Learning Methods

Transfer Learning, Meta Learning, Domain Adaptation

Hao Dong

2019, Peking University

Learning Methods

- Motivation
- Transfer Learning
- Meta Learning
- Domain Adaptation
- Summary

Motivation

Motivation

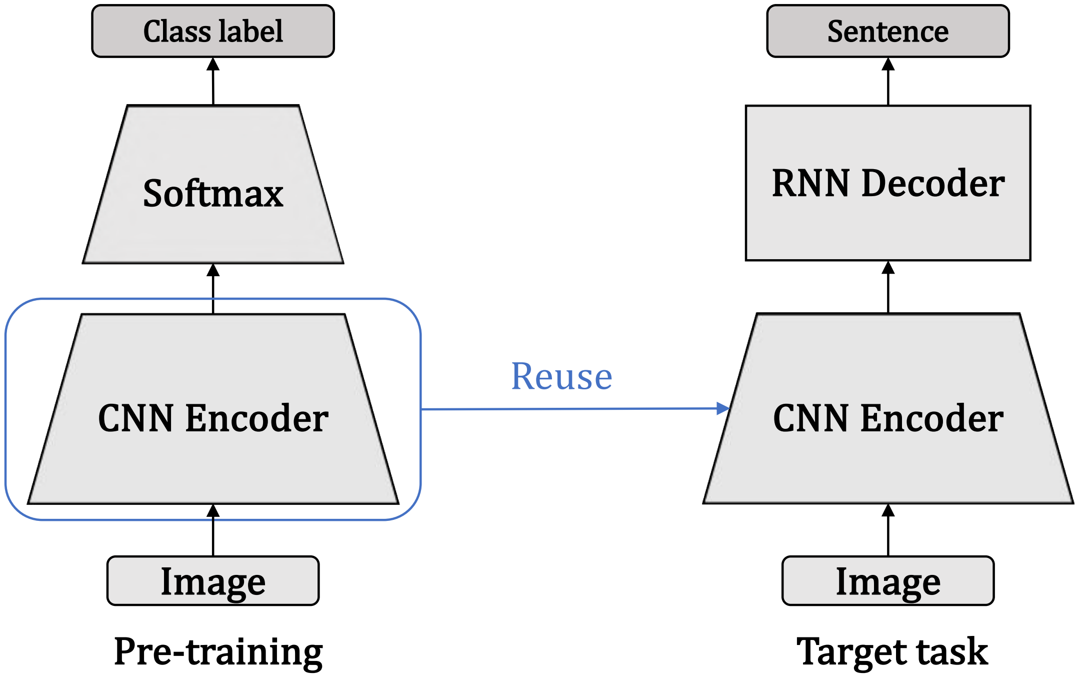
Transfer Learning

Transfer Learning

- Transfer learning, also known as inductive transfer, utilises the knowledge of one task and applies it to **related tasks**. In deep learning, pre-trained models are widely used for improving the performance of many tasks without labelling more data. The idea is to train a network on a large dataset with a general task and reuse the network for another network with the target task. **This transfer tends to work if the datasets include common features.**

Transfer Learning

- Reuse Model

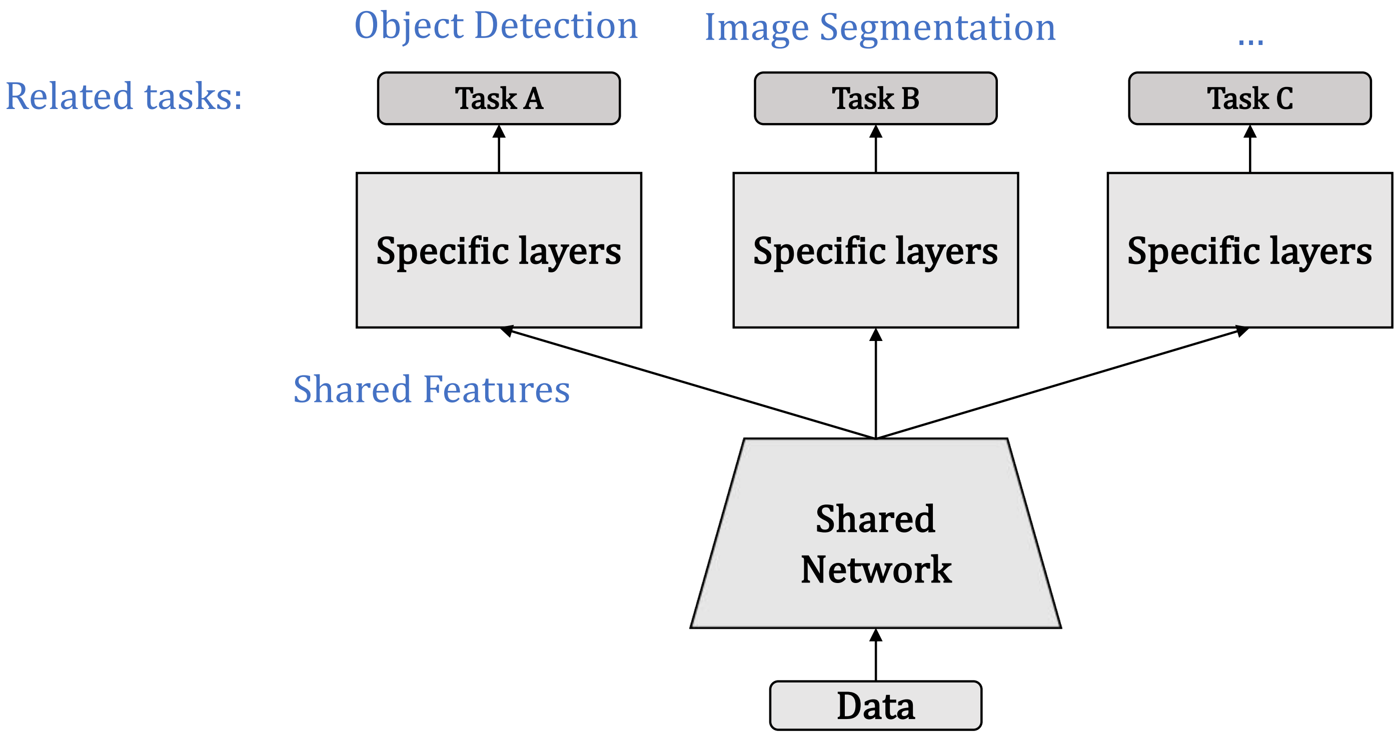


- OpenPose
- Show and Tell
- Word2Vec
- GloVe
- …

Related tasks: Image Classification Image Captioning

Transfer Learning

- Multi-task Learning



- Mask RCNN
- ...

Meta Learning

Meta Learning

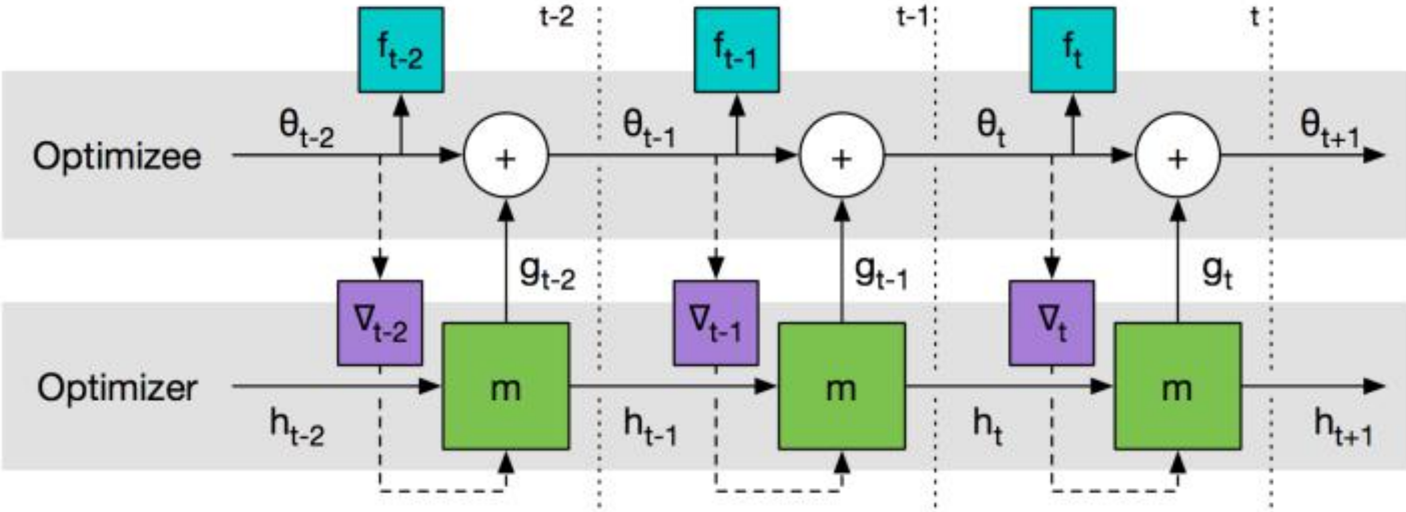
- Meta learning focuses on learning the learning process, i.e., learn to learn

Meta Learning

PEKING UNIVERSITY

- Predicting gradients

Train an extra network to predict the gradients in supervised way.

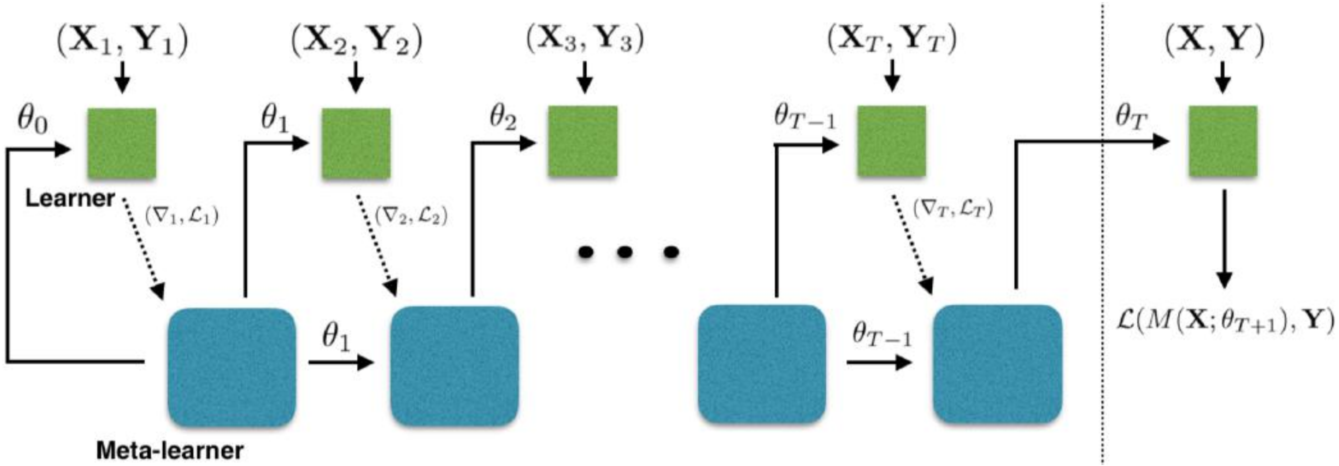


Learning to learn by gradient descent by gradient descent. *Andrychowicz, Marcin, Denil, Misha.* NIPS, 2016.

Meta Learning

- Predicting parameters

Train a LSTM network to predict the parameter of the next update

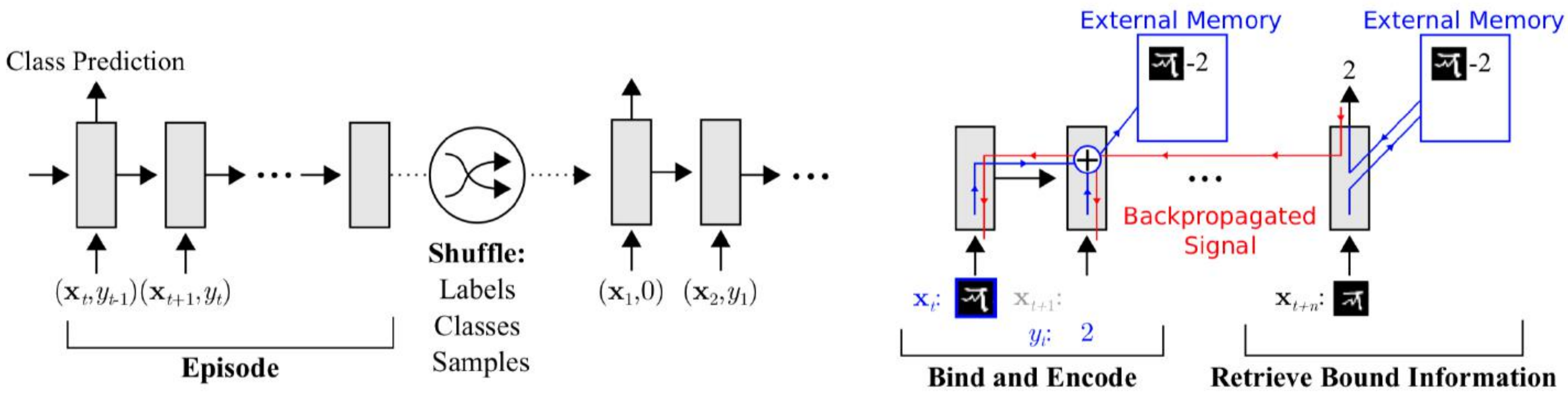


Optimization as a model for few-shot learning. *Ravi, Sachin and Larochelle, Hugo*. ICLR, 2017.

Meta Learning

- Memory based methods

Feed the previous output to the model and store the information in an external memory.



(a) Task setup (b) Network strategy

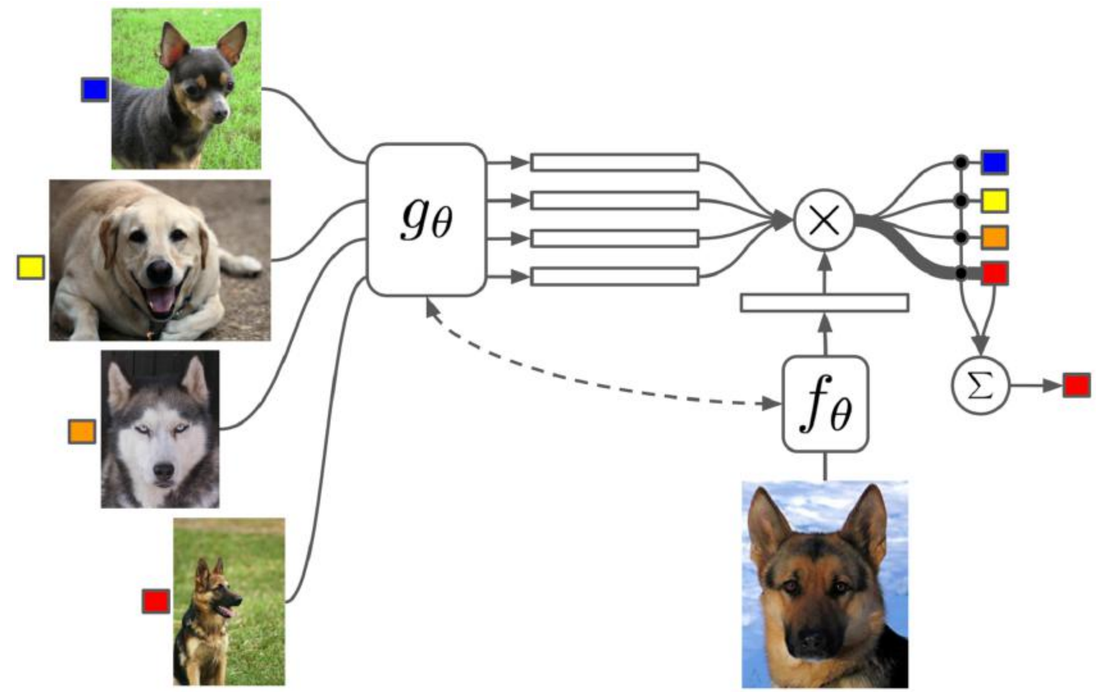
Meta-Learning with Memory-Augmented Neural Networks. *Adam Santoro, Sergey Bartunov, etal*. ICML, 2016.

Meta Networks. *Munkhdalai T, Yu H*. arXiv:1703.00837, 2017.

Meta Learning

- Memory based methods

Feed the previous output to the model and store the information in an external memory.



The output label is obtained by "attention"

$$\bar{y} = \sum_{i=1}^{k} a(\bar{x}, x_i) y_i$$

Matching Networks for One Shot Learning. *Oriol Vinyals, Charles Blundell, etal*. NIPS, 2016

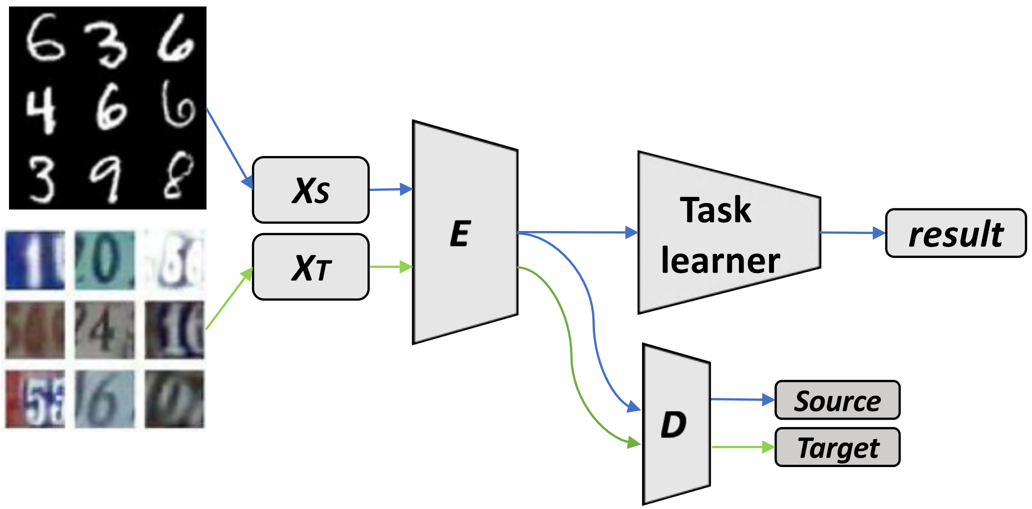
Domain Adaptation

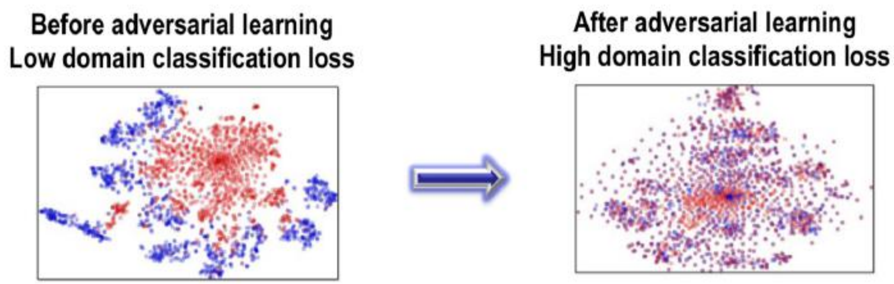
Domain Adaptation

- Domain adaptation learns from a source data distribution and perform well on a different but related target data distribution.
- vs. Transfer Learning: Domain adaptation does not need labels for the target domain.

Domain Adaptation

- Single Source



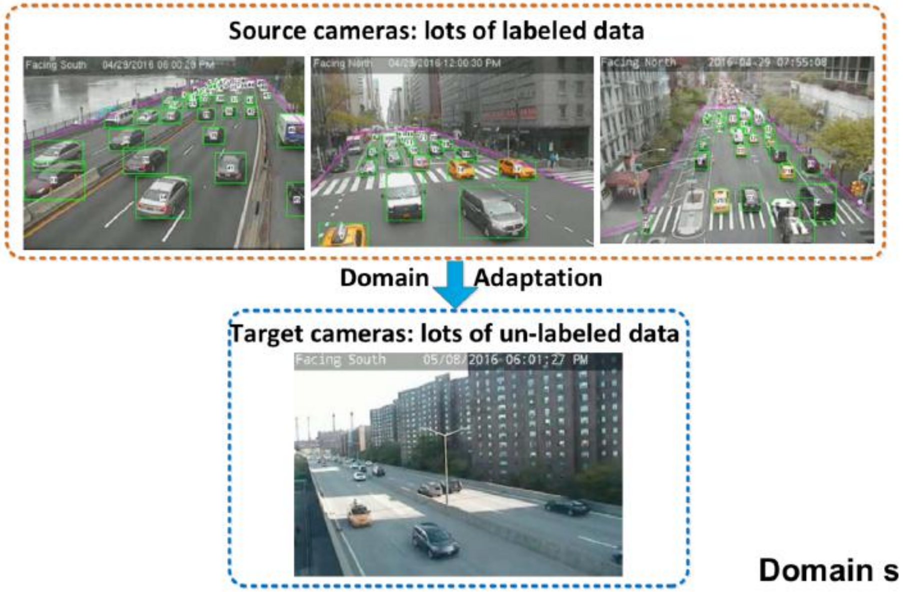


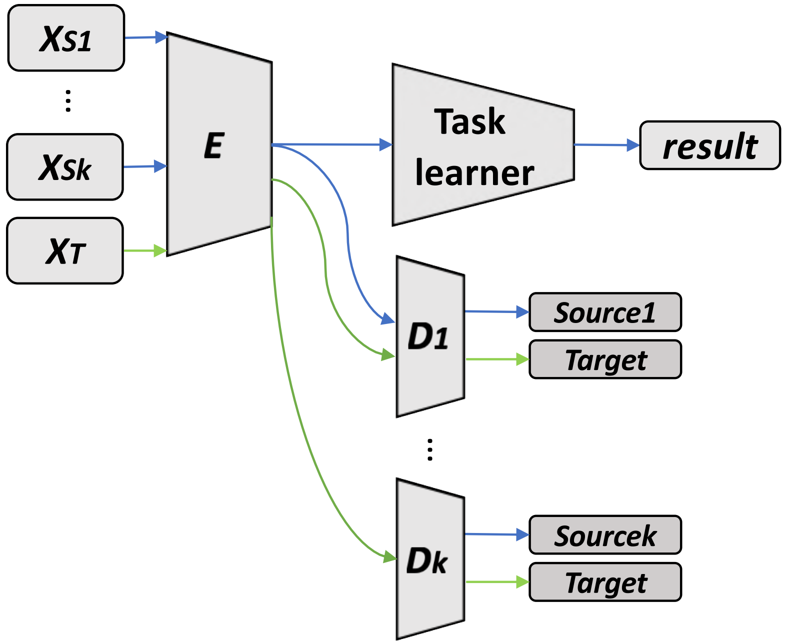
Source images have label
Target images do not have label

Domain-Adversarial Training of Neural Networks. *Y. Ganin, H. Ajakan et al. JMLR. 2016*

Domain Adaptation

- Multi-source





Source images have label
Target images do not have label

Multiple Source Domain Adaptation with Adversarial Learning. *S. Zhang, H. Zhao et al. NIPS. 2018.*

Domain Adaptation

- Example: Learn Edge, Depth, and Surface Normal

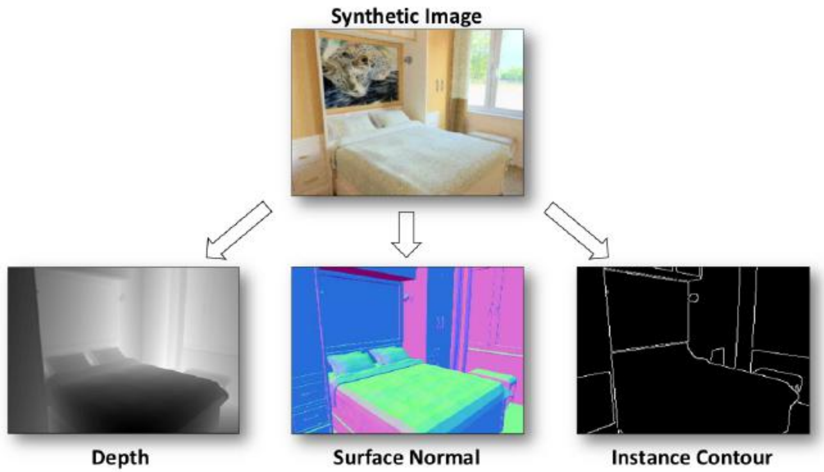


Figure 1. **Main idea.** A graphics engine can be used to easily render realistic synthetic images together with their various physical property maps. Using these images, we train a self-supervised visual representation learning algorithm in a multi-task setting that also adapts its features to real-world images.

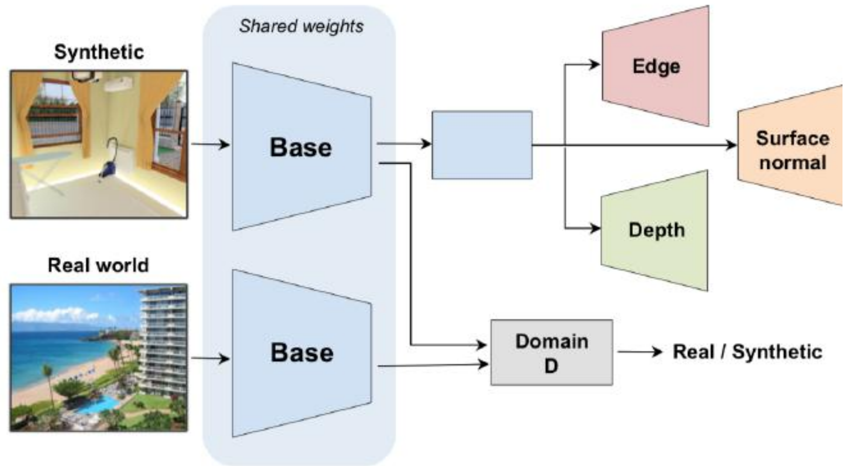


Figure 2. **Network architecture.** The upper net takes a synthetic image and predicts its depth, surface normal, and instance contour map. The bottom net extracts features from a real-world image. The domain discriminator D tries to differentiate real and synthetic features. The learned blue modules are used for transfer learning on real-world tasks.

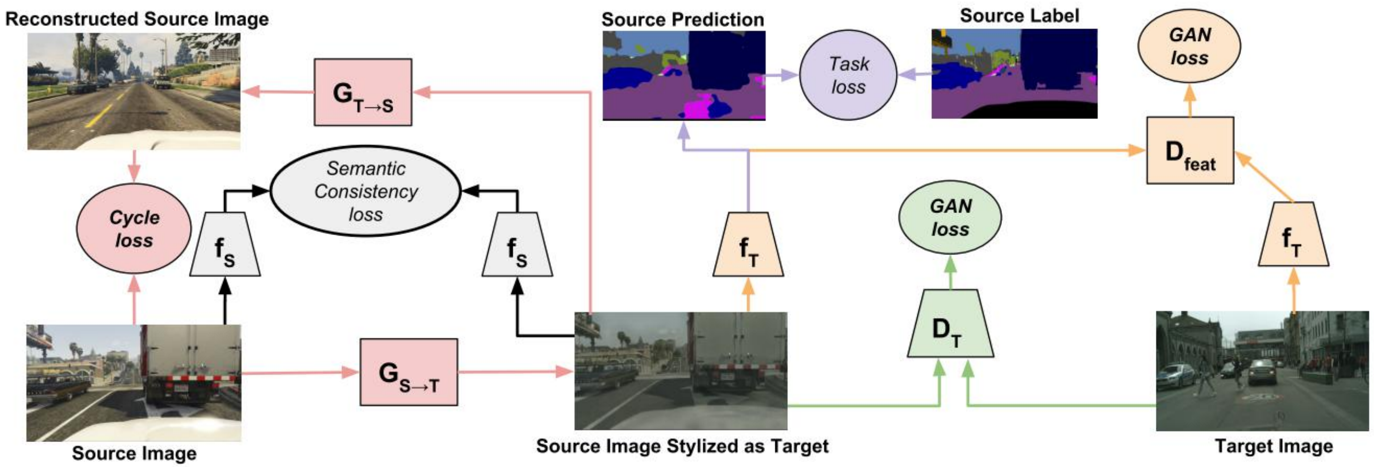
Cross-Domain Self-supervised Multi-task Feature Learning using Synthetic Imagery. *Zhongzheng Ren and Yong Jae Lee*. CVPR, 2018

Domain Adaptation

- Example: Learn Segmentation and Image-to-Image Translation

$$
\begin{tikzcd}
x_s \arrow[r, "\text{known}"] \arrow[d, "\text{CycleGAN}"'] & y_s \\
x_t \arrow[ur, "\text{supervised}"'] &
\end{tikzcd}
$$

We don't know y_t



CyCADA: Cycle-Consistent Adversarial Domain Adaptation. *Hoffman, J., Tzeng, E., Park, T., Zhu, J.,, Isola, P. ICML*, 2018.

Summary

Summary

- Motivation
  - Time-series data
- Transfer Learning
  - one-hot vector, BOW, word embedding, Word2Vec, CBOW, Skip-Gram, negative sampling, NCE
- Meta Learning
  - one-to-many, many-to-one, asynchronous many-to-many, synchronous many-to-many
- Domain Adaptation
  - Hidden vector (state), long-term dependency problem

Summary

- References
  - IJCAI13: Transfer Learning with Applications
    http://ijcai13.org/files/tutorial_slides/td2.pdf

Summary

- Exercise 1: (Optional)
  - Reimplement “Domain-Adversarial Training of Neural Networks”

Link: https://github.com/zsdonghao/deep-learning-note/

北京大學
PEKING UNIVERSITY

Questions?