華東師範大學
EAST CHINA NORMAL UNIVERSITY

图像的各种无监督表示学习

吴倩

Multi-Agent Artificial Intelligence Laboratory
East China Normal University

基于变换的无监督训练

以**预测变换**作为自监督信号进行训练的模型

RotationNet

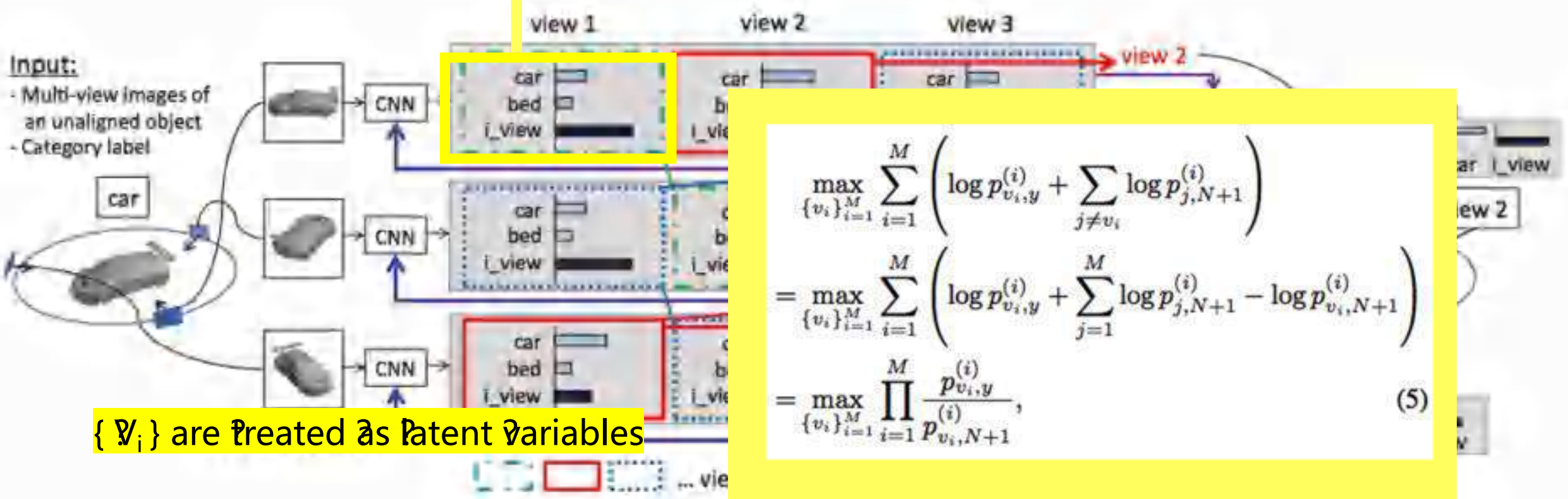
CVPR2018 RotationNet :JointiObjectiCategorizationiandiPoseiEstimationiUsingi Multiviews fromiUnsupervisediViewpoints

- Main Contribution
  - Treats the viewpoint labels as latent variables, which are learned in an unsupervised manner during the training using an unaligned object dataset. (without using known viewpoint labels for training)
  - generalizes well to a real-world image dataset that was newly created for the general task of multi-view object classification

Main idea

概率和为1

$$p_{j,k}^{(i)} = \begin{cases} 1 & (j = v_i \text{ and } k = y) \text{ or } (j \neq v_i \text{ and } k = N+1) \\ 0 & \text{(otherwise).} \end{cases} \tag{2}$$



$$\max_{\{v_i\}_{i=1}^{M}} \sum_{i=1}^{M} \left(\log p_{v_i,y}^{(i)} + \sum_{j \neq v_i} \log p_{j,N+1}^{(i)} \right)$$

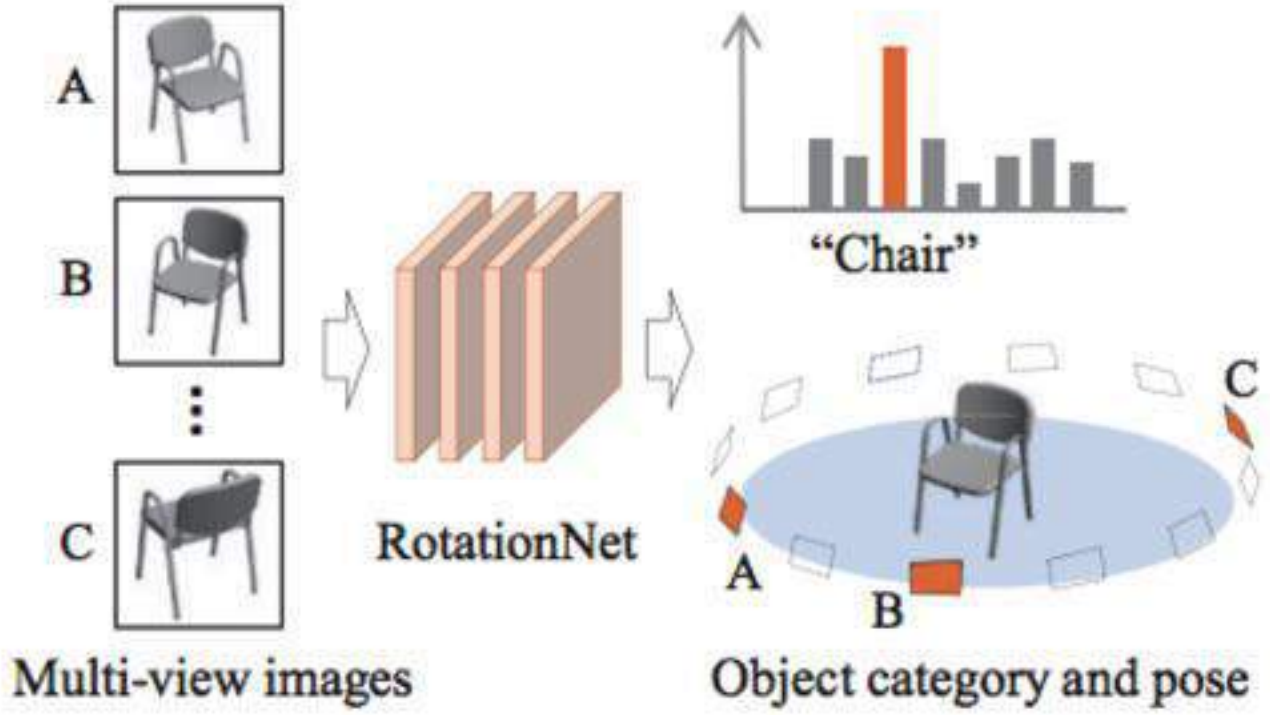
$$= \max_{\{v_i\}_{i=1}^{M}} \sum_{i=1}^{M} \left(\log p_{v_i,y}^{(i)} + \sum_{j=1}^{M} \log p_{j,N+1}^{(i)} - \log p_{v_i,N+1}^{(i)} \right)$$

$$= \max_{\{v_i\}_{i=1}^{M}} \prod_{i=1}^{M} \frac{p_{v_i,y}^{(i)}}{p_{v_i,N+1}^{(i)}}, \tag{5}$$

Number of views：M=3
Number of categories：N=3

RotationNet

- State-of-the-art methods of 3D object classification on 10- and 40-class ModelNet datasets.
- State-of-the-art performance on an object pose estimation dataset trained without known poses

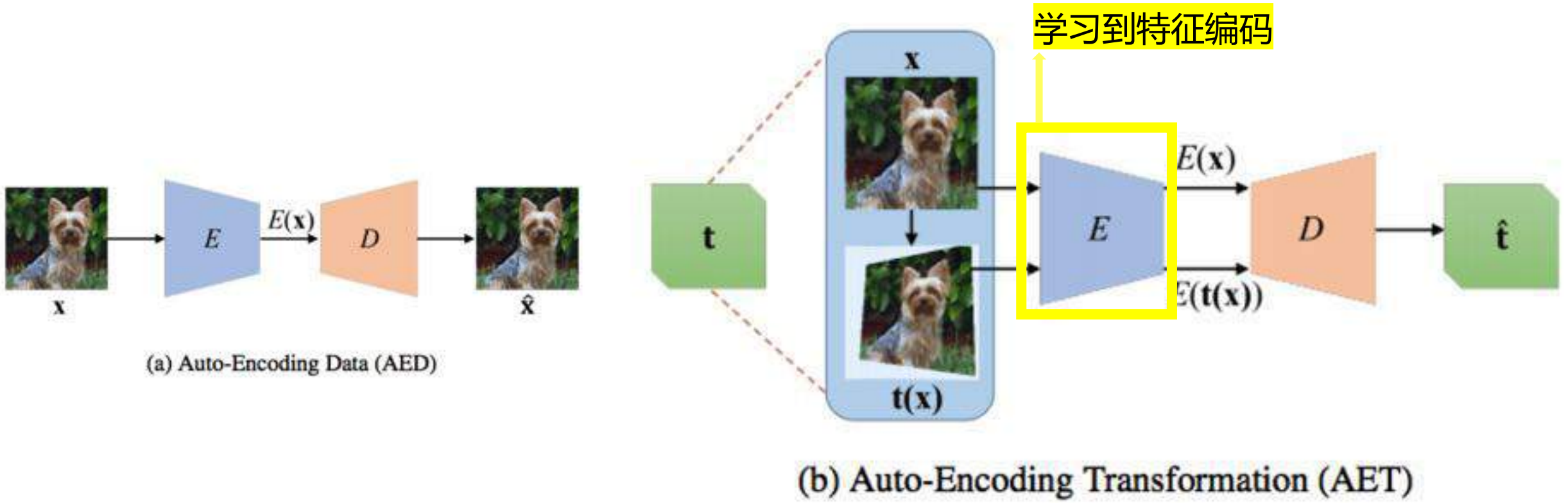


Auto-Encoding Transformation (AET)

CVPR2019: AET vs. AED: Unsupervised Representation Learning by Auto-Encoding Transformations rather than Data

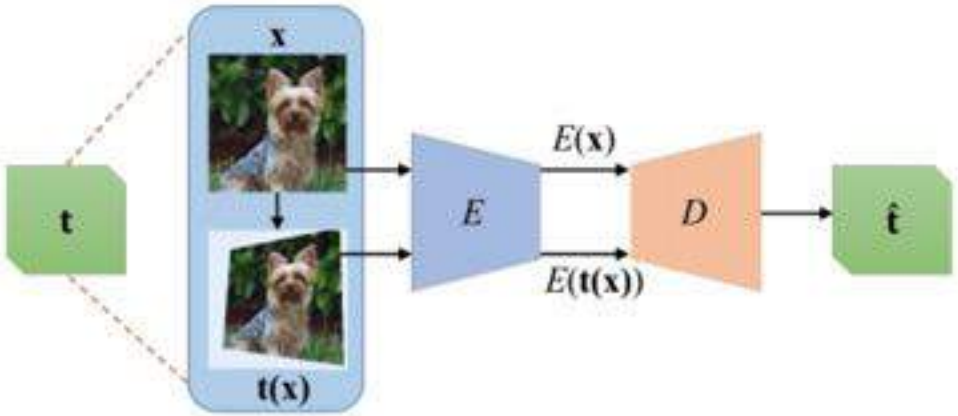
- Main Contribution
  - Learn unsupervised feature representations by AET rather than the data themselves
  - AET greatly improves over existing unsupervised approaches, setting new state-of-the-art performances being greatly closer to the upper bounds by their fully supervised counterparts on CIFAR-10, ImageNet and Places datasets

Main idea



as long as the unsupervised features successfully encode the essential information about the visual structures of original and transformed images, the transformation can be well predicted

Main Approach

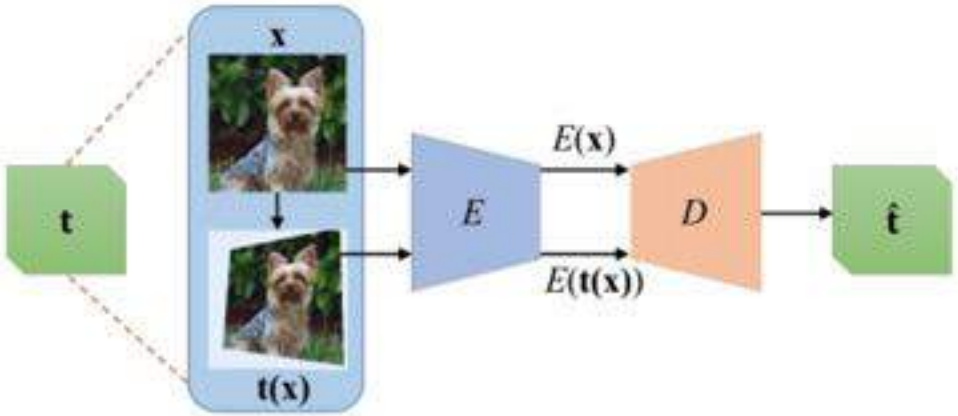


(b) Auto-Encoding Transformation (AET)

$$\min_{E,D} \mathop{\mathbb{E}}_{\mathbf{t}\sim\mathcal{T},\mathbf{x}\sim\mathcal{X}} \ell(\mathbf{t},\hat{\mathbf{t}})$$

$$\hat{\mathbf{t}} = D\left[E(\mathbf{x}), E(\mathbf{t}(\mathbf{x}))\right],$$

Main Approach



(b) Auto-Encoding Transformation (AET)

$$\min_{E,D} \mathop{\mathbb{E}}_{\mathbf{t}\sim\mathcal{T},\mathbf{x}\sim\mathcal{X}} \ell(\mathbf{t},\hat{\mathbf{t}})$$

$$\hat{\mathbf{t}} = D\left[E(\mathbf{x}), E(\mathbf{t}(\mathbf{x}))\right],$$

Parameterized Transformations

$$\mathcal{T} = \{\mathbf{t}_{\boldsymbol{\theta}} | \boldsymbol{\theta} \sim \boldsymbol{\Theta}\}$$

$$M(\boldsymbol{\theta}) \in \mathbb{R}^{3\times 3} \quad \ell(\mathbf{t}_{\boldsymbol{\theta}}, \mathbf{t}_{\hat{\boldsymbol{\theta}}}) = \frac{1}{2}\|M(\boldsymbol{\theta}) - M(\hat{\boldsymbol{\theta}})\|_2^2$$

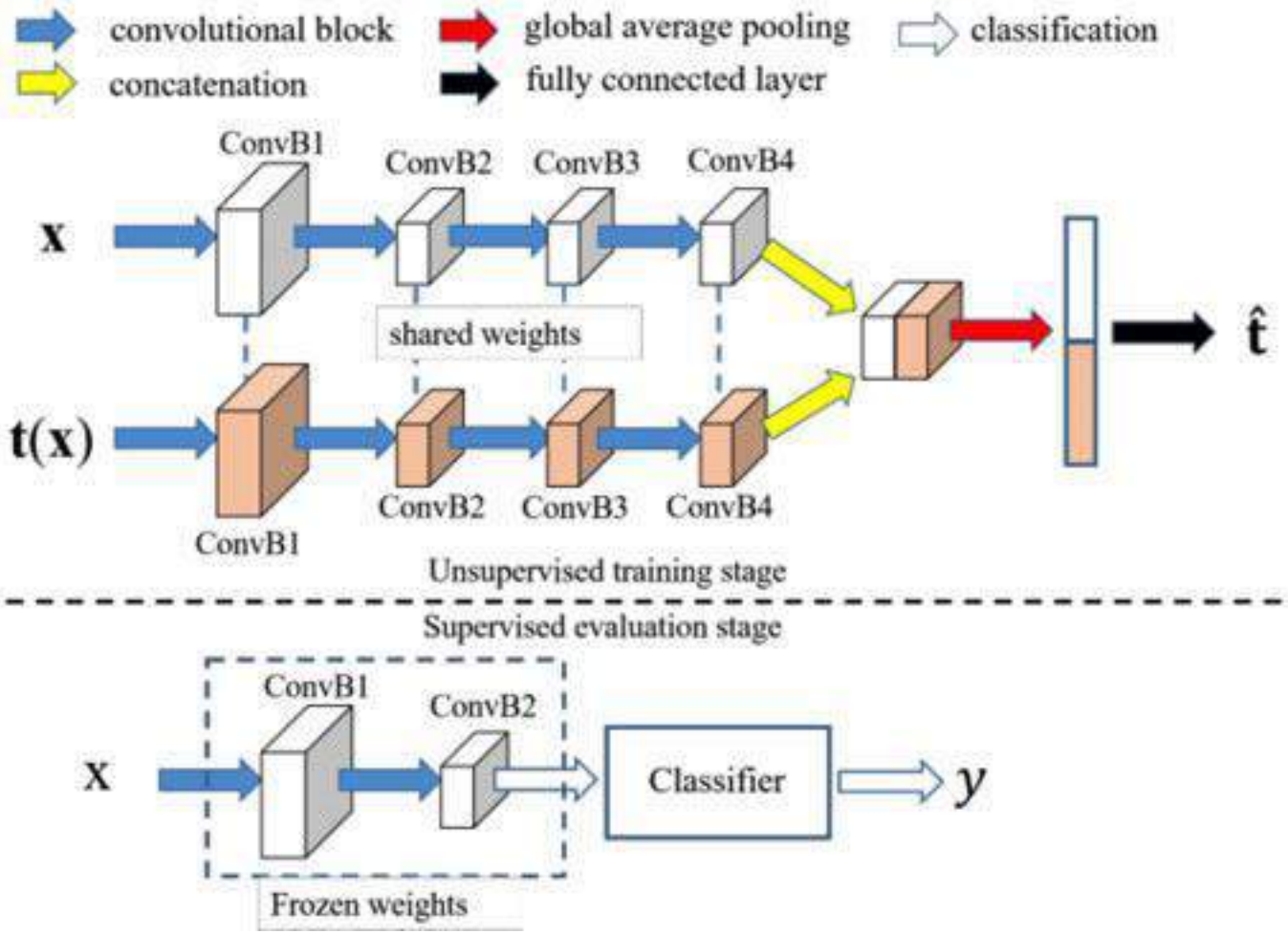
GAN-Induced Transformations

$$\mathbf{t}_{\mathbf{z}}(\mathbf{x}) = G(\mathbf{x},\mathbf{z}) \quad \ell(\mathbf{t}_{\mathbf{z}}, \mathbf{t}_{\hat{\mathbf{z}}}) = \frac{1}{2}\|\mathbf{z} - \hat{\mathbf{z}}\|_2^2$$

Non-Parametric Transformations

$$\ell(\mathbf{t},\hat{\mathbf{t}}) = \mathop{\mathbb{E}}_{\mathbf{x}\sim\mathcal{X}} \mathrm{dist}(\mathbf{t}(\mathbf{x}), \hat{\mathbf{t}}(\mathbf{x}))$$

Experiments

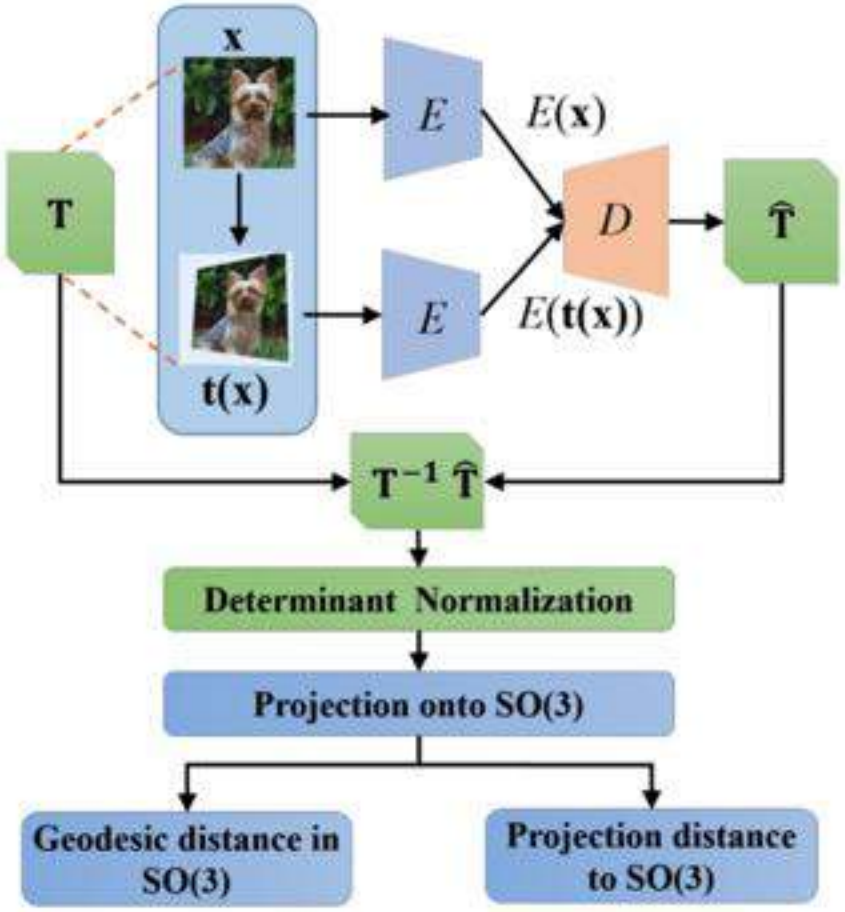


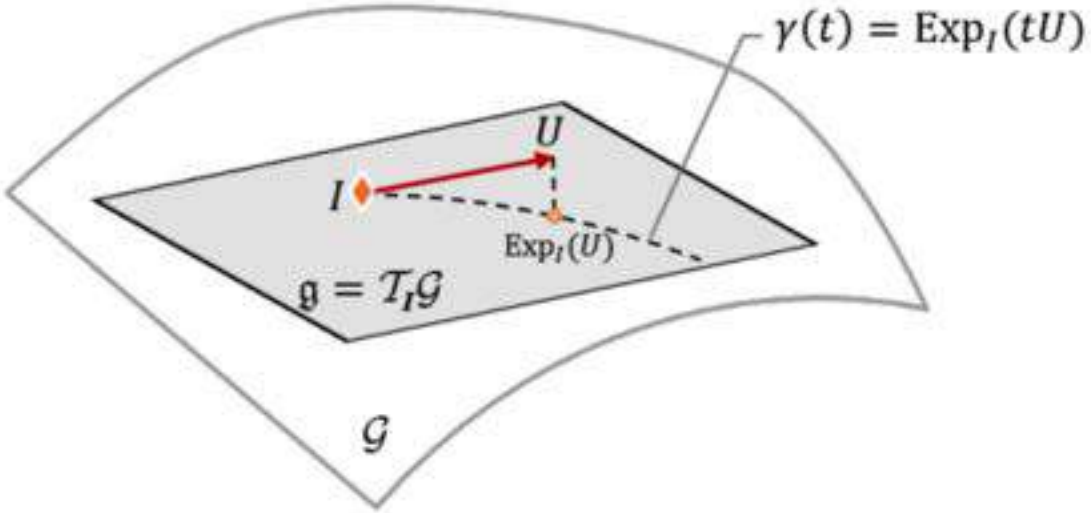
两个NIN（Network-In-Network）结构分支

Method	Error rate
Supervised NIN (Lower Bound)	7.20
Random Init. + conv (Upper Bound)	27.50
Roto-Scat + SVM [22]	17.7
ExamplarCNN [7]	15.7
DCGAN [26]	17.2
Scattering [21]	15.3
RotNet + FC [10]	10.94
RotNet + conv [10]	8.84
(Ours) AET-affine + FC	9.77
(Ours) AET-affine + conv	8.05
(Ours) AET-project + FC	**9.41**
(Ours) AET-project + conv	**7.82**

AETv2

AETv2: AutoEncoding Transformations for Self-Supervised Representation Learning by Minimizing Geodesic Distances in Lie Groups *(Submitted on 16 Nov 2019)*





所有合法变换构成的空间是一个完全的 Li group,任意两个变换之间的距离应该用测地距离而不是欧式距离

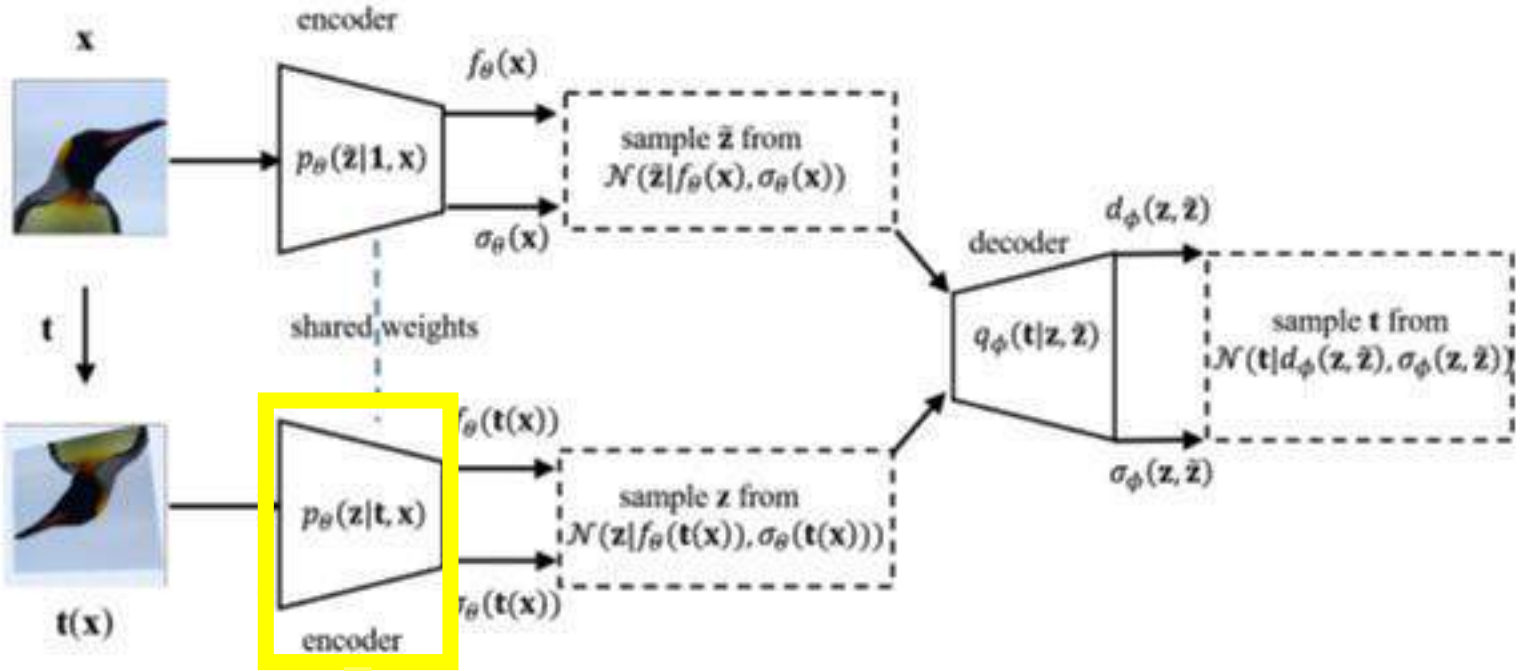
$$\hat{\ell}(\hat{\mathbf{T}}, \mathbf{T}) = \arccos\left[\frac{\mathrm{tr}(\mathbf{P}) - 1}{2}\right] + \lambda\,\mathrm{tr}(\mathbf{R}_{\Pi}^{\intercal}\mathbf{R}_{\Pi})$$

AVT

AVT: Unsupervised Learning of Transformation Equivariant Representations by Autoencoding Variational Transformations *(Submitted on 25 June 2019)*

- Main Contribution
  - Train the networks by maximizing the mutual information between the transformations and representations.
  - The proposed AVT model sets a new record for the performances on unsupervised tasks, greatly closing the performance gap to the supervised models.

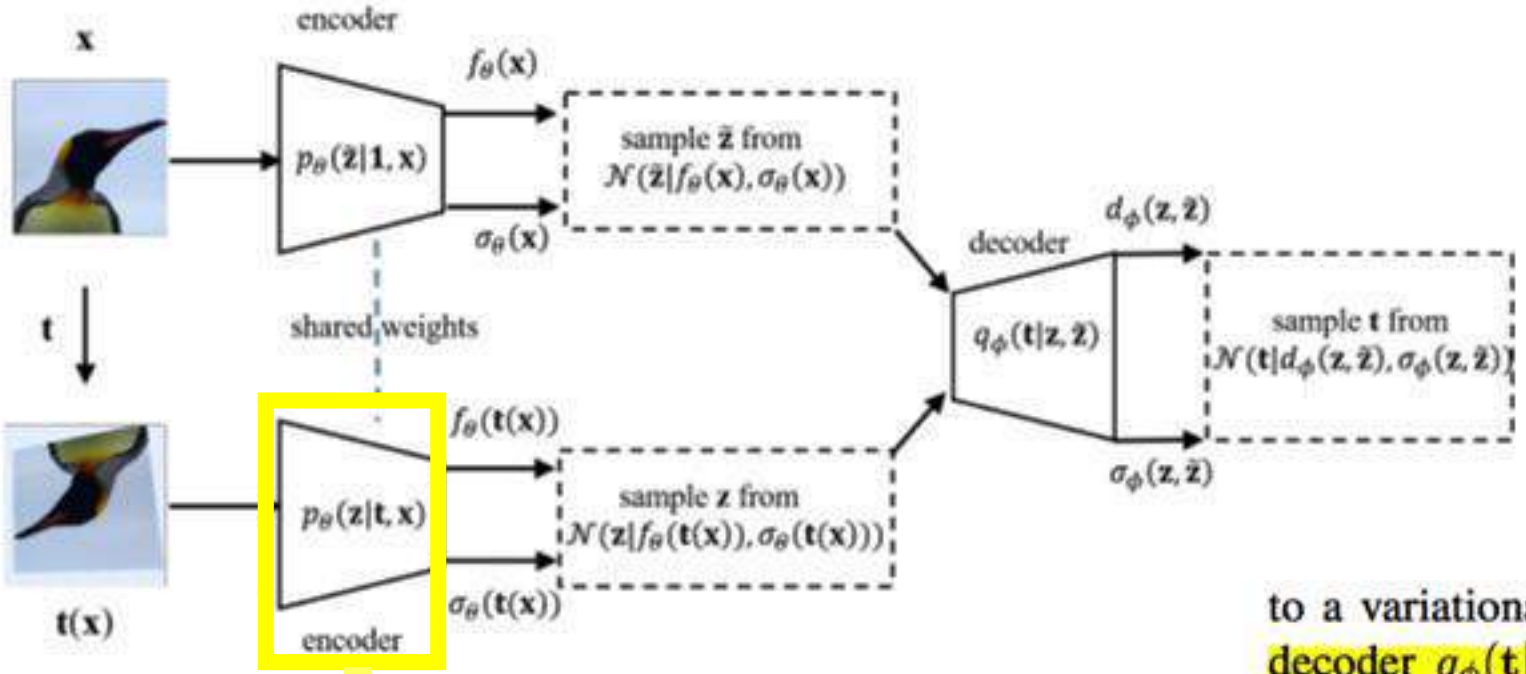
Main idea



$$\mathbf{z} = f_\theta(\mathbf{t}(\mathbf{x})) + \sigma_\theta(\mathbf{t}(\mathbf{x})) \circ \epsilon$$

$$p_\theta(\mathbf{z}|\mathbf{t},\mathbf{x}) \triangleq \mathcal{N}\left(\mathbf{z}|f_\theta(\mathbf{t}(\mathbf{x})), \sigma_\theta^2(\mathbf{t}(\mathbf{x}))\right)$$

Main idea



$$\max_\theta I(\mathbf{t};\mathbf{z}|\tilde{\mathbf{z}})$$

to a variational approach by introducing a transformation decoder $q_\phi(\mathbf{t}|\mathbf{z},\tilde{\mathbf{z}})$ with the parameter ϕ to approximate $p_\theta(\mathbf{t}|\mathbf{z},\tilde{\mathbf{z}})$. In the next section, we will elaborate on this

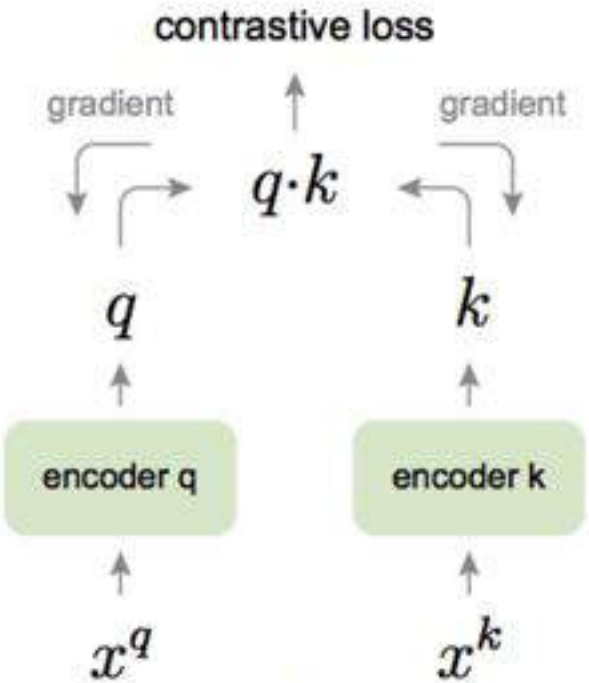
$$\mathbf{z} = f_\theta(\mathbf{t}(\mathbf{x})) + \sigma_\theta(\mathbf{t}(\mathbf{x})) \circ \epsilon$$

$$p_\theta(\mathbf{z}|\mathbf{t},\mathbf{x}) \triangleq \mathcal{N}\left(\mathbf{z}|f_\theta(\mathbf{t}(\mathbf{x})),\sigma_\theta^2(\mathbf{t}(\mathbf{x}))\right)$$

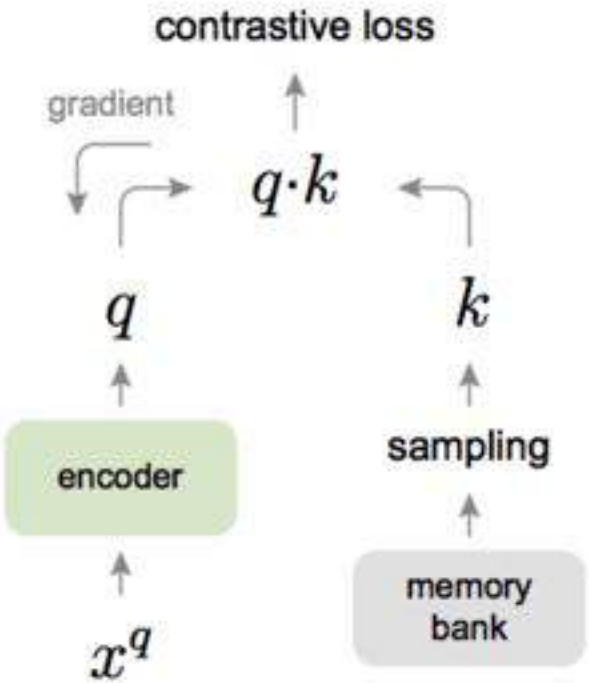
Summary

- 无监督的数据增强
- 希望学到怎样的特征？

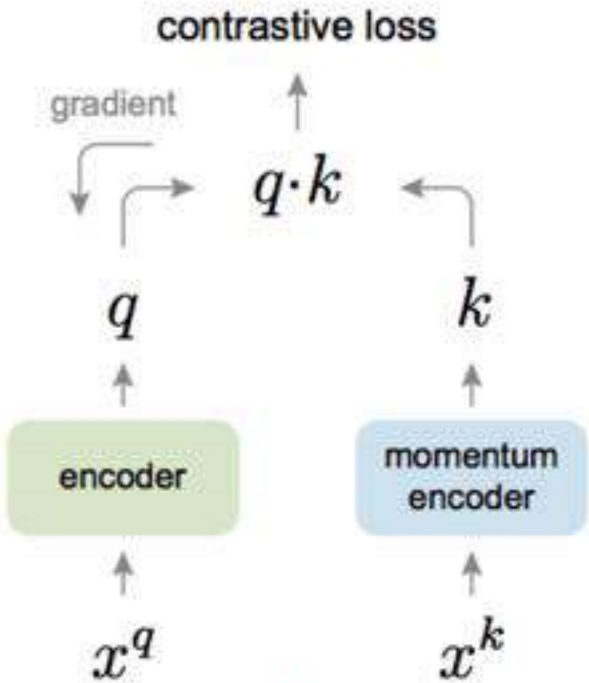
基于Instance Discrimination的无监督训练



(a) end-to-end



(b) memory bank



(c) MoCo

$$P(i|\mathbf{v}) = \frac{\exp\left(\mathbf{w}_i^T \mathbf{v}\right)}{\sum_{j=1}^{n} \exp\left(\mathbf{w}_j^T \mathbf{v}\right)}.$$

$$P(i|v) = \frac{exp(v_i^T v/\tau)}{\sum_{j=1}^{n} exp(v_j^T v/\tau)}$$

$$\mathcal{L}_q = -\log \frac{\exp(q \cdot k_+/\tau)}{\sum_{i=0}^{K} \exp(q \cdot k_i/\tau)}$$

ExamplarCNN

NIPS 2014 Discriminative Unsupervised Feature Learning with Convolutional Neural Networks

Figure 1: Exemplary patches sampled from the STL unlabeled dataset which are later augmented by various transformations to obtain surrogate data for the CNN training.

Figure 2: Several random transformations applied to one of the patches extracted from the STL unlabeled dataset. The original ('seed') patch is in the top left corner.

Parametric Classifier.

$$P(i|\mathbf{v}) = \frac{\exp\left(\mathbf{w}_i^T\mathbf{v}\right)}{\sum_{j=1}^{n}\exp\left(\mathbf{w}_j^T\mathbf{v}\right)}.$$

将每一个instance当成一个单独的类来进行学习

NCE

CVPR 2018 spotlight paper Unsupervised Feature Learning via Non-Parametric Instance-level Discrimination

- Main Contribution
  - Formulate a non-parametric classification problem at the instance-level, and use noise contrastive estimation to tackle the computational challenges imposed by the large number of instance classes.
  - With 128 features per image, our method requires only 600MB storage for a million images, enabling fast nearest neighbour retrieval at the run time.

Main idea

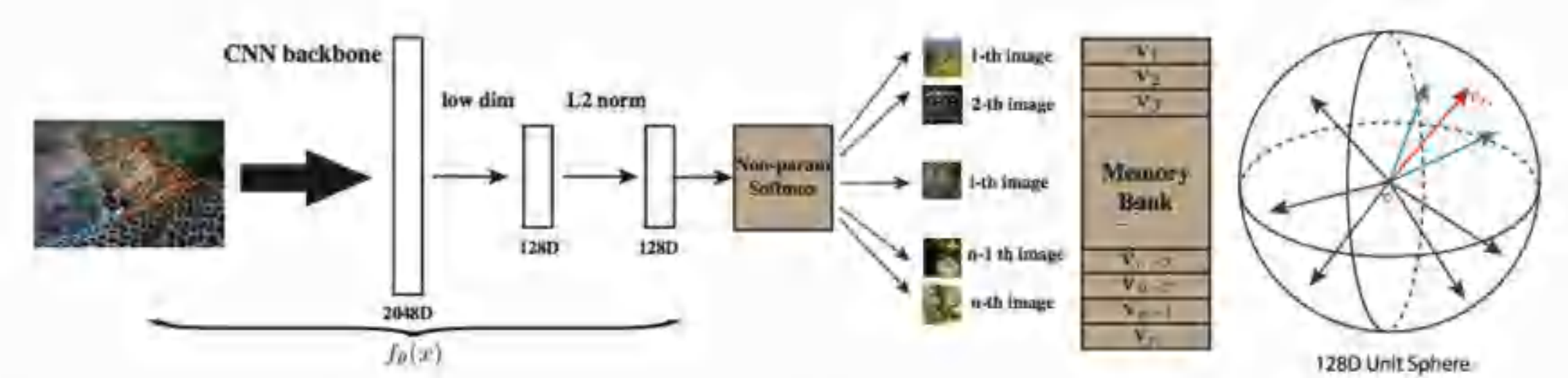
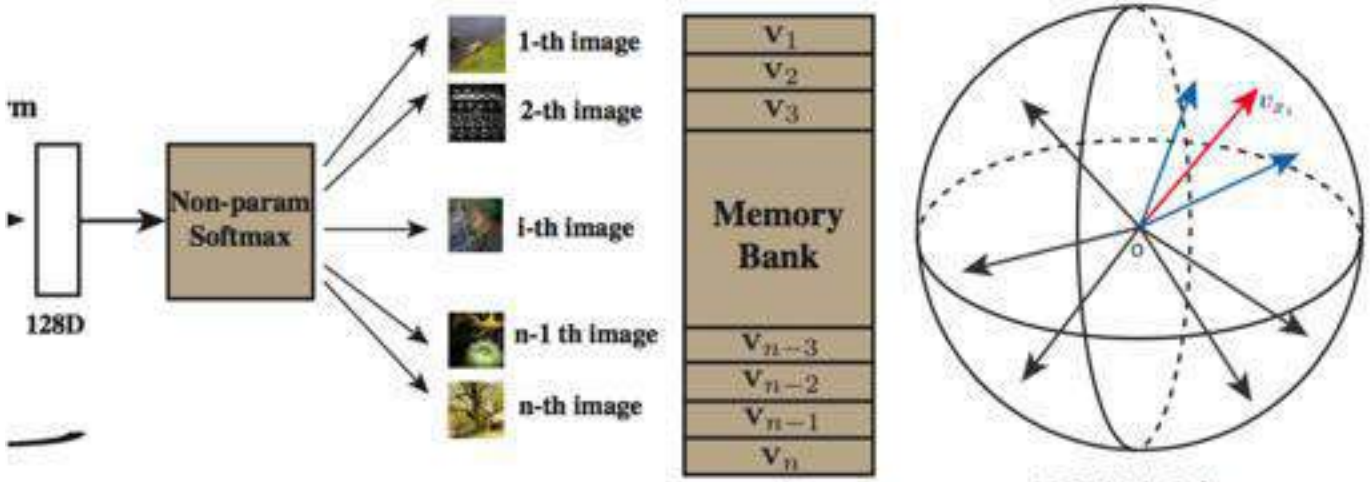


Figure 2: The pipeline of our unsupervised feature learning approach. We use a backbone CNN to encode each image as a feature vector, which is projected to a 128-dimensional space and L2 normalized. The optimal feature embedding is learned via instance-level discrimination, which tries to maximally scatter the features of training samples over the 128-dimensional unit sphere.

构造memory bank替代之前的classifier weights

Main Approach



128D Unit Sphere

① **Non-Parametric Softmax Classifier**

$$P(i|v) = \frac{exp(w_i^T v)}{\sum_{j=1}^{n} exp(w_j^T v)} \qquad P(i|v) = \frac{exp(v_i^T v/\tau)}{\sum_{j=1}^{n} exp(v_j^T v/\tau)}$$

② **Memory Bank**

the network. Let $V = \{\mathbf{v}_j\}$ be the memory bank and $\mathbf{f}_i = f_\theta(x_i)$ be the feature of x_i. During each learning itera-

θ are optimized via stochastic gradient descend. Then $\mathbf{f}_i$ is updated to V at the corresponding instance entry $\mathbf{f}_i \rightarrow \mathbf{v}_i$.

③ **Noise-Contrastive Estimation**

$$Z \simeq Z_i \simeq nE_j\left[\exp(\mathbf{v}_j^T \mathbf{f}_i/\tau)\right] = \frac{n}{m}\sum_{k=1}^{m}\exp(\mathbf{v}_{j_k}^T \mathbf{f}_i/\tau),$$

④ **Proximal Regularization**

$$h(i,v) := P(D=1|i,v) = \frac{P(i|v)}{P(i|v) + mP_n(i)}$$

$$J_{NCE}(\theta) = -E_{P_d}[\log h(i,v) - \lambda||v_i^{(t)} - v_i^{(t-1)}||_2^2] - m \cdot E_{P_n}[\log(1-h(i,v'))]$$

⑤ **Weighted k-Nearest Neighbor Classifier**

Invariant and Spreading Instance Feature

CVPR 2019 Unsupervised Embedding Learning via Invariant and Spreading Instance Feature

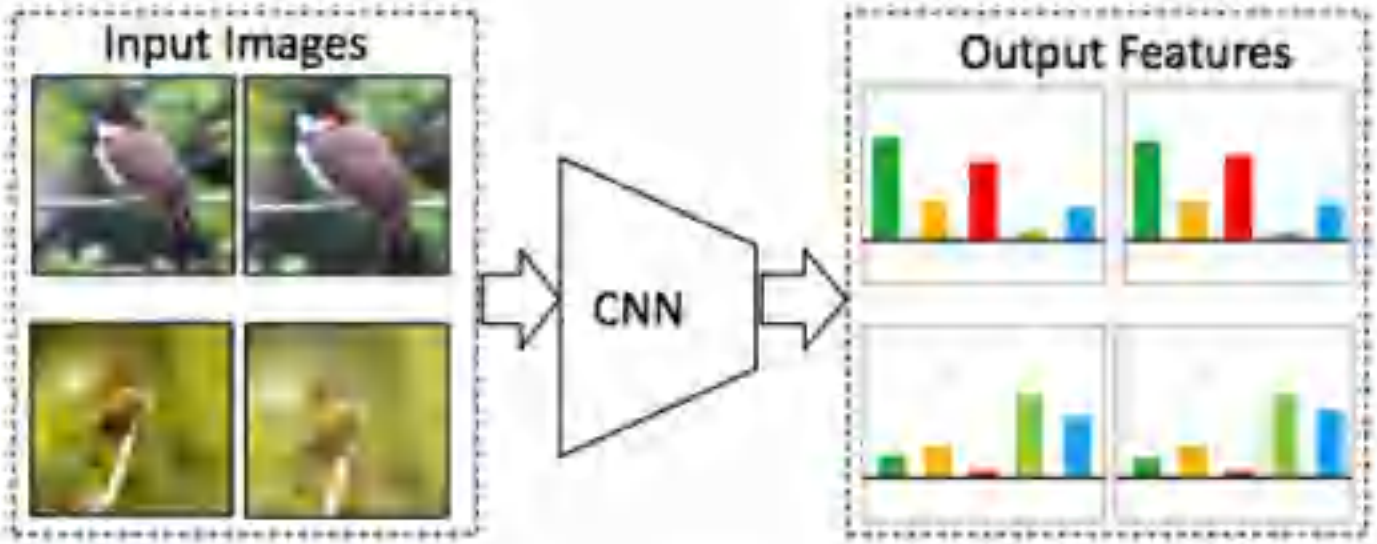
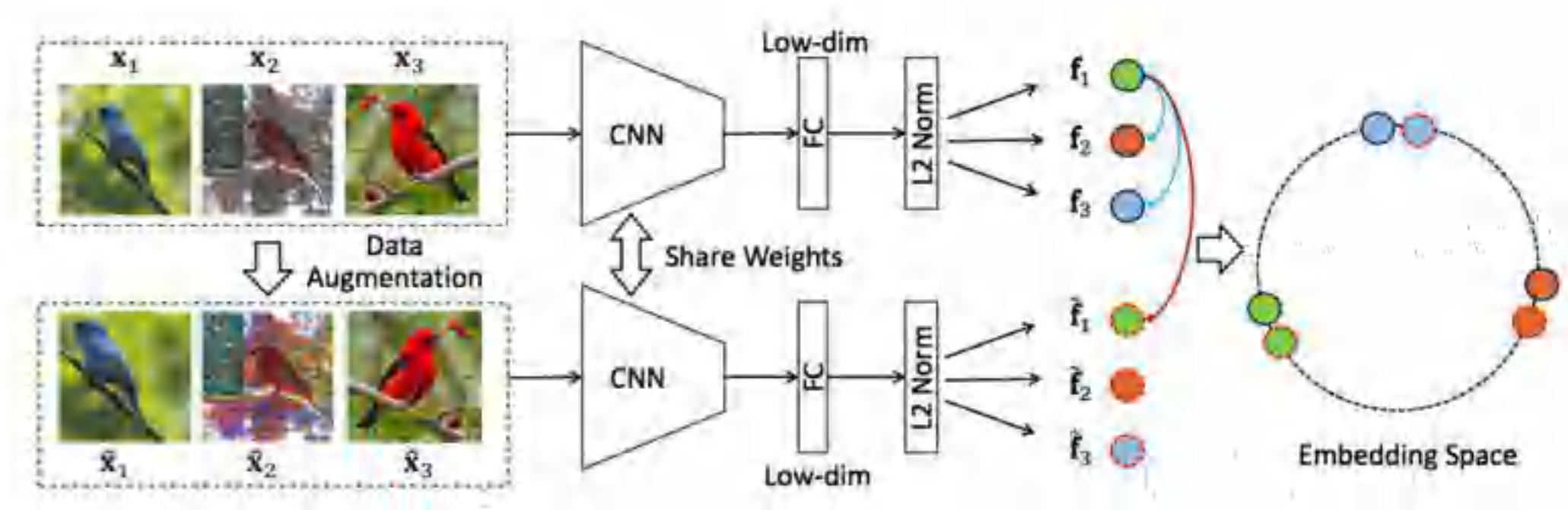


Figure 1: Illustration of our basic idea. The features of the same instance under different data augmentations should be invariant, while features of different image instances should be separated.

Main idea



$$P(i|\hat{\mathbf{x}}_i) = \frac{\exp(\mathbf{f}_i^T \hat{\mathbf{f}}_i/\tau)}{\exp(\mathbf{f}_i^T \hat{\mathbf{f}}_i/\tau) + \sum_{k \neq i} \exp(\mathbf{f}_k^T \hat{\mathbf{f}}_i/\tau)},$$

$$J = -\sum_i \log P(i|\hat{\mathbf{x}}_i) - \sum_i \sum_{j \neq i} \log(1 - P(i|\mathbf{x}_j)).$$

Experiment

Figure 4: kNN retrieval results of some example queries on CUB200-2011 dataset. The positive (negative) retrieved results are framed in (red). The similarity is measured with cosine similarity.

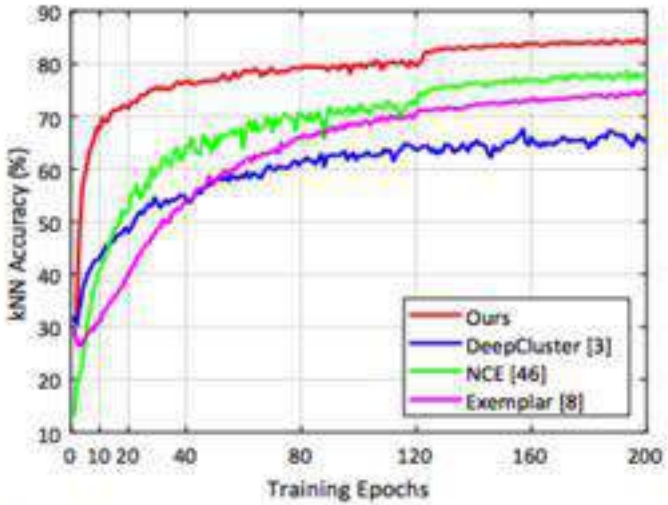
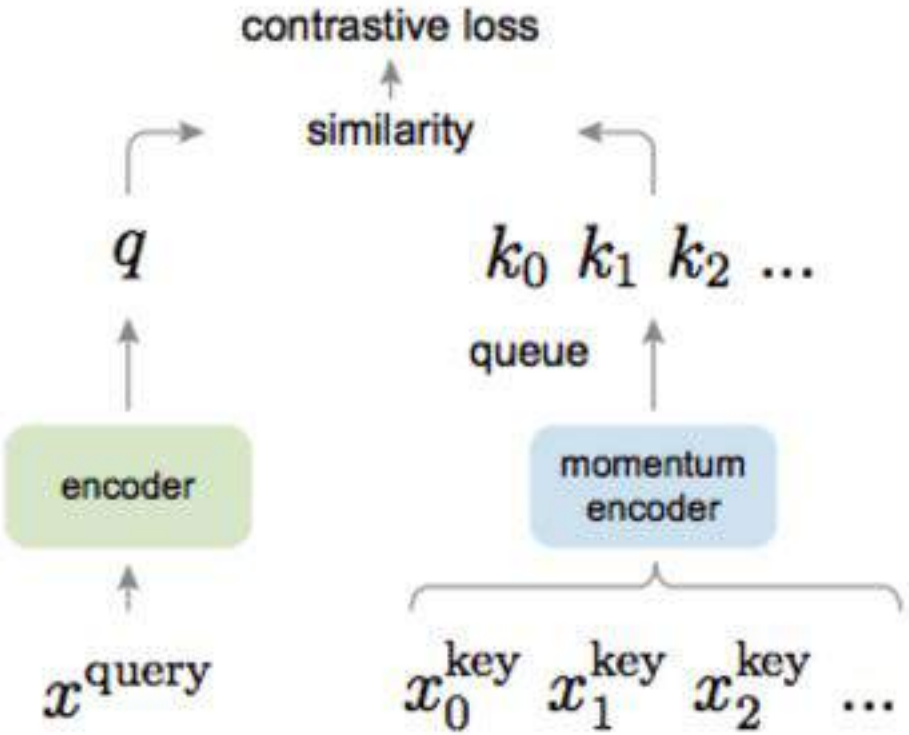


Figure 3: Evaluation of the training efficiency on CIFAR-10 dataset. kNN accuracy (%) at each epoch is reported, demonstrating the learning speed of different methods.

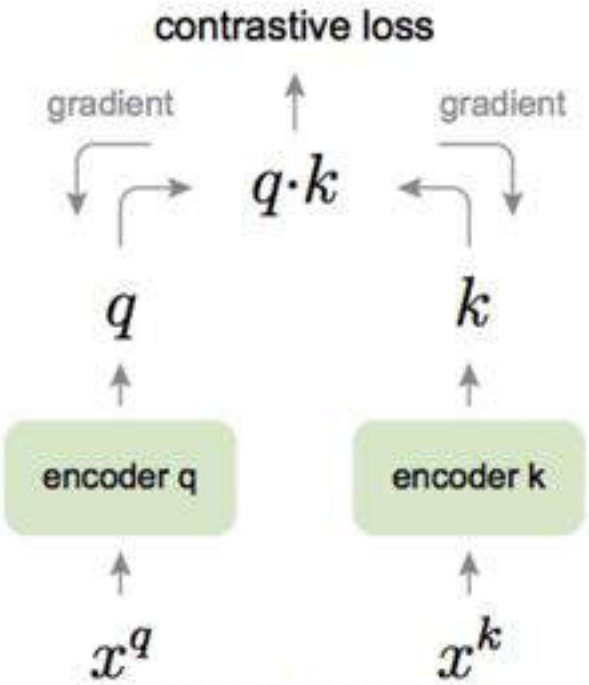
Momentum Contrast

Momentum Contrast for Unsupervised Visual Representation Learning. Nov 13 2019

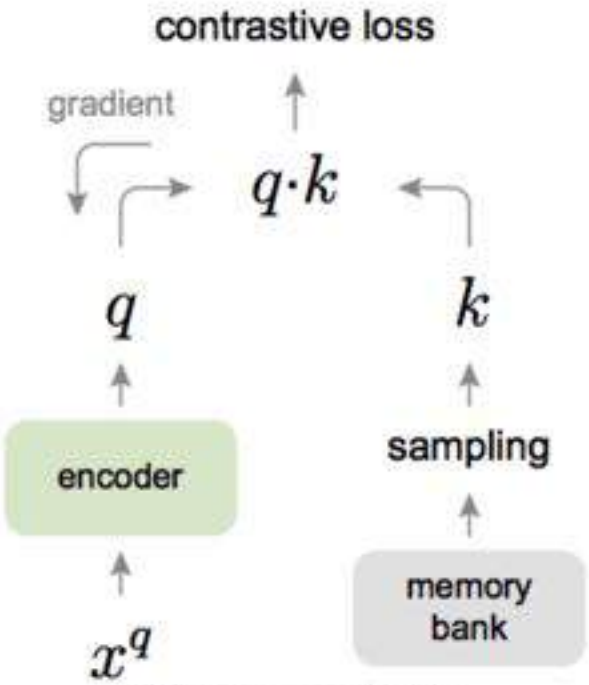


$$\theta_{\mathrm{k}} \leftarrow m\theta_{\mathrm{k}} + (1-m)\theta_{\mathrm{q}}.$$

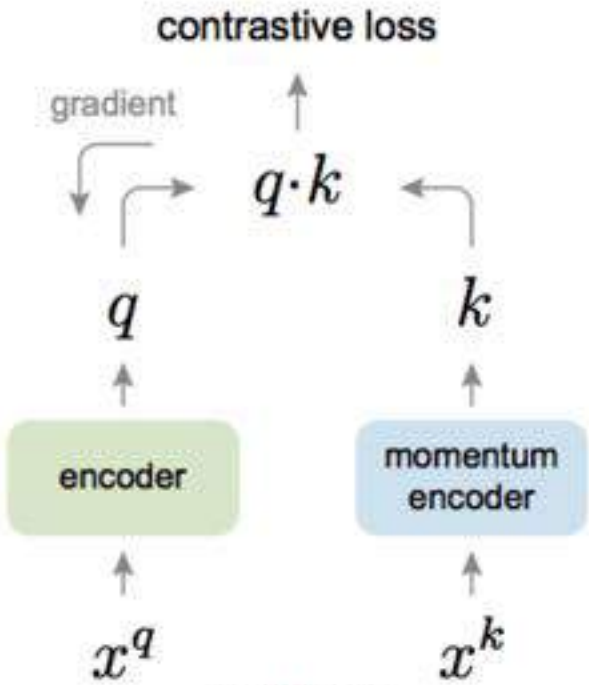
Summary



(a) end-to-end



(b) memory bank



(c) MoCo

$$P(i|\mathbf{v}) = \frac{\exp\left(\mathbf{w}_i^T \mathbf{v}\right)}{\sum_{j=1}^{n} \exp\left(\mathbf{w}_j^T \mathbf{v}\right)}.$$

$$P(i|v) = \frac{exp(v_i^T v/\tau)}{\sum_{j=1}^{n} exp(v_j^T v/\tau)}$$

$$\mathcal{L}_q = -\log \frac{\exp(q \cdot k_+/\tau)}{\sum_{i=0}^{K} \exp(q \cdot k_i/\tau)}$$

基于变换
VS
基于判别

离散表示学习

VQ-VAE (Vector Quantised)

NIPS2017 Neural Discrete Representation Learning

Our contributions can thus be summarised as:

- Introducing the VQ-VAE model, which is simple, uses discrete latents, does not suffer from "posterior collapse" and has no variance issues.
- We show that a discrete latent model (VQ-VAE) perform as well as its continuous model counterparts in log-likelihood.
- When paired with a powerful prior, our samples are coherent and high quality on a wide variety of applications such as speech and video generation.
- We show evidence of learning language through raw speech, without any supervision, and show applications of unsupervised speaker conversion.

自回归模型

PixelCNNs (and PixelRNNs) [30] model the joint distribution of pixels over an image $\mathbf{x}$ as the following product of conditional distributions, where x_i is a single pixel:

$$p(\mathbf{x}) = \prod_{i=1}^{n^2} p(x_i | x_1, ..., x_{i-1}). \quad (1)$$

- ？？？
  - 递归顺序
  - 加速采样
  - 类内联系

How to realize ?

In this work we introduce the VQ-VAE where we use discrete latent variables with a new way of training, inspired by vector quantisation (VQ). The posterior and prior distributions are categorical, and the samples drawn from these distributions index an embedding table. These embeddings are then used as input into the decoder network.

The posterior categorical distribution $q(z|x)$ probabilities are defined as one-hot as follows:

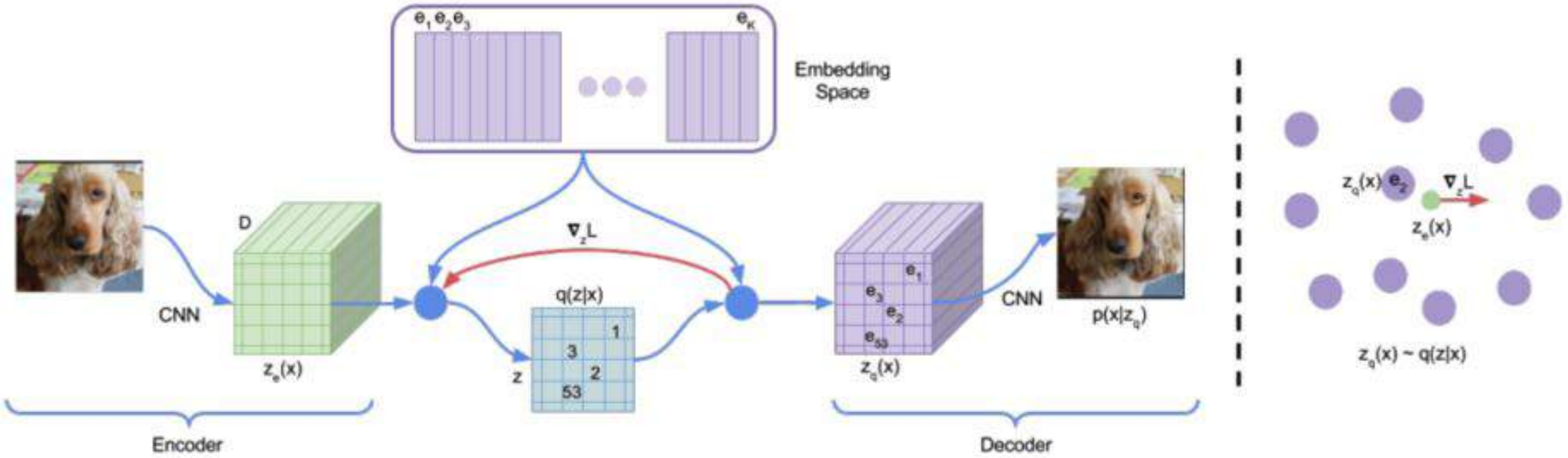
$$q(z=k|x) = \begin{cases} 1 & \text{for } k = \text{argmin}_j \|z_e(x) - e_j\|_2, \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

where $z_e(x)$ is the output of the encoder network. We view this model as a VAE in which we can bound $\log p(x)$ with the ELBO. Our proposal distribution $q(z=k|x)$ is deterministic, and by defining a simple uniform prior over z we obtain a KL divergence constant and equal to $\log K$.

The representation $z_e(x)$ is passed through the discretisation bottleneck followed by mapping onto the nearest element of embedding e as given in equations 1 and 2.

$$z_q(x) = e_k, \quad \text{where} \quad k = \text{argmin}_j \|z_e(x) - e_j\|_2 \quad (2)$$

How to realize ?



① $z_q(x) = e_k, \quad \text{where} \quad k = \text{argmin}_j \|z_e(x) - e_j\|_2$

② straight-through estimator

$$\mathcal{L}(\mathbf{x}, D(\mathbf{e})) = \|\mathbf{x} - D(\mathbf{e})\|_2^2 + \|sg[E(\mathbf{x})] - \mathbf{e}\|_2^2 + \beta\|sg[\mathbf{e}] - E(\mathbf{x})\|_2^2$$

Experiments

- Comparison with continuous variables

Figure 2: Left: ImageNet 128x128x3 images, right: reconstructions from a VQ-VAE with a 32x32x1 latent space, with K=512.

- Audio
- Video

VQ-VAE2

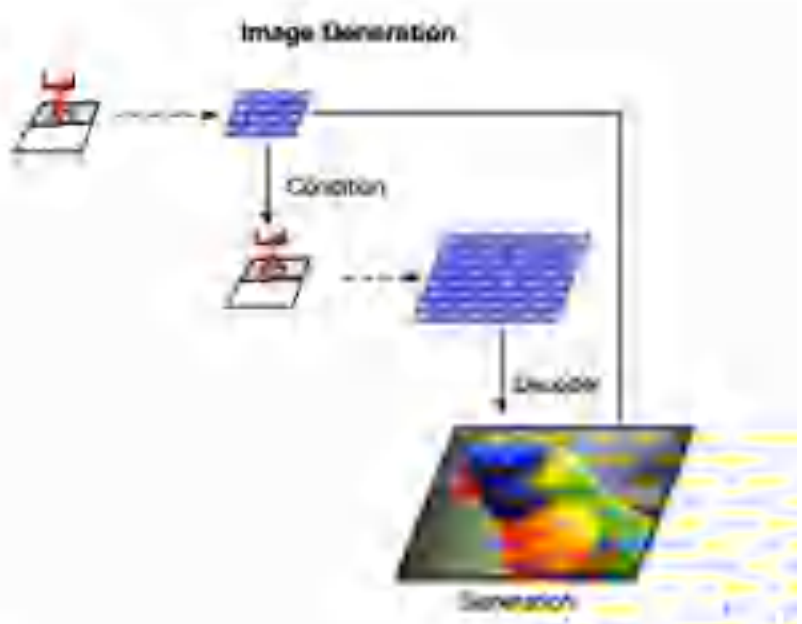
Generating Diverse High-Fidelity Images with VQ-VAE-2

Figure 1: Class-conditional 256x256 image samples from a two-level model trained on ImageNet.

Large scale image generation，效果堪比BigGAN !!!

VQ-VAE2

(a) Overview of the architecture of our hierarchical VQ-VAE. The encoders and decoders consist of deep neural networks. The input to the model is a 256 × 256 image that is compressed to quantized latent maps of size 64 × 64 and 32 × 32 for the *bottom* and *top* levels, respectively. The decoder reconstructs the image from the two latent maps.



(b) Multi-stage image generation. The top-level PixelCNN prior is conditioned on the class label, the bottom level PixelCNN is conditioned on the class label as well as the first level code. Thanks to the feed-forward decoder, the mapping between latents to pixels is fast. (The example image with a parrot is generated with this model).

Summary

- VQ-VAE 因在潜在表示空间使用自回归神经网络，捕捉到了更多的结构化的全局关联信息。

- 而 VQ-VAE-2 将顶层全局与底层局部信息分离开来，生成全局自洽，局部高清的图像。