
WAVEPAINT: RESOURCE-EFFICIENT TOKEN-MIXER FOR SELF-SUPERVISED INPAINTING

Pranav Jeevan, Dharshan Sampath Kumar, Amit Sethi

Department of Electrical Engineering
Indian Institute of Technology Bombay
Mumbai, India

{pranav13phoenix, dharshan2609 }@gmail.com

ABSTRACT

Image inpainting, which refers to the synthesis of missing regions in an image, can help restore occluded or degraded areas and also serve as a precursor task for self-supervision. The current state-of-the-art models for image inpainting are computationally heavy as they are based on transformer or CNN backbones that are trained in adversarial or diffusion settings. This paper diverges from vision transformers by using a computationally-efficient WaveMix-based fully convolutional architecture – WavePaint. It uses a 2D-discrete wavelet transform (DWT) for spatial and multi-resolution token-mixing along with convolutional layers. The proposed model outperforms the current state-of-the-art models for image inpainting on reconstruction quality while also using less than half the parameter count and considerably lower training and evaluation times. Our model even outperforms current GAN-based architectures in CelebA-HQ dataset without using an adversarially trainable discriminator. Our work suggests that neural architectures that are modeled after natural image priors require fewer parameters and computations to achieve generalization comparable to transformers.

Keywords Image inpainting · Wavelet transform · Token-mixing · image generation



Figure 1: A sample of inpainted images (bottom row) generated by WavePaint from masked images (top row) from CelebA-HQ set using wide, medium, and narrow masks, respectively

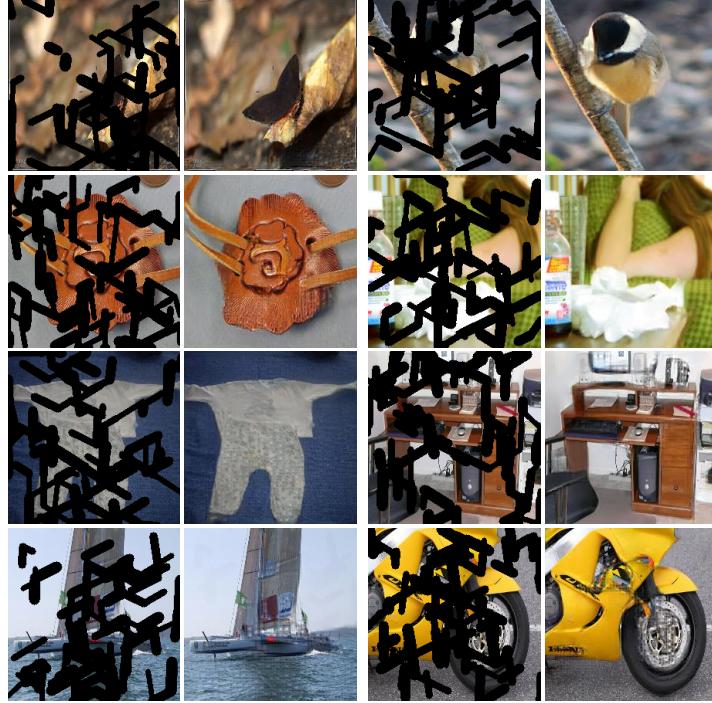


Figure 2: Inpainted images (second and fourth columns) generated by WavePaint from masked images (first and third columns) from ImageNet validation set

1 Introduction

Image inpainting refers to the process of filling of missing parts of an image (blemishes, holes, and other defects) realistically to match the available context, thereby restoring a degraded image. It requires implicitly modeling large scale structures in natural images and an ability to perform image synthesis. State-of-the-art inpainting models are based on deep neural networks trained in a self-supervised and adversarial manner by automatically generating training samples from large image datasets by randomly masking parts of the image.

Some image reconstruction tasks, such as inpainting with large masks, require networks to have large effective receptive fields [1]. Convolutional neural networks (CNN) require deep architectures (a large number of layers) for increasing the receptive fields. On the other hand, using self-attention to access all the pixels of an image right from the first layer gives transformers large receptive fields. However, the quadratic complexity with respect to sequence length (number of patches) introduces an enormous computational burden on transformers. Moreover, transformers require larger training data than CNNs since they lacks the inductive bias of spatial equivariance.

The search for efficient models that can mix global spatial information while retaining the inductive bias of CNNs has led to the development of token-mixing models such as PoolFormer [2], ConvMixer [3] and WaveMix [4] which use pooling, depth-wise convolutions and 2D-discrete wavelet transform (2D-DWT), respectively. These alternatives consume a fraction of the resources compared to transformers to achieve competitive generalization in tasks such as classification and segmentation. The performance of these models on image generation or restoration tasks has not been evaluated.

Our model is a neural architecture that is inspired by WaveMix [4] and ConvMixer [3]. We investigated the application of WaveMix architectural framework to the task of image inpainting with suitable adaptations to the previously proposed architectures. This choice is motivated by the success of WaveMix in approaching the state-of-the-art (SOTA) for different datasets on the task of parameter-efficient image classification and segmentation by modeling additional inductive priors of images, such as scale invariance.

Specifically, we have worked on large mask inpainting, where the mask occlude a substantial and non-trivial part of the image, but its shape is known. We have not worked on blind mask inpainting where the model does not see the mask. Sending mask to the model is necessary in the large-mask setting for the model to know where the mask is and where to fill information.

Our contributions are summarized below:

- We present – WavePaint – a token-mixing network modeled after natural image priors that can perform image inpainting. The network is based on recently proposed WaveMix architecture which uses 2D-discrete wavelet transform for spatial token-mixing. We also employ depth-wise convolution in our network for additional token-mixing. The presence of spatial token-mixing enables the model to have faster receptive field expansion compared to CNNs, which helps in better image reconstruction through access to global context of the image.
- The use of a parameter-free 2D-DWT and parameter-efficient depth-wise convolution helps WavePaint reconstruct images without the need for large number of model parameters. WavePaint with 5M parameters can outperform much larger models such as LaMa(27 M) and CoModGAN (109 M) [5] on CelebA-HQ dataset in multiple mask sizes. It is able to achieve these results consuming less resources and time.
- WavePaint does not need adversarial or diffusion based training routines, which are slow. The ability of wavelet token mixing to generate realistic images from masked ones shows that we can develop more efficient neural networks for image generation.
- Complicated multi-stage models have been proposed that generate intermediate predictions which are further processed to restore the missing parts [6, 7, 8]. Our model reconstructs the image using a simple single-stage network.
- We show that utilizing natural image priors in neural architectural design may be the way forward to avoid large computational costs and training datasets.

2 Related Works

Mask-Aware Dynamic Filtering (MADF) [9] uses an encoder-decoder framework to learn multi-scale features for missing regions in the encoding phase. It adopts Point-wise Normalization (PN) in decoding phase by considering the statistical nature of features at masked points. It does not use adversarial training using a discriminator.

2.1 Generative Adversarial Networks

Co-ModGAN [5] is a GAN model which introduces variability into the generated outputs by integrating input image-conditional and unconditional generators. It combines an unconditional style vector with an input-conditioned style vector through a linear transformation into a single modulated output. The conditional vector is obtained from an encoder network, and the unconditional vector is obtained by passing a noise vector through a pre-trained FCN, as done in Style GAN [10]. Finally, this combined output is passed through the decoder to generate the output.

Image completion with transformer (ICT) [11] is a transformer -CNN hybrid model that uses transformers to model the long-range relationships in images to recover pluralistic coherent structures together with coarse textures, and uses CNN for texture replenishment.

Mask-Aware Transformer [12] uses a multi-head contextual attention for long-range dependency modeling by exploiting valid tokens indicated by a dynamic mask for directly processing high-resolution images. It also proposed a modified transformer block to increase the stability of large mask training.

LaMa [13] uses Fast Fourier Convolution (FFC) blocks to understand the local and global context of an image. The use of FFC helps in having an image-wide receptive field. The use of FFC can be considered as a token-mixing operation similar to WaveMix where fast fourier transform is used for spatial token-mixing. It also uses a high receptive field perceptual loss and large training masks.

WaveFill [14] uses 2D-DWT to decompose images into multiple frequency bands and fills the missing regions in each frequency band separately. It applies L_1 reconstruction loss to the decomposed low-frequency bands and adversarial loss to high-frequency bands to mitigate inter-frequency conflicts and also uses a normalization scheme to align multi-frequency features.

2.2 Diffusion Models

Diffusion models uses a T fold pass through a fixed network to go from completely random noise to a coherent and contextually consistent image. Even though the overall performance of diffusion models are excellent, the training and inference process are extremely time-consuming.

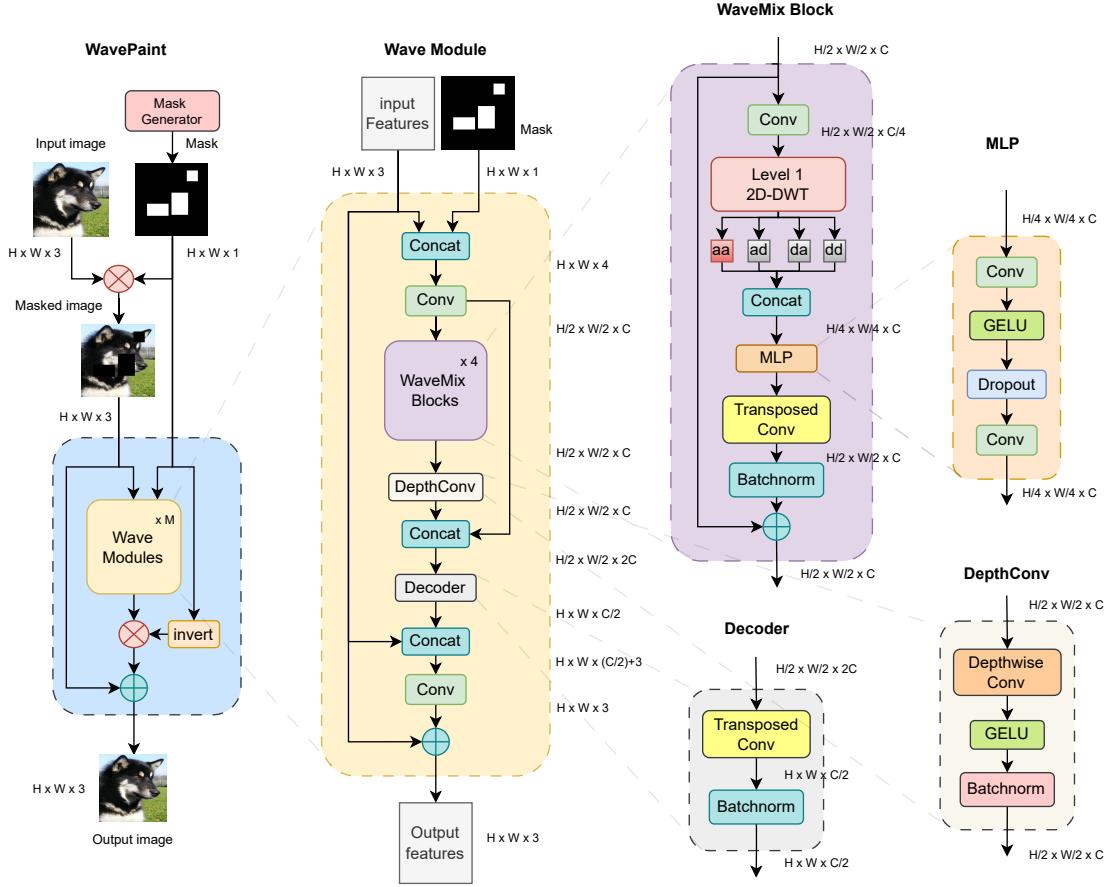


Figure 3: Architecture of WavePaint along with details of Wave module, WaveMix block, Decoder, DepthConv and MLP are shown. The resolutions of feature maps after each operation is also provided for an input of $H \times W \times 3$. WaveMix block is taken from [4]

Latent diffusion model (LDM) [15] works on a lower-dimensional feature space rather than the image space to address the time-consuming nature of diffusion training. The model uses an encoder-decoder architecture with the slow diffusion step at the neck of the chain to speed up the entire network.

RePaint [16] is a denoising diffusion probabilistic model (DDPM) based inpainting approach which employs a pretrained unconditional DDPM as the generative prior. It only alters the reverse diffusion iterations by sampling the unmasked area to condition the generation process. Additionally, they perform re-sampling on the generated output at each step, by noising and successively de-noising a fixed number of times, in order to get a coherent image. Thus. the model produces high quality and diverse output images for any masked images.

3 WavePaint Architectural Framework

Observing the success of WaveMix and ConvMixer which uses 2D-DWT and depthwise-convolutions respectively for parameter efficient token-mixing, we have created a neural architecture that can inpaint masked images using these token-mixing operations. The ability of these token-mixers to impart rapid receptive field expansion from initial layers itself helps the model grasp the global context faster than conventional CNN-based networks. Unlike other popular models for image inpainting that uses diffusion or adversarial training, our model has simple single network architecture and can perform well without the need for a discriminator network.

3.1 Overall architecture

The input image $\mathbf{x} \in \mathbb{R}^{H \times W \times 3}$ is masked by a binary mask $m \in \mathbb{R}^{H \times W \times 1}$ that is generated from a mask generator. The masked image is denoted as $\mathbf{x} \oplus m$. The mask m is concatenated with the masked image $\mathbf{x} \oplus m$, resulting in a 4-channel input $\hat{\mathbf{x}} \in \mathbb{R}^{H \times W \times 4}$ that is passed to the model as shown in Figure 3.

The network consists of a series of M Wave modules which processes the input $\hat{\mathbf{x}}$ and gives the output $\hat{\mathbf{y}} \in \mathbb{R}^{H \times W \times 3}$. $\hat{\mathbf{y}}$ is multiplied by the inverted binary mask, $1 - m$ to hide the unmasked areas of the output and retains the inpainted parts by the model. This is added back to the masked image $\hat{\mathbf{x}}$ which fills the unmasked areas and creates the final inpainted image $\mathbf{y} \in \mathbb{R}^{H \times W \times 3}$. This ensures that the model only fills the masks areas and not change pixel information of unmasked parts.

3.2 Wave Modules

Proper inpainting requires global context information of the image. WaveMix has shown rapid expansion of receptive fields from very early layers [4]. So we use 4 WaveMix blocks in series in each of the Wave modules to process the image and get global context. This is further aided by the depth-wise convolution layer which further helps with spatial token-mixing with high parameter-efficiency [3].

Denoting input and output tensors of the Wave module by $\hat{\mathbf{x}}_{in}$ and $\hat{\mathbf{x}}_{out}$, respectively; convolution operations by c_1 and c_2 and its respective trainable parameter sets by θ_1 and θ_2 respectively; the series of WaveMix blocks by WB ; DepthConv by DC ; Decoder by D ; concatenation along the channel dimension by \oplus , and point-wise addition by $+$, the operations inside a Wave module can be expressed using the following equations:

$$\hat{\mathbf{x}}_0 = \hat{\mathbf{x}}_{in} \oplus m; \quad \hat{\mathbf{x}}_{in} \in \mathbb{R}^{H \times W \times 4} \quad (1)$$

$$\hat{\mathbf{x}}_1 = c_1(\hat{\mathbf{x}}_0, \theta_1); \quad \hat{\mathbf{x}}_1 \in \mathbb{R}^{H/2 \times W/2 \times C} \quad (2)$$

$$\hat{\mathbf{x}}_2 = WB(\hat{\mathbf{x}}_1); \quad \hat{\mathbf{x}}_2 \in \mathbb{R}^{H/2 \times W/2 \times C} \quad (3)$$

$$\hat{\mathbf{x}}_3 = DC(\hat{\mathbf{x}}_2); \quad \hat{\mathbf{x}}_3 \in \mathbb{R}^{H/2 \times W/2 \times C} \quad (4)$$

$$\hat{\mathbf{x}}_4 = \hat{\mathbf{x}}_3 \oplus \hat{\mathbf{x}}_1; \quad \hat{\mathbf{x}}_4 \in \mathbb{R}^{H/2 \times W/2 \times 2C} \quad (5)$$

$$\hat{\mathbf{x}}_5 = D(\hat{\mathbf{x}}_4); \quad \hat{\mathbf{x}}_5 \in \mathbb{R}^{H \times W \times C/2} \quad (6)$$

$$\hat{\mathbf{x}}_6 = \hat{\mathbf{x}}_5 \oplus \hat{\mathbf{x}}_{in}; \quad \hat{\mathbf{x}}_6 \in \mathbb{R}^{H \times W \times (C/2+3)} \quad (7)$$

$$\hat{\mathbf{x}}_7 = c_2(\hat{\mathbf{x}}_6, \theta_2); \quad \hat{\mathbf{x}}_7 \in \mathbb{R}^{H \times W \times 3} \quad (8)$$

$$\hat{\mathbf{x}}_{out} = \hat{\mathbf{x}}_7 + \hat{\mathbf{x}}_{in}; \quad \hat{\mathbf{x}}_{out} \in \mathbb{R}^{H \times W \times 3} \quad (9)$$

Each Wave module receives the input $\hat{\mathbf{x}}_{in} \in \mathbb{R}^{H \times W \times 3}$ and the mask m which are concatenated to create $\hat{\mathbf{x}}_0$ (1). $\hat{\mathbf{x}}_0$ is sent to a convolution layer c_1 that reduces its feature resolution by half and increases the channel dimension to C (2). This feature map $\hat{\mathbf{x}}_1$ is sent to a series of 4 WaveMix blocks for token-mixing (3). The output from the WaveMix block $\hat{\mathbf{x}}_2$ is further passed through a DepthConv module where the feature maps undergo further spatial token-mixing from the depth-wise convolution (4). A skip connection from c_1 is concatenated with the output from DepthConv module $\hat{\mathbf{x}}_3$ which increases the channel dimension of the output $\hat{\mathbf{x}}_4$ to $2C$ (5). This output is further passed through a Decoder network which increases the resolution of feature maps to original resolution (6). The Decoder layer also reduces the number of channels to $C/2$ and the feature maps $\hat{\mathbf{x}}_5$ are again concatenated with the input $\hat{\mathbf{x}}_{in}$ (7). The output after concatenation $\hat{\mathbf{x}}_6$ is then passed to a final convolution layer c_2 to generate the output $\hat{\mathbf{x}}_7$ (8). A residual connection is also provided from the input for ease of gradient flow (9) and resultant feature maps are the final output of the Wave module $\hat{\mathbf{x}}_{out}$.

Model	#Params \downarrow	CelebA-HQ (256 \times 256)					
		Narrow masks		Medium masks		Wide Masks	
		FID \downarrow	LPIPS \downarrow	FID \downarrow	LPIPS \downarrow	FID \downarrow	LPIPS \downarrow
CoModGAN [5]	109 M	16.8	0.079	19.4	0.092	24.4	0.102
AOT GAN [17]	15 M	6.67	0.081	7.28	0.089	10.3	0.118
RegionWise [18]	47 M	11.1	0.124	7.52	0.101	8.54	0.121
DeepFill v2 [19]	4 M	12.5	0.130	9.05	0.105	11.2	0.126
EdgeConnect [7]	22 M	9.61	0.099	7.56	0.095	9.02	0.120
LaMa-Fourier [13]	27 M	7.26	0.085	6.13	0.080	6.96	0.098
WavePaint	3 M	8.03	0.115	8.87	0.123	21.3	0.155
WavePaint	10 M	5.53	0.085	5.59	0.090	7.22	0.112

Table 1: Quantitative evaluation metrics of inpainting on CelebA-HQ dataset. Learned perceptual image patch similarity (LPIPS) and Fréchet inception distance (FID) are reported. The best WavePaint results are highlighted in bold. The results of models which report better results than WavePaint are coloured red. The metrics are reported for three different types of test mask generation, i.e. narrow, medium and wide masks as used in LaMa [13]. Other models have much larger parameters compared to WavePaint and also employ adversarial training. Still, WavePaint manages to outperform them without using any adversarial training. It uses learnable parameters more efficiently.

3.3 WaveMix Blocks

WaveMix block [4] is the fundamental building block of WaveMix architecture which allows multi-resolution token-mixing of information using 2D-DWT. This helps in a rapid expansion of receptive field. It also reduces computational burden because 2D-DWT decreases the input resolution by half and further processing by multi-layer-perceptron (MLP) is faster and cheaper. DWT helps in lowering the number of model parameters significantly, as it lacks any parameters, while promoting global context understanding even on a shallow network. We have used the WaveMix block with one level of 2D-DWT using Haar wavelet. Details of the operations inside WaveMix block are given in [4].

3.4 DepthConv

DepthConv employs a depth-wise convolution operation followed by a GELU non-linearity and batch-normalization as shown in Figure 3. We use a depth-convolution with kernel size of 5, which is smaller than kernel size used in Convmixer models. This was done further decrease the parameter count.

3.5 Decoder

Decoder module is used to up-sample the resolution of feature maps back to original input resolution to the Wave module. It comprises of a transposed convolution layer followed by a batch-normalization. The transposed convolution layer is also used to reduce the number of channels by 4, from $2C$ to $c/2$.

4 Experiments and Results

4.1 Datasets, Loss Function and Metrics

We use CelebA-HQ[20] and ImageNet [21] datasets (under MIT Licenses) for our experiments. We use images of size 256×256 for CelebA-HQ and 224×224 for ImageNet experiments. Validation for each dataset was performed on the entire validation sets of respective datasets.

We followed the same mask generation policy employed in LaMa [13] and used their settings to generate narrow, medium and wide masks. We took the same 26,000 train images and 2,000 test images from CelebA-HQ that LaMa used for CelebA-HQ experiments. Learned perceptual image patch similarity (LPIPS) [22] and Fréchet inception distance (FID) [23] are reported as metrics since L1 and L2 distances are not enough to compare inpainted images with large masks where multiple natural completions are possible. Inference throughput on a single GPU was reported in frames/sec (FPS).

We used a hybrid loss L_{hybrid} to optimize the model parameters. Since we did not employ a discriminator for adversarial training, no adversarial loss was used. We used a weighted sum of L_1 (mean absolute error), L_2 (mean square error) and L_{LPIPS} as shown below:

Model	#Param	FID↓	GPU	Throughput (FPS)	
				Inference	Train
LaMa [13]	27 M	7.26	23 GB	32	11
WavePaint	5 M	7.09	11 GB	105	32

Table 2: Comparison of LaMa [13] and WavePaint on parameters, resource-consumption and speed. Results are reported for experiments on CelebA-HQ dataset with narrow masks on single 24 GB RTX 3090 GPU for a batch size of 10. We see that WavePaint is three times faster than LaMa, but consumes only half the GPU resource and uses less than one fifth of LaMa’s parameters.

Model	#Modules	# Blocks	DepthConv	#Params	Narrow masks		Medium masks		Wide Masks		Throughput (FPS)	
					FID↓	LPIPS↓	FID↓	LPIPS↓	FID↓	LPIPS↓	Inference	Train
CelebA-HQ (256x256)												
WavePaint	2	4	No	3.3 M	11.1	0.148	13.9	0.148	33.7	0.176	356	165
WavePaint	2	4	Yes	3.3 M	8.03	0.115	8.87	0.123	21.3	0.155	322	145
WavePaint	3	4	Yes	5.0 M	7.09	0.103	6.96	0.104	10.2	0.131	275	99
WavePaint	5	4	Yes	8.4 M	6.56	0.096	6.62	0.098	8.83	0.122	167	60
WavePaint	6	4	Yes	10 M	5.53	0.085	5.59	0.090	7.22	0.112	133	50
ImageNet (224 × 224)												
WavePaint	2	4	Yes	3.3 M	3.26	0.134	3.72	0.108	-	-	333	213
WavePaint	3	4	Yes	5.0 M	3.21	0.138	3.47	0.106	-	-	305	126

Table 3: Quantitative evaluation metrics of inpainting by WavePaint of different sizes by varying the number of modules and WaveMix blocks per modules. All models use level-1 2D DWT. The models were evaluated on 2000 images of CelebA-HQ that LaMa used for testing. For evaluation on ImageNet, we used the entire 50,000 images from ImageNet validation set. Inference and training throughput in frames per second (FPS) is reported on a single 80 GB A100 GPU.

$$L_{hybrid} = (1 - \alpha)L_1 + \alpha L_2 + L_{LPIPS} \quad (10)$$

4.2 Implementation details

Due to limited computational resources, the *maximum* number of training epochs was set to 300 for CelebA-HQ and 50 for ImageNet Experiments. All experiments were run on a single 80 GB Nvidia A100 GPU. We used AdamW optimizer ($\alpha = 0.001$, $\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 10^{-8}$) with a weight decay of 0.01 during initial epochs and then used SGD (stochastic gradient descent) with learning rate of 0.001 and momentum = 0.9 during the final 50 epochs [24, 25]. We used the maximum batch-size that could be accommodated in a single GPU for our experiments. We used an embedding dimension (C) of 128 in all the Wave modules. Each Wave module has 4 WaveMix blocks unless otherwise specified.

Model	Params	Narrow masks		Medium masks		Wide Masks		Throughput (FPS)	
		FID↓	LPIPS↓	FID↓	LPIPS↓	FID↓	LPIPS↓	Inference	Train
Level 1	5.0 M	7.09	0.103	6.96	0.104	10.2	0.131	275	99
Level 2	7.6 M	7.12	0.095	7.16	0.096	9.16	0.119	222	78
Level 3	10 M	7.74	0.094	7.62	0.092	9.26	0.112	200	67

Table 4: The variation of performance of WavePaint with different levels of 2D -DWT used in WaveMix blocks. WavePaint with 3 modules and 4 WaveMix blocks per module is used and results on CelebA-HQ are shown. The improved performance is due to the rapid expansion of receptive fields while using multi-level 2D-DWT token-mixing.

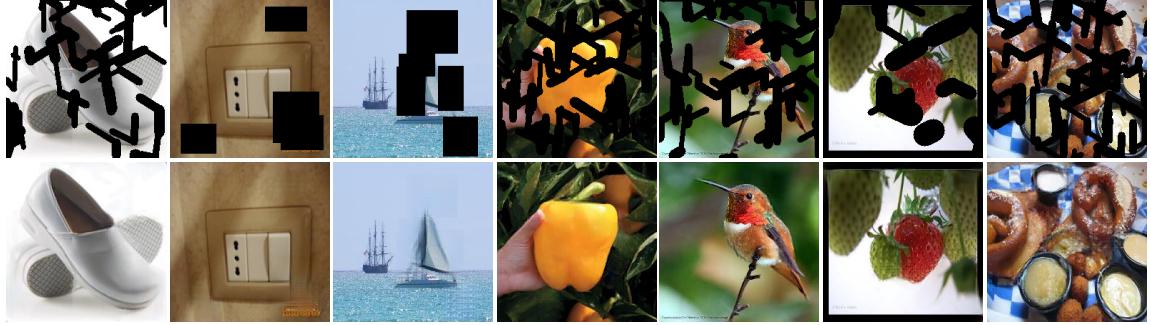


Figure 4: Inpainted images (bottom row) generated by WavePaint from masked images (top row) from ImageNet validation set



Figure 5: Inpainted images (bottom row) generated by WavePaint from masked images using wide masks (top row) from ImageNet validation set

4.3 Results and Discussion

4.4 Quantitative Results

We compare our models with the other state-of-the-art baselines as shown in Table 1 for CelebA-HQ dataset. We compare the performance of the WavePaint across narrow, medium and wide masks. WavePaint consistently outperforms most of the other models, on a variety of mask configurations. It has to be noted most of the other models have much larger parameter count and employ adversarial training using a discriminator. Since WavePaint does not employ a discriminator it is light-weight, it can be trained faster than GANs and diffusion models.

We could not compare WavePaint with latest diffusion models such as RePaint [16] because diffusion is a much slower process of image generation and we were constrained in computational resources. RePaint [16] had reported that quantitative results of LaMa [13] are better than that of RePaint in wide and narrow mask inpainting on ImageNet and CelebA-HQ datasets.

Since LaMa was a resource-efficient model for inpainting, we compared WavePaint with LaMa [13] in Table 2 to analyse its resource-efficiency. We see that WavePaint requires less than one-fifth of the parameters of LaMa to outperform it in FID metric. WavePaint is also $\sim 3\times$ faster than LaMa in both inference and training speed and utilizes less than half the GPU consumed by LaMa. Our results clearly shows that WavePaint is more resource and parameter-efficient than LaMa. The high resource-efficiency of WavePaint can be attributed to the resource-efficient token-mixing using WaveMix blocks which processes the image at a lower resolution due to lossless downsampling property of 2D-DWT. The quantitative performance of WavePaint using different hyperparameters on CelebA-HQ and ImageNet datasets are shown in Table 3.

Table 4 shows the performance of WavePaint which uses WaveMix blocks with multi-level 2D-DWT. Using higher levels of DWT can improve the performance of the model due to the exponential increase in receptive field.



Figure 6: Inpainted images (bottom row) generated by WavePaint from masked images using medium masks (top row) from ImageNet validation set



Figure 7: Inpainted images (bottom row) generated by WavePaint from masked images using narrow masks (top row) from ImageNet validation set

4.5 Qualitative Results

The images generated by WavePaint on ImageNet dataset are shown in Figure 4. We can see that WavePaint completes textures and missing details by completing the lines and filling in details. The images generated by WavePaint for wide, medium and narrow masks are shown in Figure 5, Figure 6 and Figure 7 respectively. WavePaint can fill in missing details of facial features, colour, texture, eyes and eyebrows even if major parts of the image are masks.

5 Ablation Studies

Multiple ablation experiments were conducted to optimize the network hyper-parameters and understand the utility of the network components. Table 5 shows the performance of WavePaint with 8 WaveMix blocks arranged in different number of modules. Results shows that having less number of modules with large number of WaveMix blocks is more parameter-efficient but results in poor performance. When we decrease the number of WaveMix blocks in each module and increase the number of modules, the model become larger with higher parameter count. Modules with 4 Waveblocks each retain parameter-efficiency without degrading performance.

Removing DepthConv block from WavePaint reduces the FID score by 38% and increases the training and inference throughput by 14%. Since, depth-wise convolution is a highly parameter efficient operation, its removal only reduces the number of parameters by less than 1%. Therefore, adding DepthConv block in each module is beneficial for the network as it aids the WaveMix block with further spatial token-mixing.

#Modules	#WaveMix Blocks	#Params	LPIPS
1	8	3.0 M	0.085
2	4	3.3 M	0.079
4	2	4.0 M	0.079

Table 5: Performance of WavePaint with 8 WaveMix blocks by varying the number of modules. Experiment was done on a subset of ImageNet dataset.

6 Conclusion and Future Work

This paper proposes using multi-level 2D-DWT token-mixing for the less explored task of image inpainting. The performance of the proposed model is comparable to much larger models and those that uses adversarial training on CelebA-HQ dataset. Also, our model uses only a fraction of the parameters, consumes less GPU RAM and is multiple times faster in training and inference compared to other models such as LaMa [13]. A possible direction of future work is to develop resource-efficient image generation models using WavePaint trained in an adversarial or diffusion setting. Thus, this paper points to the potential of using token-mixing as alternative to vision transformers and CNNs for resource-efficient image inpainting without the need for slower complex training procedures like adversarial and diffusion. The faster receptive field expansion leading to availability of global context information can help these models do image reconstruction on par with transformers.

References

- [1] Wenjie Luo, Yujia Li, Raquel Urtasun, and Richard Zemel. Understanding the effective receptive field in deep convolutional neural networks, 2017.
- [2] Weihao Yu, Mi Luo, Pan Zhou, Chenyang Si, Yichen Zhou, Xinchao Wang, Jiashi Feng, and Shuicheng Yan. Metaformer is actually what you need for vision, 2022.
- [3] Asher Trockman and J Zico Kolter. Patches are all you need?, 2022.
- [4] Pranav Jeevan, Kavitha Viswanathan, Anandu A S, and Amit Sethi. Wavemix: A resource-efficient neural network for image analysis, 2023.
- [5] Shengyu Zhao, Jonathan Cui, Yilun Sheng, Yue Dong, Xiao Liang, Eric I Chang, and Yan Xu. Large scale image completion via co-modulated generative adversarial networks, 2021.
- [6] Hongyu Liu, Bin Jiang, Yibing Song, Wei Huang, and Chao Yang. Rethinking image inpainting via a mutual encoder-decoder with feature equalizations, 2020.
- [7] Kamyar Nazeri, Eric Ng, Tony Joseph, Faisal Z. Qureshi, and Mehran Ebrahimi. Edgeconnect: Generative image inpainting with adversarial edge learning, 2019.
- [8] Yuhang Song, Chao Yang, Yeqi Shen, Peng Wang, Qin Huang, and C. C. Jay Kuo. Spg-net: Segmentation prediction and guidance network for image inpainting, 2018.
- [9] Manyu Zhu, Dongliang He, Xin Li, Chao Li, Fu Li, Xiao Liu, Errui Ding, and Zhaoxiang Zhang. Image inpainting by end-to-end cascaded refinement with mask awareness. *IEEE Transactions on Image Processing*, 30:4855–4866, 2021.
- [10] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks, 2018.
- [11] Ziyu Wan, Jingbo Zhang, Dongdong Chen, and Jing Liao. High-fidelity pluralistic image completion with transformers. *arXiv preprint arXiv:2103.14031*, 2021.
- [12] Wenbo Li, Zhe Lin, Kun Zhou, Lu Qi, Yi Wang, and Jiaya Jia. Mat: Mask-aware transformer for large hole image inpainting, 2022.
- [13] Roman Suvorov, Elizaveta Logacheva, Anton Mashikhin, Anastasia Remizova, Arsenii Ashukha, Aleksei Silvestrov, Naejin Kong, Harshith Goka, Kiwoong Park, and Victor Lempitsky. Resolution-robust large mask inpainting with fourier convolutions, 2021.
- [14] Yingchen Yu, Fangneng Zhan, Shijian Lu, Jianxiong Pan, Feiying Ma, Xuansong Xie, and Chunyan Miao. Wavefill: A wavelet-based generation network for image inpainting, 2021.

- [15] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models, 2021.
- [16] Andreas Lugmayr, Martin Danelljan, Andres Romero, Fisher Yu, Radu Timofte, and Luc Van Gool. Repaint: Inpainting using denoising diffusion probabilistic models, 2022.
- [17] Yanhong Zeng, Jianlong Fu, Hongyang Chao, and Baining Guo. Aggregated contextual transformations for high-resolution image inpainting, 2021.
- [18] Yuqing Ma, Xianglong Liu, Shihao Bai, Lei Wang, Aishan Liu, Dacheng Tao, and Edwin Hancock. Region-wise generative adversarial imageinpainting for large missing areas, 2019.
- [19] Jiahui Yu, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and Thomas Huang. Free-form image inpainting with gated convolution, 2019.
- [20] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation, 2018.
- [21] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009.
- [22] Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric, 2018.
- [23] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium, 2018.
- [24] Nitish Shirish Keskar and Richard Socher. Improving generalization performance by switching from adam to SGD. *CoRR*, abs/1712.07628, 2017.
- [25] Pranav Jeevan and Amit sethi. Convolutional xformers for vision, 2022.