MAPLE

Transformation Equivariance vs. Invariance: Unsupervised Learning of Visual Representations

Guo-Jun Qi
guojunq@gmail.com

Laboratory for **MA**chine **P**erception and **LE**arning (**MAPLE**)
&
Futurewei Technologies (Huawei Research USA)

Contents

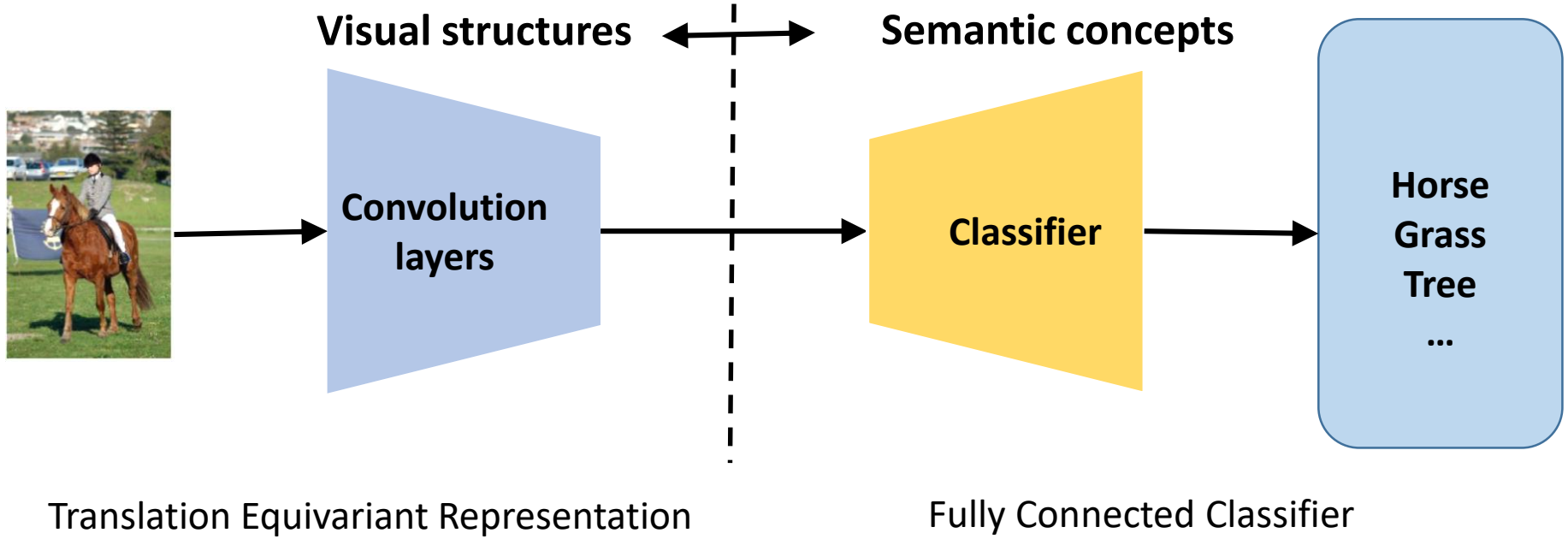
- TER: Transformation Equivariant Representations
  - Definition, Steerability
- AET: AutoEncoding Transformations
  - Deterministic approach: AET (AutoEncoding Transformations)
  - Probabilistic approach: AVT (Autoencoding Variational Transformations)
- SAT: (Semi-)supervised Autoencoding Transformations
- Conclusions and Future Work
  - Unifying the Transformation Equivariance and Invariance

Contents

- TER: Transformation Equivariant Representations
  - Definition, Steerability
- AET: AutoEncoding Transformations
  - Deterministic approach: AET (AutoEncoding Transformations)
  - Probabilistic approach: AVT (Autoencoding Variational Transformations)
- SAT: (Semi-)supervised Autoencoding Transformations
- Conclusions and Future Work

Recipe in Success of CNNs

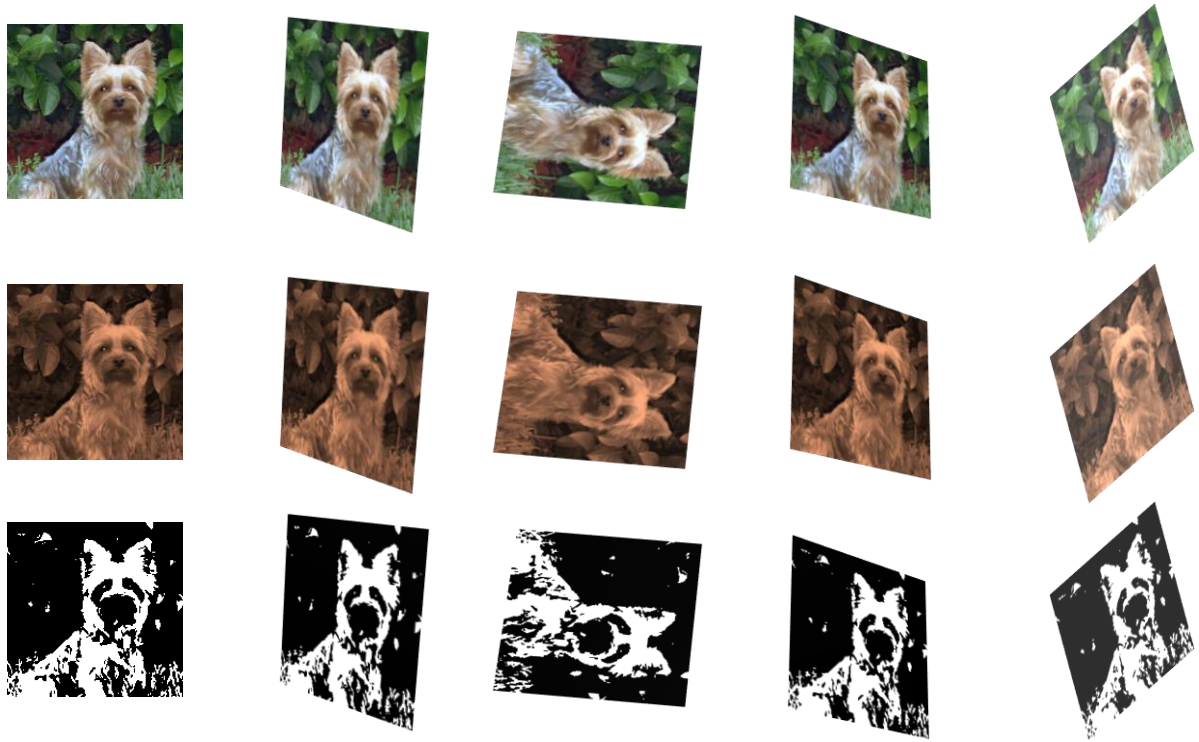
CNN = Translation-Equivariant Representation + Fully-Connected classifier



Transformation Equivariant Representations

- Beyond translations: equivariant feature maps under transformations

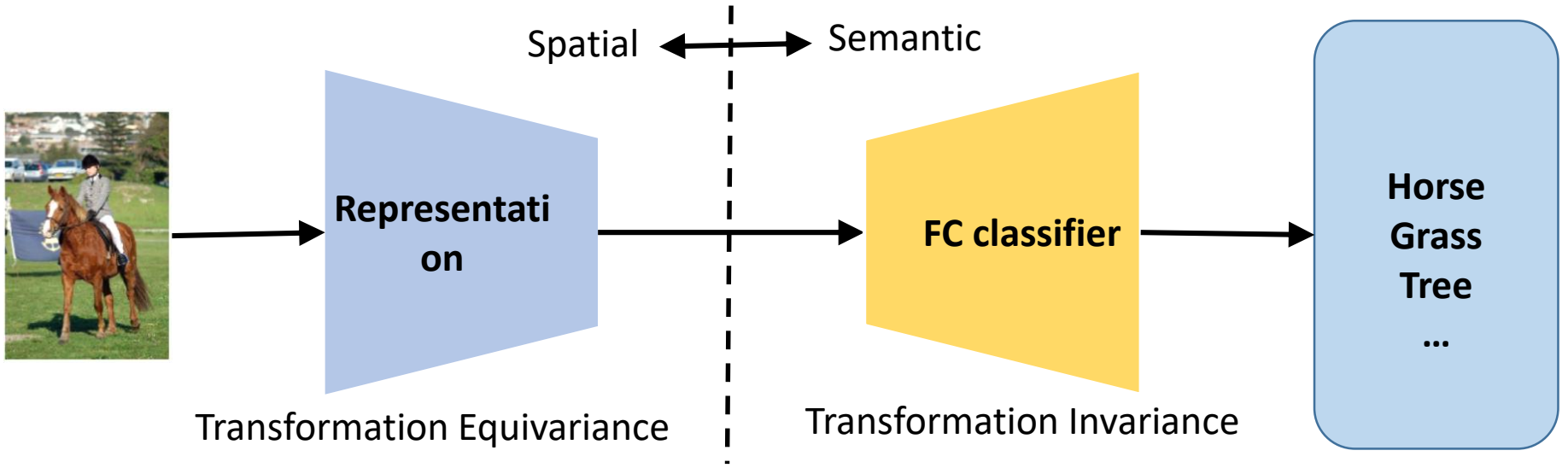
Various transformations



Representations

Generalize CNNs beyond translations

Transformation Equivariant Representations + Transformation Invariant Classifiers



Contents

- TER: Transformation Equivariant Representations
  - Definition, Steerability
- AET: AutoEncoding Transformations
  - Deterministic approach: AET (AutoEncoding Transformations)
  - Probabilistic approach: AVT (Autoencoding Variational Transformations)
- SAT: (Semi-)supervised Autoencoding Transformations
- Conclusions and Future Work

Transformation Equivariance

- Definition of transformation equivariance

$$E(\mathbf{t}(\mathbf{x})) = \boldsymbol{\rho}(\mathbf{t})[E(\mathbf{x})]$$

  - $\boldsymbol{E}$ -- the representation of a sample
  - $\mathbf{t}$ -- a transformation on samples
  - $\boldsymbol{\rho}(\mathbf{t})$ -- the representation transformation corresponding to $\boldsymbol{\rho}$.

- **Transformation invariance** is a special case of transformation equivariance when $\boldsymbol{\rho}(\mathbf{t})$ is an identity.

Steerability property

- Steerability: a transformed sample $\mathbf{t}(\mathbf{x})$ can be represented directly from the representation $E(\mathbf{x})$ of original sample, with no access to $\mathbf{x}$
  - $\boldsymbol{\rho}(\mathbf{t})$ is a function of the transformation $\mathbf{t}$, independently of sample.

$$E(\mathbf{t}(\mathbf{x})) = \boldsymbol{\rho}(\mathbf{t})[E(\mathbf{x})]$$

Our Goals

- For general transformations
  - unnecessarily limited to discrete or spatial transformations
  - Recoloring, contrasting, etc.
- Nonlinear representations between transformed and original images
  - Capturing complex visual structures from transformed images

$$E(\mathbf{t}(\mathbf{x})) = \boldsymbol{\rho}(\mathbf{t})[E(\mathbf{x})]$$

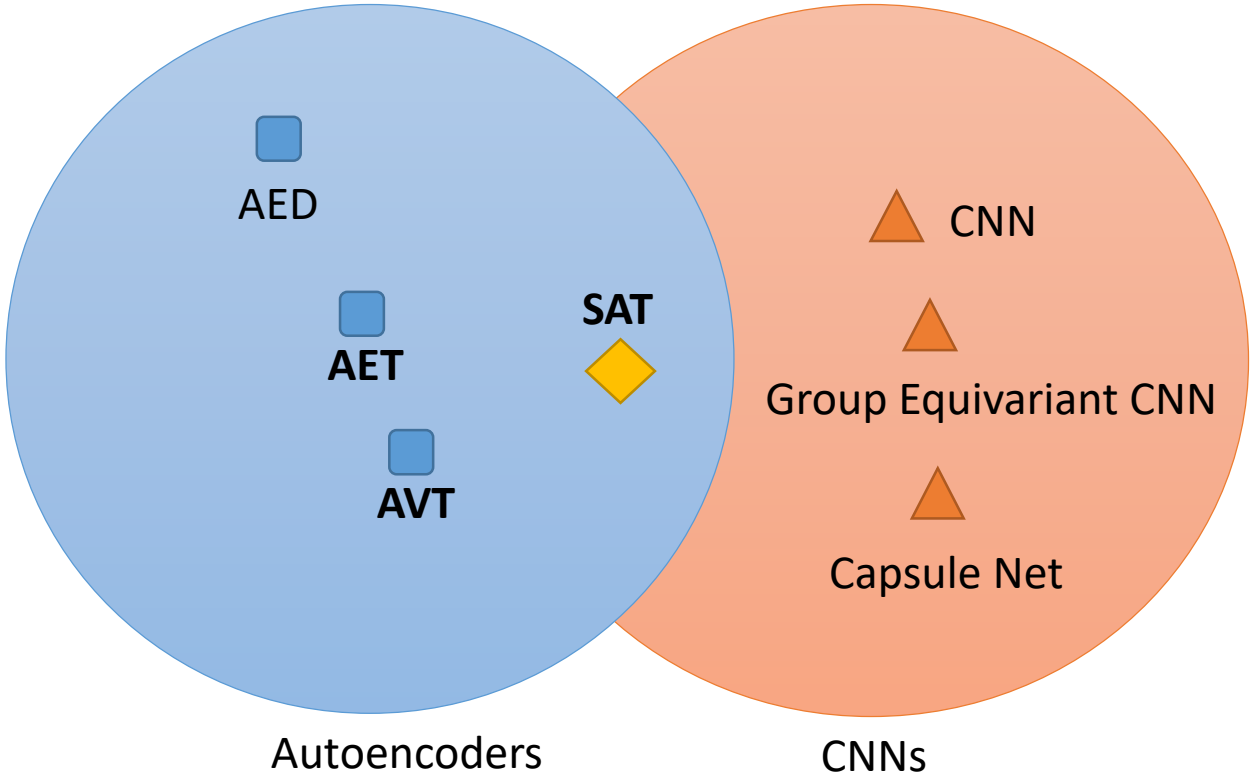
Nonlinear transformations $\boldsymbol{\rho}(\mathbf{t})$ on representations

Contents

- TER: Transformation Equivariant Representations
  - Definition, Steerability
- AET: AutoEncoding Transformations
  - Deterministic approach: AET (AutoEncoding Transformations)
  - Probabilistic approach: AVT (Autoencoding Variational Transformations)
- SAT: (Semi-)supervised Autoencoding Transformations
- Conclusions and Future Work

A Big Picture: Stack of AET

- AET learns a general representation that can be applied everywhere.



Stack of Autoencoding Transformations for learning TER

Deterministic SAT | Probabilistic SAT

Deterministic AET | Probabilistic AVT

Transformation Equivariant Representations

SAT: (Semi-)Supervised Autoencding Transformations

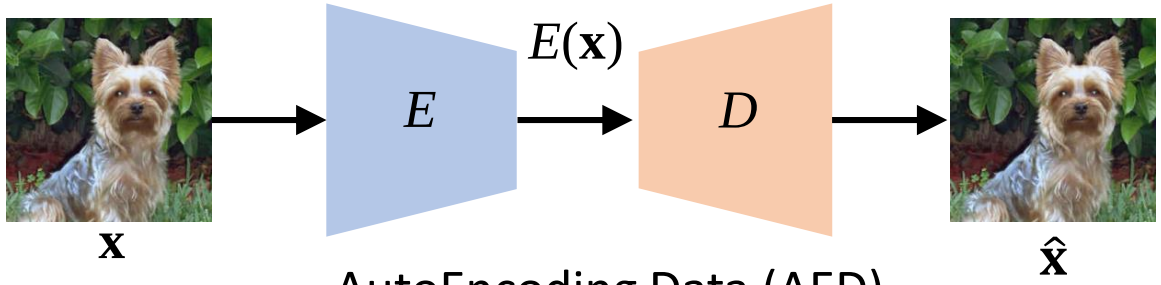
AET: AutoEncoding Transformations

AVT: Autoencoding Variational Transformations

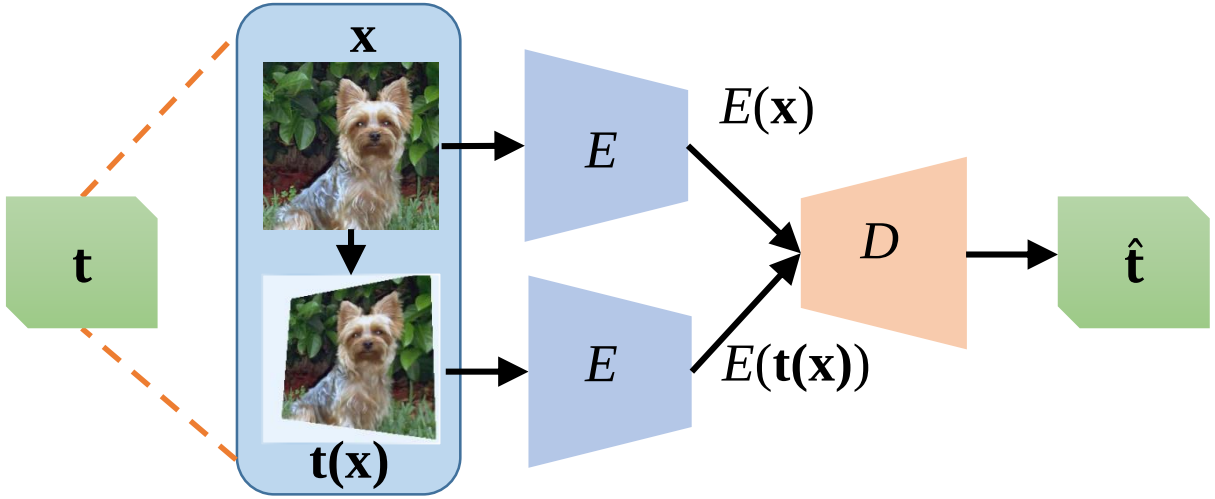
Contents

- TER: Transformation Equivariant Representations
  - Definition, Steerability
- AET: AutoEncoding Transformations
  - Deterministic approach: AET (AutoEncoding Transformations)
  - Probabilistic approach: AVT (Autoencoding Variational Transformations)
- SAT: (Semi-)supervised Autoencoding Transformations
- Conclusions and Future Work

Take A Glance: Autoencoding Transformations than Data



AutoEncoding Data (AED)

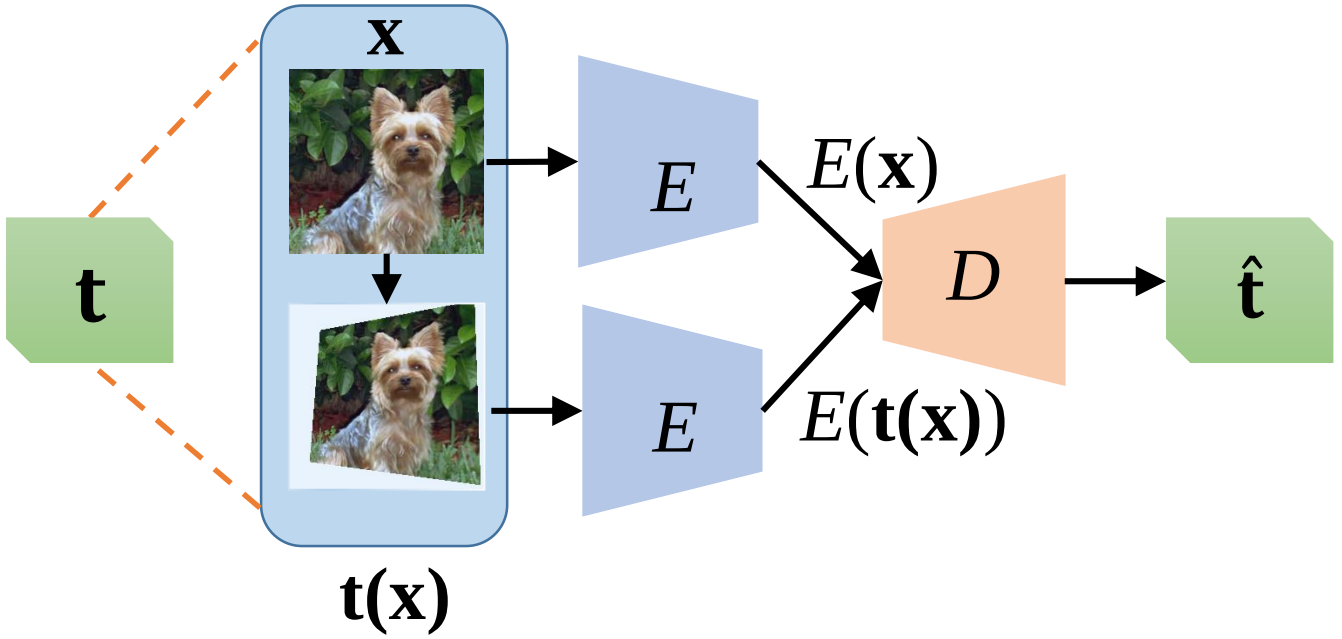


AutoEncoding Transformations (AET)

Zhang et al., AET vs. AED: Unsupervised Representation Learning by Auto-Encoding Transformations rather than Data, in CVPR 2019.

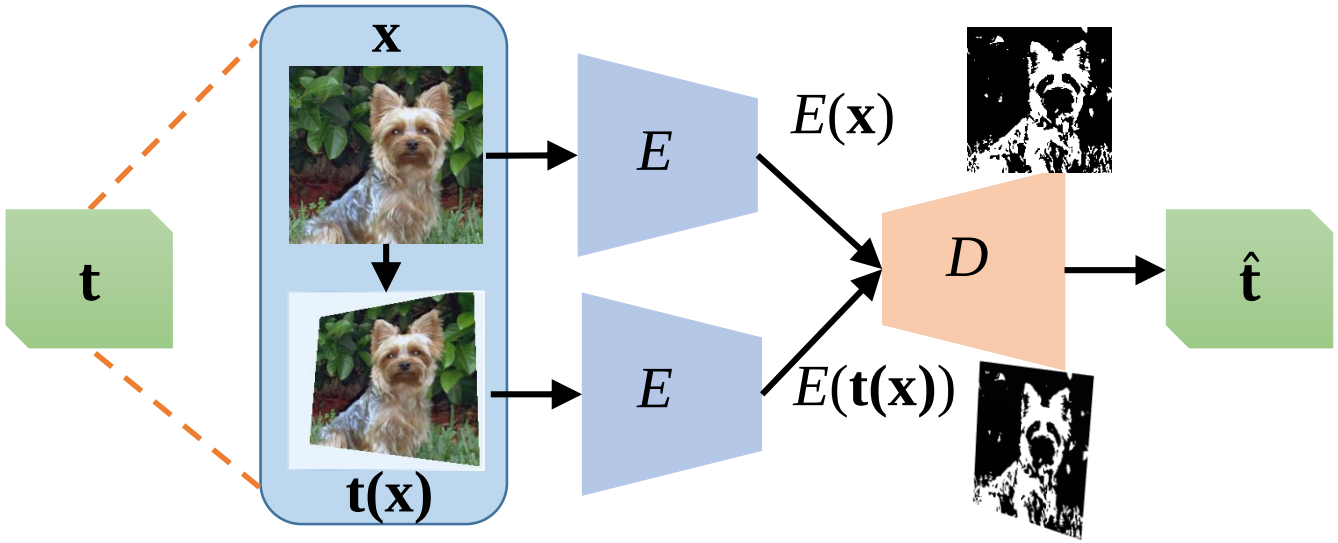
How does AET work?

- Generative Process:
  - An input image $\mathbf{x} \sim p(\mathbf{x})$
  - A random transformation $\mathbf{t} \sim p(\mathbf{t})$
  - The transformed $\mathbf{t(x)}$
  - A representation encoder
    - $\mathbf{E}: \mathbf{x} \mapsto \mathbf{E(x)}, \mathbf{t(x)} \mapsto \mathbf{E(t(x))}$
  - A transformation decoder
    - $\mathbf{D}: (\mathbf{x}, \mathbf{t(x)}) \mapsto \mathbf{D(x}, \mathbf{t(x))}$



Decoding Transformations

- A Siamese network *individually* encodes representations of images
  - E.g., Visual structures, and spatial relations among objects



- Decoding by comparing the representations before and after transformations.

AET loss for training

- Parameterized Transformations: $\mathcal{T}$= {t_θ| $\theta \sim \Theta$}

  E.g. affine or projective: $l\left(\mathbf{t}_\theta, \mathbf{t}_{\hat{\theta}}\right) = \frac{1}{2}\,||M(\theta) - M(\hat{\theta})||_2^2$

- GAN-Induced Transformations: transformed image $\mathrm{G}(\mathbf{x}, \mathbf{z})$

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

- Non-Parametric Transformations

$$l(\mathbf{t}, \hat{\mathbf{t}}) = \frac{1}{2}\,\mathbb{E}_{\mathbf{x}\sim X}\mathrm{dist}(\mathbf{t}(\mathbf{x}), \hat{\mathbf{t}}(\mathbf{x}))$$

Contents

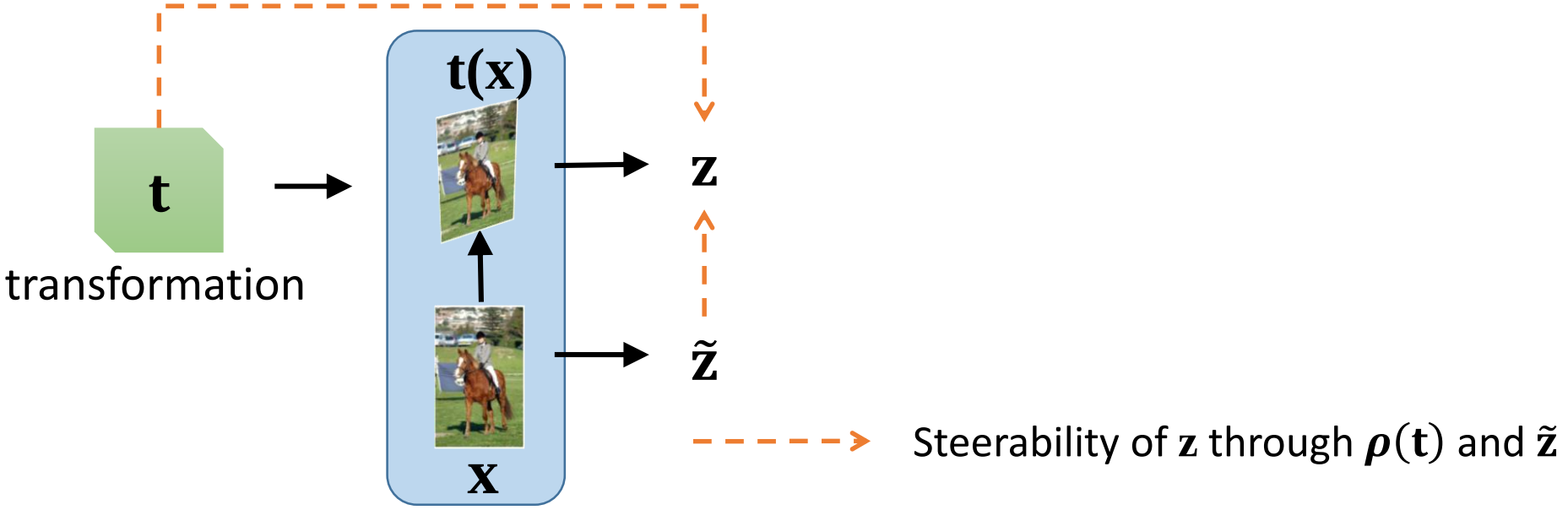
- TER: Transformation Equivariant Representations
  - Definition, Steerability
- AET: AutoEncoding Transformations
  - Deterministic approach: AET (AutoEncoding Transformations)
  - Probabilistic approach: AVT (Autoencoding Variational Transformations)
- SAT: (Semi-)supervised Autoencoding Transformations
- Conclusions and Future Work

Revisit: Steerability of TER

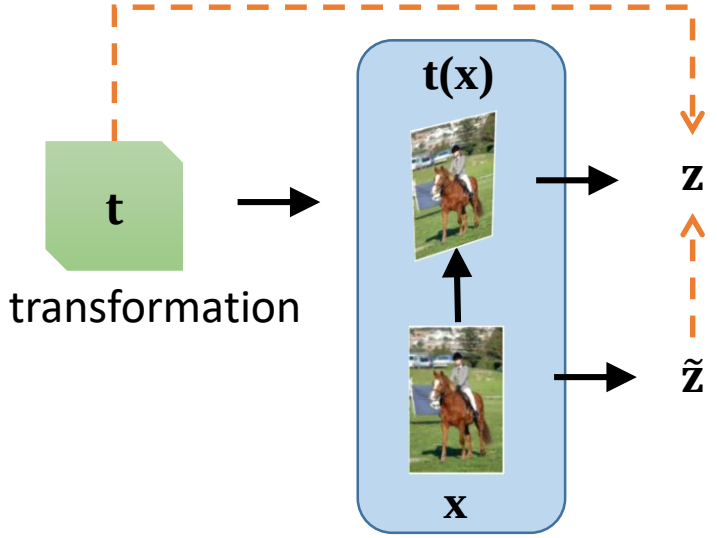
- Obtain the representation $\mathbf{z}$ of a transformed sample $\mathbf{t}(\mathbf{x})$ from $\mathbf{t}$ and $\tilde{\mathbf{z}}$ without accessing $\mathbf{x}$

$$E(\mathbf{t}(\mathbf{x})) = \boldsymbol{\rho}(\mathbf{t})[E(\mathbf{x})]$$

- Maximizing the mutual information between $\mathbf{z}$ and $(\tilde{\mathbf{z}}, \mathbf{t})$



An Information-Theoretical Insight



- Train a TER model $\boldsymbol{\theta}$ by maximizing

$$\max_{\boldsymbol{\theta}} \; \boldsymbol{I_\theta}(\mathbf{z}; \tilde{\mathbf{z}}, \mathbf{t})$$

- By chain rule of mutual information, we have

$$\boldsymbol{I_\theta}(\mathbf{z}; \tilde{\mathbf{z}}, \mathbf{t}) = \boldsymbol{I_\theta}(\mathbf{z}; \tilde{\mathbf{z}}, \mathbf{t}, \mathbf{x}) - \boldsymbol{I_\theta}(\mathbf{z}; \mathbf{x}|\tilde{\mathbf{z}}, \mathbf{t}) \leq \boldsymbol{I_\theta}(\mathbf{z}; \tilde{\mathbf{z}}, \mathbf{t}, \mathbf{x})$$

  - $\boldsymbol{I_\theta}(\mathbf{z}; \tilde{\mathbf{z}}, \mathbf{t})$ attains its maximum value $\boldsymbol{I_\theta}(\mathbf{z}; \tilde{\mathbf{z}}, \mathbf{t}, \mathbf{x})$ (the upper bound) when

$$\boldsymbol{I_\theta}(\mathbf{z}; \mathbf{x}|\tilde{\mathbf{z}}, \mathbf{t}) = \mathbf{0}$$

Steerability: Given $(\tilde{\mathbf{z}}, \mathbf{t})$, $\mathbf{x}$ contains no more information about $\mathbf{z}$.

- **Nonlinearity** of transformation $\boldsymbol{\rho}(\mathbf{t})$ in representations.

AVT: Autoencoding Variational Transformations

- Unable to maximize the mutual information directly
  - Intractable to evaluate the posterior $p_\theta(\mathbf{t}|\mathbf{z},\mathbf{x})$
- Deriving a lower bound by introducing a transformation decoder $q_{\boldsymbol{\phi}}$

$$\boldsymbol{I_\theta}(\mathbf{z};\tilde{\mathbf{z}},\mathbf{t}) \geq H(\mathbf{t}|\tilde{\mathbf{z}}) + \mathbb{E}_{\boldsymbol{p_\theta}(\boldsymbol{t},\boldsymbol{z},\tilde{\mathbf{z}})}\log q_{\boldsymbol{\phi}}(\boldsymbol{t}|\boldsymbol{z},\tilde{\mathbf{z}})$$

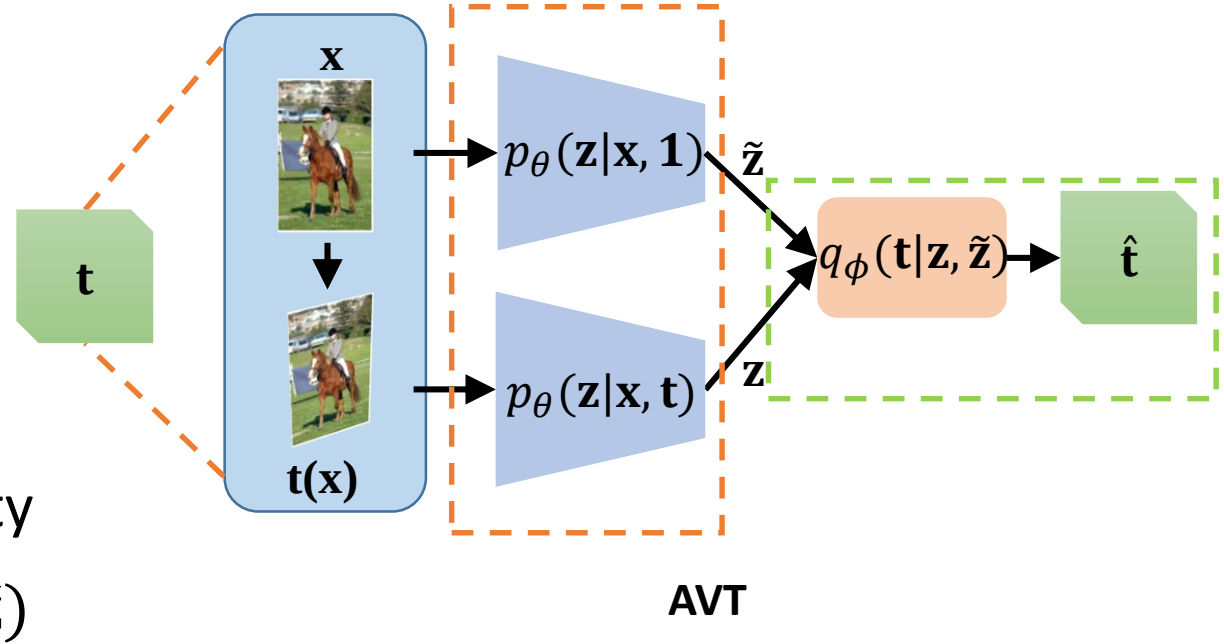
- Unsupervised loss to learn AVT

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

Qi, Learning Generalized Transformation Equivariant Representations via Autoencoding Transformations, preprint.

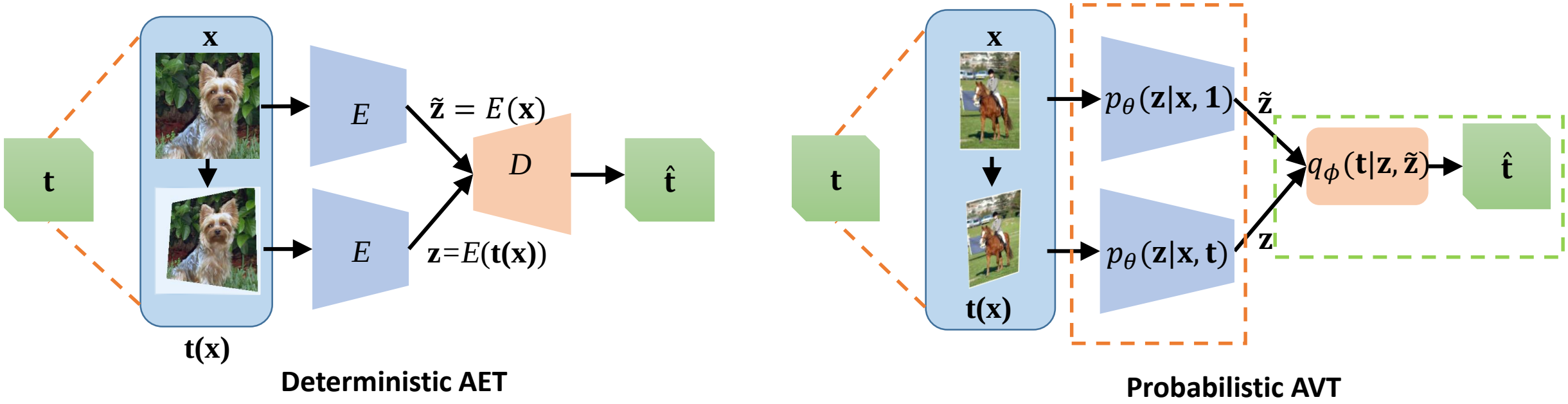
AVT: Autoencoding Variational Transformations

- Generative process
  - Given an image $\mathbf{x}$ sampled from $p(\mathbf{x})$
  - Sample a transformation $\mathbf{t}$ from $p(\mathbf{t})$
  - Apply $\mathbf{t}$ to $\mathbf{x}$, resulting in $\mathbf{t(x)}$
  - Sample a representation $\mathbf{z}$ of $\mathbf{t(x)}$ from $p_\theta(\mathbf{z}|\mathbf{x}, \mathbf{t})$
    - $\tilde{\mathbf{z}}$ is sampled by setting $\mathbf{t}$ to an identity
  - Decode transformations $\hat{\mathbf{t}}$ from $q_\phi(\mathbf{t}|\mathbf{z}, \tilde{\mathbf{z}})$



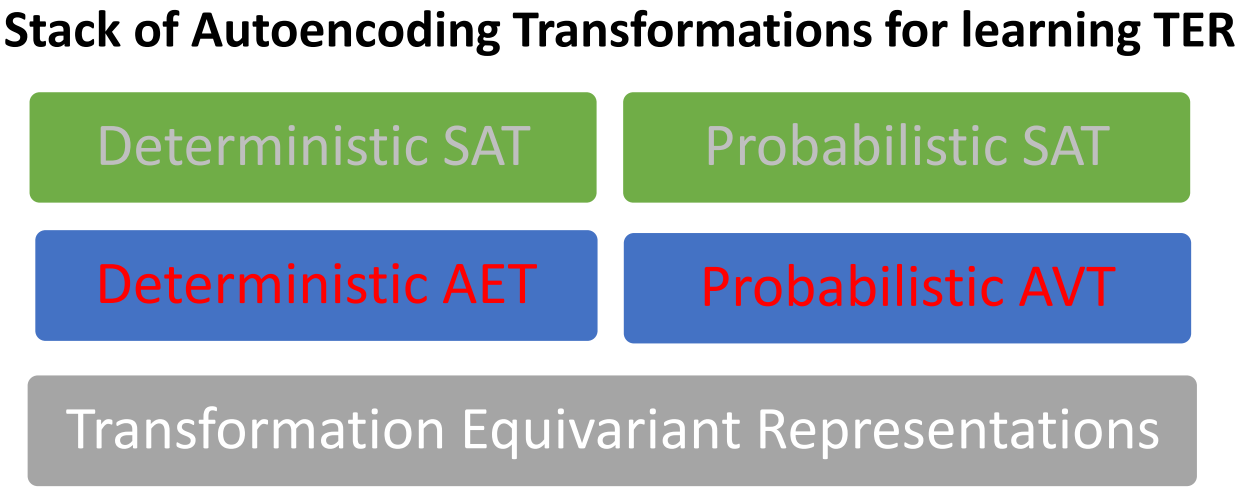
Deterministic vs. Probabilistic Approaches

- AVT can handle uncertainty in
  - A probabilistic Representing of images: $p_\theta(\mathbf{z}|\mathbf{x}, \mathbf{t})$
  - Decoding transformations with a probabilistic decoder: $q_\phi(\mathbf{t}|\mathbf{z}, \tilde{\mathbf{z}})$



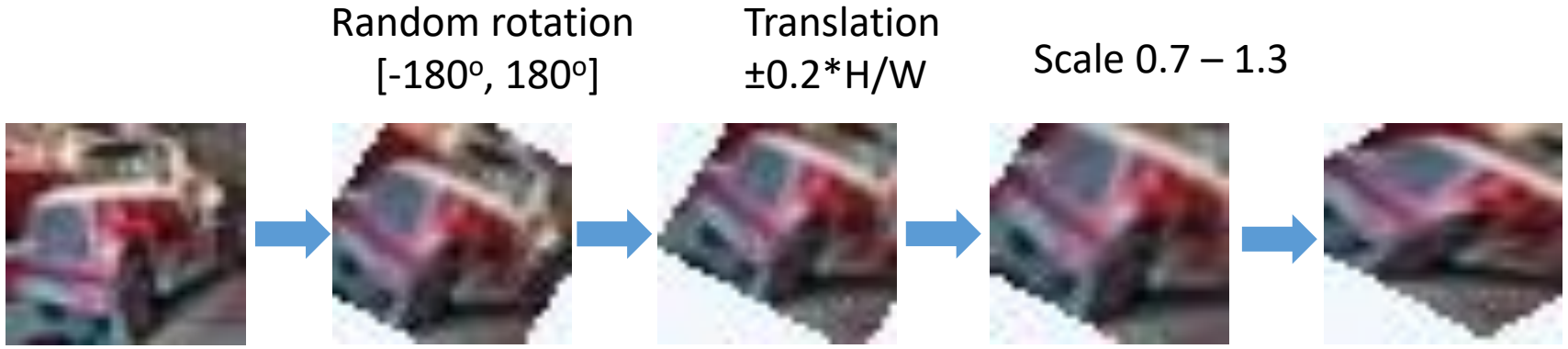
Experiments for AET and AVT

- Unsupervised Representation Learning
  - AET and AVT
- Evaluation Protocols
  - Test learned representations on downstream tasks
- Datasets
  - CIFAR10
  - ImageNet
  - Places

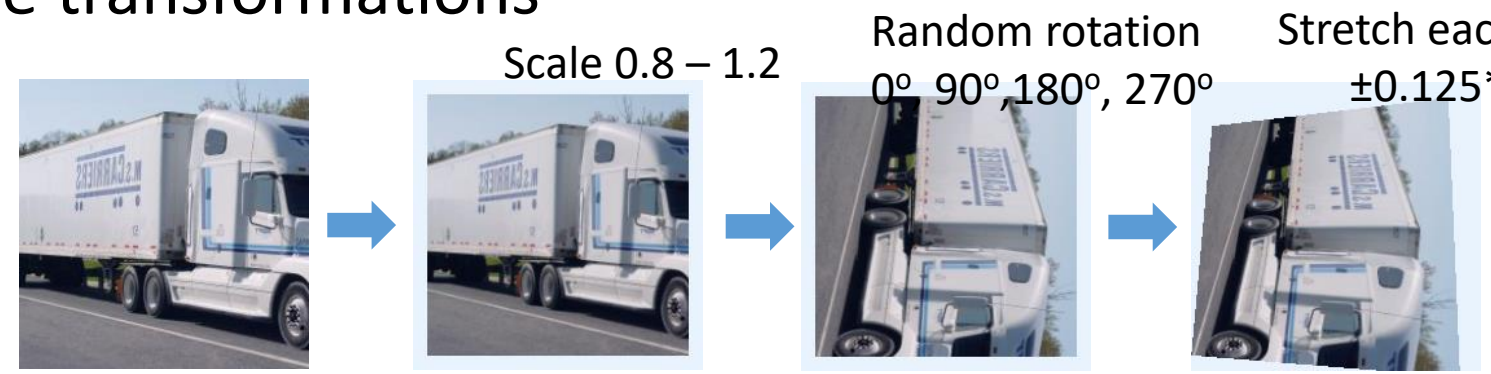


Transformations in Experiments

- Affine transformations



- Projective transformations



CIFAR10 and ImageNet

Error rates on CIFAR10

Method	Error rate
Supervised NIN (Lower Bound)	7.20
Random Init. + conv (upper Bound)	27.50
Roto-Scat + SVM	17.7
ExamplarCNN	15.7
DCGAN	17.2
Scattering	15.3
RotNet + FC	10.94
RotNet + Conv	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**
(ours) AVT-project + FC	**8.96**
(ours) AVT-project + conv	**7.75**

Top 1 Accuracy on ImageNet

Method	Conv4	Conv5
ImageNet Labels (Upper Bound)	59.7	59.7
Random (Lower Bound)	27.1	12.0
Tracking	38.8	29.8
Context	45.6	30.4
Colorization	40.7	35.2
Jigsaw Puzzles	45.3	34.6
BiGAN	41.9	32.2
NAT	-	36.0
DeepCluster	-	44.0
RotNet	50.0	43.8
(Ours) AET-project	**53.2**	**47.0**
(Ours) AVT-project	**54.2**	**48.4**

Projective transformation is slightly better than the affine.

Model-Free KNN on Representations

- Purely test the learned representations without training models
  - Error rates on CIFAR10

K=	3	5	10	15	20
RotNet baseline	25.67	25.01	24.97	25.85	26.00
AET-affine	24.88	23.29	23.07	23.34	23.94
AET-project	23.29	22.40	**22.39**	23.32	23.75
AVT-project	**22.46**	**21.62**	23.70	**22.16**	**21.51**

PASCAL VOC: Object Detection

- A milestone: even better than fully supervised counterpart

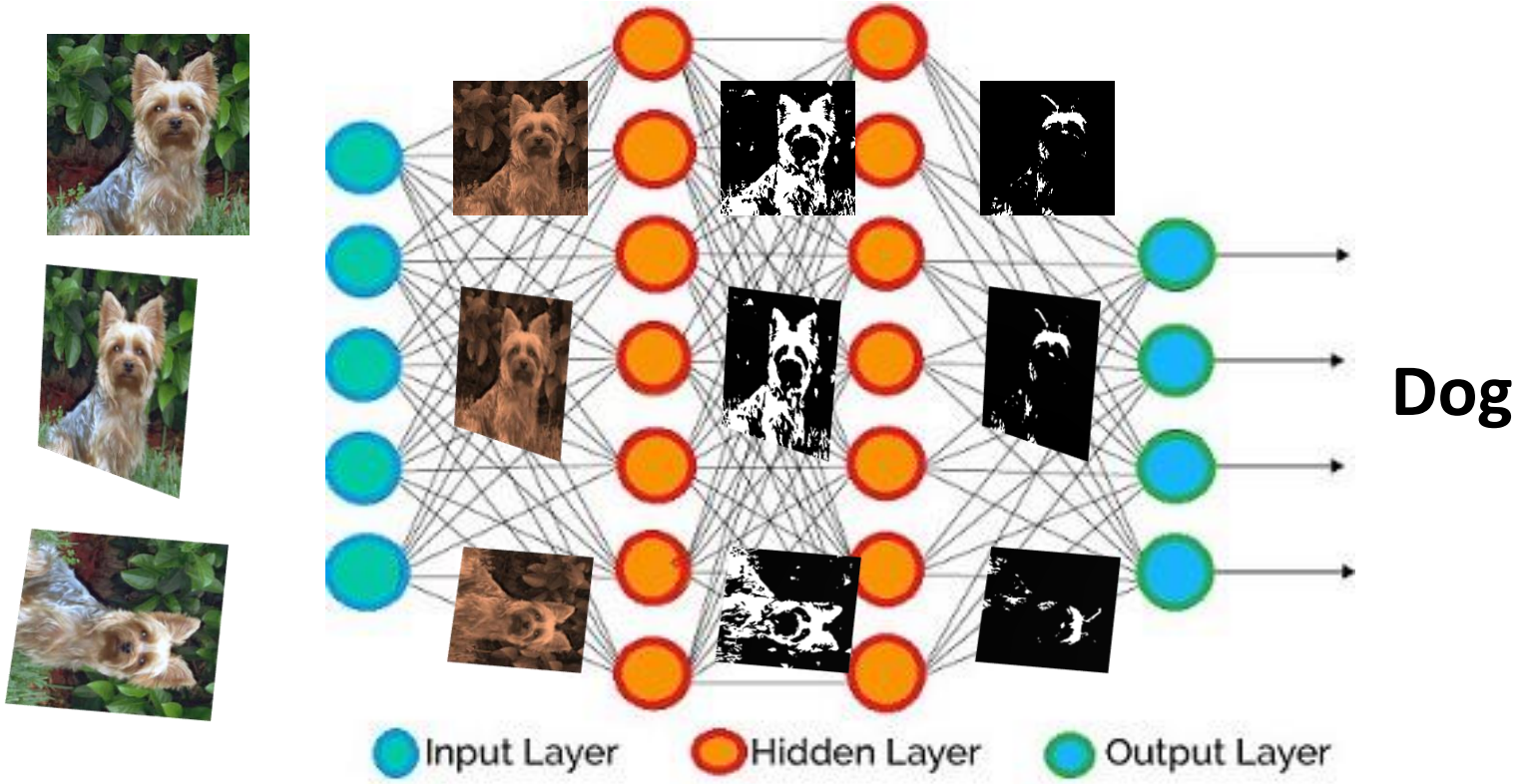
Method	Fast RCNN	Faster RCNN
Supervised Pretraining [45]	56.8	54.3
Context [29]	51.1	-
Colorization [47]	46.9	-
Jigsaw Puzzles [28]	53.2	-
NAT [45]	49.4	-
Split-Brain [51]	46.7	-
Counting [30]	51.4	-
RotNet [31]	54.4	-
IR [55]	48.1	53.1
LA [56]	-	53.5
AET-project	-	**57.4**

Contents

- TER: Transformation Equivariant Representations
  - Definition, Steerability
- AET: AutoEncoding Transformations
  - Deterministic approach: AET (AutoEncoding Transformations)
  - Probabilistic approach: AVT (Autoencoding Variational Transformations)
- SAT: (Semi-)supervised Autoencoding Transformations
- Conclusions and Future Work

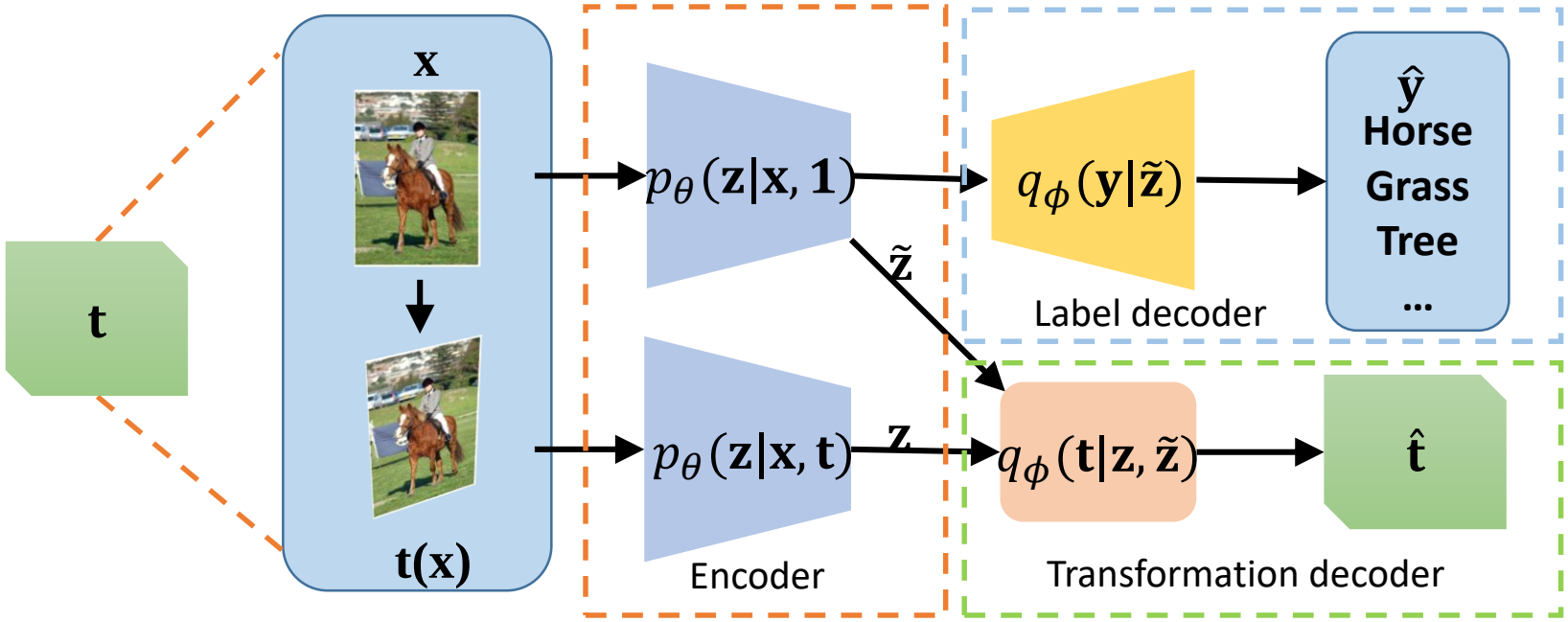
Supervised Autoencoding Transformations

- Beyond Translation Equivariance and CNNs
  - Transformation Equivariant Representations + Transformation Invariant Classifiers



SAT: Semi-Supervised Autoencoding Transformation

- Adding a label decoder $q_\phi(\mathbf{y}|\tilde{\mathbf{z}})$to approximate the posterior $p_\theta(\mathbf{y}|\mathbf{x})$



Qi, Learning Generalized Transformation Equivariant Representations via Autoencoding Transformations, in preprint.

Training SAT

- Seek to maximize the mutual information between $(\mathbf{y}, \mathbf{z})$ and $(\tilde{\mathbf{z}}, \mathbf{t})$

$$\max_{\theta} I_{\theta}(\mathbf{y}, \mathbf{z}; \tilde{\mathbf{z}}, \mathbf{t})$$

  - Assumption: $(\tilde{\mathbf{z}}, \mathbf{t})$ contains ***sufficient*** information to decode both label $\mathbf{y}$ and representation $\mathbf{z}$ of transformed images

- Intractable due to the difficulty in evaluating the posterior $p_{\theta}(\mathbf{y}, \mathbf{t}|\mathbf{z}, \tilde{\mathbf{z}})$

Variational Bound

- By introducing label decoder and transformation decoder, we have

$$I_\theta(\mathbf{y}, \mathbf{z}; \tilde{\mathbf{z}}, \mathbf{t}) \geq \mathbb{E}_{p_\theta(\mathbf{y},\mathbf{z},\tilde{\mathbf{z}})} \log q_\phi(\mathbf{y}|\mathbf{z}, \tilde{\mathbf{z}}) + \mathbb{E}_{p_\theta(\mathbf{t},\mathbf{z},\tilde{\mathbf{z}})} \log q_\phi(\mathbf{t}|\mathbf{z}, \tilde{\mathbf{z}})$$

Label decoder

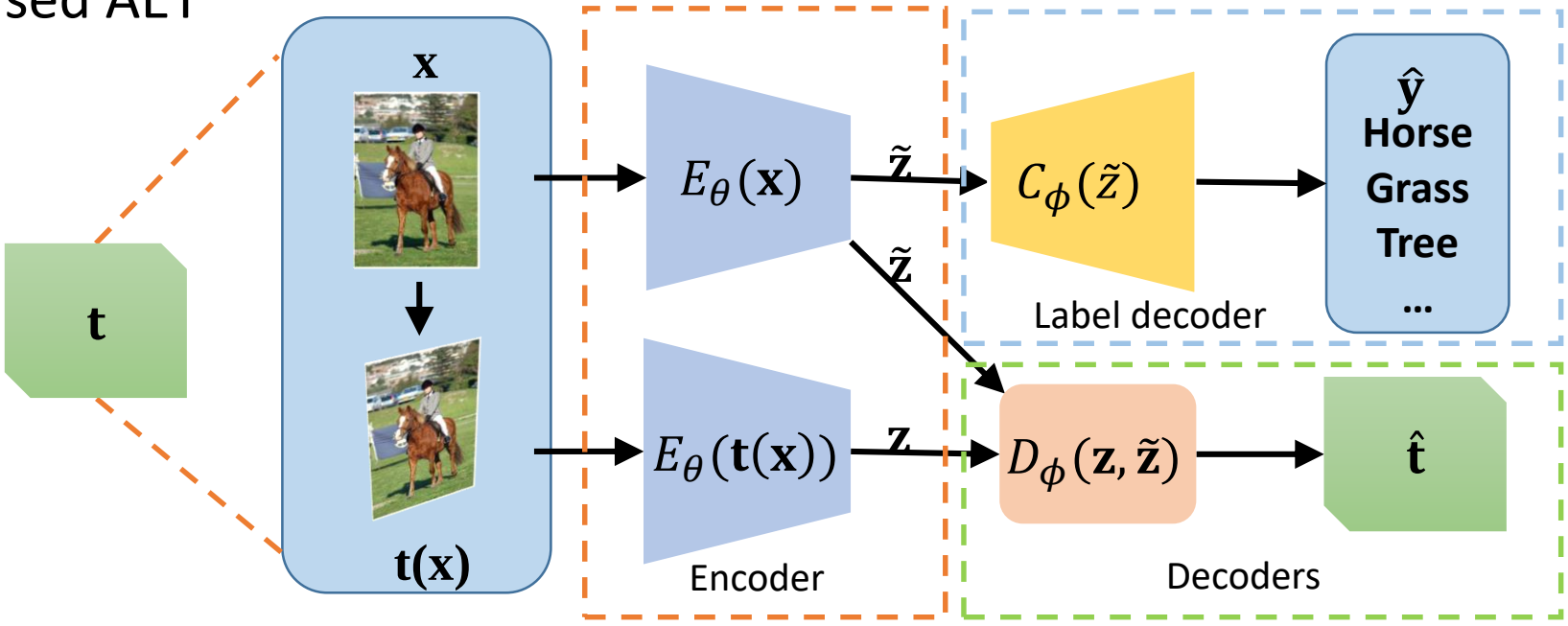
Transformation decoder

- Jointly maximizing over encoder θ and decoders ϕ

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

A Simple Case: Deterministic SAT

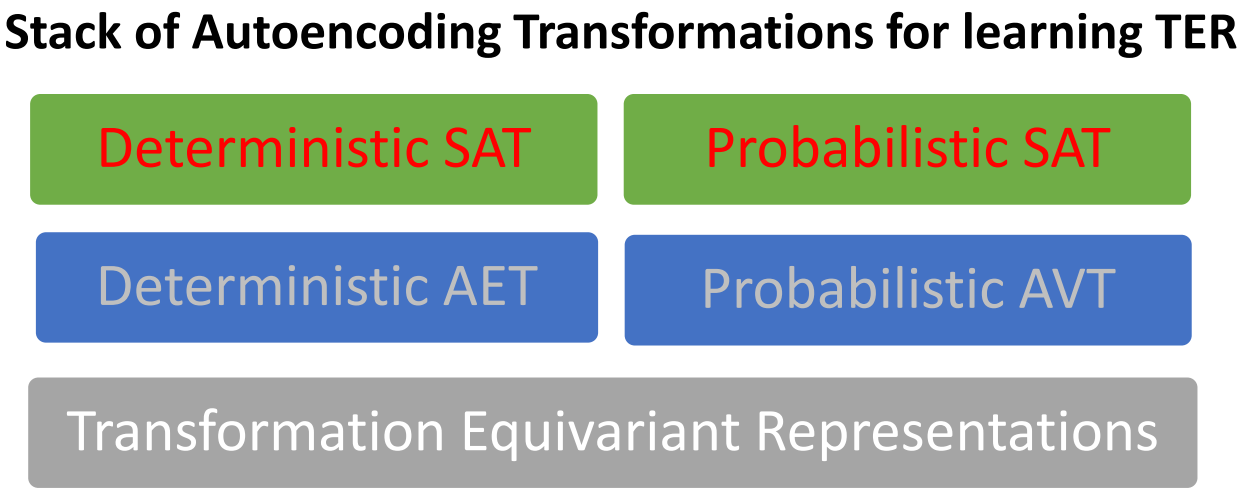
- Replace probabilistic encoder and decoders with deterministic ones
  - Supervised AET



$$\max_{\theta,\phi} \mathbb{E}_{(\mathbf{x},\mathbf{y})} l_{\text{CrossEntropy}}\left(\mathbf{y}, C_\phi(\tilde{\mathbf{z}})\right) + \mathbb{E}_{(\mathbf{t},\mathbf{x})} l_{\text{AET}}\left(\mathbf{t}, D_\phi(\mathbf{z}, \tilde{\mathbf{z}})\right)$$

Experiments on SAT

- (Semi-)supervised learning: SAT
- Evaluation Protocols
  - Follow semi-supervised protocols with varying number of labels
- Datasets
  - CIFAR10
  - SVHN



MAPLE

Error Rate on CIFAR10

Methods	1000 labels	2000 labels	4000 labels	All
GAN			18.63 $\pm$2.32	
Π model			12.36 $\pm$ 0.31	5.56 $\pm$ 0.10
Temporal Ensembling			12.16 $\pm$ 0.31	5.60 $\pm$ 0.10
VAT			10.55	
Supervised-only	46.43 $\pm$ 1.21	33.94 $\pm$ 0.73	20.66 $\pm$ 0.57	5.81 $\pm$ 0.15
Π model	27.36 $\pm$ 1.20	18.02 $\pm$ 0.60	13.20 $\pm$ 0.27	6.06 $\pm$ 0.11
Mean Teacher	21.55 $\pm$ 1.48	15.73 $\pm$ 0.31	12.31 $\pm$ 0.28	5.94 $\pm$ 0.15
SAT	**14.89 $\pm$ 0.38**	**11.71 $\pm$ 0.29**	**9.85 $\pm$ 0.11**	**4.91 $\pm$ 0.13**

Error Rate on SVHN

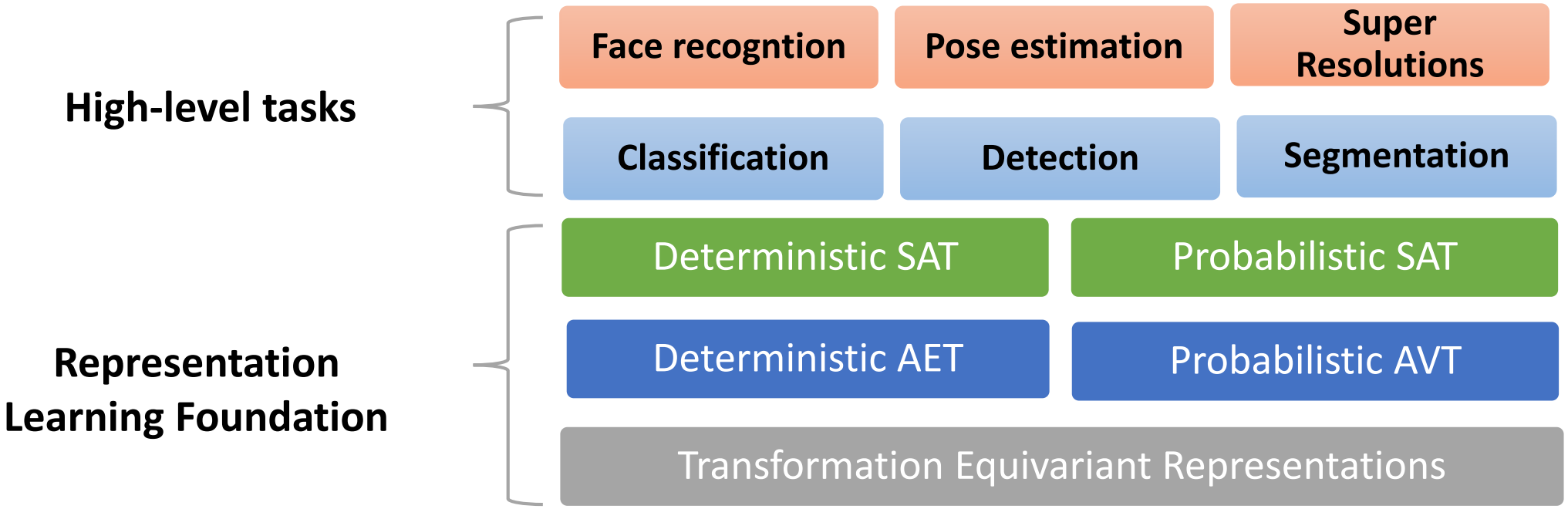
Methods	250 labels	500 labels	1000 labels	All
GAN			18.63 $\pm$2.32	
Π model			12.36 $\pm$ 0.31	5.56 $\pm$ 0.10
Temporal Ensembling			12.16 $\pm$ 0.31	5.60 $\pm$ 0.10
VAT			10.55	
Supervised-only	46.43 $\pm$ 1.21	33.94 $\pm$ 0.73	20.66 $\pm$ 0.57	5.81 $\pm$ 0.15
Π model	27.36 $\pm$ 1.20	18.02 $\pm$ 0.60	13.20 $\pm$ 0.27	6.06 $\pm$ 0.11
Mean Teacher	21.55 $\pm$ 1.48	15.73 $\pm$ 0.31	12.31 $\pm$ 0.28	5.94 $\pm$ 0.15
SAT	**14.89 $\pm$ 0.38**	**11.71 $\pm$ 0.29**	**9.85 $\pm$ 0.11**	**4.91 $\pm$ 0.13**

Contents

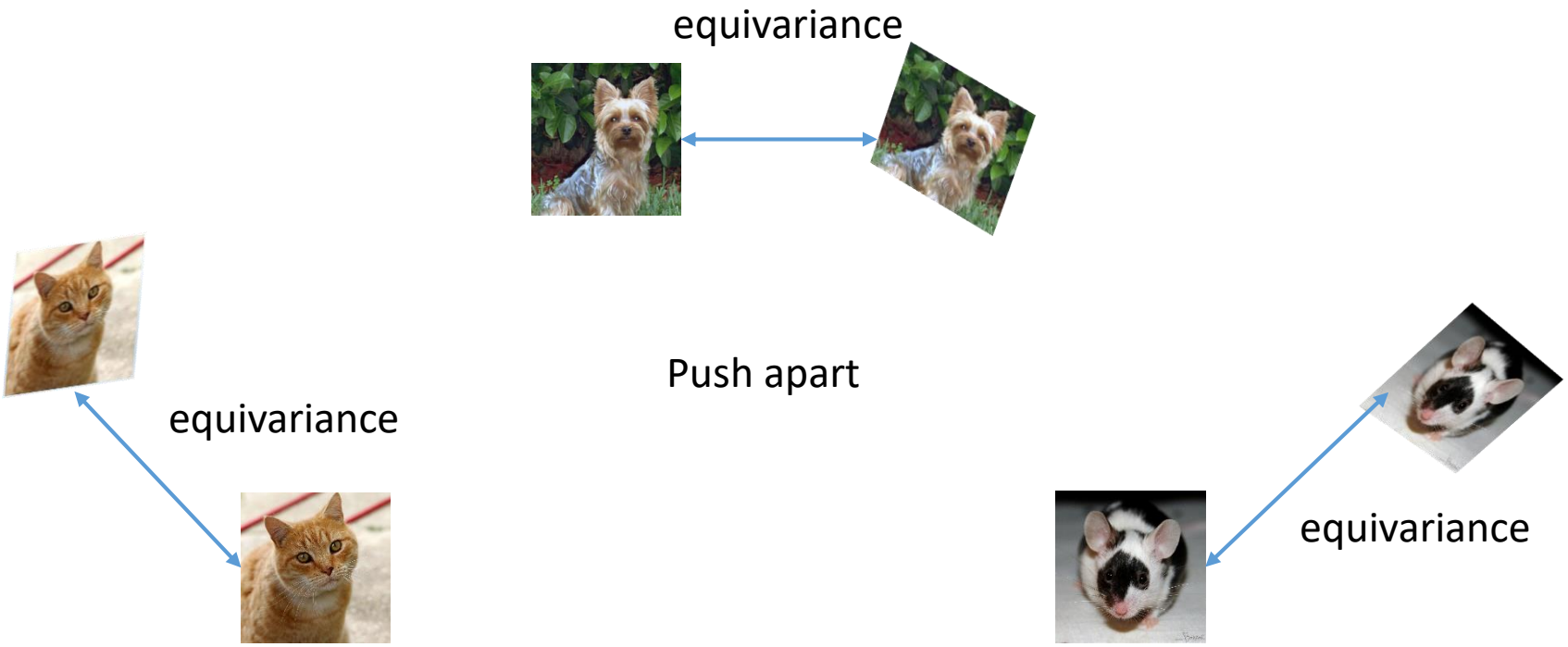
- TER: Transformation Equivariant Representations
  - Definition, Steerability
- AET: AutoEncoding Transformations
  - Deterministic approach: AET (AutoEncoding Transformations)
  - Probabilistic approach: AVT (Autoencoding Variational Transformations)
- SAT: (Semi-)supervised Autoencoding Transformations
- Conclusions and Future Work

Future Works

- Evolutionalize representation learning and various high-level tasks
  - More powerful in capturing the (non-)linear equivariant (invariant) visual structures against transformations

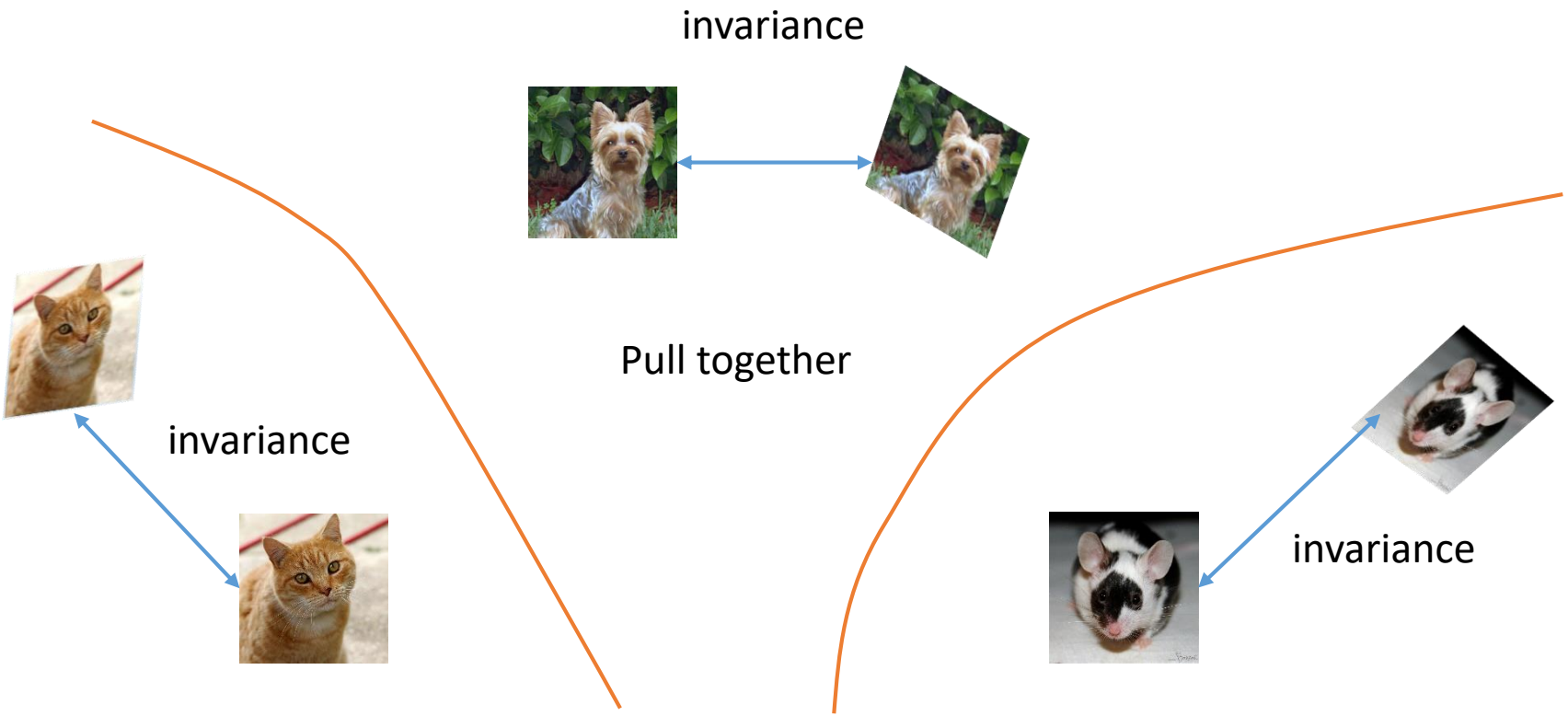


Future Work: Unifying Transformation Equivariance vs. Invariance



Transformation Equivariance discerns the difference between transformed images, thus more advantageous in modeling spatial structures for object detection and semantic segmentation.

Future Work: Unifying Transformation Equivariance vs. Invariance (cont'd)



Transformation invariance (e.g., in contrastive learning) pulls together the embeddings of transformed images, while discriminating between different instances, thus more suitable for image classification tasks.

Future Work: Unifying Transformation Equivariance vs. Invariance (cont'd)

- A naive approach (yet to be done) by combining
  - AET loss for transformation equivariance
  - Contrastive loss for transformation invariance
- Open questions
  - How can both transformation equivariance and invariance reconcile in learning feature embeddings?
  - Will the new approach excel on both spatial sensitive tasks (object detection and semantic segmentation) and image classification tasks?

Guo-Jun Qi and Jiebo Luo, Small Data Challenges in Big Data Era: A Survey of Recent Progress on Unsupervised and Semi-Supervised Methods, arXiv:1903.11260

Conclusions

- Transformation Equivariant Representation
  - Plays key role in the success of
  - Generalizes to any (Continuous, discrete, and beyond spatial) transformations
  - Nonlinear, steerable representations of transformations
- Unsupervised: AutoEncoding Transformations
  - Unsupervised Learning: AET and AVT
  - (Semi-)supervised learning: SAT
- (Semi-)Supervised: Generalizing CNNs beyond translations
  - Transformation Equivariant Representations + Transformation Invariant Classifier
  - With invariance contained in equivariance as a special case

References

Preprint

- *Qi, Learning Generalized Transformation Equivariant Representations via Autoencoding Transformations, in preprint. (long version)*

Conference Publications

- *Zhang et al., AET vs. AED: Unsupervised Representation Learning by Auto-Encoding Transformations rather than Data, in CVPR 2019. (AET)*
- *Qi et al., AVT: Unsupervised Learning of Transformation Equivariant Representations by Autoencoding Variational Transformations, in ICCV 2019. (AVT)*

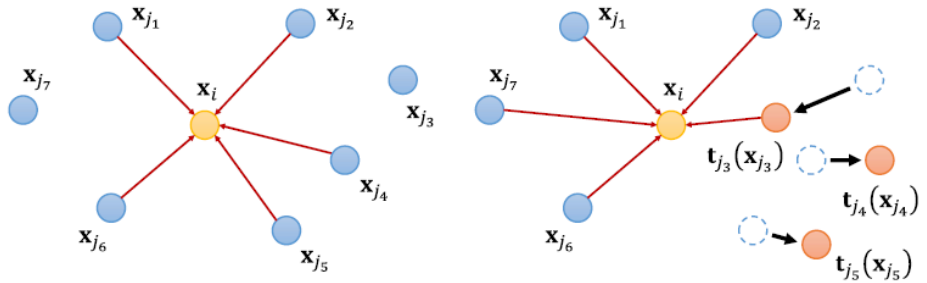
Thank you! Q&A

Recent Progresses

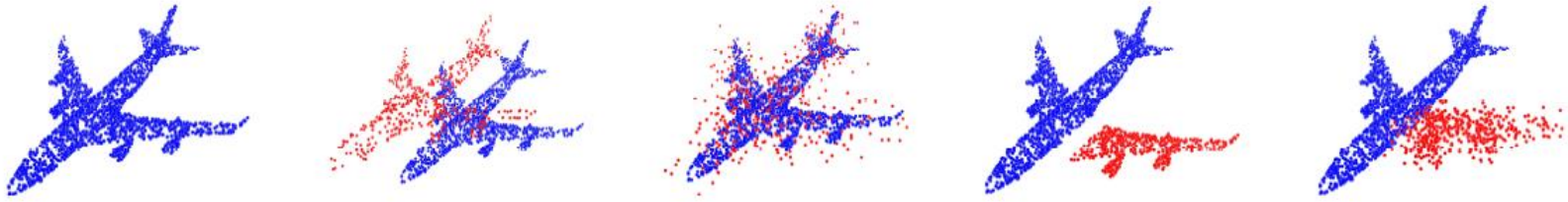
Unsupervised Learning of Transformation Equivariant Representations via Auto-Encoding Node-wise Transformations

Auto-Encoding Node-wise Transformations

- Apply transformations to individual nodes of a graph
  - Apply transformations to sampled nodes
    - Study different parts of a graph each time
    - Reduce the transformation parameters
- Global vs. local transformations
- Isotropic vs. anisotropic transformations



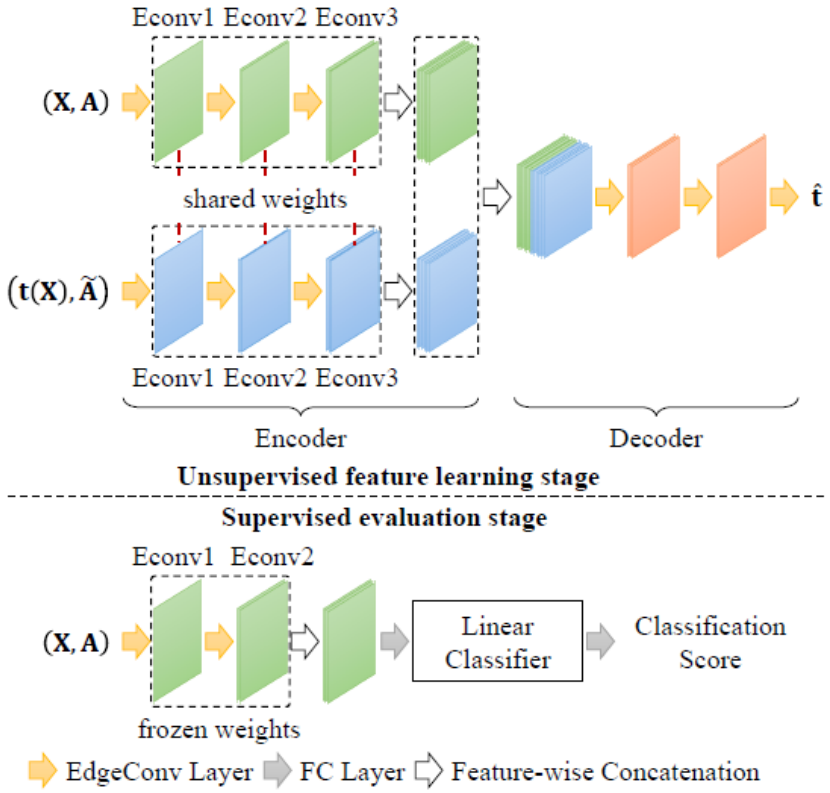
(a) Before transformation. (b) After transformation.



(a) Original model (b) Global+Isotropic (c) Global+Anisotropic (d) Local+Isotropic (e) Local+Anisotropic

Auto-Encoding Node-wise Transformations

- Decode the transformations at each node
  - Learn the representation of individual nodes on both local and global scale
- Global structures are learned by sampling nodes from different parts of graphs
- Local structures are covered by integrating information through edge convolutions on nearby nodes



Results

- Datasets: cloud data of 3D points
  - ModelNet40: graph classication
    - 12,311 meshed CAD model
    - 40 categories
  - ShapeNet part: graph segmentations
    - 16,881 3D point clouds
    - 16 categories
- Two tasks
  - Graph classification
  - Graph segmentation

Results

- Classification

Table 1. Classification accuracy (%) on ModelNet40 dataset.

Method	Year	Unsupervised	Accuracy
3D ShapeNets [47]	2015	No	84.7
VoxNet [30]	2015	No	85.9
PointNet [32]	2017	No	89.2
PointNet++ [33]	2017	No	90.7
KD-Net [21]	2017	No	90.6
PointCNN [25]	2018	No	92.2
PCNN [2]	2018	No	92.3
DGCNN [44]	2019	No	92.9
RS-CNN [28]	2019	No	93.6
T-L Network [13]	2016	Yes	74.4
VConv-DAE [39]	2016	Yes	75.5
3D-GAN [45]	2016	Yes	83.3
LGAN [1]	2018	Yes	85.7
FoldingNet [48]	2018	Yes	88.4
MAP-VAE [15]	2019	Yes	90.2
L2G-AE [27]	2019	Yes	90.6
Ours		Yes	**92.0**

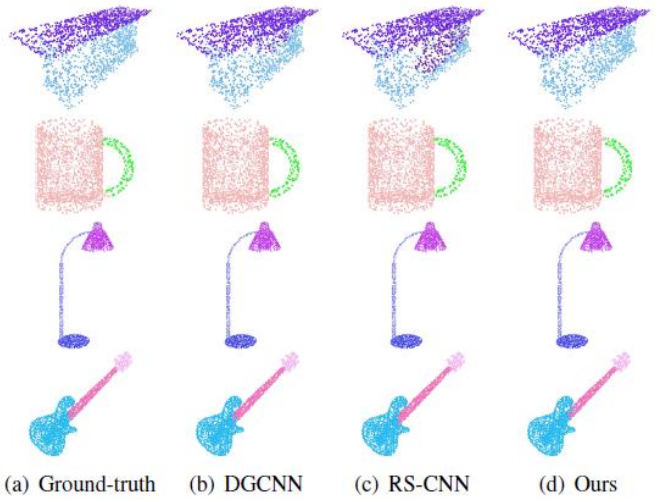
- Segmentation

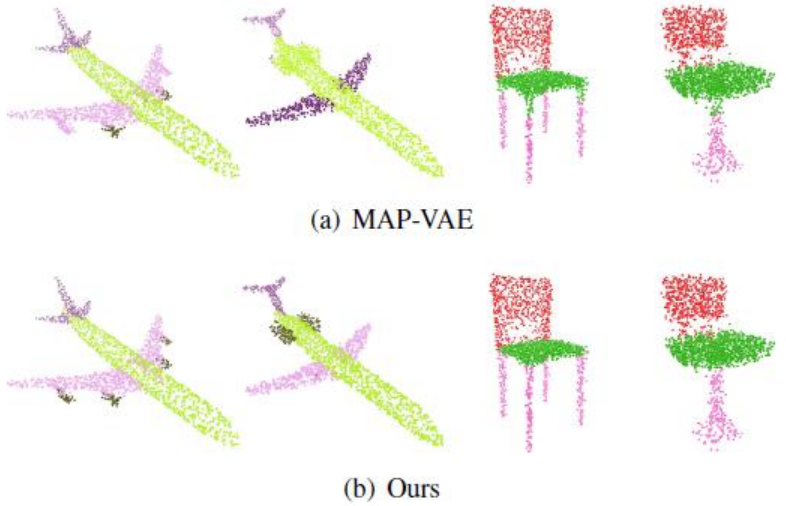
Table 4. Part segmentation results on ShapeNet part dataset. Metric is mIoU(%) on points.

	Unsup.	Mean	Aero	Bag	Cap	Car	Chair	Ear Phone	Guitar	Knife	Lamp	Laptop	Motor	Mug	Pistol	Rocket	Skate Board	Table
Samples			2690	76	55	898	3758	69	787	392	1547	451	202	184	283	66	152	5271
PointNet [32]	No	83.7	83.4	78.7	82.5	74.9	89.6	73.0	91.5	85.9	80.8	95.3	65.2	93.0	81.2	57.9	72.8	80.6
PointNet++ [33]	No	85.1	82.4	79.0	87.7	77.3	90.8	71.8	91.0	85.9	83.7	95.3	71.6	94.1	81.3	58.7	76.4	82.6
KD-Net [21]	No	82.3	80.1	74.6	74.3	70.3	88.6	73.5	90.2	87.2	81.0	94.9	57.4	86.7	78.1	51.8	69.9	80.3
PCNN [2]	No	85.1	82.4	80.1	85.5	79.5	90.8	73.2	91.3	86.0	85.0	95.7	73.2	94.8	83.3	51.0	75.0	81.8
PointCNN [25]	No	86.1	84.1	86.5	86.0	80.8	90.6	79.7	92.3	88.4	85.3	96.1	77.2	95.3	84.2	64.2	80.0	83.0
DGCNN [44]	No	85.2	84.0	83.4	86.7	77.8	90.6	74.7	91.2	87.5	82.8	95.7	66.3	94.9	81.1	63.5	74.5	82.6
RS-CNN [28]	No	86.2	83.5	84.8	88.8	79.6	91.2	81.1	91.6	88.4	86.0	96.0	73.7	94.1	83.4	60.5	77.7	83.6
LGAN [1]	Yes	57.0	54.1	48.7	62.6	43.2	68.4	58.3	74.3	68.4	53.4	82.6	18.6	75.1	54.7	37.2	46.7	66.4
MAP-VAE [15]	Yes	68.0	62.7	67.1	73.0	58.5	77.1	67.3	84.8	77.1	60.9	90.8	35.8	87.7	64.2	45.0	60.4	74.8
Ours	Yes	**81.9**	**81.7**	**68.1**	**83.7**	**74.6**	**88.1**	**68.9**	**90.6**	**86.6**	**80.0**	**95.6**	**56.3**	**90.0**	**80.8**	**55.2**	**70.7**	**79.1**

Results

- Compared with both supervised and unsupervised graph segmentations, our approach is much more competitive.



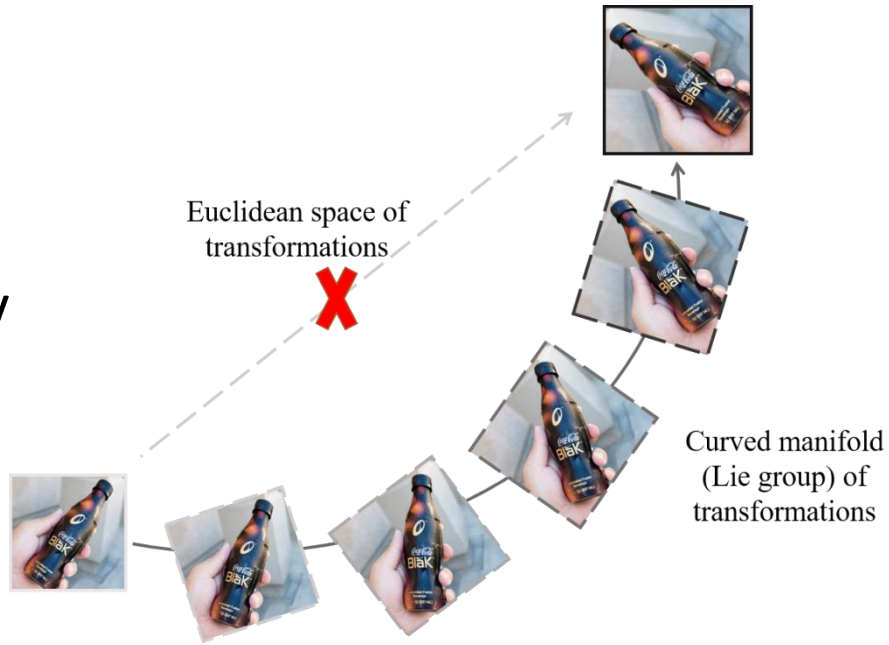


Recent Progresses

AETv2: AutoEncoding Transformations by Minimizing Geodesic Distances in Lie Groups of Transformations

Mean-Squared Estimation vs. Geodesic Distance Minimization

- (Matrix representation of) Transformations stay in a curved manifold of Lie group rather than in a flat Euclidean space
- MSE minimizes the Euclidean space between predicted and groudtruth transformations
  - Not represent how a transformation continuously evolves to another one in the Lie group
- Geodesic distance minimization (GDM) represent a “real” distance between transformations

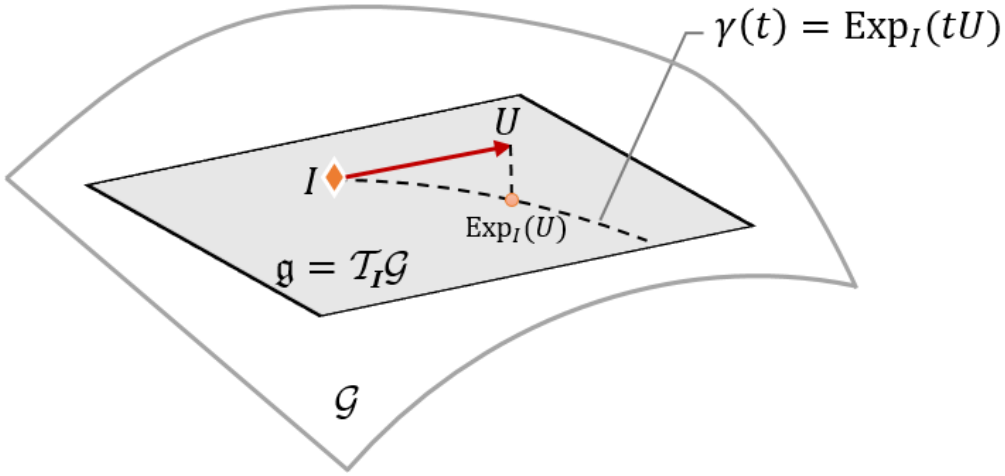


Geodesic distance between Transformations

- Given two (matrix form of) transformations $\mathbf{T}$ and $\widehat{\mathbf{T}}$, the geodesic distance between them is

$$\ell\left(\widehat{\mathbf{T}}, \mathbf{T}\right) = \frac{1}{2}\left\|\mathrm{Log}_{\mathrm{I}}\left(\mathbf{T}^{-1}\widehat{\mathbf{T}}\right)\right\|_{\mathrm{F}}$$

where $\mathrm{Log}_{\mathrm{I}}$ is the Riemannian logarithm not the matrix logarithm.



- Riemannian exp at I maps a point in tangent space to Lie group
- Riemannian logarithm at I maps back to the tangent space.

Homography Transformations

- For many spatial transformations, there is no closed-form Riemannian logarithm.
  - Homography transformations achieves the SOTA accuracy in AETv1.
- Solution
  - Use a subgroup of transformation with tractable Riemannian logarithm (which reduces to matrix logarithm)
  - Project the transformations to the subgroup
  - Minimize the resultant geodesic distance in the subgroup (together with the projection loss)

Projection into SO(3) subgroup

- Riemannian logarithm reduces to matrix logarithm in SO(3)
- Project $\mathbf{T}^{-1}\hat{\mathbf{T}}$ to SO(3)

$$\mathbf{P} = \Pi_{\mathrm{SO}(3)}(\mathbf{T}^{-1}\hat{\mathbf{T}})$$

- By Rodrigues' rotation formula,

$$\theta = \arccos[\frac{\mathrm{tr}\,(\mathbf{P}) - 1}{2}]$$

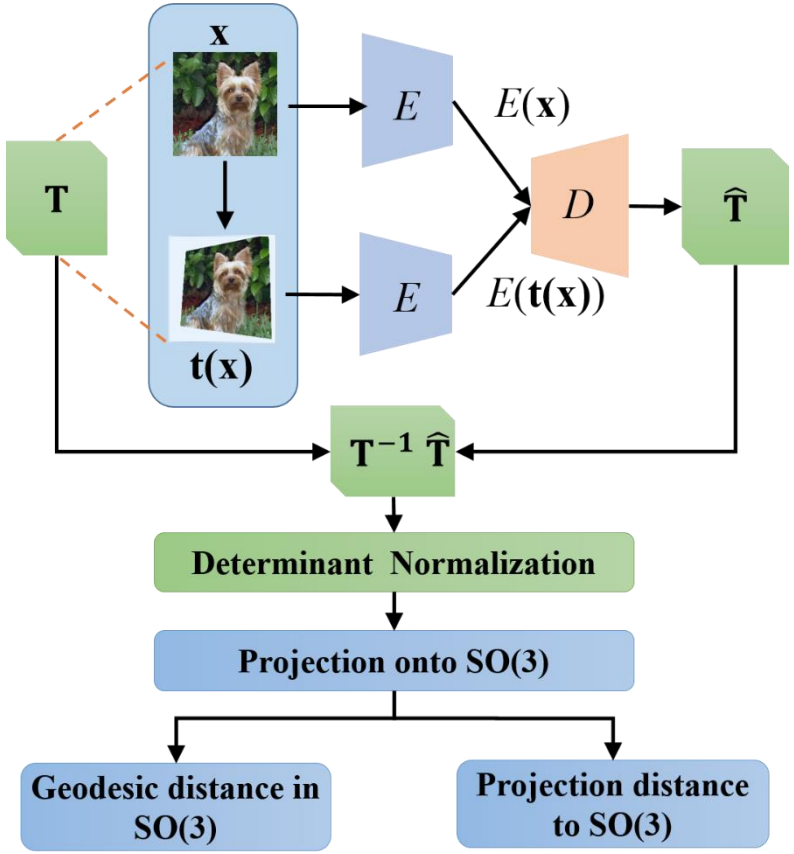
is the rotation angle around a unit 3D axis in the direction of $\log \mathbf{P}$

Objective in SO(3) subgroup

- Given two (matrix form of) transformations $\mathbf{T}$ and $\hat{\mathbf{T}}$,

$$\hat{\ell}(\hat{\mathbf{T}}, \mathbf{T}) = \theta + \lambda \|\mathbf{R}_{\Pi}\|_F^2$$

Where $\mathbf{R}_{\Pi} = \mathbf{T}^{-1}\hat{\mathbf{T}} - \Pi_{SO(3)}(\mathbf{T}^{-1}\hat{\mathbf{T}})$ is the projection residual



ImageNet Results

- Better than AETv1 (MSE)

Method	Conv4	Conv5
ImageNet Labels [2](Upper Bound)	59.7	59.7
Random [13] (Lower Bound)	27.1	12.0
Tracking [20]	38.8	29.8
Context [5]	45.6	30.4
Colorization [25]	40.7	35.2
Jigsaw Puzzles [12]	45.3	34.6
BiGAN [6]	41.9	32.2
NAT [2]	-	36.0
DeepCluster [4]	-	44.0
RotNet [9]	50.0	43.8
AETv1 [23]	53.2	47.0
(Ours) AETv2	**54.3**	**47.5**

Method	Conv1	Conv2	Conv3	Conv4	Conv5
ImageNet Labels (Upper Bound) [9]	19.3	36.3	44.2	48.3	50.5
Random (Lower Bound)[9]	11.6	17.1	16.9	16.3	14.1
Random rescaled [11](Lower Bound)	17.5	23.0	24.5	23.2	20.6
Context [5]	16.2	23.3	30.2	31.7	29.6
Context Encoders [16]	14.1	20.7	21.0	19.8	15.5
Colorization[25]	12.5	24.5	30.4	31.5	30.3
Jigsaw Puzzles [12]	18.2	28.8	34.0	33.9	27.1
BiGAN [6]	17.7	24.5	31.0	29.9	28.0
Split-Brain [24]	17.7	29.3	35.4	35.2	32.8
Counting [13]	18.0	30.6	34.3	32.5	25.7
RotNet [9]	18.8	31.7	38.7	38.2	36.5
AETv1 [23]	19.2	32.8	40.6	39.7	37.7
(Ours) AETv2	**19.6**	**34.1**	**41.9**	**40.4**	**37.9**
DeepCluster* [4]	13.4	32.3	41.0	39.6	38.2
AETv1* [23]	19.3	35.4	44.0	43.6	42.4
(Ours) AETv2*	**21.2**	**36.9**	**45.9**	**44.7**	**43.2**