vit! Architectures with Registers (Thesis outline)

Florian Weidner
Philipps-University Marburg, Germany
Department of Mathematics and Computer Science, Deep Learning Group
February 09, 2024

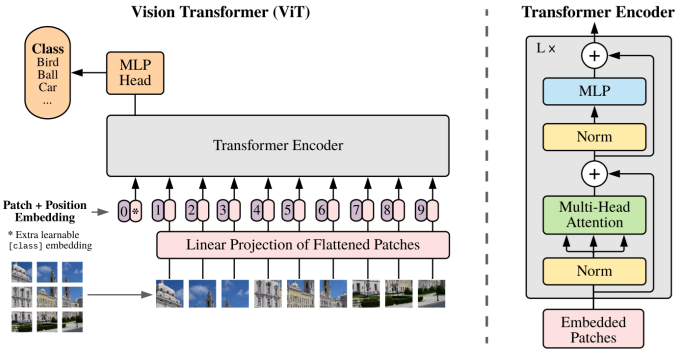


Fig. 1. Overview of a **vit!** architecture. [3]

***Abstract*—The abstract goes here.**
***Index Terms*—vit! (vit!),**

I. Introduction

Introduction to the topic...
Explanation of **vit!**s [5] [3] [**ruan2022visiontransformersstateart**] [6]

II. Vision Transformers

The Transformer architecture is a neural network model architecture, created primarily for sequence-to-sequence tasks in **nlp!** (**nlp!**). It consists of an encoder, which makes the input sequence into a continuos representation and a decoder, which then generates the output sequence. The encoder is built up of n identical layers, containing following components:

- multi-head self-attention mechanism: captures relationships between all tokens in the input, regardless of their distance
- feed-forward network: simple two-layer MLP network with ReLU activation which is applied to each token separately
- add & norm layers using residual connections and layer normalization to stabilize the training

The result outcome of the encoder is a enriched sequence representation, which is then used by the decoder to generate the output sequence. The decoder also consits of n identical layers with:

- masked multi-head self-attention mechanism: ensures a causal generation, by preventing that tokens have impact to future tokens
- encoder-decoder attention mechanism: focuses on the relevant parts of the encoder's output
- feed-forward network: similar to the encoder
- add & norm layer:ssimilar to the encoder

The input text is embedded and combined with a positional encoding to provide token order information. Because several attention layers can run in parallel, the architecture is significantly more parappelizable than **rnn!** (**rnn!**) or **cnn!** (**cnn!**) architectures, which makes it very efficient for modern hardware accelerators. That allows the Transformer to scale to very large models and datasets. [11]

Dosovitskiy et al. introduced the idea of using the stated transformer architecture for computer vision. A lot of research tried to combine self-attention mechanisms with **cnn!** architectures, not achieving a effectively scalable method for modern hardware accelerators. [3] proposed to apply a standard Transformer directly to images, that are split into fixed-size patches. Each patch is flattened into a vector and passed through a linear projection layer to form an embedding as input for the Transformer. These embeddings are used as tokens in a **nlp!** scenario. Positional embeddings are added to retain spatial information since they process images as sequences, unlike **cnn!**s which inherently capture spatial hierarchies. For classification tasks, an extra learnable [class] embeding is added in front of the embedded input. At the output of the encoder, the final representation of this token is used for classification. **vit!**s have much less image-specific inductive bias than **cnn!**s, because other than **cnn!**s, with the global self-attention mechanism spatial relationships needs to learned from scratch, but long-range dependencies across the entire image can be captured. As Transformers, **vit!**s are normally pre-trained on large datasets and then fine-tuned to more specific tasks. After pre-training, the prediction head is removed and a zero-initialized feedforeward layer,where the size is the number of classes, is added. Like Transformers, **vit!**s are also very parallelizable, which makes them very efficient. But [3] found out that without large-scale pre-training, **vit!**s often underperform. So **vit!**s requires sidnificant computational resources. But when pre-trained on large datasets, **vit!**s outperforms **cnn!**s on image classification tasks. The architecture performs well for transfer learning, where the pre-trained model can be fine-tuned already with limited labeled data. [3] [3] stated that further scaling of ViT would likely lead to improved performance. Alo self-supervised pre-training can be improved. They found out that with mimicking the masked

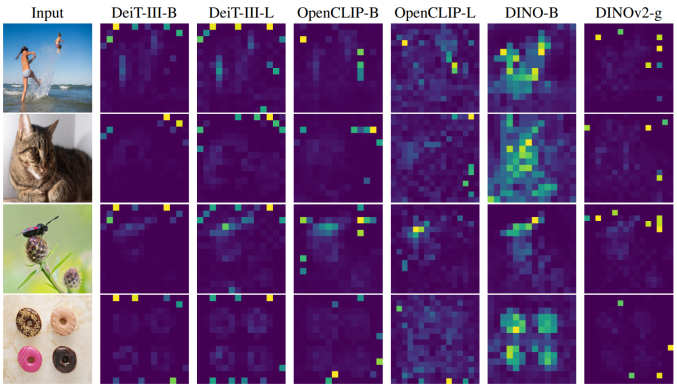


Fig. 2. Illustration of artifacts observed in the attention maps of modern vision transformers. [2]

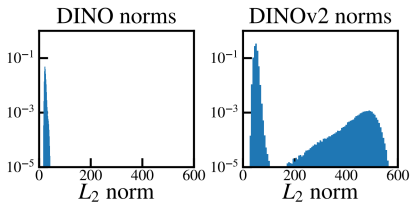
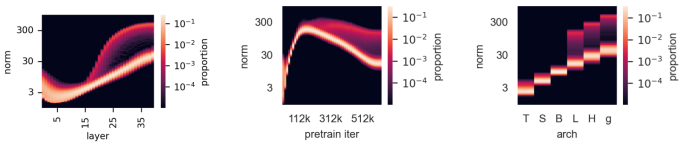


Fig. 3. Comparison of local feature norms for DINO ViT-B/16 and DINOv2 [2]



(a) Norms along layers. (b) Norms along iterations. (c) Norms across model size.

Fig. 4. Illustration of several properties of outlier tokens in the 40-layer DINOv2 ViT-g model [2]

language modeling task used in BERT, the model performs still better than **cnn!**s but a bit worse that with supervised pre-training. By now different architectures and training-tricks of **vit!**s have been proposed to further improve **vit!**s including self-supervised learning.

The architecture got adapted for image recognition, object detection, image segmentation, pose estimation, and 3D reconstruction tasks. [8]

III. Vision Transformers need registers: A summary

Summerization and explanation of the paper [**darcet2024visiontransformersneedregisters**] [12]

IV. Comparison to other papers with performance improvements of **vit!**s

I this chapter we summerize the paper [2]. The paper discoverd artifacts and proposes to use additional register tokens for **vit!**s to remove these artifacts. After introducing to **vit!**s like we did in this paper, the models they found the artifacts are introduced. The DINO algorithm is a self-supervised learning method, that uses two **vit!**s. A student network is predicting the output of a teacher network, to learn rich representaions of visual data without the need of manual annotations. [1] DINO is shown to produce models, that contain semantically consistent information in the last attention layer. Object discovery algorihtms like LOST, built on top of DINO, are using these attention maps, that often contains semantically interpretable information, used to detect objects without supervision. DINOv2 [7] is a improved followup focusing on dense predition tasks, which are tasks, where detailed outputs are required to provide fine-grained localized informations, like semantic segementation or depth estimation. Despite good performance on these dense tasks, the authors observed that DINOv2 is incompatible with LOST [2]. The different behaviour of DINO and DINOv2 can be observed in the artifacts in the last attention maps. In figure **??** you can see the different models and their artifacts on the last attention layer. While DINO shows no peak outlier values focusing the main object in the image, DINOv2 shows a lot of artifacts on the background of the images. This qualitatively observation can be also made for the label-supervised model DeiT-III and the text-supervised model OpenCLIP. Shown in figure **??**, you can observe similar artifacts in the background. To explain why and where the artifacts of **vit!**s in attention maps appear, the paper focuses on DINOv2. Artifact patches show higher norm of their token embedding at the output of the model than other patches. In figure **??** you can see the distribution of the local feature norms over a small dataset. While for DINO, the norm stays under 100 for all patches, DINOv2 shows a lot of patches with a norm higher than 150. This cutoff value can vary across different models. They define artifacts as

> ”tokens with norm higher than 150 will be considered as “high-norm” tokens” [2]

The authors found different conditions, when the artifacts appear in the training process of DINOv2. Figure **??** shows the following conditions:

- artifacts start appearing around layer 15 to 40.
- artifacts start appearing after on thrid of training.
- artifacts only appear in the three largest model versions

Another discovery is that the high-norm tokens appear where patch information is redundant. The authors tested the cosine similarity between high-nomr tokens and their four neighbors, directly after the image is emebdded. They observed, that the high norm patches appear where their cosine similarity to the neighbors is high. Compared to the observations, that shows that artifacts appear mostly in the background of images, high-norm pathes seem to have redundant information, that the model can ignore, to achive similar scores at the output.

survey: [4]

I thought of comparing the previous explained paper [**darcet2024visiontransformersneedregisters**] with papers like [13] [14] [10] [9] [6] which try to improve efficiency of **vit!**s

V. Conclusion

The conclusion goes here.

References

[1] Mathilde Caron et al. *Emerging Properties in Self-Supervised Vision Transformers*. 2021. arXiv: 2104.14294 [cs.CV]. URL: https://arxiv.org/abs/2104.14294.

[2] Timothée Darcet et al. *Vision Transformers Need Registers*. 2024. arXiv: 2309.16588 [cs.CV]. URL: https://arxiv.org/abs/2309.16588.

[3] Alexey Dosovitskiy et al. *An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale*. 2021. arXiv: 2010.11929 [cs.CV]. URL: https://arxiv.org/abs/2010.11929.

[4] Quentin Fournier, Gaétan Marceau Caron, and Daniel Aloise. "A Practical Survey on Faster and Lighter Transformers". In: *ACM Comput. Surv.* 55.14s (July 2023). ISSN: 0360-0300. DOI: 10.1145/3586074. URL: https://doi.org/10.1145/3586074.

[5] Salman Khan et al. "Transformers in Vision: A Survey". In: *ACM Comput. Surv.* 54.10s (Sept. 2022). ISSN: 0360-0300. DOI: 10.1145/3505244. URL: https://doi.org/10.1145/3505244.

[6] Yun Liu et al. "Vision transformers with hierarchical attention". en. In: *Mach. Intell. Res.* 21.4 (2024), pp. 670–683.

[7] Maxime Oquab et al. *DINOv2: Learning Robust Visual Features without Supervision*. 2024. arXiv: 2304.07193 [cs.CV]. URL: https://arxiv.org/abs/2304.07193.

[8] Bo-Kai Ruan, Hong-Han Shuai, and Wen-Huang Cheng. *Vision Transformers: State of the Art and Research Challenges*. 2022. arXiv: 2207.03041 [cs.CV]. URL: https://arxiv.org/abs/2207.03041.

[9] Michael S. Ryoo et al. *TokenLearner: What Can 8 Learned Tokens Do for Images and Videos?* 2022. arXiv: 2106.11297 [cs.CV]. URL: https://arxiv.org/abs/2106.11297.

[10] Hugo Touvron, Matthieu Cord, and Hervé Jégou. "DeiT III: Revenge of the ViT". In: *Computer Vision – ECCV 2022*. Ed. by Shai Avidan et al. Cham: Springer Nature Switzerland, 2022, pp. 516–533. ISBN: 978-3-031-20053-3.

[11] Ashish Vaswani et al. *Attention Is All You Need*. 2023. arXiv: 1706.03762 [cs.CL]. URL: https://arxiv.org/abs/1706.03762.

[12] Feng Wang et al. *Mamba-R: Vision Mamba ALSO Needs Registers*. 2024. arXiv: 2405.14858 [cs.CV]. URL: https://arxiv.org/abs/2405.14858.

[13] Yuxin Wen et al. *Efficient Vision-Language Models by Summarizing Visual Tokens into Compact Registers*. 2024. arXiv: 2410.14072 [cs.CV]. URL: https://arxiv.org/abs/2410.14072.

[14] Hongxu Yin et al. "A-ViT: Adaptive Tokens for Efficient Vision Transformer". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2022, pp. 10809–10818.