IFACD: Intermediate Features Augmented Contrastive Distillation

Edwin Arkel Rios	edwinarkelrios.ee08@nycu.edu.tw	National Yang Ming Chiao Tung University, Taiwan
Lidia Pivovarova	lidia.pivovarova@helsinki.fi	University of Helsinki, Finland
Tarun Narayanan	tarunn2799@gmail.com	SpaceML, NASA Frontier Development Lab
Ajay Krishnan	krishajay.g@gmail.com	SpaceML, NASA Frontier Development Lab
Jaesung Tae	jake.tae@yale.edu	Yale University
Min-Chun Hu	anitahu@cs.nthu.edu.tw	National Tsing Hua University, Taiwan
Bo-Cheng Lai	bclai@nycu.edu.tw	National Yang Ming Chiao Tung University, Taiwan

Index

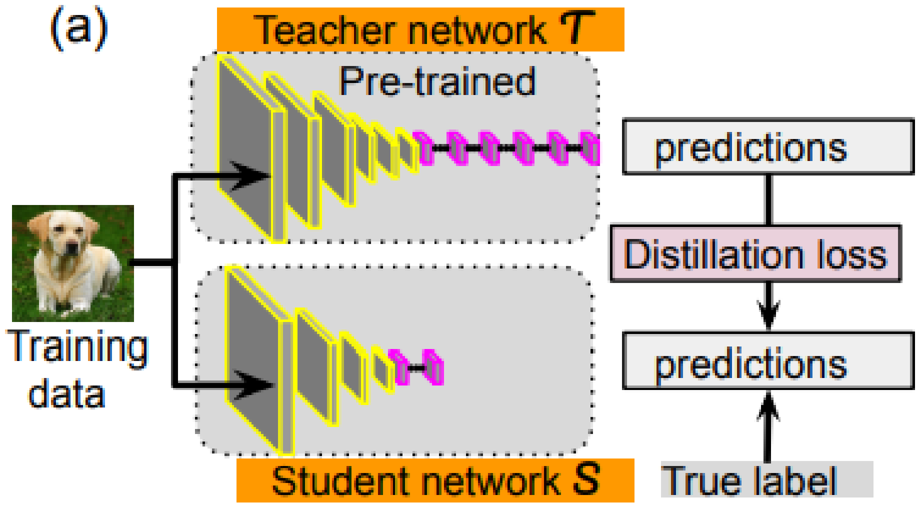
- Motivation and Related Work
- Methodology
- Results and Discussion
- Future Work
- Conclusion
- Appendix A: Intermediate Features Augmented Contrastive Learning of Representations

Motivation and Related Work

Knowledge Distillation

- Student-Teacher (S-T) learning framework for model compression (smaller model is trained to mimic larger one or ensemble of) and knowledge transfer
- First defined by Bucila et al. (2006) and popularized by Hinton et al. (2014)

$$q_i = \frac{exp(z_i/T)}{\sum_j exp(z_j/T)}$$



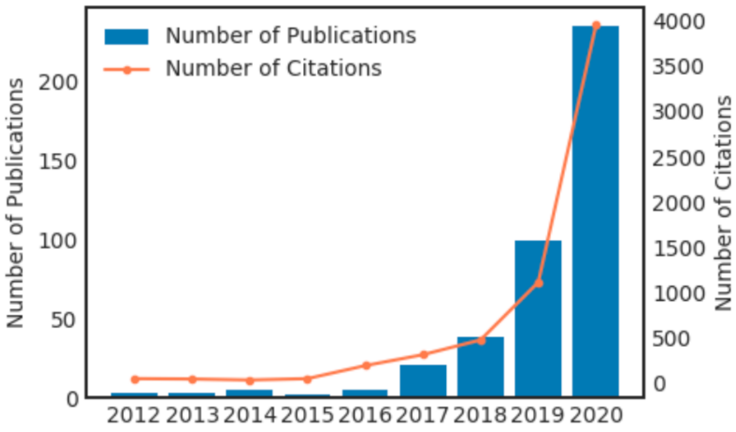
Model compression. Bucila et al. SIGKDD 2006.

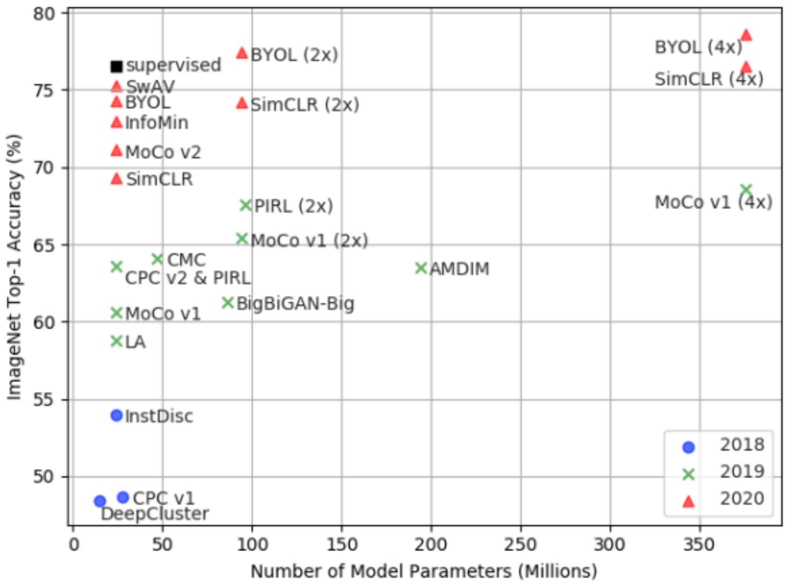
Distilling the knowledge in a neural network. Hinton et al. NIPS DL Workshop 2014.

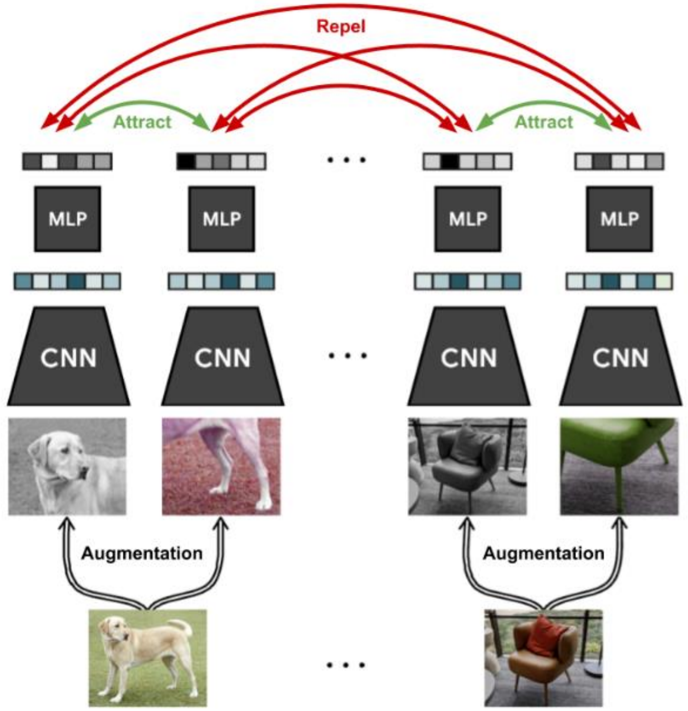
Knowledge Distillation and Student-Teacher Learning for Visual Intelligence: A Review and New Outlooks. Wang et al. IEEE TPAMI 2021.

SSL, CL, and SimCLR

- Self-supervised learning (SSL) allows us to exploit unlabeled data
- Contrastive learning (CL) of visual representations
  - Two different augmentations of a given image should have representations that are closer to each other than to any other image in a given batch
  - Minimize distance between positive pairs and maximize distance to negative ones





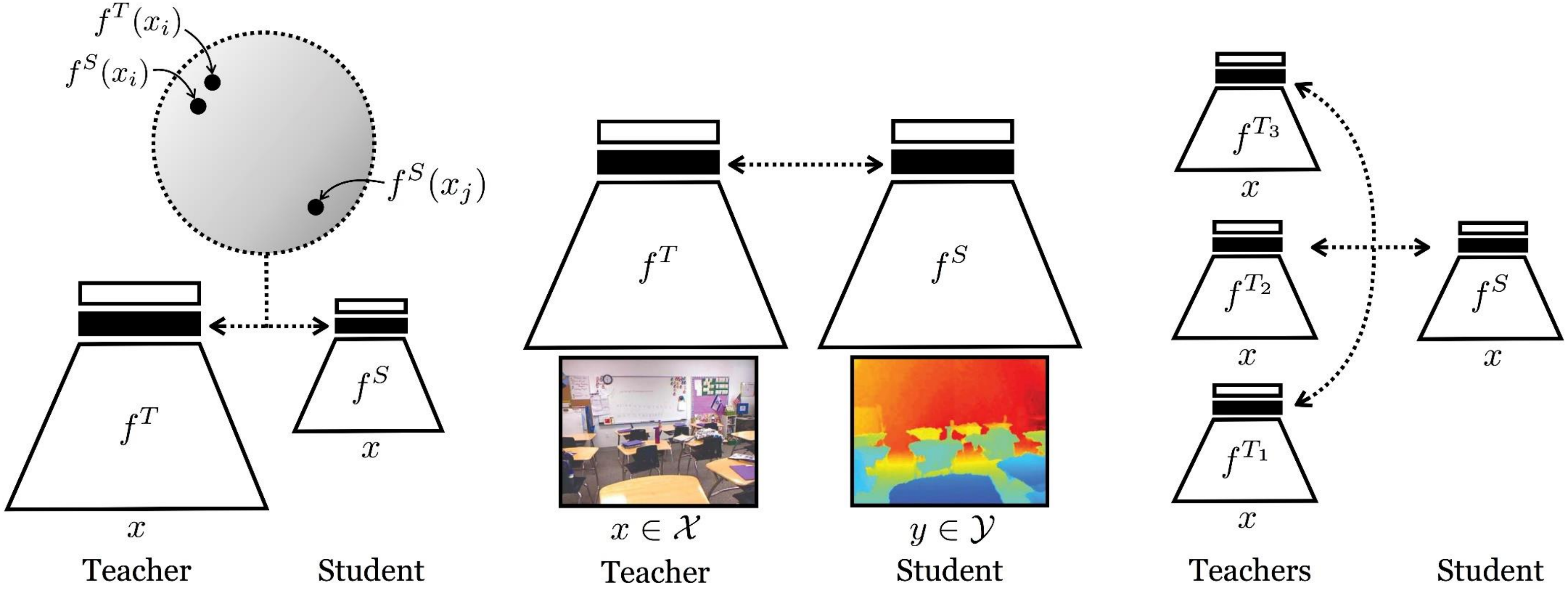


Self-supervised Learning: Generative or Contrastive. Liu et al. IEEE Transactions On Knowledge and Data Engineering 2020.

A Simple Framework for Contrastive Learning of Visual Representations. Chen et al. ICML 2020.

Contrastive Representation Distillation

- Student is trained by combination of CE classification objective and contrastive loss between teacher and student representations

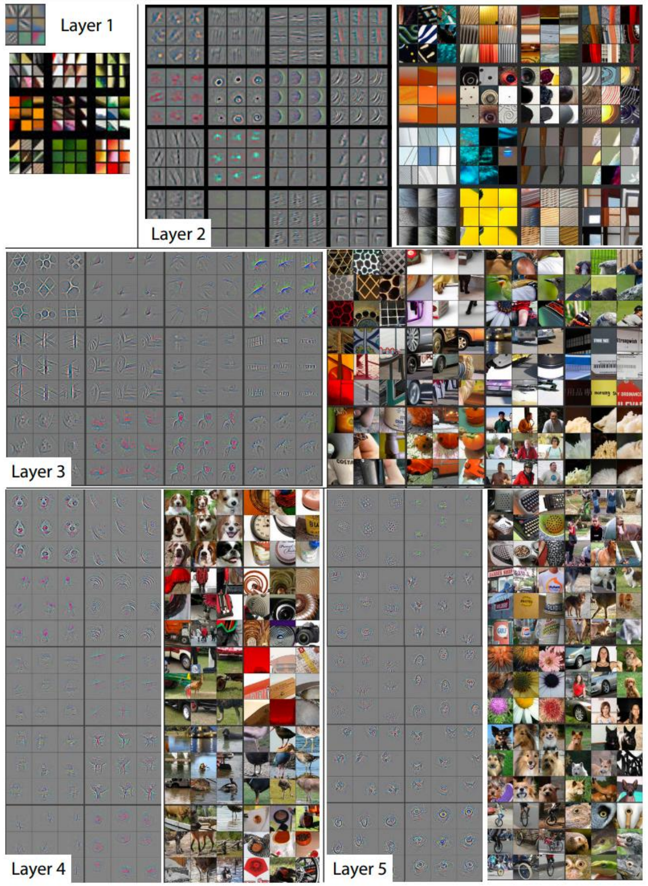


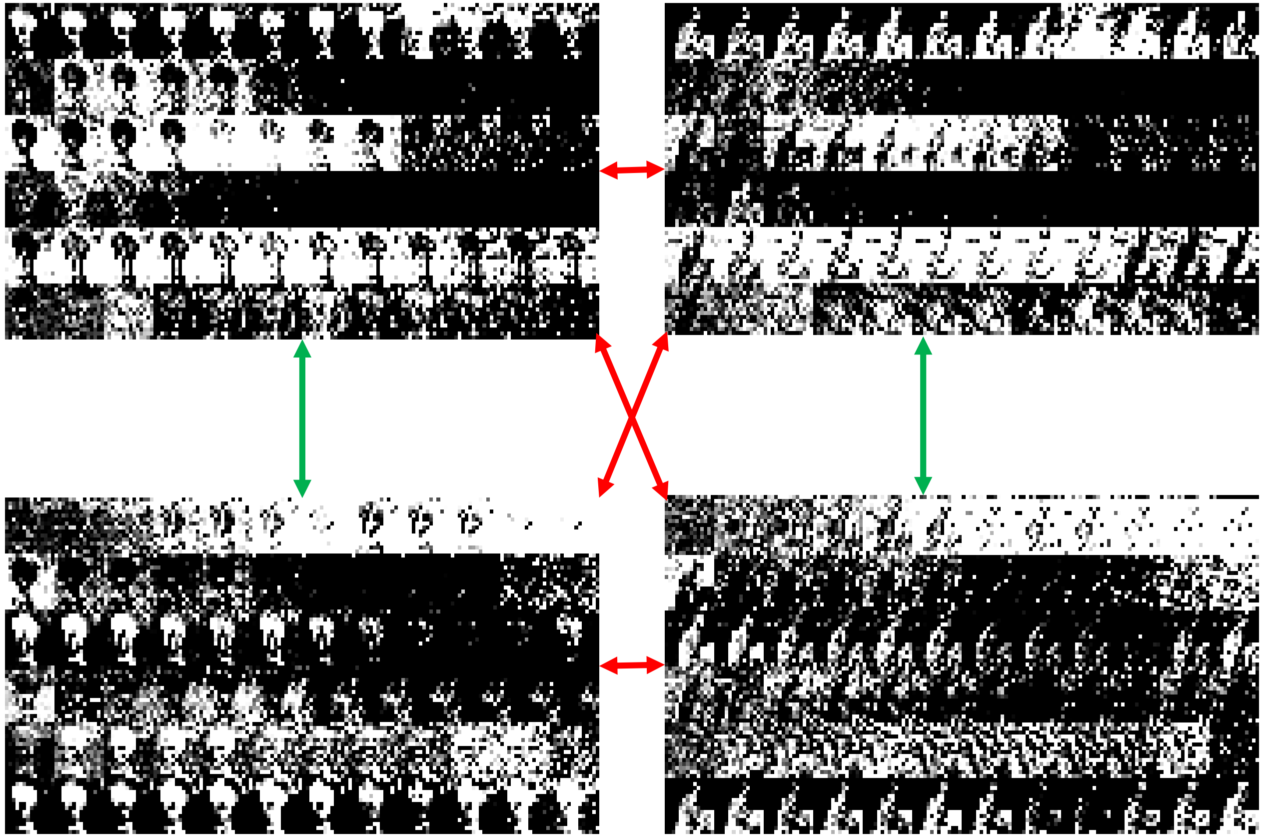
(a) Model compression (b) Cross-modal transfer (c) Ensemble distillation

Contrastive Representation Distillation. Tian et al. ICLR 2020.

Intermediate Features Augmentation

- Representation for a given image across different layers should be closer between each other than to any other image in same layer or in other layers



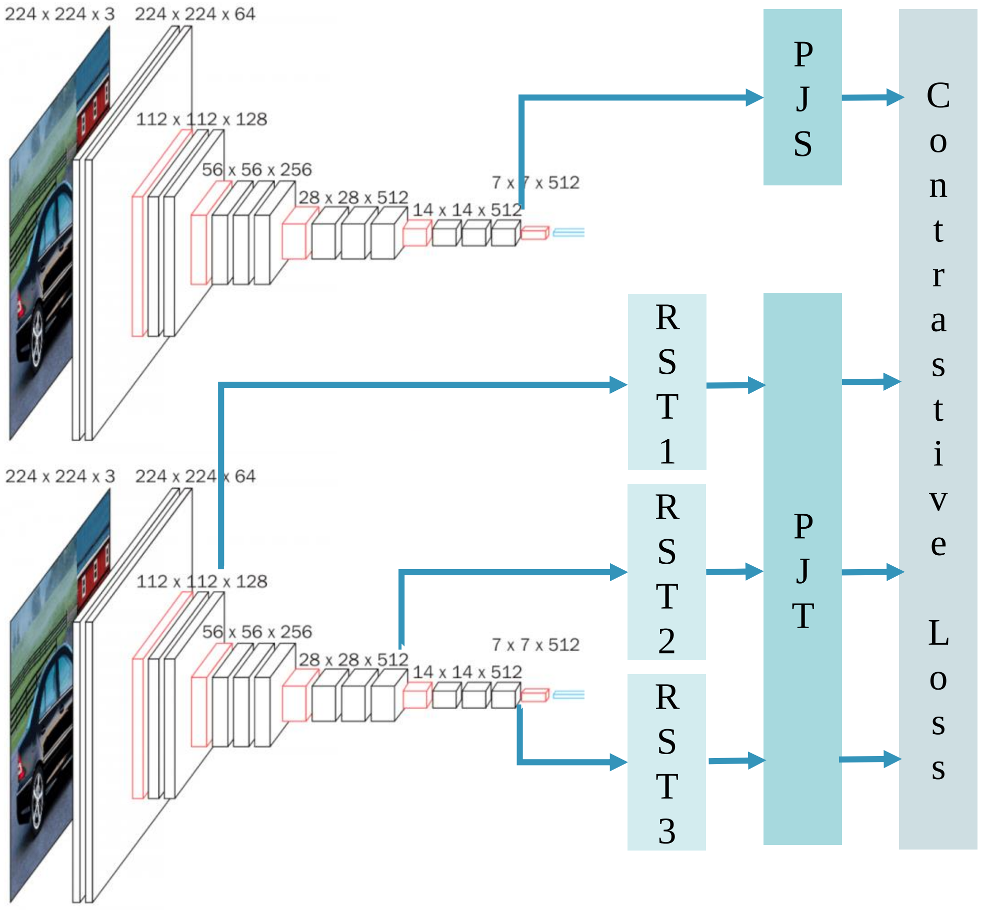


Visualizing and Understanding Convolutional Networks. Zeiler et al. ECCV 2014.

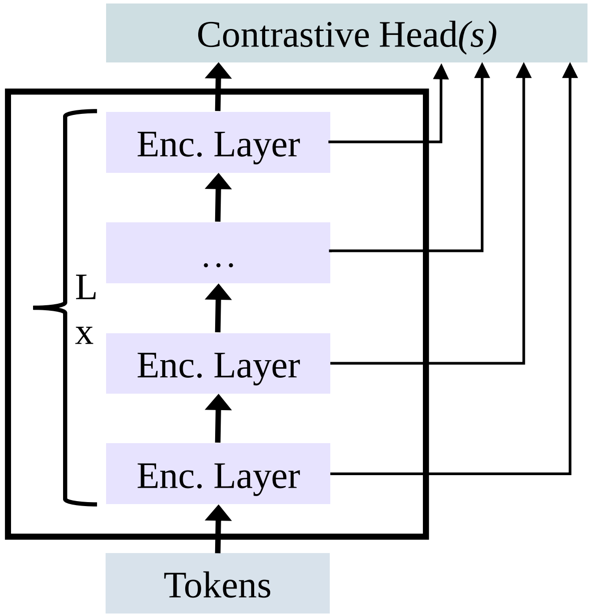
Methodology

Intermediate Features Augmented CD

- Intermediate features as extra views for contrastive loss with multiple positives and negatives pairs



$$L = \alpha L_{CE} + \beta L_{KLDiv} + \gamma L_{IFACD}$$



Contrastive Head:

1. *Rescaler MLP*
2. Projection MLP

MLP:

1. *Spatial pooling*
2. FC Layer
3. *LN/BN1d*
4. *GELU/ReLU*
5. *FC Layer*
6. *LN/BN1d*
7. *GELU/ReLU*
8. *FC Layer*
9. *LN/BN1d*

https://www.cs.toronto.edu/~frossard/post/vgg16/

Italics represent the component is optional

Experiments

- **CIFAR-100**: train on 50K **32x32 images** and report results of last epoch on 10K test averaged over runs
- **SGD**, Step LR scheduler (150, 180, 210), LR=0.05, WD=5e-4, 240 epochs
- **CIFAR-style ResNet, WideResNet, VGG**
  - resnet-d to represent CIFAR-style resnet with three groups of basic blocks, each with 16, 32 and 64 channels, respectively
  - wrn-d-w represents wide ResNet with depth d and width factor w
- $\alpha = 1$ (fully-supervised cross-entropy loss), $\beta = 1$ (KL divergence between teacher and student logits term), and their specific distillation loss terms γ

$$L = \alpha L_{CE} + \beta L_{KLDiv} + \gamma L_{Distill}$$

https://github.com/HobbitLong/RepDistiller

Results and Discussion

Accuracy Results Previous Work

- CRD overall gets the best results but not by much as overall average is 73.78% vs AT and PKT which get 73.62% and 73.63%, respectively

	wrn_40_2		vgg13	resnet56	resnet32x4	resnet110	Average
Method	wrn_16_2	wrn_40_1	vgg8	resnet20	resnet8x4	resnet20	
attention	75.31	74.43	73.51	71.20	75.76	**71.49**	73.62
correlation	75.40	74.33	72.98	71.04	75.78	71.20	73.45
crd	75.74	**74.74**	73.28	71.45	**76.13**	71.36	**73.78**
hint	75.09	73.88	**73.95**	70.32	75.05	70.28	73.09
kd	**75.77**	74.27	72.63	**71.53**	75.34	71.28	73.47
nst	75.70	74.27	72.87	71.40	75.52	71.43	73.53
pkt	75.70	74.45	73.23	71.50	75.60	71.41	73.65
rkd	75.26	74.06	73.06	71.18	75.45	70.78	73.30
similarity	75.60	74.30	73.50	71.44	75.90	71.03	73.63
vid	75.26	73.90	73.29	71.53	75.72	71.26	73.49
Student vanilla	71.12	72.89	70.16	69	72.21	69	70.73
Teacher vanilla	76.32	76.32	74.18	72.79	78.36	73.76	75.29

Green bold represents the best results in terms of top-1 classification accuracy

Accuracy Results Ours

- Improvement from using our method with multiple layers

	wrn_40_2		vgg13	resnet56	resnet32x4	resnet110	Average
Method	wrn_16_2	wrn_40_1	vgg8	resnet20	resnet8x4	resnet20	
attention	75.31	74.43	73.51	71.20	75.76	71.49	73.62
correlation	75.40	74.33	72.98	71.04	75.78	71.20	73.45
crd	75.74	**74.74**	73.28	71.45	**76.13**	71.36	73.78
hint	75.09	73.88	**73.95**	70.32	75.05	70.28	73.09
kd	75.77	74.27	72.63	71.53	75.34	71.28	73.47
nst	75.70	74.27	72.87	71.40	75.52	71.43	73.53
pkt	75.70	74.45	73.23	71.50	75.60	71.41	73.65
rkd	75.26	74.06	73.06	71.18	75.45	70.78	73.30
similarity	75.60	74.30	73.50	71.44	75.90	71.03	73.63
vid	75.26	73.90	73.29	71.53	75.72	71.26	73.49
ifacd1	75.61	74.25	73.23	71.53	75.69	71.26	73.36
ifacd2	**75.76**	74.39	73.23	**71.70**	75.78	**71.57**	**73.85**
Student vanilla	71.12	72.89	70.16	69	72.21	69	70.73
Teacher vanilla	76.32	76.32	74.18	72.79	78.36	73.76	75.29

Green bold represents the best results in terms of top-1 classification accuracy

Intermediate Features Ablations

- Improvement from using more layers but peaks at 2 layers

Method	wrn_40_2		vgg13	resnet56	resnet32x4	resnet110	Average
	wrn_16_2	wrn_40_1	vgg8	resnet20	resnet8x4	resnet20	
ifacd1	75.61	74.25	73.23	71.53	75.69	71.26	73.36
ifacd2	**75.76↑**	**74.39↑**	73.23	**71.70↑**	75.78↑	**71.57↑**	73.85↑
ifacd3	75.42↓	73.89↑	**73.45↑**	**71.70↑**	**75.87↑**	71.42↑	73.63↑

Green represents improvement in terms of classification accuracy compared to baseline with only last layer for contrastive loss

Red bold represents decrease in terms of classification accuracy compared to baseline with only last layer for contrastive loss

Bold represents the best results in terms of top-1 classification accuracy

Memory Requirements

- CRD consumes the most memory resources due to the memory bank
- IFACD with 1 layer consumes the least and with 2 layers still among the lowest

	wrn_40_2		vgg13	resnet56	resnet32x4	resnet110	Average
Method	wrn_16_2	wrn_40_1	vgg8	resnet20	resnet8x4	resnet20	
attention	0.43	0.43	**2.59**	0.23	1.52	0.23	0.43
correlation	0.41	0.43	**2.59**	0.23	1.52	0.23	0.41
crd	**1.34**	**1.38**	2.64	**1.28**	1.59	**1.28**	**1.34**
hint	0.42	0.43	2.59	0.23	1.52	0.23	0.42
ifacd1	**0.35**	**0.28**	2.72	**0.19**	**0.75**	**0.20**	0.35
ifacd2	0.42	0.41	2.60	0.21	1.14	0.21	0.42
kd	0.42	0.43	**2.59**	0.23	1.52	0.23	0.42
nst	0.95	0.52	**4.84**	0.28	**3.39**	0.29	0.95
pkt	0.41	0.43	**2.59**	0.23	1.52	0.23	0.41
rkd	0.41	0.43	**2.59**	0.23	1.52	0.23	0.41
similarity	0.41	0.43	**2.59**	0.23	1.52	0.23	0.41
vid	0.44	0.43	2.61	0.23	1.52	0.23	0.44

Green bold represents the results that consumes the less VRAM

Red bold represents the results that consumes the most VRAM

Number of Parameters

- As is expected IFACRD with multiple layers consumes the most parameters as for each intermediate layer it requires another rescaler MLP

	wrn_40_2		vgg13	resnet56	resnet32x4	resnet110	Average
Method	wrn_16_2	wrn_40_1	vgg8	resnet20	resnet8x4	resnet20	
attention	2.96	2.82	13.43	1.14	8.67	2.01	2.96
correlation	2.99	2.85	13.56	1.16	8.73	2.03	2.99
crd	3.02	2.87	13.69	1.17	8.80	2.05	3.02
hint	2.96	2.83	13.49	1.14	8.68	2.02	2.96
ifacd1	3.06	2.91	14.22	1.18	8.93	2.06	3.06
ifacd2	**3.09**	**2.94**	**14.75**	**1.19**	**9.07**	**2.07**	**3.09**
kd	2.96	2.82	13.43	1.14	8.67	2.01	2.96
nst	2.96	2.82	13.43	1.14	8.67	2.01	2.96
pkt	2.96	2.82	13.43	1.14	8.67	2.01	2.96
rkd	2.96	2.82	13.43	1.14	8.67	2.01	2.96
similarity	2.96	2.82	13.43	1.14	8.67	2.01	2.96
vid	3.02	2.88	15.25	1.16	8.93	2.03	3.02

Red bold represents the method with largest number of parameters

Training Time

- NST and CRD take the largest amount of time while KD and Hint take the least*

	wrn_40_2		vgg13	resnet56	resnet32x4	resnet110	Average
Method	wrn_16_2	wrn_40_1	vgg8	resnet20	resnet8x4	resnet20	
attention	14.35	24.43	11.77	17.75	21.59	20.12	18.33
correlation	15.52	22.61	13.29	17.31	20.79	19.52	18.17
crd	28.47	**36.00**	27.76	**31.01**	32.95	**28.87**	30.84
hint	**13.48**	22.35	11.00	**17.40**	20.97	18.79	**17.33**
ifacd1	16.05	21.09	12.62	16.73	17.47	23.11	17.72
ifacd2	15.08	24.65	12.81	20.70	22.35	20.91	19.19
kd	20.07	**22.01**	**9.72**	17.47	**17.10**	21.15	17.92
nst	**37.31**	31.38	**162.23**	20.96	**117.77**	21.58	**65.21**
pkt	14.89	23.96	17.57	17.22	20.85	18.95	18.91
rkd	15.91	24.11	18.82	18.64	22.15	20.11	19.96
similarity	15.00	22.54	13.42	17.05	20.68	18.93	17.94
vid	17.46	25.75	17.24	19.12	25.56	21.04	21.03

Green bold represents the method that takes the least amount of time to train

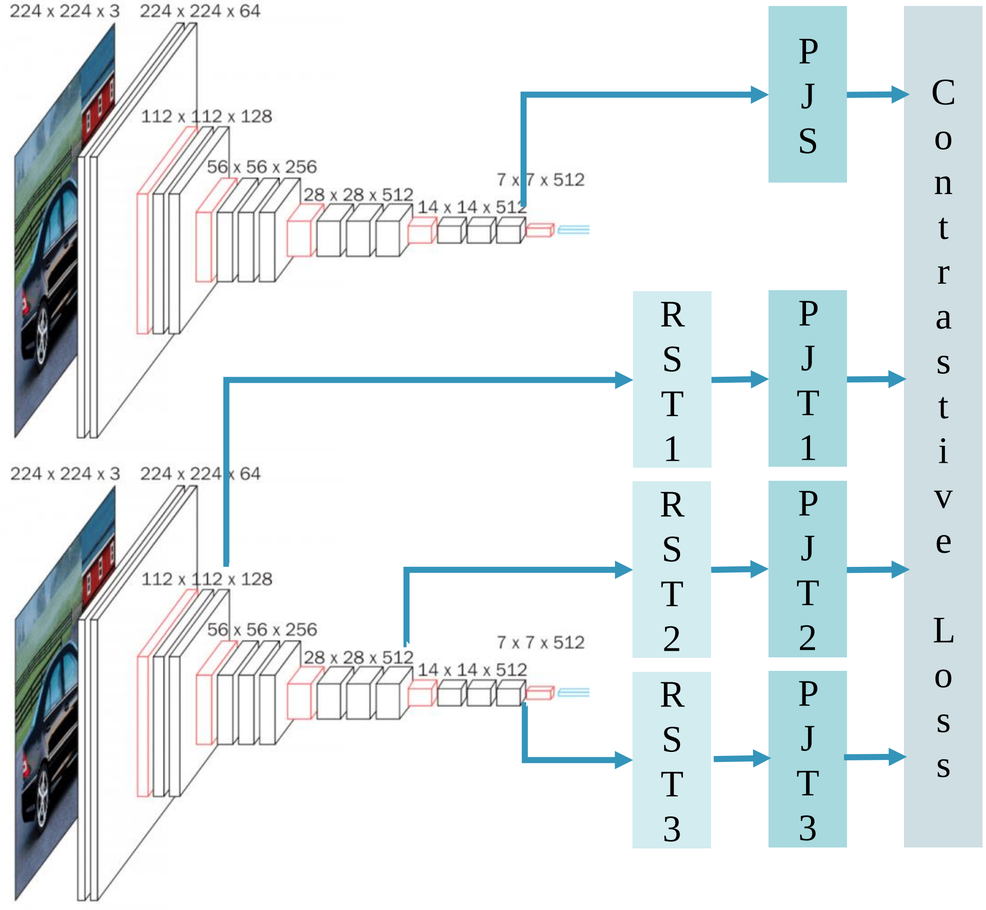
Red bold represents the method that takes the largest amount of time to train

*Experiments were conducted across a variety of workstations

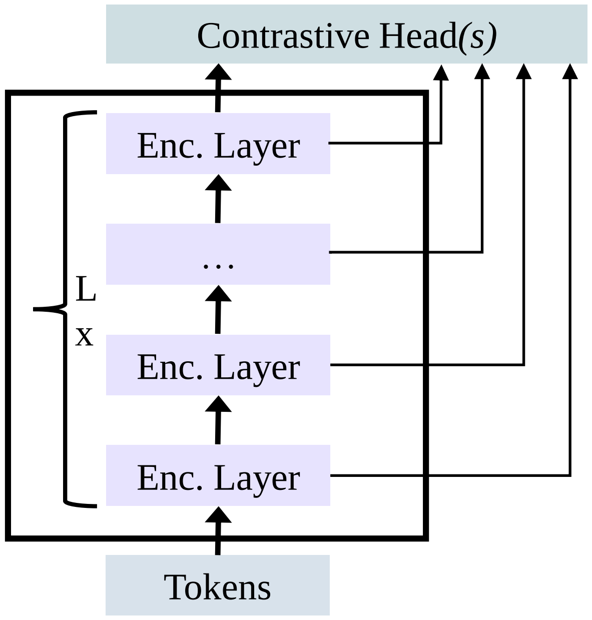
Future Work and Conclusion

Future Work

- Separate projectors for each intermediate features



$$L = \alpha L_{CE} + \beta L_{KLDiv} + \gamma L_{IFACD}$$



Contrastive Head:
1. *Rescaler MLP*
2. Projection MLP

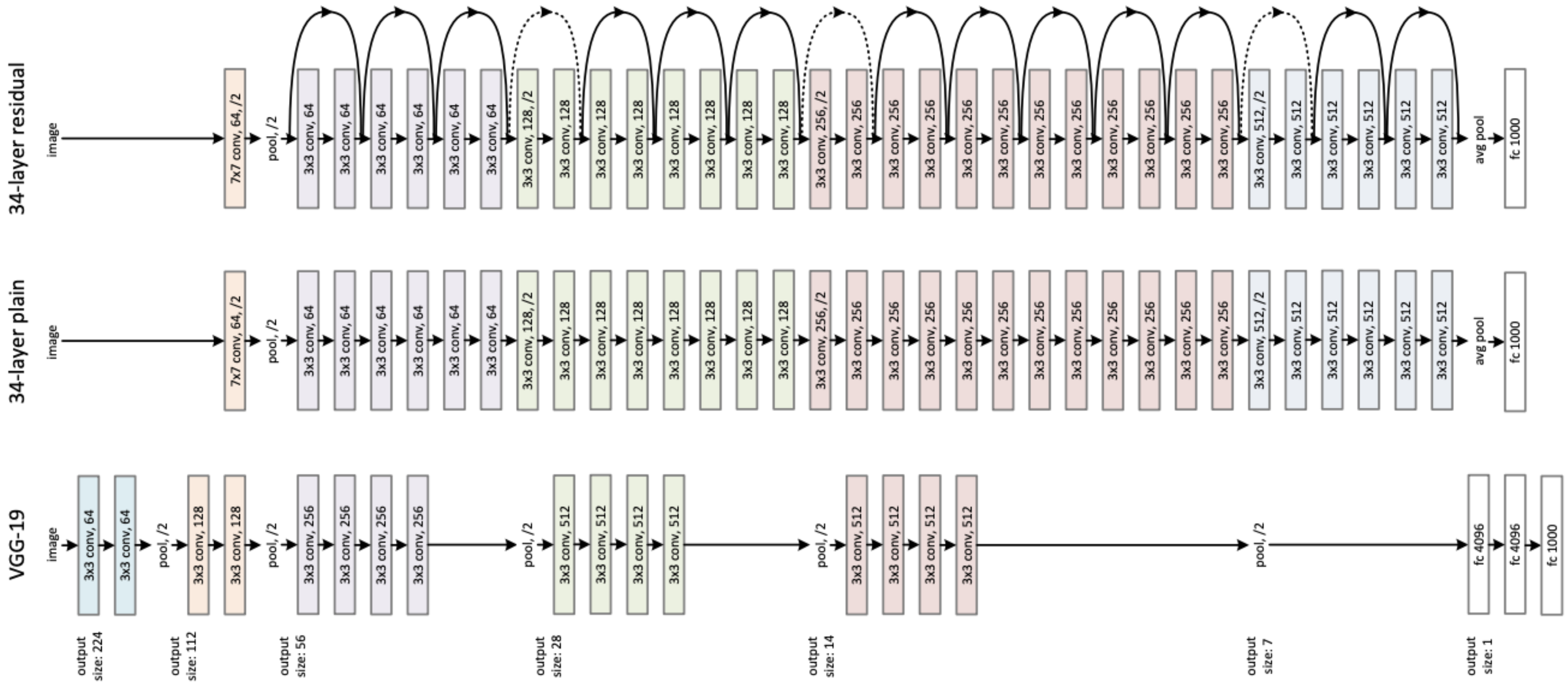
MLP:
1. *Spatial pooling*
2. FC Layer
3. *LN/BN1d*
4. *GELU/ReLU*
5. *FC Layer*
6. *LN/BN1d*
7. *GELU/ReLU*
8. *FC Layer*
9. *LN/BN1d*

https://www.cs.toronto.edu/~frossard/post/vgg16/

Italics represent the component is optional

Future Work

- Intermediate layers choices
  - Blocks vs last



Deep Residual Learning for Image Recognition. He et al. CVPR 2016.

Future Work

- Rescaler ablations
  - Size: hidden dimension size, number of layers…
  - More layers in rescaler for shallower layers

- Redesign rescaler module
  - Self-attention / transformer
  - MLP-Mixer

Rescaler:
1. Spatial pooling
2. FC Layer
3. *LN/BN1d*
4. *GELU/ReLU*
5. *FC Layer*
6. *LN/BN1d*
7. *GELU/ReLU*
8. *FC Layer*
9. *LN/BN1d*

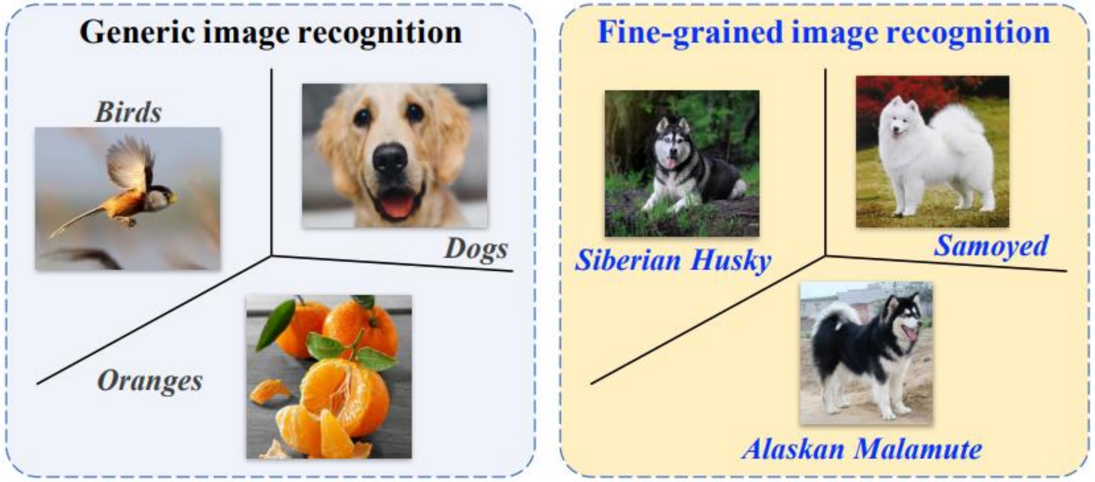
Rescaler V2:
1. *Transformer / MLP-Mixer blocks*
2. Spatial pooling
3. FC Layer
4. *LN/BN1d*
5. *GELU/ReLU*

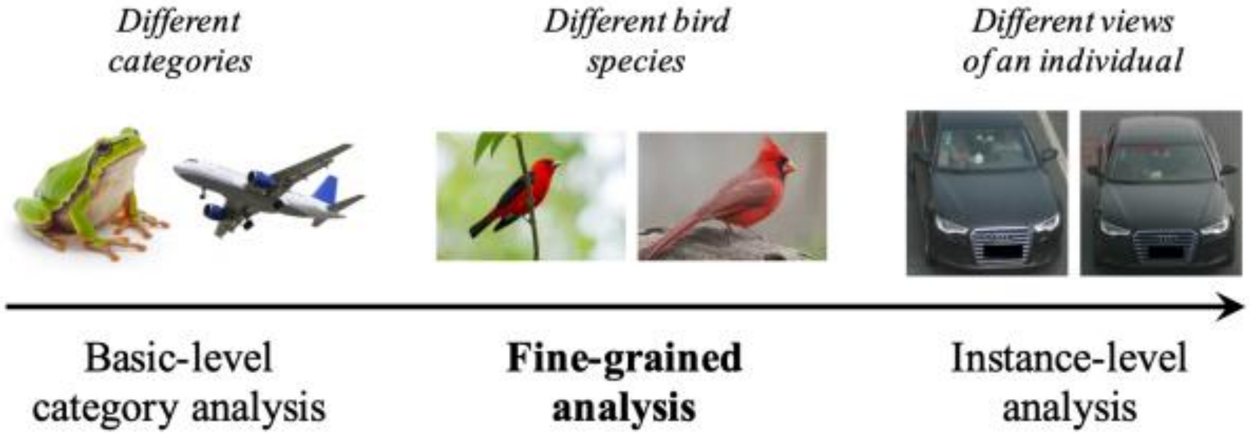
Italics represent the component is optional

Future Work

- Fine-grained applications where small variations and details may be more crucial for accurate recognition

Figure 3: Key challenge of fine-grained image analysis, *i.e.*, small inter-class variations and large intra-class variations. We here present each of four Tern species in each row in the figure, respectively.

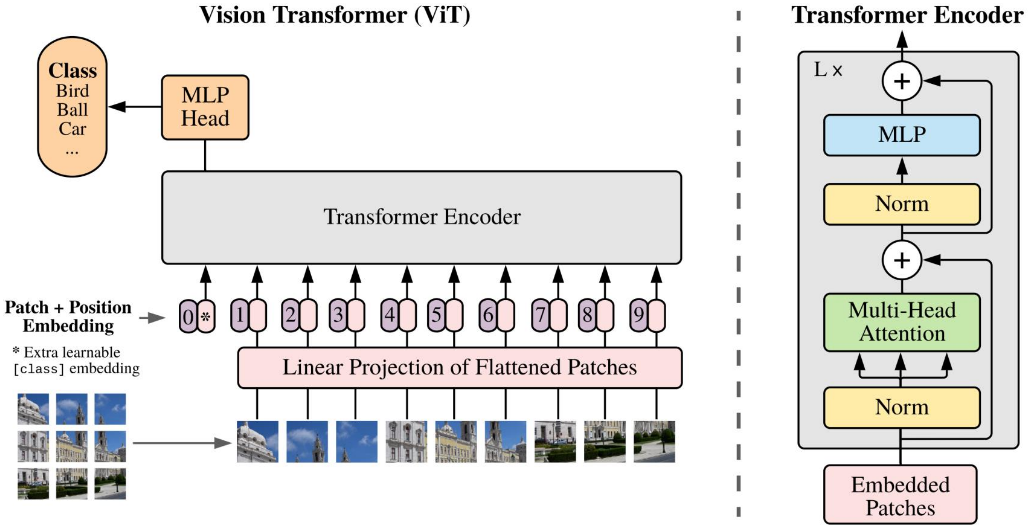


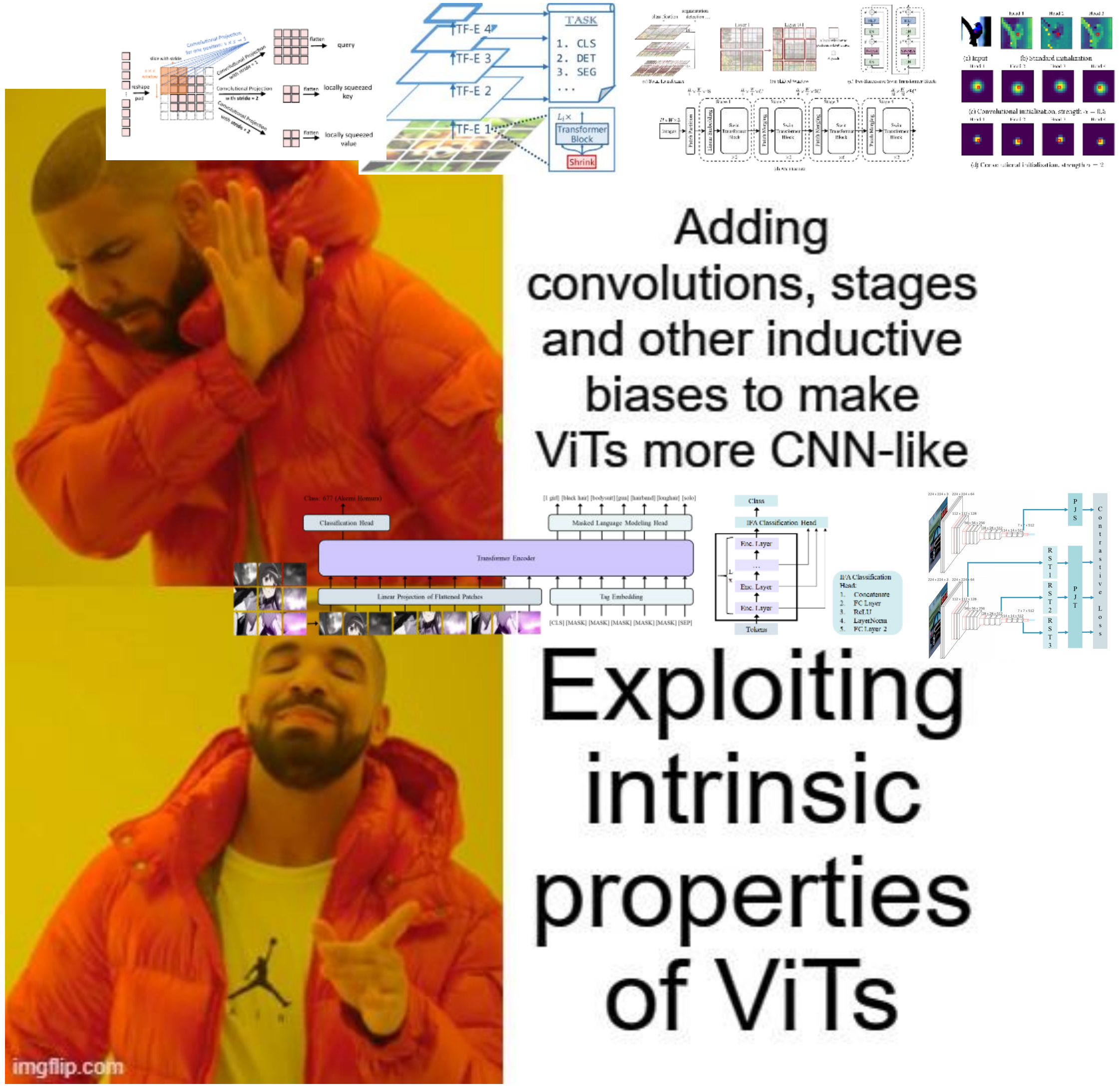


Fine-Grained Image Analysis with Deep Learning: A Survey. Wei et al. TPAMI 2021.

Future Work

- Experiment using ViT / MLP-Mixer architectures





An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. Dosovitskiy et al. ICLR 2021.
CvT: Introducing convolutions to vision transformers. Wu et al. ICCV 2021.
ConViT: Improving vision transformers with soft convolutional inductive biases. d'Ascoli,et al. ArXiv 2021.
Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. Wang et al. ICCV 2021.
Swin transformer: Hierarchical vision transformer using shifted windows. Liu et al. ICCV 2021.
Intermediate Features Augmented Contrastive Distillation
Intermediate Features Aggregation Classification Head and Tag-Augmented Classification and Tagging

Self-Contrastive Learning

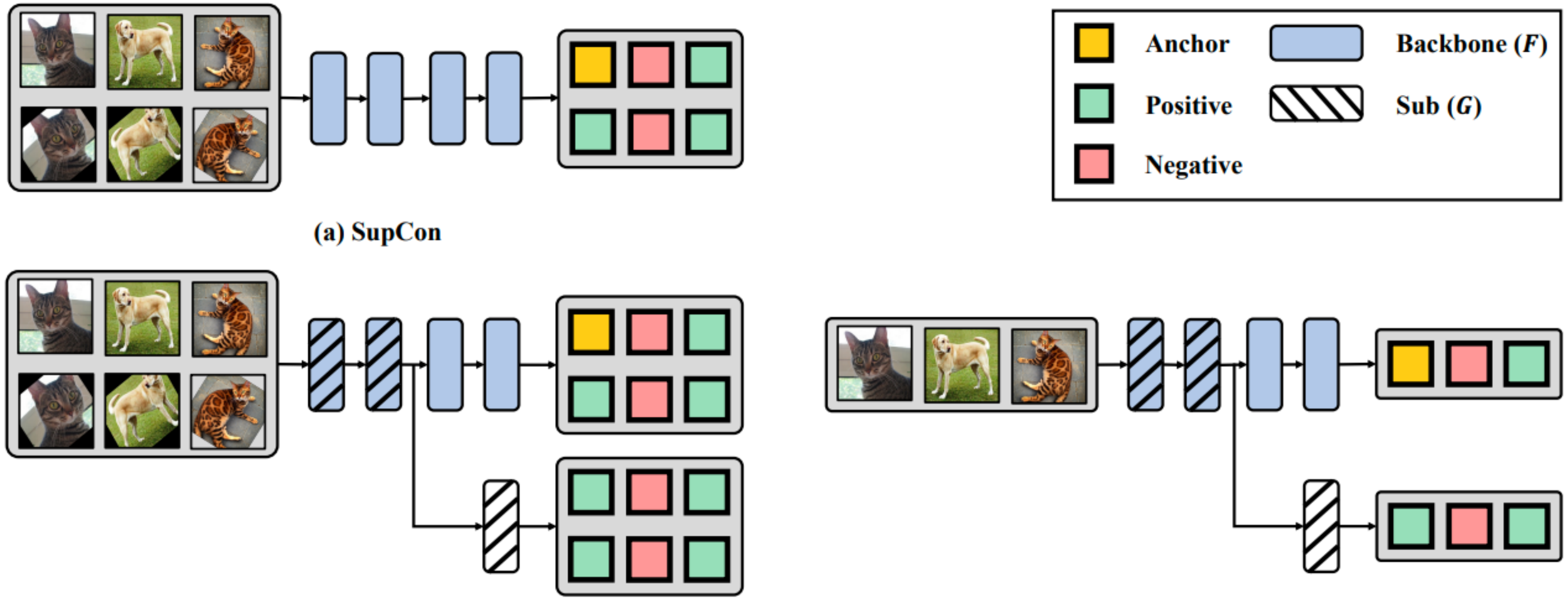
- Contrast outputs from different levels of multi-exit network

$$\mathcal{L}_{sup} = -\sum_{i,p_1} \log \frac{\exp(\boldsymbol{F}(\boldsymbol{x}_i)\cdot\boldsymbol{F}(\boldsymbol{x}_{p_1})/\tau)}{\sum_{p_2}\exp(\boldsymbol{F}(\boldsymbol{x}_i)\cdot\boldsymbol{F}(\boldsymbol{x}_{p_2})/\tau)+\sum_{n}\exp(\boldsymbol{F}(\boldsymbol{x}_i)\cdot\boldsymbol{F}(\boldsymbol{x}_{n})/\tau)}$$

$\boxed{i \in I \equiv \{1,\dots,2B\}}$ $\boxed{p_* \in P_{i*} \equiv \{p \in I \setminus \{i\} \mid y_p = y_i\}}$ $\boxed{n \in N_i \equiv \{n \in I \mid y_n \neq y_i\}}$

$$\mathcal{L}_{self} = -\sum_{\boldsymbol{\omega},\boldsymbol{\omega}_1}\sum_{i,p_1} \log \frac{\exp(\boldsymbol{\omega}(\boldsymbol{x}_i)\cdot\boldsymbol{\omega}_1(\boldsymbol{x}_{p_1})/\tau)}{\sum_{\boldsymbol{\omega}_2}\left(\sum_{p_2}\exp(\boldsymbol{\omega}(\boldsymbol{x}_i)\cdot\boldsymbol{\omega}_2(\boldsymbol{x}_{p_2})/\tau)+\sum_{n}\exp(\boldsymbol{\omega}(\boldsymbol{x}_i)\cdot\boldsymbol{\omega}_2(\boldsymbol{x}_{n})/\tau)\right)}$$

$\boxed{i \in I \equiv \begin{cases}\{1,\dots,B\} & \textbf{(SelfCon-S)}\\ \{1,\dots,2B\} & \textbf{(SelfCon-M)}\end{cases}}$ $\boxed{p_* \in P_{i*} \equiv \{p \in I \mid y_p = y_i\}}$ $\boxed{n \in N_i \equiv \{n \in I \mid y_n \neq y_i\}}$



(a) SupCon

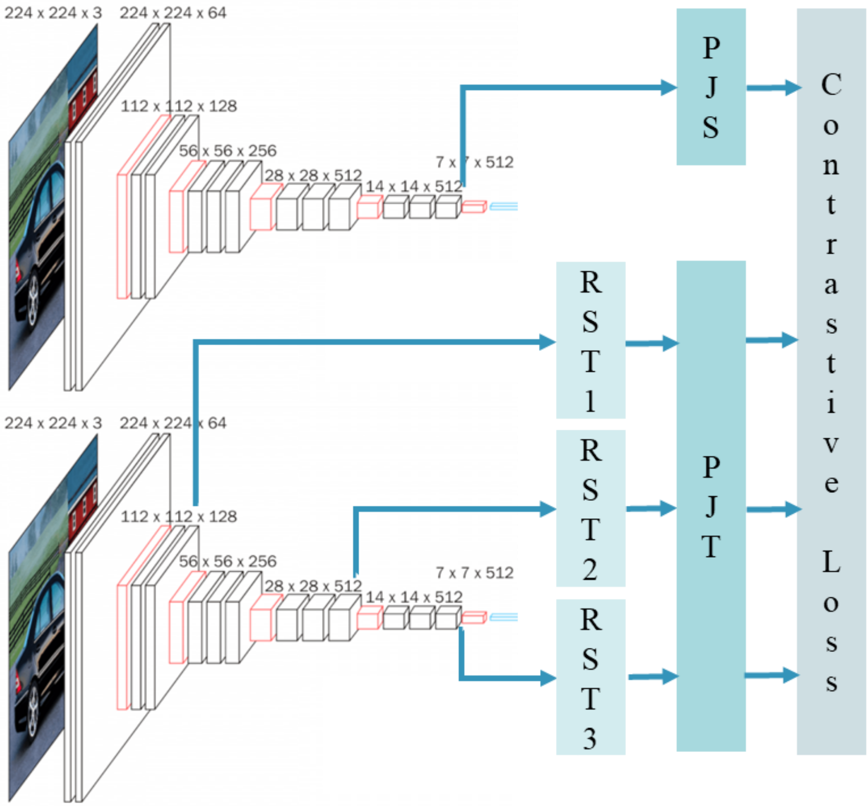
(b) SelfCon with Multi-View

(c) SelfCon with Single-View

[Self-Contrastive Learning. Bae et al. Submission to ICLR 2022.]

Conclusion

- Intermediate features as extra views for contrastive loss with multiple positives and negatives pairs
- Importance of fair comparisons, source-code sharing, and collaboration



[Intermediate Features Augmented Contrastive Distillation]

THANK YOU

Edwin Arkel Rios, PhD Student @ NYCU PCS Lab
edwinarkelrios.ee08@nycu.edu.tw

Appendix A: Intermediate Features Augmented CLR

Motivation: IFACLR

Transformers

- Transformers have revolutionized NLP and now also CV fields

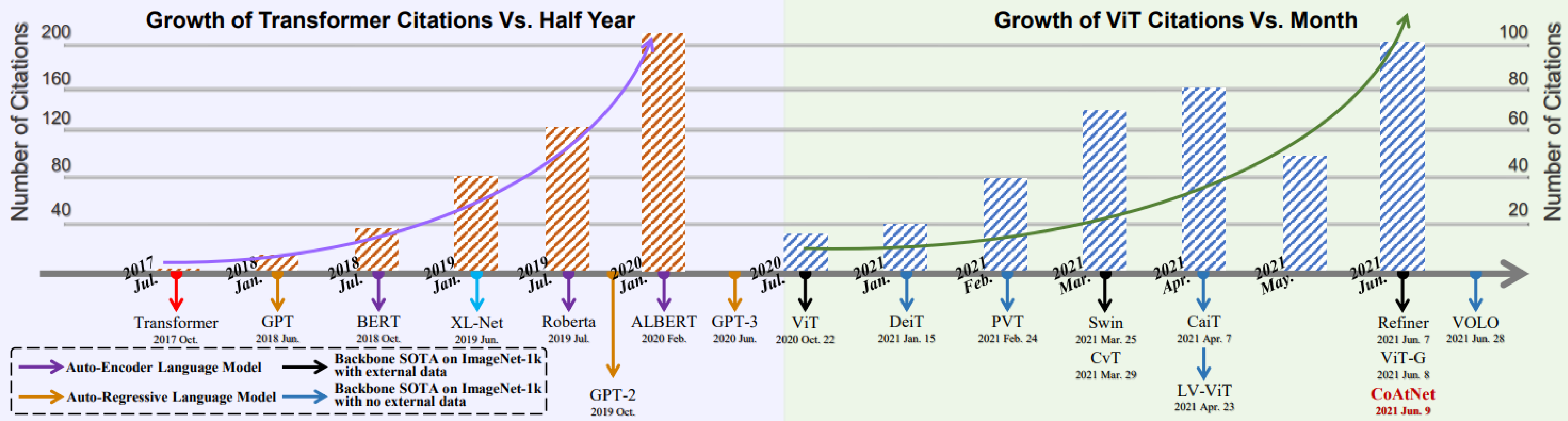
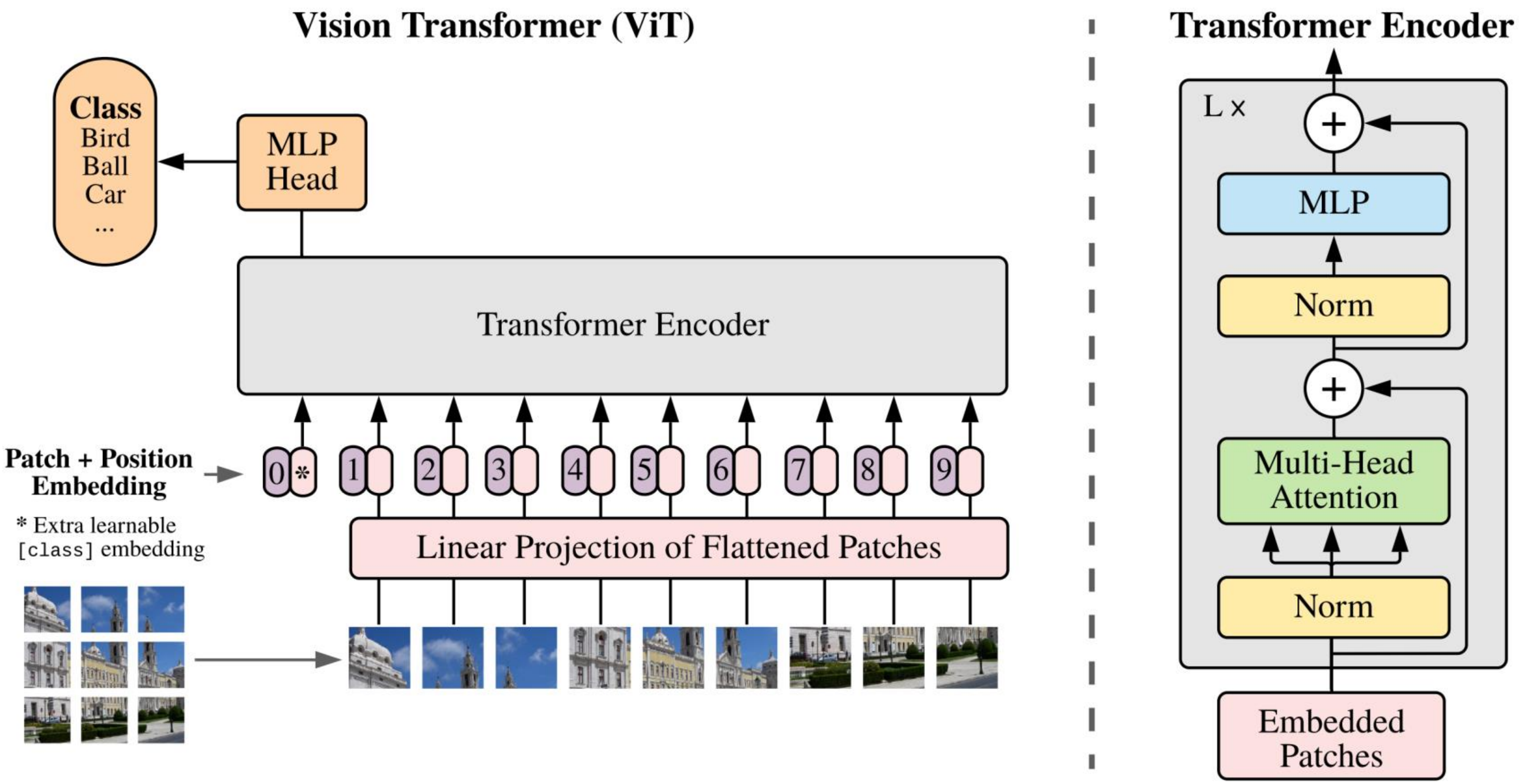


Fig. 1. Odyssey of Transformer application & Growth of both Transformer [1] and ViT [27] citations according to Google Scholar. (Upper Left) Growth of Transformer citations in multiple conference publication including: NIPS, ACL, ICML, IJCAI, ICLR, and ICASSP. (Upper Right) Growth of ViT citations in Arxiv publications. (Bottom Left) Odyssey of language model [1]–[8]. (Bottom Right) Odyssey of visual Transformer backbone where the black [27], [33]–[37] is the SOTA with external data and the blue [38]–[42] refers to the SOTA without external data (best viewed in color).

A Survey of Visual Transformers. Liu et al. arXiv 2021.

Vision Transformer (ViT)

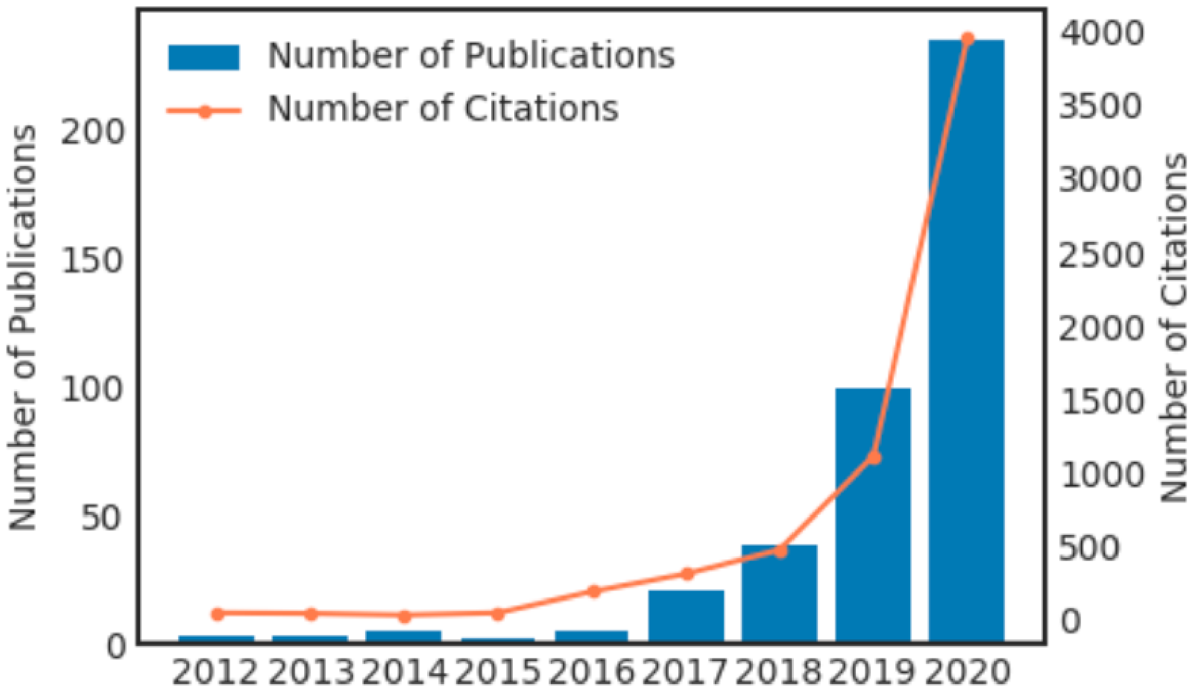
- Applies transformer network directly into images
- Describes an image as a sequence of patches

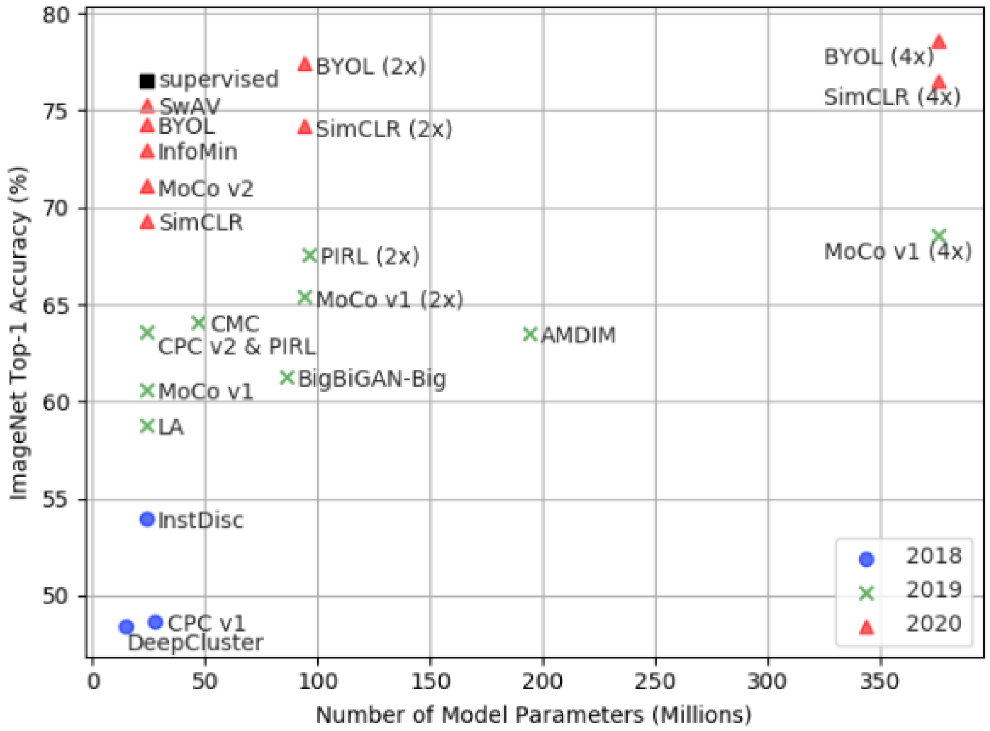


An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. Dosovitskiy et al. ICLR 2021.

Self-Supervised Learning

- Self-supervised learning (SSL) as key to transformers success in NLP
  - Masked language modeling (BERT) and autoregressive prediction (GPT)
- SSL methods increasingly popular for training CNNs and now for ViTs

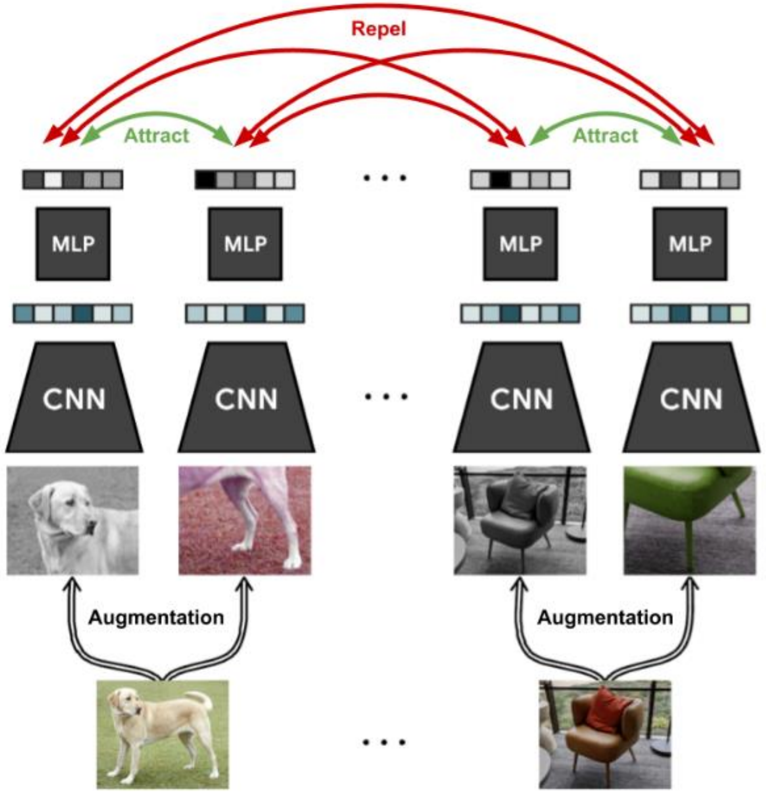




Self-supervised Learning: Generative or Contrastive. Liu et al. IEEE Transactions On Knowledge and Data Engineering 2020.

SimCLR

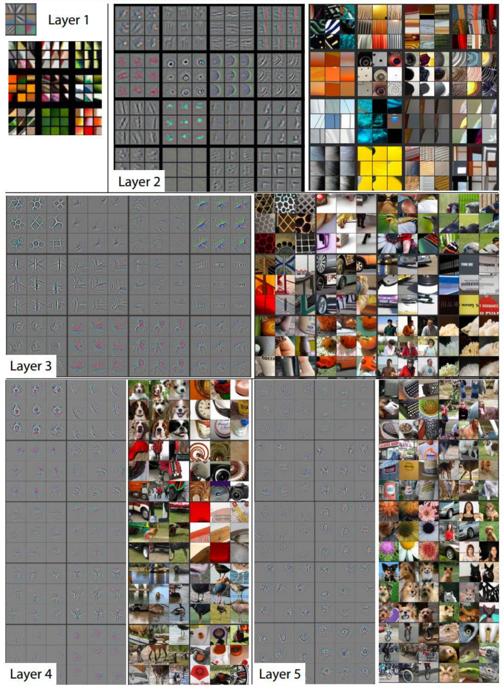
- Contrastive learning of visual representations
  - Two different augmentations of a given image should have representations that are closer to each other than to any other image in a given batch
  - Minimize distance between positive pairs and maximize distance to negative ones



A Simple Framework for Contrastive Learning of Visual Representations. Chen et al. ICML 2020.

Intermediate Features Augmentation

- Feature maps in every transformer layer are exactly the same shape but each layer should extract different features
- Representation for a given image across different layers should be closer between each other than to any other image in same layer or in other layers



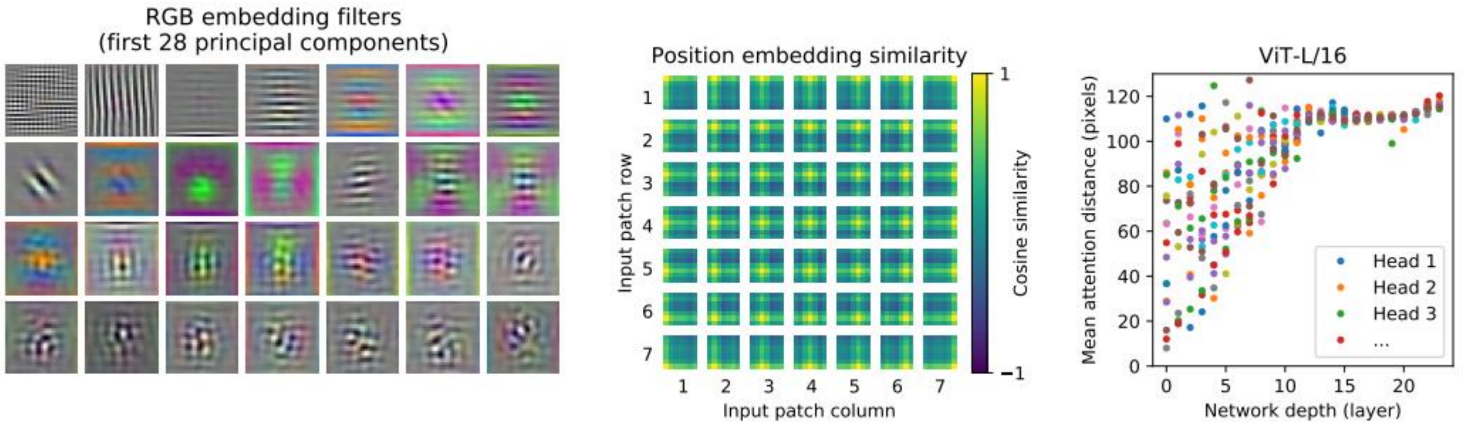


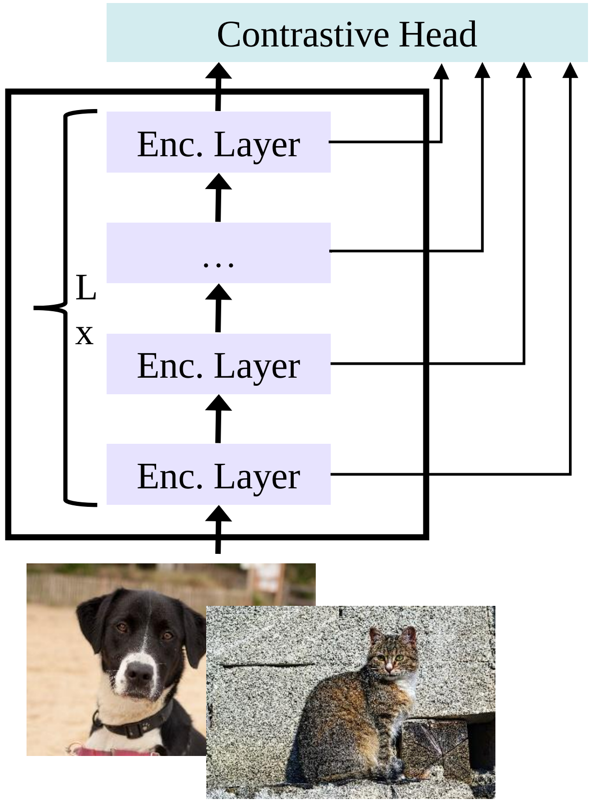
Figure 7: **Left:** Filters of the initial linear embedding of RGB values of ViT-L/32. **Center:** Similarity of position embeddings of ViT-L/32. Tiles show the cosine similarity between the position embedding of the patch with the indicated row and column and the position embeddings of all other patches. **Right:** Size of attended area by head and network depth. Each dot shows the mean attention distance across images for one of 16 heads at one layer. See Appendix D.7 for details.

Visualizing and Understanding Convolutional Networks. Zeiler et al. ECCV 2014.

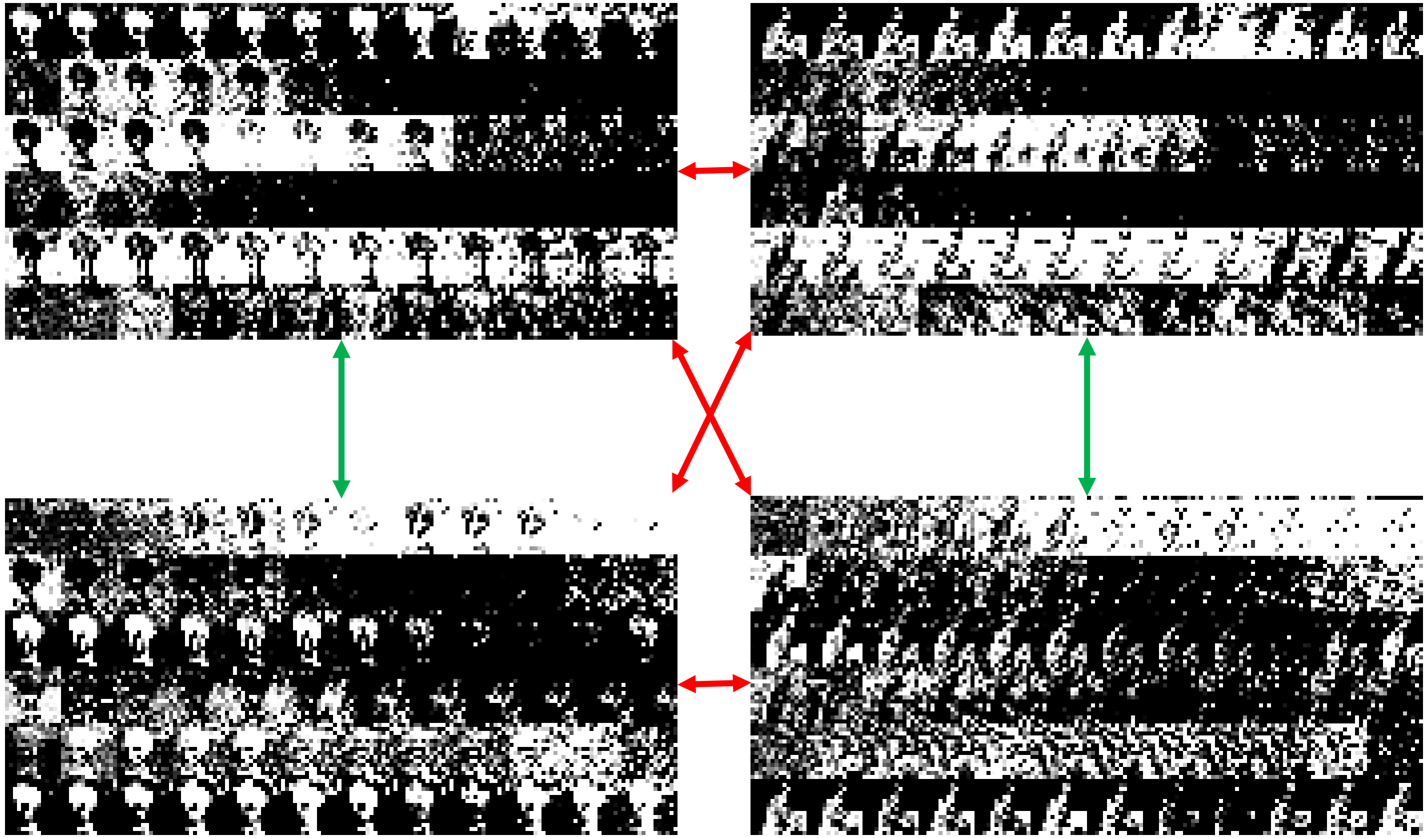
An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. Dosovitskiy et al. ICLR 2021.

IFACLR

- Representation of each layer of a given image as positive samples and representations from all other images as negative samples for contrastive loss



Contrastive Head:
1. FC Layer
2. LN/BN1d
3. ReLU
4. FC Layer

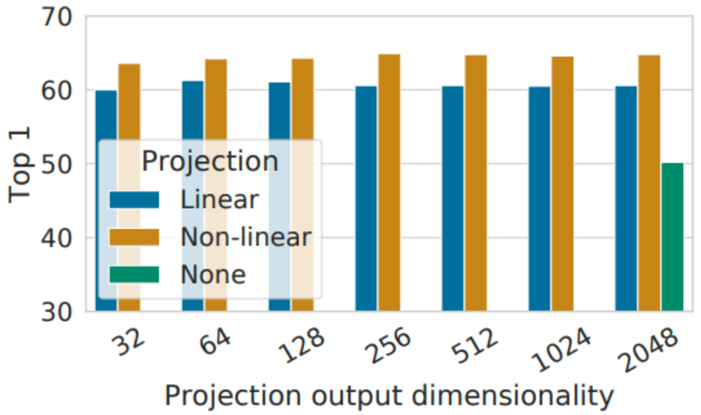


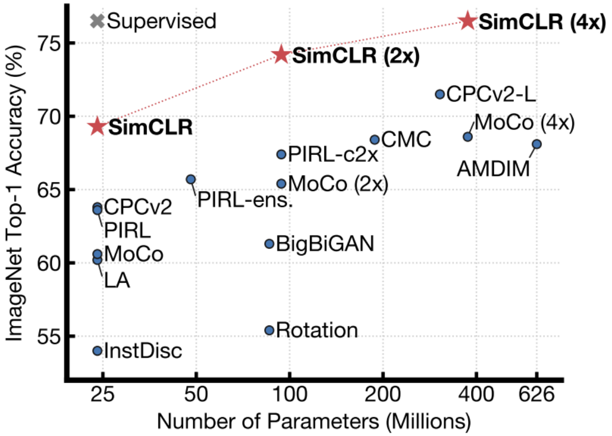
https://github.com/arkel23/layerwiseclr

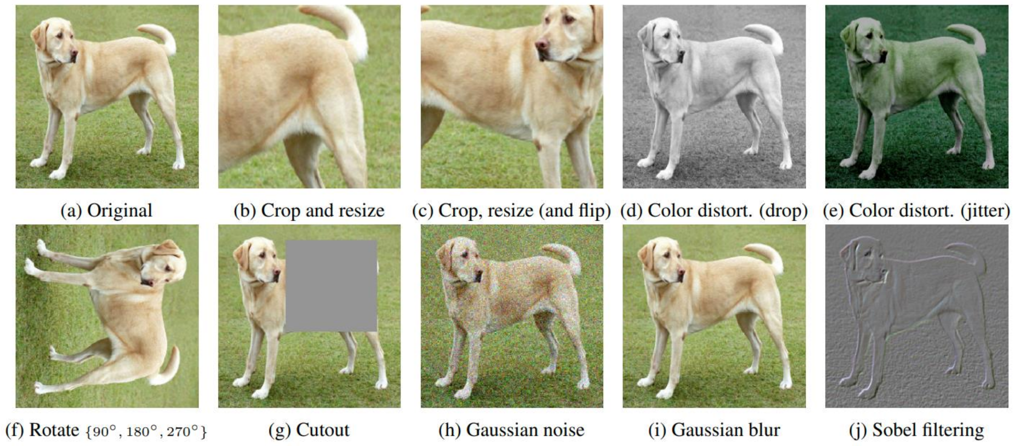
Related Work:
Self-Supervised Learning

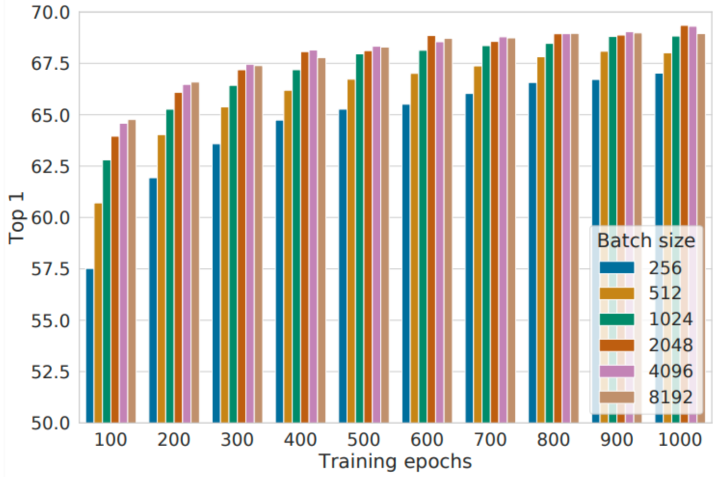
SimCLR

- Data augmentations for contrastive learning: stronger
- Non-linear transformation between representations and contrastive loss (MLP)
- Contrastive loss function choice
- Larger batch size and longer training





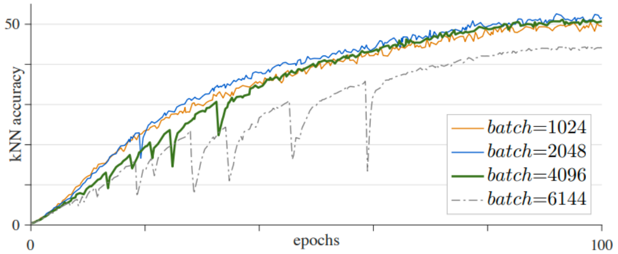




A Simple Framework for Contrastive Learning of Visual Representations. Chen et al. ICML 2020.

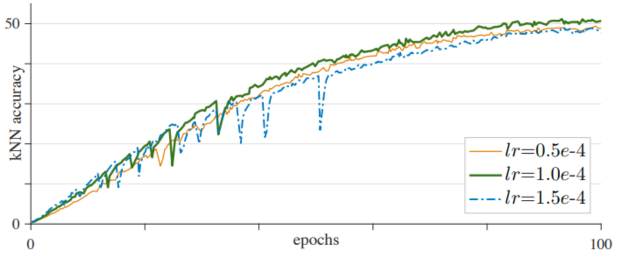
Study on Self-Supervised ViTs

- Study DNN training basics (BS, LR, and optimizer) for training ViTs using SSL
- Compares SSL methods on ViTs vs ResNets



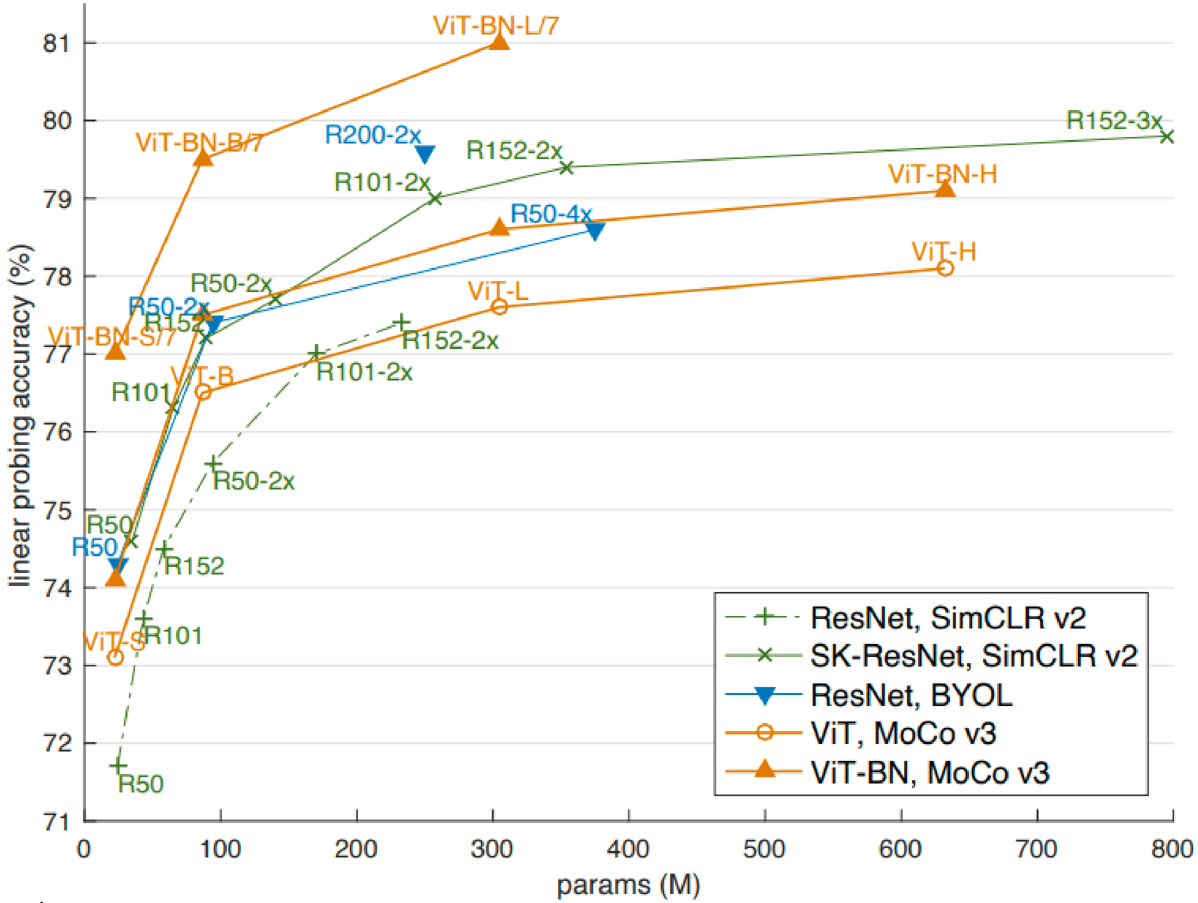
batch	1024	2048	4096	6144
linear acc.	71.5	72.6	72.2	69.7

Figure 1. **Training curves of different batch sizes** (MoCo v3, ViT-B/16, 100-epoch ImageNet, AdamW, lr=1.0e-4).



lr, ×1e-4	0.5	1.0	1.5
linear acc.	70.4	72.2	71.7

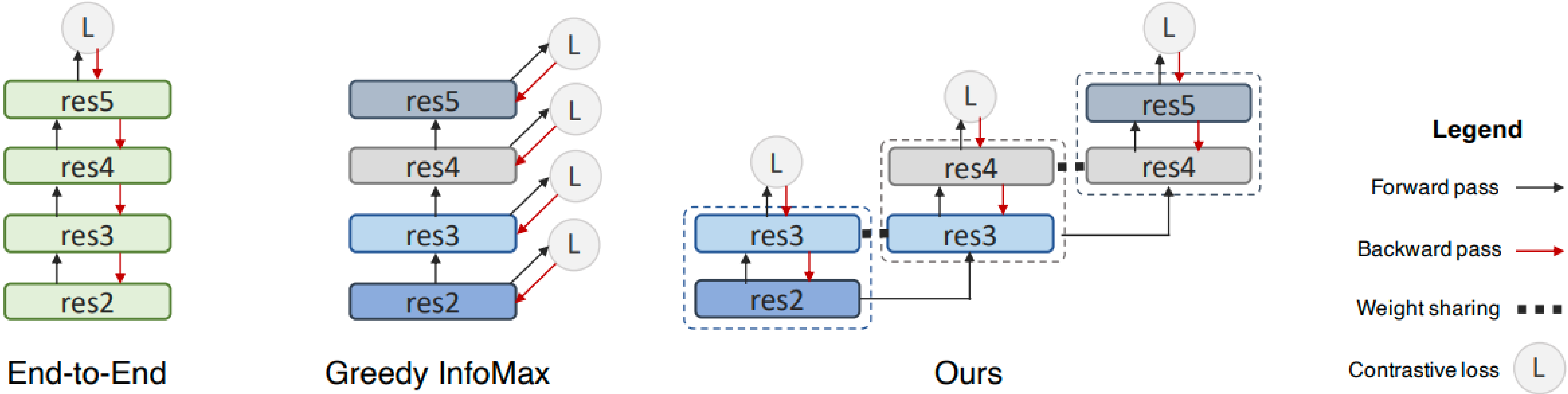
Figure 2. **Training curves of different learning rates** (MoCo v3, ViT-B/16, 100-epoch ImageNet, AdamW, batch 4096).



An Empirical Study of Training Self-Supervised Vision Transformers. Chen et al. ICCV 2021.

Layer-Wise Contrastive Learning

- Perform contrastive learning at each layer, or each few layers
  - Greedy InfoMax (GIM) learns local representations greedily in each stage of network with gradients not backpropagating between stages
  - LoCo proposes "bridges" between stages to receive feedback from deeper layers



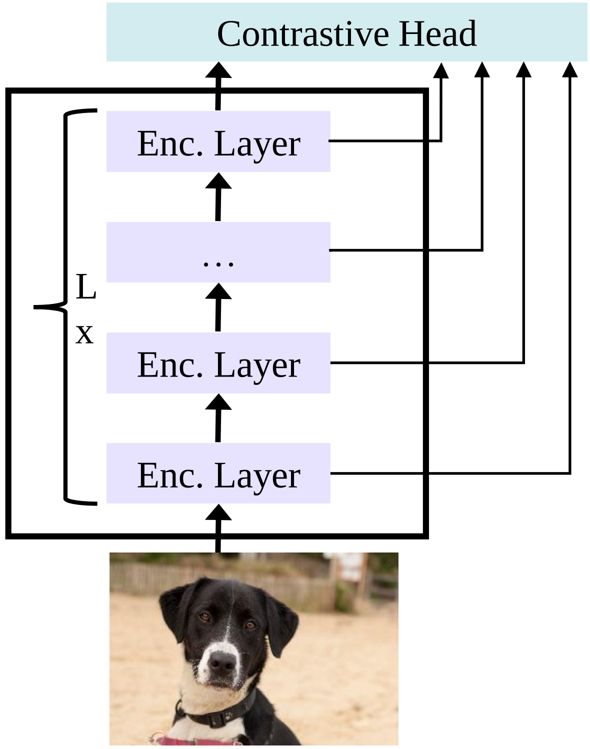
Putting An End to End-to-End: Gradient-Isolated Learning of Representations. Lowe et al. NeurIPS 2019.

LoCo: Local Contrastive Representation Learning. Xiong et al. NeurIPS 2020.

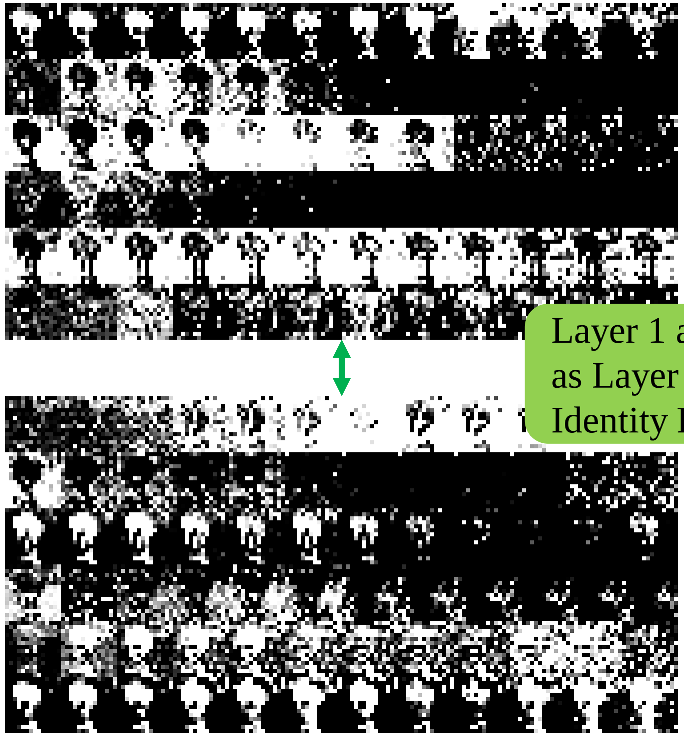
Obstacles

Obstacles

- Representation collapse and early degeneration or overfitting to pretext task
- Selection of appropriate loss for multiple positive and negative pairs
- Lack of experience and resources to properly compare hyperparameter settings and design choices in commonly used SSL settings



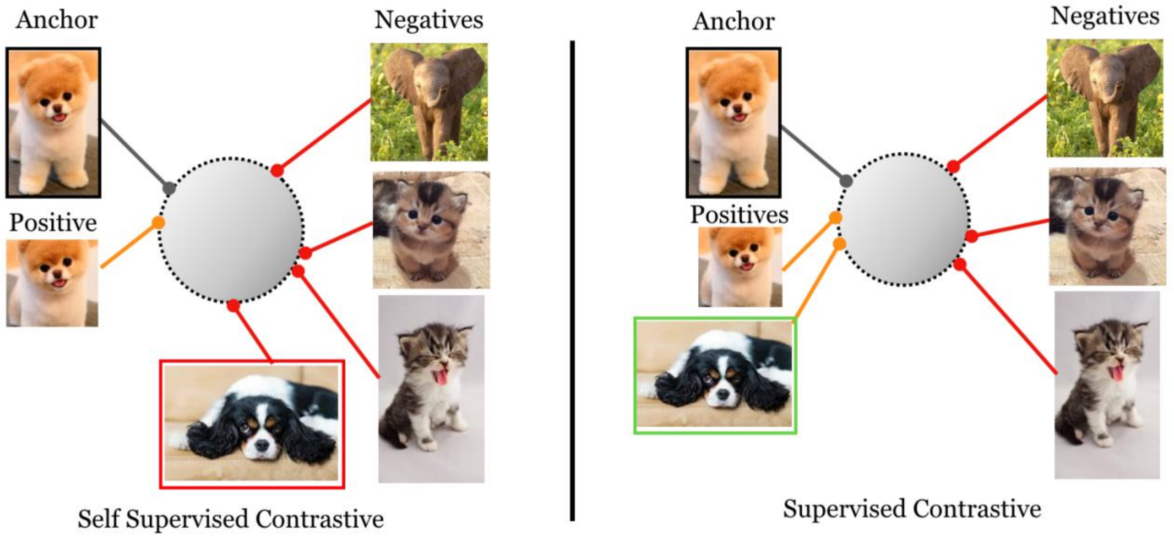
Contrastive Head:
1. FC Layer
2. LN/BN1d
3. ReLU
4. FC Layer



Layer 1 as similar as Layer N => Identity Function

Supervised Contrastive Loss

- SupCon combines contrastive (InfoNCE) and N-pairs loss
- Generalization to arbitrary positives
- Contrastive increases with more negative samples



$$\mathcal{L}^{self} = \sum_{i \in I} \mathcal{L}_i^{self} = -\sum_{i \in I} \log \frac{\exp\left(\boldsymbol{z}_i \cdot \boldsymbol{z}_{j(i)} / \tau\right)}{\sum_{a \in A(i)} \exp\left(\boldsymbol{z}_i \cdot \boldsymbol{z}_a / \tau\right)}$$

$$\mathcal{L}_{out}^{sup} = \sum_{i \in I} \mathcal{L}_{out,i}^{sup} = \sum_{i \in I} \frac{-1}{|P(i)|} \sum_{p \in P(i)} \log \frac{\exp\left(\boldsymbol{z}_i \cdot \boldsymbol{z}_p / \tau\right)}{\sum_{a \in A(i)} \exp\left(\boldsymbol{z}_i \cdot \boldsymbol{z}_a / \tau\right)}$$

Supervised Contrastive Learning. Khosla et al. NeurIPS 2020.

BYOL: SSL Without Negatives

- Two networks: online and target
- Train online network to predict target network representation of different augmentation
- Update target network with moving average of online

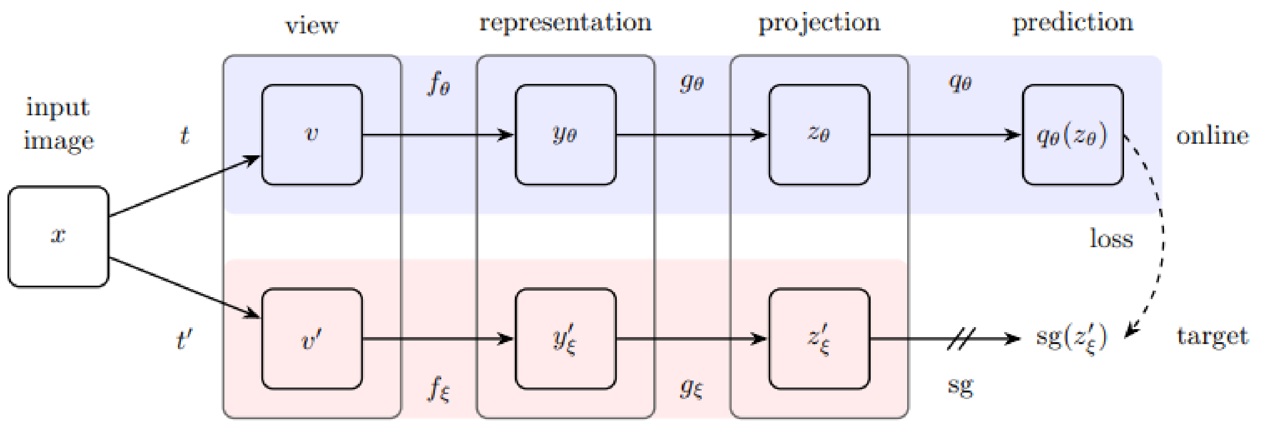
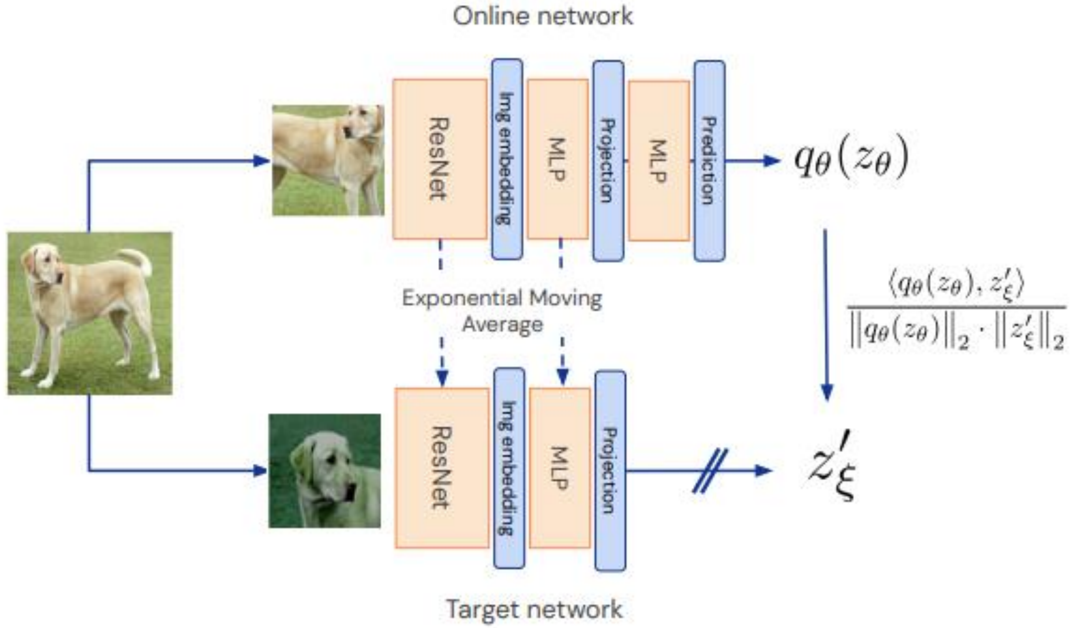


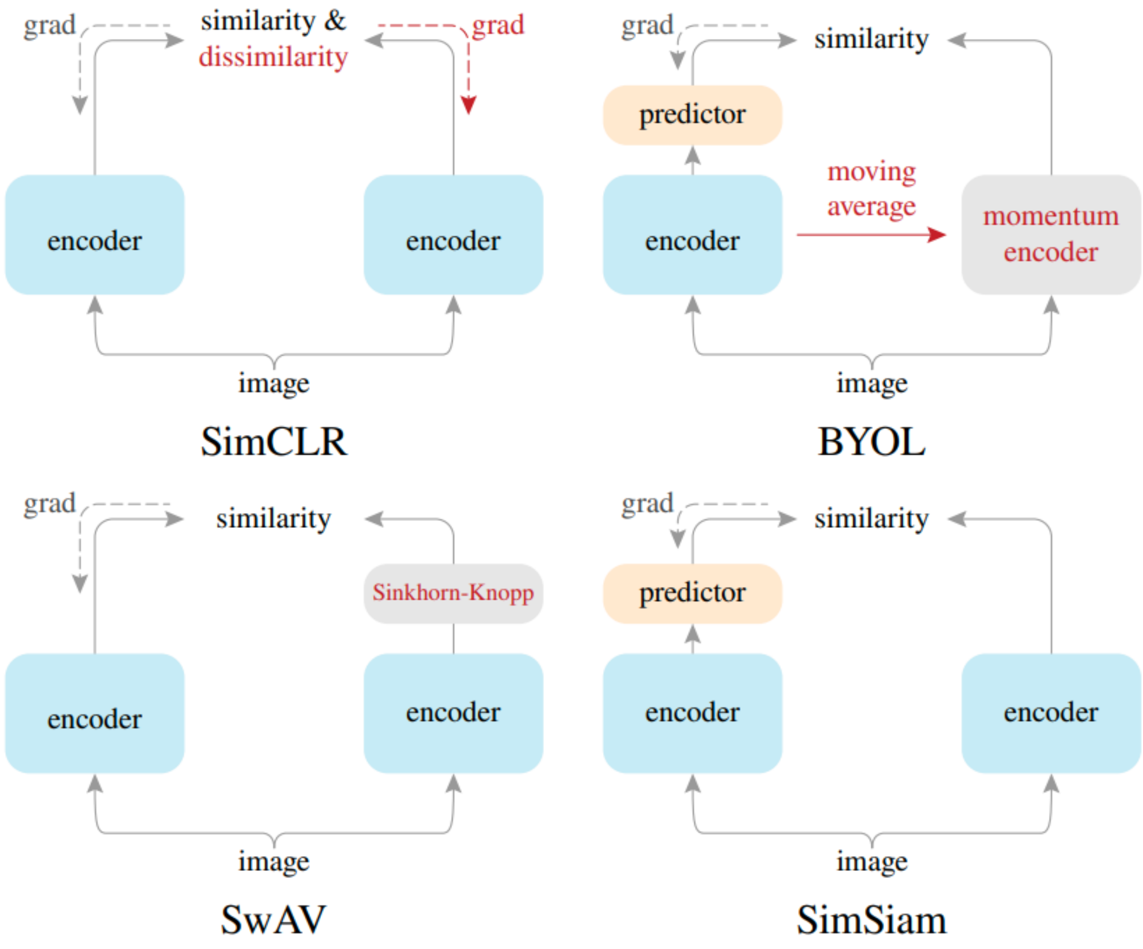
Figure 2: BYOL's architecture. BYOL minimizes a similarity loss between $q_\theta(z_\theta)$ and $\mathrm{sg}(z'_\xi)$, where θ are the trained weights, ξ are an exponential moving average of θ and sg means stop-gradient. At the end of training, everything but f_θ is discarded, and y_θ is used as the image representation.



Bootstrap Your Own Latent A New Approach to Self-Supervised Learning. Grill et al. NeurIPS 2020.

SimSiam

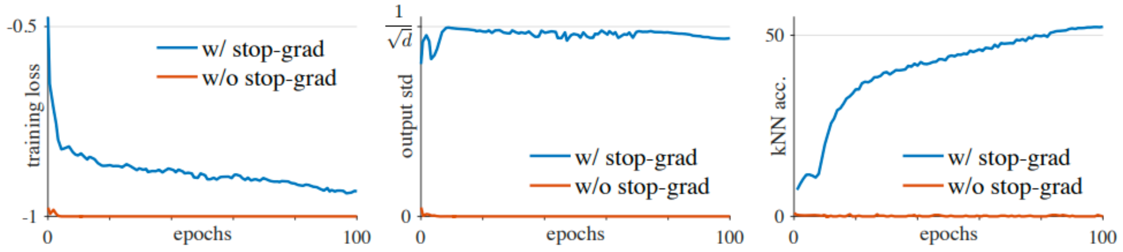
- Siamese network (single network with two views) can learn using BYOL-style objective (predict one view based on other)
  - BYOL without momentum encoder (and therefore auxiliary network)
  - Shares weights between two branches so SimCLR without negative pairs
  - ~~SwAV without online clustering~~
- Stop-gradient operation is critical to prevent collapsing/trivial solutions



Exploring Simple Siamese Representation Learning. Chen et al. CVPR 2021.

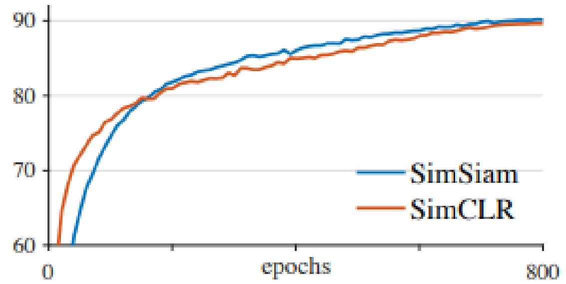
Contrastive Learning Experiments

- Most are done on ImageNet
  - Longer training and large batch sizes leads to better results
- SimSiam does experiments on CIFAR-10 for 800 epochs



	acc. (%)
w/ stop-grad	67.7±0.1
w/o stop-grad	0.1

Figure 2. **SimSiam with *vs.* without stop-gradient**. **Left plot**: training loss. Without stop-gradient it degenerates immediately. **Middle plot**: the per-channel std of the ℓ_2-normalized output, plotted as the averaged std over all channels. **Right plot**: validation accuracy of a kNN classifier [36] as a monitor of progress. **Table**: ImageNet linear evaluation ("w/ stop-grad" is mean±std over 5 trials).



	acc. (%)
SimCLR	91.1
SimSiam	91.8

Figure D.1. **CIFAR-10 experiments**. Left: validation accuracy of kNN classification as a monitor during pre-training. Right: linear evaluation accuracy. The backbone is ResNet-18.

Exploring Simple Siamese Representation Learning. Chen et al. CVPR 2021.