Image Inpainting on the Sketch-Pencil Domain with Vision Transformers

Jose Luis Flores Campana, Luís Gustavo Lorgus Decker, Marcos Roberto e Souza,
Helena de Almeida Maia and Helio Pedrini
Institute of Computing, University of Campinas, Campinas, SP, 13083-852, Brazil

Keywords: Image Inpainting, Sketch-Pencil, Image Processing, Transformers.

Abstract: Image inpainting aims to realistically fill missing regions in images, which requires both structural and textural understanding. Traditionally, methods in the literature have employed Convolutional Neural Networks (CNN), especially Generative Adversarial Networks (GAN), to restore missing regions in a coherent and reliable manner. However, CNNs' limited receptive fields can sometimes result in unreliable outcomes due to their inability to capture the broader context of the image. Transformer-based models, on the other hand, can learn long-range dependencies through self-attention mechanisms. In order to generate more consistent results, some approaches have further incorporated auxiliary information to guide the model's understanding of structural information. In this work, we propose a new method for image inpainting that uses sketch-pencil information to guide the restoration of structural, as well as textural elements. Unlike previous works that employ edges, lines, or segmentation maps, we leverage the sketch-pencil domain and the capabilities of Transformers to learn long-range dependencies to properly match structural and textural information, resulting in more consistent results. Experimental results show the effectiveness of our approach, demonstrating either superior or competitive performance when compared to existing methods, especially in scenarios involving complex images and large missing areas.

1 INTRODUCTION

Image inpainting is a task that aims to fill unknown regions of a damaged image. Over the years, the significance of image inpainting has grown considerably, as it has found applications in various real-world scenarios such as object removal (Zeng et al., 2020), photo restoration (Wan et al., 2020), and image editing (Yu et al., 2019). The major challenge of this task lies in the necessity to properly restore missing regions with content that is both visually realistic and semantically plausible. To achieve this, the restoration process must encompass not only the structural aspects but also the textural nuances of the missing regions, ensuring coherence between them in the global context of the image.

Several approaches have been proposed to pursue realism in image inpainting (Ghorai et al., 2019; Gamini and Kumar, 2019; Liu et al., 2020; Yu et al., 2018; Nazeri et al., 2019; Dong et al., 2022; Liao et al., 2021; Suvorov et al., 2022; Yang et al., 2020). Classical approaches focus on restoring missing regions using diffusion-based and patch-based models (Ghorai et al., 2019; Gamini and Kumar, 2019). However, these approaches suffer from restoring plausible structures and realistic textures by ignoring the global context of the image.

More recently, approaches based on convolutional neural networks (CNN) and generative adversarial networks (GAN) have emerged to address these problems. However, these approaches still present some challenges:

(i) limited receptive fields: this limitation raises the difficulty of achieving the restoration of semantically coherent structures, due to the difficulties of CNNs to capture the broader context of the image;

(ii) complex models: handling large masks can lead to creating models that manage to capture the global context of the image and produce high-quality results. However, this model requires the use of multiple components, which transforms the model into a more complex one and requires more parameters/time to learn;

(iii) incomplete structures: CNN-based models can produce incomplete results, due to a lack of understanding of structural information, such as edges or lines, that guide the coherent restoration of the image.

Campana, J., Decker, L., Souza, M., Maia, H. and Pedrini, H.
Image Inpainting on the Sketch-Pencil Domain with Vision Transformers.
DOI: 10.5220/0012363500003660

In *Proceedings of the 19th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2024) - Volume 3: VISAPP*, pages 122-132
ISBN: 978-989-758-679-8; ISSN: 2184-4321

In response, some methods employ various convolution and upsampling operators (Liu et al., 2020) or even dilated convolutions (Yu et al., 2018) to restore the global context. Unfortunately, these strategies often result in the creation of duplicate patterns or blurry artifacts. Other methods have used strategies such as wavelets (Yu et al., 2021) or contextual attention (Yu et al., 2018) to capture global context without the need for multiple convolution operators. However, these methods can lead to the generation of artifacts for complex patterns. Some others have employed auxiliary information such as edges (Nazeri et al., 2019), lines (Dong et al., 2022), segmentation maps (Liao et al., 2021), gradients (Yang et al., 2020) to guide the structural or texture restoration of the inpainting model. Nevertheless, applying semantically incorrect auxiliary information to image inpainting models can lead to inconsistent results.

Therefore, these challenges motivate the creation of a model capable of inferring auxiliary structural and texture information consistently to guide our inpainting model to restore the damaged regions in a semantically coherent and visually detailed way. More specifically, we employ a Transformer-based model to learn the auxiliary information since the Transformers have the ability to model long-range dependencies, compared to the limited receptive fields of CNNs. In addition, we use new auxiliary information extracted from the sketch-pencil domain (also known as hand-drawn sketch or pencil drawing). This domain helps us to effectively encapsulate the structural information, as well as infer better texture content to guide the inpainting model to obtain coherent and detailed results.

In our proposed method, a damaged image is converted into the sketch-pencil domain. This feeds a Transformer Structure Texture Restoration (TSTR) model based on the architecture proposed by Campana et al. (2022). Our TSTR model employs the patch partitioning strategy to capture relevant structural information such as edges, and texture content from the context global image to restore the missing regions. Subsequently, a Efficient Transformer Inpainting (ETI) model (Campana et al., 2022) is utilized to predict the structure and texture guided from the restored sketch-pencil image, generating an inpainted image from (i) the restored sketch-pencil image and (ii) the damaged image (Fig 1). In addition, both models employ the patch-self attention strategy (Campana et al., 2022) to reduce memory consumption and computational power compared to the global-self attention approach (Dosovitskiy et al., 2021).

The main contribution of this work is an image inpainting method based on Vision Transformers that use auxiliary information extracted from the sketch-pencil domain to guide a semantically reliable and visually realistic restoration. We conducted experiments on Places2, PSV, and CelebA datasets. Our results outperformed state-of-the-art competitors in perceptual measures, namely FID and LPIPS, for the two latter datasets and achieved competitive results on the former.

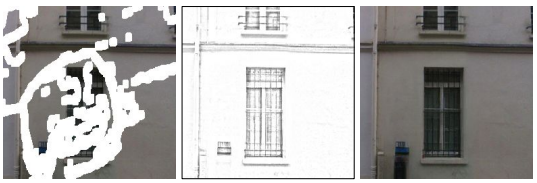

Figure 1: Approach Overview. Left: Input image with missing regions. The missing regions are represented in the white pixels. Center: Inpainted sketch-pencil image. The input image is converted into the sketch-pencil domain, resulting in a damaged sketch-pencil image. Our sketch model inpaints the damaged sketch-pencil regions to obtain the inpainted sketch-pencil image. Right: Image inpainting results of the proposed inpainting model. The inpainted sketch-pencil image combined with the damaged image is employed to restore the missing region and obtain the inpainted image.

This text is organized as follows. Section 2 presents recent image inpainting methods relevant to this work, including those that utilize auxiliary methods to guide the image inpainting task and those based on vision transformers. Section 3 thoroughly describes our approach using two vision transformers for image inpainting. Section 4 presents our results, along with information about datasets and training implementation. We conducted quantitative and qualitative comparisons with state-of-the-art methods. Section 5 provides an ablation study of the images in the sketch-pencil domain. Finally, we present our conclusions in Section 6.

2 BACKGROUND

Image Inpainting based on Auxiliary Information. Some works used auxiliary information sources to deal with difficult situations in image inpainting, such as large masks and complex elements. This additional information may include edges, lines, gradients, or segmentation maps, which guide the inpainting process. For example, Nazeri et al. (2019) proposed a two-step network called EdgeConnect. In the first step, it fills an edge map computed by the Canny edge detector. Then, it predicts the inpainted image using both the restored edge map and the original in-

put image. In another approach, Dong et al. (2022) presented a method called ZITS, which consists of two main components. The first component, Transformer Structural Restoration, completes the edges and lines predicted from the damaged image. The second component, Fourier Texture Restoration, uses the inpainted edges and lines, along with the damaged image, to achieve the final inpainted image. In this work, we employ auxiliary information extracted from the sketch-pencil domain, which brings more structural consistent information as well as texture details compared to other auxiliary information, such as edges.

Image Inpainting based on Transformers. Vision Transformers have gained great popularity in computer vision due to their exceptional performance across a range of tasks, including image classification (Dosovitskiy et al., 2021) and semantic segmentation (Strudel et al., 2021), among others. In the context of image inpainting, transformers have emerged as powerful alternatives to methods based on Convolutional and Generative Adversarial Networks, primarily due to their self-attention mechanism, which enables the capture of global context. Li et al. (2022) introduced a Transformer-based approach that employs dynamic masks to effectively handle large masks. Meanwhile, Cao et al. (2022) proposed a method based on the Masked Autoencoders (MAE) (He et al., 2022), where features extracted from the MAE model are utilized in the Attention-based CNN Restoration (ACR) model to learn the intricacies of reconstructing missing regions. Campana et al. (2022) proposed a model based on Transformers that use different patch sizes and a variable number of heads in the self-attention mechanism to capture the global context of the image efficiently in training and inference time. Based on the latter, here we propose the use of Transformers in the sketch-pencil domain to infer the structural information leveraging the global context image. This information helps our inpainting model to generate coherent and detailed results.

3 PROPOSED METHOD

This section describes the main steps of our sketch-pencil image inpainting method using Vision Transformers.

Overview. Figure 2 illustrates the proposed pipeline. Our Transformer Structure Texture Restoration (TSTR) computes the inpainted sketch-pencil image $\hat{I}_s = \text{TSTR}(I_d, I_s, M)$ (Section 3.1) from the inputs: the damaged image I_d, the damaged sketch-pencil image I_s, and a binary mask M. The Efficient Transformer Inpainting calculates the inpainted image $I_{out} = \text{ETI}(I_d, \hat{I}_s)$ (Section 3.2) guided from the inpainted sketch-pencil image and taking as input the damaged image. TSTR and ETI are, respectively, the Transformer Structure Texture Restoration and the Efficient Transformer Inpainting models.

3.1 Transformer Structure Texture Restoration

By leveraging the inherent capacity of Transformers to capture global context information (Dosovitskiy et al., 2021), we adopted the work proposed by Campana et al. (2022, 2023) as our baseline. The Transformer Structure Texture Restoration framework is employed to guide the restoration of the inpainting model by improving the structural and texture information in the sketch-pencil domain.

A sketch image is an artistic visual effect that resembles a hand-drawn sketch or a pencil drawing (Qiu et al., 2019). This effect can be achieved through various techniques in image processing and computer vision fields. Figure 3 presents the conversion process giving an input image into an image in the sketch-pencil domain or a sketch-pencil image[1].

How Much Sketch-Pencil Information Is Needed? We explored the most adequate amount of edges, lines, and texture information that our sketch-pencil image must have for better final inpainting. This amount is controlled by the Gaussian filter parameter (δ).

Figure 4 shows the impact of δ in the sketch-pencil image. We chose $\delta = 21$ since it produces darker and thicker edges and lines, making them suitable for shading and adding texture to our inpainting model. This appropriate selection of δ value enabled us to optimize the guidance provided by the sketch-pencil domain for both structural and textural restoration during the image inpainting process.

Proposed Framework. Given the original image and its inpainting mask M, both with a size of 256×256 pixels, our first step is to compute the sketch-pencil image and its damaged version I_s.

Subsequently, TSTR uses I_s and M to compute the inpainted sketch-pencil image $\hat{I}_s$. The TSTR architecture is composed of an encoder, eight transformer blocks, and a decoder. A description of these components is provided in the following subsections.

[1]https://github.com/rra94/sketchify/tree/master

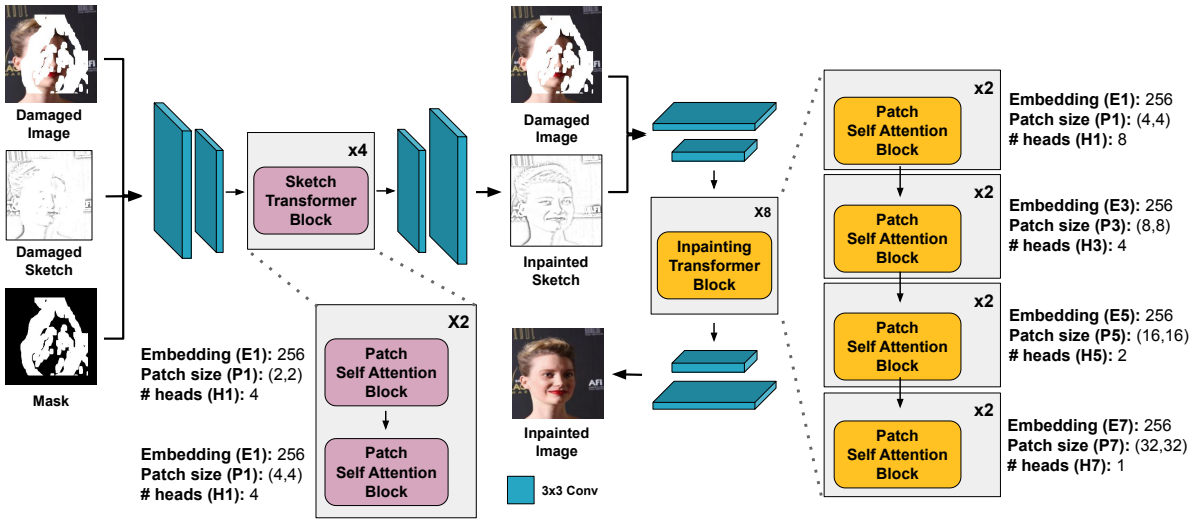

Figure 2: An illustration of our image inpainting method based on vision transformers. Left: TSTR restores the damaged sketch-pencil image from inputs including the damaged image and mask. Right: ETI computes the inpainted image by using the restored sketch-pencil image and the damaged image.

Figure 3: Conversion of an input image into the sketch-pencil domain using image processing techniques. (1) The input image is transformed into a gray-scale image. (2) The gray image is inverted. (3) Gaussian blur is applied to the inverted image. (4) The smoothed inverted image is inverted to the original. (5) The sketch-pencil image is computed by blending the smoothed gray image with the gray image.

Figure 4: Illustration showing the degree of structural and texture information that can be recovered by using a larger or smaller Gaussian filter.

3.1.1 Encoder-Decoder

We used two convolutional layers on both the encoder and decoder to reduce computations and memory usage in transformers. For each layer, we employed LeakyReLU as the activation function and Instance Normalization to help stabilize the training process and learn better representations. Furthermore, convolutional layers may be particularly advantageous to effectively capture structural information, leading to a better representation and optimization (Raghu et al., 2021).

3.1.2 Sketch-Pencil Transformer Blocks

We used the patch self-attention mechanism (Campana et al., 2022), which aims to capture the global image context while reducing memory costs in both training and inference. We employ four pairs of transformer blocks, and adopt a multi-scale patch partitioning strategy in each one (Figure 5).

In the first and second blocks, we strategically vary the patch size to strike a balance between capturing the global context and enhancing computational efficiency during both training and inference. This approach not only optimizes memory usage but also contributes to more effective structural and texture information restoration from the sketch domain, which, in turn, guides our image inpainting model to obtain coherent and reliable results.

We define the set of patch sizes denoted as $P = \{2,4\}$ for each group of transformer blocks. This selection ensures that the information belonging to edges and lines from the sketch domain is adequately restored, aligning with the requirements of our inpainting process.

Concerning the number of attention heads in our model, it is worth noting that a larger number can enable simultaneous focus on different regions of the image, enhancing the model's capacity to learn intri-

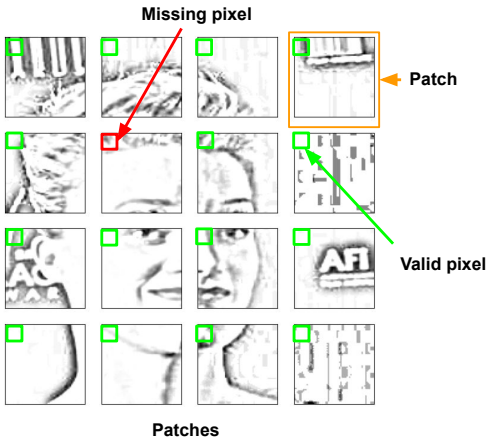

Figure 5: Patch self-attention mechanism is used to attend to the missing pixels by capturing the global context of the image based on the distance between valid pixels of each patch.

cate patterns. However, this also requires an increase in the number of parameters, resulting in higher computational costs during both training and inference, including memory usage.

In our model, we maintain a consistent number of heads, which is denoted as $K = \{4,4\}$. This choice strikes a balance between model effectiveness and computational efficiency, ensuring that the proposed model can effectively learn complex patterns while remaining manageable in terms of memory and computation.

3.2 Efficient Transformer Inpainting

In our approach, we adopt the same model configuration as presented by Campana et al. (2022). In every two Inpainting Transformer Blocks, we increase the patch size, denoted as $p = \{4,8,16,32\}$. On the other hand, we decrease the number of attention heads, denoted as $k = \{8,4,2,1\}$.

The inpainting process unfolds as follows. Given a damaged image I_d, a mask M, and an inpainted sketch-pencil image $\hat{I}s$ as input, all in a 256×256 pixel resolution, these components are jointly passed to the Efficient Transformer Inpainting (ETI) model. ETI is responsible for the restoration of both structural and textural information using prior information from the sketch-pencil domain, yielding the inpainted image denoted as I_{out} that seamlessly integrates the visually realistic content.

3.3 Loss Functions

We adopt the same loss functions as those described by Campana et al. (2022), expressed as

$$L_{total} = \lambda_{rec}L_{rec} + \lambda_{style}L_{style} + \lambda_{perc}L_{perc} + \lambda_{adv}L_{adv} \quad (1)$$

where $\lambda_{rec} = 1$, $\lambda_{style} = 90$ for TSTR and 360 for ETI, $\lambda_{perc} = 1.5$ for TSTR and 0.9 for ETI, and $\lambda_{adv} = 0.01$. We assigned higher weights to the hole, valid, and perceptual losses for TSTR, aiming to emphasize the structural aspects. In contrast, we set a higher weight to the style loss on ETI to emphasize the restoration of texture details. We define each term in the following paragraphs.

Reconstruction Loss (L_{rec}). This loss ensures the coherence between the inpainted and surrounding known regions, in addition to approaching the ground-truth information as closely as possible. We defined this loss as the sum of the hole loss L_{hole} and valid loss L_{valid}:

$$L_{hole} = \frac{1}{N_{\mathbb{I}-M}} \left\| (\mathbb{I} - M) \odot (I_{out} - I_{gt}) \right\|_1 \quad (2)$$

$$L_{valid} = \frac{1}{N_M} \left\| M \odot (I_{out} - I_{gt}) \right\|_1 \quad (3)$$

$$L_{rec} = L_{hole} + L_{valid} \quad (4)$$

where $\mathbb{I}$ the identity matrix, and $\frac{1}{N_{\mathbb{I}-M}}$ and $\frac{1}{N_M}$ denote the number of holes and valid pixels in M.

Perceptual (L_{perc}) and Style Losses (L_{style}). Perceptual loss (L_{perc}) encourages the inpainted image I_{out} to match the overall visual appearance of the ground truth image I_{gt}. In turn, Style loss (L_{style}) helps to preserve and match the texture characteristics of the ground truth image I_{gt}. We defined the perceptual and style loss, respectively, as

$$L_{perc} = \sum_i \frac{\left\| \phi_i(I_{out}) - \phi_i(I_{gt}) \right\|_1}{N_{\phi_i(I_{gt})}} \quad (5)$$

and

$$L_{style} = \sum_i \frac{\left\| \omega_i(I_{out}) - \omega_i(I_{gt}) \right\|_1}{N_{\omega_i(I_{gt})}}, \quad (6)$$

where ϕ_i (.), $i = 1,\ldots,5$ denote the activation maps ReLu1_1, ReLu2_1, ReLu3_1, ReLu4_1, and ReLu5_1 from a pre-trained VGG-16 (Simonyan and Zisserman, 2015). ω_i (I) = $\phi_i(I)^T\phi_i(I)$ denotes the Gram matrix formed by four activation maps from VGG-16: ReLu2_2, ReLu3_3, ReLu4_3, and ReLu5_2. $N_{\phi_i(I_{gt})}$ denotes the dimension of the feature map $\phi_i(I_{gt})$, and $N_{\omega_i(I_{gt})}$ denotes the dimension of the feature map $\omega_i(I_{gt})$, which are used as a normalization factor.

Adversarial Loss (L_{adv}). This loss forces the generated inpainted image I_{out} to be indistinguishable from ground truth image I_{gt}. We employ the adversarial loss proposed by Goodfellow et al. (2014), defined as

$$L_G = -\mathbb{E}_{I_{out}}[\log(D(I_{out}))], \tag{7}$$

$$L_D = -\mathbb{E}_{I_{gt}}[\log(D(I_{gt}))] - \mathbb{E}_{I_{out}}[\log(1 - D(I_{out}))], \tag{8}$$

$$L_{adv} = L_G + L_D, \tag{9}$$

where L_G and L_D represent the loss functions of the generator and discriminator.

4 EXPERIMENTS

In this section, we present our experimental results. First, we briefly describe the datasets used and some implementation details. Then, we report and discuss both our quantitative and qualitative results.

4.1 Datasets

For our experiments, we utilized three well-established datasets commonly used in inpainting research: Places2 (Zhou et al., 2017), which encompasses images from 365 diverse scene categories; CelebA (Liu et al., 2015), comprising facial images of celebrities; and Paris StreetView (PSV) (Doersch et al., 2015), which includes street views and buildings from Paris. In the case of Places2, we employed the 1.8 million images from the training set and 36.5 thousand images from the validation set for training and evaluation, respectively. These sets were obtained from the Places2 dataset available at the following link: http://places2.csail.mit.edu/index.html. For CelebA and PSV datasets, CelebA contains 202.599K and PSV 15.000K total images, so we performed a split into training and validation sets, and the reported results are based on this single training-validation split. Specifically, for CelebA, we used approximately 162.7K images for training and 19.961K images for validation. For PSV, we employed 14.9K images for training and 100 images for validation.

We used irregular masks generated online during training. For validation, we employed the mask set defined by Liu et al. (2018), which consists of 12K irregular masks equally divided into six intervals based on hole size. In this study, we employed only three intervals: 20-30%, 30-40%, and 40-50%.

4.2 Implementation Details

Our method was implemented using PyTorch. We set the batch size as 16 and resized the input image to 256×256 for both TSTR and ETI. We trained the TSTR and ETI using Adam optimizer with $\beta_1 = 0.99$ and $\beta_2 = 0.9$.

We trained TSTR for 75, 50, and 40 epochs and set the initial learning rate to 10^{-5}, 10^{-4}, and 10^{-4}, respectively, for Places2, CelebA, and PSV. Additionally, we decayed these learning rates by a factor of 10^{-1} in the last 5, 10, and 10 epochs for Places2, CelebA, and PSV, respectively. For ETI, we used 80, 75, and 75 epochs, respectively, for Places2, CelebA, and PSV. The initial learning rate was set to 10^{-4} and was decayed in the same manner as during the training of TSTR.

4.3 Quantitative Comparison

Inpainting Results. To assess our experiments, we used four well-established metrics: Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index (SSIM), Frechet Inception Distance (FID) (Heusel et al., 2017) and Learned Perceptual Image Patch Similarity (LPIPS) (Zhang et al., 2018). PSNR and SSIM are simpler measures that assess image similarity according to the ground truth. On the other hand, FID and LPIPS are particularly important for evaluating the perceptual realism of the inpainted regions.

For the Places2 dataset, our method showed better results than our baseline (Campana et al., 2022), primarily due to the incorporation of sketch-pencil domain information. However, ZITS achieved the best results for the perceptual measures, generating high-quality inpainted images. In addition, LaMa outperformed our method by a slight margin in terms of FID but was outperformed by us in terms of LPIPS.

For CelebA and PSV datasets, our method was ranked among the best for all metrics, but especially for perceptual ones, FID and LPIPS. This strong performance emphasizes the efficacy of TSTR in generating highly realistic inpainted images across various datasets and scenarios.

Sketch-Pencil Results. Table 2 shows quantitative results related to the sketch-pencil image inpainting (TSTR) on the Places2, CelebA, and PSV datasets. These results highlight the effectiveness of our model in restoring this information coherently.

To evaluate the quality of the restored edges and lines, as well as texture, we employ the SSIM and LPIPS metrics. SSIM assesses the structural similarity between the restored edges and lines, with values closer to 1 indicating greater similarity. On the other hand, LPIPS measures the perceptual similarity of the texture content within the sketch-pencil domain between the real and restored sketch-pencil image, with

Table 1: Comparison of our method against state-of-the-art approaches on Places2, CelebA and Paris Street View. The first and second-best results are marked in **bold** and underline, respectively.

Datasets	Methods	PSNR ↑			SSIM ↑			FID ↓			LPIPS ↓		
		20-30%	30-40%	40-50%	20-30%	30-40%	40-50%	20-30%	30-40%	40-50%	20-30%	30-40%	40-50%
Places2	Edge-Connect (Nazeri et al., 2019)	24.9439	22.8172	21.1207	0.8661	0.8043	0.7373	2.8315	5.5362	9.9219	0.0841	0.1253	0.1722
	CTSDG (Guo et al., 2021)	25.7374	23.4326	21.6453	0.8817	0.8212	0.7552	3.7493	8.6340	16.8813	0.0911	0.1421	0.1992
	WaveFill (Yu et al., 2021)	26.1047	23.7590	21.4553	0.8874	0.8274	0.7422	1.3011	3.2134	11.3293	0.0647	0.1028	0.1697
	SPL (Zhang et al., 2021)	**27.6768**	**25.2369**	**23.2940**	**0.9105**	**0.8618**	**0.8064**	2.0407	4.5186	8.8990	0.0722	0.1137	0.1616
	MADF (Zhu et al., 2021)	26.9094	24.5930	22.7039	0.8938	0.8430	0.7855	1.2426	2.5276	5.1664	0.0897	0.1214	0.1599
	Lama (Suvorov et al., 2022)	26.0241	23.9370	22.2043	0.8770	0.8266	0.7701	1.0391	**1.6844**	**2.6772**	0.1165	0.1426	0.1747
	Patch-Attn (Campana et al., 2022)	26.4769	24.2554	22.3163	0.8923	0.8368	0.7758	1.1783	2.3969	4.6187	0.0650	0.0995	0.1404
	ZITS (Dong et al., 2022)	26.3277	24.0073	22.1937	0.8910	0.8359	0.7746	**0.9534**	1.7659	3.1039	**0.0574**	**0.0889**	**0.1261**
	Ours	26.8025	24.4342	22.5445	0.8948	0.8398	0.7779	1.3245	2.7390	5.2472	0.0629	0.0973	0.1386
CelebA	Edge-Connect (Nazeri et al., 2019)	29.1435	26.5719	24.4178	0.9047	0.8662	0.8211	2.4361	3.6728	5.7569	0.0527	0.0755	0.1040
	RFR (Li et al., 2020)	29.8901	27.2036	25.0676	0.9280	0.8886	0.8440	1.7047	2.8320	4.4911	0.0431	0.0645	0.0899
	CTSDG (Guo et al., 2021)	30.0308	27.1553	24.9321	0.9330	0.8929	0.8473	2.3009	4.3930	7.4196	0.0515	0.0780	0.1090
	SPL (Zhang et al., 2021)	**32.6547**	29.6495	27.2305	**0.9539**	**0.9249**	**0.8897**	1.2756	2.2643	3.5706	0.0421	0.0641	0.0904
	MADF (Zhu et al., 2021)	31.8397	28.7059	26.2538	0.9475	0.9135	0.8729	0.7546	1.4399	2.6177	0.0385	0.0563	0.0787
	Patch-Attn (Campana et al., 2022)	31.3763	28.7415	26.5915	0.9420	0.9105	0.8740	0.8072	1.4175	2.4025	0.0335	0.0498	0.0697
	Ours	32.5599	**29.8027**	**27.4940**	0.9482	0.9187	0.8831	**0.5761**	**0.9274**	**1.5156**	**0.0310**	**0.0450**	**0.0636**
PSV	Edge-Connect (Nazeri et al., 2019)	28.6885	26.3160	24.7027	0.8973	0.8478	0.7943	39.9341	50.4303	67.2686	0.0677	0.1027	0.1404
	RFR (Li et al., 2020)	28.8133	26.6124	24.8159	0.8999	0.8519	0.7963	30.1260	41.7321	53.7483	0.0617	0.0912	0.1280
	CTSDG (Guo et al., 2021)	29.4851	27.0640	25.0938	0.9095	0.8599	0.8013	38.7129	56.2173	76.6186	0.0808	0.1052	0.1498
	WaveFill (Yu et al., 2021)	30.1529	27.1075	26.0107	0.9178	0.8740	0.8222	28.2945	38.0996	50.4732	**0.0482**	0.0737	0.1078
	SPL (Zhang et al., 2021)	**30.9665**	**28.4221**	**26.3540**	**0.9294**	**0.8897**	**0.8407**	35.8653	47.9462	69.6496	0.0639	0.0977	0.1415
	MADF (Zhu et al., 2021)	30.6575	28.0885	26.0039	0.9247	0.8820	0.8303	24.9763	37.4429	51.7381	0.0565	0.0836	0.1198
	Patch-Attn (Campana et al., 2022)	29.9215	27.6332	25.7936	0.9145	0.8722	0.8208	24.9832	36.6138	47.9300	0.0544	0.0794	0.1135
	Ours	30.5096	28.0505	26.0226	0.9188	0.8762	0.8242	**23.6015**	**32.9914**	**44.7338**	0.0488	**0.0730**	**0.1059**

values closer to zero signifying greater structural and textural similarity between the two images.

Capitalizing on the restored structural and textural information, our ETI model effectively guides the structural and textural restoration process for the damaged image, as shown in Table 1. These results collectively affirm the success of our approach in seamlessly restoring both structural and textural elements, presenting a high inpainting performance.

4.4 Qualitative Comparison

Inpainting Results. Figure 6 compares our qualitative results with classical and state-of-the-art methods. For the CelebA and PSV datasets, our method consistently shows better structural restoration than its competitors. We highlight highly detailed textures in eyes, flowers, and architectural elements. Our semantic reconstruction is competitive, especially when compared to Patch-Attn.

On the other hand, Edge-Connect shows poor structural and textural outcomes. RFR and MADF achieved better semantic results but performed poorly in recovering the structure and texture of large masks in the face of some building regions. SPL and Wavefill outperformed the aforementioned methods at the semantic level, but SPL produces overly smoothed content, while Wavefill introduces artifacts in high-structural regions. Finally, Patch-Attn improves semantic and textural reconstruction but also produces some artifacts in large high-textural regions, such as the flowers in the CelebA dataset.

For the Places2 dataset, our method achieved competitive visual results compared to ZITS, especially for large regions requiring inpainting in complex images. However, ZITS demonstrated possibly the best results in both semantic and textural restoration among all the methods. Edge-Connect performed poorly, particularly in images featuring rich semantical and textural content. LaMa showed improved results compared to Edge-Connect and Patch-Attn but exhibited artifacts in high-texture areas, such as islands and trees.

Sketch-Pencil Results. Figure 7 shows our qualitative results for sketch-pencil images. These results highlight the good performance of our TSTR model on Places2, CelebA, and PSV datasets.

For PSV and Places2, our model efficiently restored edges and lines coherently. In CelebA, it predicted facial features properly, including the eyes and face. Our method inpainted sketch-pencil im-

Table 2: Quantitative results on Places2, CelebA and PSV for sketch-pencil inpainting.

Methods	Datasets	SSIM ↓			LPIPS ↓		
		20-30%	30-40%	40-50%	20-30%	30-40%	40-50%
Sketch-Pencil	Places2	0.8586	0.7943	0.7248	0.0879	0.1262	0.1698
	CelebA	0.9089	0.8626	0.8107	0.0526	0.0754	0.1022
	PSV	0.8879	0.8303	0.7636	0.0623	0.0913	0.1300

Figure 6: Comparison of details for inpainting results among the proposed method and literature approaches for Places2, CelebA, and Paris Street View on the Paris Street View dataset.

ages with rich textures successfully, such as those in Places2, without introducing artifacts that might confuse the ETI in the subsequent task.

5 ABLATION STUDIES

Sketch-Pencil Domain versus Inpainting Quality. Table 3 shows the inpainting results with and without sketch-pencil information. Our proposed model achieved better results for every metric when sketch-pencil information was incorporated into the inpainting model.

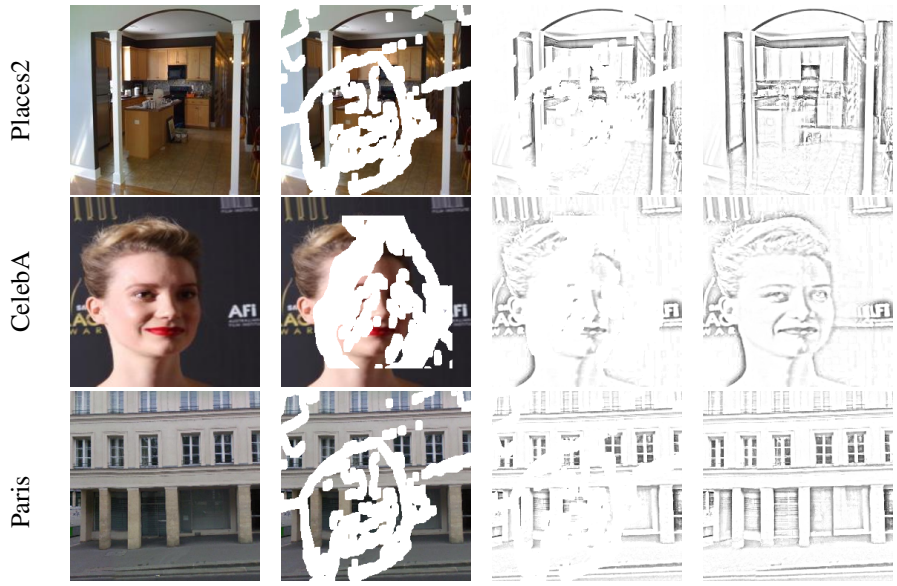

Figure 7: Visual examples of inpainted images in the sketch-pencil domain.

Table 3: Ablation study comparing inpainting results with sketch-pencil information (our full proposed model) and without sketch-pencil information (Inpainting model trained without sketch information). We employed masks with size 40-50% to evaluate our models.

Sketch-Pencil	Places2		CelebA		PSV	
	No	Yes	No	Yes	No	Yes
PSNR ↑	22.3563	**22.5445**	26.3606	**27.4940**	25.7015	**26.0226**
SSIM ↑	0.7750	**0.7779**	0.8700	**0.8831**	0.8194	**0.8242**
FID ↓	5.2778	**5.2472**	1.7700	**1.5156**	47.5351	**44.7338**
LPIPS ↓	0.1435	**0.1386**	0.0705	**0.0636**	0.1160	**0.1059**

Loss Function for Image Inpainting. Table 4 presents the impact of multiple loss function combinations, namely the reconstruction loss (L_{rec}), style loss (L_{style}), perceptual loss (L_{perc}) and adversarial loss (L_{adv}) for sketch-pencil prediction, using the CelebA dataset. We verified that we achieved the best results when using all three losses, which suggests that this combination contributes to the overall quality and realism of the inpainted images.

Sketch-Pencil Domain versus Edges from EdgeConnect. We conducted an analysis to compare the impact of using the Canny edge detector (Nazeri et al., 2019) versus sketch-pencil information. We experimented with this comparison on the CelebA dataset. We report these results in Table 5. Using sketch-pencil information improved the inpainting results, especially due to the enhanced structural information restoration and more detailed texture compared to Canny edge detector.

6 CONCLUSIONS

We propose a method based on Vision Transformers, which establishes a clear consistency between structural and texture information through the utilization of the sketch-pencil domain. Our approach is based on the use of a model that previously restores the semantic structural information using edges and lines extracted from the sketch-pencil domain. Furthermore, the proposed model also serves as a base to guide the restoration of the texture of the damaged images using the restored texture in the sketch-pencil domain.

Quantitative assessments based on experimental results demonstrate the superiority of our approach. We have achieved remarkable results on benchmark datasets such as CelebA and Paris StreetView, and our performance remains highly competitive on Places2 dataset. Moreover, qualitative evaluations reveal the compelling ability of our method to consistently and reliably restore both structural and textural elements

Table 4: Ablation study comparing loss function for sketch-pencil inpainting on CelebA.

Losses	FID ↓			LPIPS ↓		
	20-30%	**30-40%**	**40-50%**	**20-30%**	**30-40%**	**40-50%**
$L_{rec} + L_{perc} + L_{adv}$	0.8326	1.6155	3.3576	0.0348	0.0521	0.0751
$L_{rec} + L_{style} + L_{adv}$	0.7143	1.1646	1.7654	0.0339	0.0489	0.0686
$L_{rec} + L_{style} + L_{perc} + L_{adv}$	**0.5761**	**0.9274**	**1.5156**	**0.0310**	**0.0450**	**0.0636**

Table 5: Ablation study comparing sketch-pencil with edges on CelebA.

Losses	PSNR ↑			SSIM ↑			FID ↓			LPIPS ↓		
	20-30%	**30-40%**	**40-50%**	**20-30%**	**30-40%**	**40-50%**	**20-30%**	**30-40%**	**40-50%**	**20-30%**	**30-40%**	**40-50%**
Canny	31.2759	28.5871	26.4022	0.9415	0.9091	0.8713	0.6910	1.1469	1.8917	0.0335	0.0500	0.0703
Sketch-pencil	**32.5599**	**29.8027**	**27.4940**	**0.9464**	**0.9187**	**0.8793**	**0.6185**	**1.0203**	**1.6966**	**0.0331**	**0.0485**	**0.0685**

within the missing regions, culminating in visually pleasing inpainted images.

ACKNOWLEDGEMENTS

The authors would like to thank CNPq (#309330/2018-1) and FAPESP (#2017/12646-3) for their support.

REFERENCES

Campana, J. L. F., Decker, L. G. L., Souza, M. R., Maia, H. A., and Pedrini, H. (2022). Multi-Scale Patch Partitioning for Image Inpainting Based on Visual Transformers. In *35th Conference on Graphics, Patterns and Images (SIBGRAPI)*, volume 1, pages 180–185. IEEE.

Campana, J. L. F., Decker, L. G. L., Souza, M. R., Maia, H. A., and Pedrini, H. (2023). Variable-Hyperparameter Visual Transformer for Efficient Image Inpainting. *Computers & Graphics*, 113:57–68.

Cao, C., Dong, Q., and Fu, Y. (2022). Learning Prior Feature and Attention Enhanced Image Inpainting. In *17th European Conference on Computer Vision*, pages 1–8, Tel Aviv, Israel.

Doersch, C., Singh, S., Gupta, A., Sivic, J., and Efros, A. A. (2015). What Makes Paris Look Like Paris? *Communications of the ACM*, 31(4):1–10.

Dong, Q., Cao, C., and Fu, Y. (2022). Incremental Transformer Structure Enhanced Image Inpainting with Masking Positional Encoding. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11358–11368.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., and Houlsby, N. (2021). An Image Is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *9th International Conference on Learning Representations*, pages 1–22.

Gamini, S. and Kumar, S. (2019). Image Inpainting Based on Fractional-Order Nonlinear Diffusion for Image Reconstruction. *Circuits, Systems, and Signal Processing*, 15(38):3802–3817.

Ghorai, M., Samanta, S., Mandal, S., and Chanda, B. (2019). Multiple Pyramids Based Image Inpainting Using Local Patch Statistics and Steering Kernel Feature. *IEEE Transactions on Image Processing*, 28(11):5495–5509.

Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A. C., and Bengio, Y. (2014). Generative adversarial nets. In Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N. D., and Weinberger, K. Q., editors, *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems*, Montreal, Quebec, Canada.

Guo, X., Yang, H., and Huang, D. (2021). Image Inpainting via Conditional Texture and Structure Dual Generation. In *IEEE/CVF International Conference on Computer Vision*, pages 14134–14143.

He, K., Chen, X., Xie, S., Li, Y., Dollár, P., and Girshick, R. B. (2022). Masked Autoencoders Are Scalable Vision Learners. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15979–15988, Orleans, LA, USA.

Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., and Hochreiter, S. (2017). GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium. In *Neural Information Processing Systems*, pages 1–12.

Li, J., Wang, N., Zhang, L., Du, B., and Tao, D. (2020). Recurrent Feature Reasoning for Image Inpainting. In *Conference on Computer Vision and Pattern Recognition*, pages 7760–7768.

Li, W., Lin, Z., Zhou, K., Qi, L., Wang, Y., and Jia, J. (2022). MAT: Mask-Aware Transformer for Large Hole Image Inpainting. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10758–10768.

Liao, L., Xiao, J., Wang, Z., Lin, C., and Satoh, S. (2021). Image Inpainting Guided by Coherence Priors of Semantics and Textures. In *IEEE Conference on Com-*

puter Vision and Pattern Recognition. Computer Vision Foundation / IEEE.

Liu, G., Reda, F. A., Shih, K. J., Wang, T.-C., Tao, A., and Catanzaro, B. (2018). Image Inpainting for Irregular Holes Using Partial Convolutions. In *European Conference on Computer Vision*, pages 85–100.

Liu, H., Jiang, B., Song, Y., Huang, W., and Yang, C. (2020). Rethinking Image Inpainting via a Mutual Encoder-Decoder with Feature Equalizations. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*.

Liu, Z., Luo, P., Wang, X., and Tang, X. (2015). Deep Learning Face Attributes in the Wild. In *IEEE International Conference on Computer Vision*, pages 3730–3738.

Nazeri, K., Ng, E., Joseph, T., Qureshi, F. Z., and Ebrahimi, M. (2019). EdgeConnect: Generative Image Inpainting with Adversarial Edge Learning. *arXiv*.

Qiu, J., Liu, B., He, J., Liu, C., and Li, Y. (2019). Parallel fast pencil drawing generation algorithm based on gpu. *IEEE Access*.

Raghu, M., Unterthiner, T., Kornblith, S., Zhang, C., and Dosovitskiy, A. (2021). Do Vision Transformers See Like Convolutional Neural Networks? In Ranzato, M., Beygelzimer, A., Dauphin, Y. N., Liang, P., and Vaughan, J. W., editors, *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems*, pages 1–8.

Simonyan, K. and Zisserman, A. (2015). Very Deep Convolutional Networks for Large-Scale Image Recognition. In *3rd International Conference on Learning Representations*, pages 1–14.

Strudel, R., Garcia, R., Laptev, I., and Schmid, C. (2021). Segmenter: Transformer for Semantic Segmentation. In *IEEE/CVF International Conference on Computer Vision*, pages 7262–7272.

Suvorov, R., Logacheva, E., Mashikhin, A., Remizova, A., Ashukha, A., Silvestrov, A., Kong, N., Goka, H., Park, K., and Lempitsky, V. (2022). Resolution-robust Large Mask Inpainting with Fourier Convolutions. In *Winter Conference on Applications of Computer Vision*, pages 2149–2159.

Wan, Z., Zhang, B., Chen, D., Zhang, P., Chen, D., Liao, J., and Wen, F. (2020). Bringing Old Photos Back to Life. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2747–2757.

Yang, J., Qi, Z., and Shi, Y. (2020). Learning to incorporate structure knowledge for image inpainting. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*.

Yu, J., Lin, Z., Yang, J., Shen, X., Lu, X., and Huang, T. (2019). Free-form Image Inpainting with Gated Convolution. *arXiv 1806.03589*.

Yu, J., Lin, Z., Yang, J., Shen, X., Lu, X., and Huang, T. S. (2018). Generative Image Inpainting with Contextual Attention. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*.

Yu, Y., Zhan, F., Lu, S., Pan, J., Ma, F., Xie, X., and Miao, C. (2021). WaveFill: A Wavelet-Based Generation Network for Image Inpainting. In *IEEE/CVF International Conference on Computer Vision*, pages 14114–14123.

Zeng, Y., Lin, Z., Yang, J., Zhang, J., Shechtman, E., and Lu, H. (2020). High-Resolution Image Inpainting with Iterative Confidence Feedback and Guided Upsampling. In *European Conference on Computer Vision*, pages 1–17.

Zhang, R., Isola, P., Efros, A. A., Shechtman, E., and Wang, O. (2018). The Unreasonable Effectiveness of Deep Features as a Perceptual Metric. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 586–595.

Zhang, W., Zhu, J., Tai, Y., Wang, Y., Chu, W., Ni, B., Wang, C., and Yang, X. (2021). Context-Aware Image Inpainting with Learned Semantic Priors. In Zhou, Z., editor, *Thirtieth International Joint Conference on Artificial Intelligence*, pages 1–7.

Zhou, B., Lapedriza, A., Khosla, A., Oliva, A., and Torralba, A. (2017). Places: A 10 Million Image Database for Scene Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(6):1452–1464.

Zhu, M., He, D., Li, X., Li, C., Li, F., Liu, X., Ding, E., and Zhang, Z. (2021). Image Inpainting by End-to-End Cascaded Refinement With Mask Awareness. *IEEE Transactions on Image Processing*, 30:4855–4866.