DeepRob Final Project Report: Improving Masked Autoencoders by Testing the Viability of Different Features and Adding GAN

1st Jirong Yang
Computer Science, LSA
University of Michigan
Ann Arbor, United States
yjrcs@umich.edu

2nd Fangyi Dai
Computer Science, LSA
University of Michigan
Ann Arbor, United States
fdai@umich.edu

3rd Vaibhav Gurunathan
College of Engineering
University of Michigan
Ann Arbor, United States
gvaibhav@umich.edu

***Abstract*—This project extends the capabilities of masked autoencoders (MAE) based on the research paper ”Masked Autoencoders are Scalable Vision Learners”. We aim to enhance MAE in three key areas: data augmentation, masking approaches, and Generative Adversarial Networks (GAN) loss. First, we advanced the current data augmentation techniques to diversify the training data and improve model generalization. This enhances the model’s ability to learn diverse image representations given a more diverse training set. Second, we explore different masking strategies by introducing random and adaptive masking proportions. Unlike fixed masking values used in previous approaches, our method dynamically adjusts masking proportions, either randomly or based on the current training epoch. This dynamic adaptation enables the model to learn more effectively from occluded regions, improving reconstruction quality. Lastly, we integrate GAN loss into the MAE framework. By leveraging GAN’s superior image generation capabilities, we enhance the realism and detail performance of image reconstruction. This addition enables the model to learn from additional high-quality data generated by the GAN, further refining its representations and enhancing overall performance.**

I. Introduction

Self-supervised learning is a type of machine learning algorithm where the model attempts to train itself using segments of data compared to human labeling. Self-supervised learning has become more prominent, especially when labeled data is scarce.

In the field of self-supervised learning, masked autoencoders (MAE) have proved to be an effective tool to efficiently learn the intrinsic features of an image, especially when dealing with large-scale image datasets. MAE trains a model by masking portions of an image and reconstructing these masked portions, thus extracting useful representations of the data without the need for explicit labeling. However, although MAE performs well in feature extraction, it still has room for improvement in reconstructing details and maintaining image realism, especially when dealing with datasets with complex visual environments.

In this study, the Progress Robot Object Perception Samples (PROPS) dataset was selected, which is a dataset designed for robot visual recognition tasks and contains images from multiple angles and distances with rich scene variations and complex background information. The characteristics of this data pose additional challenges to the image reconstruction task, especially in terms of maintaining image detail and quality. To address these challenges and to improve the performance of MAE in terms of image reconstruction quality, this study proposes the integration of Generative Adversarial Networks (GAN) into MAE.

The main objective of introducing GAN is to utilize its powerful image generation capabilities to enhance the reconstruction of MAE. In this integrated model, the generator of GAN, also known as the original network framework of MAE, is responsible for reconstructing images to achieve a realistic effect; And the discriminator is dedicated to identifying whether the image is reconstructed or the original image, thus further enhancing the realism and detail performance of the reconstruction through adversarial training. This not only significantly improves the efficiency of robotic vision systems in dealing with complex environments, but also increases their adaptability and accuracy in real-world applications.

Through this study, we hope to show how the combination of MAE and GAN can overcome the limitations faced when using MAE alone and enhance the generalization ability and utility of the model by generating realistic reconstructed images. The experimental design, integration strategy, and experimental results on the PROPS dataset will be presented in detail, aiming to provide new perspectives and approaches to the field of self-supervised learning, especially for applications in robot visual recognition and handling tasks in variable environments.

II. Related Work

Masking Autoencoder (MAE) Originally proposed by He et al [1]. to improve the representation of image features through self-supervised learning, MAE learns the underlying structure and complex features of the data by randomly masking a portion of the input image (typically up to 75 percent), forcing the model to reconstruct the missing content. This method has shown excellent performance in a variety of vision tasks, especially on large-scale datasets such as ImageNet, where MAE significantly improves processing efficiency and learning speed by effectively reducing the redundancy of training data.

Masked autoencoders have been researched for their usability from videos to images to natural language processing with varying results [2], [3]. With videos, masked autoencoders have been very effective at learning. With natural language processing at scale, they are less effective. Masked autoencoders are still being researched in depth for their potential use cases.

Generative Adversarial Networks (GANs) Introduced by Goodfellow et al [4], GAN is a powerful generative model that mainly consists of a generator and a discriminator, both of which fight against each other during the training process. The generator is responsible for producing images that are as realistic as possible, while the discriminator tries to distinguish between real and generated images. GAN is particularly good at creating high-quality and detail-rich images and is therefore widely used in fields such as image synthesis, image super-resolution, and artistic creation. Despite its ability to generate high-quality images, the training process of GAN often faces the challenges of instability and pattern collapse.

III. Paper Reproduction

We thoroughly analyzed and executed the code provided by the paper, enabling us to replicate their reported results. Our reproduction effort includes visualizations of the original images, masked images, and reproduced images.

Firstly we tried a code that was sourced from the GitHub repository associated with the paper. Within this repository, two main files were crucial: ”pre-training” and ”regular-classification.” To ensure comprehensive replication, we executed both notebooks. These files leverage TensorFlow and Python to implement the masked autoencoder methods.

Pretraining: The pretraining notebook was extremely helpful in reproducing some of the original work. With the pre-training notebook, we were able to visualize the masked images compared to the original images. You can see how the masked images are created and how they hide information compared to the original images. Figure 1 shows a random sample of masked images next to their original counterparts.

With the pre-training notebook, we were also able to generate the learning-rate schedule graph. Although this is not directly in the paper, this is a very useful reproduction to show how the learning rate changes by step count. Figure 2 is the learning rate schedule for the pre-training model.

We also have the masked images with the reconstructed image generated and their counterparts. Figure 3 is a collection of examples of the original image, the masked image, and the reconstructed image. With the pre-training code, we were able to generate an accuracy of 40.98 percent.

Regular Classification: With regular classification, we ran the code and generated the corresponding accuracy of over 76.84 percent. This is in line with what is expected as the regular classification finetunes the results of the pretraining accuracy.

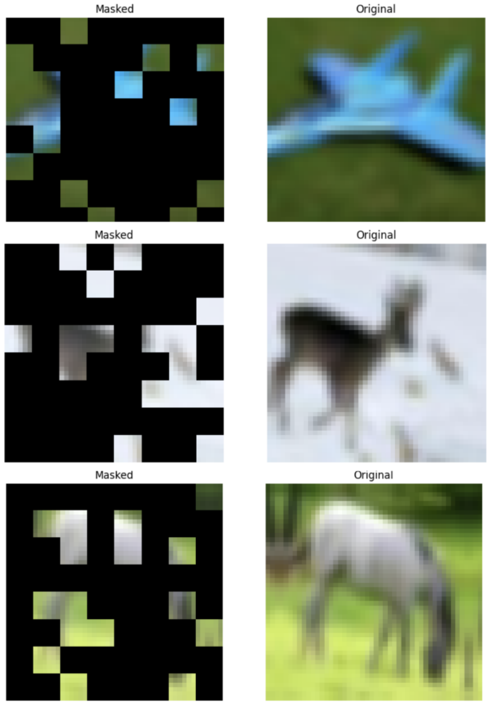

Fig. 1. Masked Images Next to Original Images

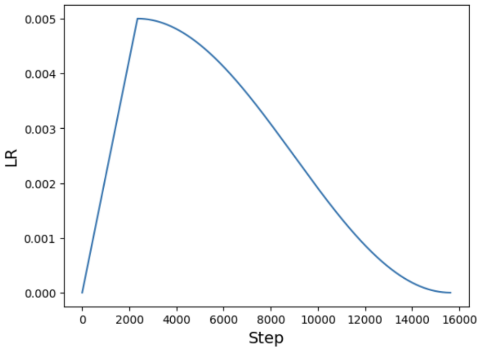

Fig. 2. Learning Rate Schedule for Pre-training

Integration of original model: After that, we reviewed the code of the original paper on GitHub and rewrote it, trying to integrate all files into an ipynb file. We modified the parameters of the original model and applied it to the PROPS dataset. After 30 epochs of training, we achieved an accuracy of over 70 percent at the beginning of the fine-tuning stage.

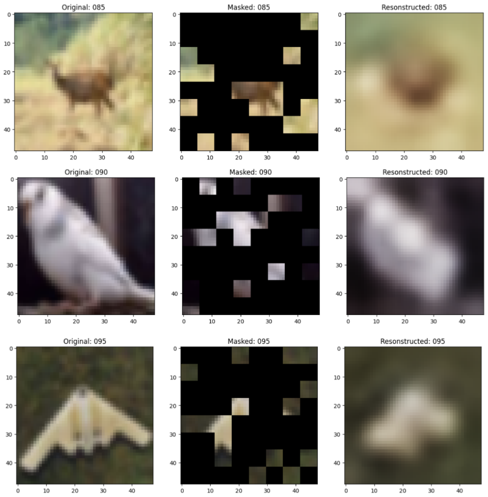

Fig. 3. Sample of original, masked, and reconstructed images

IV. Algorithmic Extension

A. GAN LOSS

Given a generator G and a discriminiator D, the loss function / objective functions to be minimized are given by

$$\begin{aligned}\mathcal{L}_{cGAN}(G,D) = \frac{1}{N}&\left(\sum_{i=1}^{N} logD(G\,(x_i), y_i)\right.\\ &+ \sum_{i=1}^{N} log(1 - D\,(\,G\,(x_i), y_i)\,)\end{aligned}$$

, where (x_i, y_i) refers to the pair to the ground-truth input-output pair and $G(x_i)$ refers to the image translated by the Generator.

$$\mathcal{L}_{L1}(G,D) = \frac{1}{N}\sum_{i=1}^{N} \parallel y - G(x_i) \parallel_1$$

The final objective is just a combination of these objectives.

$$\mathcal{L}_{final}(G,D) = \mathcal{L}_{cGAN}(G,D) + \mathcal{L}_{L1}(G,D)$$

$$G^* = \underset{G}{\operatorname{argmin}}\,\max_{D}\,\mathcal{L}_{final}(G,D)$$

In this model, the ”Input” is the token generated by the encoder, the ”Generator” is the decoder, and the goal is to make the image generated by the decoder as close to the original image as possible. Therefore, we added training a discriminator to the original model to enable it to recognize the reconstructed image as a fake image, and optimized the discriminator; After a step, we then optimize the generator with the goal of deceiving the discriminator.

B. Data Augmentation:

First we analyzed the current data augmentation methods and realized they were lacking. We added five more elements to help improve the robustness of the data set. These were random rotation, random zoom, random contrast, random brightness, and random translation. These were important because the picture will not always be in the exactly perfect frame every time. There are differences between cameras, lighting, etc that change the frame of the photos. Applying the data augmentation methods help catch and solve for those problems by improving the dataset.

We applied data augmentation techniques in two steps. In the first attempt, we only used random rotation, random zoom, and random contrast. In the next training attempt, we used all five data augmentation approaches. This was to evaluate the impact of adding more techniques or to determine if the data was already robust enough.

To implement these features, we imported keras from tensorflow. Using keras from tensorflow, we then imported layers. With these layers, we called the different function. For every functio we applied a value of 0.2 except for random rotation which applied a value of 0.15 and random height and width which had a value of 0.1. These numbers were picked as they are the standard choices used.

C. Adaptive Masking

This paper used very high masking ratios as used for the patch encoder. We wanted to try out using different masking ratios every time. We were not convinced that the set masking ratio was always the best approach so we tried to use a different masking ratio in every epoch. To do this, we used two different approaches.

The first was randomization. To do randomization, we initially set the current masking proportion to a random value. Then, after every epoch, we would set the next masking proportion to a different random value. Each random value was generated using the python random library. From here we called the random.random() function. This helped pick a random number from 0-1.

The second approach was to use a masking ratio that was dependent on the current epoch. Since the epochs ranged from 1-100, the masking ratio changed from 0.00 to 0.99. This allows the model to learn from a wide range of masking ratios. To do this, we simply set the current masking ratio equal to the current epoch divided by the total number of epochs. This also generated a number between 0-1.

V. Experiments and Results

A. Experimental Setup

We extended the paper in two key ways. First we improved the existing methods the paper implemented in pre-training. This refers to data augmentation and masking strategies. Then we introduced a new part, GAN loss, so that the model will be even more realistic as a scalable learner.

B. Results

One approach used the standard CIFAR10 dataset that was used in the repository provided. This was to measure the accuracy compared to what they already used. This dataset was loaded with the keras datasets library list. The other approach used PROPS dataset.

Data Augmentation: Both times we ran the code - with just three additional augmentation techniques and with all five augmentation techniques, generated the same final accuracy of 41.25 percent. This is an improvement over the 40.98 percent accuracy that was achieved with just the standard baseline augmentation techniques.

Adaptive Masking: As shown in figure 4, the first randomness method seemed to work effectively as it generated a higher accuracy on the pretraining data. Compared to the previous value of 40.98 percent, this generated an accuracy of 46.52 percent.

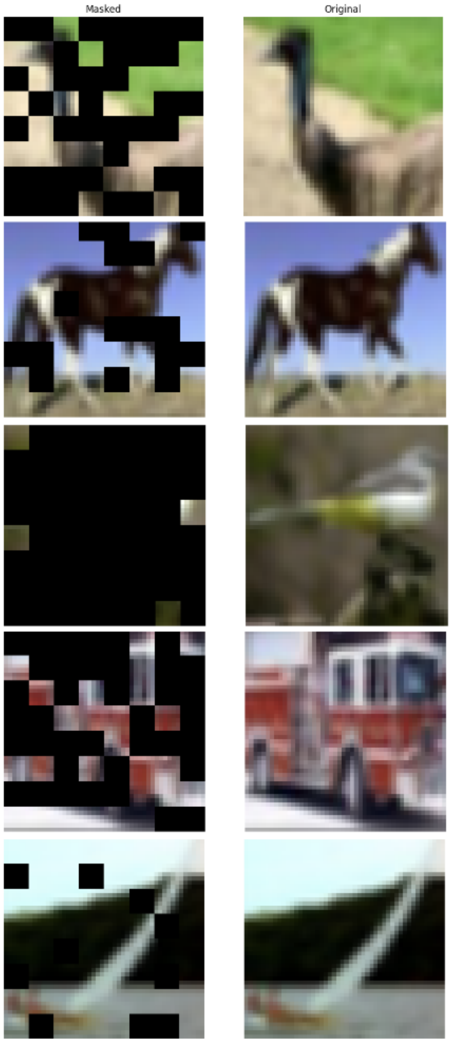

Fig. 4. Five randomly chosen images with randomly chosen mask proportions

The second method was also very effective and managed to generate the same accuracy of 46.52 percent, which is shown in the figure 5.

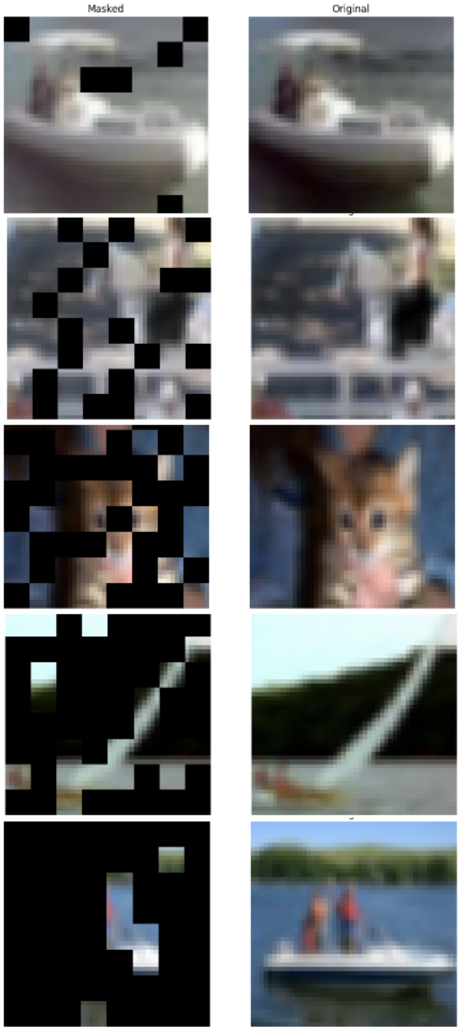

Fig. 5. Five images with a masking proportion increasing linearly from 0.1 to 0.9

Generative Adversarial Networks: After introducing generative adversarial networks into masked autoencoders, this study compared them using the Progress Robot Object Perception Samples dataset. The experimental results show that the weight between the generator and the discriminator loss should be at least 100:1, otherwise it will weaken the effectiveness of the generator. Using 50000 images as the training set and 10000 images as the testing set, experimental results did not show significant differences from the quantity perspective, as shown in Figure 6. However, in terms of the effect of reconstructing images, models with GAN loss seem to be better at handling blurry edges.

From a quality perspective, we tested using small-scale datasets and found that models with GAN loss were able to extract image features more significantly with less data, bigger masked patches, and fewer pre training epochs, and shown as Figure 7, the reconstruction effect was significantly better than the original models.

VI. Conclusions

This paper achieved two significant advancements. Firstly, we enhanced the performance of the masked autoencoders

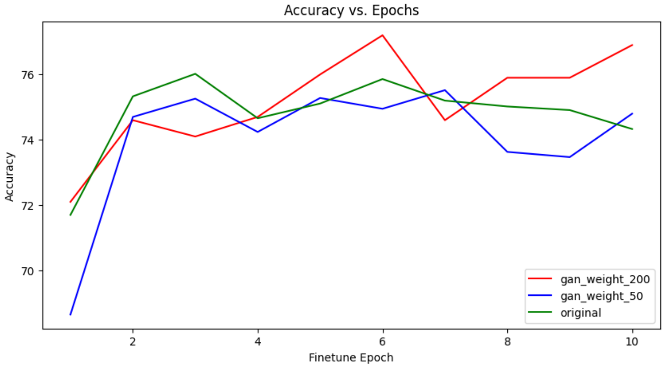

Fig. 6. The classification accuracy of models

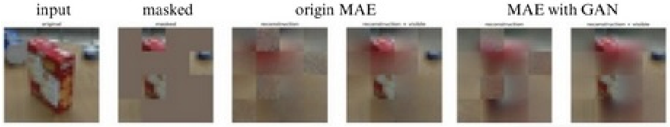

Fig. 7. Comparison of effects between models w&w/o GAN loss

model through a combination of advanced data augmentation techniques and adjustments to the masking ratio. These improvements are pivotal in enhancing the model's predictive accuracy, thereby advancing the state-of-the-art in this domain. However, there are opportunities for further refinement to align our work more closely with the methodologies employed in the referenced paper. Notably, the paper underscores the importance of employing notably high masking ratios. In contrast, our approach utilized a masking ratio of 0.5 across both algorithms, falling short of the 0.75-0.80 ratio advocated by the referenced work. To enhance comparability and ensure alignment with the referenced research, it would be beneficial to explore the performance differences achieved by employing algorithms with masking ratios within the range highlighted in the paper. This adjustment would afford a more nuanced understanding of our model's performance regarding the paper analyzed.

The second main action accomplished was to introduce GAN loss to masked autoencoder models. Subsequent papers have confirmed that GAN Loss can significantly enhance image reconstruction performance. However, due to limited computing resources, we can only use 32 * 32 images and barely can tune the discriminator. Therefore, in our results, GAN Loss does not significantly improve the reconstruction effect and classification accuracy after fine-tuning. But we were inspired to test in harsh environments with limited data, short training time, and increasing patch size to reduce semantics, and found that GAN Loss significantly improves the performance in such situations. By the way, it should be noted that this research only tested the GAN functions on the PROPS dataset. By testing it on the CIFAR-10 dataset, we would have the ability to compare the different techniques similar to the methods from before used to compare the improvements.

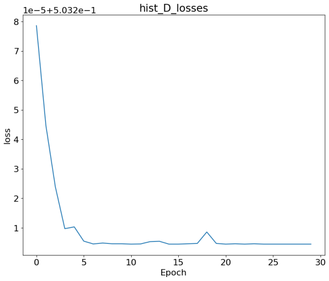

Fig. 8. Discriminator Loss

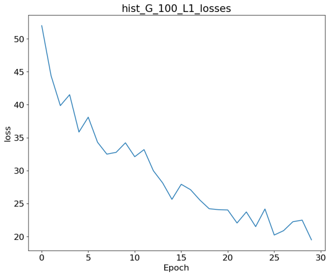

Fig. 9. MAE Loss

We completed this research because we are extremely interested in self-supervised learning. We believe that there is a lot of methods to improve on this work in the future. There are also many other future directions not related to the work we completed. One major idea we considered for this project was token-based or register based information. One potential area would be to convert patches to tokens such as in natural language processing. Then the model could relate the image corresponding to the token with the masked image it sees. In this way, it can have a type of guide for how the rest of the image may look. Another method is register based approaches that we thought of implementing. This method can save more global features, making the reconstruction more realistic and eliminating artifacts. This is extremely useful for the large masking proportions such as in this paper, and could help in being used to improve the masked autoencoder model.

VII. Appendix

https://github.com/Polarisyjr/Masked-Autoencoder

References

[1] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. Girshick, "Masked autoencoders are scalable vision learners," 2021.

[2] C. Feichtenhofer, H. Fan, Y. Li, and K. He, "Masked autoencoders as spatiotemporal learners," 2022.

[3] F. Weers, V. Shankar, A. Katharopoulos, Y. Yang, and T. Gunter, "Masked autoencoding does not help natural language supervision at scale," 2023.

[4] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial networks," 2014.