# Diffusion-Based Inpainting of Corrupted Spectrogram

**Mahsa Massoud**
Department of Computer Science
McGill University, Mila
mahsa.massoud@mail.mcgill.ca

**Reyhane Askari-Hemmat**
FAIR at Meta, Mila
reyhane.askari.hemmat@umontreal.ca

**Kai-Feng Chen**
Department of Physics, Massachusetts Institute of Technology
kfchen@mit.edu

**Adrian Liu**
Department of Physics and Space Institute
McGill University
adrian.liu2@mcgill.ca

**Siamak Ravanbakhsh**
Department of Computer Science
McGill University, Mila
siamak.ravanbakhsh@mila.quebec

## Abstract

A spectrogram is a visual representation of the spectrum of frequencies in a signal as it varies with time. An important problem in radio astronomy is removing radio frequency interference (RFI) from the spectrogram produced by radio telescopes. Given the competitive performance of diffusion models image inpainting, and the similarity of spectrogram to image data, such methods seem like a natural choice for correcting the corruption due to RFI. Unfortunately, this application is complicated because the entire dataset is corrupted; therefore, existing solutions relying on a clean training set are not readily applicable. Moreover, we observe that in contrast to image data, spectrograms do not have translation symmetry along the frequency axis undermines the prevalent use of ConvNets in processing this kind of data. Fortunately, the RFI corruptions are often local in nature (in time or frequency), meaning we can easily identify and mask the RFI region. In this paper, we investigate a progressively improving series of solutions to the problem of image inpainting, where all the training data is corrupted through masking. Moreover, we propose a positional encoding scheme to break the translation symmetry assumed by ConvNets. We experiment with the CIFAR10 dataset and synthetic spectrogram data, where our initial results are quite supportive.

## 1 Introduction

A spectrogram is an image-like representation of a time-varying signal, with time and frequency as the two dimensions of the image. It is used in various domains, from speech recognition and music analysis to medical imaging and seismology. In this work, we target astrophysical data from the Hydrogen Epoch of Reionization Array (HERA) experiment. HERA is a large radio interferometer consisting of 300 radio dishes, and is located in the South African Karoo Desert [7, 3]. Its goal is to characterize the three-dimensional spatial properties of our Universe around the time that first-generation galaxies were forming. The raw data from an interferometer come in the form of *visibilities*, which are complex-valued data as a function of frequency (of the cosmological radio emission) and time.

Machine Learning and the Physical Sciences Workshop, NeurIPS 2024.

Unfortunately, the low-frequency radio band observed by HERA is also home to human-made radio signals such as radio stations. Collectively, these terrestrial sources are known as Radio Frequency Interference (RFI). RFI is typically orders of magnitude brighter than any astrophysical source of radio waves, and thus parts of the data that are contaminated by RFI are usually taken to be irretrievably ruined and masked. Viewing the visibility as two-dimensional functions living on the frequency-time plane, we therefore have precisely the same problem as with the other datasets: a 2D image with missing parts that need to be inpainted [21]. For the time being, we consider simulations of results from this telescope, which gives us access to the ground truth – i.e., clean data before RFI corruption. However, there are two major differences with the general inpainting problems, which are addressed in this work:

- In contrast with the problem of image in-painting, our entire training data is corrupted by RFI. As we show, naive application of the existing in-painting methods fails simply because it learns the distribution of RFI along with the clean signal. We propose three progressively more accurate solutions to this problem. We evaluate these solutions on image data and spectrograms from the domain of interest.
- While applying ConvNets to spectrogram is a common practice, we observe that in contrast to image data, the spectrogram does not possess translation symmetry along the frequency axis. This is because different frequencies have consistently different characteristics. This lack of symmetry is inconsistent with the assumptions of a convolution layer, and therefore, a ConvNet is overly restricted when it comes to processing spectrograms. To resolve this issue, we propose to use additional channels for positional encoding of frequencies. This solution is analogous to the breaking of permutation symmetry in transformers using the positional encoding of tokens.

## 2 Background

See Appendix for a background on denoising diffusion probabilistic models (DDPM). Here, we discuss RePaint [20], which is an image inpainting algorithm based on DDPM and involves conditioning on known image regions. The inpainting process begins with initializing $x_T$ from a normal distribution $\mathcal{N}(0, I)$. The algorithm then iterates from the last timestep $T$ down to 1. During each timestep, the image $x_t$ is processed to estimate $x_{t-1}$. RePaint assumes that in each step we have a known and an unknown part of the final image, denoted by $x_t^{known}$ and $x_t^{unknown}$ respectively. $x_{t-1}^{known}$ is sampled using:

$$x_{t-1}^{known} \sim \mathcal{N}(\sqrt{\alpha_t} x_0, (1-\alpha_t)I)$$

Here, $\alpha_t$ is a factor that controls how much the starting or (known) state of the image, and $x_0$, affects the process as the algorithm works backward in time.

$x_{t-1}^{unknown}$ is computed based on:

$$x_{t-1}^{unknown} = \frac{1}{\sqrt{\alpha_t}} \left( x_t - \frac{\beta_t}{\sqrt{1-\alpha_t}} \epsilon_\theta(x_t, t) \right) + \sigma_t z$$

where $\epsilon_\theta(x_t, t)$ is the predicted noise for timestep $t$ provided by the neural network, and $z$ is sampled from $\mathcal{N}(0, I)$. The final step for calculating $x_{t-1}$ combines known and unknown regions:

$$x_{t-1} = m \odot x_{t-1}^{known} + (1-m) \odot x_{t-1}^{unknown}$$

where $m$ is a binary mask that indicates known (unmasked) and unknown (masked) regions. This iterative nature helps better estimate the unknown regions based on the model's learning and the known context.

## 3 Method

A minimal baseline for inpainting with corrupted data is to pretend that the dataset is not corrupted. In the following, we introduce progressively more accurate methods that address this problematic assumption.

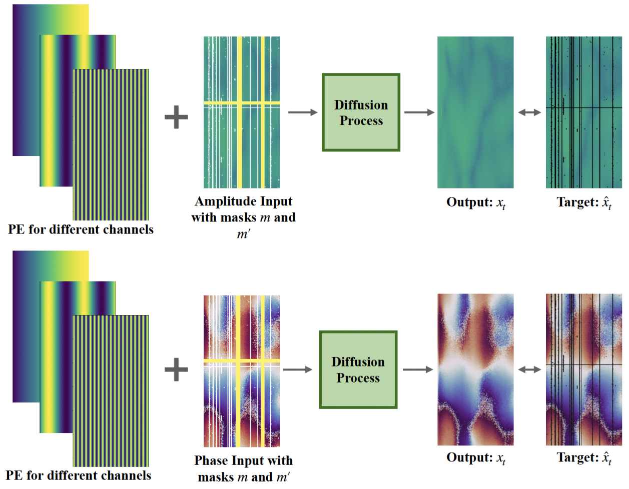
PE for different channels
Amplitude Input with masks $m$ and $m'$
Diffusion Process
Output: $x_t$
Target: $\hat{x}_t$
PE for different channels
Phase Input with masks $m$ and $m'$
Diffusion Process
Output: $x_t$
Target: $\hat{x}_t$

Figure 1: The amplitude and phase of a spectrogram sample. As the input to the diffusion process, we feed the positional encoding along with the mask amplitude and phase. We have two sets of masks $m$ (pre-existing mask in black) and $m'$ (additional masks in yellow). The U-Net diffusion model is then trained by computing $\nabla_\theta(1-m')\alpha l_1 + (1-\alpha)\|x_t - \hat{x}\|^2$ where $l_1 = \|(1-m)x_t - (1-m)\hat{x}\|^2$.

**Method 1: Training with Masked Data** The first step in addressing the corruption in training data is to train the model only on clean parts of the training data. We then use the RePaint algorithm for inpainting, using a score that is trained on clean parts of the input image/spectrogram.

**Method 2: Better Training with Mixed Masking** A major problem with method 1 is that the denoising network has no incentive to denoise the masked regions of the input; we also observed this behavior in practice. To encourage the network to denoise masked regions of the input, we introduce fake random masks. The model is penalized for its mistakes in denoising the regions with a fake mask, where the ground truth is available. Since the model cannot distinguish between two types of mask it attemps to denoise all masked regions. Here again, we used the RePaint algorithm for sampling.

**Method 3: Better Alignment of Training and Sampling** We noted that during the interactive sampling process of RePaint, the signal in masked regions is slowly recovered, and therefore in following iterations the noise is added to this input where the original mask is slowly disappearing. This is different from training conditions where the Gaussian noise is always added to the masked regions that basically have a value of zero. That is we noticed a minor discrepancy between the training and sampling statistics of the input to the denoising network. To mitigate this, we modify the RePaint algorithm.

The algorithms for training and sampling for all methods are detailed in Algorithms 1 and 2, provided below.

Algorithm 1 summarizes the training algorithm for Method 2&3, and Algorithm 1 presents the algorithm for sampling in all Mathods; the change from inpaint in Method 3 is highlighted.

**Algorithm 1** Training

```
Require: masks $m$ (pre-existing mask) and $m'$ (additional mask).
repeat
    $x_0 \sim q(x_0)$
    $t \sim \text{Uniform}(\{1, \dots, T\})$
    $\epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
    $l_1 = \|(1-m)x_t - (1-m)\hat{x}\|^2$
    Take gradient descent step on:
    $\nabla_\theta\big((1-m')\alpha l_1 + (1-\alpha)\|x_t - \hat{x}\|^2\big)$
until converged
```

**Algorithm 2** Inpainting

```
Require: masks $m$ and $m'$.
$x_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
for $t = T, \dots, 1$ do
    for $u = 1, \dots, U$ do
        $\epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ if $t > 1$, else $\epsilon = \mathbf{0}$
        $x_{t-1}^{\text{known}} = \sqrt{\bar{\alpha}_t}x_0 + (1 - \bar{\alpha}_t)\epsilon$
        $z \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ if $t > 1$, else $\mathbf{z} = \mathbf{0}$
        $x'_t = x_t \odot m' + (1 - m')(1 - \bar{\alpha}_t)\epsilon$
        $x_{t-1}^{\text{unknown}} = \frac{1}{\sqrt{\alpha_t}}\left(x'_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\boldsymbol{\epsilon}_\theta(x'_t, t)\right) + \sigma_t z$
        $x_{t-1} = m \odot x_{t-1}^{\text{known}} + (1-m) \odot x_{t-1}^{\text{unknown}}$
        if $u < U$ and $t > 1$ then
            $x_t \sim \mathcal{N}(\sqrt{1-\beta_{t-1}}x_{t-1}, \beta_{t-1}\mathbf{I})$
        end if
    end for
end for
return $x_0$
```

**Positional Encoding** To incorporate spatial positional information in the model, we leverage positional encoding [26]. Given a phase or amplitude tensor $x \in \mathbb{R}^{F \times T \times C}$ where $f \in \{0, \dots, F-1\}$, $t \in \{0, \dots, T\}$ and $c \in \{0, \dots, C-1\}$ are the frequency, time and channels. positional encodings are independent of time, and are computed as follows:

$$\text{PE}(f, t, c) = \sin\left(\frac{f \times \pi}{2F^{(c+1)/C}}\right) \tag{1}$$

PE allows different channels to encode different frequencies.

## 4 Experiments

We evaluated our method on three datasets: DermaMNIST, CIFAR10, and a synthetic spectrogram dataset [1]. The results of DermaMNIST can be found in the appendix.

**CIFAR 10** We corrupt the CIFAR10 [19] dataset using random masks and evaluate the inpainted images using the FID score. Table 1 shows that our methods are on par with the clean baseline, in terms of the FID compared. The clean baseline uses uncorrupted data. We significantly outperform the naive approach of ignoring the corruption in the data (corrupted baseline); also see Fig. 4 in the Appendix.

Table 1: Inpainting results for CIFAR10

| Method | FID score |
|---|---|
| Baseline (No mask) | 3.81 |
| Corrupted Baseline | 18.54 |
| Method 1 | 3.834 |
| Method 2 | 3.817 |
| Method 3 | 3.82 |

Table 2: Results for Synthetic spectrograms.

| Method | Average PSNR |
|---|---|
| Baseline (No mask) | 56.9176 |
| Method 1 | 46.9958 |
| Method 2 | 47.2915 |
| Method 3 | 47.7325 |
| Method 3 + PE | 48.3748 |

**Synthetic Spectrogram** The dataset is generated through simulation using the Vispb software [27] and contains only 191 clean samples of ($T = 640, F = 480$) complex-valued pixels. More plots of synthetic data are in the Appendix. We use PSNR (peak signal-to-noise ratio) for the evaluation of the

[1] All experimentation and processing was conducted by the authors at McGill and MIT Universities on their servers.

synthetic data as we have access to the ground truth values for the masked parts of the spectrogram. As seen in Table 2, using the combination of positional encoding and method three produces the best results.

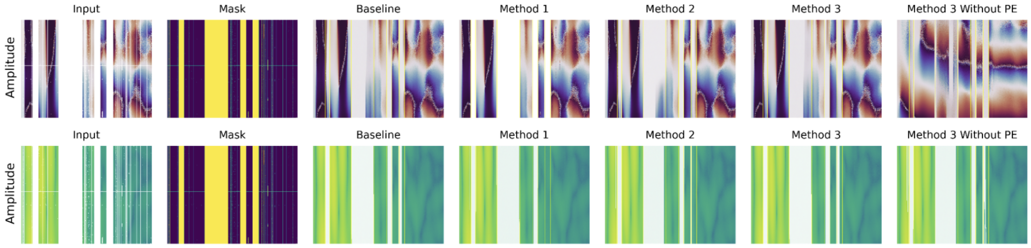

Figure 2: Inpainted example from the spectrogram dataset. On the left, the clean sample and the mask applied to it are plotted. We then show the final inpainted image using different methods. Yellow parts indicate masks of the image, and what is important astrophysical is actually the prediction for narrow band masks (the big yellow parts are consistent masks that are completely ignored)

### 4.1 CIFAR-10

See the visual inpainting samples of the CIFAR-10 dataset in Figure 4.

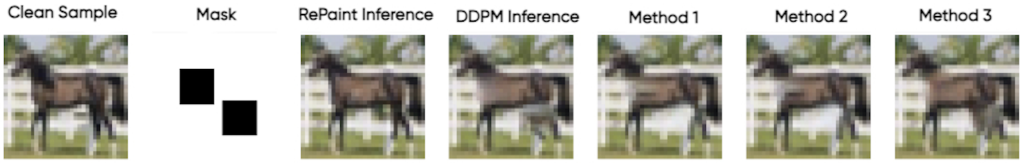

Figure 3: Inpainted example from the CIFAR-10 dataset. On the left the clean sample and the mask applied to it are plotted. We then show the final inpainted image using different methods.

## Conclusion

We presented simple solutions using a diffusion probabilistic model for replacing RFI-corrupted regions of data in radio astronomy. Our solutions address the problem of the unavailability of clean data in this setting and build on an existing method for diffusion-based image inpainting. Moreover, we identify an important distinction between image and spectrogram data, in terms of their symmetry assumption and propose a solution, that may be of independent interest in many other domains, such as speech and audio processing where spectrograms are used. Experiments on toy image datasets as well as synthetic astronomical data paint a favourable picture for our methods when compared against simple baselines that provide upper and lower bounds on inpainting performance. In the future, we plan to extensively compare our method against existing solutions for RFI inpainting in radio astronomy, and thoroughly validate its performance in this target domain.

## References

[1] Coloma Ballester et al. Filling-in by joint interpolation of vector fields and gray levels. *IEEE Transactions on Image Processing*, 10(8):1200–1211, 2001.

[2] Connelly Barnes, Eli Shechtman, Adam Finkelstein, and Dan B. Goldman. PatchMatch: A randomized correspondence algorithm for structural image editing. *ACM Transactions on Graphics*, 28(3):24, 2009.

[3] Lindsay M. Berkhout, Daniel C. Jacobs, Zuhra Abdurashidova, Tyrone Adams, James E. Aguirre, Paul Alexander, Zaki S. Ali, Rushelle Baartman, Yanga Balfour, Adam P. Beardsley, Gianni Bernardi, Tashalee S. Billings, Judd D. Bowman, Richard F. Bradley, Philip Bull, Jacob Burba, Ruby Byrne, Steven Carey, Chris L. Carilli, Kai-Feng Chen, Carina Cheng, Samir Choudhuri, David R. DeBoer, Eloy de Lera Acedo, Matt Dexter, Joshua S. Dillon, Scott Dynes, Nico Eksteen, John Ely, Aaron Ewall-Wice, Nicolas Fagnoni, Randall Fritz, Steven R. Furlanetto, Kingsley Gale-Sides, Hugh Garsden, Bharat Kumar Gehlot, Abhik Ghosh, Brian Glendenning, Adelie Gorce, Deepthi Gorthi, Bradley Greig, Jasper Grobbelaar, Ziyaad Halday, Bryna J. Hazelton, Jacqueline N. Hewitt, Jack Hickish, Tian Huang, Alec Josaitis, Austin Julius, MacCalvin Kariseb, Nicholas S. Kern, Joshua Kerrigan, Honggeun Kim, Piyanat Kittiwisit, Saul A. Kohn, Matthew Kolopanis, Adam Lanman, Paul La Plante, Adrian Liu, Anita Loots, Yin-Zhe Ma, David Harold Edward MacMahon, Lourence Malan, Cresshim Malgas, Keith Malgas, Bradley Marero, Zachary E. Martinot, Andrei Mesinger, Mathakane Molewa, Miguel F. Morales, Tshegofalang Mosiane, Steven G. Murray, Abraham R. Neben, Bojan Nikolic, Chuneeta Devi Nunhokee, Hans Nuwegeld, Aaron R. Parsons, Robert Pascua, Nipanjana Patra, Samantha Pieterse, Yuxiang Qin, Eleanor Rath, Nima Razavi-Ghods, Daniel Riley, James Robnett, Kathryn Rosie, Mario G. Santos, Peter Sims, Saurabh Singh, Dara Storer, Hilton Swarts, Jianrong Tan, Nithyanandan Thyagarajan, Pieter van Wyngaarden, Peter K. G. Williams, Haoxuan Zheng, and Zhilei Xu. Hydrogen Epoch of Reionization Array (HERA) Phase II Deployment and Commissioning. *Publications of the Astronomical Society of the Pacific*, 136(4):045002, April 2024.

[4] Marcelo Bertalmío, Li-Tien Cheng, Stanley Osher, and Guillermo Sapiro. Variational problems and partial differential equations on implicit surfaces. *Journal of Computational Physics*, 174(2):759–780, 2001.

[5] Ciprian Corneanu, Raghudeep Gadde, and Aleix M Martinez. Latentpaint: Image inpainting in latent space with diffusion models. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 4334–4343, 2024.

[6] Soheil Darabi et al. Image melding: Combining inconsistent images using patch-based synthesis. *ACM Transactions on Graphics (TOG)*, 31(4):1–10, 2012.

[7] David R. DeBoer, Aaron R. Parsons, James E. Aguirre, Paul Alexander, Zaki S. Ali, Adam P. Beardsley, Gianni Bernardi, Judd D. Bowman, Richard F. Bradley, Chris L. Carilli, Carina Cheng, Eloy de Lera Acedo, Joshua S. Dillon, Aaron Ewall-Wice, Gcobisa Fadana, Nicolas Fagnoni, Randall Fritz, Steve R. Furlanetto, Brian Glendenning, Bradley Greig, Jasper Grobbelaar, Bryna J. Hazelton, Jacqueline N. Hewitt, Jack Hickish, Daniel C. Jacobs, Austin Julius, MacCalvin Kariseb, Saul A. Kohn, Telalo Lekalake, Adrian Liu, Anita Loots, David MacMahon, Lourence Malan, Cresshim Malgas, Matthys Maree, Zachary Martinot, Nathan Mathison, Eunice Matsetela, Andrei Mesinger, Miguel F. Morales, Abraham R. Neben, Nipanjana Patra, Samantha Pieterse, Jonathan C. Pober, Nima Razavi-Ghods, Jon Ringuette, James Robnett, Kathryn Rosie, Raddwine Sell, Craig Smith, Angelo Syce, Max Tegmark, Nithyanandan Thyagarajan, Peter K. G. Williams, and Haoxuan Zheng. Hydrogen Epoch of Reionization Array (HERA). *Publications of the Astronomical Society of the Pacific*, 129(974):045001, April 2017.

[8] Alexei A. Efros and Thomas K. Leung. Texture synthesis by non-parametric sampling. In *Proceedings of the Seventh IEEE International Conference on Computer Vision*, volume 2, pages 1033–1038. IEEE, September 1999.

[9] Aaron Ewall-Wice, Nicholas Kern, Joshua S Dillon, Adrian Liu, Aaron Parsons, Saurabh Singh, Adam Lanman, Paul La Plante, Nicolas Fagnoni, Eloy de Lera Acedo, David R DeBoer, Chuneeta Nunhokee, Philip Bull, Tzu-Ching Chang, T Joseph W Lazio, James Aguirre, and Sean Weinberg. <tt>dayenu:</tt> a simple filter of smooth foregrounds for intensity mapping power spectra. *Monthly Notices of the Royal Astronomical Society*, 500(4):5195–5213, October 2020.

[10] Abhik Ghosh, Florent Mertens, Gianni Bernardi, Mário G. Santos, Nicholas S. Kern, Christopher L. Carilli, Trienko L. Grobler, et al. Foreground modelling via Gaussian process regression: an application to HERA data. *Monthly Notices of the Royal Astronomical Society*, 495(3):2813–2826, 2020.

[11] Asya Grechka, Guillaume Couairon, and Matthieu Cord. Gradpaint: Gradient-guided inpainting with diffusion models. *Computer Vision and Image Understanding*, 240:103928, 2024.

[12] Qiang Guo, Shanshan Gao, Xiaofeng Zhang, Yilong Yin, and Caiming Zhang. Patch-based image inpainting via two-stage low rank approximation. *IEEE Transactions on Visualization and Computer Graphics*, 24(6):2023–2036, 2017.

[13] Jan Herling and Wolfgang Broll. High-quality real-time video inpainting with PixMix. *IEEE Transactions on Visualization and Computer Graphics*, 20(6):866–879, 2014.

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

[15] J. A. Högbom. Aperture Synthesis with a Non-Regular Distribution of Interferometer Baselines. *Astronomy and Astrophysics Supplement Series*, 15:417, 1974.

[16] Bahjat Kawar, Noam Elata, Tomer Michaeli, and Michael Elad. Gsure-based diffusion model training with corrupted data. *arXiv preprint arXiv:2305.13128*, 2023.

[17] Nicholas S Kern and Adrian Liu. Gaussian process foreground subtraction and power spectrum estimation for 21cm cosmology. *Monthly Notices of the Royal Astronomical Society*, 501(1):1463–1480, December 2020.

[18] Jaeho Kim, Seungwoo Lee, Jiwon Kim, Joonseok Kim, and Kyounghoon Lee. Smartbrush: Text and shape guided object inpainting with diffusion model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–10, 2022.

[19] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

[20] Andreas Lugmayr, Martin Danelljan, Andres Romero, Fisher Yu, Radu Timofte, and Luc Van Gool. Repaint: Inpainting using denoising diffusion probabilistic models, 2022.

[21] Michael Pagano et al. Characterization of inpaint residuals in interferometric measurements of the epoch of reionization. *Monthly Notices of the Royal Astronomical Society*, 520(4):5552–5572, 2023.

[22] Chitwan Saharia, William Chan, Huiwen Chang, Chris Lee, Jonathan Ho, Tim Salimans, David Fleet, and Mohammad Norouzi. Palette: Image-to-image diffusion models. In *ACM SIGGRAPH 2022 Conference Proceedings*, pages 1–10, 2022.

[23] David Slepian. Prolate spheroidal wave functions, Fourier analysis, and uncertainty—V: The discrete case. *Bell System Technical Journal*, 57(5):1371–1430, 1978.

[24] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*, pages 2256–2265. PMLR, 2015.

[25] David Tschumperlé and Rachid Deriche. Vector-valued image regularization with PDEs: A common framework for different applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(4):506–517, 2005.

[26] A Vaswani. Attention is all you need. *Advances in Neural Information Processing Systems*, 2017.

[27] Vispb. Vispb, 2022. Available online: `https://github.com/vispb/vispb`.

[28] Yifan Yan, Yifan Zhang, Yujia Chen, Zongyu Huang, Shuang Wang, and Shijie Xu. Latentpaint: Image inpainting in latent space with diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–10, 2022.

[29] Jiancheng Yang, Rui Shi, Donglai Wei, Zequan Liu, Lin Zhao, Bilian Ke, Hanspeter Pfister, and Bingbing Ni. Medmnist v2-a large-scale lightweight benchmark for 2d and 3d biomedical image classification. *Scientific Data*, 10(1):41, 2023.

[30] Shiyuan Yang, Xiaodong Chen, and Jing Liao. Uni-paint: A unified framework for multimodal image inpainting with pretrained diffusion model. In *Proceedings of the 31st ACM International Conference on Multimedia*, pages 3190–3199, 2023.

# 5 Supplementary

## 5.1 Related Work

**Inpainting:** Inpainting is the process of filling in missing or damaged parts of images and is one of the major challenges in computer vision. Initial techniques for image inpainting relied on the low-level features of corrupted images. These include methods based on partial differential equations (PDEs) ([4]; [1]; [25]) as well as patch-based methods ([8]; [2]; [6]; [14]; [13]; [12]).

Those techniques derived from PDEs tend to propagate the information from the edges and boundaries to achieve smooth inpainted images. While these methods handle boundaries well, reconstructing the image textures is still difficult. On the other hand, patch-based strategies do not solely focus on the edges and boundaries; they fill the unknown parts by finding and replicating similar patches from the known areas. Patch-based methods can fill in textures and even whole objects in small gaps if they find similar ones somewhere else in the image. Despite their strengths, PDE-based methods often fail to reproduce complex textures, and patch-based methods struggle with larger masks or highly irregular areas. The introduction of diffusion models, especially highlighted in the "RePaint" paper [20], represents a major breakthrough in this area. These models employ a pretrained diffusion model and incorporate a defined mask during the inference time to iteratively denoise images, enabling them to fill in gaps with remarkable accuracy and detail that align with the surrounding context.

More recently, it has shown that for cases where the training data is corrupted, one can use the GSURE loss [16]. The method achieves comparable results to supervised models, reducing data requirements for tasks like such as MRI reconstruction where the acquiring data is expensive.

**RFI Inpainting:** The CLEAN algorithm [15] iteratively identifies and subtracts the brightest points (or "clean components") in the observed data, then reconstructs the image by adding these components back in a more accurate way. So, this can reduce the effect of RFI and other noise sources.

In the HERA data analysis pipeline [21], authors have used a modified version where of CLEAN tailored to their inpainting requirements; Moreover, they investigated some other techniques including least square spectral analysis (LSSA), Gaussian process regression (GPR) [10] [17], and discrete prolate spheroidal sequences (DPSS) [23] [9].

These approaches leverage uncorrupted data to construct a basic model for the corrupted data, which is then substituted into the RFI-flagged regions; This can help mitigate the impact of RFI on the spectrum. However, these restored data may cause some problems and introduce possible errors in the analysis.

In addition to methods for specific RFI inpainting, there are generally three types of inpainting methods with the diffusion models that we discuss below.

**Methods that Modify the sampling:** in [11] authors introduce a method called GradPaint, which uses a custom loss to measure the coherence of the denoised image estimation with the masked input image at each step in the denoising process. In [30] authors suggest a different approach to sampling. They fine-tune pretrained diffusion models without any conditioning initially; Then during sampling, the models can then be conditioned on different modalities, such as text. Moreover, [20] used the same approach of modifying only sampling.

**Denoising Diffusion Probabilistic Models:** The concept of DDPMs was first introduced by [24] and since then it has been used in a lot of applications of computer vision. For a full description of the DDPM algorithm, see Appendix 5.2.1

**Methods that Modify the training:** In [22], authors guide the model to fill in masks based on the context provided by the existing parts of the image. During the training, loss functions are designed to ensure that the generated images are both high-quality and contextually consistent. Also, in [28] they modify the latent space using diffusion models, to enhance the efficiency of the inpainting process.

**Methods that modify both sampling and training:** In [5] authors propose a new approach for image inpainting using diffusion models that does not require expensive training and is fast at inference time. This is achieved by performing the forward-backward fusion step in the latent space rather than the image space. [18] introduces a multi-modal object inpainting approach that utilizes text and shape guidance with diffusion models to improve inpainting quality and efficiency.

## 5.2 Background

### 5.2.1 Denoising Diffusion Probabilistic Models (DDPM)

We first introduce the mathematical notation used throughout the paper. Let $\mathbf{x}$ denote the input data, $\mathbf{y}$ represent the output or target data, and $\theta$ denote the parameters of the model. We use $\mathcal{D}$ to represent the dataset consisting of input-output pairs $\{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^N$, where $N$ is the number of samples in the dataset.

Denoising Diffusion Probabilistic Models (DDPM) are a class of generative models that learn to model the conditional distributions of data under sequential levels of diffusion. Given an initial noisy sample, DDPM iteratively applies denoising steps to reduce the noise level until reaching the desired distribution. Formally, DDPM models the conditional distribution $p(\mathbf{x}_t|\mathbf{x}_0)$, where $\mathbf{x}_t$ is the data at time step $t$ and $\mathbf{x}_0$ is the initial noisy input.

During training, DDPM methods define a diffusion process that transforms an image $x_0$ to white Gaussian noise $(x_T \sim N(0,1))$ in T time steps.

The forward process of adding Gaussian noise is given by

$$q(x_t|x_{t-1}) = \mathcal{N}(x_t; \sqrt{1-\beta_t}x_{t-1}, \beta_t\mathbf{I}) \tag{2}$$

The DDPM is trained to reverse the process in (1). The reverse process is modeled by a neural network that predicts the parameters $\mu_\theta(x_t, t)$ and $\Sigma_\theta(x_t, t)$ of a Gaussian distribution; The reverse process of denoising is modeled by

$$p_\theta(x_{t-1}|x_t) = \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \Sigma_\theta(x_t, t)) \tag{3}$$

## 5.3 Experiments

### 5.3.1 Model Architecture and Hyperparameters

We utilized a U-Net architecture for our model, which is well-suited for image reconstruction tasks. Our hyperparameter search included varying the learning rate, batch size, and the number of filters in each convolutional layer. The final chosen hyperparameters were as follows: Learning rate: 0.001, Batch size: 32, Number of filters: 64, 128, 256, 512, Optimizer: Adam.

## 5.4 RFI dataset

For better clarification, we depict the plot for the masks that we actually care about being filled with our methods. The red masks below showcase the differences in gaps between two consecutive nights. which are being fully recovered in the figure 2.

## 5.5 Synthetic RFI dataset

See the visual inpainting samples of the synthetic in Figure 5.

### 5.5.1 DermaMNIST Dataset

DermaMNIST is a collection of medical images similar in structure to the MNIST dataset that contains Dermatoscope data [29]. We evaluate our method on this dataset. The results can be found in the appendix.

For the DermaMNIST dataset, the performance metrics are summarized in Table 3 and the visualizations are depicted in Figure 6.

We used the following range of hyper-paramters to tune the model for the Derma-MNIST dataset: Learning Rates: {0.0001, 0.00001, 0.000001}, Coefficients: {0.3, 0.6, 0.9}, Batch Sizes: {16, 32, 64, 128}.

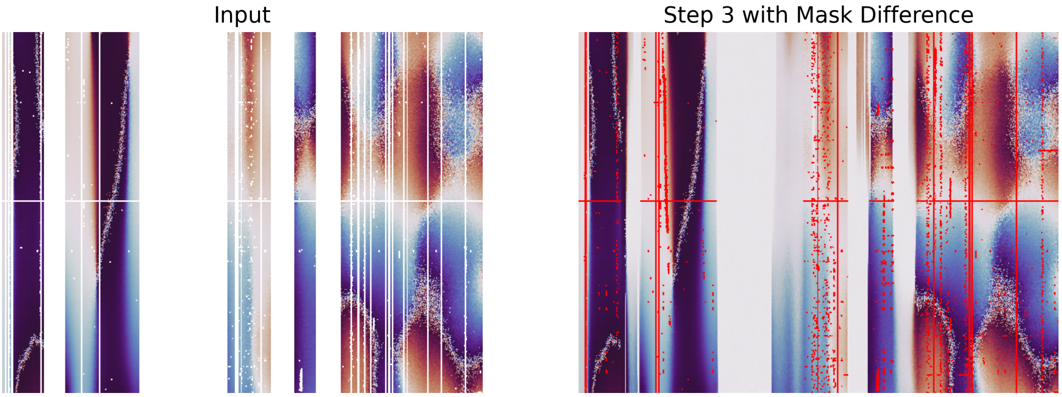

Figure 4: Inpainted example from the real data spectrogram dataset. On the left, the real data sample is the input and the right figure shows the final inpainted image using different methods. Red parts indicate the narrow band masks.

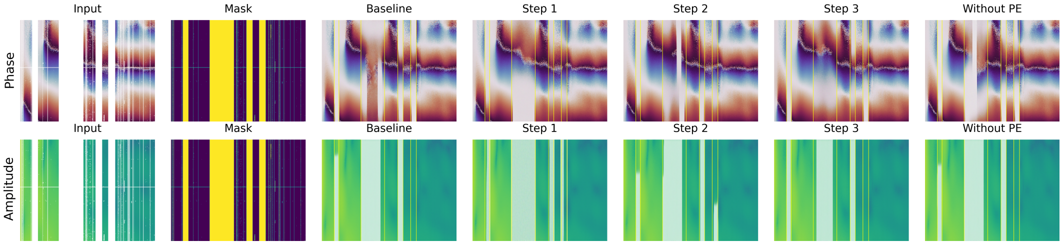

Figure 5: Inpainted example from the real data spectrogram dataset. On the left, the clean sample and the mask applied to it are plotted. We then show the final inpainted image using different methods. Yellow parts indicate masks of the image, and what is important astrophysical is actually the prediction for narrow band masks (the big yellow parts are considered as unknown areas)

| Method | Average PSNR | Average MSE | Average Std (mask) |
|---|---|---|---|
| Baseline | 47.9335 | 1.9273 | - |
| Step 1 | 46.7960 | 3.0519 | 1.294018 |
| Step 2 | 47.8519 | 2.2406 | 1.275268 |

Table 3: Performance metrics for DermaMNIST dataset

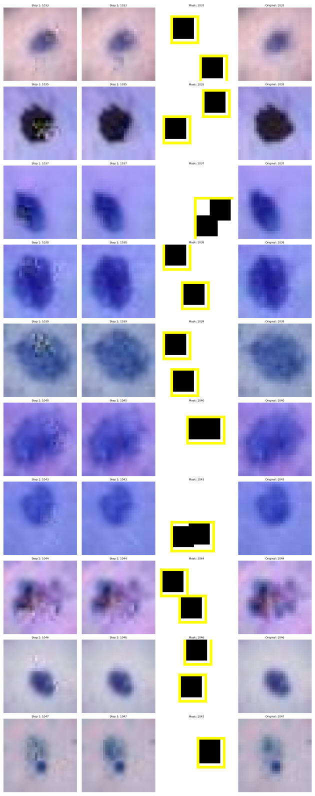

Figure 6: Results on the DermaMNIST dataset