Composite Inpainting:
Addressing Shortcomings in GAN and Patch-Based Inpainting with a Hybrid Approach

Kai Zhu, Maria Alejandra Escalante

Abstract

Deep learning-based inpainting models such as generative adversarial networks (GANs) and stable diffusion offer state of the art generation of realistic content in complex scenes using contextual surroundings. There exist, however, drawbacks for each method: GANs are difficult to train and often fail to capture textures effectively ; Stable diffusion is prone to hallucinations. Conversely, classical patch-based methods reportedly excel at replicating textural details but struggle with generating larger structurally consistent content. We propose a hybrid approach using GAN and patch-based methods to leverage their strengths and improve both structural and textural consistency. Preliminary results demonstrate that this approach yields minor qualitative improvements in texture details, but the quantitative assessments, possibly limited by the current metrics, remain inconclusive.

I. Introduction

Image inpainting is an area that has been actively researched because of its many applications such as image restoration (recovering damaged or missing portions of an image), and photo editing(removing unwanted objects). Several inpainting techniques have been developed, ranging from deep learning-based methods including stable diffusion, generative adversarial networks (GANs), convolutional neural networks, (CNNs), to classical patch-based approaches. Each method has strengths and drawbacks in inpainting textures, facial features, large masked regions, and edge preservation [3].

While capable of generating highly realistic inpainting, modern deep learning approaches face several challenges. Neural networks are often considered black boxes with limited explainability [4]. The complexity of the inner workings of models create difficulties when addressing specific issues. For example, GANs are difficult to train and often struggle with textural consistency [3]. Diffusion models improve upon some of these weaknesses, but are prone to hallucinations [2].

Patch-based techniques, now considered classical, have shown promises in inpainting repetitive patterns such as grass, water, and architectural features [5]. Additional benefit of these approaches include same-image sourcing to potentially improve textural consistency, no requirement for training, and rule-based algorithms that are highly explainable and thus more intuitive for manual tuning and modification. Drawbacks to the classical methods include processing times and reliance on the availability of repeating textures and patterns [3].

We propose utilizing a combination of both deep learning-based and classical patch-based approaches to complement the strengths of each technique and minimize their shortcomings. For example, the ability of GANs in generating structurally consistent content can be exploited to create larger shapes based on the surrounding context, while a patch-based method fills low texture regions with improved details.

II. Related Work

Free-Form Image Inpainting with Gated Convolution presents a novel system using gated convolutions and SN-PatchGAN, enhancing inpainting quality and color consistency over prior methods [9]. The proposed framework [9] consists of 3 main components. The first stage features a gated convolution coarse network which generates an inpainted image with blurry but contextually plausible

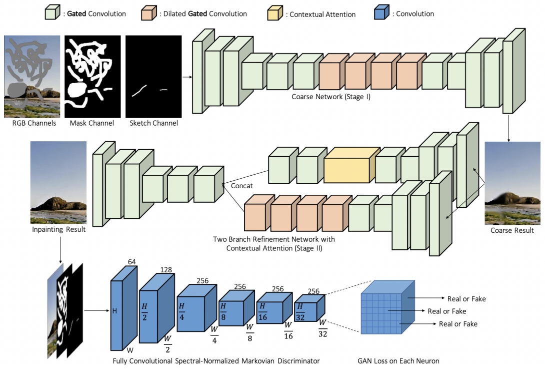


Fig. 1: Overview of the framework with gated convolution and SN-PatchGAN for free-form image inpainting [9]

regions. Stage 2 is a gated convolution refinement network with contextual attention responsible for generating finer details based on the coarse output. The final component is a fully convolutional spectral-normalized Markovian discriminator for identifying real and synthetic results during training time (figure 1).

Generative Patch Nearest-Neighbor (GPNN) is an efficient, high quality patch-based single-image generation method. GPNN adopts SinGAN's multi-scale architecture while replacing the generator and discriminator with patch nearest-neighbor modules [5]. While the paper showcased results comparable to or better than GAN-based models, it does not discuss inpainting in detail so we were unable to replicate their results.

Latent diffusion models (LDM) introduced by Rombach et al. enhances efficiency and quality in high-resolution image synthesis and inpainting [7]. The runwayml/stable-diffusion-inpainting model used in our comparison is based on this model.

The **Gabor filter** is a linear filter employed in texture analysis. In essence, it examines whether there is particular frequency information in the image within specific directions in a localized area around the point or region under scrutiny. Gabor filters are recognized as a prominent method in texture classification applicable in textural analysis for inpainting. Bianconi et al [1] examine the impact of various Gabor filter parameters on texture discrimination. Their research indicates that while increasing frequencies and orientations has limited effect, smoothing parameters significantly enhance classification performance.

III. Methodology

We examined recent GAN [9] and patch-based [5] methods to explore suitable options to base a hyrbid method upon. A frequent challenge during this process was that available implementations featured in or built upon these studies were poorly maintained with insufficient documentation, written in multiple programming languages unsuited to the scope of our study, and in many cases not executable. In addition, implementation of inpainting-specific tasks were often incomplete [6] or mentioned but unexplained in the original papers [5]. Hence, while we were able to use off-the-shelf stable diffusion and GAN methods, a classical patch-based method was implemented based on a combination of patchmatch algorithm and the pyramidal architecture introduced in the GPNN paper [5], henceforth referred to as multi-scale patch-nearest neighbor (MPNN).

The base methods including stable diffusion, GAN, and GPNN were implemented in Python to facilitate modifications, composition, and evaluation.

Single-method results from DeepFill GAN and MPNN were composited using different blending approaches based on high pass (figure 2) and Gabor linear filtering. An additional hybrid method was explored by substituting the refinement stage of the DeepFill GAN with a patchmatch implementation pass, which we named refinement patch nearest-neighbor (RPNN).

A. Base Methods

1) Stable Diffusion: A popular off-the-shelf diffusion model [7] pretrained for inpainting-specific tasks was obtained from Hugging Face to generate inpainted results for comparison with other base and hybrid methods.

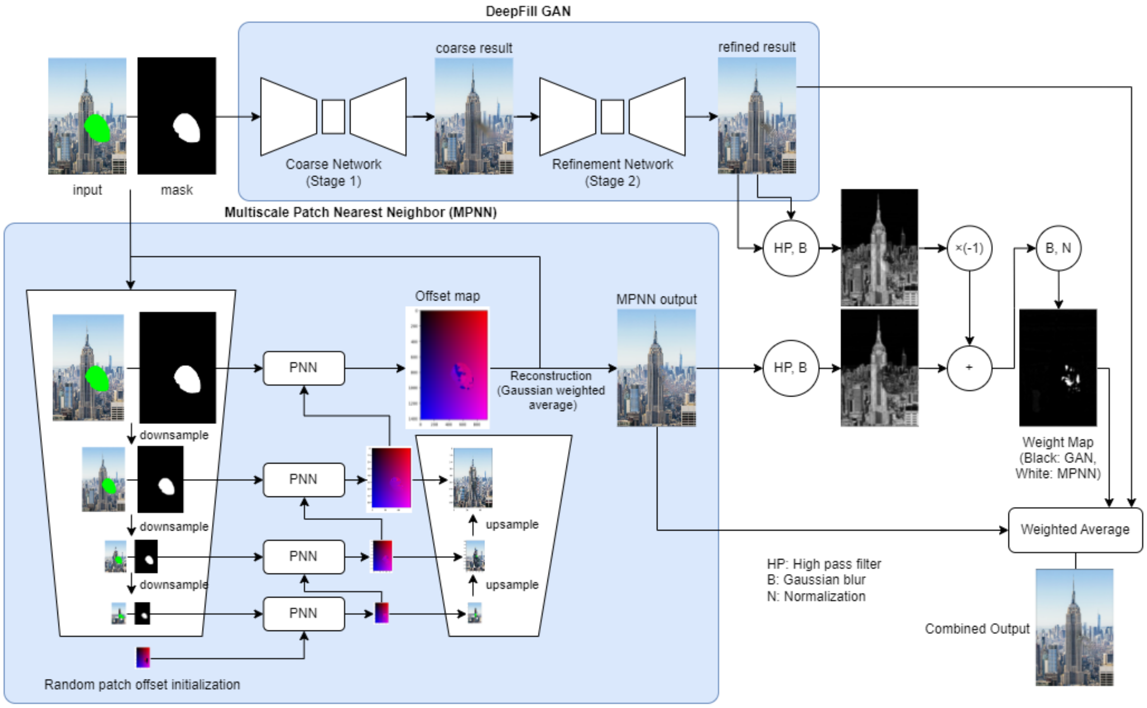


Fig. 2: Overview of the GAN-MPNN hybrid architecture. The final composition is produced using high pass filtering to substitute low-detail areas of the GAN result with highly textured MPNN output.

2) DeepFill GAN: We selected DeepFillV2 for GAN-based inpainting, primarily for its PyTorch framework which aligned with our Python-centric workflow. Notably, this model demonstrated exceptional performance in handling complex inpainting tasks involving free-form masks, which is essential for reconstructing irregular regions within images. The availability of its pretrained model significantly accelerated our process and provided a robust basis for our hybrid methods.

3) MPNN: Our implementation of a multi-scale patch nearest-neighbor (MPNN) method is inspired by the architecture showcased in Drop the GAN [5], which substitutes the UNet neurons with patch nearest neighbor (PNN) modules. We further modified the architecture (figure 2) by iteratively halving both the source (x_t where t denotes the number of times the image has been downsampled) and mask (m_t) images until the masked dimensions are below the constant patch size. Random noise is used within the masked area to initialize the nearest-neighbor field (NNF), an offset matrix mapping destination patches to every pixel on the source image. Using the initial NNF (f_T where T denotes the lowest scale in the pyramid), a patchmatch algorithm is used to propagate best matches from neighboring pixels followed by random search steps to escape local minimums.

$$f_t = \mathrm{PNN}(\mathrm{upsample}(f_{t+1}), x_t, m_t)$$

We propose that downsampling while keeping a constant patch size allows the PNN to focus on progressively larger areas in the image, thereby generating low resolution inpainting using structural context. The upsampling steps focus on smaller portions in the source image to refine textural details. Rather than using single-pixel best-matches, our reconstruction method averages overlapping patches weighed using a Gaussian kernel to enhance consistency and smoothness between neighboring reconstructed pixels.

B. Hybrid Methods

1) RPNN: We propose addressing the shortcomings of patch-based methods in structural consistency by using the coarse output from the first stage of the DeepFill GAN as the reference image for a refinement patch nearest-neighbor pass. In contrast to MPNN, the RPNN receives the structural context from the coarse GAN output rather than downsampling, so multi-scaling is not required. To introduce additional texture based on the blurry GAN output, we perturb the RPNN input with Gaussian blur and noise ($\sigma = 0.1$) to encourage more stochastic patch matching.

2) High Pass blending: An alternative composition method generates inpaintings using DeepFill GAN and MPNN in parallel. High pass filtering (hpf) is applied on each output ($o_{\text{GAN}}, o_{\text{MPNN}}$) followed by blurring ($B$) to isolate areas featuring higher frequency which may correspond to higher levels of texturization. The blurred responses from respective methods are subtracted such that the output is a mask representing areas where the MPNN output contained more textural details than the GAN output. The mask is further blurred and normalized (N) to improve smoothness of the output. The base method outputs are finally blended using the mask using weighed average (figure 2).

$$\text{mask} = N(B(B(hpf(o_{\text{MPNN}})) - B(hpf(o_{\text{GAN}}))))$$
$$o_{\text{hybrid}} = (1 - \text{mask}) \odot o_{\text{GAN}} + \text{mask} \odot o_{\text{MPNN}}$$

3) Gabor linear blending: In this alternate method, Gabor filters were applied to blend the DeepFill GAN and MPNN outputs by extracting highly textured regions from each of output image. The process involved:

- **Gabor Filter Application**: Gabor filters were applied at orientations (θ) $0, \frac{\pi}{4}, \frac{\pi}{2}, \frac{3\pi}{4}$, using a fixed kernel size (11,23), scale of 8, spatial aspect ratio of 0.8, and standard deviation (σ) of 10. The Gabor filter function is given by:

  $$\begin{aligned} &G(x, y; \lambda, \theta, \psi, \sigma, \gamma) \\ &= \exp\left(-\frac{x'^2 + \gamma^2 y'^2}{2\sigma^2}\right) \cos\left(2\pi\frac{x'}{\lambda} + \psi\right) \end{aligned}$$

  λ represents the wavelength of the sinusoidal factor, ψ the phase offset, and γ the spatial aspect ratio. The chosen parameters resulted in reasonable processing times and generated 4 Gabor response matrices for each image.
- **Aggregate Response Calculation**: The aggregate Gabor response for each pixel was the maximum normalized response from the filter outputs, computed as:

  $$\text{agg\_response} = \max_i \left(\frac{|\text{responses}[i]|}{\max\left(|\text{responses}[i]|\right)} \right)$$

  Where i represents the index in the set of previously generated responses. This step focused on identifying the most pronounced textured areas.
- **Image Blending**: The final image was a linear blend of GAN and MPNN outputs, weighted by their respective Gabor responses:

  $$o_{\text{blended}} = o_{\text{GAN}} * R_{\text{GAN}} + o_{\text{MPNN}} * R_{\text{MPNN}}$$

  o_{GAN} and o_{MPNN} are the images from the respective methods, and R_{GAN} and R_{MPNN} their Gabor responses. This method aimed to leverage textural details from both techniques, resulting in a composite image with enhanced textures.

C. Data Source and Analysis

A variety of images featuring landscape, architecture, and people were procured from Google image search. Our selection favored images with a broad spectrum of colours and textural densities. Black and white masks were created manually in Adobe Photoshop for each input image. Additionally, the Places2 data set was included during testing.

Quantitative analysis were performed using mean square error (MSE), structural similarity index (SSIM), and peak signal-to-noise ratio (PSNR) of the inpainted results compared with the unmasked source images.

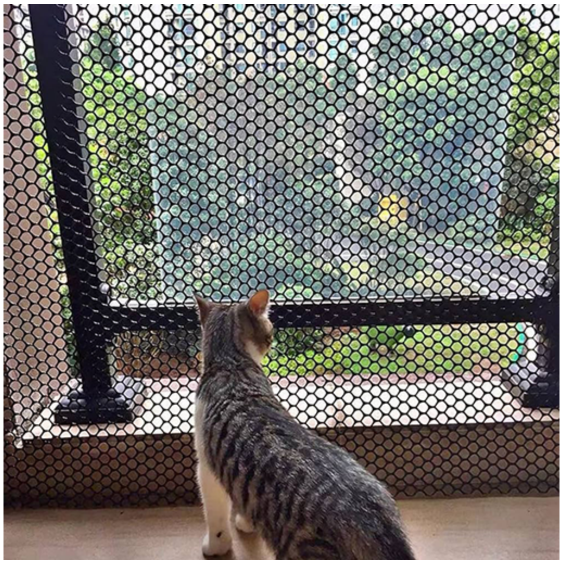


Fig. 3: Original cat image

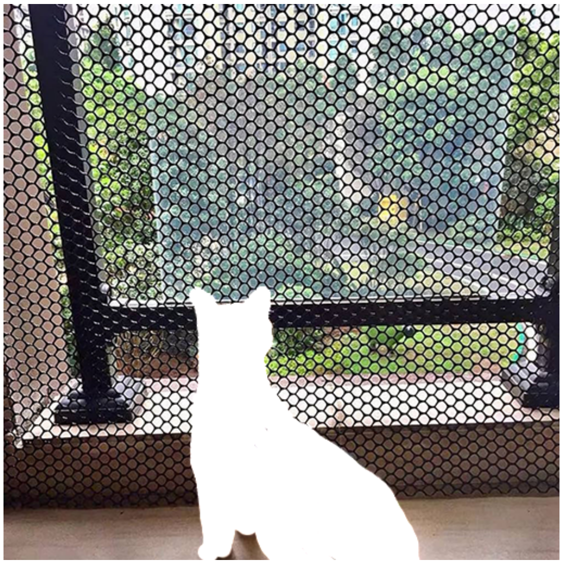


Fig. 4: Image with the cat masked out

IV. Experimental Results

Our methodologies were applied to a diverse set of images, including nature, urban, and nocturnal scenes with varying textures and colors. The sample image in figure 3 with the corresponding mask in figure 4 exemplifies the varied results due to the fence texture.

Figure 5 presents the inpainting results. The base methods (first row) demonstrate varying effectiveness: the diffusion method often hallucinated irrelevant objects (e.g., a hand or a paintbrush); DeepFill GAN produced visibly distorted and blurry fence background that lacked textural accuracy; MPNN showed a more realistic texture matching the surrounding fence, but with noticeable structural inconsistency in the form of incomplete fence coverage. These observations appear consistent with the mentioned drawbacks in previous literature for each method.

The hybrid methods (second row) showed varied improvements. High pass blending improved fence texture integrity by sampling from MPNN result in areas where GAN results were poor without introducing the structural gap present in the MPNN result. Gabor blending similarly yielded a smoother image with enhanced textures though certain artifacts and inconsistencies remained noticeable. RPNN produced the least satisfactory results, with increased blurriness, larger gaps, and inconsistent edges.

Quantitative results (figure 6) for the different base and hybrid methods, evaluated using PSNR, SSIM, and MSE, showed minor differences indistinguishable from statistical noise. Specifically in the case of SSIM, all 6 tested methods yielded scores of approximately 0.96. PSNR values were also similar, ranging between 25.2 to 25.9 dB. High pass blending showed an improved MSE compared to the base methods, but fared slightly worse than other hybrid methods despite achieving better qualitative results. Gabor blending had the lowest MSE and highest PSNR, suggesting the most accurate reconstruction. RPNN's performance was similar to high pass blending. Overall, the various scores showed little variation across the methods, providing limited insights.

The generation of novel content to fill the masked regions presents an inherent challenge to these metrics, which compare the generated and original images which can feature highly different pixel values

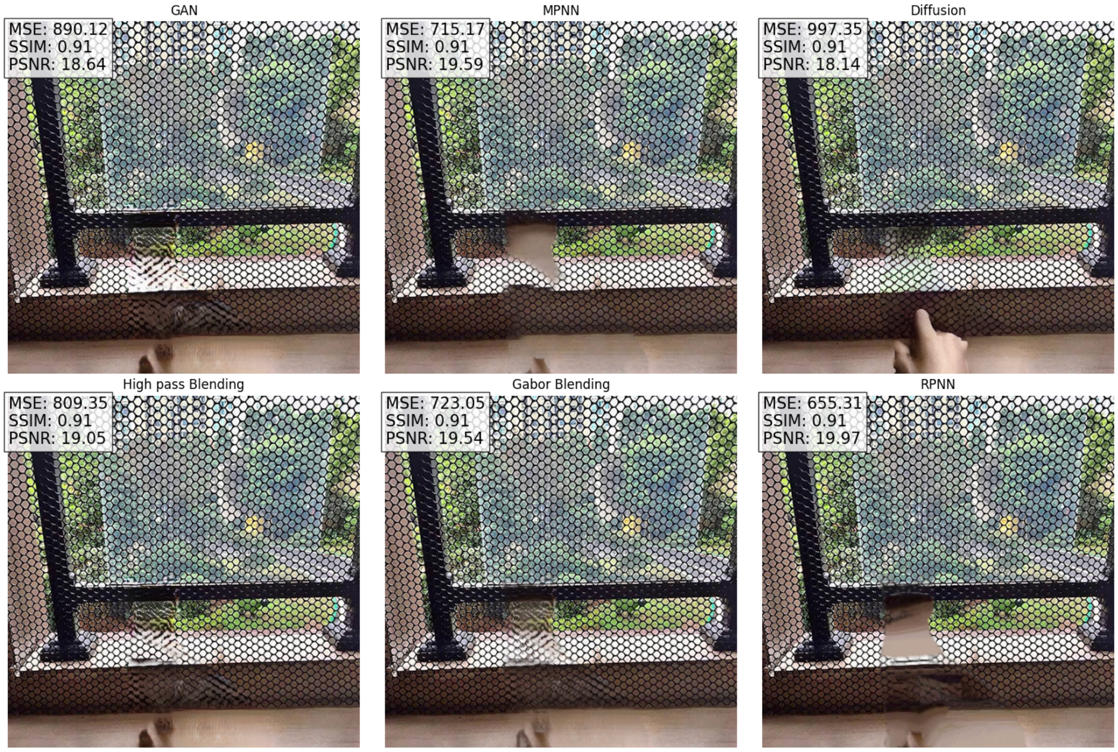
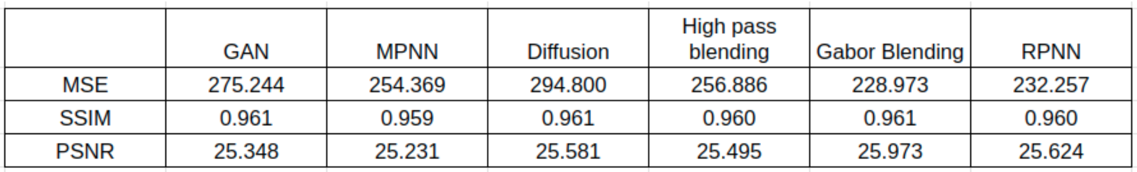


Fig. 5: Comparison of base and hybrid method output



	GAN	MPNN	Diffusion	High pass blending	Gabor Blending	RPNN
MSE	275.244	254.369	294.800	256.886	228.973	232.257
SSIM	0.961	0.959	0.961	0.960	0.961	0.960
PSNR	25.348	25.231	25.581	25.495	25.973	25.624

Fig. 6: Mean quantitative measurements for each method

independent of qualitative consistency assessment.

V. Conclusion

Our study aimed to explore the efficacy of hybrid inpainting methods using GAN and PNN techniques. While quantitative analysis using metrics such as PSNR, MSE, and SSIM provided a foundational understanding of each method's ability to recover the masked regions in source images, it is crucial to acknowledge that such measurements do not adequately correspond to qualitative factors such as textural and structural consistency and plausibility, which we assessed through visual inspection.

Contrary to some assertions in literature such as “Drop the GAN” [5], our findings do not robustly support the notion that patch-based methods can generate results comparable with modern deep learning-based models. This discrepancy partly stem from the unavailability of a working version of their GPNN source code which limited our ability to directly compare methodologies.

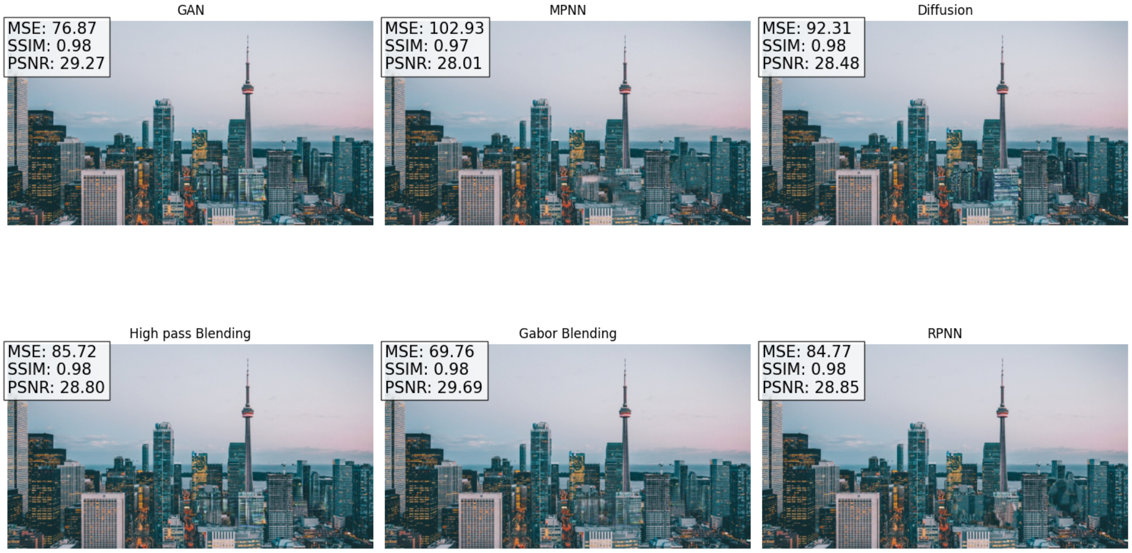


Fig. 7: Inpainting results from various method using a masked photo of Toronto's skyline

Nevertheless, our hybrid methods, such as high pass blending, demonstrated slight improvements in specific scenarios where the the quality of regular repeating textures surpassed the results produced by GAN and diffusion models. These enhancement came at the cost of increased processing time, a challenge that could potentially be mitigated through the implementation of GPU processing.

While our study particularly focused on enhancing texture details, we must acknowledged that an increase in textural density does not necessarily correlate with improved image correctness or plausibility.

Our research suggests that segmentation of the GAN and patch-based output using per-pixel discriminators [8] may produce more a rational and plausible criteria for blending, compared to the more naive methods employed in our study. Furthermore, applying discriminators to the final output could serve as a better quantitative metric for assessing inpainting efficacy than current measurements. This direction presents a promising avenue for future research, potentially addressing the limitations observed in our study.

References

[1] Francesco Bianconi and Antonio Fernández. "Evaluation of the effects of Gabor filter parameters on texture classification". In: *Pattern recognition* 40.12 (2007), pp. 3325–3335.

[2] Majed El Helou and Sabine Süsstrunk. "Bigprior: toward decoupling learned prior hallucination and data fidelity in image restoration". In: *IEEE Transactions on Image Processing* 31 (2022), pp. 1628–1640.

[3] Omar Elharrouss et al. “Image Inpainting: A Review”. In: *Neural Processing Letters* 51.2 (Dec. 2019), pp. 2007–2028. DOI: 10.1007/s11063-019-10163-0. URL: https://doi.org/10.1007%2Fs11063-019-10163-0.

[4] Leilani H. Gilpin et al. *Explaining Explanations: An Overview of Interpretability of Machine Learning*. 2019. arXiv: 1806.00069 [cs.AI].

[5] Niv Granot et al. “Drop the gan: In defense of patches nearest neighbors as single image generative models”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2022, pp. 13460–13469.

[6] Mingtao Guo. *Patchmatch*. https://github.com/MingtaoGuo/PatchMatch. 2018.

[7] Robin Rombach et al. “High-resolution image synthesis with latent diffusion models”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2022, pp. 10684–10695.

[8] Edgar Schönfeld, Bernt Schiele, and Anna Khoreva. *A U-Net Based Discriminator for Generative Adversarial Networks*. 2021. arXiv: 2002.12655 [cs.CV].

[9] Jiahui Yu et al. “Free-form image inpainting with gated convolution”. In: *Proceedings of the IEEE/CVF international conference on computer vision*. 2019, pp. 4471–4480.