Published in Image Processing On Line on YYYY–MM–DD.
Submitted on YYYY–MM–DD, accepted on YYYY–MM–DD.
ISSN 2105–1232 © YYYY IPOL & the authors CC–BY–NC–SA
This article is available online with supplementary materials,
software, datasets and online demo at
https://doi.org/10.5201/ipol

Ilias Sarbout[1], Miguel Colom[1]

[1] Centre Borelli, ENS Paris Saclay, France

PREPRINT April 11, 2023

In this paper we analyze Transformers-based masked auto-encoders (MAE) for low-resolution images. They are presented in the literature as self-supervised learners improving supervised models performances through pre-training. We show that pre-training an image segmentation model with MAE increases its performance.

1 Introduction

Transformers architectures have pushed the development of unifying model architectures to solve problems in natural language processing and computer vision [25, 18, 24]. Transformers [23] have been successfully introduced as a general learning building block in both language and vision.

For self-supervised representation learning, the masked auto-encoding methodology in BERT [10] has been shown effective for learning text representations, and later also to learn visual representations [2]. Recently, it was shown that the introduction of visual (discrete) tokens is not necessary in image processing tasks. Therefore, very simple transformer-based masked auto-encoder (MAE) can learn robust image representations [15].

MAEs are currently among the most-used self-supervised learning models in computer vision. Image classification after fine-tuning on Imagenet reaches its best performances with pre-trained MAE models.

2 Related work

Transformer architectures

Transformer architectures have reached unseen performances for natural language processing (NLP) tasks since their introduction in 2017. The introductory paper [23] proposed in the context of NLP to put aside recurrent neural networks. The transformer blocks only rely on attention mechanisms. In [11], it was proposed to favor attention mechanisms instead of classic convolutional neural networks (CNN) in image processing. Unlike CNNs which are based on local and translation-invariant operations, the transformer-based models only use global attention.

Self-supervised Transformers for vision

After the introduction of Transformers [23], self-supervised language and vision modeling has made substantial progress in recent years. Pre-training on a large corpus of unlabeled images gives image representations that generalize well to downstream tasks in low data regimes [16]. Inspired by their success in NLP, a large diversity of transformers-based self-supervision methods processing were proposed for image processing [8]. Image GPT (iGPT) [5] use a transformer decoder to reconstruct 32×32 pixel images. Vision Transformers (ViT) [11] were introduced with the aim to keep transformer architectures as close as possible to its initial NLP formulation. The sequences are no longer at the pixel level but at the image patch level.

Masked auto-encoders

In [11], ViT self-supervision with masked patch prediction was explored for feature extraction and showed great performance. This track has been further explored in BEiT [2] and Vision Transformers MAE (ViTMAE) [15], successfully performing image reconstruction on large amount of unlabeled images. ViTMAE obtained impressive performance in pre-training for image classification. Recent works successfully attempted to extend transformers based MAE for images [4], videos [12, 22, 14], point clouds [17], and text-image pairs [13].

3 Representation learning problem

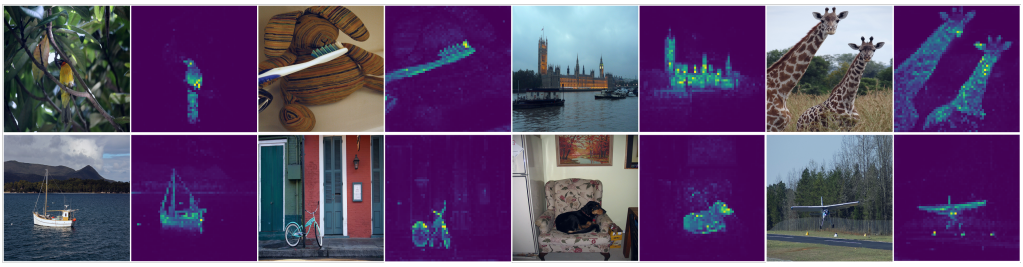

Figure 1: DINO self-attention maps (no supervision)

There is a significant inferential bias in directly evaluating self-supervised learning representations on supervised tasks, as self-supervised learning is not intended or designed to solve supervised problems. We can evaluate self-supervised methods abilities to facilitate supervised training, in terms of time or performances, but this evaluation remains incomplete, as the comparative results could be sensitive to the dataset or the task itself. Also, some contexts may require fully unsupervised image descriptors.

Effect of fine-tuning is also important. One could say that an evaluation based on supervised criteria is at least complete for a given task such as classification, arguing that some models are more suitable for it. For example, the attention maps from DINO (see figure 1) clearly show that the model representations are based on main objects within the image, which makes image representations very suitable for object classification. Evaluation results in Imagenet object classification confirm this visual hypothesis, but after finetuning, models whose representations are more abstract in terms of attention – such as MAEs, see table 3 – outperform DINO, which is also not very efficient for scene-scale tasks.

Model	Top-1 acc. before fine-tuning	top-1 acc. after fine-tuning	Parameters
MAE [15]	76.6%	**87.8**%	632M
SimCLRv2 [7]	**79.8**%	83.1%	795M
DINO [3]	78.2%	82.8	84M

Table 1: Performances on Imagenet classification

Therefore, representations analysis must also be based on the model's properties. This study – rarely performed because it does not provide objective indices of comparability between different models – is essential both to understand the behavior of deep architectures, and to identify essential properties whose causes help us define new efficient methods.

We argue for the fact that in the field of self-supervised image learning, Transformers based MAEs provide good descriptors in inferential terms, as it was already proven [citer], and we try to show in this paper that these models also have good properties at the patch and image scale, and that it is possible to directly use the descriptors from self-supervised learning for efficient dimension reduction.

4 ViT masked auto-encoder

ViTMAE consists in a ViT encoder and a ViT decoder. Figure 2 shows the general architecture. The encoder takes as input unmasked tokens and encodes them. The decoder adds masked patches to their original position and try to reconstruct them. Both encoder and decoder use 2D positional embedding. MSE loss is only computed with masked patches, similarly to BERT.

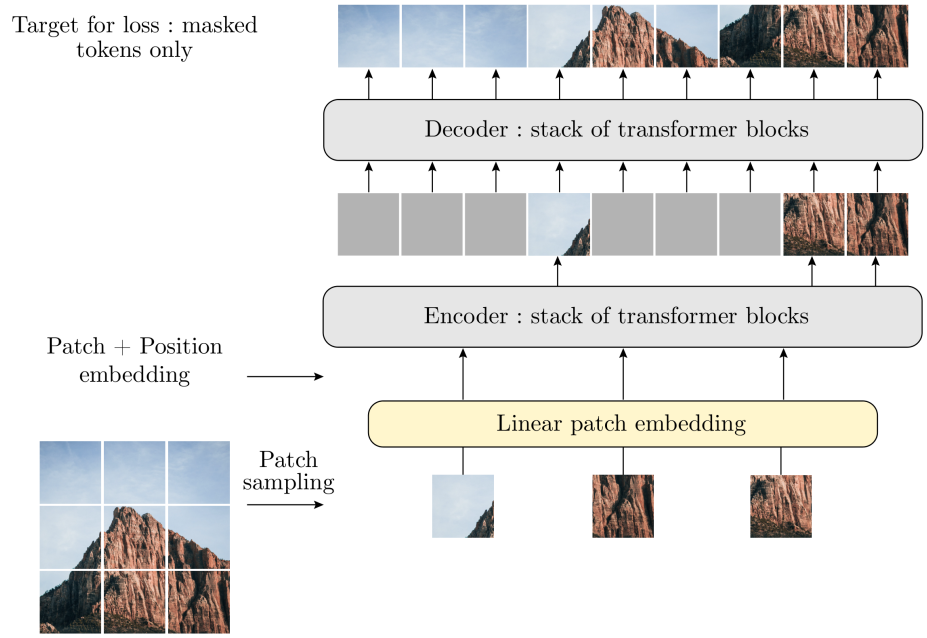

Figure 2: General architecture of ViTMAE.

With ViTMAE, it was argued [15] that using discrete tokens for mask prediction – the way it was proposed in BEiT – has no advantage over predicting masked tokens at pixel scale. MAE training need a high masking ratio (75% is optimal according to classification linear probing criteria [15]) to be efficient. Generally, the best masking ratio increases with the inputs dimensionality (15% for NLP [10], 75% for images [15], 90% for videos [12, 22]).

In transformer blocks, classic multi-head attention are defined as follows:

$$\begin{aligned} \text{Multihead}(X) &= \text{Concat}\left(\text{ head }_1, \ldots, \text{ head }_h\right) W^O \\ \text{where head}_i &= \text{Attention}\left(XW_i^Q, XW_i^K, XW_i^V\right) \\ \text{and Attention}(Q,K,V) &= \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right) V. \end{aligned}$$

Experimental setup

We plan to train a ViTMAE model on low resolution image and another encoder-decoder model (segmenter) for image segmentation friom scratch. Then, we will analyze the effect of MAE pre-training for image segmentation by replacing the segmenter's encoder by our MAE encoder before training.
We train a transformer-based MAE with 64×64 pixel images from Imagenet, with patch size of 8.

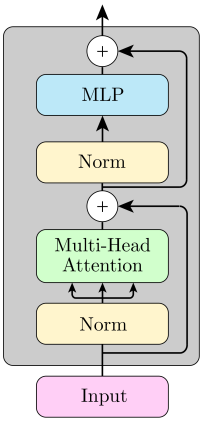

Figure 3: Transformers block

The MAE's encoder is 10 layers deep, each with 12 heads transformer blocks and embedding dimension of 240. The decoder is 6 layers deep, 16 heads, and embedding dimension 160. Layer normalization is used. It has a total of 8915872 parameters.

We train for 100 epochs, with the ADAM optimizer, weight decay, linear scheduler. We start training with 2 warmup epochs and initial learning rate of $1.5e^{-4}$. Patch sampling follows a uniform distribution. We perform data augmentation with random horizontal flips and random crops.

Figure 4: Reconstructed images from Imagenet validation dataset

5 Embedding tuning and analysis

We can express effect of single-head self-attention on an image set of n unmasked tokens of dimension d, the way it was implemented on our work. The set of unmasked patches is a matrix X.

$$\text{Self-attention}(X) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V = \text{softmax}\left(\frac{XW_qW_k^TX^T}{\sqrt{n}}\right)XW_V$$
$$= W_{att}X_v$$

Embeddings modelisation of Transformers

Each new image patch is now almost a linear combination of all other patches, except that this isn't directly about patchs but their projection into a value matrix X_v. It appears that under IID hypothesis on input, this formula may be conducive to the creation of normal distributed outputs, according to the central limit theorem.

Experiments show that no matter of attention layer initialization and input chosen distribution, self-attention tends to produce normal distributed output with independent inputs. Figure 6 show output of attention mechanism on a sample that contains 4 tokens of dimension 16000 for visualization, with uniform distributed input and uniform initialization on attention layer. We performed the Kolmogorov-Smirnov (KS) test to challenge the hypothesis of these outputs being normally distributed, using now 1000 chi2 sampled inputs and each sample containing 4 tokens of dimension 160. Distribution parameters are estimated independently for each sample. Results in figure 7 show that not a single sample conducted to reject our hypothesis.

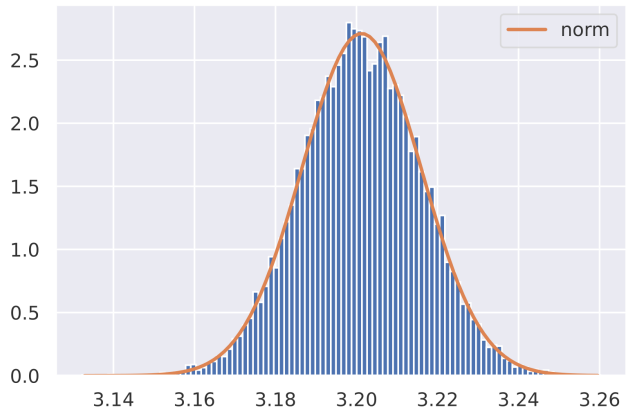

Figure 5: output of an attention layer on a uniform sampled input

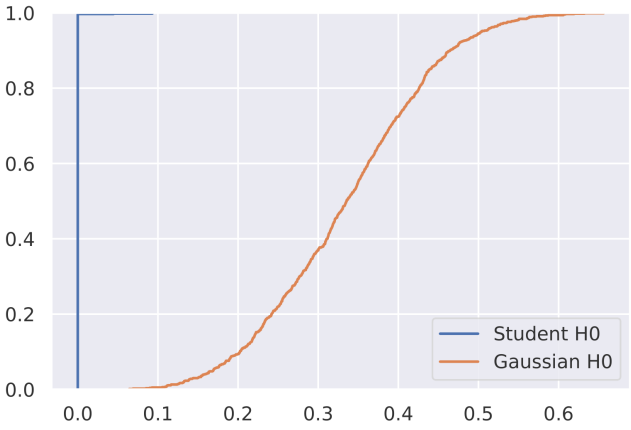

Figure 6: p-values empirical distribution function for gaussian and student KS tests for a single attention layer on chi2 distributed inputs

Using a wisely chosen layer normalization [1] can thus help us to obtain well-shaped descriptors. We use parametrized (γ, β) centering layer normalization defined as below :

$$\text{Norm}(x) = \frac{x - \mathbb{E}[x]}{\sqrt{\text{Var}[x] + \epsilon}} * \gamma + \beta$$

Under the hypothesis that x follows gaussian distribution, and neglecting $\epsilon = 1e - 5$, one could see $\text{Norm}(x)$ as a $t-value$ but this is not the case. Experiments shows that student distribution actually emerge from a combination of elements of which normalization is part. We make an experiment on 6688 image samples from weather dataset[1], which is a dataset containing scenes of 11 different weathers. variable x now denotes ViTMAE's encoder output. Figure 7 shows the empirical distribution of the p-values obtained with the KS test for student and gaussian distributions. Sample parameters are estimated individually. Results shows that the student hypothesis is not rejected on more than 83% of the initial samples for a 5% confidence level (0% for gaussian H_0 hypothesis). Figure 8 and table 2 compare most suitable densities according to KS test for a random sample histogram.

[1] https://www.kaggle.com/datasets/jehanbhathena/weather-dataset/metadata

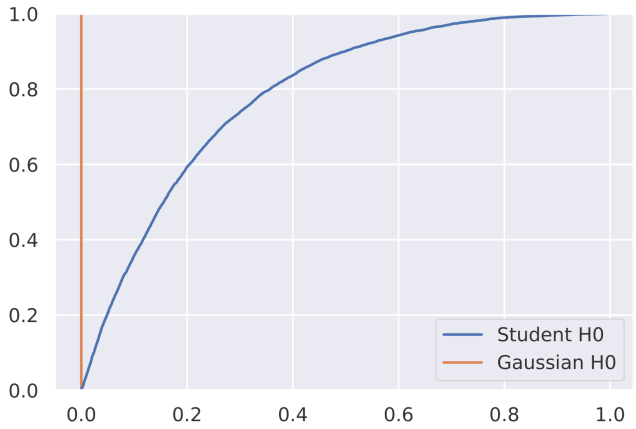

Figure 7: p-values empirical distribution function for gaussian and student KS tests on real data encoder's output

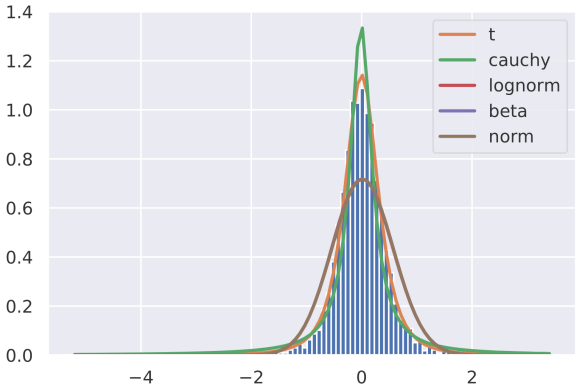

Figure 8: Density of most suitable distributions over a sample histogram

Distribution	Sum of square errors
t	**0.040851**
cauchy	0.403937
lognorm	0.813929
beta	0.819374
norm	0.823028

Table 2: Sum of distribution function square errors

Patch attention properties

First we perform a small experiment to find a standard way of masking patches. We compute a global attention matrix W per decoder's layers as mean of attention matrixes of each head. We extract *most important patches* by summing on columns of W and taking indexes of the 25% biggest values. It means these patches are the most used ones for reconstructing the other ones. We compute on Imagenet validation dataset the frequency of the most important patches matching exactly the 25% unmasked tokens on the decoder's first layer. For random masking perfect matching is done with accuracy 0.9900 and for grid masking (as done in figure 4) it is done with frequency 0.9998. This accuracy shows the model is **consistent** data information is distributed. According to these results, we now use grid masking that seems to have better performances about consistency.

We also perform a small visualization experiment to extract other qualitative properties. First, we extract most important patches from W matrix again. For each other patch (lines) we take top $-n$ attention values and their indexes, that give information on which patch influence them the most. Finally, we visualize the results as influence field of each most important patches with colorization. For each patch we display color of the patch that influences it the most. A patch may remain black if no one of his top$n$ influencer patchs was one of the *most important ones*. For our experiments we

Figure 9: image reconstruction attention maps over a 6 layers (left to right) MAE decoder

use $n = 5$.

Results from random Imagenet validation samples are displayed in figure 9. From them we can extract 2 other qualitative properties. **Locality** denotes the fact that influence fields of most important patches tends to be in their direct neighboring.

We can make some hypothesis when looking at the attention maps : consistency and locality properties tend to weaken through decoder's layers. Also, attention between not most important patches tends to increase through decoder's layers, which is visible my the apparition of more black patches on the figure through layers. To challenge these hypothesis, we perform an experiment on Imagenet validation dataset. We evaluate locality one each layer with the proportion of influenced patches belonging to the 3×3 neighborhood of their influencer patch. We evaluate consistency with the proportion images being fully consistent (the most important patches match exactly the unmasked patches) on each layer. We also compute the proportion of black patches on each layers. Black patchs proportion and locality are evaluated only for fully consistent images, to hide consistence's effect on locality, which is non negligible. Results are displayed in figure 5. Results show that lower layers of decoder tend to learn more local features, in accordance with previous work on this topic [19].

Most important patches colors depends on their importance. It seems that there is a positional bias as we observe strong similarities between images in color's locations, which is confirmed by color's location not matching uniform distribution on KS test. Also there could be a positional bias influence fields shapes, which is not evaluated here.

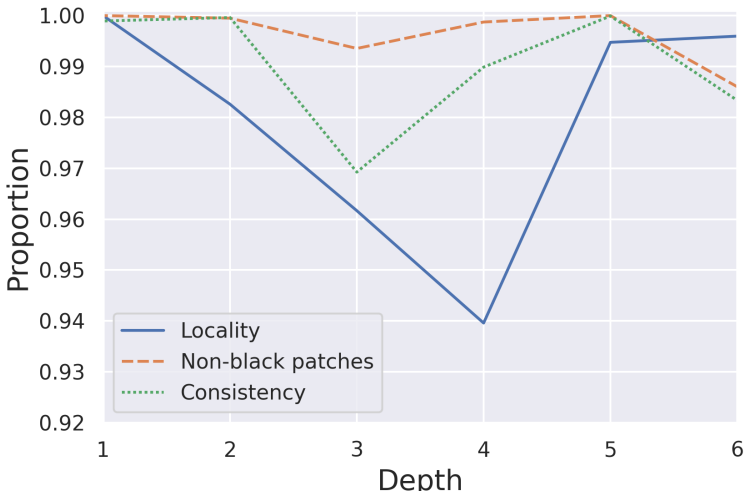

Figure 10: Attention properties through layers

6 Applications

Image similarity search

We can easily compute similarity between image embeddings and expect the two images to have similar structures. A small experiment on Imagenet data was done using cosine simiaarity between encoder's outputs with 25% masking ratio, results are shown in figure 10.

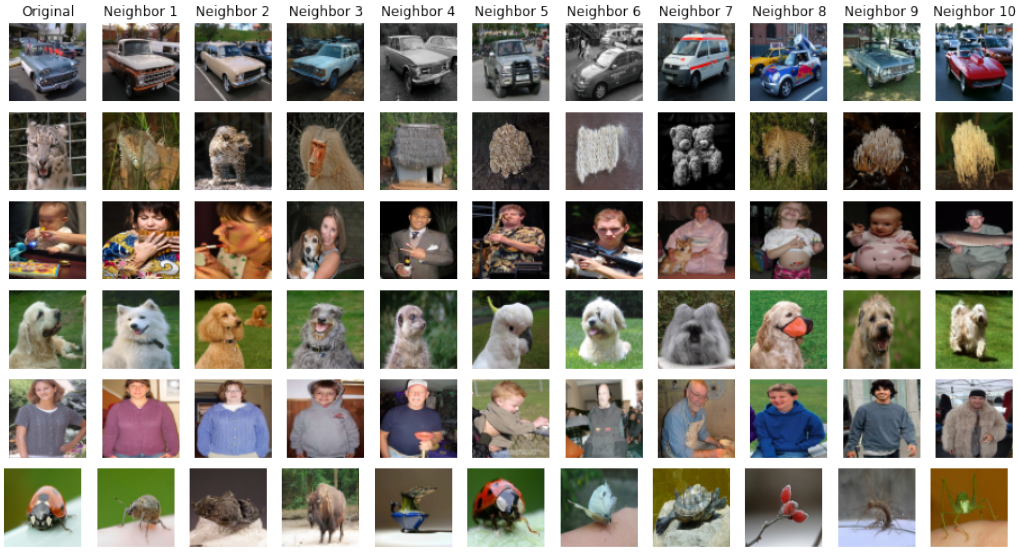

Figure 11: Similarity search results

Note that it is quite possible to compare the patches differently and mask the image more or less. For example, although it is trained with a masking ratio of 25%, the model's projections are more efficient in knn classification when the image is not masked at all. There are also ways to do object tracking and matching, in particular with smaller patch size models.

Pre-training

To show MAE's self-supervised pre-training abilities, we train three Transformers-based models for segmentation on ADE20K dataset, using Segmenter architecture [20] that also use an encoder-decoder architecture. To analyze influence of pre-training, we train image segmenters from scratch and with pre-trained encoder. This encoder is thus pre-trained with a Transformers based MAE. We also studied influence of colorimetric space. When pre-training with Y, C_b, C_r channels that are linear transformation of RGB channels, we got same performances on both RGB image reconstruction and segmentation loss. However, when using a weighted MSE (MSE_w) giving more importance to the luminance channel Y, convergence was slightly different. In our experiment, it was faster, but more validation will be required on this topic. Figure 6 show the results of training over 120 epochs for the three models.

$$\begin{aligned} Y &= (65.481/255)R &&+(128.553/255)G &&+(24.966/255)B &&+16/255 \\ C_b &= -(37.797/255)R &&-(74.203/255)G &&+(112/255)B &&+128/255 \\ C_r &= (112/255)R &&-(93.786/255)G &&-(18.214/255)B &&+128/255 \end{aligned}$$

$$MSE_w = 1.20(\hat{Y} - Y)^2 + 0.90(\hat{C_b} - C_b)^2 + 0.90(\hat{C_r} - C_r)^2$$

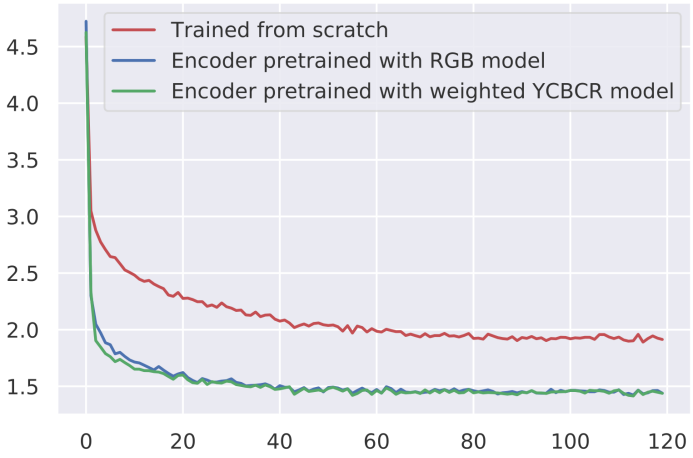

Figure 12: Segmenter loss on ADE20K evaluation set

Dimension reduction for visualization

We performed some experiments on dimension reduction and compared the results with the features obtained with a contrastive model SimCLR, based on Resnet50 architecture. We choose SimCLR [6] because its performances can be considered as close to MAE's ones. The reason is that on object-based image classification on Imagenet, SimCLR and ViTMAE have quite comparable performances (76.5% vs 76.6%) before finetuning. Also, the two models have comparable sizes (11504832 parameters vs 8915872). The SimCLR architecture is based on Resnet-50 architecture. The loss function is not based on original image like with an auto-encoder but ensures that similar images get similar representations. Assume a pair (x_i, x_j) considered as similar and representations of them (z_i, z_j). Let sim() be a similarity function such as cosine similarity. We can exprress loss for a positive pair as below :

$$\ell_{i,j} = -\log \frac{\exp\left(\text{sim}\left(\boldsymbol{z}_i, \boldsymbol{z}_j\right)/\tau\right)}{\sum_{k=1}^{2N} \mathbb{1}_{[k \neq i]} \exp\left(\text{sim}\left(\boldsymbol{z}_i, \boldsymbol{z}_k\right)/\tau\right)}$$

where τ is a temperature hyperparameter. The batch contains $2N$ datapoints where the positive pairs are sampled with data augmentation, performed with combination of three operations : random cropping followed by resize back to original resolution, random color distortions and random gaussian blurs

We show on one side that ViTMAE features outperforms by far SimCLR ones for image scene recognition, and also show that using methods that suit well ViTMAE's feature properties for dimension reduction gives better results. We experiment on 20000 samples from 4 classes of Places365 training dataset (airfield, amphitheater, aquarium, baseball field). We perform dimension reduction to visualize data on a plan, and perform *k-nearest neighbor* (k=20) to compute classification scores. We perform dimension reduction through *PCA* to 50 axes and then *t-sne* to project principal components a plan. Two dimensional projection are displayed in figure 6 and 6.

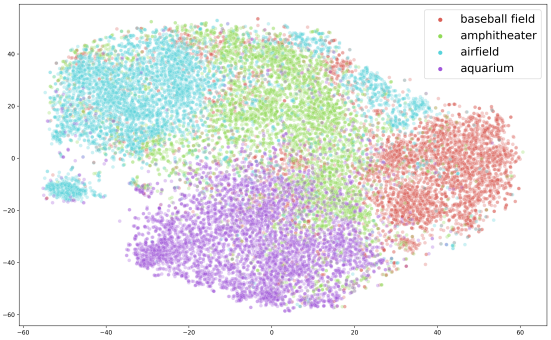

Figure 13: ViTMAE features

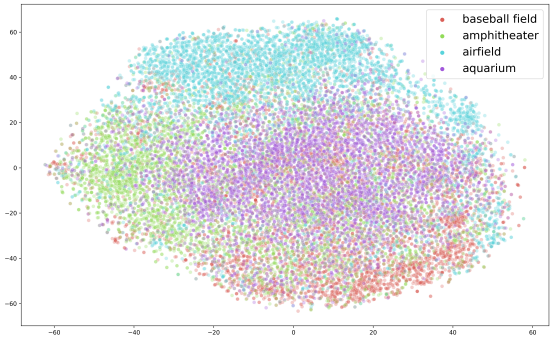

Figure 14: SimCLR features

Method	PCA	PCA+t-sne
ViTMAE [15]	**84.7**%	**73.7**%
SimCLR [6]	64.2%	49.9%

Table 3: Performances on Places classification

Since t-sne needs to be fitted on an entire dataset, we also try to evaluate parametric methods that can embed descriptors to a two dimensional plan while conserving well the descriptors structure. The parametric methods were trained on Imagenet. We evaluate the two-dimensional new descriptors on weather scene classification. Results are displayed in table 4. We used gaussian and student variational auto-encoders with same size and trained them on Imagenet descriptors. Evaluation results on scene classification seem to show that student method suits better for dimension reduction.

Conclusion

This study showed that that transformers based MAEs can facilitate supervised learning. It also showed that early attention layers have more local properties than others ones and that MAEs

Method	Top 1. acc (knn, k=20)
PCA+t-sne (fitted on the same data)	52.3%
Student VAE [21]	48.7%
Gaussian VAE	41.2%
Parametric t-sne [9]	46.6%

Table 4: Performances on weather scene classification

develop consistent attention patterns. We also show that it is possible to modelize the vision transformers descriptors as random variables following a student's law, which allows us to obtain robust representations even in very low dimension and at the scene scale.

References

[1] J. L. Ba, J. R. Kiros, and G. E. Hinton, *Layer normalization*, 2016.

[2] H. Bao, L. Dong, and F. Wei, *Beit: Bert pre-training of image transformers*, 2021.

[3] M. Caron, H. Touvron, I. Misra, H. Jégou, J. Mairal, P. Bojanowski, and A. Joulin, *Emerging properties in self-supervised vision transformers*, 2021.

[4] J. Chen, M. Hu, B. Li, and M. Elhoseiny, *Efficient self-supervised vision pretraining with local masked reconstruction*, 2022.

[5] M. Chen, A. Radford, R. Child, J. Wu, H. Jun, D. Luan, and I. Sutskever, *Generative pretraining from pixels*, in Proceedings of the 37th International Conference on Machine Learning, H. D. III and A. Singh, eds., vol. 119 of Proceedings of Machine Learning Research, PMLR, 13–18 Jul 2020, pp. 1691–1703.

[6] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, *A simple framework for contrastive learning of visual representations*, 2020.

[7] T. Chen, S. Kornblith, K. Swersky, M. Norouzi, and G. Hinton, *Big self-supervised models are strong semi-supervised learners*, 2020.

[8] X. Chen, S. Xie, and K. He, *An empirical study of training self-supervised vision transformers*, 2021.

[9] F. Crecchi, C. de Bodt, M. Verleysen, J. A. Lee, and D. Bacciu, *Perplexity-free parametric t-sne*, CoRR, abs/2010.01359 (2020).

[10] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, *Bert: Pre-training of deep bidirectional transformers for language understanding*, 2018.

[11] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, *An image is worth 16x16 words: Transformers for image recognition at scale*, CoRR, abs/2010.11929 (2020).

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

[13] X. Geng, H. Liu, L. Lee, D. Schuurmans, S. Levine, and P. Abbeel, *Multimodal masked autoencoders learn transferable representations*, 2022.

[14] R. Girdhar, A. El-Nouby, M. Singh, K. V. Alwala, A. Joulin, and I. Misra, *Omnimae: Single model masked pretraining on images and videos*, 2022.

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

[16] A. Newell and J. Deng, *How useful is self-supervised pretraining for visual tasks?*, 2020.

[17] Y. Pang, W. Wang, F. E. H. Tay, W. Liu, Y. Tian, and L. Yuan, *Masked autoencoders for point cloud self-supervised learning*, 2022.

[18] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever, *Learning transferable visual models from natural language supervision*, 2021.

[19] M. Raghu, T. Unterthiner, S. Kornblith, C. Zhang, and A. Dosovitskiy, *Do vision transformers see like convolutional neural networks?*, in Advances in Neural Information Processing Systems, A. Beygelzimer, Y. Dauphin, P. Liang, and J. W. Vaughan, eds., 2021.

[20] R. Strudel, R. Garcia, I. Laptev, and C. Schmid, *Segmenter: Transformer for semantic segmentation*, 2021.

[21] H. Takahashi, T. Iwata, Y. Yamanaka, M. Yamada, and S. Yagi, *Student-t variational autoencoder for robust density estimation*, in Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI-18, International Joint Conferences on Artificial Intelligence Organization, 7 2018, pp. 2696–2702.

[22] Z. Tong, Y. Song, J. Wang, and L. Wang, *Videomae: Masked autoencoders are data-efficient learners for self-supervised video pre-training*, 2022.

[23] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, *Attention is all you need*, 2017.

[24] Z. Yang, J. Wang, Y. Tang, K. Chen, H. Zhao, and P. H. S. Torr, *Lavt: Language-aware vision transformer for referring image segmentation*, 2021.

[25] J. Yu, Z. Wang, V. Vasudevan, L. Yeung, M. Seyedhosseini, and Y. Wu, *Coca: Contrastive captioners are image-text foundation models*, 2022.