

AVT: Unsupervised Learning of Transformation Equivariant Representations by Autoencoding Variational Transformations

Guo-Jun Qi^{1,2,*}, Liheng Zhang¹, Chang Wen Chen³, Qi Tian⁴

¹Laboratory for MACHine Perception and LEarning (MAPLE)

<http://maple-lab.net/>

²Huawei Cloud, ⁴Huawei Noah's Ark Lab

³The Chinese University of Hong Kong at Shenzhen and Peng Cheng Laboratory

guojun.qi@huawei.com

<http://maple-lab.net/projects/AVT.htm>

Abstract

The learning of Transformation-Equivariant Representations (TERs), which is introduced by Hinton et al. [16], has been considered as a principle to reveal visual structures under various transformations. It contains the celebrated Convolutional Neural Networks (CNNs) as a special case that only equivary to the translations. In contrast, we seek to train TERs for a generic class of transformations and train them in an unsupervised fashion. To this end, we present a novel principled method by Autoencoding Variational Transformations (AVT), compared with the conventional approach to autoencoding data. Formally, given transformed images, the AVT seeks to train the networks by maximizing the mutual information between the transformations and representations. This ensures the resultant TERs of individual images contain the intrinsic information about their visual structures that would equivary extricably under various transformations in a generalized nonlinear case. Technically, we show that the resultant optimization problem can be efficiently solved by maximizing a variational lower-bound of the mutual information. This variational approach introduces a transformation decoder to approximate the intractable posterior of transformations, resulting in an autoencoding architecture with a pair of the representation encoder and the transformation decoder. Experiments demonstrate the proposed AVT model sets a new record for the performances on unsupervised tasks, greatly closing the performance gap to the supervised models.

*Corresponding author: G.-J. Qi. Email: guojunq@gmail.com. The idea was conceived and formulated by G.-J. Qi, and L. Zhang performed experiments while interning at Huawei Cloud.

1. Introduction

Convolutional Neural Networks (CNNs) have demonstrated tremendous successes when a large volume of labeled data are available to train the models. Although a solid theory is still lacking, it is thought that both *equivariance* and *invariance* to image translations play a critical role in the success of CNNs [6, 7, 34, 16], particularly for supervised tasks.

Specifically, while the whole network is trained in an end-to-end fashion, a typical CNN model consists of two parts: the convolutional *feature maps* of an input image through multiple convolutional layers, and the classifier of *fully connected layers* mapping the feature maps to the target labels. It is obvious that a supervised classification task requires the fully connected classifier to predict labels invariant to transformations. For training a CNN model, such a transformation invariance criterion is achieved by minimizing the classification errors on the labeled examples augmented with various transformations [22]. Unfortunately, it is impossible to simply apply transformation invariance to learn an unsupervised representation without label supervision, since this would result in a trivial constant representation for any input images.

On the contrary, it is not hard to see that the representations generated through convolutional layers are equivariant to the transformations – the feature maps of translated images are also shifted in the same way subject to edge padding effect [22]. It is natural to generalize this idea by considering more types of transformations beyond translations, e.g., image warping and projective transformations [6].

In this paper, we formalize the concept of *transformation equivariance* as the criterion to train an unsupervised representation. We expect it could learn the representation-

s that are generalizable to unseen tasks without knowing their labels. This is contrary to the criterion of transformation invariance in supervised tasks, which aims to tailor the representations to predefined tasks and their labels. Intuitively, training a transformation equivariant representation is not surprising – a good representation should be able to preserve the *intrinsic* visual structures of images, so that it could *extrinsically* equivary to various transformations as they change the represented visual structures. In other words, a transformation will be able to be decoded from such representations that well encode the visual structures before and after the transformation [37].

For this purpose, we present a novel paradigm of Autoencoding Variational Transformations (AVT) to learn powerful representations that equivary against a generic class of transformations. We formalize it from an information-theoretic perspective by considering a joint probability between images and transformations. This enables us to use the mutual information to characterize the dependence between the representations and the transformations. Then, the AVT model can be trained directly by maximizing the mutual information in an unsupervised fashion without any labels. This will ensure the resultant representations contain intrinsic information about the visual structures that could be transformed extrinsically for individual images. Moreover, we will show that the representations learned in this way can be computed directly from the transformations and the representations of the original images *without a direct access to the original samples*, and this allows us to generalize the existing linear Transformation Equivariant Representations (TERs) to more general nonlinear cases [31].

Unfortunately, it is intractable to maximize the mutual information directly as it is impossible to exactly evaluate the posterior of a transformation from the associated representation. Thus, we instead seek to maximize a variational lower bound of the mutual information by introducing a transformation decoder to approximate the intractable posterior. This results in an efficient autoencoding transformation (instead of data) architecture by jointly encoding a transformed image and decoding the associated transformation.

The resultant AVT model disruptively differs from the conventional auto-encoders [18, 19, 35] that seek to learn representations by reconstructing images. Although the transformation could be decoded from the reconstructed original and transformed images, this is a quite strong assumption as such representations could contain *more than enough* information about both necessary and unnecessary visual details. The AVT model is based on a weaker assumption that the representations are trained to contain only the *necessary* information about visual structures to decode the transformation between the original and trans-

formed images. Intuitively, it is harder to reconstruct a high-dimensional image than decoding a transformation that has fewer degrees of freedom. In this sense, conventional auto-encoders tend to over-represent an image with every detail, no matter if they are necessary or not. Instead, the AVT could learn more generalizable representations by identifying the most essential visual structures to decode transformations, thereby yielding better performances for downstream tasks.

This remainder of this paper is organized as follows. In Section 2, we will review the related works on unsupervised methods. We will formalize the proposed AVT model by maximizing the mutual information between representations and transformations in Section 3. It is followed by the variational approach elaborated in Section 4. Experiment results will be demonstrated in Section 5 and we conclude the paper in Section 6.

2. Related Works

In this section, we will review some related methods for training transformation-equivariant representations, along with the other unsupervised models.

2.1. Transformation-Equivariant Representations

The study of transformation-equivariance can be traced back to the idea of training capsule nets [34, 16, 17], where the capsules are designed to equivary to various transformations with vectorized rather than scalar representations. However, there was a lack of explicit training mechanism to ensure the resultant capsules be of transformation equivariance.

To address this problem, many efforts have been made in literature [6, 8, 24] to extend the conventional translation-equivariant convolutions to cover more transformations. For example, group equivariant convolutions (G-convolution) [6] have been developed to equivary to more types of transformations so that a richer family of geometric structures can be explored by the classification layers on top of the generated representations. The idea of group equivariance has also been introduced to the capsule nets [24] by ensuring the equivariance of output pose vectors to a group of transformations with a generic routing mechanism.

However, these group equivariant convolutions and capsules must be trained in a supervised fashion [6, 24] with labeled data for specific tasks, instead of learning unsupervised transformation-equivariant representations generalizable to unseen tasks. Moreover, their representations are restricted to be a function of groups, which limits the ability of training future classifiers on top of more flexible representations.

Recently, Zhang et al. [37] present a novel Auto-Encoding Transformation (AET) model by learning a representation from which an input transformation can be re-

constructed. This is closely related to our motivation of learning transformation equivariant representations, considering the transformation can be decoded from the learned representation of original and transformed images. On the contrary, in this paper, we approach it from an information-theoretic point of view in a more principled fashion.

Specifically, we will define a joint probability over the representations and transformations, and this will enable us to train unsupervised representations by directly maximizing the mutual information between the transformations and the representations. We wish the resultant representations can generalize to new tasks without access to their labels beforehand.

2.2. Other Unsupervised Representations

Auto-Encoders and GANs. Training auto-encoders in an unsupervised fashion has been studied in literature [18, 19, 35]. Most auto-encoders are trained by minimizing the reconstruction errors on input *data* from the encoded representations. A large category of auto-encoder variants have been proposed. Among them is the Variational Auto-Encoder (VAE) [20] that maximizes the lower-bound of the data likelihood to train a pair of probabilistic encoder and decoder, while beta-VAE seeks to disentangle representations by introducing an adjustable hyperparameter on the capacity of latent channel to balance between the independence constraint and the reconstruction accuracy [15]. Denoising auto-encoder [35] seeks to reconstruct noise-corrupted data to learn robust representation, while contrastive Auto-Encoder [33] encourages to learn representations invariant to small perturbations on data. Along this line, Hinton et al. [16] propose capsule nets by minimizing the discrepancy between the reconstructed and target data.

Meanwhile, Generative Adversarial Nets (GANs) have also been used to train unsupervised representations in literature. Contrary to the auto-encoders, a GAN model generates data from the noises drawn from a simple distribution, with a discriminator trained adversarially to distinguish between real and fake data. The sampled noises can be viewed as the representation of generated data over a manifold, and one can train an encoder by inverting the generator to find the generating noise. This can be implemented by jointly training a pair of mutually inverse generator and encoder [10, 12]. There also exist better generalizable GANs in producing unseen data based on the Lipschitz assumption on the real data distribution [30, 3], which can give rise to more powerful representations of data out of training examples [10, 12, 13]. Compared with the Auto-Encoders, GANs do not rely on learning one-to-one reconstruction of data; instead, they aim to generate the entire distribution of data.

Self-Supervisory Signals. There exist many other unsupervised learning methods using different types of self-

supervised signals to train deep networks. Mehdi and Favaro [25] propose to solve Jigsaw puzzles to train a convolutional neural network. Doersch et al. [9] train the network by predicting the relative positions between sampled patches from an image as self-supervised information. Instead, Noroozi et al. [26] count features that satisfy equivalence relations between downsampled and tiled images, while Gidaris et al. [14] classify a discrete set of image rotations to train deep networks. Dosovitskiy et al. [11] create a set of surrogate classes by applying various transformations to individual images. However, the resultant features could over-discriminate visually similar images as they always belong to different surrogate classes. Unsupervised features have also been learned from videos by estimating the self-motion of moving objects between consecutive frames [2].

3. Formulation

We begin with the notations for the proposed unsupervised learning of the transformation equivariant representations (TERs). Consider a random sample \mathbf{x} drawn from the data distribution $p(\mathbf{x})$. We sample a transformation \mathbf{t} from a distribution $p(\mathbf{t})$, and apply it to \mathbf{x} , yielding a transformed image $\mathbf{t}(\mathbf{x})$.

Usually, we consider a distribution $p(\mathbf{t})$ of parameterized transformations, e.g., affine transformations with the rotations, translations and shearing constants being sampled from a simple distribution, and projective transformations that randomly shift and interpolate four corners of images. Our goal is to learn an unsupervised representation that contains as much information as possible to recover the transformation. We wish such a representation is able to compactly encode images such that it could equivary as the visual structures of images are transformed.

Specifically, we seek to learn an encoder that maps a transformed sample $\mathbf{t}(\mathbf{x})$ to the mean f_θ and variance σ_θ of a desired representation. This results in the following probabilistic representation \mathbf{z} of $\mathbf{t}(\mathbf{x})$:

$$\mathbf{z} = f_\theta(\mathbf{t}(\mathbf{x})) + \sigma_\theta(\mathbf{t}(\mathbf{x})) \circ \epsilon \quad (1)$$

where ϵ is sampled from a normal distribution $\mathcal{N}(\epsilon|\mathbf{0}, \mathbf{I})$, and \circ denotes the element-wise product. In this case, the probabilistic representation \mathbf{z} follows a normal distribution $p_\theta(\mathbf{z}|\mathbf{t}, \mathbf{x}) \triangleq \mathcal{N}(\mathbf{z}|f_\theta(\mathbf{t}(\mathbf{x})), \sigma_\theta^2(\mathbf{t}(\mathbf{x})))$ conditioned on the randomly sampled transformation \mathbf{t} and input data \mathbf{x} . Meanwhile, the representation $\tilde{\mathbf{z}}$ of the original sample \mathbf{x} can be computed as a special case when \mathbf{t} is set to an identity transformation.

As discussed in Section 1, we seek to learn a representation \mathbf{z} equivariant to the sampled transformation \mathbf{t} whose information thus should be recovered from \mathbf{z} as much as possible. Therefore, the most natural choice to formalize this notion of transformation equivariance is the mutual in-

formation $I(\mathbf{t}, \mathbf{z}|\tilde{\mathbf{z}})$ between \mathbf{z} and \mathbf{t} from an information-theoretic perspective. The larger the mutual information, the more knowledge about \mathbf{t} can be inferred from the representation \mathbf{z} that equivaries to \mathbf{t} .

Moreover, it can be shown that the mutual information $I(\mathbf{t}; \mathbf{z}|\tilde{\mathbf{z}})$ is the lower bound of the joint mutual information $I(\mathbf{z}; (\mathbf{t}, \tilde{\mathbf{z}}))$ that attains its maximum value when $I(\mathbf{z}; \mathbf{x}|\tilde{\mathbf{z}}, \mathbf{t}) = 0$. In this case, \mathbf{x} provides no additional information about \mathbf{z} once $(\tilde{\mathbf{z}}, \mathbf{t})$ are given. This implies one can estimate \mathbf{z} directly from $(\tilde{\mathbf{z}}, \mathbf{t})$ without accessing the original sample \mathbf{x} , which generalizes the *linear* transformation equivariance to *nonlinear* case. For more details, we refer the readers to the long version [31] and the supplementary material of this paper, where we elaborate on such generalized transformation-equivariant representations.

Therefore, we maximize the mutual information between the representation and the transformation to train the model

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

Unfortunately, this maximization problem requires us to evaluate the posterior $p_{\theta}(\mathbf{t}|\mathbf{z}, \tilde{\mathbf{z}})$ of the transformation, which is often difficult to compute directly. This makes it intractable to train the representation by directly maximizing the above mutual information. Thus, we will turn 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 variational approach.

4. Autoencoding Variational Transformations

First, we present a variational lower bound of the mutual information $I(\mathbf{t}; \mathbf{z}|\mathbf{x})$ that can be maximized over q_{ϕ} in a tractable fashion.

Instead of lower-bounding data likelihood in other variational approaches such as variational auto-encoders [20], it is more natural for us to maximize the lower bound of the mutual information [1] between the representation \mathbf{z} and the transformation \mathbf{t} in the following way

$$\begin{aligned} I(\mathbf{t}; \mathbf{z}|\tilde{\mathbf{z}}) &= H(\mathbf{t}|\tilde{\mathbf{z}}) - H(\mathbf{t}|\mathbf{z}, \tilde{\mathbf{z}}) \\ &= H(\mathbf{t}|\tilde{\mathbf{z}}) + \mathbb{E}_{p_{\theta}(\mathbf{t}, \mathbf{z}, \tilde{\mathbf{z}})} \log p_{\theta}(\mathbf{t}|\mathbf{z}, \tilde{\mathbf{z}}) \\ &= H(\mathbf{t}|\tilde{\mathbf{z}}) + \mathbb{E}_{p_{\theta}(\mathbf{t}, \mathbf{z}, \tilde{\mathbf{z}})} \log q_{\phi}(\mathbf{t}|\mathbf{z}, \tilde{\mathbf{z}}) \\ &+ \mathbb{E}_{p(\mathbf{z}, \tilde{\mathbf{z}})} D(p_{\theta}(\mathbf{t}|\mathbf{z}, \tilde{\mathbf{z}}) \| q_{\phi}(\mathbf{t}|\mathbf{z}, \tilde{\mathbf{z}})) \\ &\geq H(\mathbf{t}|\tilde{\mathbf{z}}) + \mathbb{E}_{p_{\theta}(\mathbf{t}, \mathbf{z}, \tilde{\mathbf{z}})} \log q_{\phi}(\mathbf{t}|\mathbf{z}, \tilde{\mathbf{z}}) \triangleq \tilde{I}_{\theta, \phi}(\mathbf{t}; \mathbf{z}|\tilde{\mathbf{z}}) \end{aligned}$$

where $H(\cdot)$ denotes the (conditional) entropy, and $D(p_{\theta}(\mathbf{t}|\mathbf{z}, \tilde{\mathbf{z}}) \| q_{\phi}(\mathbf{t}|\mathbf{z}, \tilde{\mathbf{z}}))$ is the Kullback divergence between p_{θ} and q_{ϕ} , which is always nonnegative.

We choose to maximize the lower variational bound $\tilde{I}(\mathbf{t}; \mathbf{z}|\tilde{\mathbf{z}})$. Since $H(\mathbf{t}|\tilde{\mathbf{z}})$ is independent of the model parameters θ and ϕ , we simply maximize

$$\max_{\theta, \phi} \mathbb{E}_{p_{\theta}(\mathbf{t}, \mathbf{z}, \tilde{\mathbf{z}})} \log q_{\phi}(\mathbf{t}|\mathbf{z}, \tilde{\mathbf{z}}) \quad (2)$$

to learn θ and ϕ under the expectation over $p(\mathbf{t}, \mathbf{z}, \tilde{\mathbf{z}})$.

This variational approach differs from the variational auto-encoders [20]: the latter attempts to lower bound the data loglikelihood, while we instead seek to lower bound the mutual information here. Although both are derived based on an auto-encoder structure, the mutual information has a simpler form of lower bound than the data likelihood – it does not contain an additional Kullback-Leibler divergence term, and thus shall be easier to maximize.

4.1. Algorithm

In practice, given a batch of samples $\{\mathbf{x}^i | i = 1, \dots, n\}$, we first draw a transformation \mathbf{t}^i corresponding to each sample. Then we use the reparameterization (1) to generate the probabilistic representation \mathbf{z}^i with f_{θ} and σ_{θ} as well as a sampled noise ϵ^i .

On the other hand, we use a normal distribution $\mathcal{N}(\mathbf{t} | d_{\phi}(\mathbf{z}, \tilde{\mathbf{z}}), \sigma_{\phi}^2(\mathbf{z}, \tilde{\mathbf{z}}))$ as the decoder $q_{\phi}(\mathbf{t}|\mathbf{z}, \tilde{\mathbf{z}})$, where the mean $d_{\phi}(\mathbf{z}, \tilde{\mathbf{z}})$ and variance $\sigma_{\phi}^2(\mathbf{z}, \tilde{\mathbf{z}})$ are implemented by deep network respectively.

With the above samples, the objective (2) can be approximated as

$$\max_{\theta, \phi} \frac{1}{n} \sum_{i=1}^n \log \mathcal{N}(\mathbf{t}^i | d_{\phi}(\mathbf{z}^i, \tilde{\mathbf{z}}^i), \sigma_{\phi}^2(\mathbf{z}^i, \tilde{\mathbf{z}}^i)) \quad (3)$$

where

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

and

$$\tilde{\mathbf{z}}^i = f_{\theta}(\mathbf{x}^i) + \sigma_{\theta}(\mathbf{x}^i) \circ \tilde{\epsilon}.$$

and $\epsilon^i, \tilde{\epsilon}^i \sim \mathcal{N}(\epsilon | \mathbf{0}, \mathbf{I})$, and $\mathbf{t}^i \sim p(\mathbf{t})$ for each $i = 1, \dots, n$.

4.2. Architecture

As illustrated in Figure 1, we implement the transformation decoder $q_{\phi}(\mathbf{t}|\mathbf{z}, \tilde{\mathbf{z}})$ by using a Siamese encoder network with shared weights to represent the original and transformed images with $\tilde{\mathbf{z}}$ and \mathbf{z} respectively, where the mean d_{ϕ} and the variance σ_{ϕ}^2 of the sampled transformation are predicted from the concatenation of both representations.

We note that, in a conventional auto-encoder, error signals must be backpropagated through a deeper decoder to reconstruct images before they train the encoder of interest. In contrast, the AVT allows a shallower decoder to estimate transformations with fewer variables so that stronger training signals can reach the encoder before it attenuates remarkably. This can more sufficiently train the encoder to represent images in downstream tasks.

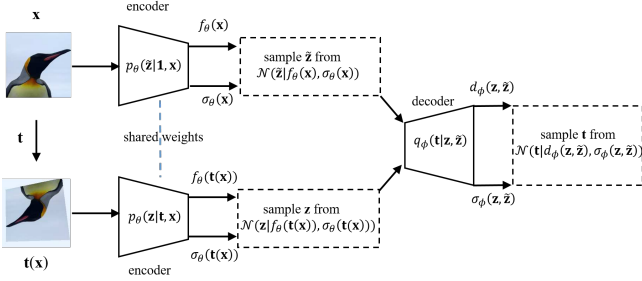


Figure 1: The architecture of the proposed AVT. The original and transformed images are fed through the encoder p_θ where $\mathbf{1}$ denotes an identity transformation to generate the representation of the original image. The resultant representations $\tilde{\mathbf{z}}$ and \mathbf{z} of original and transformed images are sampled and fed into the transformation decoder q_ϕ from which the transformation \mathbf{t} is sampled.

5. Experiments

In this section, we evaluate the proposed AVT model by following the standard protocol in literature.

5.1. CIFAR-10 Experiments

We evaluate the AVT model on the CIFAR-10 dataset.

5.1.1 Experiment Settings

Architecture For a fair comparison with existing models, the Network-In-Network (NIN) is adopted on the CIFAR-10 dataset for the unsupervised learning task [37]. The NIN consists of four convolutional blocks, each containing three convolutional layers. The AVT has two NIN branches, each of which takes the original and transformed images as its input, respectively. We average-pool and concatenate the output feature maps from the fourth block of two branches to form a 384-d feature vector. Then an output layer follows to output the mean d_ϕ and the log-of-variance $\log \sigma_\phi^2$ of the predicted transformation, with the logarithm scaling the variance to a real value.

The two branches share the same network weights, with the first two blocks of each branch being used as the encoder network to directly output the mean f_θ of the representation. An additional 1×1 convolution followed by a batch normalization layer is added on top of the representation mean to output the log-of-variance $\log \sigma_\theta^2$.

Implementation Details The AVT networks are trained by the SGD with a batch size of 512 images and their transformed versions. Momentum and weight decay are set to 0.9 and 5×10^{-4} , respectively. The model is trained for a total of 4,500 epochs. The learning rate is initialized to 10^{-3} . Then it is gradually decayed to 10^{-5} from 3,000 epochs after it is increased to 5×10^{-3} after the first 50 epochs. The previous research [37] has shown the projective transformation outperforms the affine transformation in

training unsupervised models, and thus we adopt it to train the AVT for a fair comparison. The projective transformation is composed of a random translation of the four corners of an image in both horizontal and vertical directions by ± 0.125 of its height and width, after it is scaled by a factor of $[0.8, 1.2]$ and rotated randomly with an angle from $\{0^\circ, 90^\circ, 180^\circ, 270^\circ\}$.

During training the AVT model, a single representation is randomly sampled from the encoder $p_\theta(\mathbf{z}|\mathbf{t}, \mathbf{x})$, which is fed into the surrogate decoder $q_\phi(\mathbf{t}|\mathbf{x}, \mathbf{z})$. In contrast, to fully exploit the uncertainty of probabilistic representations in training the downstream classification tasks, five random samples are drawn and averaged as the representation of an image used by the classifiers. We found averaging randomly sampled representations outperforms only using the mean of the representation to train the downstream classifiers.

5.1.2 Results

Comparison with Other Methods. A classifier is usually trained upon the representation learned by an unsupervised model to assess the performance. Specifically, on CIFAR-10, the existing evaluation protocol [28, 11, 32, 27, 14, 37] is strictly followed by building a classifier on top of the second convolutional block.

First, we evaluate the classification results by using the AVT features with both model-based and model-free classifiers. For the model-based classifier, we follow [37] by training a non-linear classifier with three Fully-Connected (FC) layers – each of the two hidden layers has 200 neurons with batch-normalization and ReLU activations, and the output layer is a soft-max layer with ten neurons each for an image class. We also test a convolutional classifier upon the unsupervised features by adding a third NIN block whose output feature map is averaged pooled and connected to a linear soft-max classifier.

Table 1 shows the results by the AVT and other models. It compares the AVT with both fully supervised and unsupervised methods on CIFAR-10. The unsupervised AVT with the convolutional classifier almost achieves the same error rate as its fully supervised NIN counterpart with four convolutional blocks (7.75% vs. 7.2%). This remarkable result demonstrates the AVT could greatly close the performance gap with the fully supervised model on CIFAR-10.

We also evaluate the AVT when varying numbers of FC layers and a convolutional classifier are trained on top of unsupervised representations respectively in Table 2. The results show that AVT can consistently achieve the smallest errors no matter which classifiers are used.

Comparison based on Model-free KNN Classifiers. We also test the model-free KNN classifier based on the averaged-pooled feature representations from the second convolutional block. The KNN classifier is model-free without training a classifier from labeled examples. This en-

Table 1: Comparison between unsupervised feature learning methods on CIFAR-10. The fully supervised NIN and the random Init. + conv have the same three-block NIN architecture, but the first is fully supervised while the second is trained on top of the first two blocks that are randomly initialized and stay frozen during training.

Method	Error rate
Supervised NIN [14] (Upper Bound)	7.20
Random Init. + conv [14] (Lower Bound)	27.50
Roto-Scat + SVM [28]	17.7
ExemplarCNN [11]	15.7
DCGAN [32]	17.2
Scattering [27]	15.3
RotNet + non-linear [14]	10.94
RotNet + conv [14]	8.84
AET-affine + non-linear [37]	9.77
AET-affine + conv [37]	8.05
AET + non-linear [37]	9.41
AET + conv [37]	7.82
AVT + non-linear	8.96
AVT + conv	7.75

Table 2: Error rates of different classifiers trained on top of the learned representations on CIFAR 10, where n -FC denotes a classifier with n fully connected layers and conv denotes the third NIN block as a convolutional classifier. Two AET variants are chosen for a fair direct comparison since they are based on the same architecture as the AVT and have outperformed the other unsupervised representations before [37].

	1 FC	2 FC	3 FC	conv
AET-affine [37]	17.16	9.77	10.16	8.05
AET-project [37]	16.65	9.41	9.92	7.82
(Ours) AVT	16.19	8.96	9.55	7.75

Table 3: The comparison of the KNN error rates by different models with varying numbers K of nearest neighbors on CIFAR-10.

K	3	5	10	15	20
AET-affine [37]	24.88	23.29	23.07	23.34	23.94
AET-project [37]	23.29	22.40	22.39	23.32	23.73
(Ours) AVT	22.46	21.62	23.7	22.16	21.51

ables us to make a direct evaluation on the quality of learned features. Table 3 reports the KNN results with varying numbers of nearest neighbors. Again, the AVT outperforms the

compared representations when they are used to calculate K nearest neighbors for classifying images.

Comparison with Small Labeled Data. Finally, we also conduct experiments when a small number of labeled examples are used to train the downstream classifiers on top of the learned representations. This will give us some insight into how the unsupervised representations could help with only few labeled examples. Table 4 reports the results of different models on CIFAR-10. The AVT outperforms the fully supervised models when only a small number of labeled examples (≤ 1000 samples per class) are available. It also performs better than the other unsupervised models in most of cases. Moreover, if we adopt the widely used 13-layer network [23] on CIFAR-10 to train the unsupervised and supervised parts, the error rates can be further reduced significantly particularly when very few labeled examples are used.

5.2. ImageNet Experiments

We further evaluate the performance by AVT on the ImageNet dataset. The AlexNet is used as the backbone to learn the unsupervised features.

5.2.1 Architectures and Training Details

Two AlexNet branches with shared parameters are created with original and transformed images as inputs respectively to train unsupervised AVT. The 4,096-d output features from the second last fully connected layer in two branches are concatenated and fed into the output layer producing the mean and the log-of-variance of eight projective transformation parameters. We still use SGD to train the network, with a batch size of 768 images and the transformed counterparts, a momentum of 0.9, a weight decay of 5×10^{-4} . The initial learning rate is set to 10^{-3} , and it is dropped by a factor of 10 at epoch 300 and 350. The AVT is trained for 400 epochs in total. Finally, the projective transformations are randomly sampled in the same fashion as on CIFAR-10, and the unsupervised representations fed into the classifiers are the average over five sampled representations from the probabilistic encoder.

5.2.2 Results

Table 5 reports the Top-1 accuracies of the compared methods on ImageNet by following the evaluation protocol in [25, 38, 14, 37]. Two settings are adopted for evaluation, where Conv4 and Conv5 mean to train the remaining part of AlexNet on top of Conv4 and Conv5 with the labeled data. All the bottom convolutional layers up to Conv4 and Conv5 are frozen after they are trained in an unsupervised fashion. From the results, in both settings, the AVT model consistently outperforms the other unsupervised models.

We also compare with the fully supervised models that give the upper bound of the classification performance by

Table 4: Error rates on CIFAR-10 when different numbers of samples per class are used to train the downstream classifiers. A third convolutional block is trained with the labeled examples on top of the first two blocks of the NIN (* the 13-layer network) pre-trained with the unlabeled data. We compare with the fully supervised models that are trained with all the labeled examples from scratch.

	20	100	400	1000	5000
Supervised conv	66.34	52.74	25.81	16.53	6.93
Supervised non-linear	65.03	51.13	27.17	16.13	7.92
RotNet + conv [14]	35.37	24.72	17.16	13.57	8.05
AET-project + conv [37]	34.83	24.35	16.28	12.58	7.82
AET-project + non-linear [37]	37.13	25.19	18.32	14.27	9.41
AVT + conv	35.44	24.26	15.97	12.27	7.75
AVT + non-linear	37.62	25.01	17.95	14.14	8.96
AVT + conv (13 layers)*	26.2	18.44	13.56	10.86	6.3

Table 6: Top-1 accuracy with linear layers on ImageNet. AlexNet is used as backbone to train the unsupervised models under comparison. A 1,000-way linear classifier is trained upon various convolutional layers of feature maps that are spatially resized to have about 9,000 elements. Fully supervised and random models are also reported to show the upper and the lower bounds of unsupervised model performances. Only a single crop is used and no dropout or local response normalization is used during testing for the AVT, except the models denoted with * where ten crops are applied to compare results.

Method	Conv1	Conv2	Conv3	Conv4	Conv5
ImageNet labels(Upper Bound)	19.3	36.3	44.2	48.3	50.5
Random (Lower Bound)	11.6	17.1	16.9	16.3	14.1
Random rescaled [21]	17.5	23.0	24.5	23.2	20.6
Context [9]	16.2	23.3	30.2	31.7	29.6
Context Encoders [29]	14.1	20.7	21.0	19.8	15.5
Colorization[39]	12.5	24.5	30.4	31.5	30.3
Jigsaw Puzzles [25]	18.2	28.8	34.0	33.9	27.1
BIGAN [10]	17.7	24.5	31.0	29.9	28.0
Split-Brain [38]	17.7	29.3	35.4	35.2	32.8
Counting [26]	18.0	30.6	34.3	32.5	25.7
RotNet [14]	18.8	31.7	38.7	38.2	36.5
AET-project [37]	19.2	32.8	40.6	39.7	37.7
(Ours) AVT	19.5	33.6	41.3	40.3	39.1
DeepCluster* [5]	13.4	32.3	41.0	39.6	38.2
AET-project* [37]	19.3	35.4	44.0	43.6	42.4
(Ours) AVT*	20.9	36.1	44.4	44.3	43.5

training the whole AlexNet with all labeled data end-to-end. The classifiers of random models are trained on top of Conv4 and Conv5 whose weights are randomly sampled, which set the lower bounded performance. By comparison, the AVT model further closes the performance gap to the full supervised models to 5.5% and 11.3% on Conv4 and Conv5 respectively. This is a relative improvement by 15% and 11% over the previous state-of-the-art AET model.

Moreover, we also follow the testing protocol adopted in

[37] to compare the models by training a 1,000-way linear classifier on top of different numbers of convolutional layers in Table 6. Again, the AVT consistently outperforms all the compared unsupervised models in terms of the Top-1 accuracy.

5.3. Places Experiments

Finally, we evaluate the AVT model on the Places dataset. Table 7 reports the results. Unsupervised models are pretrained on the ImageNet dataset, and a linear logis-

Table 7: Top-1 accuracy on the Places dataset. A 205-way logistic regression classifier is trained on top of various layers of feature maps that are spatially resized to have about 9,000 elements. All unsupervised features are pre-trained on the ImageNet dataset, and then frozen when training the logistic regression classifiers with Places labels. We also compare with fully-supervised networks trained with Places Labels and ImageNet labels, as well as with random models. The highest accuracy values are in bold and the second highest accuracy values are underlined.

Method	Conv1	Conv2	Conv3	Conv4	Conv5
Places labels(Upper Bound)[40]	22.1	35.1	40.2	43.3	44.6
ImageNet labels	22.7	34.8	38.4	39.4	38.7
Random (Lower Bound)	15.7	20.3	19.8	19.1	17.5
Random rescaled [21]	21.4	26.2	27.1	26.1	24.0
Context [9]	19.7	26.7	31.9	32.7	30.9
Context Encoders [29]	18.2	23.2	23.4	21.9	18.4
Colorization[39]	16.0	25.7	29.6	30.3	29.7
Jigsaw Puzzles [25]	<u>23.0</u>	31.9	35.0	34.2	29.3
BIGAN [10]	22.0	28.7	31.8	31.3	29.7
Split-Brain [38]	21.3	30.7	34.0	34.1	32.5
Counting [26]	23.3	33.9	36.3	34.7	29.6
RotNet [14]	21.5	31.0	35.1	34.6	33.7
AET-project [37]	22.1	32.9	<u>37.1</u>	<u>36.2</u>	<u>34.7</u>
AVT	22.3	<u>33.1</u>	37.8	36.7	35.6

Table 5: Top-1 accuracy with non-linear layers on ImageNet. AlexNet is used as backbone to train the unsupervised models. After unsupervised features are learned, nonlinear classifiers are trained on top of Conv4 and Conv5 layers with labeled examples to compare their performances. We also compare with the fully supervised models and random models that give upper and lower bounded performances. For a fair comparison, only a single crop is applied in AVT and no dropout or local response normalization is applied during the testing.

Method	Conv4	Conv5
Supervised from [4](Upper Bound)	59.7	59.7
Random from [25] (Lower Bound)	27.1	12.0
Tracking [36]	38.8	29.8
Context [9]	45.6	30.4
Colorization [39]	40.7	35.2
Jigsaw Puzzles [25]	45.3	34.6
BIGAN [10]	41.9	32.2
NAT [4]	-	36.0
DeepCluster [5]	-	44.0
RotNet [14]	50.0	43.8
AET-project [37]	53.2	47.0
(Ours) AVT	54.2	48.4

tic regression classifier is trained on top of different layers of convolutional feature maps with Places labels. It assess-

es the generalizability of unsupervised features from one dataset to another. The models are still based on AlexNet variants. We compare with the fully supervised models trained with the Places labels and ImageNet labels respectively, as well as with the random networks. The AVT model outperforms the other unsupervised models, except performing slightly worse than Counting [38] with a shallow representation by Conv1 and Conv2.

6. Conclusion

In this paper, we present a novel paradigm of learning representations by Autoencoding Variational Transformations (AVT) instead of reconstructing data as in conventional autoencoders. It aims to maximize the mutual information between the transformations and the representations of transformed images. The intractable maximization problem on mutual information is solved by introducing a transformation decoder to approximate the posterior of transformations through a variational lower bound. This naturally leads to a new probabilistic structure with a representation encoder and a transformation decoder. The resultant representations should contain as much information as possible about the transformations to equivary with them. Experiment results show the AVT representations set new state-of-the-art performances on CIFAR-10, ImageNet and Places datasets, greatly closing the performance gap to the supervised models as compared with the other unsupervised models.

References

- [1] David Barber Felix Agakov. The im algorithm: a variational approach to information maximization. *Advances in Neural Information Processing Systems*, 16:201, 2004. 4
- [2] Pulkit Agrawal, Joao Carreira, and Jitendra Malik. Learning to see by moving. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 37–45, 2015. 3
- [3] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein gan. *arXiv preprint arXiv:1701.07875*, 2017. 3
- [4] Piotr Bojanowski and Armand Joulin. Unsupervised learning by predicting noise. *arXiv preprint arXiv:1704.05310*, 2017. 8
- [5] Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features. *arXiv preprint arXiv:1807.05520*, 2018. 7, 8
- [6] Taco Cohen and Max Welling. Group equivariant convolutional networks. In *International conference on machine learning*, pages 2990–2999, 2016. 1, 2
- [7] Taco S Cohen, Mario Geiger, and Maurice Weiler. Intertwiners between induced representations (with applications to the theory of equivariant neural networks). *arXiv preprint arXiv:1803.10743*, 2018. 1
- [8] Taco S Cohen and Max Welling. Steerable cnns. *arXiv preprint arXiv:1612.08498*, 2016. 2
- [9] Carl Doersch, Abhinav Gupta, and Alexei A Efros. Unsupervised visual representation learning by context prediction. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1422–1430, 2015. 3, 7, 8
- [10] Jeff Donahue, Philipp Krähenbühl, and Trevor Darrell. Adversarial feature learning. *arXiv preprint arXiv:1605.09782*, 2016. 3, 7, 8
- [11] Alexey Dosovitskiy, Jost Tobias Springenberg, Martin Riedmiller, and Thomas Brox. Discriminative unsupervised feature learning with convolutional neural networks. In *Advances in Neural Information Processing Systems*, pages 766–774, 2014. 3, 5, 6
- [12] Vincent Dumoulin, Ishmael Belghazi, Ben Poole, Olivier Mastropietro, Alex Lamb, Martin Arjovsky, and Aaron Courville. Adversarially learned inference. *arXiv preprint arXiv:1606.00704*, 2016. 3
- [13] Marzieh Edraki and Guo-Jun Qi. Generalized loss-sensitive adversarial learning with manifold margins. In *Proceedings of European Conference on Computer Vision (ECCV 2018)*, 2018. 3
- [14] Spyros Gidaris, Praveer Singh, and Nikos Komodakis. Unsupervised representation learning by predicting image rotations. *arXiv preprint arXiv:1803.07728*, 2018. 3, 5, 6, 7, 8
- [15] Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-vae: Learning basic visual concepts with a constrained variational framework. In *International Conference on Learning Representations*, 2017. 3
- [16] Geoffrey E Hinton, Alex Krizhevsky, and Sida D Wang. Transforming auto-encoders. In *International Conference on Artificial Neural Networks*, pages 44–51. Springer, 2011. 1, 2, 3
- [17] Geoffrey E Hinton, Sara Sabour, and Nicholas Frosst. Matrix capsules with em routing. 2018. 2
- [18] Geoffrey E Hinton and Richard S Zemel. Autoencoders, minimum description length and helmholtz free energy. In *Advances in neural information processing systems*, pages 3–10, 1994. 2, 3
- [19] Nathalie Japkowicz, Stephen Jose Hanson, and Mark A Gluck. Nonlinear autoassociation is not equivalent to pca. *Neural computation*, 12(3):531–545, 2000. 2, 3
- [20] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013. 3, 4
- [21] Philipp Krähenbühl, Carl Doersch, Jeff Donahue, and Trevor Darrell. Data-dependent initializations of convolutional neural networks. *arXiv preprint arXiv:1511.06856*, 2015. 7, 8
- [22] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012. 1
- [23] Samuli Laine and Timo Aila. Temporal ensembling for semi-supervised learning. *arXiv preprint arXiv:1610.02242*, 2016. 6
- [24] Jan Eric Lenssen, Matthias Fey, and Pascal Libuschewski. Group equivariant capsule networks. *arXiv preprint arXiv:1806.05086*, 2018. 2
- [25] Mehdi Noroozi and Paolo Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles. In *European Conference on Computer Vision*, pages 69–84. Springer, 2016. 3, 6, 7, 8
- [26] Mehdi Noroozi, Hamed Pirsiavash, and Paolo Favaro. Representation learning by learning to count. In *The IEEE International Conference on Computer Vision (ICCV)*, 2017. 3, 7, 8
- [27] Edouard Oyallon, Eugene Belilovsky, and Sergey Zagoruyko. Scaling the scattering transform: Deep hybrid networks. In *International Conference on Computer Vision (ICCV)*, 2017. 5, 6
- [28] Edouard Oyallon and Stéphane Mallat. Deep roto-translation scattering for object classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2865–2873, 2015. 5, 6
- [29] Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros. Context encoders: Feature learning by inpainting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2536–2544, 2016. 7, 8
- [30] Guo-Jun Qi. Loss-sensitive generative adversarial networks on lipschitz densities. *arXiv preprint arXiv:1701.06264*, 2017. 3
- [31] Guo-Jun Qi. Learning generalized transformation equivariant representations via autoencoding transformations. *arXiv preprint arXiv:1906.08628*, 2019. 2, 4
- [32] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015. 5, 6

- [33] Salah Rifai, Pascal Vincent, Xavier Muller, Xavier Glorot, and Yoshua Bengio. Contractive auto-encoders: Explicit invariance during feature extraction. In *Proceedings of the 28th International Conference on International Conference on Machine Learning*, pages 833–840. Omnipress, 2011. [3](#)
- [34] Sara Sabour, Nicholas Frosst, and Geoffrey E Hinton. Dynamic routing between capsules. In *Advances in Neural Information Processing Systems*, pages 3856–3866, 2017. [1](#), [2](#)
- [35] Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. Extracting and composing robust features with denoising autoencoders. In *Proceedings of the 25th international conference on Machine learning*, pages 1096–1103. ACM, 2008. [2](#), [3](#)
- [36] Xiaolong Wang and Abhinav Gupta. Unsupervised learning of visual representations using videos. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2794–2802, 2015. [8](#)
- [37] Liheng Zhang, Guo-Jun Qi, Liqiang Wang, and Jiebo Luo. Aet vs. aed: Unsupervised representation learning by auto-encoding transformations rather than data. *arXiv preprint arXiv:1901.04596*, 2019. [2](#), [5](#), [6](#), [7](#), [8](#)
- [38] Richard Zhang, Phillip Isola, and Alexei A Efros. Split-brain autoencoders: Unsupervised learning by cross-channel prediction. [6](#), [7](#), [8](#)
- [39] Richard Zhang, Phillip Isola, and Alexei A Efros. Colorful image colorization. In *European Conference on Computer Vision*, pages 649–666. Springer, 2016. [7](#), [8](#)
- [40] Bolei Zhou, Agata Lapedriza, Jianxiong Xiao, Antonio Torralba, and Aude Oliva. Learning deep features for scene recognition using places database. In *Advances in neural information processing systems*, pages 487–495, 2014. [8](#)