A Simple Framework for Contrastive Learning of Visual Representations

Ting Chen, Simon Kornblith, Mohammad Norouzi, Geoffrey Hinton

Presented by Yien Xu and Lichengxi Huang

Previously…

Learning without human supervision is a long-standing problem

Two mainstream approaches

Generative

+ Learns to generate model pixels in the input space
- Computationally expensive
- May not be necessary for representation learning

Discriminative

+ Learns representations using objective functions
+ Train networks to perform pretext tasks
+ Both the inputs and labels are derived from an unlabeled dataset

SimCLR

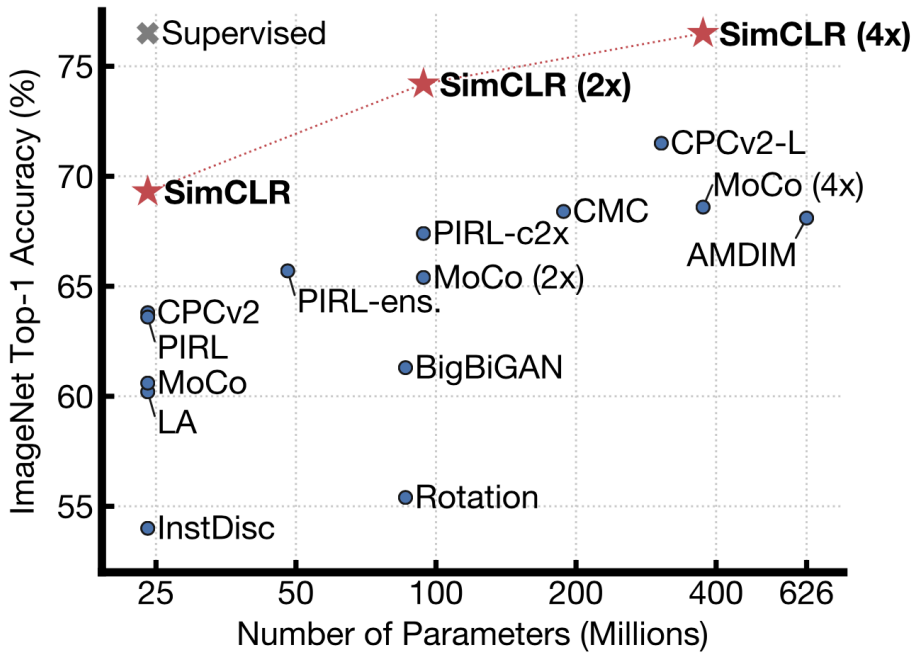


Figure 1. ImageNet Top-1 accuracy of linear classifiers trained on representations learned with different self-supervised methods (pretrained on ImageNet). Gray cross indicates supervised ResNet-50. Our method, SimCLR, is shown in bold.

Experiment Results

Evaluated on ImageNet

- SimCLR achieves 76.5% top-1 accuracy
- 7% relative improvement over previous state-of-the-art

When fine-tuned with only 1% of the ImageNet labels

- SimCLR achieves 85.8% top-5 accuracy
- 10% relative improvement over previous state-of-the-art

When fine-tuned on other natural image classification datasets

- SimCLR performs on par with or better than a strong supervised baseline
- On 10 out of 12 datasets

Outline

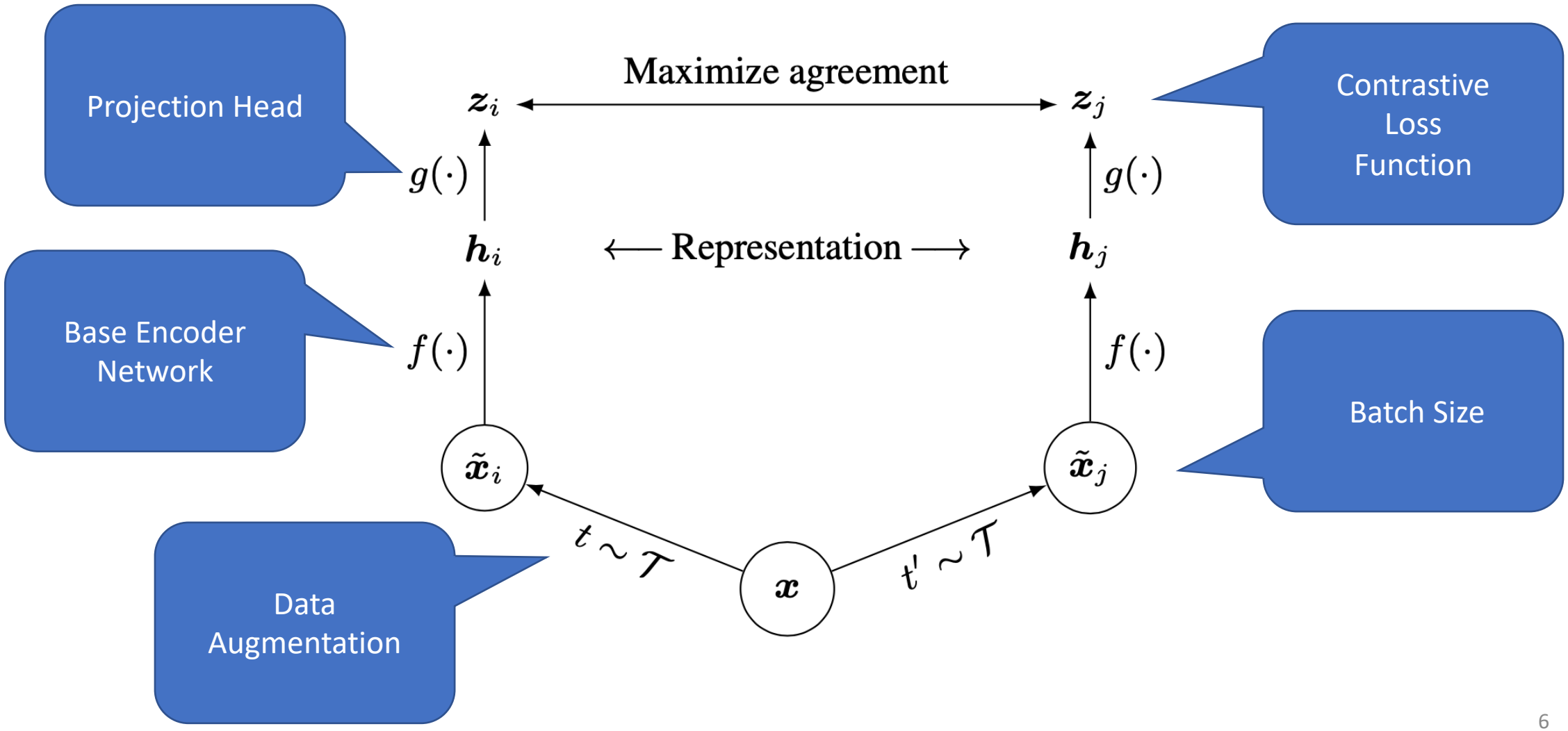
Motivation

Framework

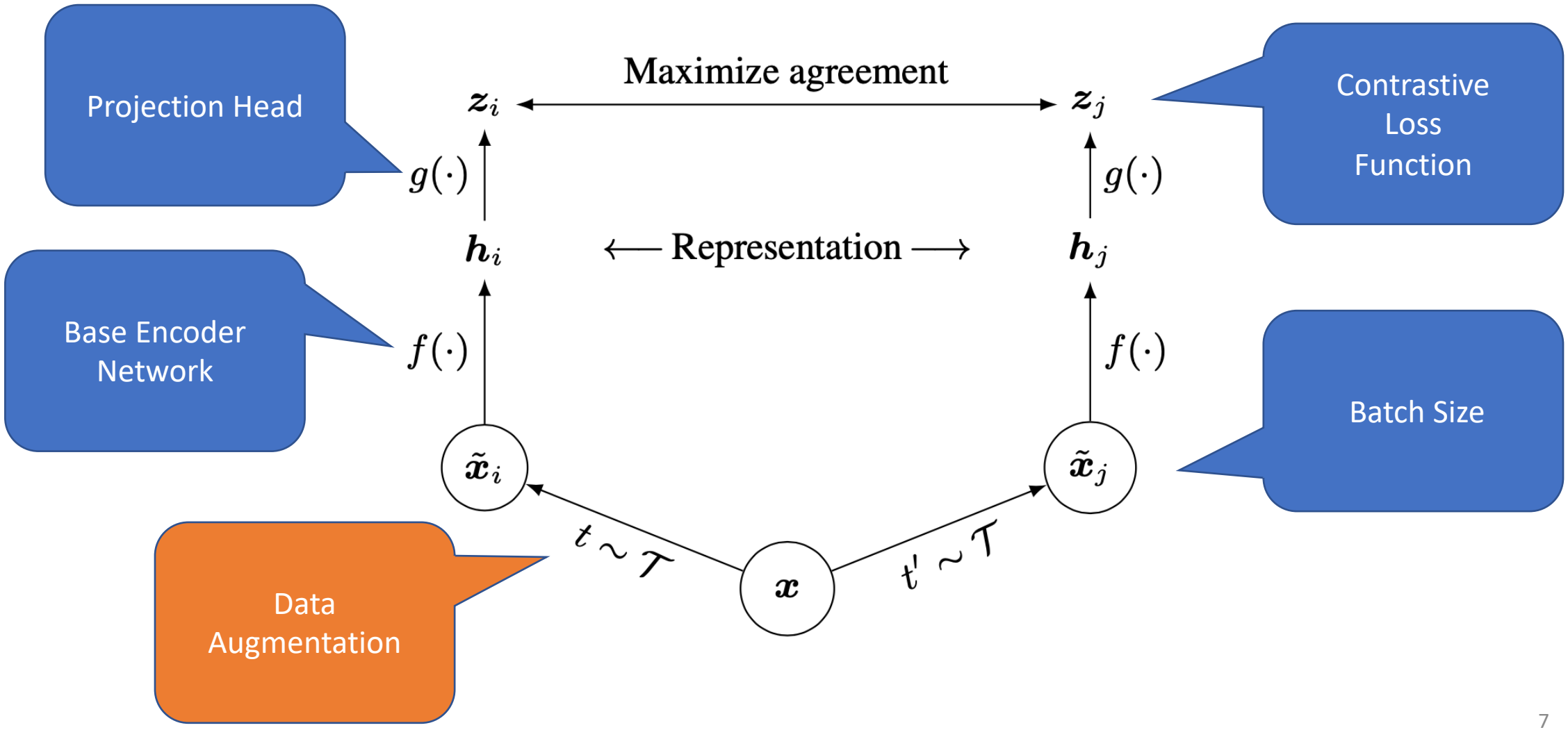
Evaluation

Conclusion

Framework


Projection Head
Base Encoder Network
Data Augmentation
Contrastive Loss Function
Batch Size
Maximize agreement
z_i
z_j
$g(\cdot)$
$g(\cdot)$
h_i
Representation
h_j
$f(\cdot)$
$f(\cdot)$
$\tilde{x}_i$
$\tilde{x}_j$
$t \sim \mathcal{T}$
x
$t' \sim \mathcal{T}$

Framework


Projection Head
Base Encoder Network
Data Augmentation
Contrastive Loss Function
Batch Size
Maximize agreement
z_i
z_j
$g(\cdot)$
$g(\cdot)$
h_i
Representation
h_j
$f(\cdot)$
$f(\cdot)$
$\tilde{x}_i$
$\tilde{x}_j$
$t \sim \mathcal{T}$
x
$t' \sim \mathcal{T}$

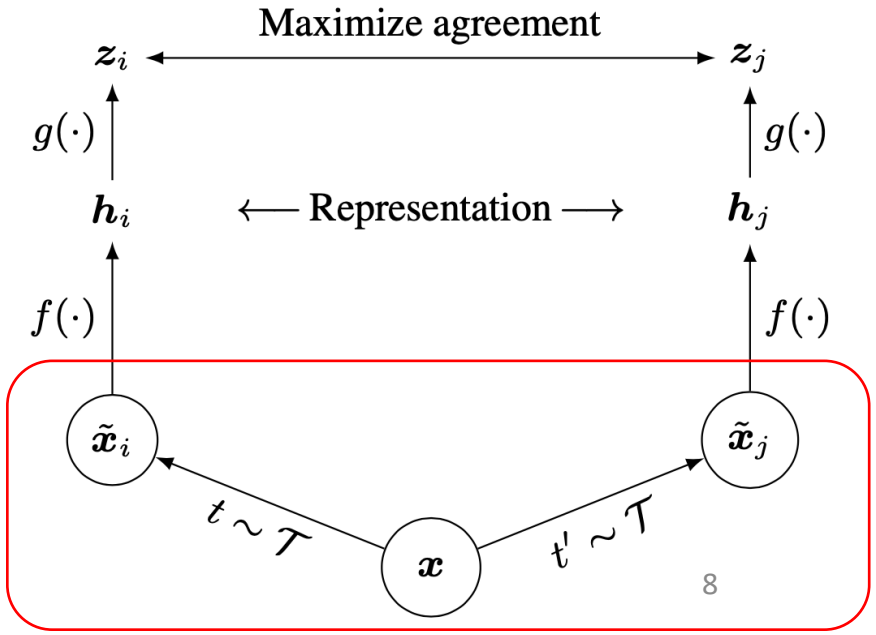
Data Augmentation

Transforms any given data example randomly

Results in two correlated views of the same example

Three augmentations applied sequentially

- Random cropping
- Random color distortions
- Random Gaussian blur



Data Augmentation

- Why? - Data augmentation defines predictive tasks
- Previous approaches
  - Define contrastive prediction tasks by changing the architecture
    - global-to-local view prediction via constraining the receptive field in the network architecture
    - neighboring view prediction via a fixed image splitting procedure and a context aggregation network
- Can be avoided by performing simple random cropping (with resizing)
- Broader contrastive prediction tasks can be defined
  - By extending the family of augmentations
  - By composing them stochastically

Data Augmentation

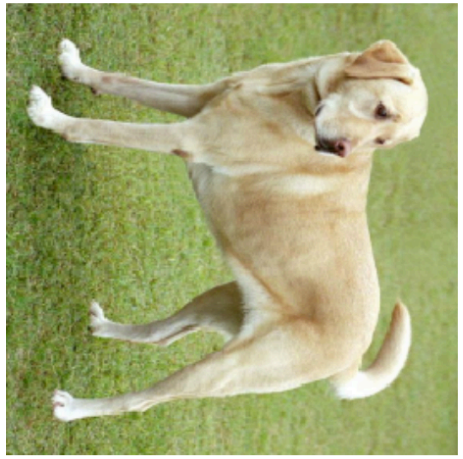
(a) Original

(b) Crop and resize

(c) Crop, resize (and flip)

(d) Color distort. (drop)

(e) Color distort. (jitter)



(f) Rotate $\{90°, 180°, 270°\}$

(g) Cutout

(h) Gaussian noise

(i) Gaussian blur

(j) Sobel filtering

Composition of Data Augmentation

To investigate which data augmentation to perform

- Apply augmentations individually or in pairs

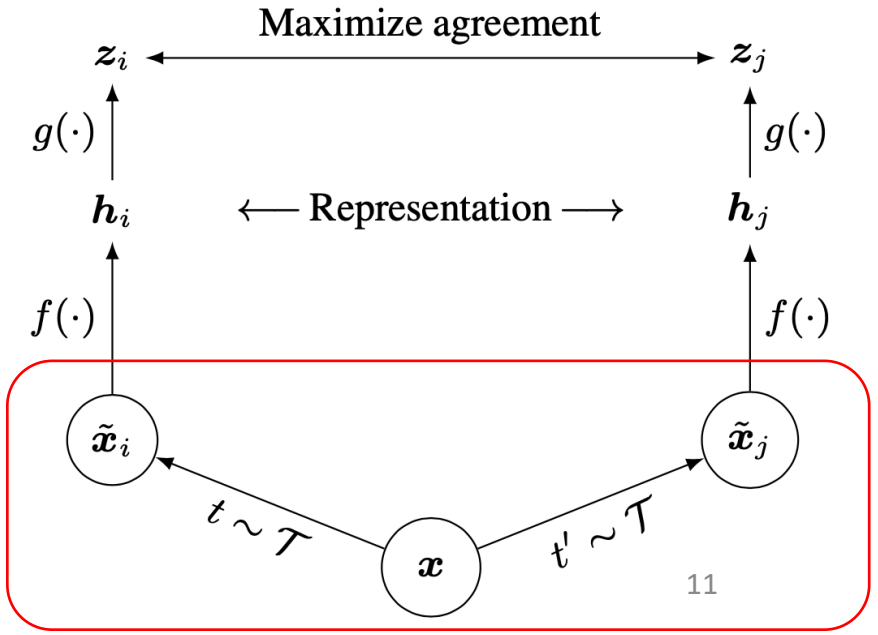
Always apply crop and resize images (since ImageNet are of diff. size)

On one branch

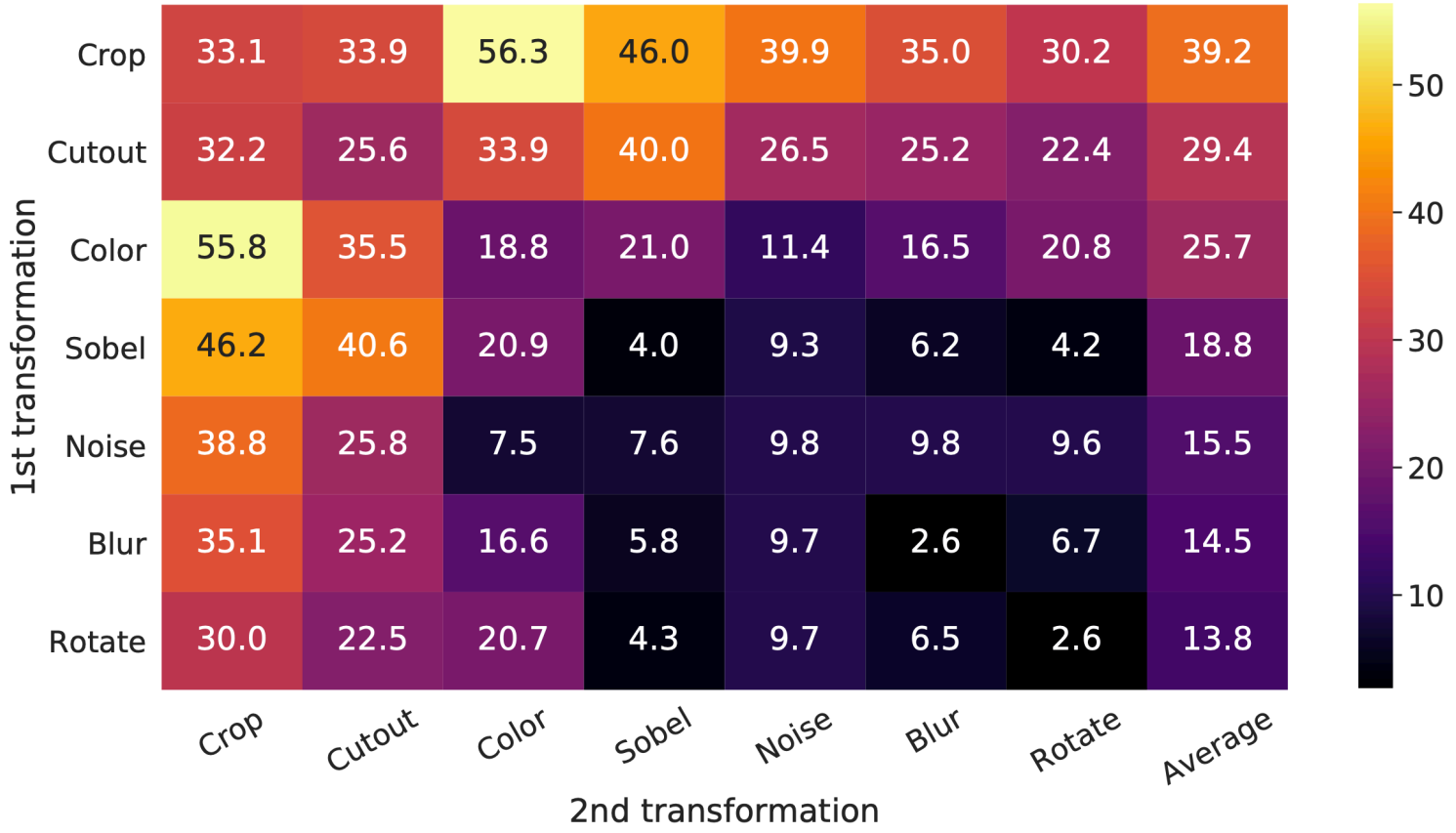
- apply the targeted transformation(s)

On the other branch

- Leave it as identity ($t(\boldsymbol{x}_i) = \boldsymbol{x}_i$)



Composition of Data Augmentation



Composition of Data Augmentation

No single transformation suffices to learn good representations

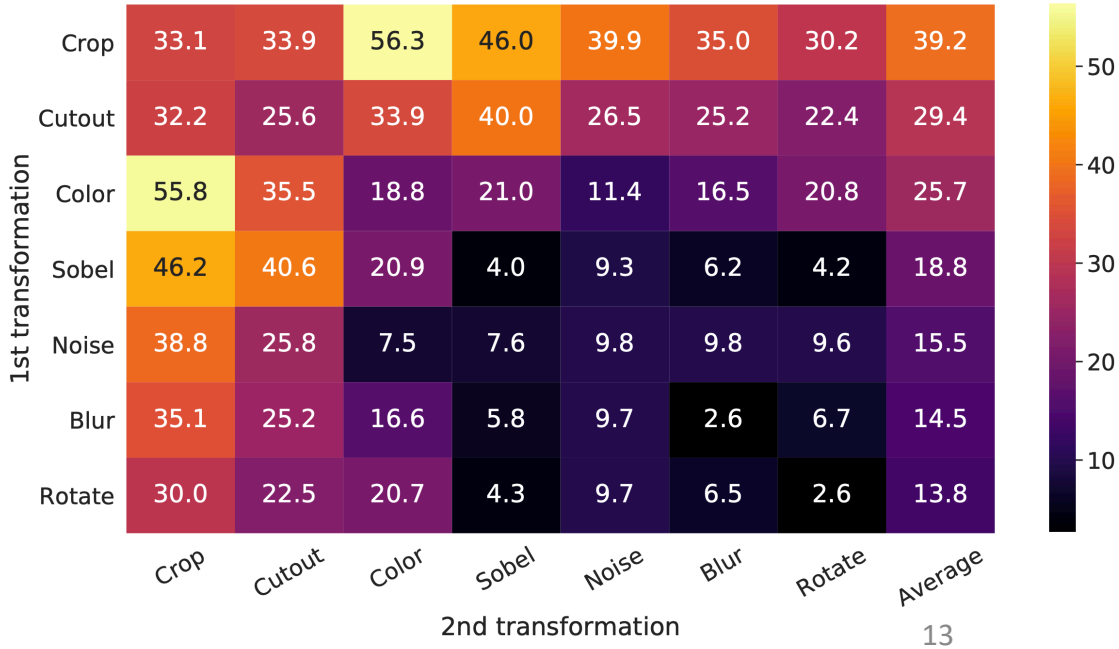
When composing augmentations

The quality of representation improves

Note a composition of augmentations:

random cropping

random color distortion



Composition of Data Augmentation

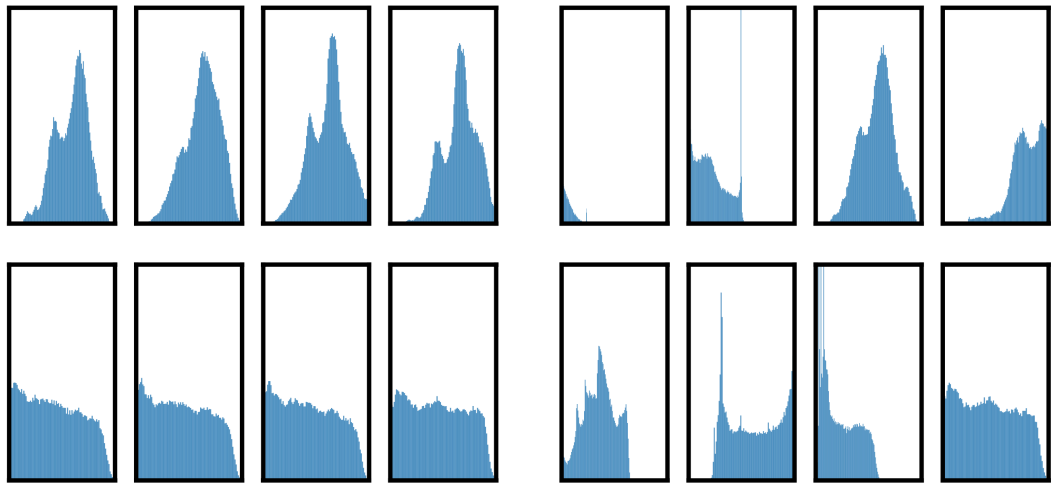
Why Color Distortion?

Before

- Random cropping of images share similar color distributions
- NN may take this shortcut

After

- Suffices to distinguish images



(a) Without color distortion. (b) With color distortion.

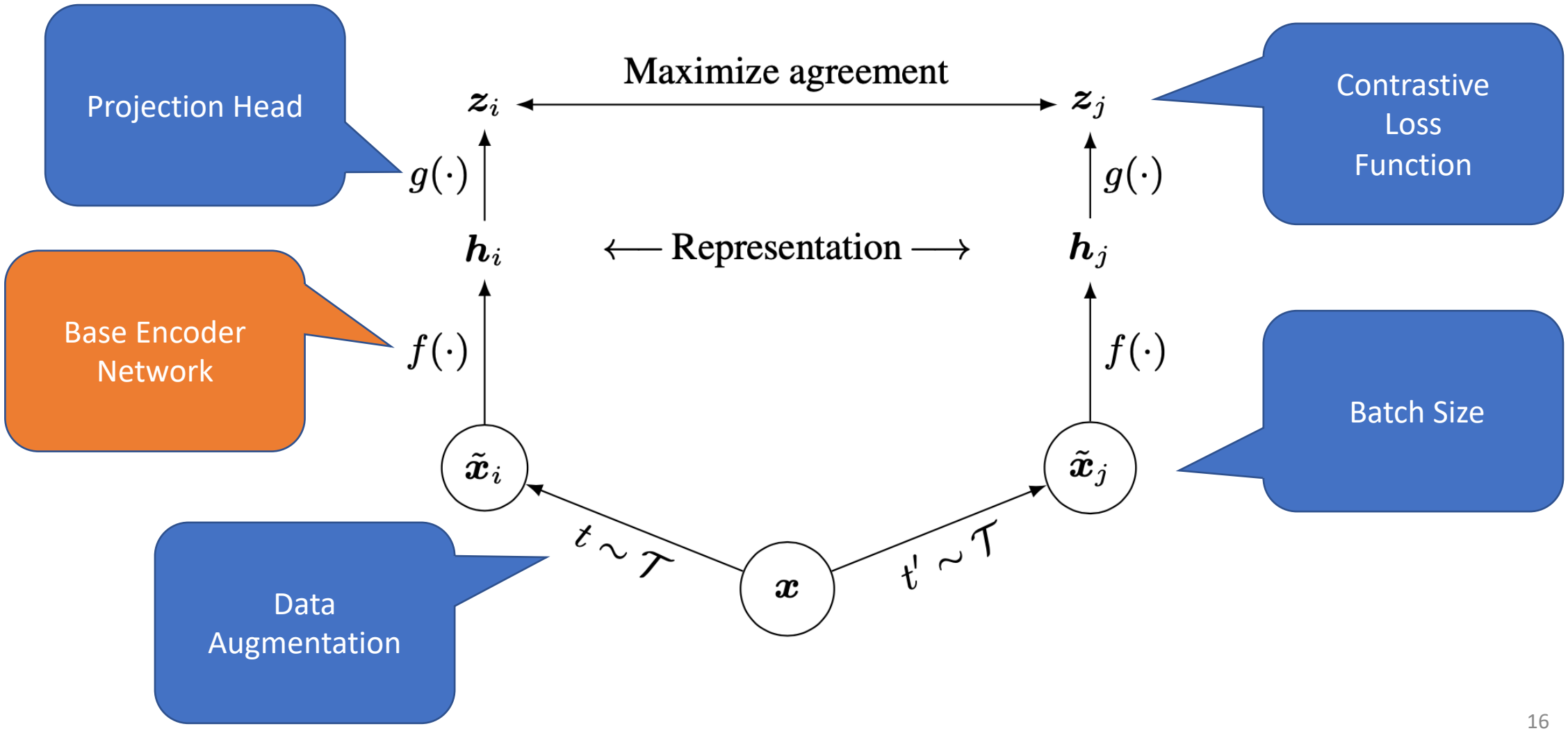
Stronger Data Augmentation

Methods	Color distortion strength 1/8	1/4	1/2	1	1 (+Blur)	AutoAug
SimCLR	59.6	61.0	62.6	63.2	64.5	61.1
Supervised	77.0	76.7	76.5	75.7	75.4	77.1

Unsupervised contrastive learning benefits from

Stronger (color) data augmentation than supervised learning

Framework


Projection Head
Base Encoder Network
Data Augmentation
Contrastive Loss Function
Batch Size
Maximize agreement
z_i
z_j
$g(\cdot)$
$g(\cdot)$
h_i
Representation
h_j
$f(\cdot)$
$f(\cdot)$
$\tilde{x}_i$
$\tilde{x}_j$
$t \sim \mathcal{T}$
x
$t' \sim \mathcal{T}$

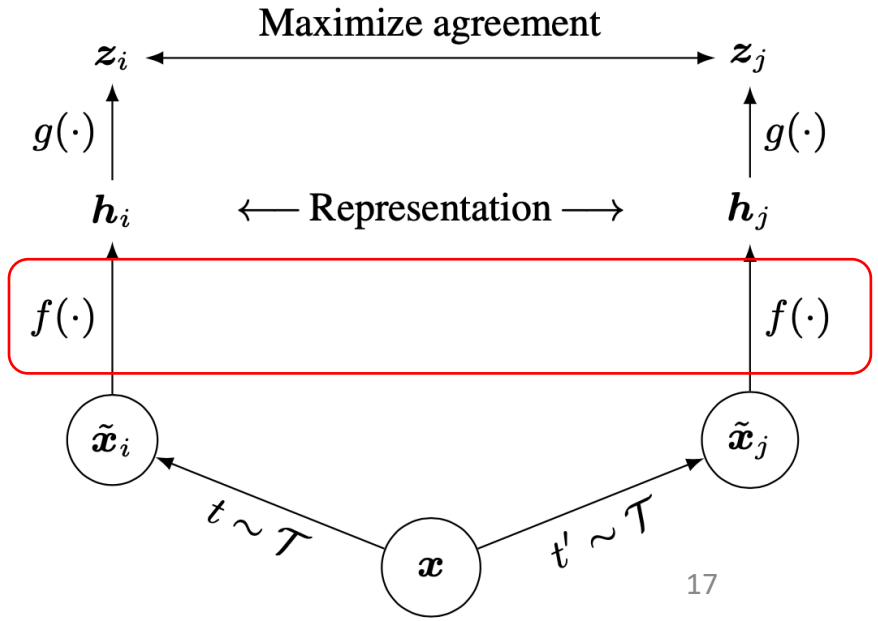
Base Encoder

Extracts representation vectors from augmented data examples

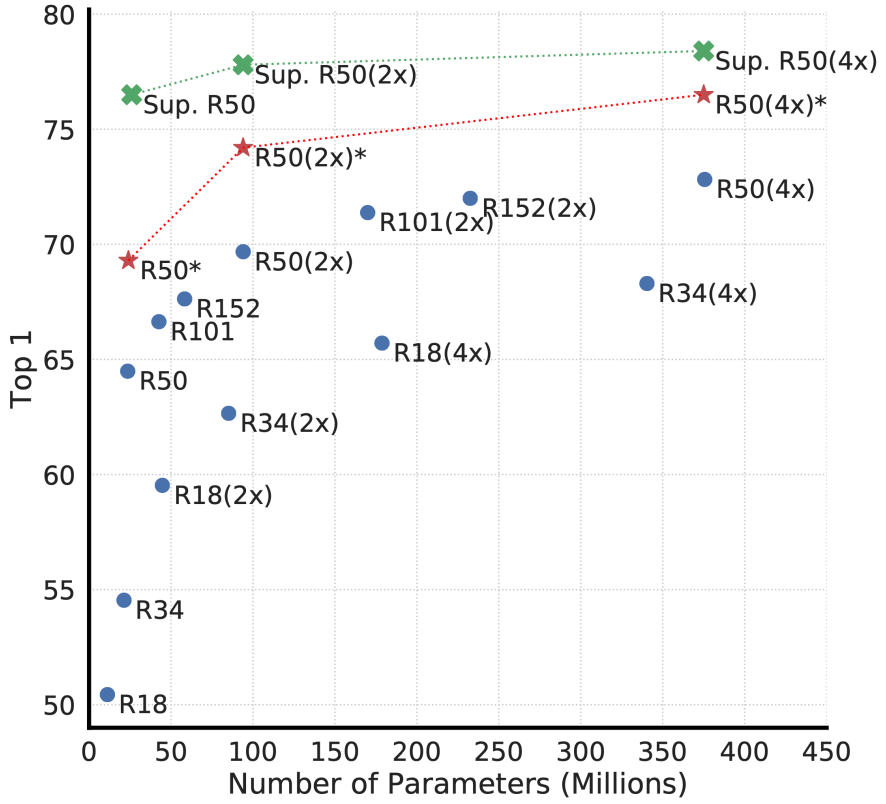
Framework allows various choices of the network architecture

SimCLR chooses ResNet

$$\boldsymbol{h}_i = f(\tilde{\boldsymbol{x}}_i) = ResNet(\tilde{\boldsymbol{x}}_i)$$



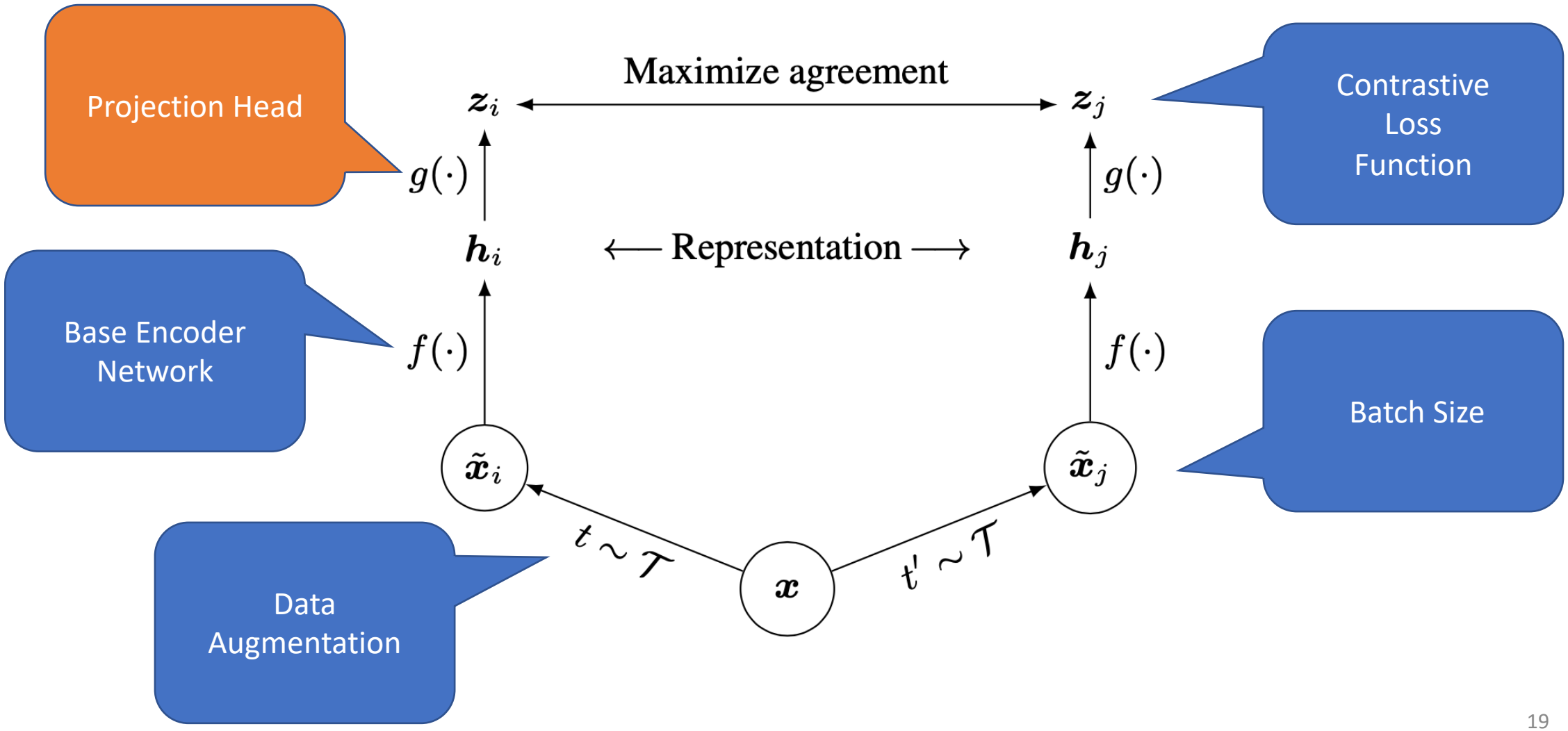
Base Encoder



Performance gap shrinks as model size increases

Unsupervised learning benefits more from bigger models

Framework


Projection Head
Base Encoder Network
Data Augmentation
Contrastive Loss Function
Batch Size
Maximize agreement
z_i
z_j
$g(\cdot)$
$g(\cdot)$
h_i
Representation
h_j
$f(\cdot)$
$f(\cdot)$
$\tilde{x}_i$
$\tilde{x}_j$
$t \sim \mathcal{T}$
x
$t' \sim \mathcal{T}$

Projection Head

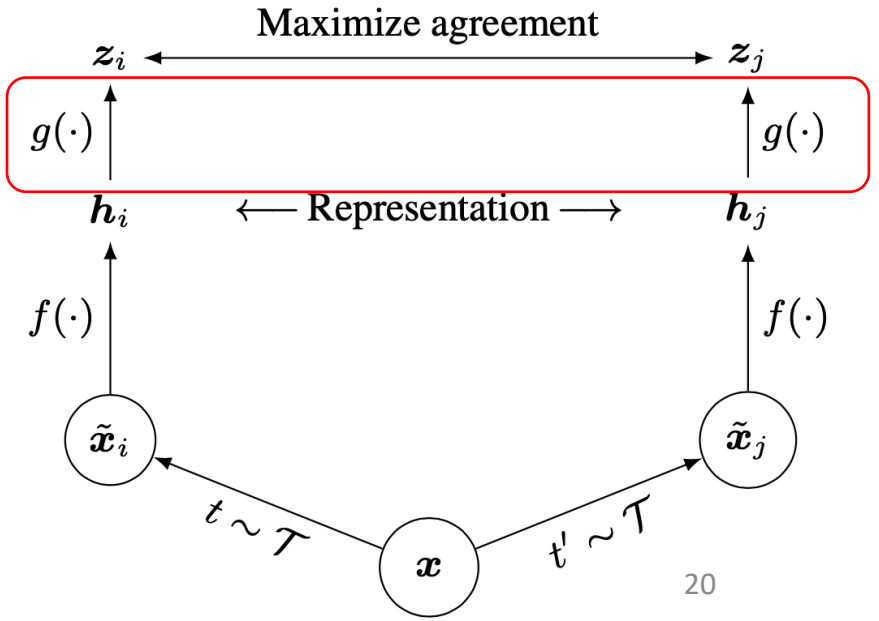
Maps representations to the space where contrastive loss is applied

A small neural network

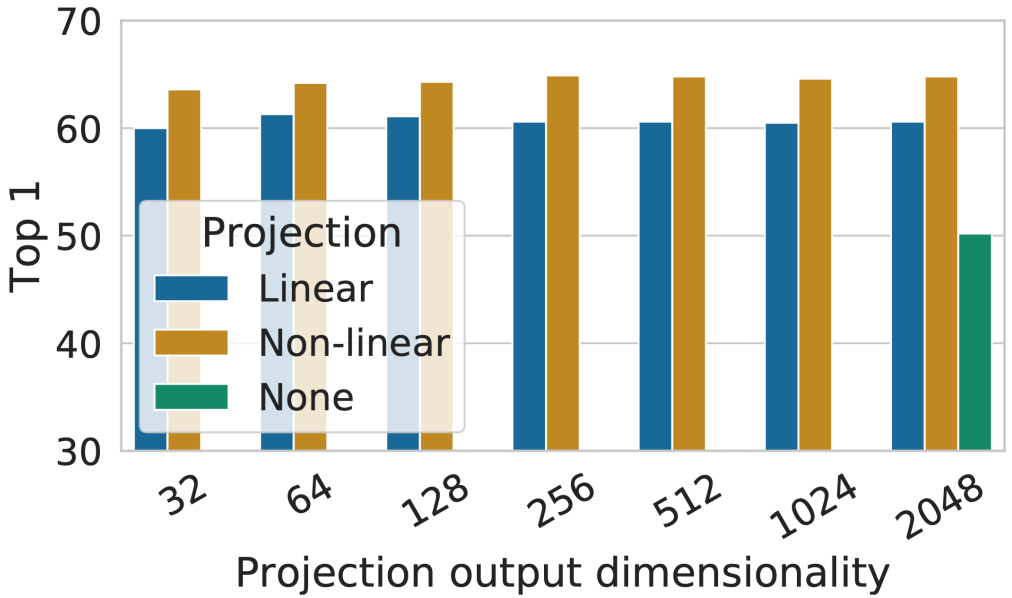
Multilayer Perceptron (MLP)

$\boldsymbol{z}_i = g(\boldsymbol{h}_i) = W^{(2)}\sigma(W^{(1)}\boldsymbol{h}_i)$

σ is ReLU (non-linearity)



Projection Head



Non-Linear > Linear >> None

h vs g(h)

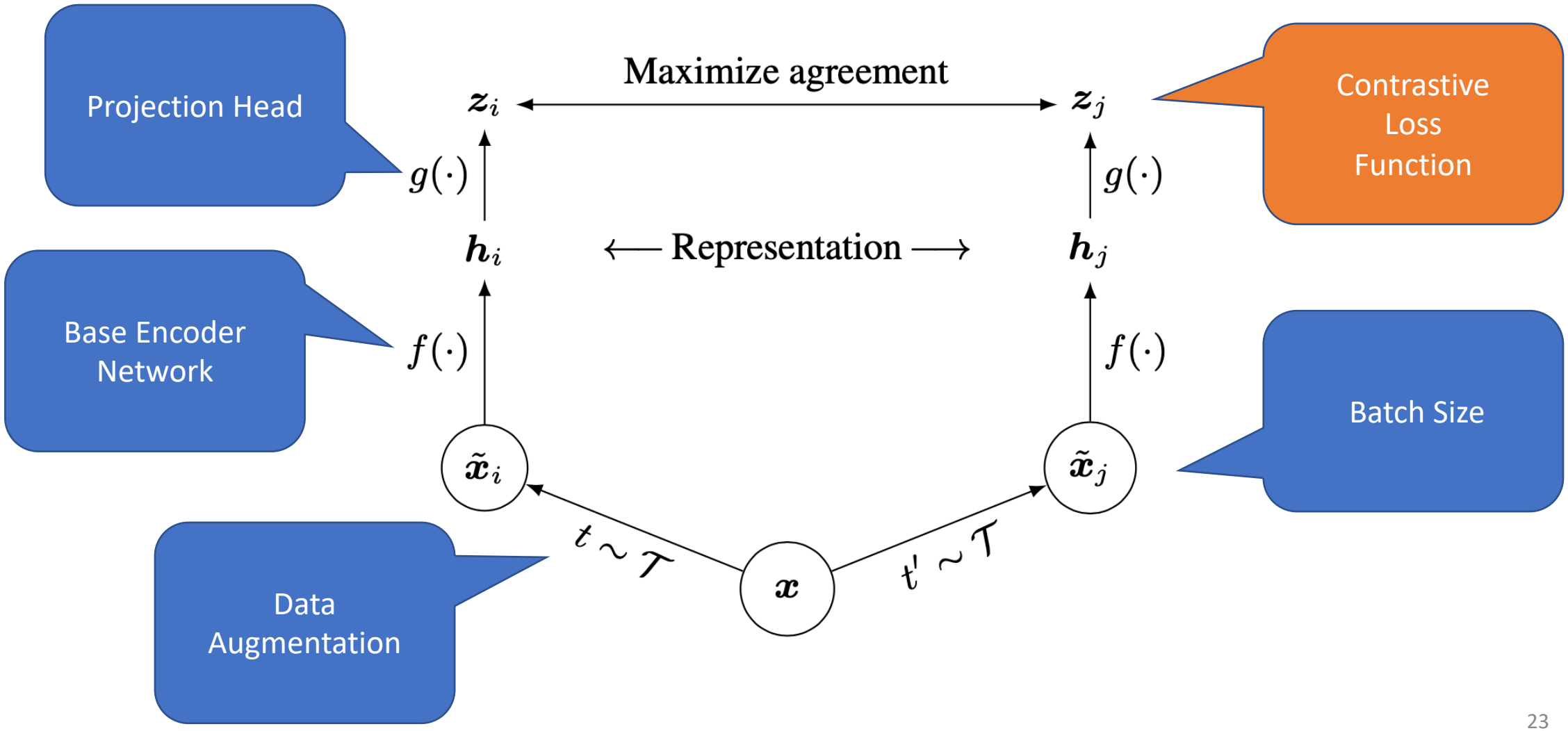
What to predict?	Random guess	Representation $\boldsymbol{h}$	Representation $g(\boldsymbol{h})$
Color vs grayscale	80	99.3	97.4
Rotation	25	67.6	25.6
Orig. vs corrupted	50	99.5	59.6
Orig. vs Sobel filtered	50	96.6	56.3

h > g(h)

Conjecture: due to loss of information induced by the contrastive loss

g can remove information that may be useful for the downstream task

Framework


Projection Head
Base Encoder Network
Data Augmentation
Contrastive Loss Function
Batch Size
Maximize agreement
z_i
z_j
$g(\cdot)$
$g(\cdot)$
h_i
Representation
h_j
$f(\cdot)$
$f(\cdot)$
$\tilde{x}_i$
$\tilde{x}_j$
$t \sim \mathcal{T}$
x
$t' \sim \mathcal{T}$

Contrastive Loss Function

Given a set $\{\tilde{\boldsymbol{x}}_k\}$, the contrastive prediction task aims to

Identify $\tilde{\boldsymbol{x}}_j$ in $\{\tilde{\boldsymbol{x}}_k\}_{k\neq i}$ for a given $\tilde{\boldsymbol{x}}_i$

Randomly sample a minibatch of N examples

Define the task on pairs of augmented examples (2N images)

Pick out one pair (2) as positives

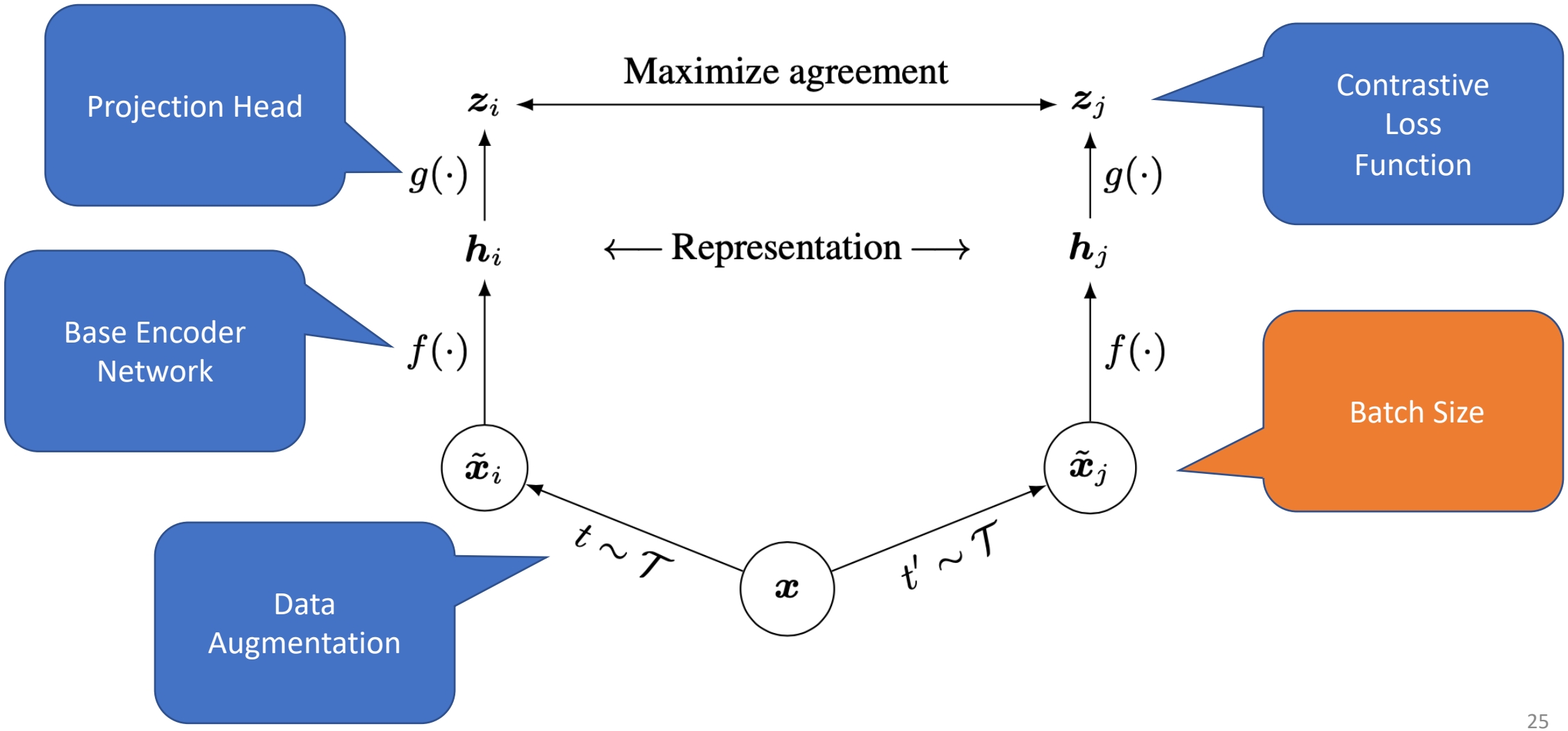
Treat the other 2(N − 1) augmented examples as negatives

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

NT-Xent (the normalized temperature-scaled cross entropy loss)

SimCLR computes the loss from all positive pairs in a mini-batch

Framework

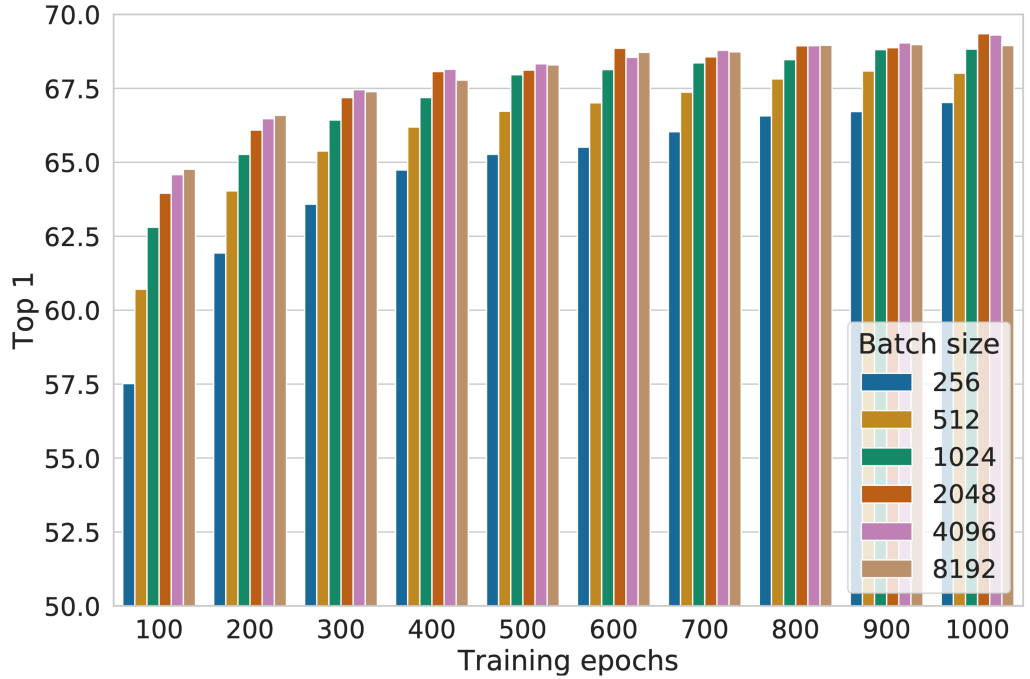

Projection Head
Base Encoder Network
Data Augmentation
Contrastive Loss Function
Batch Size
Maximize agreement
z_i
z_j
$g(\cdot)$
$g(\cdot)$
h_i
Representation
h_j
$f(\cdot)$
$f(\cdot)$
$\tilde{x}_i$
$\tilde{x}_j$
$t \sim \mathcal{T}$
x
$t' \sim \mathcal{T}$

Batch Size

Vary the training batch size N from 256 to 8192

Use the LARS optimizer (You et al., 2017)

Batch Size vs Training



Contrastive learning benefits from larger batch sizes and longer training

Algorithm 1 SimCLR's main learning algorithm.

input: batch size N, constant τ, structure of f, g, $\mathcal{T}$.
for sampled minibatch $\{\boldsymbol{x}_k\}_{k=1}^{N}$ **do**
  for all $k \in \{1, \ldots, N\}$ **do**
    draw two augmentation functions $t \sim \mathcal{T}$, $t' \sim \mathcal{T}$
    # the first augmentation
    $\tilde{\boldsymbol{x}}_{2k-1} = t(\boldsymbol{x}_k)$
    $\boldsymbol{h}_{2k-1} = f(\tilde{\boldsymbol{x}}_{2k-1})$ # representation
    $\boldsymbol{z}_{2k-1} = g(\boldsymbol{h}_{2k-1})$ # projection
    # the second augmentation
    $\tilde{\boldsymbol{x}}_{2k} = t'(\boldsymbol{x}_k)$
    $\boldsymbol{h}_{2k} = f(\tilde{\boldsymbol{x}}_{2k})$ # representation
    $\boldsymbol{z}_{2k} = g(\boldsymbol{h}_{2k})$ # projection
  end for
  for all $i \in \{1, \ldots, 2N\}$ and $j \in \{1, \ldots, 2N\}$ **do**
    $s_{i,j} = \boldsymbol{z}_i^\top \boldsymbol{z}_j / (\|\boldsymbol{z}_i\| \|\boldsymbol{z}_j\|)$ # pairwise similarity
  end for
  define $\ell(i, j)$ **as** $\ell(i, j) = -\log \frac{\exp(s_{i,j}/\tau)}{\sum_{k=1}^{2N} \mathbb{1}_{[k \neq i]} \exp(s_{i,k}/\tau)}$
  $\mathcal{L} = \frac{1}{2N} \sum_{k=1}^{N} [\ell(2k-1, 2k) + \ell(2k, 2k-1)]$
  update networks f and g to minimize $\mathcal{L}$
end for
return encoder network $f(\cdot)$, and throw away $g(\cdot)$

Outline

Motivation

Framework

Evaluation

Conclusion

Evaluation Protocol

Dataset: ImageNet ILSVRC-2012

- Also evaluated on CIFAR-10 and others (for transfer learning)

Protocol: linear evaluation

- A linear classifier is trained on top of the frozen base network
- Test accuracy is used as a proxy for representation quality

Data Augmentation: crop & resize, color distortion, and Gaussian blur

Optimizer: LARS with LR=4.8

Batch size: 4096

Epochs: 100

Comparison with State-of-the-art

Method	Architecture	Param (M)	Top 1	Top 5
Methods using ResNet-50:				
Local Agg.	ResNet-50	24	60.2	-
MoCo	ResNet-50	24	60.6	-
PIRL	ResNet-50	24	63.6	-
CPC v2	ResNet-50	24	63.8	85.3
SimCLR (ours)	ResNet-50	24	**69.3**	**89.0**
Methods using other architectures:				
Rotation	RevNet-50 ($4\times$)	86	55.4	-
BigBiGAN	RevNet-50 ($4\times$)	86	61.3	81.9
AMDIM	Custom-ResNet	626	68.1	-
CMC	ResNet-50 ($2\times$)	188	68.4	88.2
MoCo	ResNet-50 ($4\times$)	375	68.6	-
CPC v2	ResNet-161 ($*$)	305	71.5	90.1
SimCLR (ours)	ResNet-50 ($2\times$)	94	74.2	92.0
SimCLR (ours)	ResNet-50 ($4\times$)	375	**76.5**	**93.2**

Semi-Supervised Learning

Sample 1% or 10% of the labelled ImageNet training datasets

- Class-balanced
- 12.8 and 128 images per class respectively

Semi-Supervised Learning

Method	Architecture	Label fraction (Top 5) 1%	10%
Supervised baseline	ResNet-50	48.4	80.4
Methods using other label-propagation:			
Pseudo-label	ResNet-50	51.6	82.4
VAT+Entropy Min.	ResNet-50	47.0	83.4
UDA (w. RandAug)	ResNet-50	-	88.5
FixMatch (w. RandAug)	ResNet-50	-	89.1
S4L (Rot+VAT+En. M.)	ResNet-50 (4×)	-	91.2
Methods using representation learning only:			
InstDisc	ResNet-50	39.2	77.4
BigBiGAN	RevNet-50 (4×)	55.2	78.8
PIRL	ResNet-50	57.2	83.8
CPC v2	ResNet-161(∗)	77.9	91.2
SimCLR (ours)	ResNet-50	75.5	87.8
SimCLR (ours)	ResNet-50 (2×)	83.0	91.2
SimCLR (ours)	ResNet-50 (4×)	**85.8**	**92.6**

Transfer Learning

	Food	CIFAR10	CIFAR100	Birdsnap	SUN397	Cars	Aircraft	VOC2007	DTD	Pets	Caltech-101	Flowers
Linear evaluation:												
SimCLR (ours)	**76.9**	**95.3**	80.2	48.4	**65.9**	60.0	61.2	**84.2**	**78.9**	89.2	**93.9**	**95.0**
Supervised	75.2	**95.7**	**81.2**	**56.4**	64.9	**68.8**	**63.8**	83.8	**78.7**	**92.3**	**94.1**	94.2
Fine-tuned:												
SimCLR (ours)	**89.4**	**98.6**	**89.0**	**78.2**	**68.1**	**92.1**	**87.0**	**86.6**	**77.8**	92.1	**94.1**	97.6
Supervised	88.7	98.3	**88.7**	**77.8**	67.0	91.4	**88.0**	86.5	**78.8**	**93.2**	**94.2**	**98.0**
Random init	88.3	96.0	81.9	**77.0**	53.7	91.3	84.8	69.4	64.1	82.7	72.5	92.5

Outline

Motivation

Framework

Evaluation

Conclusion

Conclusion

Composition of multiple data augmentation operations is crucial

Nonlinear transformation (g) substantially improves the quality

Larger batch sizes and more training steps bring more benefits

Quiz Questions

Which of the following statements are true about effective visual representation learning?

- Heuristics may limit generality of representations.
- Generative approaches train networks using pretext tasks with unlabeled data.
- Discriminative approaches are more widely used than generative approaches.
- Pixel-level generation is expensive in computation.

Which of the following statements are true about SimCLR?

- Given batch size N, SimCLR's learning algorithm requires computing similarities between all pairs of 2N projections.
- SimCLR's learning algorithm computes contrastive loss across all positive and all negative pairs.
- The representation h is achieved after max pooling.
- It is beneficial to define loss on z instead of h due to a nonlinear projection head can improve the representation quality of the layer before it.

Which of the following statements are true on project heads?

- Nonlinear projection is better than linear projection.
- Nonlinear projection is worse than linear projection.
- Linear projection is better than no projection.
- Linear projection is worse than no projection.