2. Although recent outpainting methods also consider single-side painting to generalize the image completion problem, we constrain our discussions to both sides painting. However, note that our proposed hint-based methodology, which will be described in Section 3.3, is model-agnostic and easily generalizable to various settings with small changes.

3.2. Exploiting Hint for Outpainting Problem

We demonstrate the effectiveness of inside information of an input image via feasibility test. We used the Beach dataset [20] consisting of $9,465$ images for training and $1,050$ images for testing with a pixel resolution of 256×256. The hints, similar to outside patches of ground truth, were chosen from the input image by using Mean Absolute Error (MAE), and these were attached to both sides of an input image as "Hint" in Fig. 2. We set the hint size as 256×16 in this feasibility test.

In the top left example of Fig. 2, hints without a tree were selected, so the tree was not repeated in Recurrent Feature Reasoning Network (RFR-Net) (w/ Hint) compared to RFR-Net (w/o Hint). In the top right example, RFR-Net (w/ Hint) was able to obtain a clearer result than RFR-Net (w/o Hint) because the detailed information regarding the wave was provided by the hint. By incorporating a hint-based technique, our method could enhance the original image completion network without any architectural change. By simply adding two small patches at both ends, our RFR-Net (w/ Hint) could achieve large performance gain (FID:

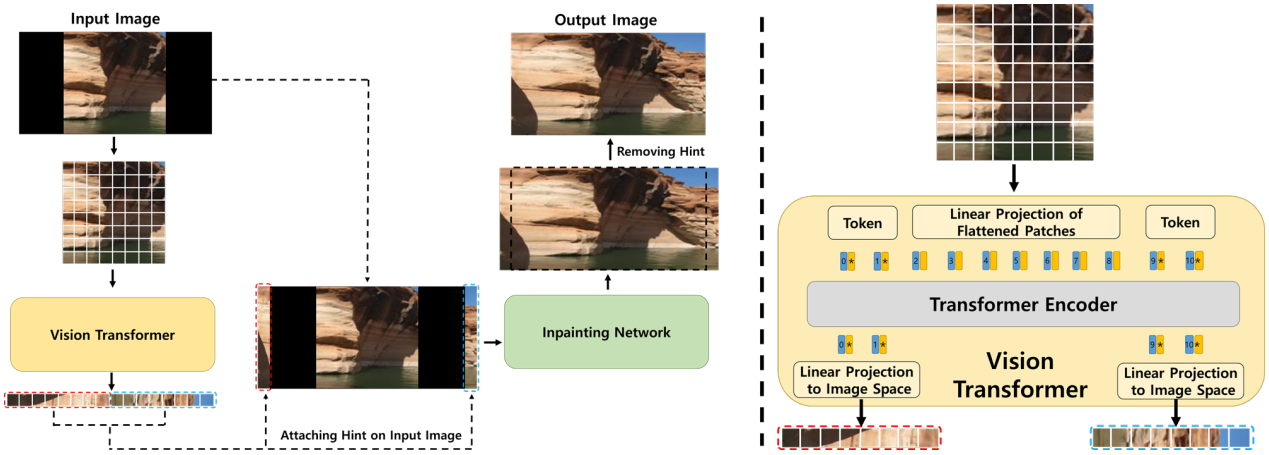

Figure 3. **Overall framework of the proposed hint-based conversion method from outpainting to inpainting.** ***Left***: An input image is divided by patches and passed through Vision Transformer to generate the hint patches. Then, hints are attached on both ends of the input image, and the missing regions are predicted by an inpainting network. The output image can be obtained by removing the hints. ***Right***: To handle 2D images of token embeddings, the patches are reshaped into a sequence of flattened 2D patches. Furthermore, learnable token embeddings are utilized to estimate the hint patches. Standard learnable 1D position embeddings are added to the patch embeddings to retain positional information. After all tokens pass through the transformer encoder, learnable token embeddings are re-projected to 2D patches and reshaped to create hints for both sides of an input image.

6.6 ↓). However, as the ground truth cannot be used in the inference stage, we need to estimate the hint using the deep neural network. The next section will describe overall framework of the proposed hint-based outpainting method.

3.3. Hint-based Conversion Framework from Outpainting to Inpainting

As shown in Fig. 3, we propose a hint-based conversion framework from outpainting to inpainting that consists of two neural network modules: 1) hint generation module and 2) inpainting module. The first module follows a Vision Transformer (ViT) model [6], *i.e.*, this module splits an image into patches and provides the sequence of linear embeddings of these patches as an input to the transformer. By comparing the representation of patches through self-attention, we can estimate the most plausible patches (hints) for the exterior of the image. After attaching the estimated hints to both ends of the input image, this image passes through the second module of our framework, *i.e.*, the inpainting network. For inpainting, we used the representative inpainting networks named EdgeConnect [15] and RFR-Net [12] in this study, but any other inpainting network can be used as well. Then, the final output image can be obtained by removing the hint from the result of the inpainting network.

Hint Generation Module As we saw in the feasibility test (Fig. 2, the performance of outpainting can be substantially improved if the patches within the image are well selected. Therefore, the goal of the hint generation module is to effectively *select the patches of the input image* to create the hint. In previous inpainting researches, patch-based methods [30, 31, 32] fill in missing regions by copying information from similar regions of the same image. These methods can easily select appropriate patches compared to other inpainting methods, but the quality is not good because the deep neural network is not used for representation learning. To exploit the patch-based methodology using the deep learning approach, we adopt a Vision Transformer [6]. This method compares the relationship between patches through self-attention and chooses the best patch through a learnable token.

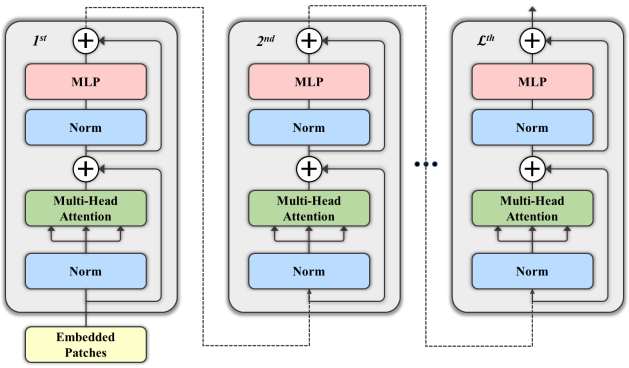

Figure 4. Detailed structure of transformer encoder. The layers of Multiheaded Self-Attention (MSA) and Multi-Layer Perceptron (MLP) blocks are alternated $\mathcal{L}$ times.

The Vision Transformer model is depicted in the right-hand side of Fig. 3. In this model design, we follow the original transformer as closely as possible. To handle 2D images, we reshape the image $\mathbf{I}_{input} \in \mathbb{R}^{H\times W/2\times C}$ into a sequence of flattened 2D patches $\mathbf{x}_p \in \mathbb{R}^{N\times(P^2\cdot C)}$, where

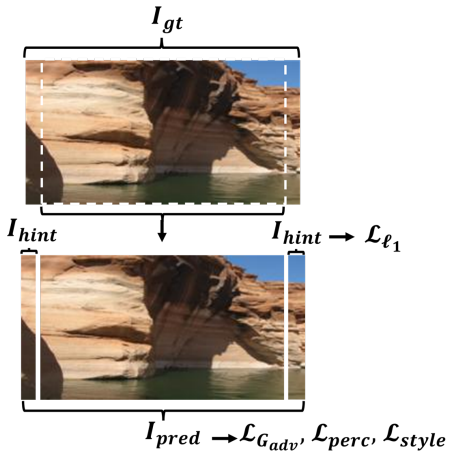

Figure 5. In the training stage of Vision Transformer, we insert the inner part of ground-truth image into the region between two hints I_{hint} to apply image-level loss such as adversarial loss, perceptual loss, and style loss.

$(H, W/2)$ is the resolution of the original image having only half the width, C is the number of channels, (P, P) is the resolution of each image patch, and $N = H \cdot (W/2)/P^2$ is the resulting number of patches. To estimate the hint patch, similar to original Vision Transformer, we utilize learnable token embeddings $\mathbf{x}_h \in \mathbb{R}^{M \times (P^2 \cdot C)}$ where $M = H \cdot (2K)/P^2$, M is the number of hint patches and K is the width of the hint patch. Position embeddings are also added to the patch embeddings to retain positional information. We use standard learnable 1D position embeddings.

In Fig. 4, the transformer encoder [33] consists of alternating layers of Multiheaded Self-Attention (MSA) and Multi-Layer Perceptron (MLP) blocks. LayerNorm (LN) is applied before every block, and residual connections after every block are as follows:

$$\mathbf{z}_0 = \left[\mathbf{x}_h^1; \mathbf{x}_h^2; \cdots \mathbf{x}_h^M; \mathbf{x}_p^1; \mathbf{x}_p^2; \cdots \mathbf{x}_p^N\right] \mathbf{E} + \mathbf{E}_{pos}, \quad (1)$$

$$\mathbf{z}'_l = \mathrm{MSA}(\mathrm{LN}(\mathbf{z}_{l-1})) + \mathbf{z}_{l-1}, \quad (2)$$

$$\mathbf{z}_l = \mathrm{MLP}(\mathrm{LN}(\mathbf{z}'_l)) + \mathbf{z}'_l, \quad (3)$$

$$\mathbf{y} = \mathrm{LN}(\mathbf{z}_L), \quad (4)$$

where $\mathbf{E} \in \mathbb{R}^{(P^2 \cdot C) \times D}, \mathbf{E}_{pos} \in \mathbb{R}^{(M+N) \times D}, l = 1, 2, ..., L$, and D is the constant latent vector size. After all tokens $\mathbf{y} \in \mathbb{R}^{(M+N) \times (P^2 \cdot C)}$ pass through the transformer encoder, hint token embeddings $\mathbf{y}_{hint} \in \mathbb{R}^{M \times (P^2 \cdot C)}$ are re-projected to 2D patches and reshaped by $\mathbf{I}_{hint} \in \mathbb{R}^{H \times (2 \cdot K) \times C}$ which indicates hints for both sides of image.

To train the proposed Vision Transformer, we exploit a joint loss similar to EdgeConnect [15]. As the estimated hint should be naturally connected to the inner region of the image, we additionally used image-level loss (adversarial loss, perceptual loss, and style loss) as well as pixel-level loss (ℓ_1 loss). Then, to apply the image-level loss, we generated I_{pred} by inserting inner part of ground-truth image into the region between two hints I_{hint} as in Fig. 5. Using the I_{pred}, the adversarial loss is employed to generate realistic results as follows:

$$\mathcal{L}_{G_{adv}} = -\mathbb{E}[D_{adv}(I_{pred})], \quad (5)$$

$$\begin{aligned} \mathcal{L}_{D_{adv}} = \mathbb{E}\left[\max\left(0, 1 - D_{adv}(I_{gt})\right)\right] \\ + \mathbb{E}\left[\max\left(0, 1 + D_{adv}(I_{pred})\right)\right], \end{aligned} \quad (6)$$

where D_{adv} is the discriminator. The perceptual loss, which penalizes results that are not perceptually similar, is defined as follows:

$$\mathcal{L}_{perc} = \mathbb{E}\left[\sum_i \frac{1}{N_i} ||\varphi_i(I_{gt}) - \varphi_i(I_{pred})||_1\right], \quad (7)$$

where φ_i is the activation map in the i'th layer of the VGG-19 network pre-trained on the ImageNet dataset [34]. The style loss is defined as follows:

$$\mathcal{L}_{style} = \mathbb{E}_j\left[||G_j^{\varphi}(I_{gt}) - G_j^{\varphi}(I_{pred})||_1\right], \quad (8)$$

where G_j^{φ} is a $C_j \times C_j$ gram matrix constructed from the activation map φ_j. To ensure proper scaling, the ℓ_1 loss is normalized by the hint size. The final total loss is defined as

$$\mathcal{L}_{ALL} = \lambda_{\ell_1}\mathcal{L}_{\ell_1} + \lambda_{G_{adv}}\mathcal{L}_{G_{adv}} + \lambda_p\mathcal{L}_{perc} + \lambda_{style}\mathcal{L}_{style}. \quad (9)$$

For our experiments, we set λ_{ℓ_1} to 1, $\lambda_{G_{adv}}$ to 0.2, λ_p to 0.1, and λ_{style} to 250, respectively. In Section 4.7, we compared the outpainting performance of the proposed Vision-Transformer-based hint and ResNet50-based hint.

Inpainting Module Next, generated hints are attached to both ends of the input image as in Fig. 3. Given that the outpainting problem is converted to an inpainting problem, we can utilize various kinds of image inpainting networks. After passing through the inpainting network, the final output can be obtained by removing the hints. The hint generation module and inpainting module are trained separately, *i.e.*, end-to-end learning is not performed for stable training.

4. Experiments

In this section, we first describe the baseline inpainting methods, datasets, and system set-up. Then, we analyze the impact of the proposed hints by using quantitative, qualitative and subjective test results. We additionally conducted ablation studies for hint sizes and hint generation modules.

Table 1. Quantitative results for conventional and proposed models on Beach dataset [20]. Evaluation of BRISQUE [35] (the lower, the better), PSNR and SSIM [36] (the higher, the better), and Fŕenchet Inception Distance (FID) [37] (the lower, the better). The best result of each column is in red and the second-best is in blue.

Image Completion Method	Outpainting to Inpainting Conversion Method	No-Reference IQA	Reference IQA		
		BRISQUE ↓	PSNR ↑	SSIM ↑	FID ↓
Image-Outpainting [20]	**w/o**	-	**14.63**	**0.34**	-
Outpainting-srn [21]	**w/o**	-	**18.22**	**0.51**	-
Boundless [23]	**w/o**	**22.22**	**19.33**	**0.79**	**35.14**
SieNet [24]	**w/o**	-	**20.80**	**0.65**	-
In-N-Out [38]	**w/o**	-	**19.52**	**0.71**	**30.17**
EdgeConnect [15]	BR [1]	-	**18.96**	**0.81**	-
	Hint	**21.93**	**20.05**	**0.81**	**29.86**
RFR-Net [12]	**w/o**	**15.28**	**19.78**	**0.80**	**36.00**
	BR [1]	**15.47**	**18.39**	**0.78**	**37.19**
	MR [2]	**14.56**	**18.02**	**0.77**	**36.65**
	Hint	**13.44**	**20.01**	**0.81**	**31.81**

Table 2. Quantitative results for conventional and proposed models on SUN dataset [39]. Evaluation of BRISQUE [35] (the lower, the better), PSNR and SSIM [36] (the higher, the better) and FID [37] (the lower, the better). The best result of each column is in red and the second-best is in blue.

Image Completion Method	Outpainting to Inpainting Conversion Method	No-Reference IQA	Reference IQA		
		BRISQUE ↓	PSNR ↑	SSIM ↑	FID ↓
Pix2Pix [40]	**w/o**	-	-	-	**19.73**
GLC [41]	**w/o**	-	-	-	**14.82**
CA [13]	**w/o**	**24.46**	**20.42**	**0.84**	**19.04**
StructureFlow [16]	**w/o**	**26.36**	**22.94**	**0.85**	**15.69**
NS-OUT [22]	**w/o**	**23.59**	**19.53**	**0.72**	**13.71**
EdgeConnect [15]	**w/o**	**23.62**	**21.41**	**0.84**	**17.75**
	BR [1]	**21.61**	**22.45**	**0.86**	**15.72**
	Hint	**23.53**	**22.15**	**0.86**	**17.17**
RFR-Net [12]	**w/o**	**20.08**	**21.95**	**0.86**	**22.90**
	BR [1]	**18.04**	**20.73**	**0.84**	**23.16**
	MR [2]	**17.95**	**20.35**	**0.83**	**24.30**
	Hint	**18.04**	**22.45**	**0.86**	**21.39**

4.1. Baseline Inpainting Methods

The inpainting module of the proposed methodology is model-agnostic. Among various types of networks, we used the representative inpainting network named EdgeConnect [15] and RFR-Net [12] in our experiments. The detailed descriptions of EdgeConnect and RFR-Net are provided in Appendix A.

4.2. Datasets

We evaluate our method on SUN [39] and Beach [20] datasets that are the most representative used in outpainting.
SUN dataset Modified Sun dataset has a pixel resolution of 256×128, which ns-out [22] used. Half of the images are taken from nature scenes of SUN dataset while others are collected from the internet. It consists of $5,000$ images for training and $1,000$ images for testing.
Beach dataset This dataset is a subset of Places365 dataset comprising beach scenes. It consists of $9,465$ images for training and $1,050$ images for testing, each having a pixel resolution of 256×256.

4.3. System Set-up

Using ViT to generate hints, we used a patch size of 16×16, transformer with embedding dimension D of 1024, MLP dimension of 2048, 6 layers, and 16 heads at each layer. The loss function consists of l_1 loss, adversarial loss, perceptual loss, and style loss with a batch size of 16 and learning rate of $1e^{-4}$. Adam optimizer with $\beta_1 = 0$ and $\beta_2 = 0.9$ was used. For training inpainting module with hints, the original setup was used. All networks were implemented using a PyTorch framework with an 24G NVIDIA RTX3090 GPU.

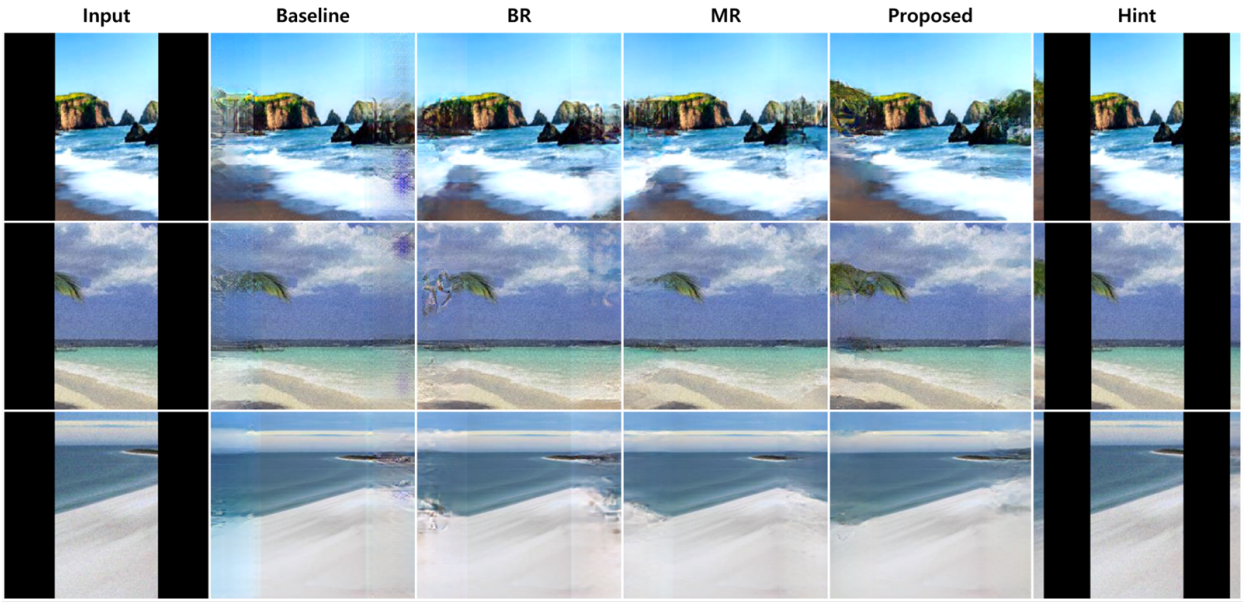

Figure 6. Qualitative results for conventional and proposed methods on non-symmetric scenes in Beach dataset.

Figure 7. Qualitative results for conventional and proposed methods on object scenes in Beach dataset.

4.4. Quantitative Results

We used a variety of evaluation metrics to diagnose the effectiveness of our method, including Blind/Referenceless Image Spatial Quality Evaluator (BRISQUE) [35], Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index Measure (SSIM) [36], and Ŕenchet Inception Distance (FID) [37]. The results for Beach and SUN datasets are presented in Tables 1 and 2, respectively. In these tables, the specific outpainting-to-inpainting conversion method was applied for each RFR-Net and EdgeConnect. Note that BR [1], MR [2] and the proposed method were tested in the same environment; the learning methods and optimizations proposed in each paper are not applied. The only difference is the conversion method from outpainting to inpainting. Our method exhibited improved performance compared to the baseline (*w/o*) result in all metrics. Also, compared to BR and MR, the proposed hint achieved similar values in terms of No-Reference IQA (BRISQUE), but better values in Reference IQA (PSNR, SSIM, and FID). It means that the proposed hint method generates a high-quality image close to the ground-truth image compared to BR and MR. Especially, our method was more effective on the Beach dataset

Table 3. Effect of hint size; The proposed hint generated by ViT is applied to an RFR-Net on SUN dataset [39] with varied horizontal sizes of hint patch K.

Hint Size (Pixels)	RFR-Net (Hint) BRISQUE ↓	PSNR ↑	SSIM ↑	FID ↓
8	**17.94**	**22.42**	**0.86**	**19.71**
16	**18.04**	**22.45**	**0.86**	**21.39**
32	**22.79**	**22.30**	**0.85**	**24.12**

because the Beach dataset includes more asymmetric images than SUN dataset.

4.5. Qualitative Results

Figs. 6 and 7 show a comparison of qualitative results among Baseline, BR, MR, and the proposed method. Fig. 6 portrays the results on asymmetric scenes in Beach dataset while Fig. 7 portrays results on objects scenes in Beach dataset. In Fig. 6, MR and BR tend to produce structurally unnatural image for asymmetric data. They could not produce images of beachfronts or waves that are naturally connected in a straight line. Instead, they produced unnatural images of V- or N-shaped curves. In Fig. 7, MR often generated repetitive structure for the boundary regions of an image when there were objects in the center of the image due to the mirrored flipping. As BR rearranges the images at both ends, distortion often occurred because of the bias of connecting the images on both sides (Fig. 7). Our proposed method produced images that had high content preserving ability with structure coherence and were more natural than those obtained by using MR and BR. Additional results are in Appendix B.

4.6. Subjective Test using Mean Opinion Score

We have performed a Mean Opinion Score (MOS) test to quantify the quality of outpainting-to-inpainting conversion methods. Specifically, we asked 30 raters to assign an integral score from 1 (bad quality) to 5 (excellent quality) to the outpainted images. Fig. 8 shows the summarized results. For each method 900 samples (30 images $\times$ 30 raters) were assessed. Consequently, the comparison results verified that our proposed method outperforms other methods in generating a visually clearer image for human viewers.

4.7. Ablation Studies

Effect of Hint Size Table. 3 summarizes the performance of RFR-Net with various hint sizes. It shows that the performance deteriorates as the size of hint increases. The difference stands out in the FID score. This result supports the reason we did not use ViT for generating the entire missing regions and only use for generating small size of hints.

Effectiveness of Hint Generation Module We used ResNet50 and ViT [6] for generating a hint to compare the performance of convolutional neural network (CNN) model and patch-based model. From Table. 4, we can conclude that the patch-based model is better than CNN model for hint generation.

Table 4. Effect of hint module architectures; The hints generated by Vision Transformer [6] or ResNet50 [42] were applied to an RFR-Net on SUN dataset [39].

Hint Module (Architecture)	RFR-Net (Hint) BRISQUE ↓	PSNR ↑	SSIM ↑	FID ↓
ViT [6]	**18.04**	**22.45**	**0.86**	**21.39**
ResNet50 [42]	**20.39**	**21.60**	**0.85**	**23.16**

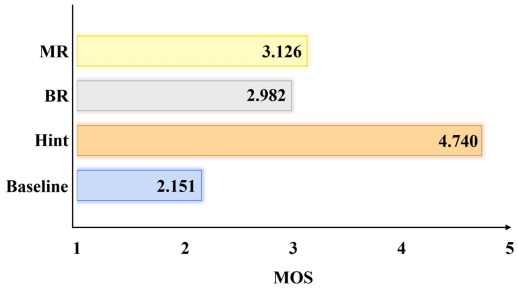

Figure 8. MOS scores on Beach dataset. Comparison of the outpainting-to-inpainting conversion methods.

5. Conclusion

Recent studies attempted to solve the outpainting problem by converting image outpainting to image inpainting. However, there are some limitations in that they require datasets where both ends of the image have to be similar or symmetric. In this paper, we proposed a novel outpainting-to-inpainting conversion method that uses Vision Transformer to select the image-adaptive hints. The proposed hint-based conversion framework consists of two neural network modules: 1) hint generation module and 2) inpainting module. By incorporating the hint-based technique, our method enhanced the performance of the original image completion network without requiring any architectural change. Among the outpainting to inpainting conversion methods, our method achieved outstanding performance and realistic image generation.

6. Acknowledgments

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. 2020M3H4A1A02084899 and No. 2021R1A2C1004208), Convergent Technology RD Program for Human Augmentation through the National Research Foundation of Korea (NRF) funded by Ministry of Science and ICT (2020M3C1B8081320), and Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (2020R1A6A3A01098940 and 2021R1I1A1A01051225).

References

[1] Kyunghun Kim, Yeohun Yun, Keon-Woo Kang, Kyeongbo Kong, Siyeong Lee, and Suk-Ju Kang. Painting outside as inside: Edge guided image outpainting via bidirectional rearrangement with progressive step learning. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2122–2130, 2021.

[2] Naofumi Akimoto, Daiki Ito, and Yoshimitsu Aoki. Scenery image extension via inpainting with a mirrored input. *IEEE Access*, 9:59286–59300, 2021.

[3] Kaiming He, Huiwen Chang, and Jian Sun. Rectangling panoramic images via warping. *ACM Trans. Graph.*, 32(4), July 2013.

[4] Sung In Cho and Suk-Ju Kang. Extrapolation-based video retargeting with backward warping using an image-to-warping vector generation network. *IEEE Signal Processing Letters*, 27:446–450, 2020.

[5] Qingguo Xiao, Guangyao Li, and Qiaochuan Chen. Image outpainting: Hallucinating beyond the image. *IEEE Access*, 8:173576–173583, 2020.

[6] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2020.

[7] Coloma Ballester, Marcelo Bertalmio, Vicent Caselles, Guillermo Sapiro, and Joan Verdera. Filling-in by joint interpolation of vector fields and gray levels. *IEEE transactions on image processing*, 10(8):1200–1211, 2001.

[8] Marcelo Bertalmio, Guillermo Sapiro, Vincent Caselles, and Coloma Ballester. Image inpainting. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, pages 417–424, 2000.

[9] Alexei A Efros and Thomas K Leung. Texture synthesis by non-parametric sampling. In *Proceedings of the seventh IEEE international conference on computer vision*, volume 2, pages 1033–1038. IEEE, 1999.

[10] Guilin Liu, Fitsum A. Reda, Kevin J. Shih, Ting-Chun Wang, Andrew Tao, and Bryan Catanzaro. Image inpainting for irregular holes using partial convolutions. In *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018.

[11] Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A. Efros. Context encoders: Feature learning by inpainting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[12] Jingyuan Li, Ning Wang, Lefei Zhang, Bo Du, and Dacheng Tao. Recurrent feature reasoning for image inpainting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7760–7768, 2020.

[13] Yu, Jiahui and Lin, Zhe and Yang, Jimei and Shen, Xiaohui and Lu, Xin and Huang, Thomas S. Generative image inpainting with contextual attention. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5505–5514, 2018.

[14] Jiahui Yu, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and Thomas S. Huang. Free-form image inpainting with gated convolution. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019.

[15] Kamyar Nazeri, Eric Ng, Tony Joseph, Faisal Qureshi, and Mehran Ebrahimi. Edgeconnect: Structure guided image inpainting using edge prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops*, Oct 2019.

[16] Yurui Ren, Xiaoming Yu, Ruonan Zhang, Thomas H. Li, Shan Liu, and Ge Li. Structureflow: Image inpainting via structure-aware appearance flow. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019.

[17] Josef Sivic, Biliana Kaneva, Antonio Torralba, Shai Avidan, and William T Freeman. Creating and exploring a large photorealistic virtual space. In *2008 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, pages 1–8. IEEE, 2008.

[18] Yinda Zhang, Jianxiong Xiao, James Hays, and Ping Tan. Framebreak: Dramatic image extrapolation by guided shift-maps. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1171–1178, 2013.

[19] Miao Wang, Yu-Kun Lai, Yuan Liang, Ralph R Martin, and Shi-Min Hu. Biggerpicture: data-driven image extrapolation using graph matching. *ACM Transactions on Graphics*, 33(6), 2014.

[20] Mark Sabini and Gili Rusak. Painting outside the box: Image outpainting with gans. *arXiv preprint arXiv:1808.08483*, 2018.

[21] Yi Wang, Xin Tao, Xiaoyong Shen, and Jiaya Jia. Wide-context semantic image extrapolation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1399–1408, 2019.

[22] Zongxin Yang, Jian Dong, Ping Liu, Yi Yang, and Shuicheng Yan. Very long natural scenery image prediction by outpainting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10561–10570, 2019.

[23] Piotr Teterwak, Aaron Sarna, Dilip Krishnan, Aaron Maschinot, David Belanger, Ce Liu, and William T Freeman. Boundless: Generative adversarial networks for image extension. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10521–10530, 2019.

[24] Xiaofeng Zhang, Feng Chen, Cailing Wang, Ming Tao, and Guo-Ping Jiang. Sienet: Siamese expansion network for image extrapolation. *IEEE Signal Processing Letters*, 2020.

[25] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *International Conference on Machine Learning*, pages 10347–10357. PMLR, 2021.

[26] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European Conference on Computer Vision*, pages 213–229. Springer, 2020.

[27] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable {detr}: Deformable transformers for end-to-end object detection. In *International Conference on Learning Representations*, 2021.

[28] Fuzhi Yang, Huan Yang, Jianlong Fu, Hongtao Lu, and Baining Guo. Learning texture transformer network for image super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5791–5800, 2020.

[29] Hanting Chen, Yunhe Wang, Tianyu Guo, Chang Xu, Yiping Deng, Zhenhua Liu, Siwei Ma, Chunjing Xu, Chao Xu, and Wen Gao. Pre-trained image processing transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12299–12310, 2021.

[30] Connelly Barnes, Eli Shechtman, Adam Finkelstein, and Dan B Goldman. Patchmatch: A randomized correspondence algorithm for structural image editing. *ACM Trans. Graph.*, 28(3):24, 2009.

[31] Soheil Darabi, Eli Shechtman, Connelly Barnes, Dan B Goldman, and Pradeep Sen. Image melding: Combining inconsistent images using patch-based synthesis. *ACM Transactions on graphics (TOG)*, 31(4):1–10, 2012.

[32] Jia-Bin Huang, Sing Bing Kang, Narendra Ahuja, and Johannes Kopf. Image completion using planar structure guidance. *ACM Transactions on graphics (TOG)*, 33(4):1–10, 2014.

[33] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.

[34] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015.

[35] Anish Mittal, Anush Krishna Moorthy, and Alan Conrad Bovik. No-reference image quality assessment in the spatial domain. *IEEE Transactions on image processing*, 21(12):4695–4708, 2012.

[36] Wang, Zhou and Bovik, Alan C and Sheikh, Hamid R and Simoncelli, Eero P. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.

[37] Heusel, Martin and Ramsauer, Hubert and Unterthiner, Thomas and Nessler, Bernhard and Hochreiter, Sepp. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *Advances in neural information processing systems*, pages 6626–6637, 2017.

[38] Changho Jo, Woobin Im, and Sung-Eui Yoon. In-n-out: Towards good initialization for inpainting and outpainting. *arXiv preprint arXiv:2106.13953*, 2021.

[39] Jianxiong Xiao, James Hays, Krista A Ehinger, Aude Oliva, and Antonio Torralba. Sun database: Large-scale scene recognition from abbey to zoo. In *2010 IEEE computer society conference on computer vision and pattern recognition*, pages 3485–3492. IEEE, 2010.

[40] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1125–1134, 2017.

[41] Iizuka, Satoshi and Simo-Serra, Edgar and Ishikawa, Hiroshi. Globally and locally consistent image completion. *ACM Transactions on Graphics (ToG)*, 36(4):1–14, 2017.

[42] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.