이영준(M2023089)

# A Simple Framework for Contrastive Learning of Visual Representations

*Ting Chen, Simon Kornblith, Mohammad Norouzi, Geoffery Hinton*

ICML 2020

# 0. Index

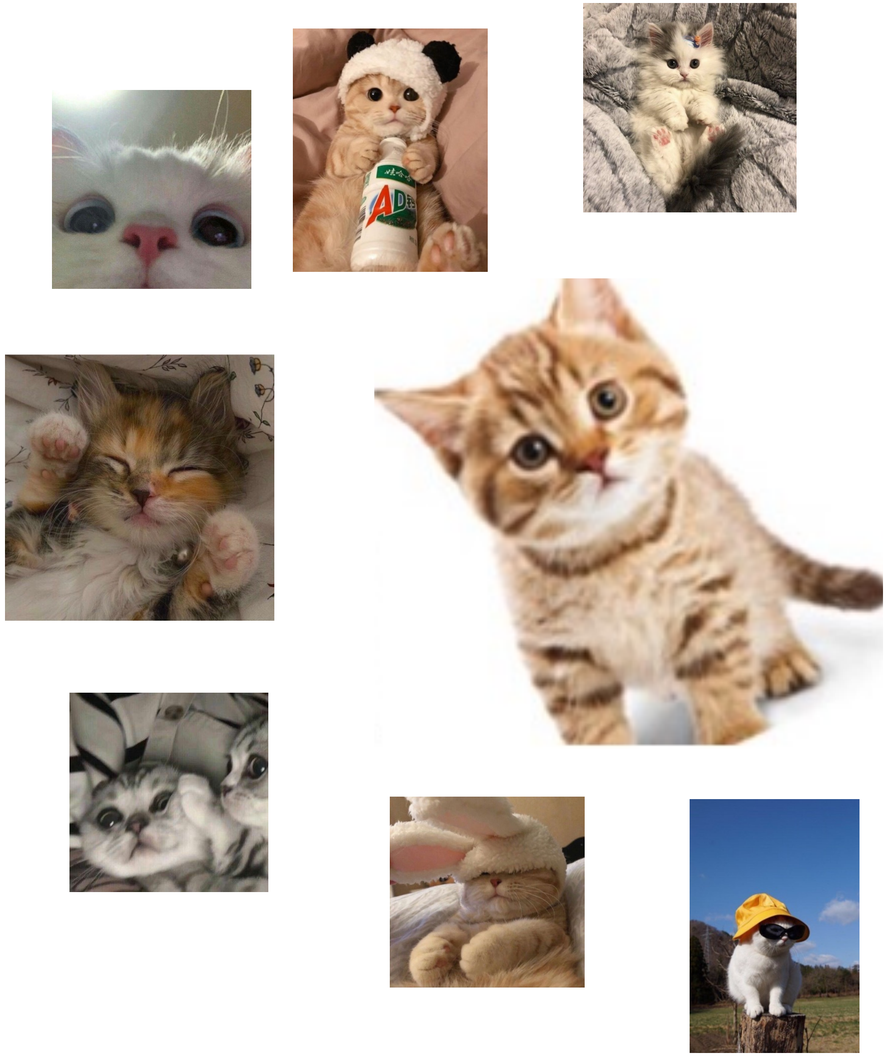
- Introduction
- Design
- Evaluation
- Conclusion

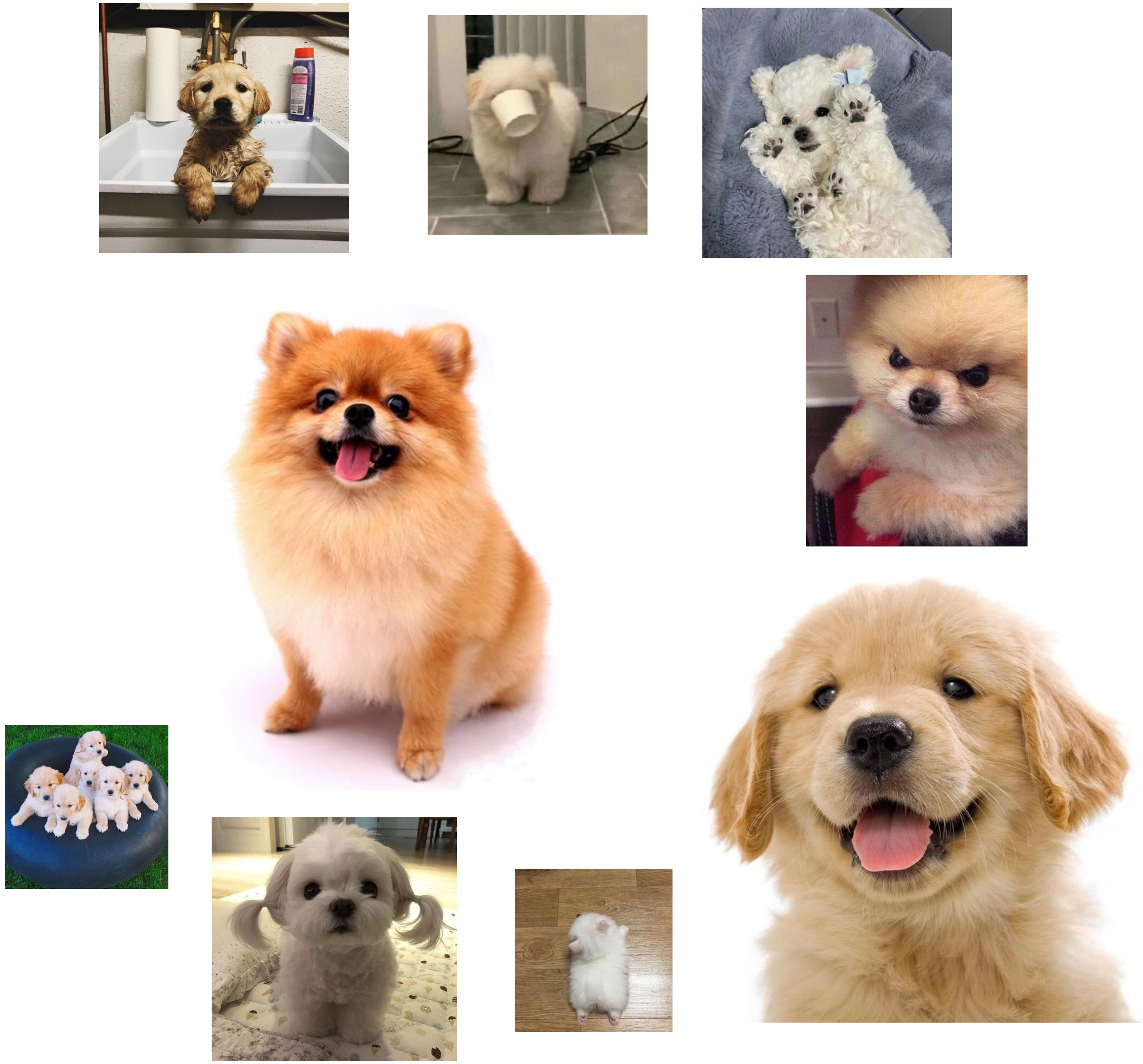
# 01. Introduction

# 01. Introduction : Representation Learning

How to Solve Cats VS Dogs problem?

# 01. Introduction : Representation Learning





# 01. Introduction : Representation Learning

# 01. Introduction : Representation Learning

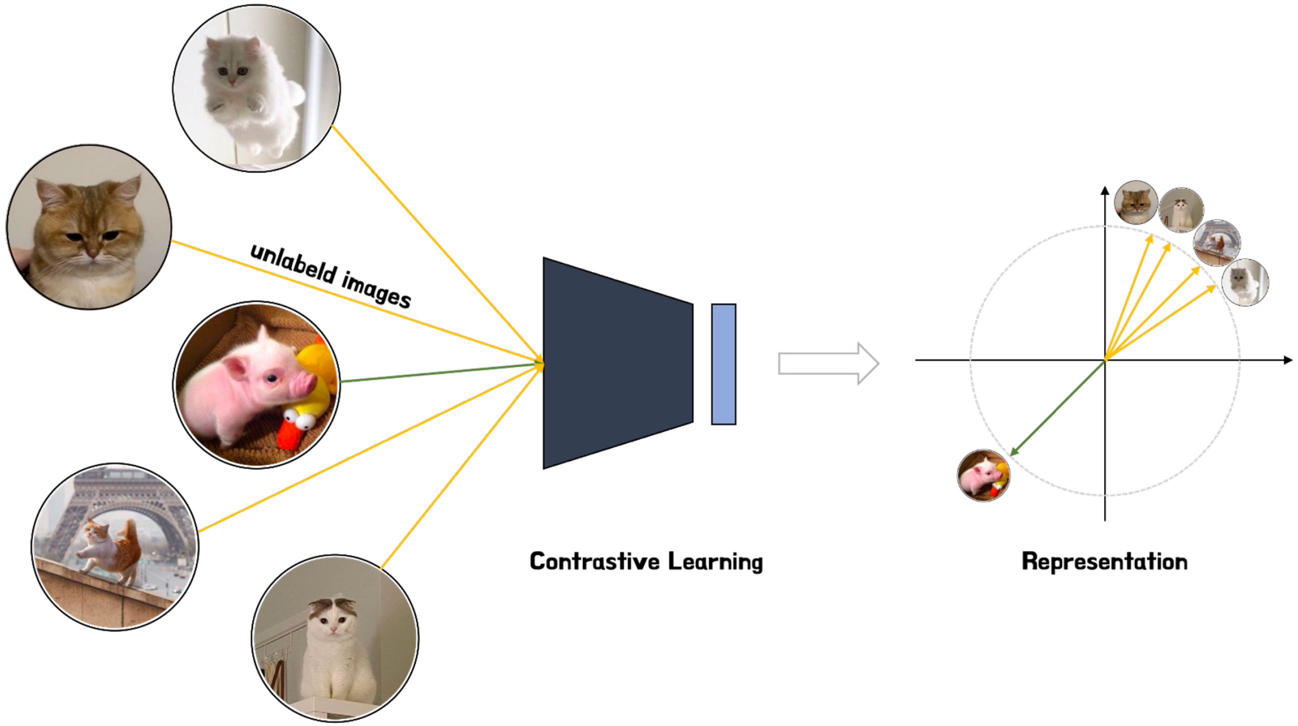
=

≠

# 01. Introduction : Contrastive Learning

## ◆ Contrastive Learning

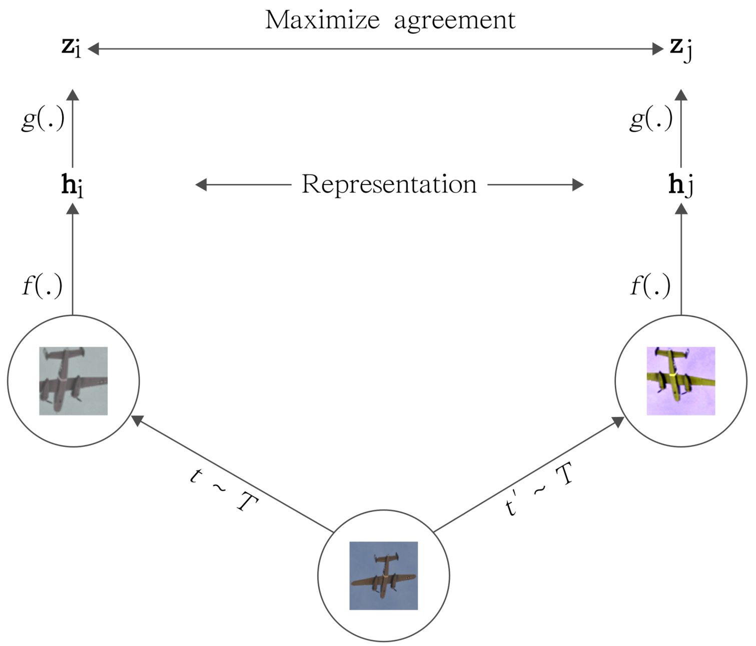
- 유사한 이미지가 저차원 공간에서 서로 가깝게, 다른 이미지는 서로 멀리 떨어져 있도록 저차원 공간에서 이미지를 인코딩하는 방법을 모델이 학습하는 것을 의미



# 01. Introduction : SimCLR

- A Simple Framework for Contrastive Learning of Visual Representations (SimCLR)
  - 비슷한 이미지끼리는 이미지를 더 가까운 벡터로,
  - 다른 이미지와는 더 먼 벡터로 만들어 주는 모델

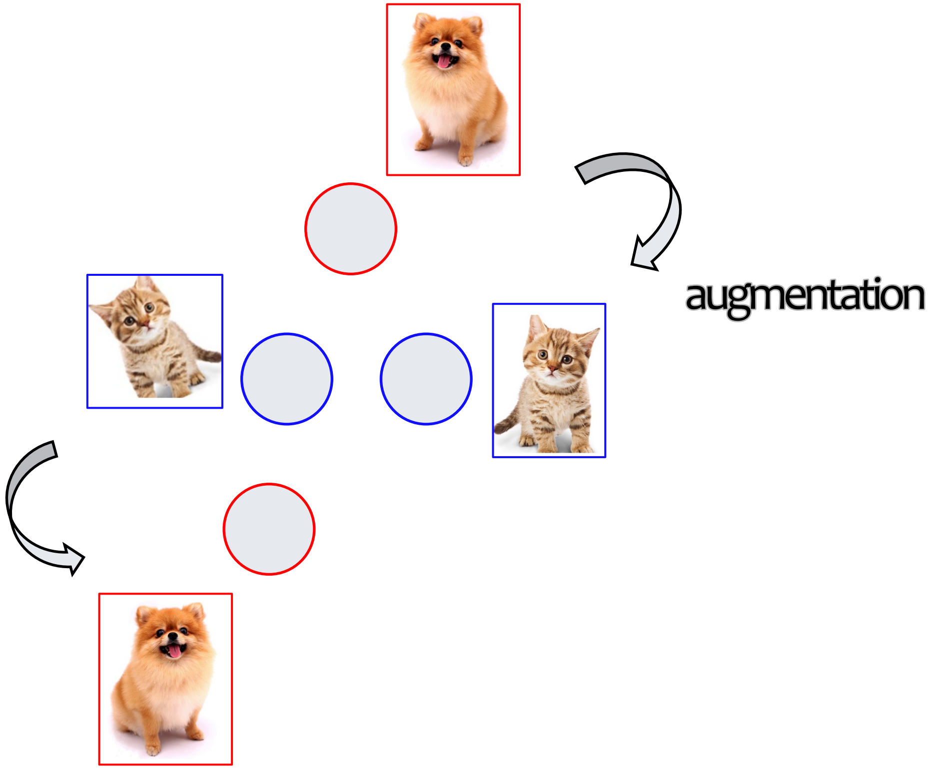
- 연구 내용
  - Data augmentation을 활용한 representation learning
  - Nonlinear transformation을 사용한 contrastive loss 설정
  - Batch size와 contrastive learning의 관계



# 02. Design

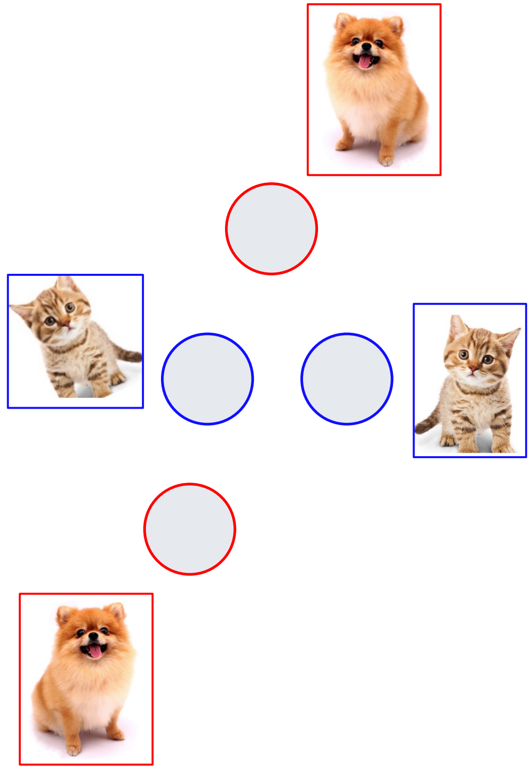
# 02. Design: Overview

- Data augmentaion을 활용한 contrastive learning
  - Positive(cat) samples
  - Negative(dog) samples



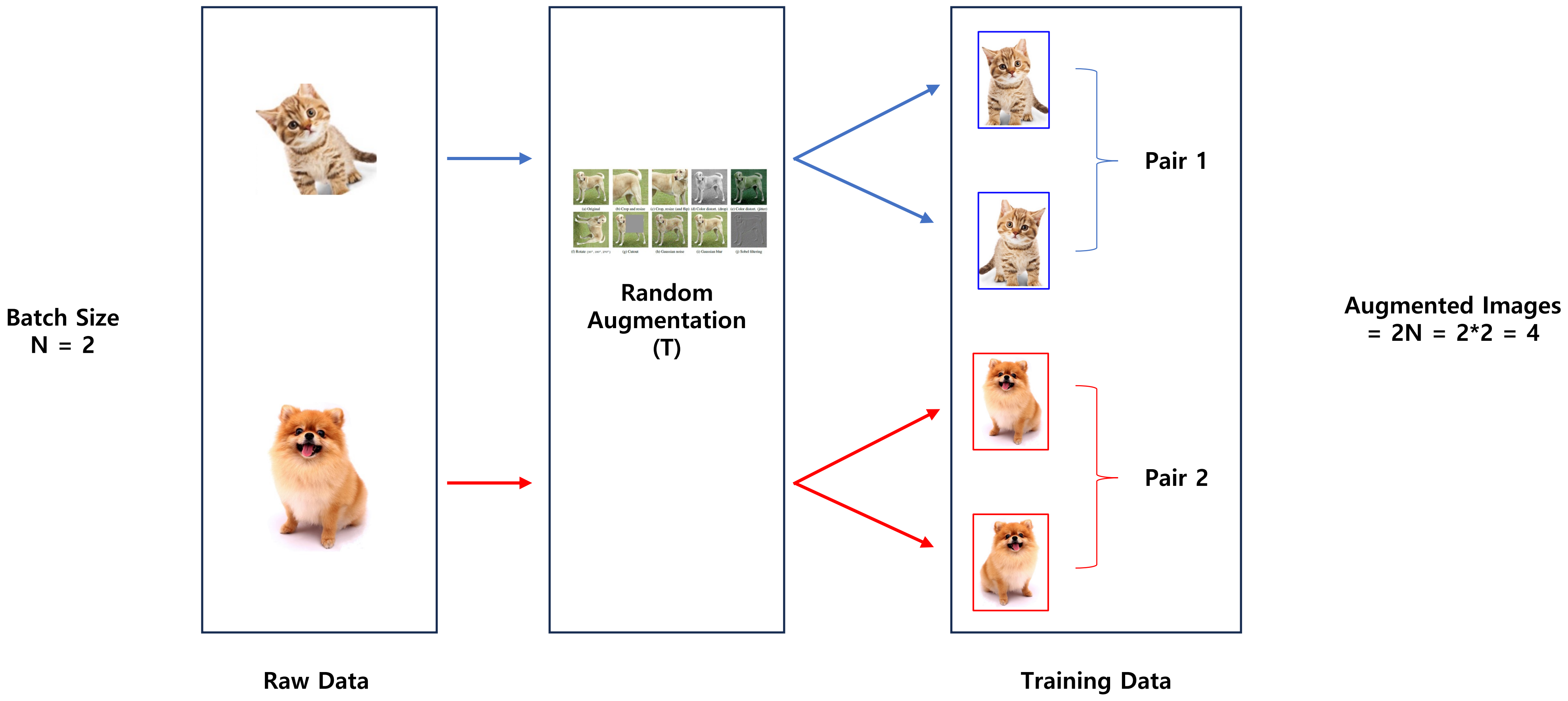
# 02. Design: Overview

- Data augmentaion을 활용한 contrastive learning
  - Positive(cat) samples
  - Negative(dog) samples

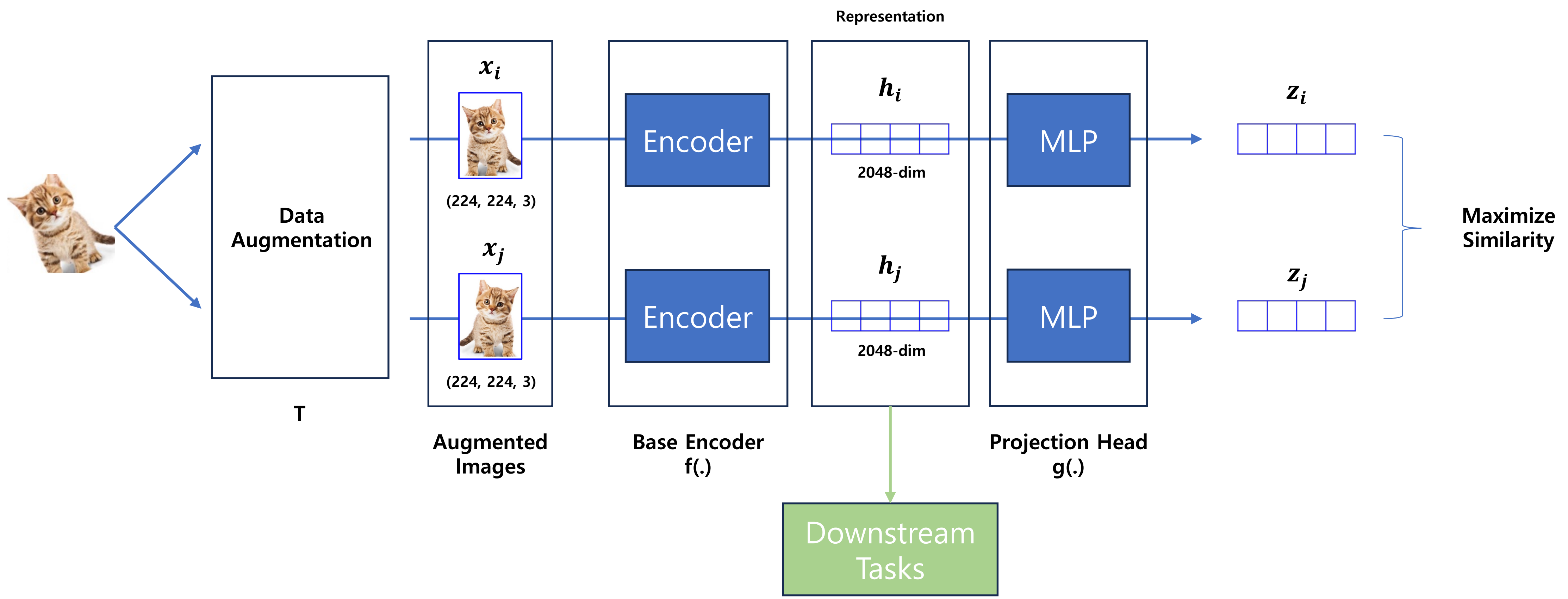


# 04. Design: Overview

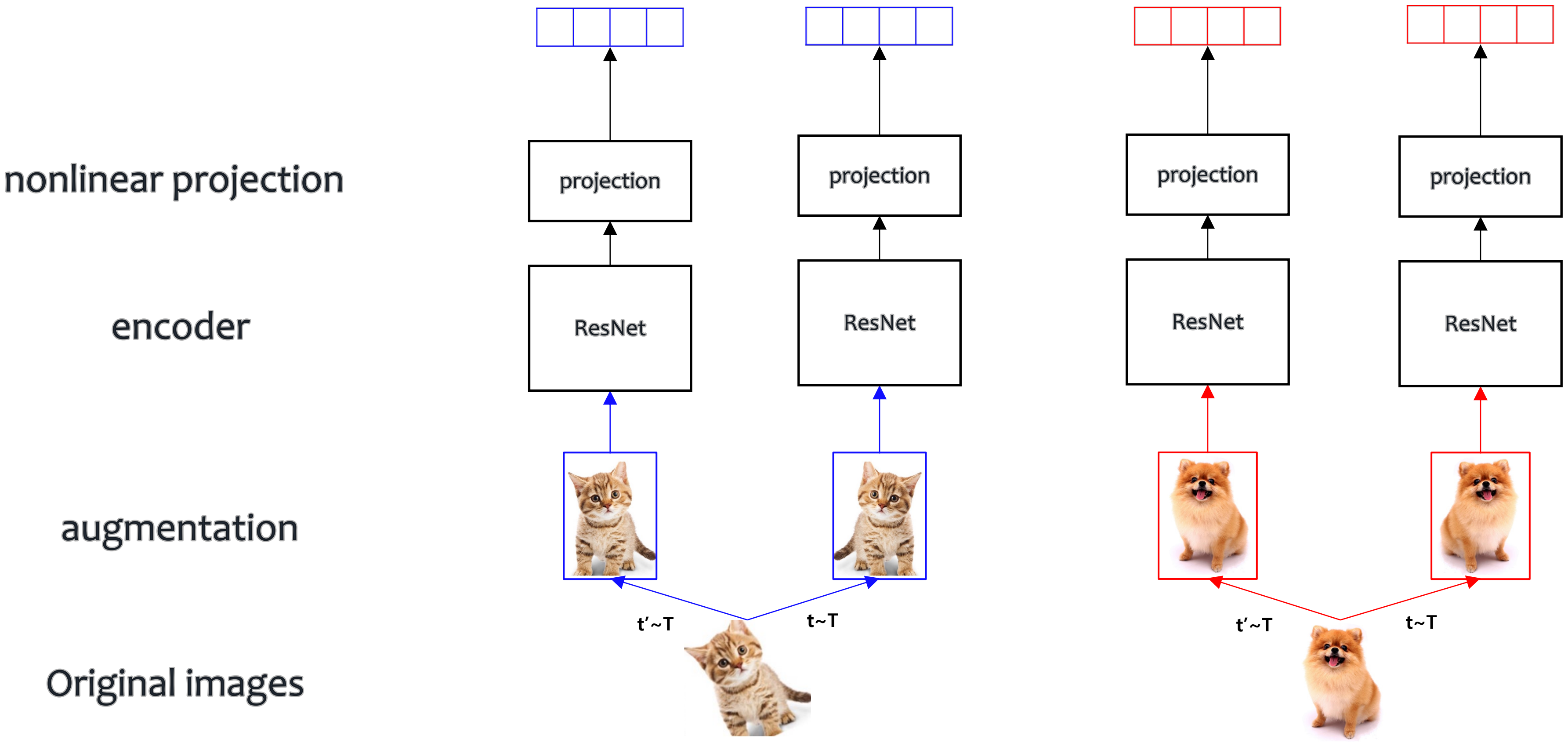
# 02. Design: Data Augmentation



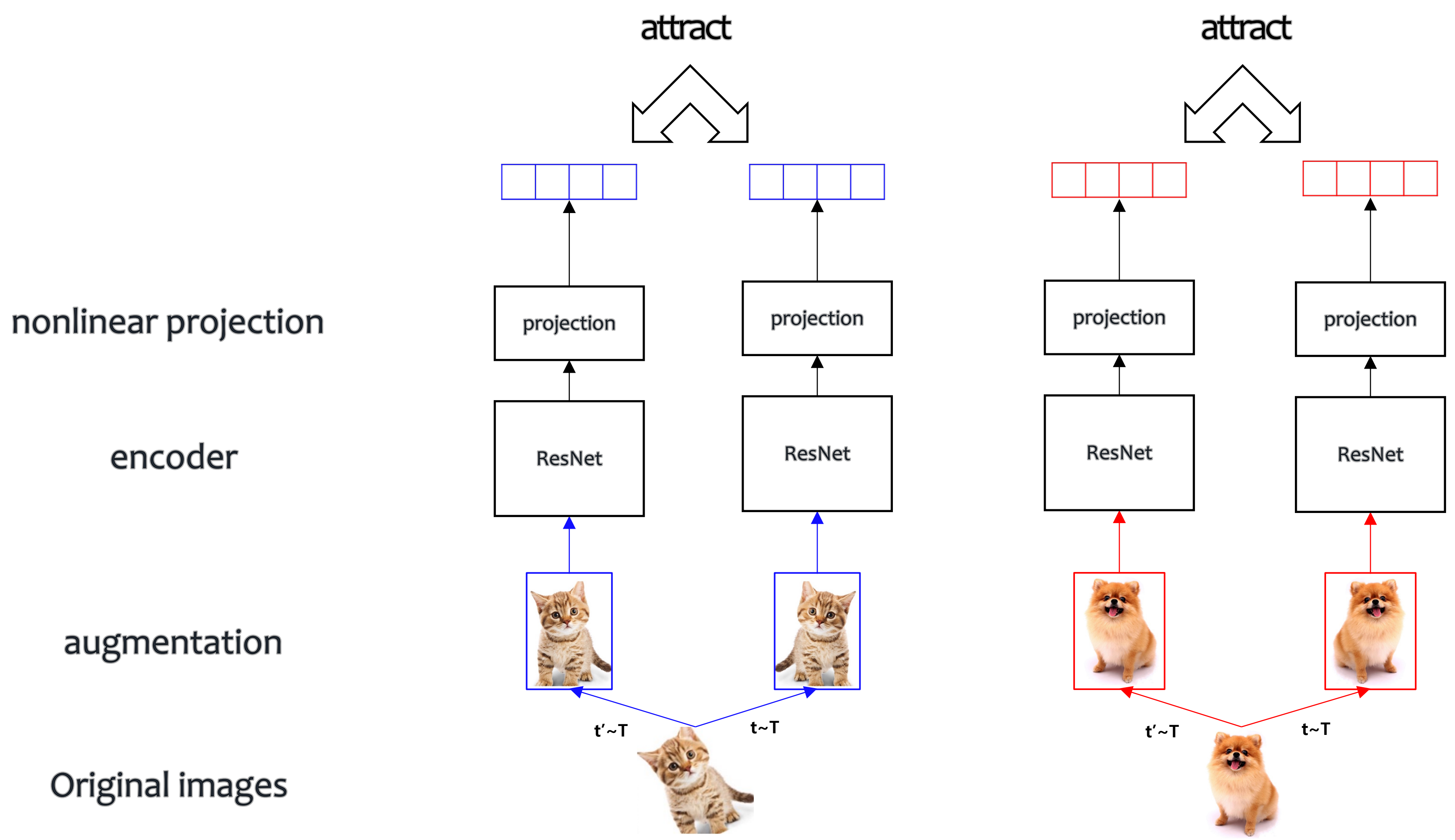
# 02. Design: Model Structure



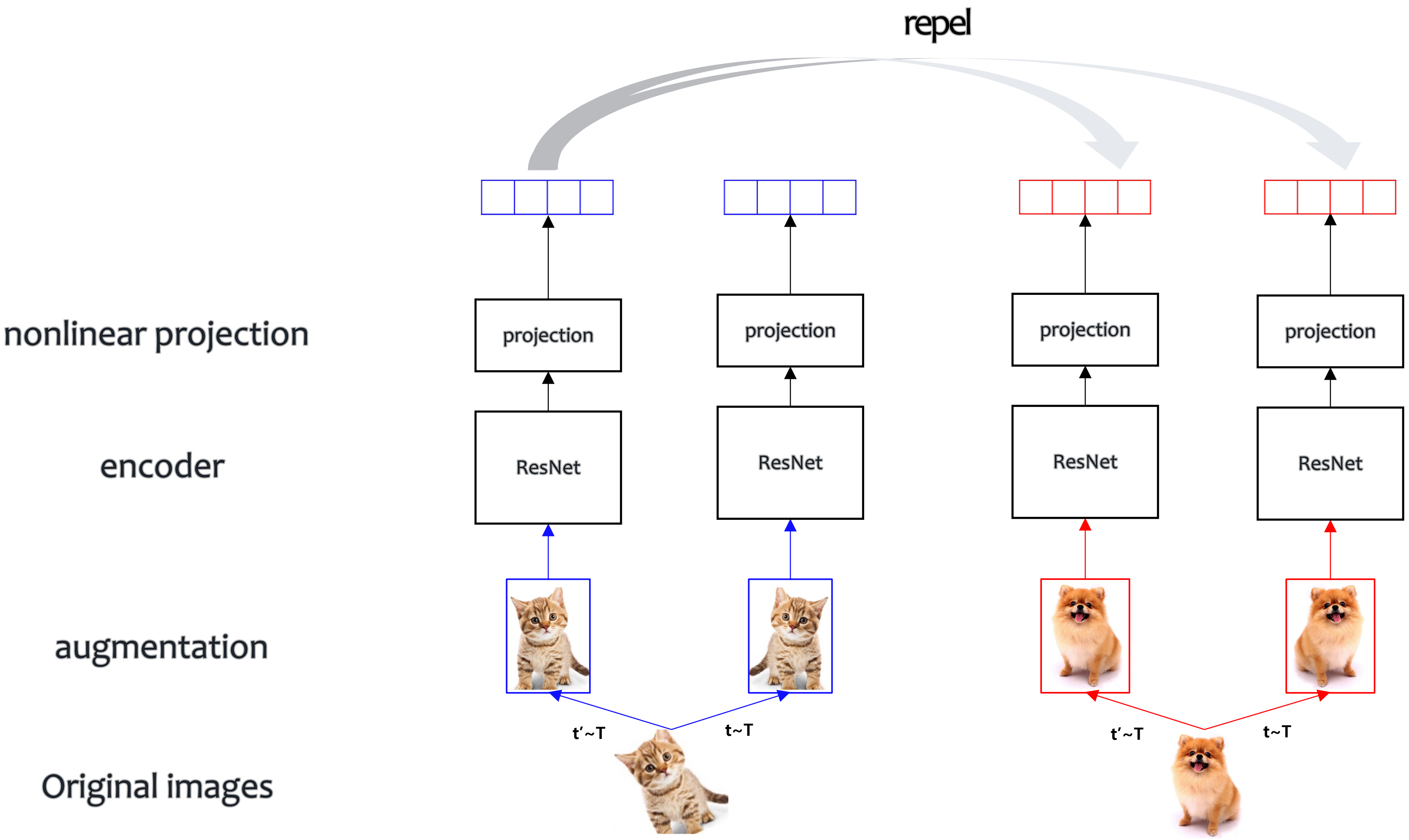
# 02. Design: Model Structure



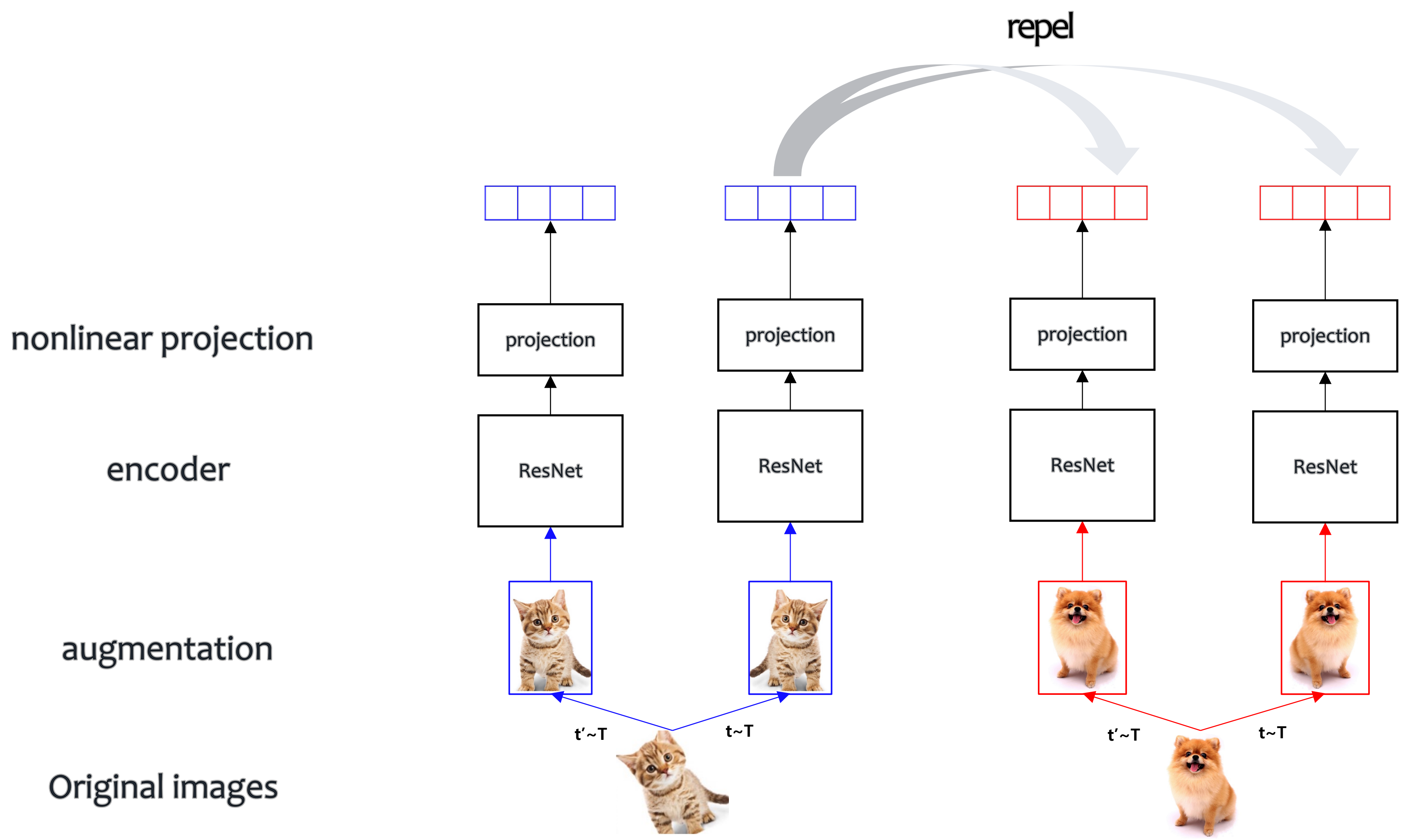
# 02. Design: Model Structure



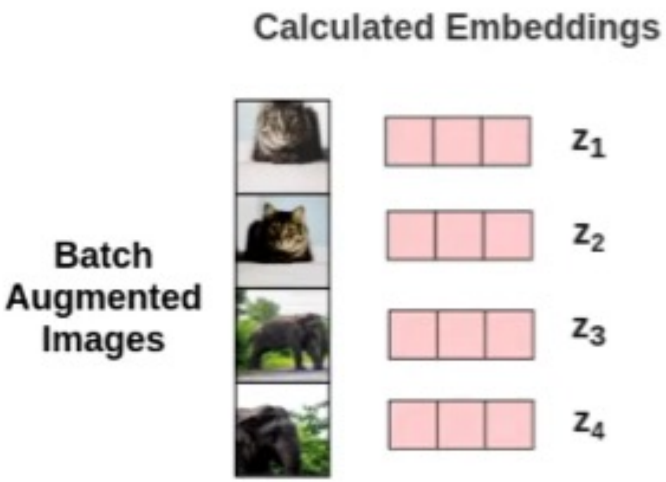
# 02. Design: Model Structure



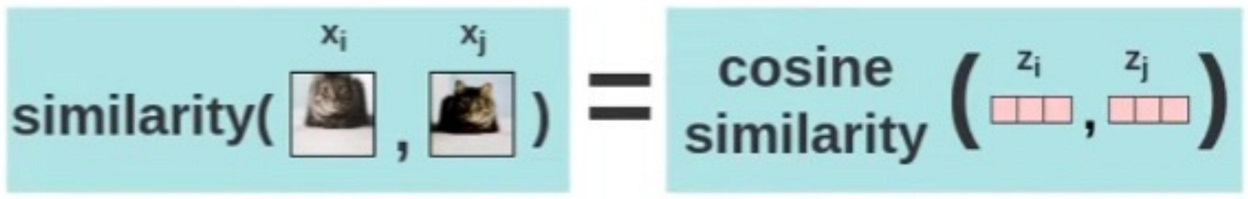
# 02. Design: Model Structure



# 02. Design: Loss Function

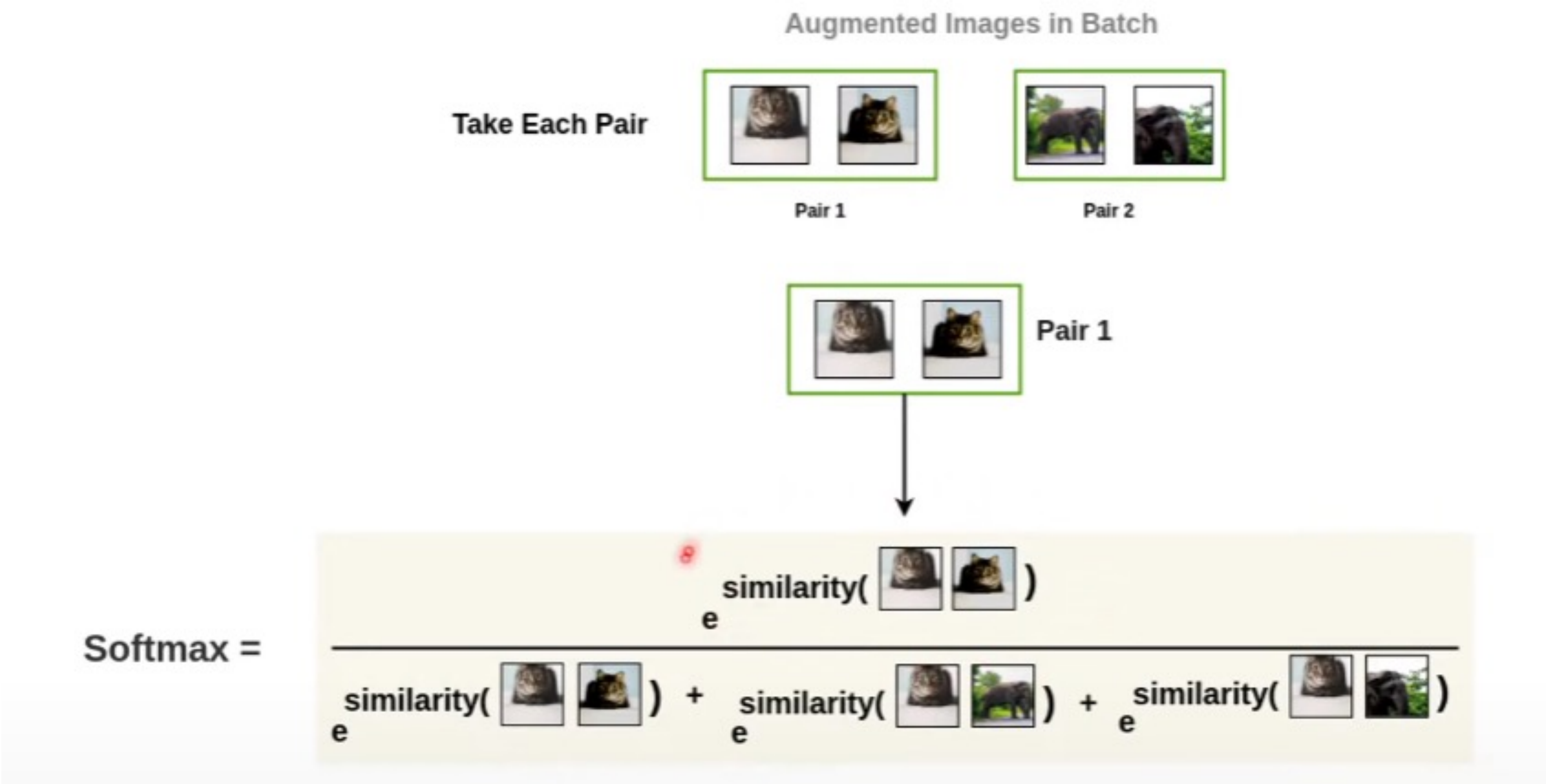



**Similarity Calculation of Augmented Images**




$$s_{i,j} = \frac{z_i^T z_j}{(\tau ||z_i|| ||z_j||)}$$

# 02. Design: Loss Function

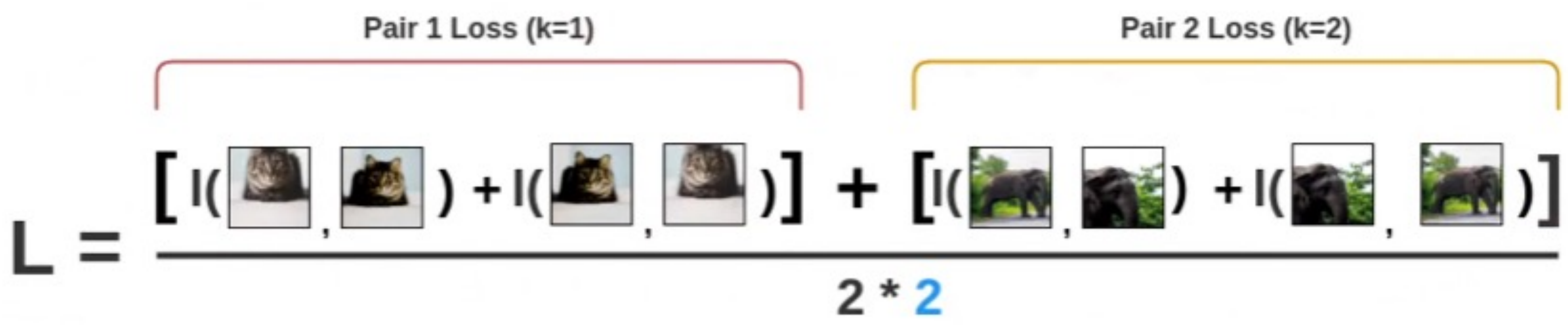


# 02. Design: Loss Function

$$l(i,j) = -log\frac{exp(s_{i,j})}{\sum_{k=1}^{2N} l_{[k!=i]} exp(s_{i,k})}$$

# 02. Design: Loss Function

$$L = \frac{1}{2N} \sum_{k=1}^{N} [l(2k-1, 2k) + l(2k, 2k-1)]$$



# 03. Evaluation

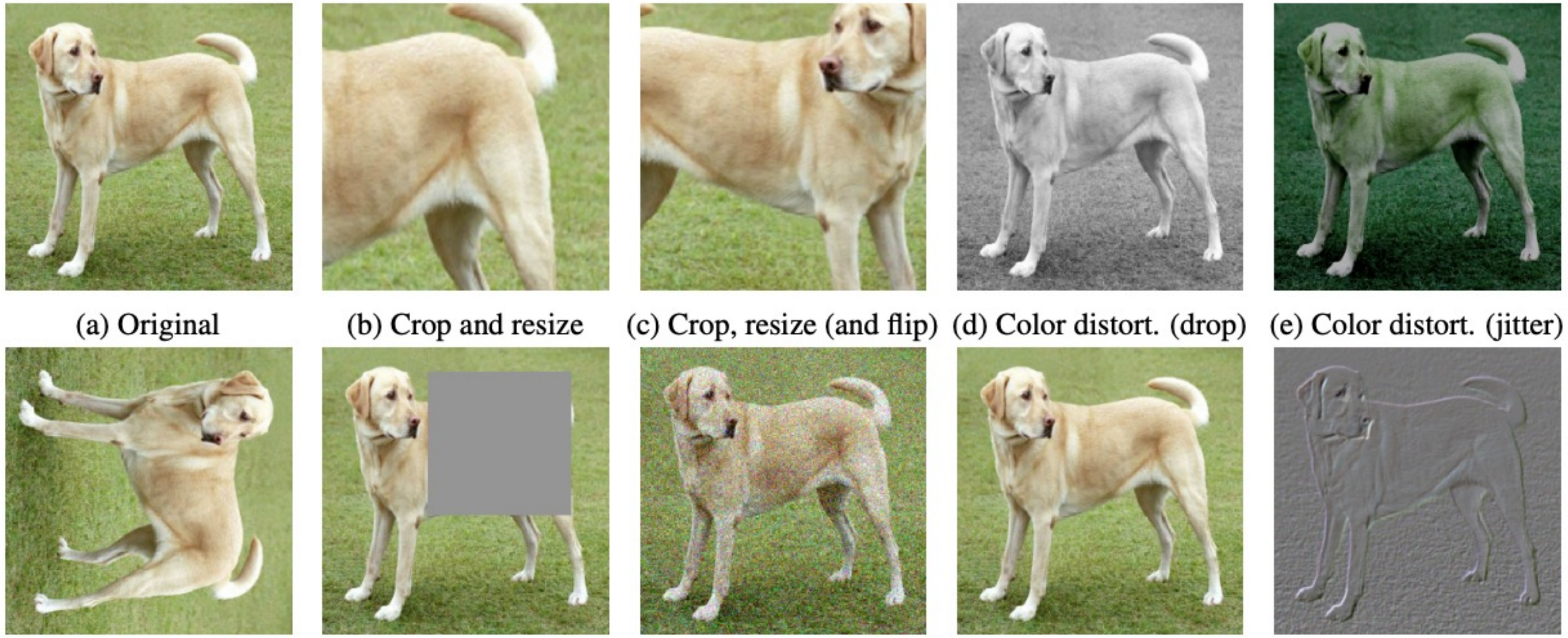
# 03. Evaluation

**Parameter and Dataset**

- Training :
  - Batch size : 256 ~ 8192 (Default 값 : 4096)
  - Large Dataset에 적합한 LARS optimizer를 사용함
- Dataset
  - ImageNet 2012

# 03. Evaluation

- Data Augmentation



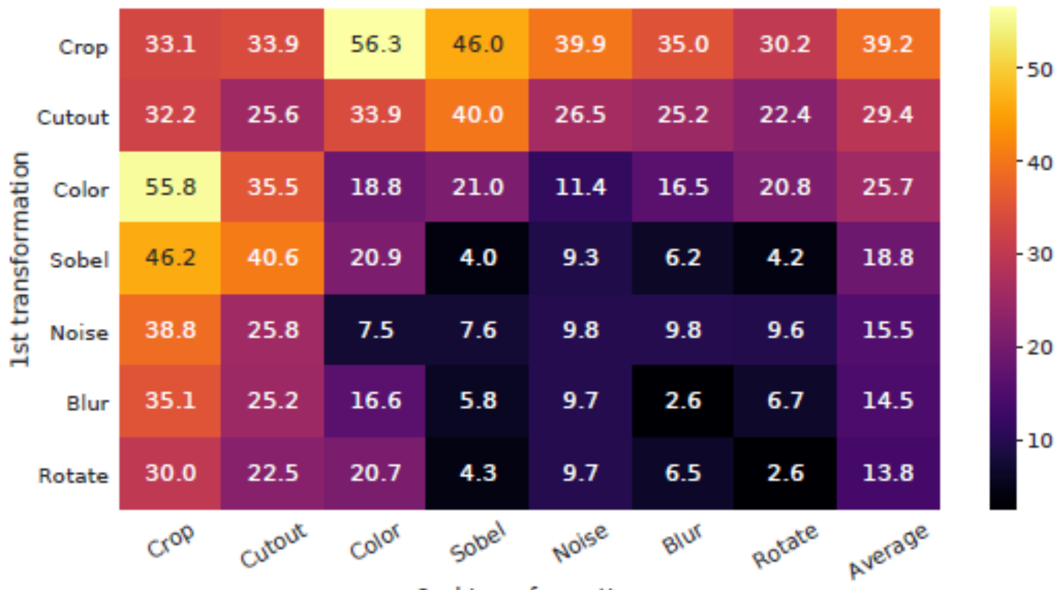
(a) Original (b) Crop and resize (c) Crop, resize (and flip) (d) Color distort. (drop) (e) Color distort. (jitter)

(f) Rotate $\{90°, 180°, 270°\}$ (g) Cutout (h) Gaussian noise (i) Gaussian blur (j) Sobel filtering

*Figure 4.* Illustrations of the studied data augmentation operators. Each augmentation can transform data stochastically with some internal parameters (e.g. rotation degree, noise level). Note that we *only* test these operators in ablation, the *augmentation policy used to train our models* only includes *random crop (with flip and resize)*, *color distortion*, and *Gaussian blur*. (Original image cc-by: Von.grzanka)

# 03. Evaluation

- Data Augmentation
  - Augmentation 1개일 때 Cropping
  - Augmentation 2개일 때 Cropping + coloring이 가장 좋은 결과를 내는 기법임을 알 수 있음



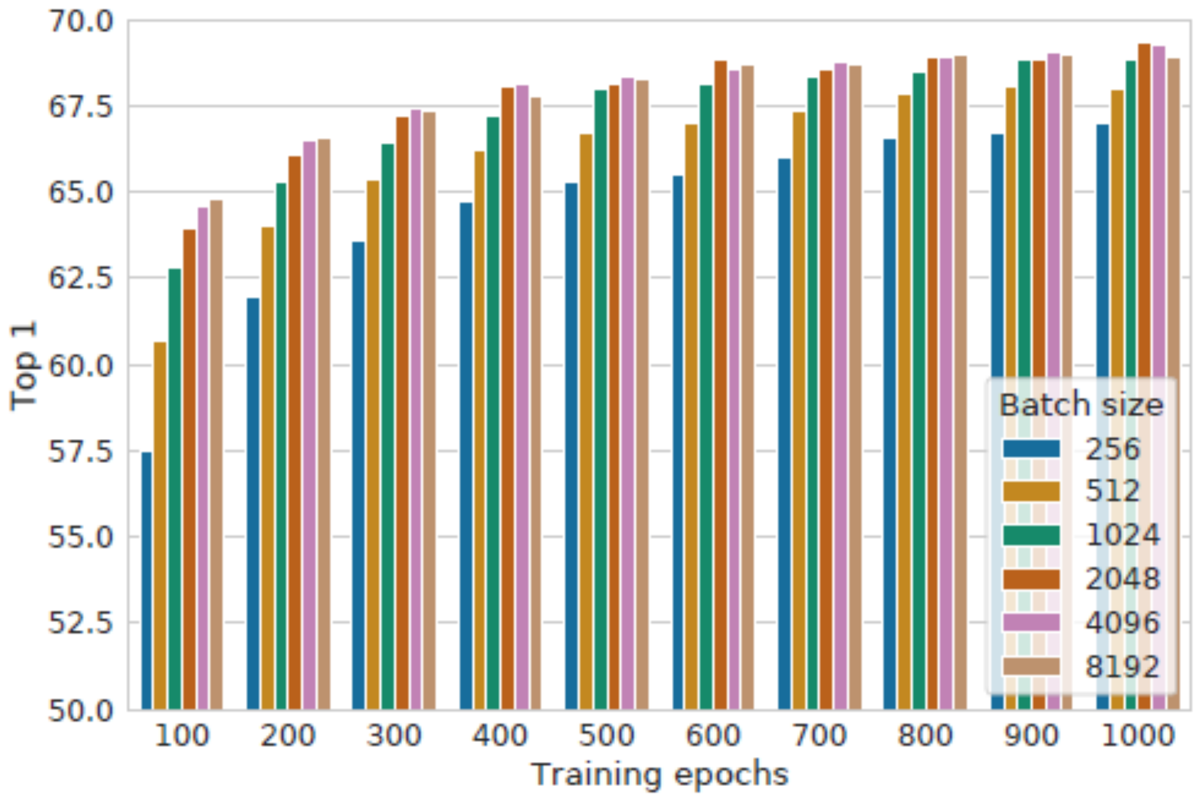
# 03. Evaluation

- Data Augmentation : Color Distortion
  - Supervised learning에서 Color augmentation을 강하게 주면 성능이 떨어지는 경향이 있음
  - SimCLR의 경우 Color augmentaion을 강하게 줄수록 점진적인 성능향상을 보임
  - 데이터가 많아지면 더 성능이 비례적으로 좋아질 것임

| Methods | Color distortion strength | | | | | AutoAug |
|---|---|---|---|---|---|---|
| | 1/8 | 1/4 | 1/2 | 1 | 1 (+Blur) | |
| 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 |

# 03. Evaluation

- Batch Size
  - Batch Size가 클수록, 학습 기간이 길수록 더 좋은 성능을 냄



# 03. Evaluation

- Comparison with supervised learning
  - SimCLR은 supervised learning에 못지 않게 좋은 성능을 낼 수 있음

| | 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 |

# 04. Conclusion

# 04. Conclusion

- Contrastive Learning을 통해 Visual Representation Embedding Vector에 대한 학습이 가능하게 하였음
- 이를 위한 간단한 Framework와 그 구현 방법을 제시하였음
- Framework Design 구성요소들을 활용하여 지도학습에 준하는 Downstream Task 성능을 얻음

Thank You
감사합니다