Adversarial based Unsupervised Domain Adaptation

Chih-Hui (John) Ho

2018/08/22

Outline

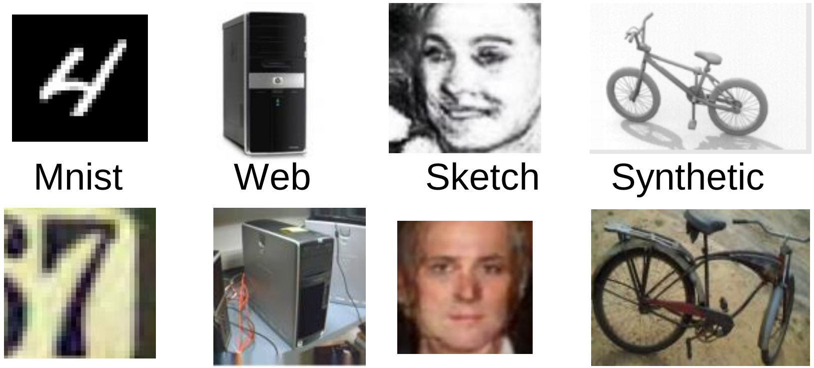
- Introduction
- Overview
- Adversarial Based domain adapation
  - Collaborative and adversarial network for unsupervised domain adaptation
  - Maximum classifier discrepeancy for unsupervised domain adaptation
  - Detach and adapt: Learning cross domain disentangled deep representation
  - Learning from synthetic data: Addressing domain shift for semantic segmentation
- Reference

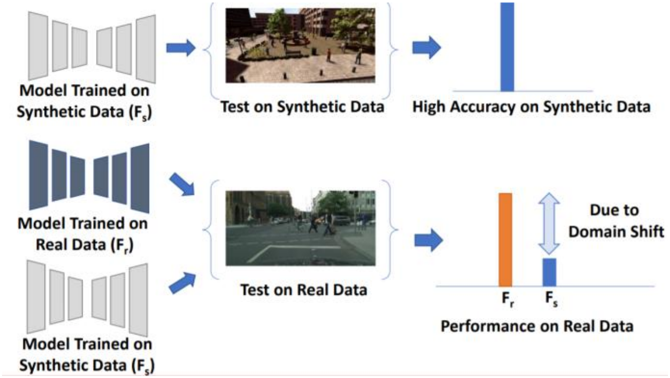
Introduction

- Deep neural network has achieved good performance by utilizing labeled data
- Labels for data we are interested in (target domain) might not be available
- Can we use labeled data from different domains (source domain)?
    - Data distribution between source and target data is different (domain shift)
    - Illumination, pose, image quality can cause domain shift
    - Performace degrades

Source data (Xs, Ys)

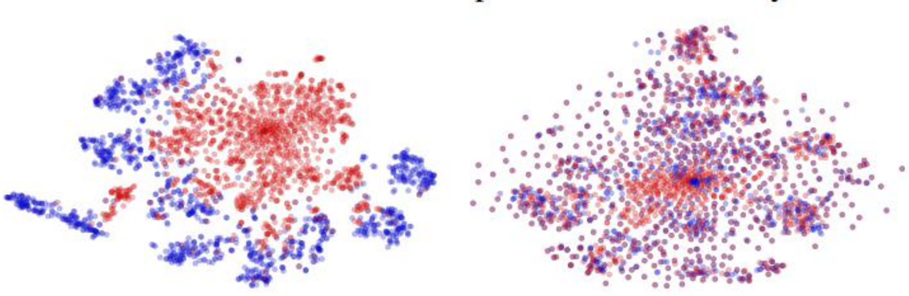
Target data Xt





Introduction

- Unsupervised domain adaptation aims to bridge the gap between source and target data/domain by matching their distribution
- Goal
    - Learn a cross domain representation of data
    - Transfer knowledge from source to target domain
    - Learn a classifier that generalizes well on target domain
- We will focus on adversarial based unsupervised domain adaptation

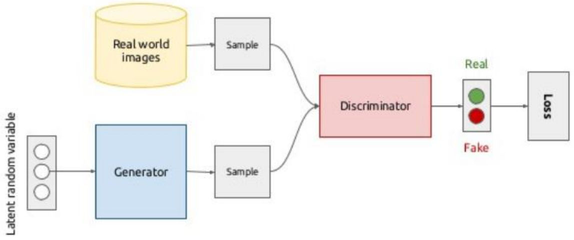


(a) Non-adapted (b) Adapted

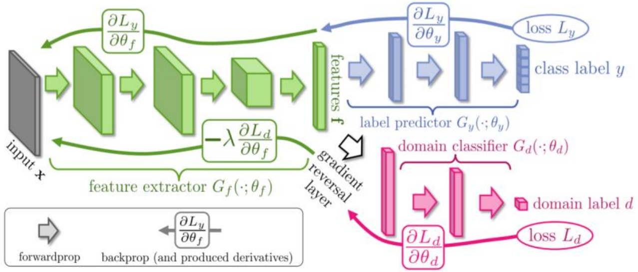
Blue points: source domain features
Red points: target domain features

Overview

- Generative adversarial network(GAN) has great success on estimating generative models with adversarial training
- The adversarial process of GAN can be applied to domain adaptation task
  - A feature extractor extracts the data distribution
  - A domain classifier D distinguishes whether the data distribution comes from source or target domain
  - Learn a domain invariant feature eventually
- GAN vs Gradient reversal layer(GRL) based architecture



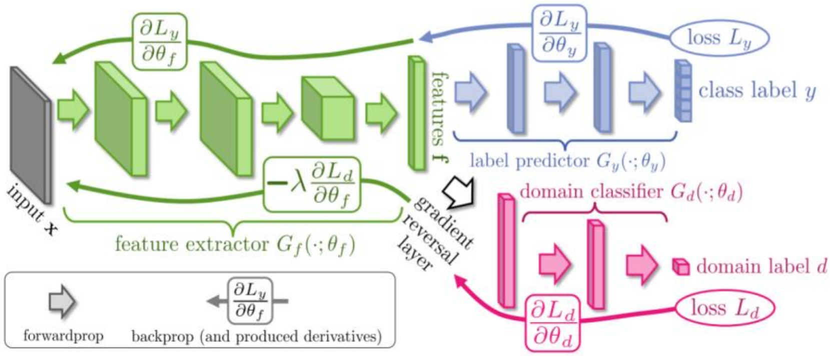
GAN based



GRL based

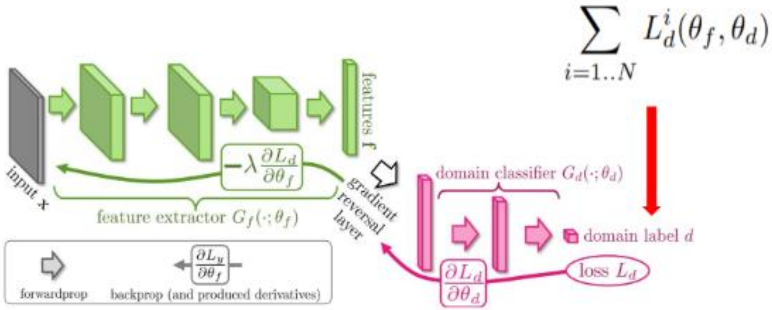
Overview

- Label classifier
  - Given a feature, label classifier classifies its label
  - Minimize label classification loss (Cross entropy loss)
- Domain classifier
  - Assume inputs from source domain and target domain have labels 1 and 0 respectively
  - Given a feature, domain classifier classifies which domain the feature comes from
  - Minimize domain classification loss (Binary cross entropy loss)
- Feature extractor
  - Extract the feature given the input
  - Minimize the label classification loss
  - Maximize domain classification loss



[Unsupervised Domain Adaptation by Backpropagation]

Overview

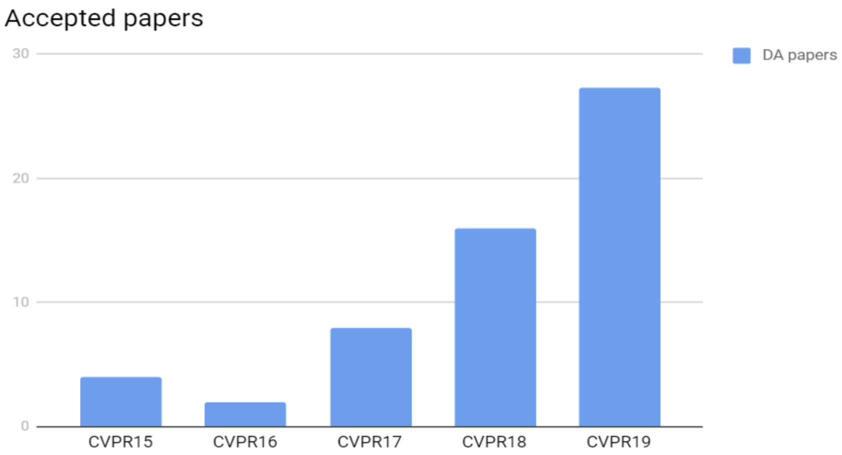


- Gradient reversal layer
  - Forword: GRL is an identity transformation
  - Backward: GRL takes gradient from subsequent layer and multiply by constant -w and pass to previous layer

	Switching between submodules	Drawback
GAN	yes	Need to decide how many iteration of training G and D
GRL	No	Gradient vanishing, need to decide w

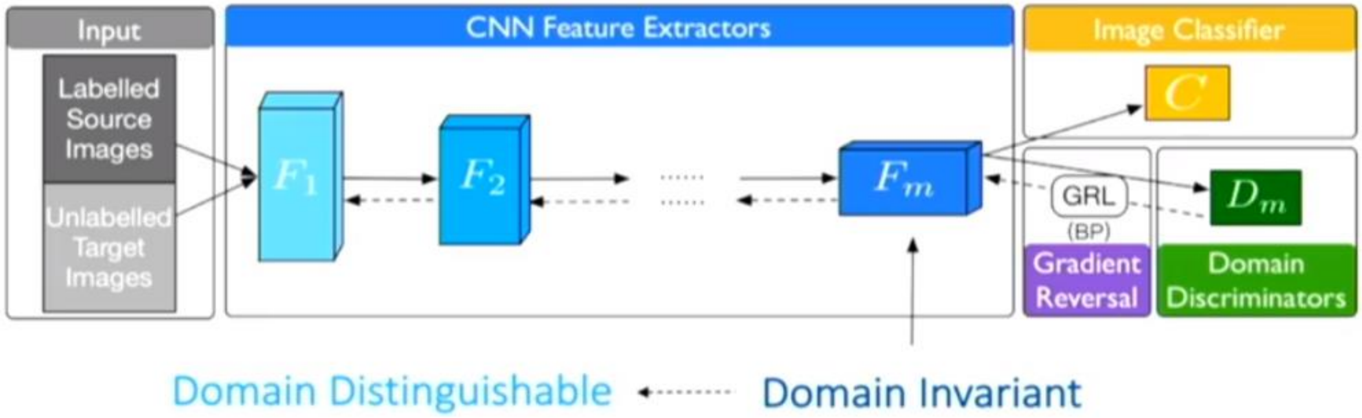
Adversarial Based domain adapation

- GRL
  - Collaborative and adversarial network for unsupervised domain adaptation (Spotlight)
- GAN
  - Maximum classifier discrepeancy for unsupervised domain adaptation (Oral)
  - Detach and adapt: Learning cross domain disentangled deep representation (Spotlight)
  - Learning from synthetic data: Addressing domain shift for semantic segmentation (Spotlight)



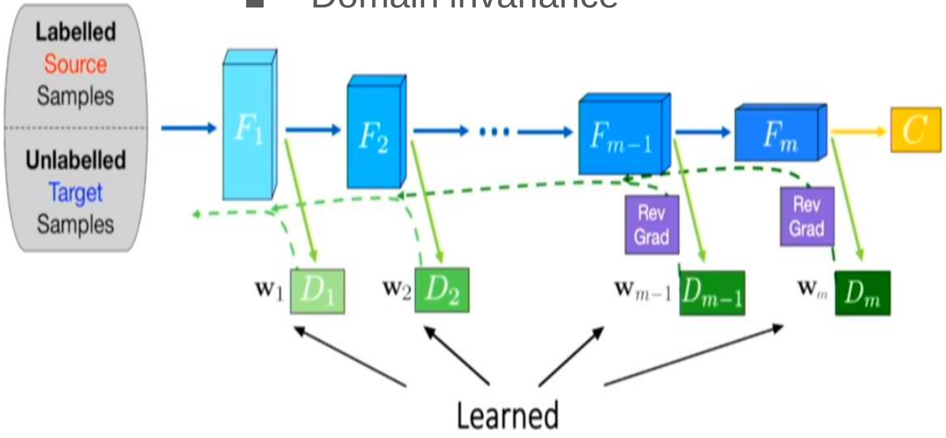
Collaborative and adversarial network for unsupervised domain adaptation

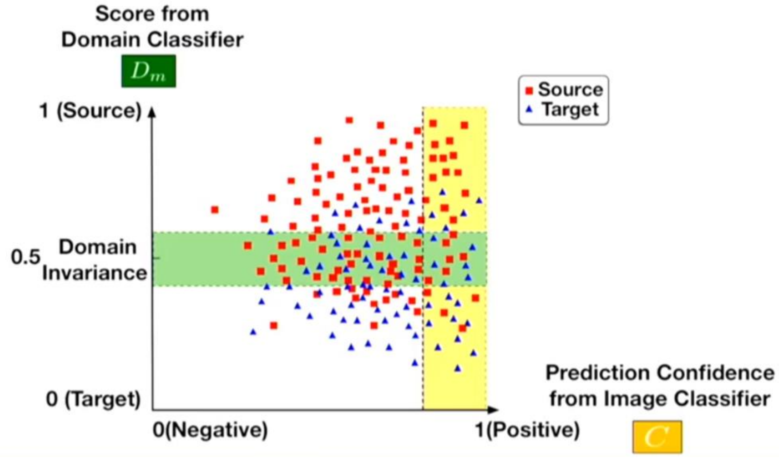
- DNN tries to learn domain invariant feature at the final layer
  - All the features are learned to be domain invariant
- Motivation
  - Low level details (corner and edges) are useful to represent features in different domains
  - Learn domain **variant** feature in low level layers (informative / collaborative learning)
  - Learn domain **invariant** feature in high level layers (uninformative / adversarial learning)



Collaborative and adversarial network for unsupervised domain adaptation

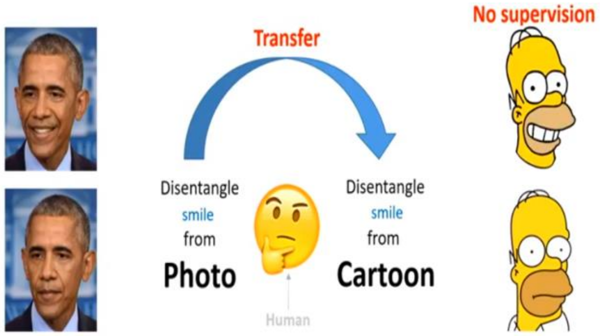
- Proposed methods and novelty
  - The optimal combination of multiple domain classifiers (each D has different w)
    - Low level features should be domain variant: w > 0
    - High level features should be domain invariant: w < 0
  - Iteratively incorporate target data to training set with psuedo label
    - High class prediction confidence
    - Domain invariance

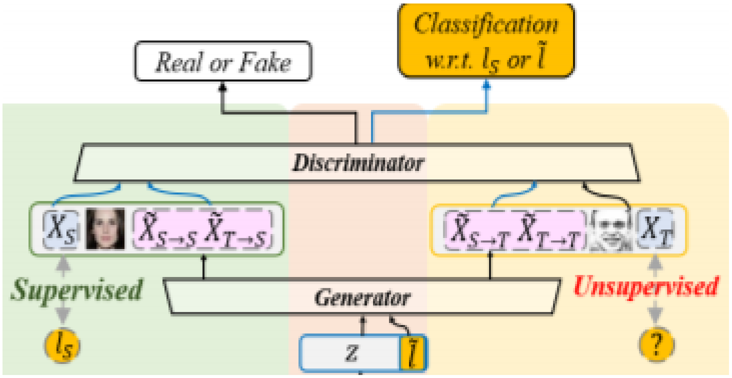


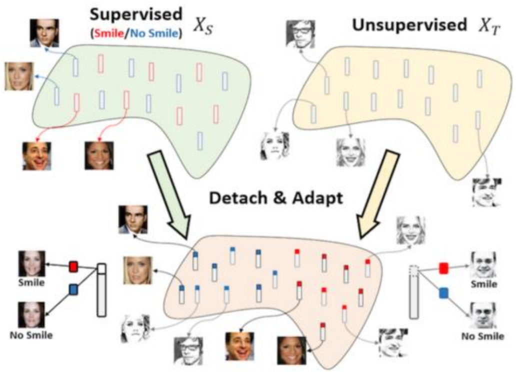


Detach and adapt: Learning cross domain disentangled deep representation

- Motivation
  - Can machine imagine same image with different attribute?
    - Representation disentanglement
  - Can machine transfer attribute across domains?
    - Cross domain representation
- Proposed method and novelty
  - Representation disentanglement for cross domain data
  - Share weight in higher layer in G and D to bridge the gap in high level representation







Detach and adapt: Learning cross domain disentangled deep representation

- Proposed method and novelty
  - Auxiliary classifier in Dc to maximize the mutual information between assigned label and generated images

$$\tilde{X}_S \sim G_S(G_C(z,\tilde{l})), \tilde{X}_T \sim G_T(G_C(z,\tilde{l}))$$

$$\mathcal{L}^S_{adv} = \mathbb{E}[\log(D_C(D_S(X_S)))] + \mathbb{E}[\log(1 - D_C(D_S(\tilde{X}_S)))],$$

$$\mathcal{L}^T_{adv} = \mathbb{E}[\log(D_C(D_T(X_T)))] + \mathbb{E}[\log(1 - D_C(D_T(\tilde{X}_T)))],$$

$$\mathcal{L}_{adv} = \mathcal{L}^S_{adv} + \mathcal{L}^T_{adv}.$$

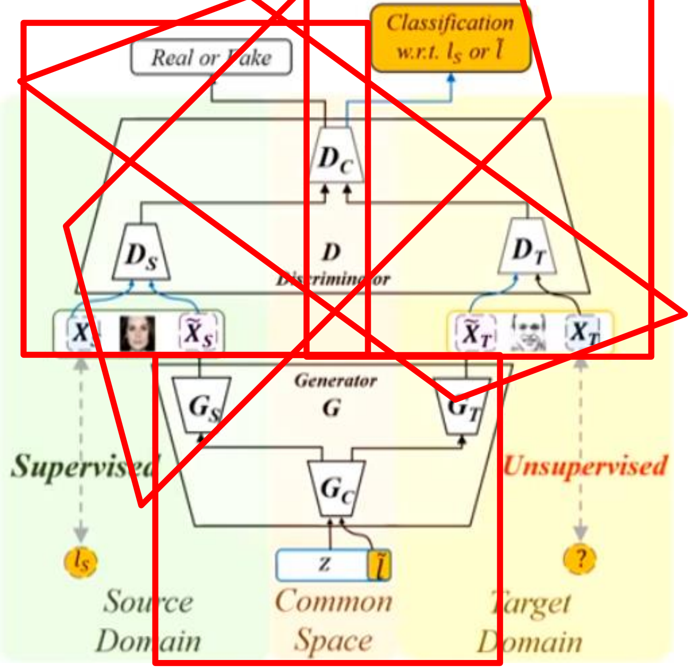
$$\mathcal{L}^S_{dis} = \mathbb{E}[\log P(l = \tilde{l}|\tilde{X}_S)] + \mathbb{E}[\log P(l = l_S|X_S)],$$

$$\mathcal{L}^T_{dis} = \mathbb{E}[\log P(l = \tilde{l}|\tilde{X}_T)],$$

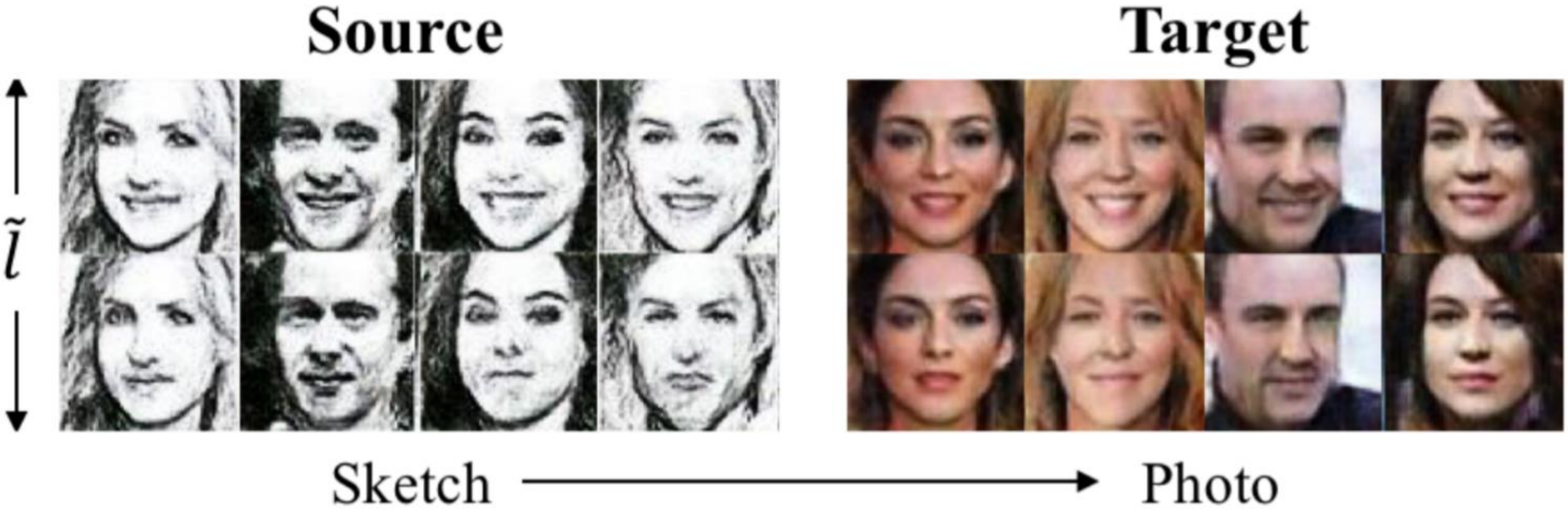
$$\mathcal{L}_{dis} = \mathcal{L}^S_{dis} + \mathcal{L}^T_{dis}.$$

$$\theta_G \xleftarrow{+} -\Delta_{\theta_G}(-\mathcal{L}_{adv} + \lambda\mathcal{L}_{dis})$$

$$\theta_D \xleftarrow{+} -\Delta_{\theta_D}(\mathcal{L}_{adv} + \lambda\mathcal{L}_{dis})$$

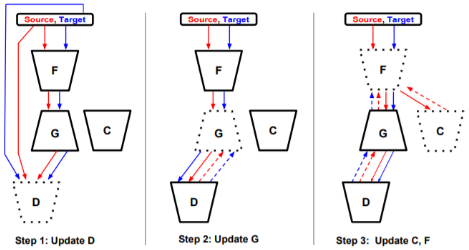


Detach and adapt: Learning cross domain disentangled deep representation

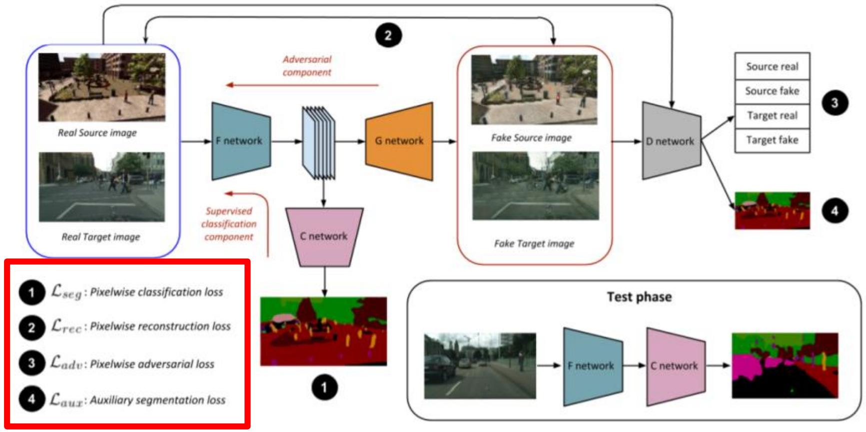


Learning from Synthetic Data: Addressing Domain Shift for Semantic Segmentation

- Motivation
  - Previous DA approaches mostly focus on classification task and does not tailor for segmentation task
  - Labels are even harder to obtain in image segmentation task
- Proposed method and novelty
  - Adapt the representation learned by segmentation network across synthetic and real domains
  - Adversarial loss were calculated on image space (better performance) instead of feature space
  - F is trained to extract domain invariant feature



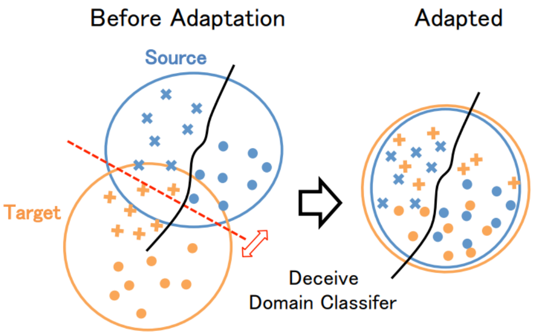
Type	Variants	Description
Within-domain	$\mathcal{L}^s_{adv,D}$	Classify real source input as *src-real*; fake source input as *src-fake*
	$\mathcal{L}^s_{adv,G}$	Classify fake source input as *src-real*
	$\mathcal{L}^t_{adv,D}$	Classify real target input as *tgt-real*; fake target input as *tgt-fake*
	$\mathcal{L}^t_{adv,G}$	Classify fake target input as *tgt-real*
Cross-domain	$\mathcal{L}^s_{adv,F}$	Classify fake source input as real target (*tgt-real*)
	$\mathcal{L}^t_{adv,F}$	Classify fake target input as real source (*src-real*)



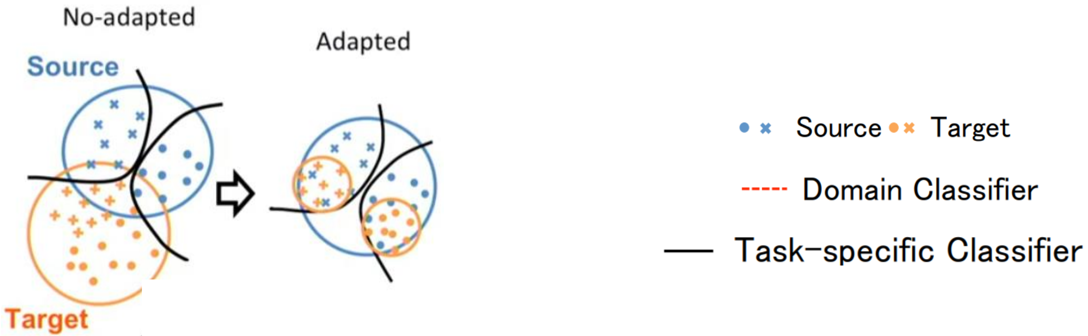
Maximum classifier discrepeancy for unsupervised domain adaptation

- Previous approaches use domain classifier to force generator to generate domain invariant features
- Motivation
  - Generated target feature can be closed to the classifier’s boundary
  - Relationship between decision boundary and target data should be considered
  - Align source and target distribution using task-specific decision boundaries

Previous method

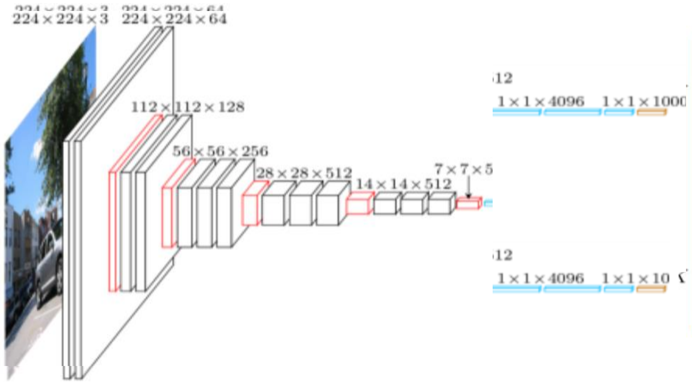


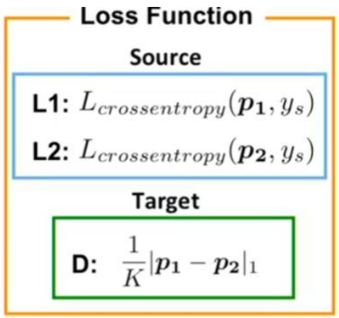
Proposed method



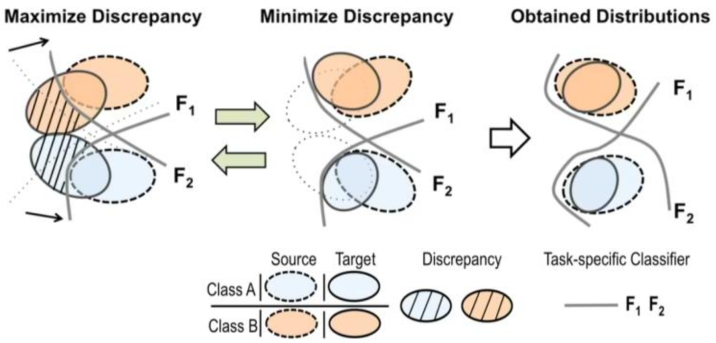
Maximum classifier discrepeancy for unsupervised domain adaptation

- Proposed method and novelty
  - Training procedure
    - Train G, F1, F2 until they can classify source data correctly
    - Train F1, F2 to maximize discrepancy on target data (Fix G)
    - Train G to minimize discrepancy (Fix F1, F2)
    - Repeat step 2 and 3
  - Use 2 classifiers from one network as discriminator and force G to avoid generating features close to decision boundaries (novel training method)





discrepancy



Reference

- Unsupervised domain adaptation via backpropagation
- Collaborative and adversarial network for unsupervised domain adaptation
- Maximum classifier discrepeancy for unsupervised domain adaptation
- Detach and adapt: Learning cross domain disentangled deep representation
- Learning from synthetic data: Addressing domain shift for semantic segmentation
- Deep Visual Domain Adaptation: A Survey
- https://github.com/zhaoxin94/awsome-domain-adaptation

- Some slide images are cropped from the CVPR presentations