Patch ViT

Mengxue

Improve Vision Transformers Training by Suppressing Over-smoothing

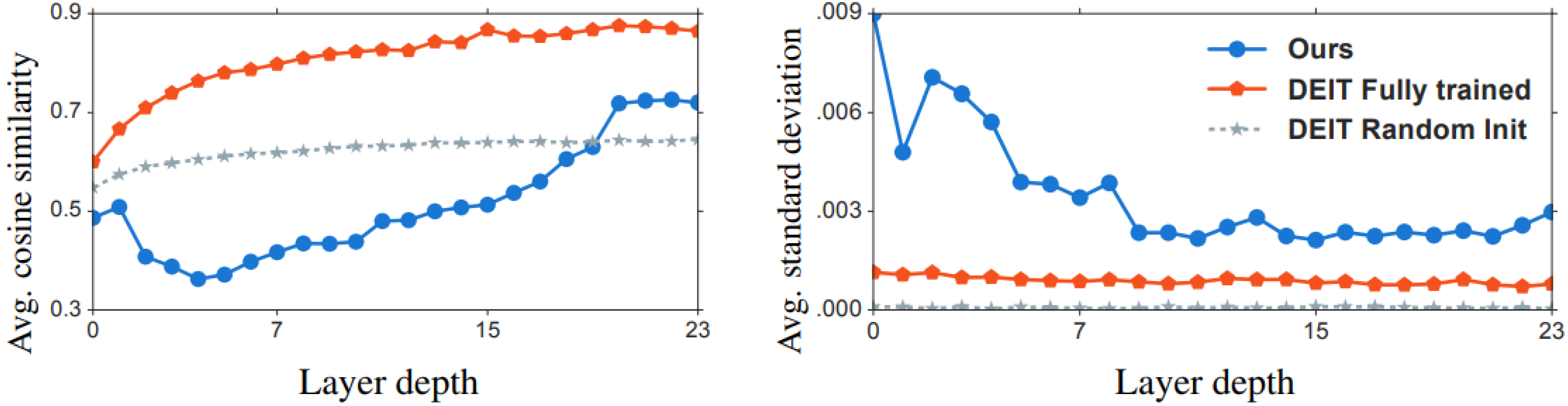
Chengyue Gong[1], Dilin Wang[2], Meng Li[2], Vikas Chandra[2], Qiang Liu[1]
[1] University of Texas at Austin [2] Facebook
{cygong, lqiang}@cs.utexas.edu, {wdilin, meng.li, vchandra}@fb.com

arxiv 2021.04

Motivation:

We observe that the **instability** of transformer training on vision tasks can be attributed to a **over-smoothing** problem, that the self-attention layers tend to map the different patches from the input image into a similar latent representation, hence yielding the loss of information and degeneration of performance, especially when the number of layers is large.

Contribution



(a) Layer-wise Cosine Similarity

(b) Layer-wise *s.t.d.* of Attention

- In this work, we first design extensive experiments to examine **the phenomenon of over-smoothing** in vision transformers across various architecture settings.
- We then investigate **three different strategies** to alleviate the over-smoothing problem in vision transformers.

Approach

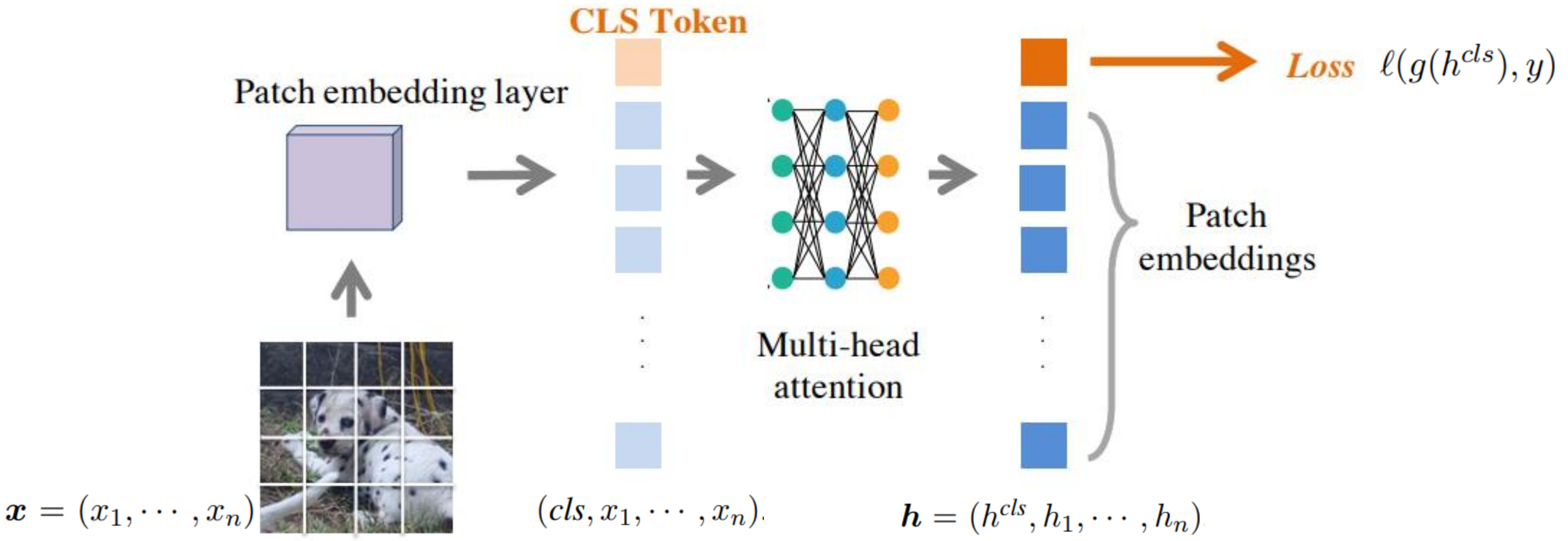


Figure 1: An overview of vision transformers by following (Dosovitskiy et al., 2020). Each image patch is first transformed to a latent representation using a convolutional patch embedding layer. The *dog* image is from ImageNet (Deng et al., 2009).

Examining Over-smoothness in Vision Transformers

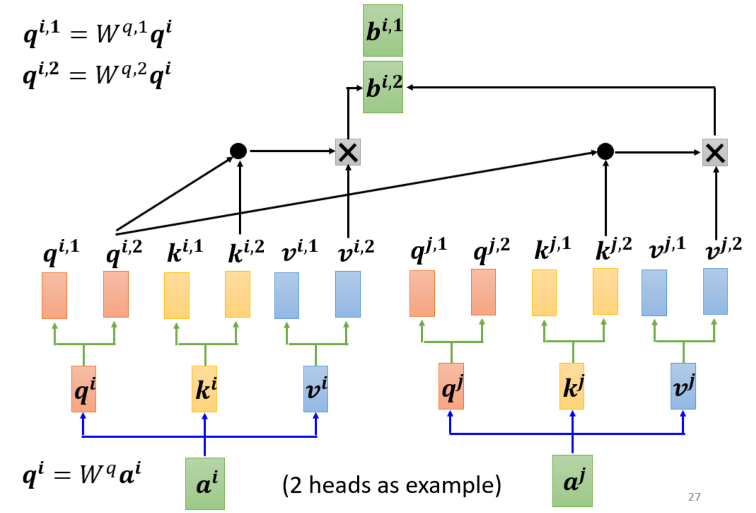
- Layer-wise cosine similarity between patch representations

$$\boldsymbol{h} = (h^{cls}, h_1, \cdots, h_n)\ (h_j \in \mathcal{R}^d),$$

$$\text{CosSim}(\boldsymbol{h}) = \frac{1}{n(n-1)} \sum_{i \neq j} \frac{h_i^\top h_j}{\| h_i \| \| h_j \|},$$

where $\| \cdot \|$ denotes the Euclidean norm.

- Layer-wise standard deviation of softmax attention scores



Multi-head Attention Self-attention is an attention mechanism relating different positions of a single sequence in order to compute a representation of the sequence, and is used as a module in the multi-head attention layers. Given an input representation matrix T, the multi-head self-attention first applies three different linear transformations on T and output K, Q, V. Then the multi-head attention is performed as follows,

$$\text{MutliHead}(K, Q, V) = \text{Concat}\Big(\text{head}_1, \cdots, \text{head}_M\Big) W^p, \quad \text{where}$$

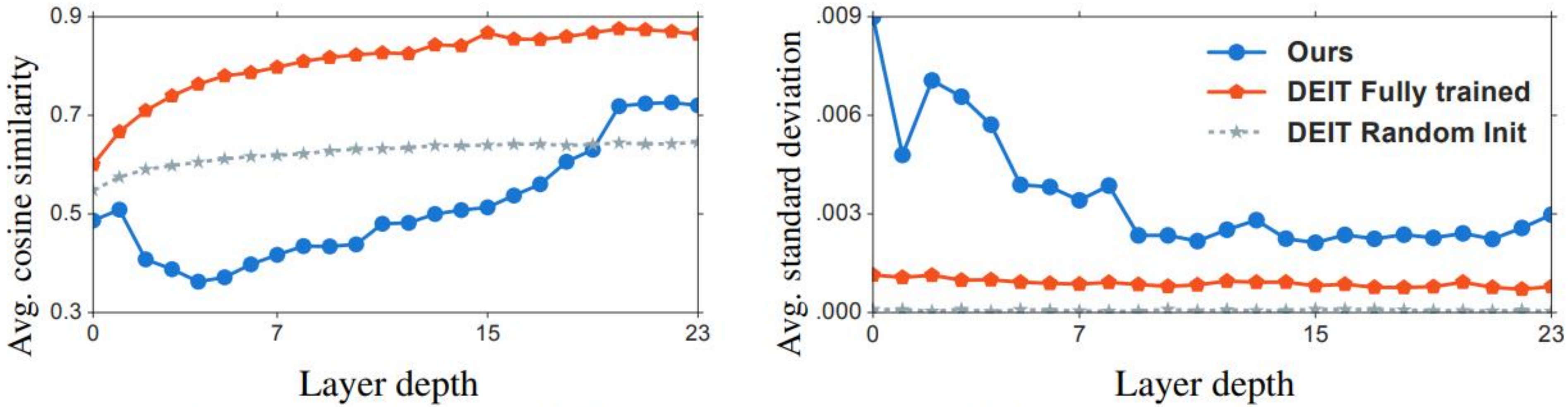
$$\text{head}_i = \text{Attention}(K_i, Q_i, V_i),$$

$$\text{Attention}(K_i, Q_i, V_i) = \text{Softmax}\left(\frac{Q_i K_i^T}{\sqrt{d_k}}\right) V_i,$$

where $K = [K_1, \cdots, K_M], Q = [V_1, \cdots, V_M], V = [V_1, \cdots, V_M]$ is split into M fragments evenly along the feature dimension, W^p denotes a linear projection layer and d_k denotes the feature dimension of K.

metric to measure the diversification of its attention patterns. Specifically, given a patch representation h_i in $\boldsymbol{h}$ and its Softmax attention score as $S(h_i)$ (see Eqn.(2)), with $S(h_i) \in \mathcal{R}^n$ and n the number of patches. We use the standard deviation of the Softmax attention score $\text{std}(S(h_i))$ to quantify the smoothness. For multi-head attention, we simply average the standard deviations over all different heads and patches. Small standard deviation values imply that each patch would attend all other patches with similar weights hence in turn leading to similar patch representations.

Examining Over-smoothness in Vision Transformers



(a) Layer-wise Cosine Similarity

(b) Layer-wise *s.t.d.* of Attention

Figure 2: An illustration of the over-smoothing phenomenon in vision transformers. We use a 24-layer DEIT-Base model as our testbed. ‘Ours’ and ‘DEIT random init’ denotes the metrics of the model trained by our proposed loss and a random initialized DEIT model, respectively. All metrics are computed on a sub-sampled ImageNet training set, which contains 10,000 images.

Suppressing Over-smoothing in Vision Transformers

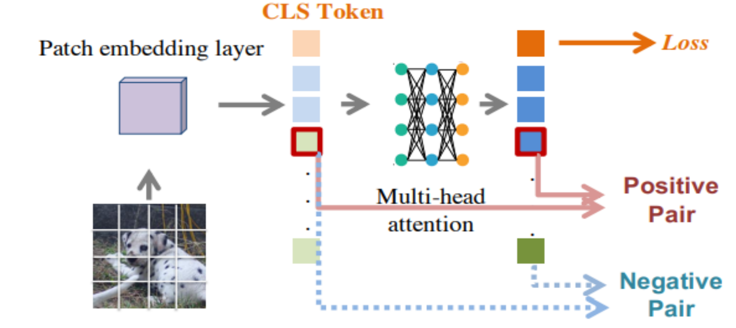
- **Pairwise Patch Cosine Similarity Regularization**

final-layer patch representation $\boldsymbol{h} = (h^{cls}, h_1, \cdots, h_n)$, we add a new loss $\ell_{cos} = \frac{1}{n(n-1)} \sum_{i \neq j} \frac{h_i^\top h_j}{\|h_i\| \|h_j\|}$.

- **Patch Contrastive Loss** (e is its patch representations at a early layer and h is its patch representations at a deep layer)

$$\ell_{cons} = -\frac{1}{n} \sum_{i=1}^{n} \log \frac{\exp(e_i^\top h_i)}{\exp(e_i^\top h_i) + \exp(e_i^\top (\sum_{j=1}^{n} h_i / n))},$$

In this work, we fix $\boldsymbol{e}$ and $\boldsymbol{h}$ to be the first layer features and last layer features, respectively. In practice, we stop the gradient on $\boldsymbol{e}$.



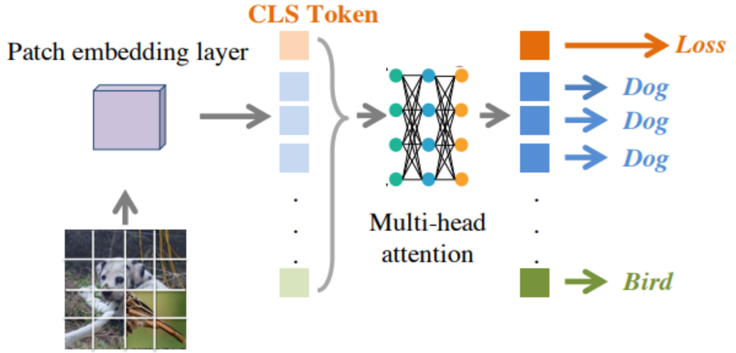
(a) Patch Constrastive Loss

- **Patch Mixing Loss** (cutmix)

This patch mixing loss could be formulated as follows,

$$\ell_{token} = \frac{1}{n} \sum_{i=1}^{n} \ell_{ce}(g(h_i), y_i),$$

where h_i represents patch emebddings in the last layer, g denotes the additional linear classification head, y_i is the class label and ℓ_{ce} denotes the cross entropy loss.



(b) Patch Mixing Loss

EXPERIMENTAL RESULTS

Cosine Reg	Patch Constrastive	Patch Mixing	Top-1 Acc (%)
×	×	×	81.8
✓	×	×	82.0
×	×	✓	**82.4**
×	✓	×	82.3
✓	×	✓	82.3
✓	✓	×	82.3
×	✓	✓	**82.6**

Table 1: Improved ImageNet accuracy using our anti-oversmootheness regularization strategies.

$$\ell_{ce} + \ell_{cutmix} + \ell_{cons}.$$

EXPERIMENTAL RESULTS

Method	Model Size	+ Teacher Models	+ Conv Layers	Top-1 Acc (%)
DEIT-S12 (Touvron et al., 2020)	22M	×	×	79.9
DEIT-S12 + Ours	22M	×	×	**81.2**
DEIT-S24	44M	×	×	79.6
DEIT-S24 + Ours	44M	×	×	**82.2**
DIET-B12	86M	×	×	81.8
DEIT-B12 + Ours	86M	×	×	**82.9**
DEIT-B24	172M	×	×	81.4
DEIT-B24 + Ours	172M	×	×	**83.3**
DIET-B12↑384	86M	×	×	83.1
DEIT-B12 + Ours ↑384	86M	×	×	**84.2**
DEIT-B24 + Ours ↑512	172M	×	×	**85.0**
CaiT-S36 (Touvron et al., 2021)	68M	×	×	83.3
CaiT-M36	271M	×	×	85.1
CaiT-M48↑448	356M	✓	×	**86.5**
SWIN-Base (Liu et al., 2021)	88M	×	×	83.3
SWIN-Base↑384	88M	×	×	84.2
CVT-21 (Wu et al., 2021)	32M	×	✓	82.5
CvT-21↑384	32M	×	✓	83.3
LV-ViT-M (Jiang et al., 2021)	56M	✓	✓	84.0
LV-ViT-L↑448	150M	✓	✓	86.2

Table 2: Compared to other recent methods for training transformers. Top-1 accuracy on ImageNet validation set is reported.

EXPERIMENTAL RESULTS

S	D	PatchConstrastive	PatchMixing	Talking-Head	Epoch	Top-1 Acc (%)
224	12	×	×	×	300	81.8
384	12	×	×	×	300	83.1
224	12	×	✓	×	300	82.4
224	12	✓	×	×	300	82.3
224	12	✓	✓	×	300	82.6
224	12	✓	✓	✓	300	82.7
224	12	✓	✓	✓	400	**82.9**
224	24	✓	✓	✓	400	**83.3**
384	12	✓	✓	✓	-	84.2
512	12	✓	✓	✓	-	84.5
512	24	✓	✓	✓	-	85.0

Table 3: Ablation study on DEIT-Base on ImageNet validation set. ‘S’ and ‘D’ denotes image size and depth, respectively.

Model	DEIT	Ours
Standard	81.7	82.9
- Repeat Augmentation	76.5	82.9
- Random Erasing	5.6	82.9
- Mixup	80.0	82.9
- Drop Path	3.4	80.4
+ Depth (24 Layer)	77.3	83.3

Table 4: Compared to DEIT training strategies (Touvron et al., 2020), our proposed losses make the training of transformers more robust. The results on the DEIT-Base model is reported.

Talking-Heads Attention

Noam Shazeer*
Google
noam@google.com

Zhenzhong Lan*
Google
lanzhzh@google.com

Youlong Cheng*
Google
ylc@google.com

Nan Ding*
Google
dingnan@google.com

Le Hou*
Google
lehou@google.com

EXPERIMENTAL RESULTS

PatchConstrastive	PatchMixing	Drop Path Rate	Top-1 Acc (%)
×	✓	0.10	81.8
×	✓	0.50	82.7
×	✓	0.75	82.5
✓	✓	0.10	82.0
✓	✓	0.50	83.0
✓	✓	0.75	**83.3**

Table 5: Ablation study on 24-layer DEIT-Base on ImageNet validation set. We demonstrate that by using the token constrastive loss, we are able to use stronger drop path and achieve better generalization. The image size is set to 224×224, while talking head attention is used. In our experiments, following (Touvron et al., 2020), we linearly increase the drop path rate by layer.

Model	Image Size	Top-1 Acc (%)
VIT-Large (Dosovitskiy et al., 2020)	384	85.1
VIT-Large + Ours	224	83.9
VIT-Large + Ours	384	85.3

Table 6: We download Dosovitskiy et al. (2020)'s checkpoint and finetune it with 40 epochs on ImageNet.

Thank you