TFill: Image Completion via a Transformer-Based Architecture

Chuanxia Zheng Tat-Jen Cham
School of Computer Science and Engineering
Nanyang Technological University, Singapore
{chuanxia001,astjcham}@ntu.edu.sg

Jianfei Cai
Department of Data Science & AI
Monash University, Australia
Jianfei.Cai@monash.edu

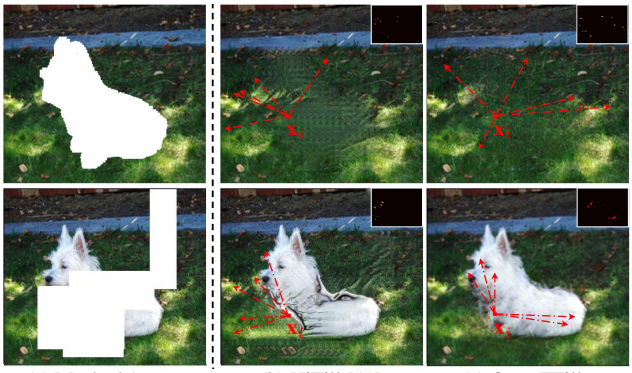
Abstract

Bridging distant context interactions is important for high quality image completion with large masks. Previous methods attempting this via deep or large receptive field (RF) convolutions cannot escape from the dominance of nearby interactions, which may be inferior. In this paper, we propose treating image completion as a directionless sequence-to-sequence prediction task, and deploy a transformer to directly capture long-range dependence in the encoder in a first phase. Crucially, we employ a restrictive CNN with small and non-overlapping RF for token representation, which allows the transformer to explicitly model the long-range context relations with equal importance in all layers, without implicitly confounding neighboring tokens when larger RFs are used. In a second phase, to improve appearance consistency between visible and generated regions, a novel attention-aware layer (AAL) is introduced to better exploit distantly related features and also avoid the insular effect of standard attention. Overall, extensive experiments demonstrate superior performance compared to state-of-the-art methods on several datasets. The code will be available at https://github.com/lyndonzheng/TFill.

1. Introduction

Image completion refers to the task of filling reasonable content with photorealistic appearance into missing regions, conditioned on partially visible information. Earlier methods infer the pixels of missing regions by propagating or copying pieces from neighboring visible regions [3, 1, 7, 2], while more recent ones directly learn to generate content and appearance using deep neural networks [28, 15, 42, 23, 49, 24, 27, 41].

A main challenge in this task is the requirement of *bridging and exploiting visible information globally, after it had been degraded by arbitrary masks*. As depicted in Fig. 1 top row, when the entire dog is masked, the natural expectation is to complete the masked area based on the visible background context. In contrast, in the bottom row, when the free-form regular mask covers the bulk of the dog but leaves the head and tail visible, it is necessary but highly challenging to globally capture *long-range* dependencies between the two separated foreground regions, so that the masked area can be completed in not just a photorealistic, but also semantically correct, manner.



(a) Masked 1nput (b) HiFill [41] (c) Ours-TFill

Figure 1. **An example of information flow in image completion with free-form masks.** The position $\mathbf{x}_i$'s response (flow) is calculated by inferring the *Jacobian* matrix between it to all pixels in the given masked input. Here, only the highest flows are shown. Our TFill correctly captures long-range visible context flow, even with a large mask splitting two semantically important zones.

To achieve this goal, many *two-stage* approaches [42, 27, 41, 44] have been proposed, consisting of a *content inference network* and an *appearance refinement network*. They typically infer a coarse image or edge/semantic map based on globally visible information in a first phase, and then fill in visually realistic appearance in a second phase. However, this global perception is achieved by repeated *local* convolutional operations, which have several limitations. First, due to translation equivariance in convolutions, the information flow tends to be predominantly local, with global information only shared gradually through heat-like propagation across multiple layers. Second, during inference, the elements between adjacent layers are connected via learned but fixed weights, rather than input-dependent adaptive weight-

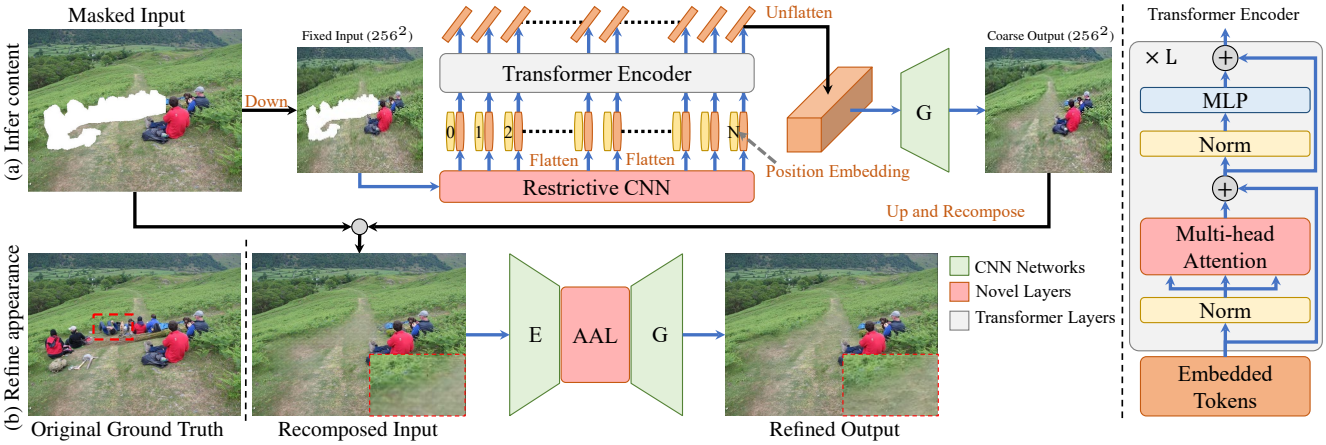


Figure 2. **The overall pipeline of the proposed method.** (a) Masked input is resized to a fixed low resolution (256^2) and it is then fed into the transformer to generate semantically correct content. (b) The inferred content is merged with the original high resolution image and passed to a refinement network with an **A**ttention-**A**ware **L**ayer (**AAL**) to transfer high-quality information from both visible and masked regions. Note the recomposed input has repeating artifacts, which is resolved in our refined network. Zoom in to see the details.

ings. These issues mean long-distance messages are only delivered inefficiently in a very deep layer, resulting in a strong inclination for the network to fill holes based on nearby rather than distant visible pixels (Fig. 1 (b)).

In this paper, we propose an alternative perspective by treating image completion as a *directionless sequence-to-sequence* prediction task. In particular, instead of modeling the global context using deeply stacked convolutional layers, we design a new content inference model, called TFill, that uses a **T**ransformer-based architecture to **Fill** reasonable content into the missing holes. An important insight here is that a transformer directly exploits long-range dependencies at every encoder layer through the attention mechanism, which *creates an equal flowing opportunity for all visible pixels, regardless of their relative spatial positions* (Fig. 1 (c)). This reduces the proximity-dominant influence that can lead to semantically incoherent results.

Our design is motivated by the transformer literature in natural language processing (NLP) [37, 8, 29, 30]. However, it remains a challenge to directly apply these transformer models to visual generation tasks. Particularly, unlike the NLP that naturally treats each word as a vector for token embedding, it is unclear what a good token representation should be for visual task. If we use every pixel as a token, the memory cost will make this infeasible except for very small images [6]. To mitigate this issue, our model embeds the masked image into an intermediate latent space for token representation, an approach also broadly taken by recent vision transformer models [4, 53, 10, 50]. However, unlike these models that use traditional CNN-based encoders to embed the tokens, we propose a *restrictive CNN* for token representation, which has a profound influence on how the visible information is connected in the network. To do so, we ensure the individual tokens represent visible information independently, each with a *small* and *non-overlapping* receptive field (RF). This forces *the long-range context relationships between tokens to be explicitly and co-equally perceived in every transformer encoder layer*, without neighboring tokens being entangled by implicit correlation through overlapping RF. As a result, each token will *not* be gradually affected by neighboring regions.

While the proposed transformer-based architecture can achieve better results than state-of-the-art methods [42, 49, 41, 10], by itself it only works for a fixed sequence length because of the position embedding (Fig. 2(a)). To allow our approach to flexibly scale to images of different sizes, a fully convolutional encoder-decoder network (Fig. 2(b))) is subsequently applied to refine the visual appearance built upon the coarse content previously inferred. We also design a novel **A**ttention-**A**ware **L**ayer (AAL) between the encoder and decoder that adaptively balances the attention paid to visible and generated content, leading to semantically superior feature transfer.

We highlight our main contributions as follows: **1)** A restrictive CNN head is introduced for individual token representation, which mitigates the proximity influence when propagating from local visible regions to missing holes. **2)** Through a transformer-based architecture, the long-range interactions between these tokens are explicitly modeled, in which the masked tokens are perceptive of other visible tokens with equal opportunity, regardless of their positions. This results in a profound improvement over previous CNN-based architecture. **3)** We designed a novel attention-aware layer with adaptive attention balancing in a refined stage to obtain higher quality and resolution results. **4)** Finally, extensive experiments demonstrate that the proposed model outperforms the existing state-of-the-art image completion models based on convolutional architectures.

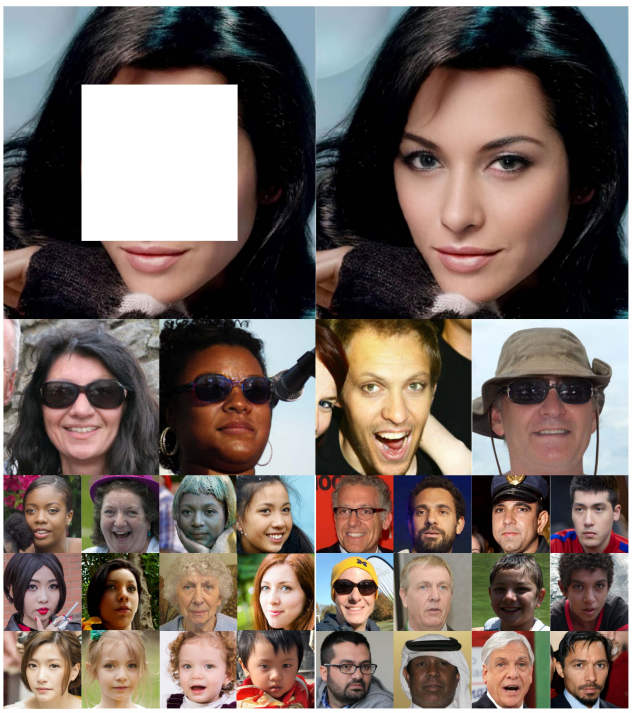


Figure 3. **Example completion results of our method (config E) on face datasets.** Here, a center mask was used for all input images. The corresponding quantitative results are reported in Tables 1 and 2. One center masked example input is shown top-left.

2. Methods

In this section, we describe our framework together with covering related work, to provide a better context.

Given a masked image $\mathbf{I}_m$, degraded from a real image $\mathbf{I}$ by a free-form mask, our goal is to learn a model Φ to infer the content for missing regions, as well as filling in with visually realistic appearance. To achieve this, our image completion framework, illustrated in Fig. 2, consists of a content inference network and an appearance refinement network. The former is responsible for capturing the global context through a transformer encoder at a fixed scale. The embedded tokens have small receptive fields (RF) and limited capacity, preventing their states from being implicitly dominated by visible pixels nearby than far. While similar transformer-based architectures have recently been explored for visual tasks [6, 9, 4, 53, 10, 38, 5, 50], we believe our work is the first to explore this for free-form image completion, where we discover *how the token representation has a profound effect on the flow of visible information in the network, in spite of the supposedly global reach of transformers*. The latter network is designed to refine visual appearance by utilizing high-resolution visible features, and also frees the limitation to fixed image sizes.

	Method	CelebA-HQ		FFHQ	
		LPIPS↓	FID↓	LPIPS↓	FID↓
	CA [42]	0.104	9.53	0.127	8.78
	PICNet [49]	0.061	6.43	0.068	4.61
	MEDFE [24]	0.067	7.01	-	-
A	Traditional *Conv*	0.060	6.29	0.066	4.12
B	+ Attention in G	0.059	6.34	0.064	4.01
C	+ Restrictive *Conv*	0.056	4.68	0.060	3.87
D	+ Transformer	0.051	4.02	0.057	3.66
E	+ Masked Attention	0.050	3.92	0.057	3.63
F	+ Refine Network	**0.048**	**3.86**	**0.053**	**3.50**

Table 1. Learned Perceptual Image Patch Similarity (LPIPS) and Fréchet Inception Distance (FID) for various completion networks on center masked images. In this paper, we calculate the LPIPS and FID using all images in the corresponding test sets.

2.1. Transformer-based Architecture

Background: We begin by briefly reviewing the transformer [37]. As depicted on Fig. 2 (right), a transformer encoder layer consists of multihead `self-attention` (MSA) and `Multilayer Perception` (MLP) blocks (see Appendix C.1). The MSA is responsible for capturing long-range dependencies, while the MLP is applied to further transform merged features. The `Layernorm` (LN) is used for non-linear projection. These are expressed by:

$$\mathbf{z}_0 = [\mathbf{x}^1; \mathbf{x}^2; \dots; \mathbf{x}^N] + \mathbf{E}_{pos} \quad (1)$$

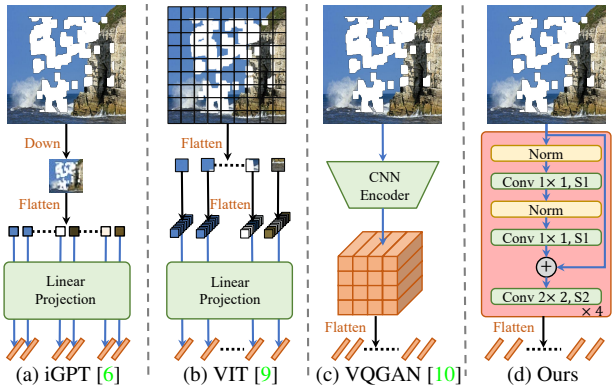
$$\mathbf{z}'_\ell = \text{MSA}(\text{LN}(\mathbf{z}_{\ell-1})) + \mathbf{z}_{\ell-1} \quad (2)$$

$$\mathbf{z}_\ell = \text{MLP}(\text{LN}(\mathbf{z}'_\ell)) + \mathbf{z}'_\ell \quad (3)$$

where $\mathbf{z} \in \mathbb{R}^{N \times C}$ is the 1D sequence of N tokens $\mathbf{x}$ with C channels, and $\mathbf{E}_{pos} \in \mathbb{R}^{N \times C}$ is the position embedding.

Transformer-Encoder: In order to feed a 2D masked image $\mathbf{I}_m$ into the transformer encoder, we first downsample the high-resolution image to a fixed size, *e.g.* 256^2. However, it is *not* feasible to run the transformer model if we directly *flatten* image pixels into a 1D sequence with 196,608 tokens. To achieve independent token representation and reduce its length, a projection is implemented using our proposed *restrictive CNN*, a decision we will analyze in Sec. 2.1.1. After that, we obtain a 2D feature map with size $\frac{256}{16} \times \frac{256}{16} \times C$, and then flatten it to a 1D sequence of $256 \times C$, where 256 is the sequence length and C=512 is the feature dimension. As shown in Fig. 2 (a), once we embed the image to a 1D sequence, a transformer encoder distills long-range relationships between all tokens in every layer.

To encourage the model to *bias* to the important visible values, we replace the self-attention layer with the *masked* self-attention layer, in which a weight is applied to scale the attention scores. The initial weight $w_{key} \in (0.02, 1.0]$ is obtained by calculating the fraction of visible pixels in a



(a) iGPT [6] (b) VIT [9] (c) VQGAN [10] (d) Ours

Figure 4. **Token representation.** (a) Pixel to token. (b) Patch to token. (c) Feature to token. (d) Restrictive **R**eceptive **F**ield (RF) feature to token. Note our token has a small and non-overlapping RF like VIT [9], but uses a complex CNN embedding. Each token represents locally isolated contexts, leaving the long-range relationship to be cleanly modeled in the transformer encoder.

small RF, *e.g.* $192/16^2$ means $3/4$ of the region in the 16^2 RF contains visible pixels. It will then be gradually amplified by updating $w_{key} \leftarrow \sqrt{w_{key}}$ after every encoder layer, to *reflect* visible information flow. This initial ratio for each token is efficiently implemented in our restrictive CNN encoder using a modified partial convolution layer [23]. The implementation details can be found in Appendix C.

CNN-based Decoder: While a one-layer non-linear projection may be used to directly map the output features back to a completed image, the visual appearance is slightly worse than using a stacked decoder. Therefore, following existing works [42, 49, 41], a gradual upsampling decoder is implemented to generate photorealistic images.

2.1.1 Results and Analysis

Results: We first demonstrate experimentally that the transformer-based model outperforms previous CNN-based models. Table 1 shows Learned Perceptual Image Patch Similarity (LPIPS)[1] [47] and Fréchet Inception Distance (FID) [12] for various image completion architectures on CelebA-HQ [25, 19] and FFHQ [20] datasets degraded by center masks. The traditional image quality results are given in Appendix B. Here, we compared with three CNN-based models, for which CA [42] and PICNet [49] had the appropriate pretrained models available, while the latest MEDFE [24] was reproduced using their publicly available code. All scores are reported for 256^2 resolution. Without bells and whistles, our TFill-*Coarse* with configuration (E) improved LPIPS (18% relative improvement) and FID (39% relative improvement) quite significantly on CelebA-HQ, despite only using the transformer-based content inference network, without our refinement network.

[1]While multi-modal generation tasks had previously been evaluated with LPIPS [52, 13, 49], it was used to measure diversity. Here, we apply it to measure the similarity between completed images and original ground-truth. A smaller value means the completed image is closer to the ground-truth image w.r.t. the learned perceptual similarity, rather than pixel-level reconstruction. We refer readers to [47] for details.

	Method	LPIPS↓	FID↓	Mem↓	Time↓
	IGPT [6] (RF 1)	0.609	148.42	3.16	26.45
	VIT [9] (RF 16)	0.062	5.09	1.16	0.167
	VQGAN [10]	0.226	11.92	2.36	4.29
B	*Conv* (RF 229)	0.064	4.01	0.99	0.162
C	Ours *R-Conv* (RF 16)	0.060	3.87	0.90	**0.157**
	T-based (RF 229)	0.062	3.92	1.25	0.188
E	T-based (RF 16)	**0.057**	**3.63**	**1.15**	0.180

Table 2. The effect of restrictive token embedding and transformer block in our transformer-based completion network on FFHQ dataset. "RF" indicates the Receptive Field size. "Mem" denotes the memory (GB) cost during testing and "Time" is the testing time (s) for each center masked image.

Fig. 3 shows the visual results of our TFill on CelebA-HQ and FFHQ datasets. More visual results for other datasets are given in Appendix A. Here, all images are center masked in order to demonstrate its ability to go beyond object removal and to generate reasonable semantic content for large missing regions. As can be seen, the completed images are on average of high quality. Even for some challenging cases, such as when eyeglasses are center masked, our TFill can correctly repair the face *with* eyeglasses. Furthermore, it generally works well for varied skin tones, poses, expressions, ages, and illumination.

Analysis: Our baseline configuration (A) used the same encoder-decoder structure as VQGAN [10], except here attention layers were removed for a pure CNN-based version. When combined with the powerful discriminator of StyleGANv2 [21], the performance was comparable to PICNet [49], in which the best results were selected from 50 diverse samples. We first added the attention layer to the decoder (Generator, G) in (B), but the performance remained similar to baseline (A). In contrast, when we use our proposed *restrictive CNN* in (C), the performance improved substantially, especially for FID. This suggests that the input feature representation is significant for the attention layer to equally deliver all messages, as explained later. We then improved this new baseline by adding the transformer encoder (D), which benefits from globally delivered messages at every layer. Finally, we introduced masked weights to each attention layer of the transformer (E), improving results further.

To study the influence of the token representation, we conducted two experiments that compared with recent visual transformer works [6, 9, 10] and provided an ablation

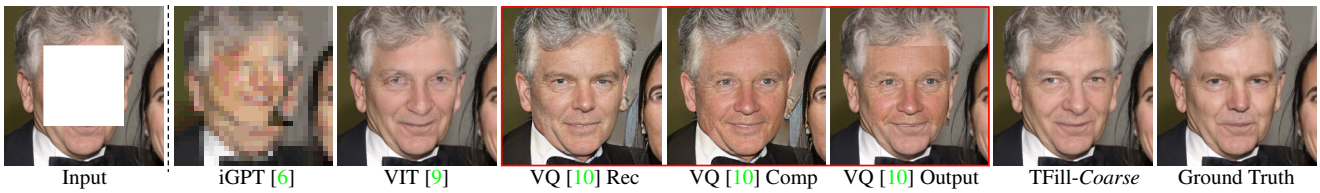


Figure 5. **Comparing results under different token representations.** All transformers are based on the same transformer backbone [37]. For VQGAN [10], we report reconstruction (Rec) image, completed (Comp) image and recomposed output image. TFill-*Coarse* is our model with configuration E in Tables 1 and 2, *i.e*. TFill without the refinement network. Please see main text for details.

study by controlling the RF in Table 2.

As illustrated in Fig. 4, iGPT [6] downsamples the image to a fixed scale, *e.g*. 32^2 resolution, and embeds *each pixel to a token*. While this may not impact the original classification task which is robust to low resolutions [35], it has a large negative effect on generating high-quality images. Furthermore, the auto-regressive form resulted in the completed image being inconsistent with the bottom-right visible region (iGPT in Fig. 5), and each image runs an average of 26.45s during the testing. This is because the conditional sequence generation can only utilize the top-left visible pixels, generating new pixels one-by-one. In contrast, VIT [9] divides an image to a set of fixed patches and embeds *each patch to a token*. As shown in Table 2 and Fig. 5, it can achieve relatively good quantitative and qualitative results. However, some details are perceptually poor, *e.g*. the strange eyes in Fig. 5, possibly due to the limited one-layer linear projection. Finally, VQGAN [10] employs a traditional CNN to encode an image to the feature domains and then *quantizes each feature as a token* through a learned codebook [36, 31]. Fig. 5 shows the generated images using tokens embedded from ground truth (VQ Rec), and tokens extracted from the center masked image (VQ Comp). While it generates the content of missing regions sequentially conditioned only on top-left visible tokens, we found the completed pixels to be consistent with the bottom-right region, even though these tokens were *not* used to infer missing content in the transformer encoder. We believe this is due to the large RF in the CNN-based encoder causing each token to capture extended dependencies in a deep layer. However, this leads to two issues: **1)** even the original visible tokens are modified, resulting in different appearances for the visible regions *e.g*. see VQ Rec *vs* VQ Comp in Fig. 5; **2)** inferred tokens are unduly influenced by implicit CNN-based correlation to nearby tokens, and cannot establish ties cleanly to important but distant tokens. Thus it generates a visually realistic completion, but when pasted to the original masked input (VQ Output in Fig. 5), there is an obvious gap between generated and visible pixels.

In contrast to [6, 9, 10], our token representation is extracted using a *restrictive CNN* (Fig. 4(d)). In particular, the 1×1 filter and `layernorm` is applied for non-linear projection, followed by a partial convolution layer [23] that uses a 2×2 filter with stride 2 to extract visible information and reduce feature resolution simultaneously. For instance, if half of the pixels in a window are masked, we only embed the other 50% comprising visible pixels as our token representation, and establish an initial weight of 0.5 for the *masked* self-attention layer. To do this, we ensure each token represents only the visible information in a small RF, *leaving the long-range dependencies to be explicitly modeled by the transformer encoder in every layer*, without cross-contamination from implicit correlation due to larger CNN RF. To demonstrate the impact of RF, a thorough ablation study result is reported in Table 2, in which we find the small RF CNN improves both LPIPS and FID significantly, with the added benefit of low memory cost. Furthermore, our model runs at 180ms per image on an Nvidia GTX 1080Ti (+21ms CPU time for resizing input and storing output), due to predicting all output heads together, rather than auto-regressively as in existing work [6, 10].

2.1.2 Discussion on prior art

Driven by the advances of GANs [11], CGANs [26] and VAEs [22], many learning-based approaches have been proposed for image completion, consisting of *one-stage* methods [28, 15, 23, 49, 17, 48, 24] and *two-stage* methods [40, 42, 33, 34, 27, 43, 41, 45, 44]. While these approaches rapidly improved results, content inference has mostly been directed by CNN operations [23, 43] and new architectures [49, 24], or by using exact auxiliary information, *e.g*. edges [27]. In contrast, our TFill is designed to directly model long-range effects of visible information through a transformer model, in which the system is *not* unduly influenced to fill in masked regions based on neighboring pixels first, but rather to exploit distant visible content.

Inspired by the dramatic success of transformers in NLP [37, 8, 30], recent works have explored applying a standard transformer for visual tasks, such as image classification [6, 9], object detection [4, 53], semantic segmentation [38, 50], image generation and translation [10, 5, 14, 16]. Many of them embed the tokens using the methods shown in Fig. 4 (a)-(c). Compared to these general token representations, our *restrictive CNN* is particularly well suited due to its compact representation that limits implicit correlation.

(a) Ground Truth (b) Input

(c) TFill-*Coarse* (d) TFill-*Refined*

Figure 6. **Coarse and Refined results**. (a) Ground truth. (b) Masked input degraded by free-form masks. (c) Coarse output. (d) Refined output. We can see that the refinement network not only increased image quality to a high resolution (256^2 *vs* 512^2), but also encourages the left eyeball to be consistent with the visible right eyeball using our attention-aware layer.

2.2. Attention-Aware Layer (AAL)

Although our TFill-*Coarse* model correctly infers reasonable content by equally utilizing the global visible information in every layer, two limitations remain. First, it is *not* suitable for high-resolution input due to the fixed length position embedding. One solution is to follow the directional sequence-to-sequence methods [6, 10] that only use the top-left context to predict the next token, in an auto-regressive manner. However, this will not adequately capture the global visible information needed for image completion. Second, the realistic completed results may not be fully consistent with the original visible appearances, *e.g.* the generated left eye having a different shape and color to the visible right eye in Fig. 6 (c). This is because the embedded tokens are extracted from a 16^2 resolution feature map, where important high-frequency details may be lost.

To mitigate these issues, a CNN-based encoder-decoder refinement network, trained on high-resolution images, is proposed (Fig. 2 (b)). In particular, to further utilize the visible high-frequency details, an **A**ttention-**A**ware **L**ayer (AAL) is designed to capture long-range dependencies.

As depicted in Fig. 7, given a decoded feature $\mathbf{x}_d$, we first calculate the attention score of:

$$\mathbf{A} = \phi(\mathbf{x}_d)^\intercal \theta(\mathbf{x}_d) \tag{4}$$

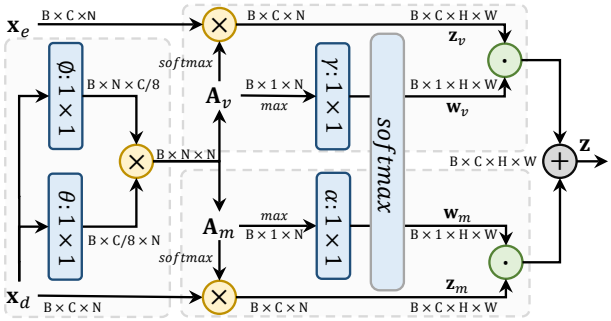


Figure 7. **Attention-aware layer.** The feature maps are shown as tensors. "$\otimes$" denotes matrix multiplication, "$\odot$" denotes element-wise multiplication and "$\oplus$" is element-wise sum. The blue boxes denote 1×1 convolution filters that are learned.

where $\mathbf{A}_{ij}$ represents the similarity of the i^{th} patch to the j^{th} patch, and ϕ, θ are 1×1 convolution filters.

Interestingly, we discover that using $\mathbf{A}$ directly in a standard self-attention layer is suboptimal, because the $\mathbf{x}_d$ features for visible regions are generally distinct from those generated for masked regions. Consequently, *the attention tends to be insular*, with masked regions preferentially attending to masked regions, and vice versa. To avoid this problem, we explicitly handled the attention to visible regions separately from masked regions. So before `softmax` normalization, $\mathbf{A}$ is split into two parts: $\mathbf{A}_v$ — similarity to visible regions, and $\mathbf{A}_m$ — similarity to generated masked regions. Next, we get long-range dependencies via:

$$\mathbf{z}_v = \texttt{softmax}(\mathbf{A}_v)\mathbf{x}_e \quad , \quad \mathbf{z}_m = \texttt{softmax}(\mathbf{A}_m)\mathbf{x}_d \tag{5}$$

where $\mathbf{z}_v$ contains features of contextual flow [42] for copying high-frequency details from the encoded high-resolution features $\mathbf{x}_e$ to masked regions, while $\mathbf{z}_m$ has features from the self-attention that is used in SAGAN [46] for high-quality image generation.

Instead of learning fixed weights [49] to combine $\mathbf{z}_v$ and $\mathbf{z}_m$, we learn the *weights mapping* based on the largest attention score in each position. Specifically, we first obtain the largest attention score of $\mathbf{A}_v$ and $\mathbf{A}_m$, respectively. Then, we use the 1×1 filter γ and α to *modulate* the ratio of the weights. `Softmax` normalization is applied to ensure $\mathbf{w}_v+\mathbf{w}_m=1$ in every spatial position:

$$[\mathbf{w}_v, \mathbf{w}_w] = \texttt{softmax}([\gamma(\texttt{max}(\mathbf{A}_v)), \alpha(\texttt{max}(\mathbf{A}_m)]])) \tag{6}$$

where `max` is executed on the attention score channel. Finally, an attention-balanced output $\mathbf{z}$ is obtained by:

$$\mathbf{z} = \mathbf{w}_v \cdot \mathbf{z}_v + \mathbf{w}_m \cdot \mathbf{z}_m \tag{7}$$

where $\mathbf{w}_v, \mathbf{w}_m \in \mathbb{R}^{B\times1\times H\times W}$ hold different values for various positions, dependent on the largest attention scores in the visible and masked regions, respectively.

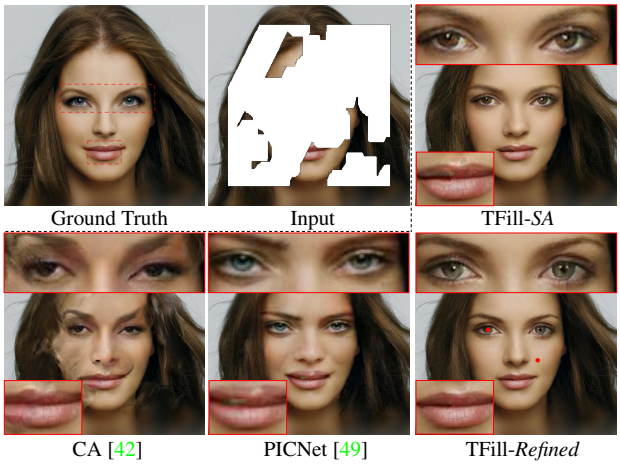


Figure 8. **Results with different attention modules** in various methods. Our attention-ware layer is able to adaptively select the features from both visible and generated content. In this example, the ratio for the two query points is $\mathbf{w}_v/\mathbf{w}_m = 0.77/0.23$ (skin) and $\mathbf{w}_v/\mathbf{w}_m = 0.08/0.92$ (eye), respectively.

Mask Type	Metric	SA	CA	SLTA	Ours-AAL
center	LPIPS	0.058	0.061	0.056	**0.053**
	FID	3.62	3.86	3.61	**3.50**
random	LPIPS	0.047	0.044	0.045	**0.041**
	FID	2.69	2.66	2.64	**2.57**

Table 3. The effect of various attention layers on FFHQ dataset. "center" denotes the center mask, "random" denotes the random regular mask and "SA" is the basic self-attention layer. These attention layers were implemented within our TFill refinement framework.

2.2.1 Results and Analysis

We ran ablations to analyze our proposed AAL by replacing it with existing contextual attention models of SA [37, 46], CA [42] and SLTA [49]. As shown in Table 3, SA showed similar performance to the coarse results in Table 1, due to the insular attention problem mentioned earlier. CA [42] performed worse on large center masks than random regular masks (even worse than the coarse results of (E) in Table 1), as it borrows context from visible regions only. When important context is not visible, *e.g.* when both eyes are missing in Fig. 8, it is unable to find the right context to copy. While PICNet [49] focuses on both visible and invisible regions, selection was done by *fixed* weights learned during training. This is also inferior, and in some cases we observed that it can have difficulty in selecting the best features for generation, especially on free-form masks. In contrast, our AAL selects features based on the largest attention scores, using weights *dynamically mapped* during inference. For instance, in Fig. 6, only the left eye was masked, and it had a large attention score to the visible right eye, resulting in a ratio of $\mathbf{w}_v/\mathbf{w}_m = 0.91/0.09$. Conversely, when two eyes were masked in Fig. 8, the attention score between the two eyes was still high, but the ratio was correctly flipped to $\mathbf{w}_v/\mathbf{w}_m = 0.08/0.92$ for the left eye.

2.2.2 Discussion on prior art

While contextual attention [42] has recently been widely applied in image completion [42, 33, 39, 41], it is fundamentally different from the attention in our transformer-based architecture — the contextual attention is used to refine visual appearance by copying high-frequency information from visible regions to masked holes, rather than capturing and modeling long-range context for content inference. In addition, our AAL focuses on automatically selecting features from both visible and generated features, instead of copying only from visible regions [42, 33, 39, 41] or selecting through fixed weights [49].

3. Further Experiments

Datasets: We evaluated our TFill with arbitrary mask types on various datasets, including CelebA-HQ [25, 19], FFHQ [20], Places2 [51], and ImageNet [32].

Metrics: As proposed in previous works [42, 49], it is not reasonable to require the completed image to be exactly the same as the original image. Hence, we only report the LPIPS [47] and the FID [12] scores in the main paper, leaving the traditional pixel- and patch-level evaluation results, *e.g.* the mean ℓ_1 loss, in Appendix B.

Implementation details: Our model is trained in two stages: **1)** the content inference network is first trained for 256^2 resolution; and **2)** the visual appearance network is then trained for 512^2 resolution. Both networks are optimized using the loss $L = L_{pixel} + L_{per} + L_{GAN}$, where L_{pixel} is the ℓ_1 reconstruction loss, L_{per} is the perceptual loss [18], and L_{GAN} is the discriminator loss [11]. More implementation details are provided in Appendix C.

3.1. Comparison with Existing Work

Here we compared with these image completion methods: **PM** [2], a classical approach; **GL** [15], the first learning-based method for arbitrary regions; **CA** [42], the first method combining learning and patch-based methods; **PICNet** [49], the first work considering multiple solutions; **HiFill** [41], the latest very high-resolution (8K) method. Our TFill introduces a transformer-based architecture for this challenging image completion problem.

Table 4 shows quantitative evaluation results on Place2 [51], in which the images were degraded by free-form masks provided in the PConv [23] testing set. The size column denotes the range of masking proportion applied to the images. We observe that our transformer-based model

Figure 9. **Qualitative comparison on various datasets with free-form masks.** Here, we show results for ImageNet [32] (top two examples) and Places2 [51] (bottom two examples). Our model generated more reasonable object and scene structures, with better visual results. Please zoom in to see the details.

	Size	GL	CA	PICNet	HiFill	TFill
LPIPS	[0.01, 0.1]	0.057	0.083	0.037	0.056	**0.027**
	(0.1, 0.2]	0.112	0.134	0.074	0.105	**0.055**
	(0.2, 0.3]	0.185	0.195	0.118	0.163	**0.092**
	(0.3, 0.4]	0.254	0.249	0.167	0.226	**0.133**
	(0.4, 0.5]	0.319	0.306	0.225	0.305	**0.180**
	(0.5, 0.6]	0.370	0.364	0.330	0.412	**0.259**
FID	[0.01, 0.1]	16.86	10.21	7.04	9.10	**5.22**
	(0.1, 0.2]	26.11	18.93	13.58	16.72	**9.67**
	(0.2, 0.3]	39.22	30.31	21.62	26.89	**15.28**
	(0.3, 0.4]	53.24	40.29	29.59	38.40	**19.99**
	(0.4, 0.5]	68.46	53.39	41.60	56.24	**25.88**
	(0.5, 0.6]	74.95	59.85	61.17	83.36	**34.58**

Table 4. Quantitative comparisons on Places2 [51] with free-form masks [23]. Without bells and whistles, TFill outperformed all traditional CNN-based models. The results are reported on 256^2 resolution, as earlier works were trained only on this scale.

improved both LPIPS and FID quite significantly over the CNN-based state-of-the-art models in all mask scales. Specifically, it achieves relative 27% and 21% improvement for LPIPS at scales of [0.01, 0.1] and (0.5, 0.6], respectively. Furthermore, our completed images form closer distributions to the real testing set, with FID scores averaging 32% relative improvement on all mask scales.

The qualitative comparisons are visualized in Figs. 8 and 9. TFill achieved superior visual results even under challenging conditions. In Fig. 8, we compare with CA and PICNet trained on CelebA-HQ dataset. Our TFill generates photorealistic high-resolution (512^2) results, even when significant semantic information is missing due to large free-form masks. Fig. 9 shows visual results on natural images that were degraded by random masks. GL and CA, while good at object removal, failed to infer shapes needed for object completion. PICNet produced multiple diverse results in which some shapes were correct but of limited quality. TFill inferred the correct shapes for even heavily masked objects in ImageNet, *e.g.* the fish even with head and tail separated by a large mask. It also outperformed all previous methods on high-resolution masked images in Places2, especially for some large masked regions. More examples and applications are presented in Appendix A.

4. Conclusion

Through our detailed analyses and experiments, we demonstrate that the transformer-based architecture has exciting potential for image completion, due to its capacity for effectively modeling connections between distant image content. Unlike recent vision transformer models that either use shallow projections or large receptive fields for token representation, our *restrictive CNN projection* provides the necessary separation between explicit attention modeling and implicit RF correlation that leads to substantial im-

provement in results. We also introduced a novel attention-aware layer that adaptively balances the attention for visible and masked regions, further improving the completed image quality.

References

[1] 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. 1

[2] Connelly Barnes, Eli Shechtman, Adam Finkelstein, and Dan B Goldman. Patchmatch: A randomized correspondence algorithm for structural image editing. *ACM Transactions on Graphics (ToG)*, 28:24, 2009. 1, 7, 8

[3] 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. ACM Press/Addison-Wesley Publishing Co., 2000. 1

[4] 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. 2, 3, 5

[5] 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. *arXiv preprint arXiv:2012.00364*, 2020. 3, 5

[6] Mark Chen, Alec Radford, Rewon Child, Jeffrey Wu, Heewoo Jun, David Luan, and Ilya Sutskever. Generative pretraining from pixels. In *International Conference on Machine Learning*, pages 1691–1703. PMLR, 2020. 2, 3, 4, 5, 6

[7] Antonio Criminisi, Patrick Pérez, and Kentaro Toyama. Region filling and object removal by exemplar-based image inpainting. *IEEE Transactions on image processing*, 13(9):1200–1212, 2004. 1

[8] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018. 2, 5

[9] 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. *arXiv preprint arXiv:2010.11929*, 2020. 3, 4, 5

[10] Patrick Esser, Robin Rombach, and Björn Ommer. Taming transformers for high-resolution image synthesis, 2020. 2, 3, 4, 5, 6, 22

[11] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014. 5, 7

[12] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. 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. 4, 7

[13] Xun Huang, Ming-Yu Liu, Serge Belongie, and Jan Kautz. Multimodal unsupervised image-to-image translation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 172–189, 2018. 4

[14] Drew A Hudson and C. Lawrence Zitnick. Generative adversarial transformers. *arXiv preprint*, 2021. 5

[15] Satoshi Iizuka, Edgar Simo-Serra, and Hiroshi Ishikawa. Globally and locally consistent image completion. *ACM Transactions on Graphics (ToG)*, 36(4):1–14, 2017. 1, 5, 7, 8, 15, 16, 20

[16] Yifan Jiang, Shiyu Chang, and Zhangyang Wang. Transgan: Two transformers can make one strong gan. *arXiv preprint arXiv:2102.07074*, 2021. 5

[17] Youngjoo Jo and Jongyoul Park. Sc-fegan: face editing generative adversarial network with user's sketch and color. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1745–1753, 2019. 5

[18] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *European conference on computer vision*, pages 694–711. Springer, 2016. 7

[19] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. In *International Conference on Learning Representations*, 2018. 4, 7, 13, 14, 17

[20] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4401–4410, 2019. 4, 7, 13, 14, 17

[21] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8110–8119, 2020. 4, 22

[22] Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, 2014. 5

[23] 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. 1, 4, 5, 7, 8, 13, 20

[24] Hongyu Liu, Bin Jiang, Yibing Song, Wei Huang, and Chao Yang. Rethinking image inpainting via a mutual encoder-decoder with feature equalizations. In *Proceedings of the European Conference on Computer Vision*, 2020. 1, 3, 4, 5, 20

[25] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, December 2015. 4, 7, 13, 14, 17

[26] Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784*, 2014. 5

[27] Kamyar Nazeri, Eric Ng, Tony Joseph, Faisal Qureshi, and Mehran Ebrahimi. Edgeconnect: Structure guided image inpainting using edge prediction. In *The IEEE International Conference on Computer Vision (ICCV) Workshops*, Oct 2019. 1, 5

[28] 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*, pages 2536–2544, 2016. 1, 5

[29] Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. Improving language understanding by generative pre-training. Technical report, OpenAI, 2018. 2

[30] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019. 2, 5

[31] Ali Razavi, Aaron van den Oord, and Oriol Vinyals. Generating diverse high-fidelity images with vq-vae-2. *arXiv preprint arXiv:1906.00446*, 2019. 5

[32] 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. 7, 8, 12, 13, 15, 17, 18

[33] Yuhang Song, Chao Yang, Zhe Lin, Xiaofeng Liu, Qin Huang, Hao Li, and C-C Jay Kuo. Contextual-based image inpainting: Infer, match, and translate. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 3–19, 2018. 5, 7

[34] Yuhang Song, Chao Yang, Yeji Shen, Peng Wang, Qin Huang, and C-C Jay Kuo. Spg-net: Segmentation prediction and guidance network for image inpainting. *arXiv preprint arXiv:1805.03356*, 2018. 5

[35] Antonio Torralba, Rob Fergus, and William T Freeman. 80 million tiny images: A large data set for nonparametric object and scene recognition. *IEEE transactions on pattern analysis and machine intelligence*, 30(11):1958–1970, 2008. 5

[36] Aaron van den Oord, Oriol Vinyals, and Koray Kavukcuoglu. Neural discrete representation learning. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pages 6309–6318, 2017. 5

[37] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. 2, 3, 5, 7, 21

[38] Bichen Wu, Chenfeng Xu, Xiaoliang Dai, Alvin Wan, Peizhao Zhang, Masayoshi Tomizuka, Kurt Keutzer, and Peter Vajda. Visual transformers: Token-based image representation and processing for computer vision. *arXiv preprint arXiv:2006.03677*, 2020. 3, 5

[39] Zhaoyi Yan, Xiaoming Li, Mu Li, Wangmeng Zuo, and Shiguang Shan. Shift-net: Image inpainting via deep feature rearrangement. In *Proceedings of the European conference on computer vision (ECCV)*, pages 1–17, 2018. 7

[40] Chao Yang, Xin Lu, Zhe Lin, Eli Shechtman, Oliver Wang, and Hao Li. High-resolution image inpainting using multi-scale neural patch synthesis. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6721–6729, 2017. 5

[41] Zili Yi, Qiang Tang, Shekoofeh Azizi, Daesik Jang, and Zhan Xu. Contextual residual aggregation for ultra high-resolution image inpainting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7508–7517, 2020. 1, 2, 4, 5, 7, 8, 15, 16, 20

[42] Jiahui Yu, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and Thomas S Huang. Generative image inpainting with contextual attention. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5505–5514, 2018. 1, 2, 3, 4, 5, 6, 7, 8, 14, 15, 16, 20

[43] 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*, pages 4471–4480, 2019. 5

[44] Yu Zeng, Zhe Lin, Huchuan Lu, and Vishal M Patel. Image inpainting with contextual reconstruction loss. *arXiv preprint arXiv:2011.12836*, 2020. 1, 5

[45] Yu Zeng, Zhe Lin, Jimei Yang, Jianming Zhang, Eli Shechtman, and Huchuan Lu. High-resolution image inpainting with iterative confidence feedback and guided upsampling. In *European Conference on Computer Vision*, pages 1–17. Springer, 2020. 5

[46] Han Zhang, Ian Goodfellow, Dimitris Metaxas, and Augustus Odena. Self-attention generative adversarial networks. In *International conference on machine learning*, pages 7354–7363. PMLR, 2019. 6, 7

[47] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, 2018. 4, 7, 12, 13, 16, 19

[48] Lei Zhao, Qihang Mo, Sihuan Lin, Zhizhong Wang, Zhiwen Zuo, Haibo Chen, Wei Xing, and Dongming Lu. Uctgan: Diverse image inpainting based on unsupervised cross-space translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5741–5750, 2020. 5

[49] Chuanxia Zheng, Tat-Jen Cham, and Jianfei Cai. Pluralistic image completion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 1, 2, 3, 4, 5, 6, 7, 8, 14, 15, 16, 20

[50] Sixiao Zheng, Jiachen Lu, Hengshuang Zhao, Xiatian Zhu, Zekun Luo, Yabiao Wang, Yanwei Fu, Jianfeng Feng, Tao Xiang, Philip H.S. Torr, and Li Zhang. Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers, 2020. 2, 3, 5

[51] Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. Places: A 10 million image database for scene recognition. *IEEE transactions on pattern analysis and machine intelligence*, 40(6):1452–1464, 2018. 7, 8, 20

[52] Jun-Yan Zhu, Richard Zhang, Deepak Pathak, Trevor Darrell, Alexei A Efros, Oliver Wang, and Eli Shechtman. Toward multimodal image-to-image translation. In *Advances in neural information processing systems*, pages 465–476, 2017. 4

[53] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. *arXiv preprint arXiv:2010.04159*, 2020. 2, 3, 5

TFill: Image Completion via a Transformer-Based Architecture Supplementary Material

The supplementary material for our work *TFill: Image Completion via a Transformer-Based Architecture* is organized as follows: First, in Section A, we present additional visual results, including results of TFill-*Coarse* on ImageNet [32] and Places2 datasets [47], qualitative comparison to the state-of-the-art models on various datasets, and some examples for free-form editing on high-resolution images. Next, extending the quantitative comparisons of Tables 1 and 4 in the main paper, Section B presents additional evaluation results under the traditional pixel-level and patch-level image quality metrics. Finally, we discuss more technological details of our TFill model in Section C.

A. Additional Examples

A.1. Additional Results for TFill-*Coarse*

In Figs. A.1 and A.2, we show more examples on ImageNet [32] and Places2 [47] dataset images that were degraded by large center masks. This is an extension of Fig. 3 in the main paper.

Here, all examples shown are chosen from the corresponding testing set. In Fig. A.1, we show more examples for object completion, such as the various items and animals on the top half. In Fig. A.2, we display the completed images for various natural scenes. These examples are good evidence that our TFill model is suitable for both *foreground* object completion and *background* scene completion, where it can synthesize semantically consistent content with visually realistic appearance based on the presented visible pixels.

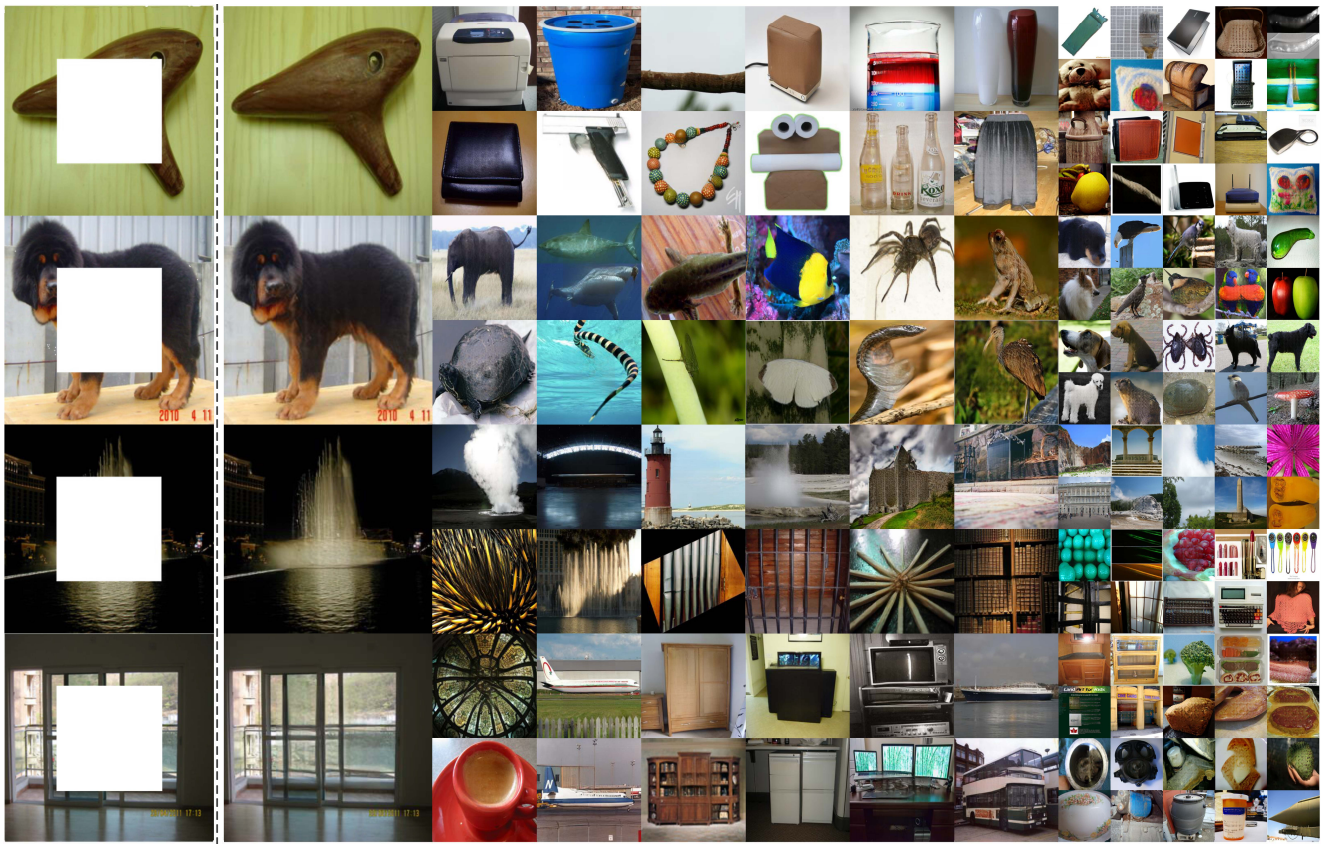


Figure A.1. **More completion results of our method (config E) on ImageNet datasets [32].** All images come from the corresponding testing set that were **degraded by center masks**. Here, we show results for various categories, such as commodity, animal, plant, natural scene, building, food, furniture and so on. The center masked example inputs are shown on the left. Our model is able to complete both object shape and background scene via a transformer-based architecture to correctly bridge the visible tokens.

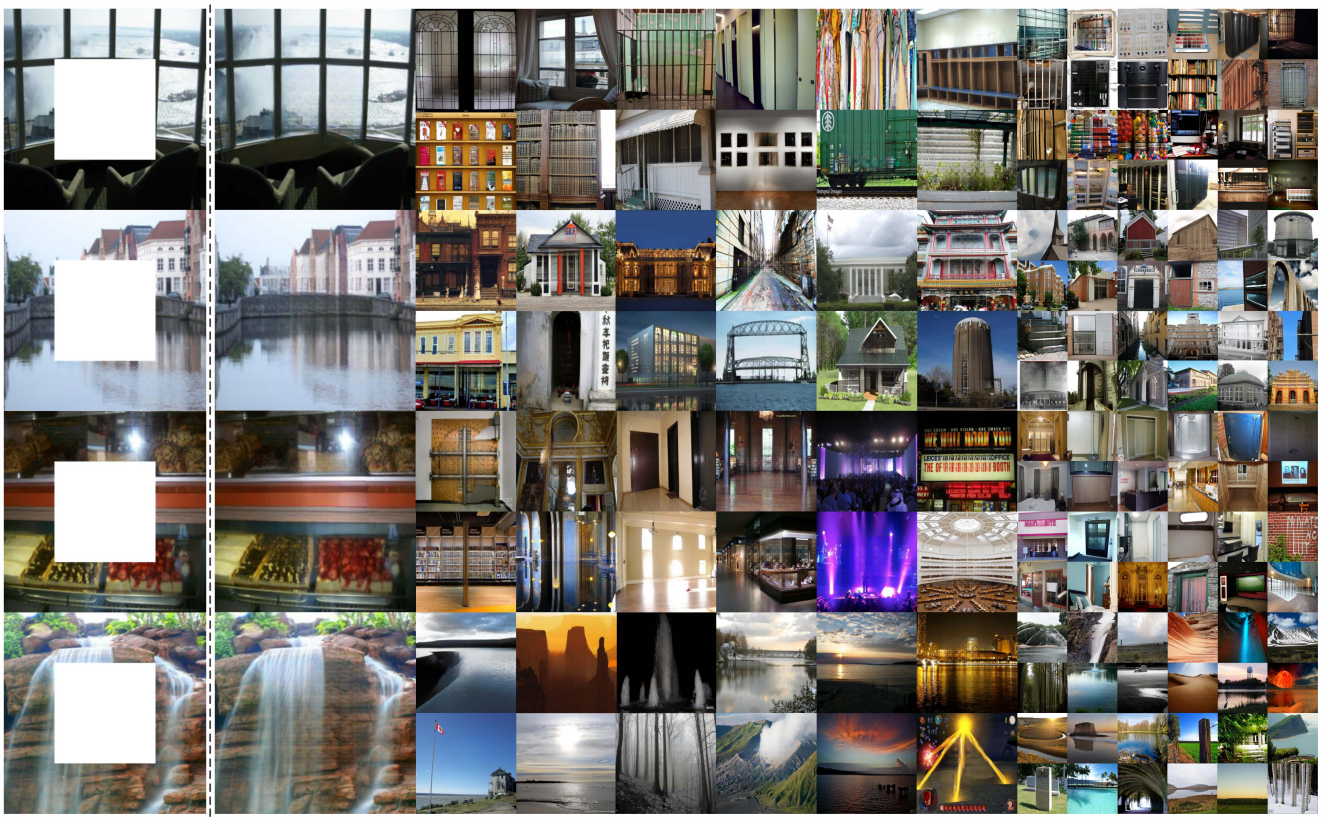


Figure A.2. **More completion results of our method (config E) on Places2 datasets [47].** All images come from the corresponding testing set that were **degraded by center masks**. Here, we show results for varied scene categories. The center masked example inputs are shown on the left. Our model is able to complete both object shape and background scene via a transformer-based architecture to correctly bridge the visible tokens.

A.2. Additional Comparisons

In Figs. A.3, A.4 and A.5, we show additional comparison results on various datasets with free-form masks provided in PConv [23]. This is an extension of Figs. 8 and 9 in the main paper. Here, our results on CelebA-HQ [25, 19] and FFHQ [20] testing set are reported for 512^2 resolution. On the other hand, the results on ImageNet [32] and Places2 [47] are reported for higher resolution images that were resized such that the short side is 512 pixels, with the long side in multiples of $2^5 = 32$, *e.g.* $640 = 32 \times 20$. The size variability is possible due to our fully convolutional encoder-decoder network structures. The 32-base scale is required because our refinement network downsamples the images 5 times with step 2.

As can be seen from these results, our TFill model filled appropriate semantic content with visually realistic appearance into the various masks. For instance, in the third row of Fig. A.3, even with an extensive mask on an obliquely angled face, it was able to generate high-quality results. It achieved good results even under challenging conditions for various objects (Fig. A.4) and scenes (Fig. A.5).

A.3. Free-Form Editing on High-Resolution Images

In Figs. A.6, A.7, A.8 and A.9, we show qualitative results for free-form image masking on various higher resolution datasets.

In Fig. A.6, we show some examples for face editing at 512^2 resolution. For conventional object removal, *e.g.* watermark removal, our TFill addresses them easily. Furthermore, our TFill can handle more extensive face editing, such as removing substantial facial hair and changing mouth expressions.

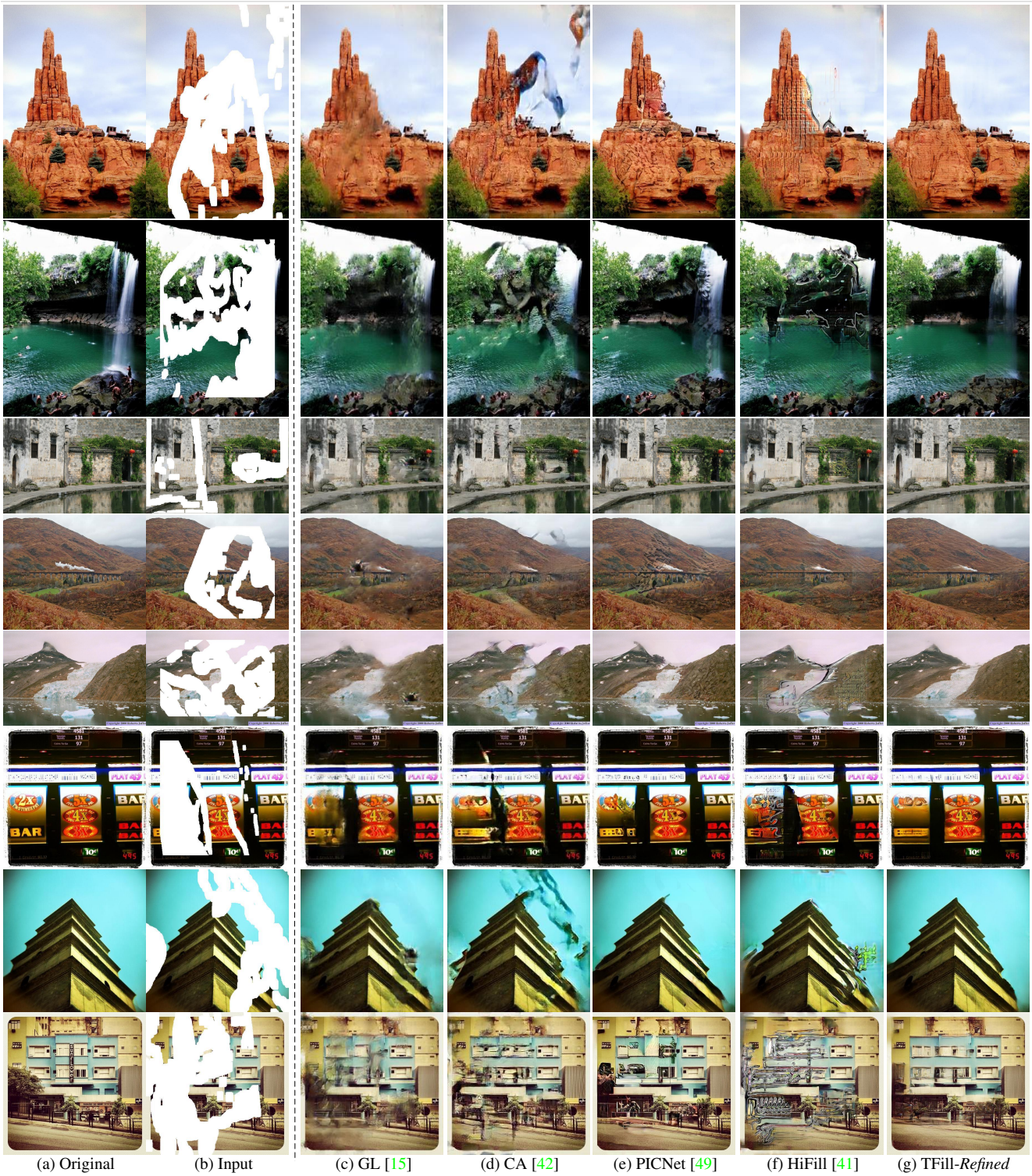
In Figs. A.7, A.8 and A.9, we show some examples of editing images of natural / outdoor scenes, with object removal being the main task, as it is the main practice for image inpainting. Here, we enforce the input image size to be multiples of 32, *e.g.* 960×640 and provide the high-resolution results on the corresponding image size. As we can see, our TFill-*Refined* model is able to handle high-resolution images for object removal in traditional image inpainting task.

(a) Original (b) Input (c) CA [42] (d) PICNet [49] (e) TFill-*Coarse* (f) TFill-*Refined*

Figure A.3. **Additional results on CelebA-HQ [25, 19] and FFHQ [20] testing set among CA [42], PICNet [49] and Ours.** Our results are reported for 512^2 resolution. While PICNet [49] works well for frontal facing faces, it may generate more uncanny faces with mismatched features at larger angles, *e.g*. the examples in third and last row. In contrast, our model generated consistent facial features with photorealistic appearance for various faces angles. Zoom in to see the details.

(a) Original (b) Input (c) GL [15] (d) CA [42] (e) PICNet [49] (f) HiFill [41] (g) TFill-*Refined*

Figure A.4. **Additional results on ImageNet [32] testing set among GL [15], CA [42], PICNet [49], HiFill [41] and Ours.** Our results are evaluated in higher resolution, with the short side at 512 pixels and the long side at multiples of 2^5, *e.g.* 640. Our TFill model generated better visual results even under very challenging situations, *e.g.* the heavily masked chicken in the second last row.



(a) Original (b) Input (c) GL [15] (d) CA [42] (e) PICNet [49] (f) HiFill [41] (g) TFill-*Refined*

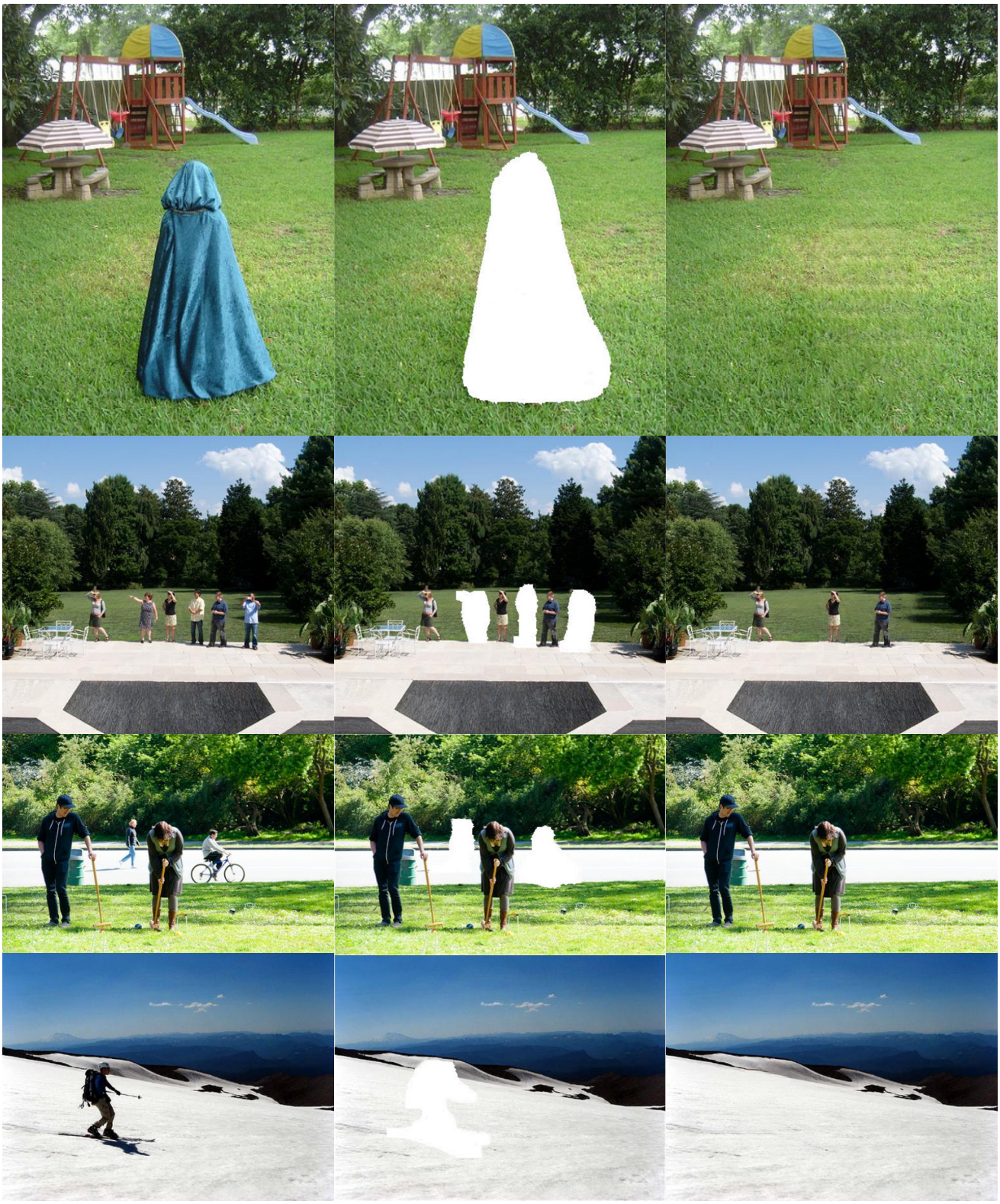
Figure A.5. **Additional results on Places2 [47] testing set among GL [15], CA [42], PICNet [49], HiFill [41] and Ours.** Our results are evaluated in higher resolution, with the short side at 512 pixels and the long side at multiples of 2^5, *e.g.* 640.

(a) Original (b) Edited Output (a) Original (b) Edited Output

Figure A.6. **Additional results on CelebA-HQ [25, 19] and FFHQ [20] testing set for free-form mask editing.** All results are reported at 512^2 resolution. Our model works well for traditional object removal, such as removing eyeglasses and watermarks. Furthermore, we provide examples of more substantial modifications, *e.g*. facial hair removal, and expression modification in the last row.

(a) Original (b) Input (c) Edited Output

Figure A.7. **Additional free-form editing results on ImageNet [32].** The original image size in ImageNet is 500×375. Here, we resized slightly to 512×384 for image completion. We highlight the generated content, which has consistent textures to those in the visible regions.



(a) Original (b) Input (c) Edited Output

Figure A.8. **Additional results on ImageNet [32] testing set for free-form editing.** Here, we enforce the input image size to be multiples of 32, *e.g*. 960×640 and provide the high-resolution results on the corresponding image size. Zoom in to see the completed details.

(a) Original (b) Input (c) Edited Output

Figure A.9. **Additional results on Places2 [47] testing set for free-form editing.** Here, we enforce the input image size to be multiples of 32, *e.g.* 960×640 and provide the high-resolution results on the corresponding image size. Zoom in to see the completed details.

B. Additional Quantitative Results

We further report quantitative results using traditional pixel-level and patch-level image quality evaluation metrics.

	Method	CelebA-HQ			FFHQ		
		$\ell_1$loss ↓	SSIM↑	PSNR↑	$\ell_1$loss ↓	SSIM↑	PSNR↑
	CA [42]	0.0310	0.8201	23.5667	0.0337	0.8099	22.7745
	PICNet [49]	0.0209	0.8668	24.6860	0.0241	0.8547	24.3430
	MEDFE [24]	0.0208	0.8691	24.4733	-	-	-
A	Traditional *Conv*	0.0199	0.8693	24.5800	0.0241	0.8559	24.2271
B	+ Attention in G	0.0196	0.8717	24.6512	0.0236	0.8607	24.4384
C	+ Restrictive *Conv*	0.0191	0.8738	24.8067	0.0220	0.8681	24.9280
D	+ Transformer	0.0189	0.8749	24.9467	0.0197	0.8751	25.1002
E	+ Masked Attention	0.0183	0.8802	25.2510	0.0188	0.8765	25.1204
F	+ Refine Network	**0.0180**	**0.8821**	**25.4220**	**0.0184**	**0.8778**	**25.2061**

Table B.1. Quantitative results for traditional pixel-level and patch-level metrics on center masked images.

Table B.1 provides a comparison of our results to state-of-the-art CNN-based models, as well as various alternative configurations for our design, on the center masked face testing set. This is an extension of Table 1 in the main paper. All images were normalized to the range [0,1] for quantitative evaluation. While there is no necessity to strongly encourage the completed images to be the same as the original ground-truth images, our TFill model nonetheless achieved better performance on these metrics too, including ℓ_1 loss, structure similarity index (SSIM) and peak signal-to-noise ration (PSNR), suggesting that our TFill model is more capable of generating closer content to the original unmasked images.

	Size	GL [15]	CA [42]	PICNet [49]	HiFill [41]	TFill
$\ell_1$loss†	[0.01, 0.1]	0.0233	0.0241	0.0097	0.0195	**0.0093**
	(0.1, 0.2]	0.0346	0.0338	0.0164	0.0282	**0.0153**
	(0.2, 0.3]	0.0500	0.0471	0.0249	0.0390	**0.0231**
	(0.3, 0.4]	0.0659	0.0612	0.0348	0.0513	**0.0322**
	(0.4, 0.5]	0.0808	0.0753	0.0456	0.0657	**0.0422**
	(0.5, 0.6]	0.0945	0.0925	0.0641	0.0885	**0.0591**
SSIM*	[0.01, 0.1]	0.9150	0.9079	0.9634	0.9245	**0.9695**
	(0.1, 0.2]	0.8526	0.8447	0.9137	0.8603	**0.9253**
	(0.2, 0.3]	0.7672	0.7652	0.8520	0.7838	**0.8686**
	(0.3, 0.4]	0.6823	0.6906	0.7850	0.7057	**0.8063**
	(0.4, 0.5]	0.5987	0.6133	0.7119	0.6193	**0.7391**
	(0.5, 0.6]	0.5185	0.5322	0.6077	0.5137	**0.6428**
PSNR*	[0.01, 0.1]	28.4151	26.8452	32.2579	28.3955	**33.0585**
	(0.1, 0.2]	24.4074	23.1766	27.3320	24.5495	**28.0670**
	(0.2, 0.3]	21.3296	20.4427	24.4423	22.0604	**25.0951**
	(0.3, 0.4]	19.1118	18.6337	22.3238	20.1451	**22.8942**
	(0.4, 0.5]	17.5594	17.2978	20.7146	18.4715	**21.2200**
	(0.5, 0.6]	16.4831	16.0824	18.7234	16.4998	**19.1040**

Table B.2. Quantitative comparisons on Places2 [51] with free-form masks [23]. †Lower is better. *Higher is better. Without bells and whistles, TFill outperformed all traditional CNN-based models.

Table B.2 provides a comparison of our results to state-of-the-art methods on the Places2 [51] testing set with free-form masks [23]. This is an extension of Table 4 in the main paper. As we can see in Fig. A.5, while our TFill model does *not* generate the same content as the original unmasked images, it filled the masked holes with semantically appropriate content of consistent realistic appearance. There were no obvious artifacts when the completed pixels were recomposed with the original visible pixels, resulting in quite a significant improvement in image quality.

C. Experiment Details

Here we first present the novel layers and loss functions used to train our model, followed by the training details.

C.1. Multihead *Masked* Self-Attention

Our transformer encoder is built on the standard **qkv** `self-attention` (SA) [37] with a learned position embedding in each layer. Given an input sequence $\mathbf{z} \in \mathbb{R}^{N \times C}$, we first calculate the pairwise similarity $\mathbf{A}$ between each two elements as follows:

$$[\mathbf{q}, \mathbf{k}, \mathbf{v}] = \mathbf{W}_{qkv}\mathbf{z} \tag{C.1}$$

$$\mathbf{A} = \texttt{softmax}(\mathbf{q}\mathbf{k}^{\top}/\sqrt{C_h}) \tag{C.2}$$

where $\mathbf{W}_{qkv} \in \mathbb{R}^{C \times 3C_h}$ is the learned parameter to refine the features $\mathbf{z}$ for the query $\mathbf{q}$, the key $\mathbf{k}$ and the value $\mathbf{v}$. $\mathbf{A} \in \mathbb{R}^{N \times N}$ is the dot similarity of N tokens, which is scaled by the square root of feature dimension C_h. Then, we compute a weighted sum over all values $\mathbf{v}$ via:

$$\mathrm{SA}(\mathbf{z}) = \mathbf{A}\mathbf{v} \tag{C.3}$$

where the value z in the sequence is connected through their learned similarity A, rather than purely depending on a fixed learned weight w.

The multihead `self-attention` (MSA) is an extention of SA, in which H heads are run in parallel to get multiple attention scores and the corresponding projected results. Then we get the following function:

$$\mathrm{MSA}(\mathbf{z}) = [\mathrm{SA}_1(\mathbf{z}); \mathrm{SA}_2(\mathbf{z}); \ldots; \mathrm{SA}_h(\mathbf{z})] \tag{C.4}$$

To encourage the model to *bias* to the important visible values, we further modify the MSA with a *masked* self-attention layer, in which a masked weight is applied to scale the attention score $\mathbf{A}$. Given a feature $\mathbf{x}$ and the corresponding mask $\mathbf{m}$ (1 denotes visible pixel and 0 is masked pixel). The original partial convolution operation is operated as:

$$x' = \begin{cases} \mathbf{W}_p(\mathbf{x}_p \bigodot \mathbf{m}_p)\frac{1}{\sum(\mathbf{m}_p)} + b, & \text{if } \sum(\mathbf{m}_p) > 0 \\ 0, & \text{otherwise} \end{cases} \tag{C.5}$$

$$m' = \begin{cases} 1, & \text{if } \sum(\mathbf{m}_p) > 0 \\ 0, & \text{otherwise} \end{cases} \tag{C.6}$$

where $\mathbf{W}_p$ contain the convolution filter weights, b is the corresponding bias, while $\mathbf{x}_p$ and $\mathbf{m}_p$ are the feature values and mask values in the current convolution window (*e.g.* 2×2 in our *restrictive CNN*), respectively. Here, we replace the m' as a float value:

$$m' = \frac{\sum(\mathbf{m}_p)}{S} \tag{C.7}$$

where S is the size of each convolution filter, 2×2 used in our *restrictive CNN*. To do this, each token only extracts the visible information. What's more, the final m for each token denotes the percentage of valid values in each token under a small RF. Then, for each sequence $\mathbf{z} \in \mathbb{R}^{N \times C}$, we obtain a corresponding masked weight $\mathbf{m} \in \mathbb{R}^{N \times 1}$ by flattening the updated mask. Finally, we update the original attention score by multiplying with the repeated masked weight $\mathbf{m} \in \mathbb{R}^{N \times 1}$:

$$\mathbf{A}_m = \mathbf{A} \bigodot \mathbf{m}_r \tag{C.8}$$

where $\mathbf{m}_r \in \mathbb{R}^{N \times N}$ is the extension of masked weight $\mathbf{m} \in \mathbb{R}^{N \times 1}$ in the final dimension.

C.2. Loss Functions

Our work focuses on exploiting the *token representation* in the visual transformer architecture. We do *not* modify the discriminator architecture or design the loss function in any way. Both TFill-*Coarse* and TFill-*Refined* is trained with loss $L = L_{pixel} + L_{per} + L_{GAN}$. In particular, each loss is given as:

$$L_{pixel} = ||\mathbf{I}_{gt} - \mathbf{I}_g||_1 \tag{C.9}$$

$$L_{per} = ||\Phi_n(\mathbf{I}_{gt}) - \Phi_n(\mathbf{I}_g)||_1 \tag{C.10}$$

$$L_{GAN} = log(1 + \exp(-D(\mathbf{I}_g))) \tag{C.11}$$

where $\mathbf{I}_g$ and $\mathbf{I}_{gt}$ is the generated image and original ground truth image, respectively. Φ_n is the activation map of the nth selected layer in VGG. D is the discriminator and here we show only the generator loss for the generative adversarial traning.

C.3. Training Details

Our model was trained on two NVIDIA A100 GPUs in two stages: **1)** the content inference network was first trained with 256^2 resolution with batch size of 96; **2)** the visual appearance network was then trained with 512^2 resolution with batch size of 24. Both networks were optimized using the loss $L = L_{pixel} + L_{per} + L_{GAN}$. The design of the encoder-decoder backbone follows the architecture presented in [10]. For the discriminator, we used the architecture of StyleGANv2 [21].