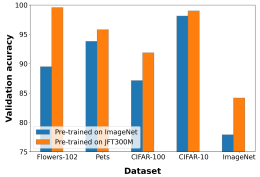
Where are my Neighbors?
Exploiting Patches Relations in Self-Supervised Vision Transformer

Guglielmo Camporese, Elena Izzo, Lamberto Ballan
University of Padova, Italy

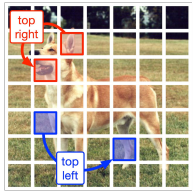
BMVC 2022
Paper: http://arxiv.org/abs/2206.00481
Code: https://github.com/guglielmocamporese/relvit

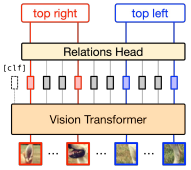
Training Set Size Problem



Vision Transformers need a lot of data in order to shine and their performance is strictly linked to the training set size used during pre-training.
For this reason, if trained from scratch, Vision Transformers are less accurate on relatively small datasets than convolutional neural networks.

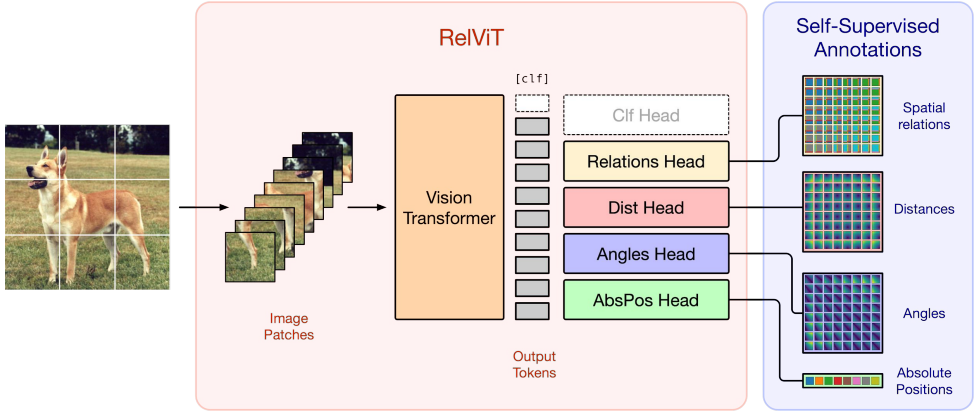
Contributions





- We propose and investigate various self-supervised tasks based on image patches;
- Our method, optimizing all input tokens naturally used by ViT and not only the classification one, is beneficial under different settings and for various self-attention-based models;
- We obtain very competitive results, outperforming supervised baselines and similar state-of-the-art models.

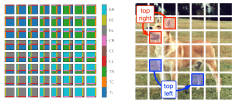
Model Architecture



Our Self-supervised Tasks

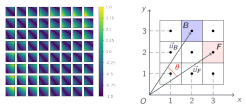
Spatial Relations

To learn the spatial relations among each couple of input image patches



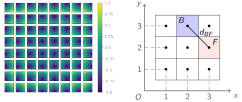
Angles

To learn a measure of angle among the location of each couple of input image patches



Distances

To learn a measure of distance among the location of each couple of input image patches



Absolute Positions

To recognize the 2D locations of each input image patch

Relational Vision Transformer (RelViT)

Backbone

The input image patches are processed by a standard Vision Transformer backbone which provides a set of output tokens.

Self-supervised heads

The backbone is equipped with specialized heads designed for solving the different tasks. Each head takes the output tokens of the transformer encoder as input and processes them thanks a MSA layer generating a relation for each couple of patches. Each obtained relation is used to compute a loss function for the task under investigation.

Learning process

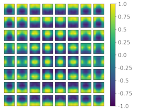
During training, the sum of all the losses of the tasks we want to solve is minimized. With this formulation, for each training step, all the combinations of couples of patches and all input tokens are optimized at the same time in parallel.

Experimental Results

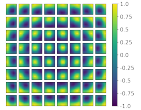
Pre-training plus fine-tuning Results

	Backbone	Supervised	RelViT	Improv.
CIFAR-10	ViT-S/4	86.09 ±0.46	**90.23** ±0.09	+4.14 ↑
SVHN	ViT-S/4	96.01 ±0.07	**97.14** ±0.03	+1.13 ↑
CIFAR-100	ViT-S/4	59.19 ±0.84	**64.99** ±0.46	+5.85 ↑
Flower-102	ViT-S/32	42.08 ±0.29	**45.78** ±0.75	+3.70 ↑
TinyImagenet	ViT-S/8	43.19 ±0.78	**51.98** ±0.20	+8.79 ↑
Imagenet100	ViT-S/32	58.04 ±0.91	**66.46** ±0.45	+8.42 ↑

Accuracy on several small datasets
For more details see the paper!



Positional embeddings learned by ViT



Positional embeddings learned by RelViT

Downstream-only Results

Backbone	Method	CIFAR-10	SVHN	CIFAR-100	Flower-102
ViT [11]		ViT-S/4	ViT-S/4	ViT-S/4	ViT-S/32
	Supervised	85.86 ±0.37	95.94 ±0.05	59.51 ±0.99	41.74 ±0.97
	RelViT (ours)	**89.27** ±0.23	**96.41** ±0.39	**61.92** ±0.19	**45.78** ±0.75
	(Improv.)	(+3.41 ↑)	(+0.47 ↑)	(+2.41 ↑)	(+4.04 ↑)
Swin [22]	Supervised [21]	59.47	71.6	53.28	34.51
	Swin+$\mathcal{L}_{drloc}$ [21]	83.89	94.23	66.23	39.37
	RelSwin (ours)	**92.33** ±0.22	**96.64** ±0.07	**68.69** ±0.91	**58.47** ±0.71
	(Improv.)	(+8.44 ↑)	(+2.41 ↑)	(+2.46 ↑)	(+19.10 ↑)
T2T-ViT [38]	Supervised [21]	84.19	95.36	65.16	31.73
	T2T-ViT+$\mathcal{L}_{drloc}$ [21]	87.56	96.49	**68.03**	34.35
	RelT2T-ViT (ours)	**90.82** ±0.16	**96.52** ±0.05	66.27 ±0.88	**50.53** ±1.45
	(Improv.)	(+3.26 ↑)	(+0.03 ↑)	(-1.76 ↓)	(+16.18 ↑)

Accuracy on various self-attention based models
For more details see the paper!

[11] A. Dosovitskiy et al., An image is worth 16x16 words..., *ICLR*, 2021.
[21] Y. Liu et al., Efficient training of visual transformers..., *NeurIPS*, 2021.
[22] Z. Liu et al., Swin transformer..., in *ICCV*, 2021.
[38] L. Yuan et al., Tokens-to-token vit..., in *ICCV*, 2021.