Masked Autoencoders Are More Than Scalable Vision Learners

Kho Tze Jit
A0215110E

Chia Yi Min Matthew
A0217187Y

Liaw Zheng Kai
A0222733M

Liu Han
A0194490X

Muhammad Haidi Bin Azaman
A0216941E

Foong Xin Yu
A0213920R

1 Introduction

The paper ”Masked Autoencoders are Scalable Vision Learners” [1] introduces the masked autoencoder (MAE) model to solve various Computer Vision (CV) tasks. The MAE involves heavily masking inputs and using Vision Transformers (ViT) [2] and scales well when performing tasks such as image classification, object detection and semantic segmentation. While this architecture showcases its successful application in the field of CV, we aim to extend this innovation to investigate if the masking concept can tackle tasks beyond CV and Natural Language Processing (NLP). Specifically, we explore its applicability in a 2D-Segmentation task and three other diverse domains: 3D-Volumetric data, Time Series Forecasting, and Data Imputation.

2 Background

2.1 Related Work

Introduced in 2010 by Pascal Vincent et al. [3], the MAE model utilises autoencoders to learn useful features by corrupting the input data in order to reconstruct the underlying uncorrupted data. Despite the data intensive training process, MAEs tend to show improved learning of generalizable features [4] and dimensionality reduction [5] making them versatile for a variety of machine learning tasks. The MAE model also encourages the combination of encoder learning and other pretraining tasks, simplifying the overall architecture as compared to the context autoencoder architecture [6].

2.2 Model Architecture

The masked autoencoder (MAE) model comprises several components, each contributing to its functionality in learning from input data. Below, we delve into these components and their roles within the MAE architecture:

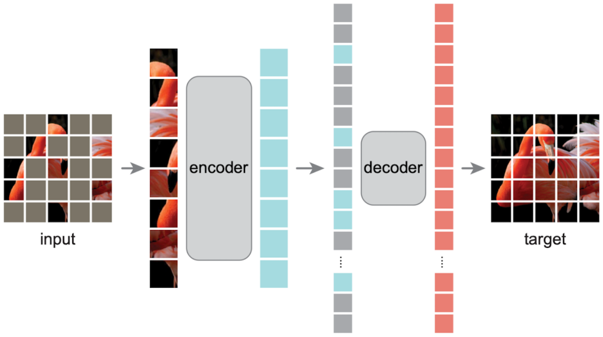


Figure 1: Proposed MAE architecture.

1. Input: The input image $x \in \mathbb{R}^{H \times W}$ is reshaped to a sequence of $N = HW/P^2$ non-overlapping patches $x_p \in \mathbb{R}^{N \times (P^2)}$, where N is the number of patches, (H,W) is the resolution of each image slice and (P,P) is the patch size. The patches are then linearly embedded, and positional embedding of these patches is added to capture the positional information.

2. Masking: A subset of the patches is masked and removed, while the remainder is fed into the MAE encoder. The random sampling technique is used as our masking strategy, with a sufficiently high masking ratio to ensure that the task would not be solved simply by extrapolation of unmasked neighbouring patches.

3. MAE Encoder: The MAE encoder is a vanilla ViT architecture applied only on unmasked patches. This significantly reduces the computational time and memory. After being fed into the MAE encoder, the unmasked patches are mapped into the latent space.

4. MAE Decoder: The MAE decoder takes in the encoded features of unmasked patches in the latent space and the mask token as inputs, where the mask token is a learnable and shared vector. A series of transformer blocks are adopted as the decoder.

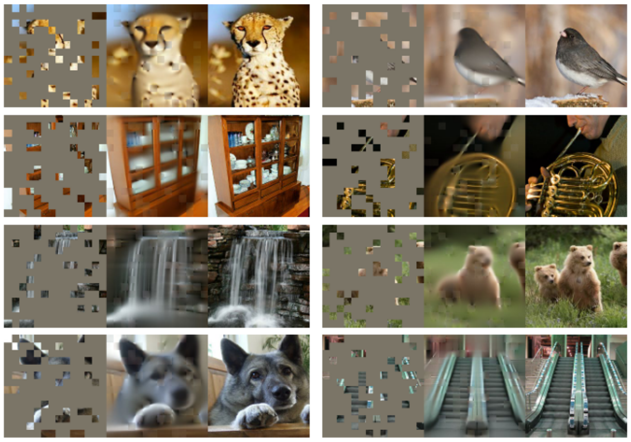


Figure 2: Example results from the paper "Masked Autoencoders are Scalable Vision Learners." Each triplet represents the masked input image on the left, the reproduced image in the center, and the original image on the right.

5. Reconstruction: Finally, the MAE predicts the pixel values for each masked patch to reconstruct the input volume. The mean squared error between the original and reconstructed image is then computed as the loss function, though this is only applied to the masked patches. Specifically, let $y^{input} \in \mathbb{R}^{HW \times 1}$ represent the input pixel values and $y^{pred} \in \mathbb{R}^{HW \times 1}$ represent the predicted pixel values. The reconstruction loss can then be written as

$$L_{reconstruction} = \frac{1}{\Omega(y_M^{input})} \sum_{i \in M} (y_i^{pred} - y_i^{input})^2 \quad (1)$$

where M represents the set of masked pixels, i represents the pixel index and $\Omega(\cdot)$ represents the cardinality of the set.

In the paper "Masked Autoencoders are Scalable Vision Learners", He et al. [1] masks random patches of input images and reconstructs the missing pixels using various asymmetric encoder-decoder architecture. By masking a high proportion of the input image ($\sim$75%), the approach manages to reduce training times and increases accuracy. With the vanilla ViT-Huge model achieving 87.8% accuracy on ImageNet-1K data, this approach presents an efficient yet accurate method that is hypothesized to be attributed to a rich hidden representation of data within the approach. Therefore, this paper aims to expand and explore the MAE approach to four other tasks outside of image reconstruction. These tasks include:

1. Time series reconstruction and prediction
2. Semantic Segmentation
3. 3D segmentation of medical CT scans
4. Generating appropriate samples for data imputation

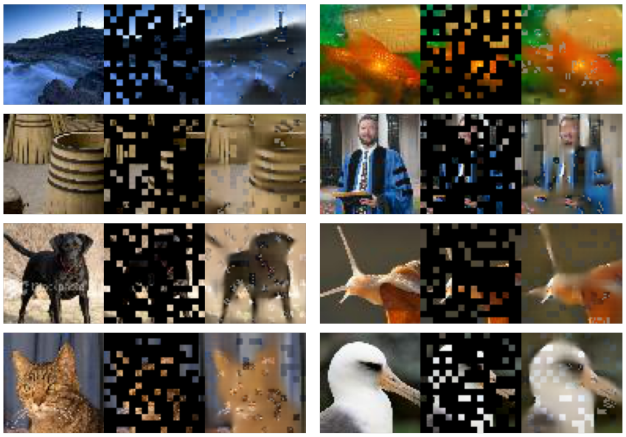


Figure 3: Example results of image reconstruction of TinyImageNet data using MAE architecture. For each triplet, the leftmost image displays the base image, the middle image displays the input data and the rightmost image displays the reconstructed image.

3 Reproduction

In this section, our objective is to replicate the primary experiment outlined in [1] in order to validate its efficacy on the dataset: TinyImageNet.

3.1 Methodology

The TinyImageNet dataset is a downscaled version of the original ImageNet dataset, which is a widely used benchmark dataset in the field of computer vision. While the original ImageNet dataset contains millions of images across thousands of categories, TinyImageNet is a reduced version that retains the same hierarchical structure and image categories but with a smaller scale consisting of around 200 image classes, each containing 500 training images, 50 validation images, and 50 test images. Each image in the dataset is 64×64 pixels reducing computational time required.

Self-supervised pre-training of the transformer is first conducted on the TinyImageNet training set. Subsequently, supervised training to evaluate the representations through end-to-end fine-tuning.

3.2 Results

The top-1 validation accuracy of a single 64 × 64 crop is reported below in Table 1. We demonstrate the superiority of MAE in capturing meaningful representations to heavily improve the classification performance. However, it should be noted that the performance did not manage to meet a similar accuracy in [1]. We argue that the downsizing of images in the TinyImageNet dataset might have caused the performance to be lower due to reduced spatial information and loss of detail. Figure 3 shows some qualitative results of the image reconstruction performance on the validation set after pretraining.

Table 1: Reproduction Results

ViT (trained from scratch)	Baseline MAE
37.7	45.6

4 Extensions

4.1 Time Series Forecasting

4.1.1 Objective

In this expansion of the MAE architecture, we explore the implications of randomly masking segments within a 1D time series signal in the context of a forecasting task. Traditional time series forecasting methods and modern neural network architectures like Recurrent Neural Networks (RNNs) and Long-Short Term Memory (LSTM) models, are effective for short prediction windows. However, their accuracy declines when forecasting over longer time horizons [7]. In these models, the concept of memory plays a crucial role. It prioritizes recent timestep values over distant ones when making predictions. This highlights the compounding effect of forecasting errors over time. As each forecast relies on preceding forecasts, any inaccuracies in earlier predictions are amplified in subsequent ones. To address this challenge, we propose that masking input values could encourage the model to leverage a broader context of neighboring values, rather than relying solely on preceding ones [8]. Furthermore, we suggest that utilizing a transformer architecture enables the capture of complex long-range temporal dependencies, a task that is often difficult for shallower neural networks. We run experiments on 1D univariate time series data.

4.1.2 Methodology

The ETTh1 dataset, denoting ”Electricity Total Load Forecasting - Time Series 1,” constitutes the primary dataset employed in this study [9]. This dataset comprises two years' worth of hourly electricity consumption data from two distinct counties in China. To model 1D time series data, we specifically use the Oil Temperature (OT) feature.

For data preprocessing, the data is split into chunks, with each chunk consisting of a specified lag period and the target value representing the subsequent data point after the last lag value. The dataset is partitioned with a distribution of 70%, 20%, and 10% for train, validation and test respectively. Random shuffling was not employed to preserve the temporal dependencies of data and ensure the integrity of the forecasting task.

Two distinct autoencoder models are developed for comparison. Both models utilize transformer blocks for the encoder and decoder components. One model incorporates the MAE masking implementation, while the other does not. The encoder processes the input time series data, while the decoder reconstructs the input sequence based on the encoded representations [10]. These models employ conventional transformers rather than Vision Transformers (ViT).

The metrics used are Mean Absolute Error and Mean Squared Error (MSE). Mean Absolute Error represents the average magnitude of absolute errors between predicted values and actual values, thereby offering a robust measure that is relatively less influenced by outliers. Conversely, MSE calculates the average of squared deviations between predicted and actual values, consequently imposing greater emphasis on larger errors.

4.1.3 Results

A comparative analysis of the forecasting task was conducted between models without Masked Autoencoder (MAE) architecture and with MAE at masking ratios of 0.75, 0.50, 0.25.

Table 2: Experimental results

Models	Mean Sqr. Err.	Mean Abs. Err.
no MAE	0.01538	0.10043
MAE, mask=0.75	0.07757	0.26845
MAE, mask=0.50	0.07546	0.26391
MAE, mask=0.25	0.07778	0.26851
RNN	0.00072	0.01822
LSTM	0.00071	0.01799
GRU	0.00071	0.01791

The results on Table 2 indicate that the strategy of masking does not lead to an improvement in forecasting accuracy across all three masking ratios tested. Visual inspection of the Figure 4 reveals that the model without MAE implementation predicts a time series closely resembling the original signal. In contrast, models employing MAE of different masking ratios produce predictions similar to one another but diverge significantly from the original time series. This suggests that while the effects of masking across different ratios are comparable, the masking mechanism itself does not confer benefits in enhancing time series forecasting for 1D time series data.

We further investigate by training lean architectures such as RNN, LSTM, and Gated Recurrent Units (GRU). Surprisingly, they exhibited near-perfect forecasting performance on the test dataset, as evidenced by the significantly smaller Mean Absolute Error and MSE values compared to transformer-based models. This unexpected finding contrasts with the challenges encountered in achieving satis-

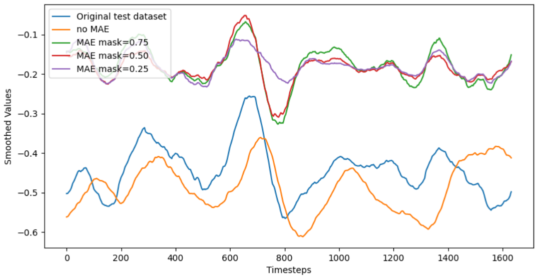


Figure 4: The smoothed models' forecast predictions on the test dataset is shown. A smoothing window has been applied using a moving average with a window size of 20 timesteps. This results in plots that only capture the general trend.

factory accuracy with transformer-based architectures. In scenarios where the dataset is relatively small, simpler recurrent architectures like RNN, LSTM, and GRU may generalize more effectively. This implies that the transformer-based architecture of MAE may necessitate more data for effective performance. It suggests that the ETTh1 dataset utilized might not have been sufficiently extensive to provide meaningful insights into the efficacy of the masking strategy employed by MAE. Therefore, future research would involve employing a dataset large enough to be adequately handled by transformer-based architectures, potentially revealing the benefits of masking. Additionally, exploring multivariate time series forecasting could be beneficial due to the richer information content it offers.

4.2 Semantic segmentation

4.2.1 Objective

The features extracted by MAE can be directly applied to other image related tasks. Among these tasks is semantic segmentation, which seeks to assign each pixel in an input image to a distinct semantic category. Given MAEs' inherent capability to capture and encode contextual information from input images, we believe that leveraging MAEs in semantic segmentation tasks should lead to performance improvements.

4.2.2 Methodology

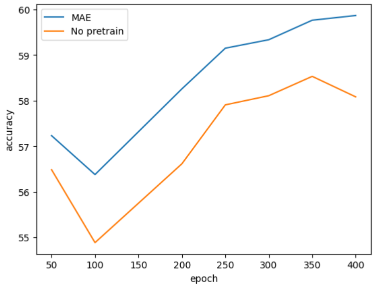


Figure 5: Graph of the accuracy of the UPerNet architecture with and without MAE on the semantic segmentation task.

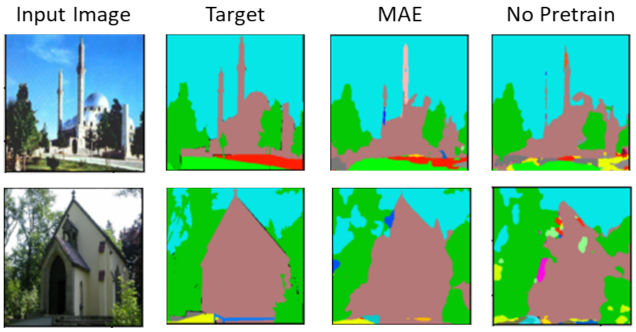


Figure 6: Example results of the models conducting semantic segmentation.

The approach used to tackle semantic segmentation is via the UPerNet (Unified Perceptual Parsing Network) architecture developed by Xiao et al. [11] on the ADE20K dataset [12]. This dataset consists of 20,210 training data images, each of which has its pixel labelled with 1 of 3,169 semantic labels. It covers a broad spectrum of object categories and scene types, including indoor spaces like bedrooms, kitchens, and living rooms, as well as outdoor scenes such as streets, parks, and landscapes. To evaluate model performance, pixel label accuracy and mean intersection over union (MIoU) will be the metric used, commonly used in other semantic segmentation evaluations.

UPerNet requires image features to be fed into a Feature Pyramid Network (FPN), which takes in features with dimensions exponentially increasing in powers of 2. In order to incorporate ViT used in MAE, some modifications were made. The output of the 3rd, 6th, 9th, and 12th transformer layers of ViT-B were modified and fed into the FPN as suggested by Li et al. [13]. The 12th layer was downsampled by a factor of 2 via a max pooling layer. The 9th layer was passed as it is, and the 3rd and 6th layers were upsampled by factors of 4 and 2 respectively using stride-two 2×2 transposed convolution layers. The dataset also contains images of various sizes. In order to fit them into ViT, all images were resized to 256-by-256 pixels.

To investigate the effectiveness of MAE on UPerNet performance, two identical ViT-UperNet architectures were trained for 400 epochs using the same seed, with the MAE model having its ViT architecture initialized using the pretrained weights.

Table 3: Semantic segmentation experiment results

Metric	MAE	No Pretrain
Accuracy	59.86	58.08
MIou	0.2888	0.2803

4.2.3 Results

From Figure 5, it is observed MAE pretraining is indeed effective for all epochs trained, leading to both a higher accuracy and MIoU score. This allows for better performance or shorter training time when using MAE pretrained weights. The final experimentation results are displayed on Table 3.

This can also be confirmed visually in Figure 6, where the output from the MAE model shows more accurate segmentation results. The segmentation output of the MAE model displays reduced patchiness and greater homogeneity, indicative of enhanced performance in recognizing complete objects. This improvement can be attributed to the masking mechanism inherent in MAEs, facilitating the extraction of informative features within the semantic segmentation task. By promoting resilience to occlusions and facilitating the learning of spatial dependencies within images, MAEs emerge as a promising avenue for augmenting the efficacy of semantic segmentation models.

4.3 3D Volumetric Segmentation of Medical Images

4.3.1 Objective

Since improvements to results have been achieved in semantic segmentation, we extended this use case to 3D semantic segmentation for the case of medical images. Zhou et al. [14] suggests that the contextual aggregation ability of MAE is valuable for medical segmentation as each anatomical structure is functionally and mechanically connected to other structures and regions. This section thus studies whether pretraining using an MAE will improve results of 3D semantic segmentation.

4.3.2 Methodology

In this regard, we utilize a dataset of CT scan images from Beyond the Cranial Vault (BTCV) to validate the performance of MAE. The dataset consists of 50 volumes, of which 30 are labelled. The scans consists of 13 different organs, with the aim of segmenting and labelling each organ in a volume.

To measure the performance of our models, we utilize the Sørensen–Dice metric, which is commonly used to evaluate the similarity between the segmented results and labelled mask [15].

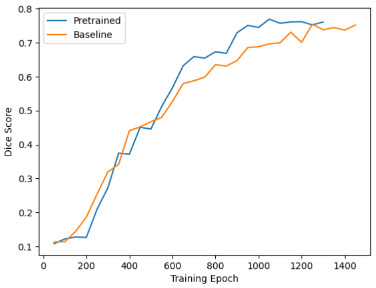


Figure 7: 3D Segmentation dice scores of various models.

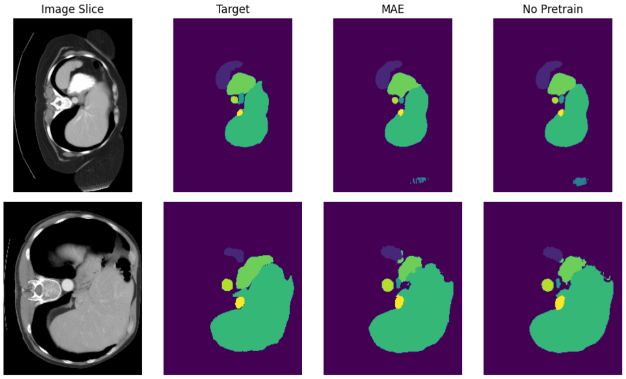


Figure 8: Example 3D Segmentation result.

A UNEt TRansformer (UNETR) architecture [16] is used to perform semantic segmentation by reformulating the segmentation task to a seq2seq task. It demonstrates excellent performance conducting multi-organ segmentation on the BTCV leaderboard. In addition, since it uses ViTs internally, we will be able to compare its performance with and without pretraining. A baseline model is first trained, using a ViT that has not been pretrained. Following this, a pretrained ViT is trained using a MAE on the scanned images, before being incorporated into the UNETR model for training on the segmentation task.

4.3.3 Results

Table 4: 3D Segmentation Experiment results

Metric	MAE	No pretraining
Dice Score	0.7712	0.7548

The validation Dice scores across training epochs are shown in Figure 7, along with sample segmented outputs in Figure 8. The calculated metrics are shown in Table 4. It is observed that the pretrained ViT is able to achieve slightly better dice score as compared to the baseline ViT, in fewer number of training epochs, thus validating the benefits of pretraining using a MAE for 3D semantic segmentation tasks.

4.4 Data Imputation

4.4.1 Objective

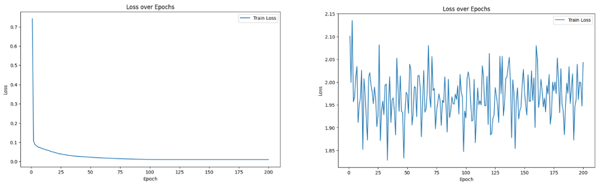
In this section, we attempt to use MAE for data imputation. Using the California Housing Price dataset, we investigate if MAE can accurately impute missing data.

4.4.2 Methodology

The data was preprocessed by removing missing values and categorical data was one-hot encoded. Features were normalised to a uniform range between 0 and 1. To simulate missing data in real-world scenarios, masks with a Bernoulli distribution of 80% probability of data retention were used to introduce a controlled amount of missingness (20%). The masked data points were then set to zero. Two neural network models were compared - an unmasked and masked model. The baseline unmasked feed-forward model mapped input dimensions to the same output dimensions with two fully connected layers with ReLU activation. On the other hand, the masked autoencoder model only takes in the input of unmasked points.

4.4.3 Results

The training performance of the models is displayed in Figure 9. The plot on the left illustrates a clear trend of decreasing loss for the baseline model while the plot on the right demonstrates significant fluctuations of loss throughout the training process of the MAE model. Furthermore, there is no sign of systematic stabilization with the MAE model.



(a) Baseline Model

(b) MAE Model

Figure 9: Training loss over epochs comparison

The MSE loss and Wasserstein Distance were used as metrics to gauge model performance in accuracy and distributional similarity. The Wasserstein Distance (or Earth Mover's distance), measures the distribution difference between the reconstructed outputs and the original data, providing a metric of the model's imputation quality across features. In table 5, the baseline imputer model is observed to return better results, with a lower MSE loss and Wasserstein distance.

Metric	Baseline	MAE
MSE Loss	0.0093	0.5171
Wasserstein Distance	0.2482	0.3098

Table 5: Evaluation Metric Comparison

To address the limitations observed in the efficacy of the MAE model, several reasons are proposed. Firstly, the masking strategy may not accurately capture the true nature of missing data within the dataset. Secondly, MAE may be sensitive to noise and outliers in the data, which can adversely affect its ability to accurately impute missing values. Future research could entail employing a dataset with little to no noise and outliers, or tweaking the masking strategy to more closely align with the inherent nature of missing data in the dataset.

5 Conclusion

The MAE architecture is hypothesized to present an efficient yet accurate method that is attributed to a rich hidden representation of data. As the MAE architecture was only tested on the image reconstruction task by He et al. [1], we aimed to validate the hypothesis on other tasks and handle various data modalities.

The MAE architecture was extended to forecast 1D time series data, implemented on 2D and 3D segmentation tasks as well as data imputation of datasets with missing values. Only on the 2D and 3D segmentation tasks do we observe an increase in performance. On the time series forecasting and data imputation tasks, the MAE architecture does not seem to provide any benefit and instead brings adverse performance.

Our experiments show that the MAE architecture favours image-related tasks and is adverse to other data modalities. This may be because image data inherently has heavy spatial redundancy [9]. For instance, a single pixel within an image typically holds less semantic meaning, making it possible to infer missing regions using nearby pixels through interpolation without fully grasping the content. As such, a high masking ratio for image applications encourages MAE to largely eliminate redundancy and prevent it from focusing only on low-level semantic information. Conversely, this effect is reversed for time series and tabular data, which exhibit characteristics akin to natural language in terms of high information density. In these modalities, each data point contributes valuable insights into trends and patterns. Therefore, a high masking ratio may hinder the MAE's learning process in these contexts. Overall, while the MAE architecture excels in image-related tasks, its efficacy across diverse data modalities remains a subject for further exploration and refinement.

References

[1] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. Girshick, "Masked autoencoders are scalable vision learners," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, New Orleans, LA, USA, Jun. 2022, pp. 16 000–16 009. DOI: `10.1109/CVPR52688.2022.01553`.

[2] A. Dosovitskiy, L. Beyer, A. Kolesnikov, *et al.*, *An image is worth 16x16 words: Transformers for image recognition at scale*, 2021. arXiv: `2010.11929 [cs.CV]`.

[3] P. Vincent, H. Larochelle, Y. Bengio, and P.-A. Manzagol, "Extracting and composing robust features with denoising autoencoders," in *International Conference on Machine Learning proceedings*, 2008.

[4] Z. Tong, Y. Song, J. Wang, and L. Wang, "Videomae: Masked autoencoders are data-efficient learners for self-supervised video pre-training," *ArXiv*, vol. abs/2203.12602, 2022. [Online]. Available: `https://api.semanticscholar.org/CorpusID:247619234`.

[5] C. Feichtenhofer, H. Fan, Y. Li, and K. He, "Masked autoencoders as spatiotemporal learners," *ArXiv*, vol. abs/2205.09113, 2022. [Online]. Available: `https://api.semanticscholar.org/CorpusID:248863181`.

[6] X. Chen, M. Ding, X. Wang, *et al.*, "Context autoencoder for self-supervised representation learning," *ArXiv*, vol. abs/2202.03026, 2022. [Online]. Available: `https://api.semanticscholar.org/CorpusID:246634394`.

[7] Z. Liu, Z. Zhu, J. Gao, and C. Xu, "Forecast methods for time series data: A survey," *Ieee Access*, vol. 9, pp. 91 896–91 912, 2021.

[8] P. Tang and X. Zhang, "Mtsmae: Masked autoencoders for multivariate time-series forecasting," in *2022 IEEE 34th International Conference on Tools with Artificial Intelligence (ICTAI)*, IEEE, 2022, pp. 982–989.

[9] Z. Li, Z. Rao, L. Pan, P. Wang, and Z. Xu, "Ti-mae: Self-supervised masked time series autoencoders," *arXiv preprint arXiv:2301.08871*, 2023.

[10] M. Zha, S. Wong, M. Liu, T. Zhang, and K. Chen, "Time series generation with masked autoencoder," *arXiv preprint arXiv:2201.07006*, 2022.

[11] T. Xiao, Y. Liu, B. Zhou, Y. Jiang, and J. Sun, *Unified perceptual parsing for scene understanding*, 2018. arXiv: `1807.10221 [cs.CV]`.

[12] B. Zhou, H. Zhao, X. Puig, *et al.*, *Semantic understanding of scenes through the ade20k dataset*, 2018. arXiv: `1608.05442 [cs.CV]`.

[13] Y. Li, S. Xie, X. Chen, P. Dollar, K. He, and R. Girshick, *Benchmarking detection transfer learning with vision transformers*, 2021. arXiv: `2111.11429 [cs.CV]`.

[14] L. Zhou, H. Liu, J. Bae, J. He, D. Samaras, and P. Prasanna, *Self pre-training with masked autoencoders for medical image classification and segmentation*, 2023. arXiv: `2203.05573 [eess.IV]`.

[15] L. Baskaran, S. Al'Aref, G. Maliakal, *et al.*, "Automatic segmentation of multiple cardiovascular structures from cardiac computed tomography angiography images using deep learning," *PLOS ONE*, vol. 15, e0232573, May 2020. DOI: `10.1371/journal.pone.0232573`.

[16] A. Hatamizadeh, Y. Tang, V. Nath, *et al.*, *Unetr: Transformers for 3d medical image segmentation*, 2021. arXiv: `2103.10504 [eess.IV]`.