Adapting Masked Autoencoders for Artistic Image Generation

Kendrick Nguyen
UCSD Electrical & Computer Engineering Department
A16045878

Paul Nguyen
UCSD Electrical & Computer Engineering Department
A16660828

Abstract

Masked Autoencoders (MAEs) are a self-supervised learning framework introduced by He et al. (2021) [He+21], known for their ability to learn meaningful representations by reconstructing masked portions of input images. Although MAEs have shown strong performance on large multi-class image sets, their effectiveness in more specialized visual domains remains underexplored. This project aims to replicate the MAE architecture and train it on a visual art dataset to investigate whether MAEs can generalize to niche domains, such as artistic imagery and representation. Primary metrics in assessing the scalability of MAEs in the art domain will be reconstruction metrics (e.g. qualitative visualization, ablation studies on the effect of changing hyper-parameters, MSE (mean squared error), and Learned Perceptual Image Patch Similarity (LPIPS).

1 Introduction

MAEs have sparked an advancement in the autoencoder architecture, prominently known for learning meaningful representations in a "forced-reasoning" and self-supervised way. This is notably demonstrated in reconstruction tasks in predicting the missing parts of an input based only on visible parts. In a visual learning task, the problem starts with randomly hiding patches of an input image; the encoder processes only the visible patches, while the lightweight decoder attempts to reconstruct the original image from both visible and masked tokens. Due to these intentional randomized masks, the model must rely on learning the spatial context, structure, and semantics of an input to successfully reconstruct the original input.

In the work of He et al. (2021) [He+21], MAEs have demonstrated significant generalization on large-scale natural image datasets like ImageNet. Traditionally, these datasets often serve as baselines for many visual learning research, and much of their images are often limited to naturalistic images, such as objects, animals, and scenes, that follow relatively predictable patterns. In contrast, images like visual art are inherently more abstract and subjective, often blending symbolic elements, non-linear compositions, and surreal forms, arguably creating unpredictable patterns. This raises the question: Can MAEs scale to capture the semantics and structure of visual art data where compositions are more human-expressive and less predictable?

This project is motivated by the hypothesis that the compositional complexity of art presents a different challenge than object-centric datasets. Testing MAEs in this domain could reveal whether their learning mechanism is robust for human-created content and broader real-world applications.

2 Related Work

Masked Autoencoders Are Scalable Vision Learners [He+21]
He et al. introduced Masked Autoencoders (MAEs), a self-supervised learning framework that reconstructs missing image patches to pre-train Vision Transformers (ViTs). Specifically, MAEs randomly mask a significant portion—typically (up to 75%) of non-overlapping image patches and task the model with reconstructing the missing content based only on the visible patches. Due to this constraint, the model is forced to learn the spatial structures and semantic context within the image to successfully reconstruct its original image. MAEs follows an asymmetric encoder-decoder design where the encoder processes only the visible (unmasked) patches and the decoder takes the encoded visible representations and attempts to reconstruct the full image. This work has notably shown that MAEs are scalable on high-capacity ImageNet models and transferable for downstream vision tasks, such as image classification, object detection, and segmentation.

SimMIM: A Simple Framework for Masked Image Modeling [Xie+22]
SimMIM, proposed by Xie et al., presents another approach to masked image modeling. SimMIM is prominently known for its lightweight architecture in employing a single transformer encoder for both masked and unmasked patches. Due to this simplification, the model greatly reduces architectural complexity and computational overhead compared to other masked image models. In SimMIM, random masking is similarly applied across image patches and tokens; however, the model is trained to reconstruct the raw pixel values of the masked patches directly (without the need for patch embeddings or latent features). Despite its lightweight design, SimMIM has been proven to efficiently infer missing image content and yield high benchmarks on ImageNet classification tasks.

BEiT: BERT Pre-Training of Image Transformers [BDW21]
Bidirectional Encoder representation from Image Transformers (BEiT) adapts prediction token tasks from natural language processing (NLP) to the visual domain. BEiTs were motivated to pre-train ViTs and also employs random images patches and visual tokens to recover the original visual tokens based on the corrupted image patches. This approach exceeds methods in predicting masked patches using raw pixels and instead utilizes semantic modeling, learning high-level representations through token distributions.

3 Method

In this work, we re-implement a MAE architecture on a curated visual art dataset to evaluate its effectiveness on non-natural and human-expressive imagery. Our implementation is based on the original MAE framework, but we introduce adaptations for training and testing our visual art dataset. Although MAEs were originally intended to transfer performance in downstream tasks, this project will be entirely experimenting with the "pre-training".

MAEs follow a ViT encoder-decoder architecture. During training (or, prior to downstream taks, during pre-training), each input $x \in \mathbb{R}^{3\times H\times W}$ is divided into non-overlapping patches of size $p \times p$. Each patch is linearly embedded into a token and later passed through a positional embedding layer. A masking ratio is applied (often 75%), random selecting a subset of visible tokens x_v o be input to the encoder E.

$$z = E(x_v) \tag{1}$$

The decoder D reconstructs the original patch pixels from both the encoded visible tokens z and learned mask tokens x_m.

$$\hat{x} = D(z \cup x_m) \tag{2}$$

We use the Mean Squared Error (MSE) loss computed only on the masked patches,

$$\mathcal{L}_{\text{MSE}} = \frac{1}{|x_m|} \sum_{i \in x_m} \|\hat{x}_i - x_i\|^2 \tag{3}$$

Where:

- x: the original image.
- x_v: the set of visible (unmasked) image patches.
- x_m: the set of masked patch tokens (usually learned embeddings).
- $\hat{x}$: the reconstructed image.
- z: the latent representation output by the encoder for the visible patches.
- $E(\cdot)$: the encoder network.
- $D(\cdot)$: the decoder network.

MAEs follow an asymmetric encoder-decoder design where the encoder processes only a small subset of visible (unmasked) image patches, and the decoder reconstructs the original image from the encoded representations and mask tokens. However, both the encoder and decoder are similarly built using transformer blocks.

1. **Encoder**

   The encoder only takes in unmasked patches.

   Key Components:

   - Patch Embedding Layer:
     - Divides the input image into non-overlapping patches
     - Each patch is linearly projected into an embedding, contains positional embeddings
   - Masking:
     - Apply masking based on masking ratio (e.g. 75%)
   - Transformer Blocks:
     - Sequence of ViT Transformer blocks
     - Processes only the visible (unmasked) patches

   Output of Encoder:

   - A set of latent feature vectors representing the unmasked patches.

2. **Decoder**

   The decoder reconstructs the full image from the encoder's output and mask tokens.

   Key Components:

   - Mask Tokens
     - Learnable vectors representing the masked patches
   - Positional Embeddings
     - Reapply linear projection to learn the same positional embedding
   - Transformer Blocks
     - Sequence of ViT Transformer blocks
     - Processes all tokens (unmasked + mask tokens) to reconstruct missing patches
   - Prediction Head
     - A linear layer that projects each decoded token back to pixel space

   Output of Decoder:

   - Reconstructed image patches (both originally masked and unmasked).

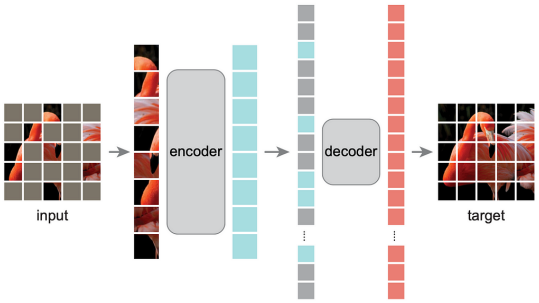


Figure 1: MAE Architecture [He+21]

Due to hardware and time constraints, our encoder and decoder architecture was reduced to smaller transformer layers and hidden dimensions (12-layer transformer, 192 hidden dimensions for encoder and 4-layer transformer, 192 hidden dimensions for encoder).

The MAE model will ideally be trained on a visual art dataset, containing numerous art works across various styles and genres. The dataset is divided into 80/10/10 into train/validation/testing sets respectively and are preprocessed similarly to He et al's [He+21] implementation. Specifically, images are transformed with a center crop, normalized with pretrained ImageNet mean and variance weights, augmented with random color jittering, and resized to 224×224. Training is performed using the AdamW optimizer supported by PyTorch with a weight decay of 0.05, batch size of 32, cosine learning rate scheduling, and gradient scaling. Models are trained for 10 epochs.

We will employ various strategies for evaluating reconstruction quality, including qualitative visualization and MSE. In the original work, MAEs were also evaluated on downstream recognition tasks; to replicate this, we will also employ LPIPS to measure perceptual similarity between the original and MAE reconstructed images. LPIPS are known for comparing deep features extracted (layer by layer) from pretrained neural networks, aligning more with human visual perception than pixel-to-pixel comparison.

$$\text{MSE} = \frac{1}{HW} \sum_{i=1}^{H} \sum_{j=1}^{W} (\hat{x}_{ij} - x_{ij})^2 \tag{4}$$

$$\text{LPIPS}(x, \hat{x}) = \sum_{l} \frac{1}{H_l W_l} \sum_{h,w} \| w_l \cdot (f_l(x)_{hw} - f_l(\hat{x})_{hw}) \|_2^2 \tag{5}$$

Where:

- x, $\hat{x}$: original and reconstructed images, respectively.
- H, W: height and width of the image in pixels.
- $f_l(x)$: activation (feature map) from layer l of a pretrained network.
- w_l: learned per-channel weights for features at layer l.
- H_l, W_l: spatial dimensions (height and width) of the feature map $f_l(x)$.
- $f_l(x)_{hw}$: feature vector at spatial location (h, w) in layer l.

Likewise to the original work, to measure the scalability of MAEs in the domain of visual art, an ablation study on testing the effect of changing masking ratios and patch sizes will also be employed.

4 Experiments

Our dataset was sourced from the Kaggle dataset Wiki-Art Visual Art Encyclopedia [Inn22]. The dataset contains 96,014 different images and 14 different class names. Each image has a single class

name, such as animal painting, cityscape, landscape, or portrait. The MAE model compares the original image with the reconstructed image so the labels are not used in the model, but knowing the diversity of image classes is still good for understanding the type of challenge the model will need to accomplish. Since the dataset is so large, some models were trained using only a subset of the dataset to determine the quality of the model architecture more quickly. The dataset is further broken down into 70% being used for training, 15% being used for validation, and the remaining 15% being used for testing. Images are transformed with a center crop, normalized with pretrained ImageNet mean and variance weights, augmented with random color jittering, and resized to 224×224. Three MAE models were implemented for 75% masking but with slightly different architecture modifications.

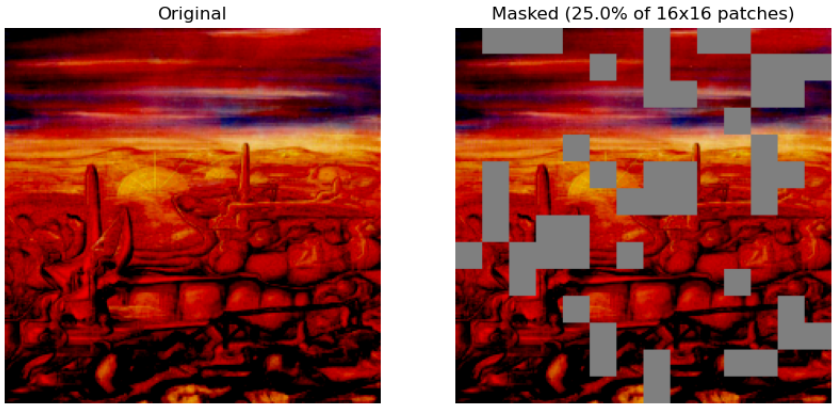


Figure 2: Image Masking Example

Model 1: Baseline Implementation
Our first model follows the architecture proposed by He et al. (2021) [He+21] but adopts reduced hyperparameters to prioritize computational efficiency over reconstruction quality. The encoder uses 384 embedding dimensions (depth=6), while the decoder employs 256 dimensions (depth=2). Despite its smaller capacity, this configuration achieved a test MSE of 0.1924 on masked regions. Reconstruction examples are shown in Figure 5a.

Model 2: Minimalist Design
Building on an alternative MAE implementation [Ica21] (originally designed for CIFAR-10), we adapted the model for our visual arts dataset. This minimalistic architecture outperformed prior variants, achieving an average MSE of 0.0488 and yielding the best patch-level reconstruction quality (Figure 5b).

Model 3: Hyperparameter Exploration
To investigate the impact of architectural scale, we modified Model 1 by increasing decoder complexity (512 dimensions, depth=8) while reducing encoder dimensions (192 dimensions, depth=12). Contrary to expectations, this led to degraded performance (MSE = 0.397), suggesting that overly complex decoders may hinder performance—consistent with the original paper's emphasis on decoder lightweightness. Results are visualized in Figure 5c.

Further comparisons among the models were also examined. The loss curves for all three models (Figure 3) reveal critical insights into their optimization behavior and convergence properties. Accordingly, the model were trained to roughly 10 epochs due to time constraints but showed stable convergence. We hypothesize our results for model 1 and 2 would further improve if given more epochs; however, this would not apply for model 3 as it was observed that over-parameterizing the decoder harms generalization.

While MSE quantifies low-level pixel accuracy, we employed LPIPS with a pretrained AlexNet to evaluate whether reconstructions preserve semantically meaningful features for downstream tasks consistent to He et al.'s motivation of MAEs as a pretraining scaffold. From the results (Figure 4), model 2 yielded the best LPIPS score of 0.2975. This indicates its reconstructions are at near perceptually indistinguishable from originals to a pretrained network. Although these LPIP scores are

not near the range of He et al.'s (0.09–0.15), it does suggests Model 3's MAE is effectively learning representations and scalable for the visual art domain.

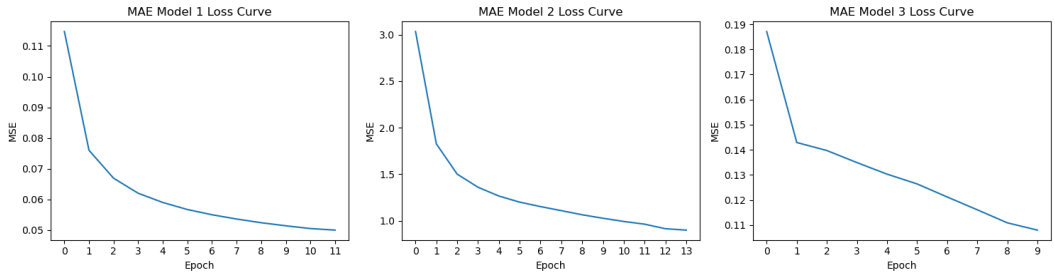


Figure 3: Loss Curves

MAE LPIPS w/ AlexNet	
Model 1	0.4118
Model 2	0.2975
Model 3	0.6058

Figure 4: LPIPS Evaluation Results Using AlexNet

5 Supplementary Material

- The code is found at `https://github.com/kendrick010/ece285_project`.
- The video presentation is found at `https://youtu.be/xO4jU_GmQsQ`
- The dataset can be found at `https://www.kaggle.com/datasets/ipythonx/wikiart-gangogh-creating-art-gan`

References

[BDW21] Hangbo Bao, Li Dong, and Furu Wei. *BEiT: BERT Pre-Training of Image Transformers*. 2021. arXiv: 2106.08254 [cs.CV].

[He+21] Kaiming He et al. *Masked Autoencoders Are Scalable Vision Learners*. 2021. arXiv: 2111.06377 [cs.CV].

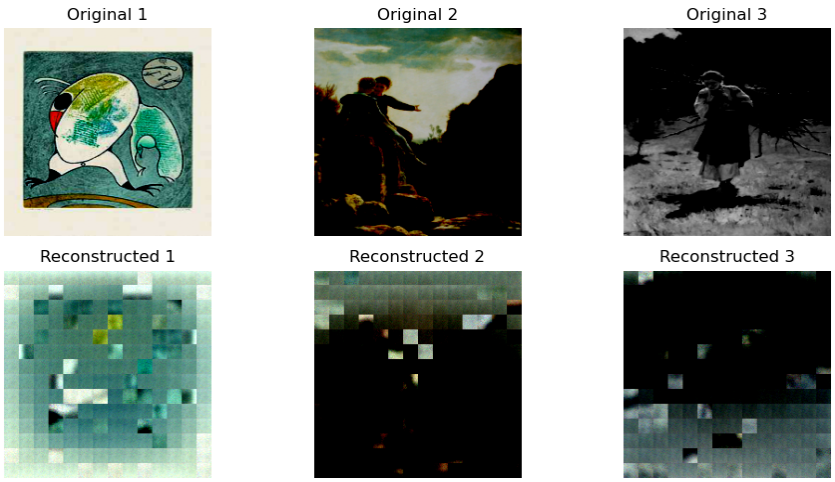
[Ica21] IcarusWizard. *MAE (Masked Autoencoder) Implementation*. https://github.com/IcarusWizard/MAE. 2021.

[Inn22] Innat. *Wikiart-GanGogh-Creating-Art-GAN*. Aug. 2022. URL: https://www.kaggle.com/datasets/ipythonx/wikiart-gangogh-creating-art-gan.

[Xie+22] Zhenda Xie et al. *SimMIM: A Simple Framework for Masked Image Modeling*. 2022. arXiv: 2111.09886 [cs.CV].

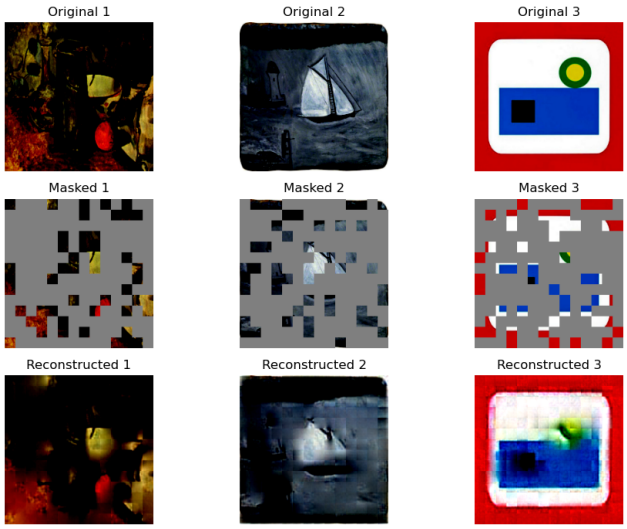
(a) Model 1

(b) Model 2

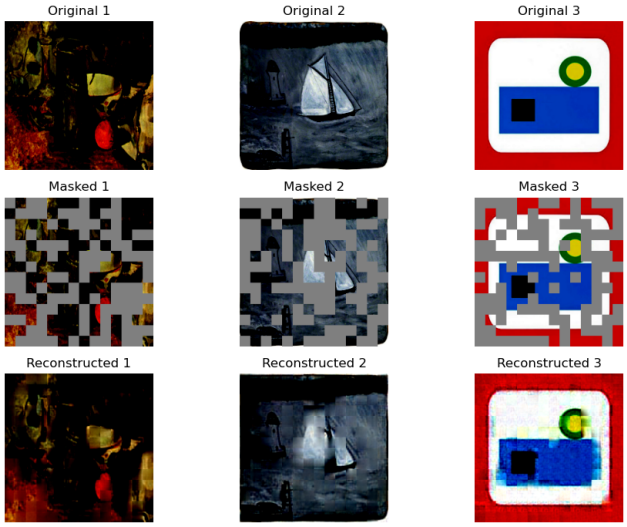


(c) Model 3

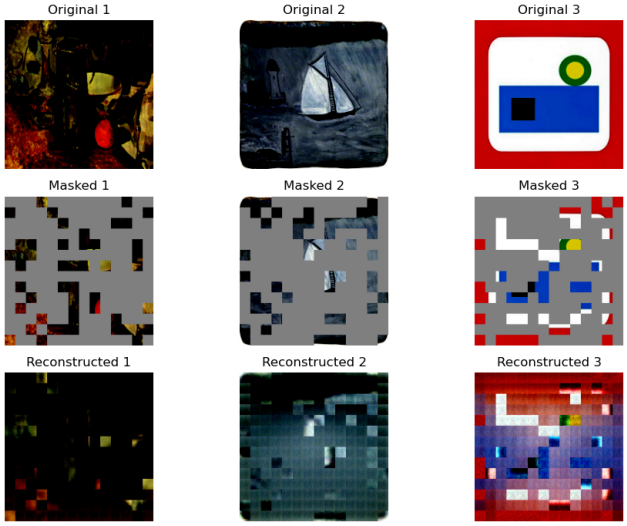
Figure 5: Reconstruction results for different MAE model variants (stacked vertically).



(a) Model 1



(b) Model 2



(c) Model 3

Figure 6: Reconstruction results for different MAE model variants (stacked vertically).