# Assignment 1 - Masked Autoencoders

Chien-Jui Huang
R14922065

## 1. Summary

The objective of this assignment is to verify the claims and analysis of [3] through a series of experiments. The paper demonstrates that Visual Transformer (ViT) based autoencoders are scalable self-supervised learners for computer vision. Inspired by masked auto-encoding in BERT[2], the proposed pre-training framework randomly masks out patches of the input images and feeds only a small subset of visible patches to the encoder. The decoder then reconstructs the missing pixels from the encoded patches and mask tokens. After pre-training, the decoder is discarded and the encoder can be fine-tuned for downstream recognition tasks. The method not only improves accuracy but also accelerates training, since a large proportion (75%) of the image patches are masked during pre-training, and the mask tokens are processed only by the lightweight decoder without being propagated through the encoder.

## 2. Experiment Settings

The experiment codebase is adapted from this GitHub repository. Instead of conducting self-supervised pre-training on ImageNet-1K with ViT-Large, the codebase implements the same pre-training framework on CIFAR-10 using a much smaller Visual Transformer with only several layers of Transformer blocks. This simplified setup enables the claims of the paper to be reproduced with lower resources.

The environment settings are listed below, with screenshots of detailed information shown in Figure 1

- Python version : 3.9.23
- Pytorch version : 2.7.1+cu118
- GPU : NVIDIA RTX A6000

## 3. MAE Pre-training

The proposed masked autoencoder (MAE) reconstructs a randomly masked image back to its original form using Visual Transformer (ViT) based encoder-decoder architecture. During training, the loss function computes the mean squared error (MSE) between the reconstructed and original images in the pixel space.

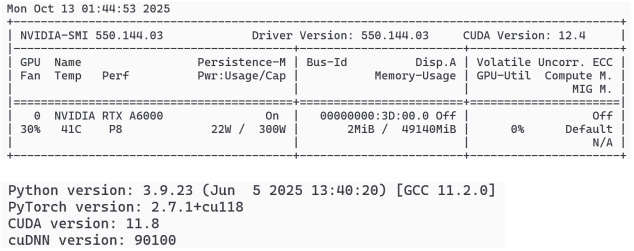



Figure 1. **Environment Settings**. The experiments were conducted using an NVIDIA RTX A6000 GPU with Python 3.9.23 and PyTorch 2.7.1+cu118.

**Masking.** Following Visual Transformer implementations, an image is first divided into regular non-overlapping patches. A randomly sampled subset of these patches is kept visible while the rest are masked (i.e. removed). If the masking ratio is low, the task becomes trivial, as masked patches can be easily inferred from nearby visible patches. Conversely, a high masking ratio encourages the encoder to learn meaningful image representations to reconstruct the original image. Random uniform sampling of masked patches also prevents them from clustering at the center, which could make the task excessively difficult. In the experiments, the default masking ratio is set to 0.75.

**MAE encoder.** The encoder is a Visual Transformer (ViT) applied only on visible (unmasked) patches. Each patch is combined with positional embeddings and processed through Transformer blocks. Masked patches are removed entirely without using mask tokens, allowing the encoder to focus on learning representations from visible patches while reducing computation time and memory at the same time. In the experiments, the encoder consists of 12 layers of Transformer blocks, with an embedding dimension of 192 and 3 attention heads.

**MAE decoder.** The decoder takes both encoded patches and mask tokens as inputs. Mask tokens are trainable vectors that indicate the locations of missing patches. The encoded patches and masked tokens are first arranged to reflect the original patch order then combined with positional

embeddings. Unlike the encoder, the decoder won't be used for downstream tasks and therefore can be asymmetrically designed. Based on the original paper, a lightweight decoder is sufficient for the reconstruction task. In the experiments, the encoder consists of 4 layers of Transformer blocks, with an embedding dimension of 192 and 3 attention heads.

The reconstructed image visualizations are shown in Figure 2, and the MAE pre-training loss curve is presented in Figure 3. Both the converging training loss and the qualitative results demonstrate that the pre-training task is learnable.

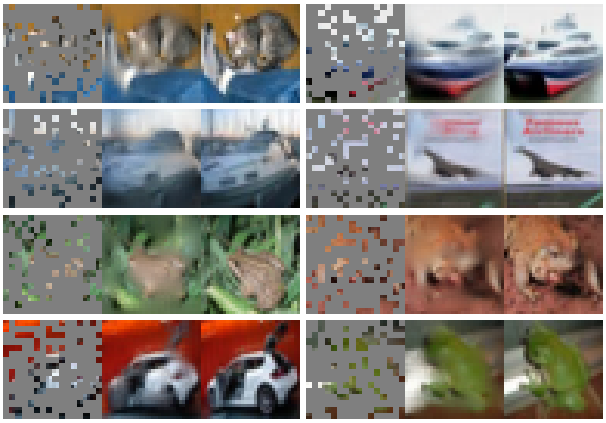


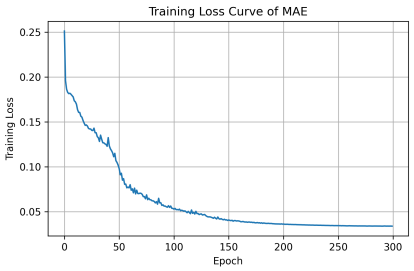
Figure 2. **MAE Reconstruction Visualization**. The first column shows random samplings that mask 75% of the image patches. The second column displays the reconstructed images, and the third column presents the original images.

## 4. Fine-tuning Results

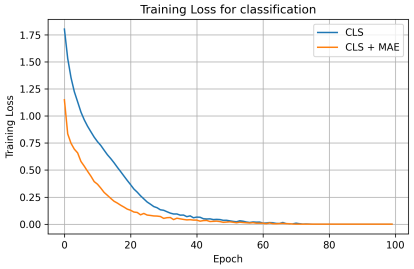
After MAE pre-training, The encoder in MAE is extracted and finetuned on the CIFAR-10 classification task. For comparison, a baseline model with identical structure but without pre-trained weights is trained from scratch. Both models are trained with 100 epochs, and their training loss curve and validation accuracy curve are shown in Figure 3. The results indicates that MAE pre-training improves model performance by enabling the encoder to learn useful image representations in advance. The pretrained model consistently achieves lower training loss and higher validation accuracy compared to the baseline model throughout the training process. Visualization of classifier predictions with MAE pre-training is shown in Figure 4

## 5. Ablation Studies

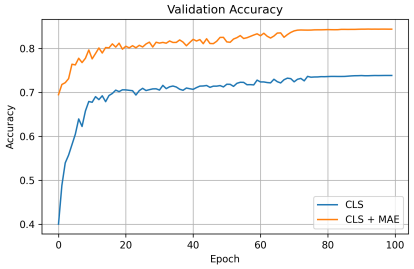
**Mask token.** As described in Section3, the encoder processes only visible image patches, while the masked tokens are introduced before being fed to the decoder afterwards. Table 1 presents an ablation study in which the mask tokens are instead provided to the encoder.




(a) MAE Pre-training



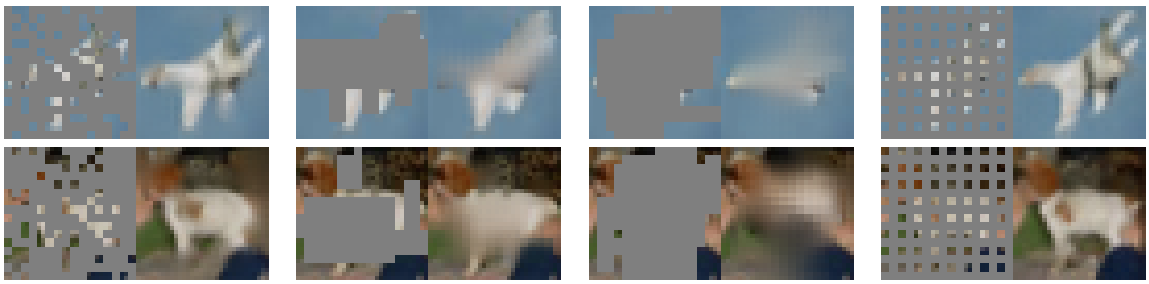

(b) Classification Training




(c) Classification Validation

Figure 3. **Training and Validation curves** of the experiments. (a): Training loss curve of the MAE pre-training process. (b): Comparison of classification training loss curve between models initialized from the MAE-pretrained weights and those trained from scratch. (c): Validation accuracy curve for classification, also comparing MAE-pretrained models and models trained from scratch.


Figure 4. **Visualization** of Classifier Predictions with MAE Pre-training. MAE pretrained models outperforms models trained from scratch with a validation accuracy of 0.844.



(a) random 75% (b) block 50% (c) block 75% (d) grid 75%

Figure 5. **Text Sampling Strategies** determine the pre-train task difficulty, influencing reconstruction quality and representations. Here each output is from an MAE trained with the specified masking strategy and masking ratio. (a): random sampling with a masking ratio of 0.75 (default). (b), (c): block-wise sampling that removes large random blocks with masking ratios of 0.5 and 0.75, respectively. (d): grid-wise sampling that keeps one of every four patches.

**(a) Mask token**

| case | ft | FLOPs |
|---|---|---|
| encoder w/ [M] | 80.6 | 2.4× |
| encoder w/o [M] | **84.4** | 1× |

**(b) Mask sampling**

| case | ratio | ft |
|---|---|---|
| random | 75 | 84.4 |
| block | 50 | **85.1** |
| block | 75 | 83.7 |
| grid | 75 | 82.0 |

Table 1. **MAE ablation experiments** with ViT-T/2 on CIFAR-10. fine-tuning (ft) accuracy (%) and relative multiply–accumulate operations (FLOPs) are reported. Default settings are marked in gray.

When mask tokens are fed to the encoder, the performance degrades. This may result from the inconsistency between pre-training and downstream fine-tuning tasks. During pre-training, the encoder needs to handle both visible patches and mask tokens, whereas in downstream tasks, the encoder only processes patches from a normal, unmasked image. Furthermore, with 75% of image patches masked, the pre-training signal is heavily influenced by the additional mask tokens, distracting the encoder from learning meaningful representations from the visible patches. Removing the mask tokens from the encoder ensures that the encoder only sees normal image patches thus improves accuracy.

In addition, skipping the mask tokens in the encoder provides the added benefit of reducing computational cost. Since the mask tokens are not propagated through the entire encoder–decoder structure, memory consumption is decreased, and as shown in Table 1, the overall FLOPs are reduced by 2.4. This computational efficiency makes MAE pre-training more scalable to larger models and datasets, and it can further accelerate training with larger batch sizes.

**Mask sampling strategy.** Table 1 also compares different mask sampling strategies, which are visualized in Figure 5.

Block-wise sampling, as proposed in [1], randomly mask the image patches within a bounding box and typically removes large regions near image centers. This results in a more challenging reconstruction task, as indicated by the higher training loss and the blurrier reconstructed images shown in Figure 5. When masking ratio is set to 75%, the performance degrades; however, when masking ratio is at 50%, block-wise sampling outperforms the default random sampling settings. This observation is different from the original paper, which reported that random sampling works best for MAE on ImageNet-1K. These results suggest that block-wise sampling, despite its difficulty, can lead the encoder to learn better representations when masking ratio is appropriately controlled and the dataset is relatively simple.

Additionally, grid-wise sampling is also examined. This strategy retains one of every four patches in a regular grid pattern. As shown in Figure 5, grid-wise sampling yields the sharpest reconstructed image, meaning that it is an easier reconstruction task compared to others. However, the learned representations are less effective, as reflected by the lowest validation accuracy among all sampling strategies.

In summary, although the paper reported that random sampling works best for MAE pre-training, the optimal masking strategy depends on both the masking ratio and the dataset, which together strongly influence the difficulty of a reconstruction task.

## 6. Conclusion

Through the experiments, it can be verified that Visual Transformers (ViT) based autoencoders are capable of capturing meaningful image representations through self-supervised learning. MAE pre-training helps improves accuracy by enabling the model to reconstruct images effectively. From the ablation studies, applying mask tokens only in the de-

coder allows the encoder to focus on visible patches, which improves performance and reduces computational consumptions. Moreover, the optimal mask sampling strategy depends on both masking ratio and dataset difficulty.

## References


[1] Hangbo Bao, Li Dong, Songhao Piao, and Furu Wei. Beit: Bert pre-training of image transformers, 2022. 3

[2] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2019. 1

[3] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners, 2021. 1