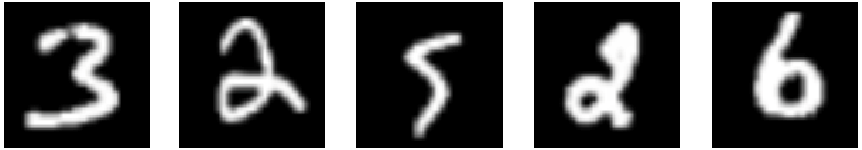
Adversarial Techniques for Visual Domain Adaptation

Kate Saenko

BOSTON UNIVERSITY

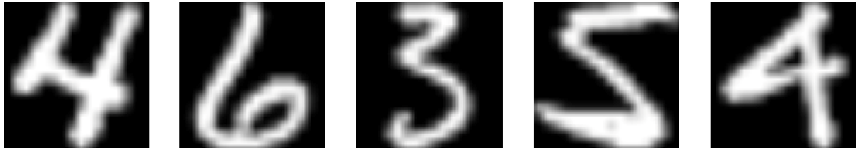
Has deep learning solved AI?

Train on MNIST:



Test on MNIST: 99% accuracy

USPS:

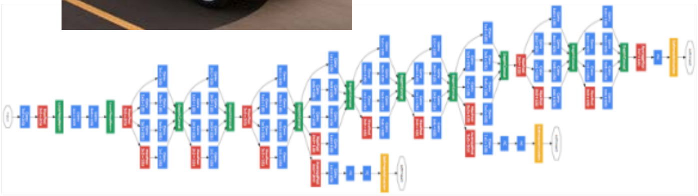


Test on USPS: 68% accuracy

"What you saw is not what you get"

What your net is trained on

What it's asked to label



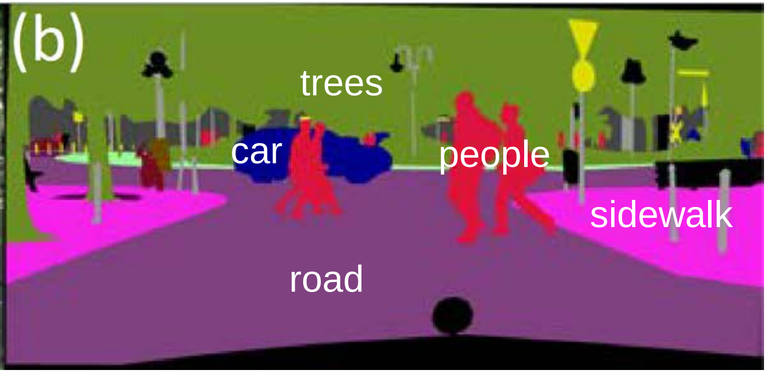
"Dataset Bias"

"Domain Shift"

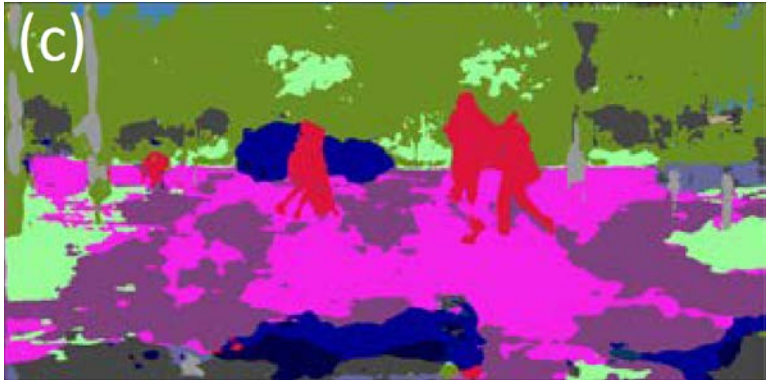
Problem: Domain Shift

Input Image

True Segmentation



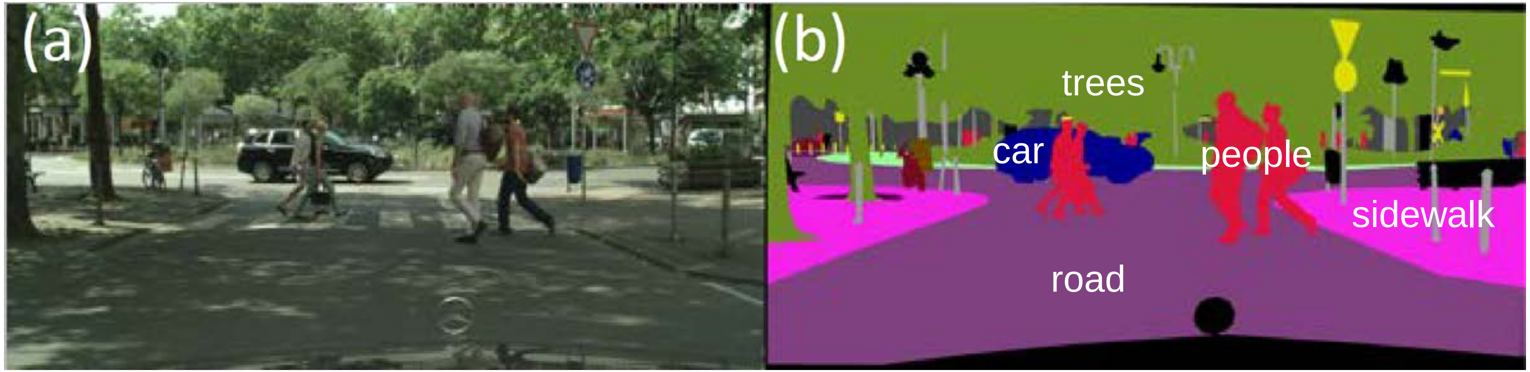
Output of model



Problem: Domain Shift

Input Image

True Segmentation



After adaptation

The Good News:

We can recover performance with no additional training and no data augmentation!

- WHAT IS DOMAIN ADAPTATION?

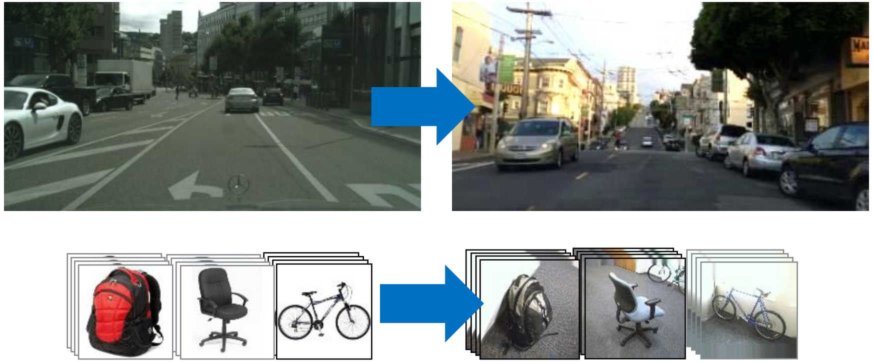
ADVERSARIAL TECHNIQUES

FEATURE ALIGNMENT
ADVERSARIAL DROPOUT

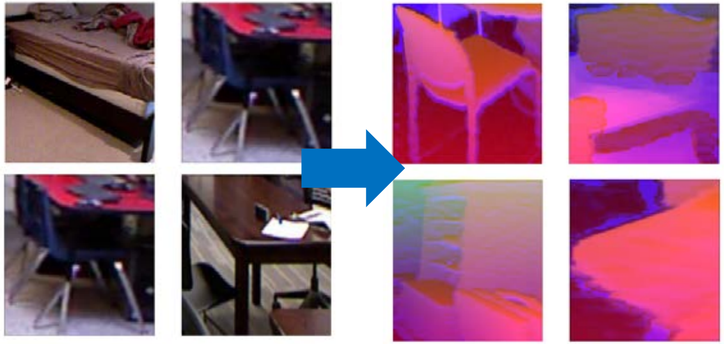
OPEN PROBLEMS

Domain Adaptation: transfer models

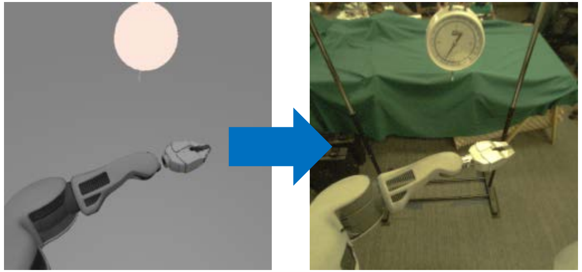
From dataset to dataset



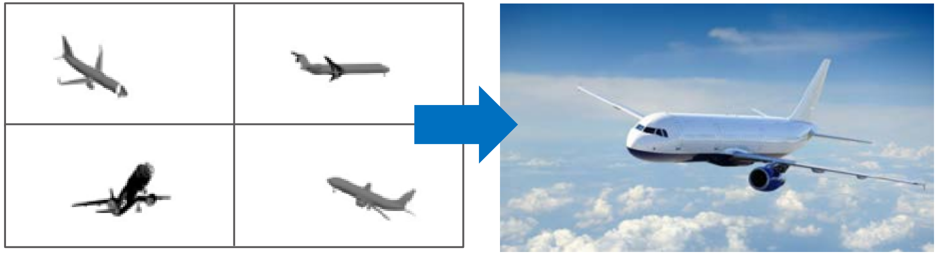
From RGB to depth



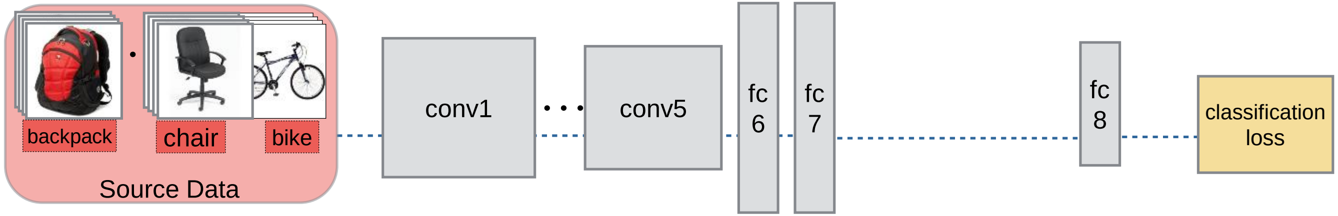
From simulated to real control



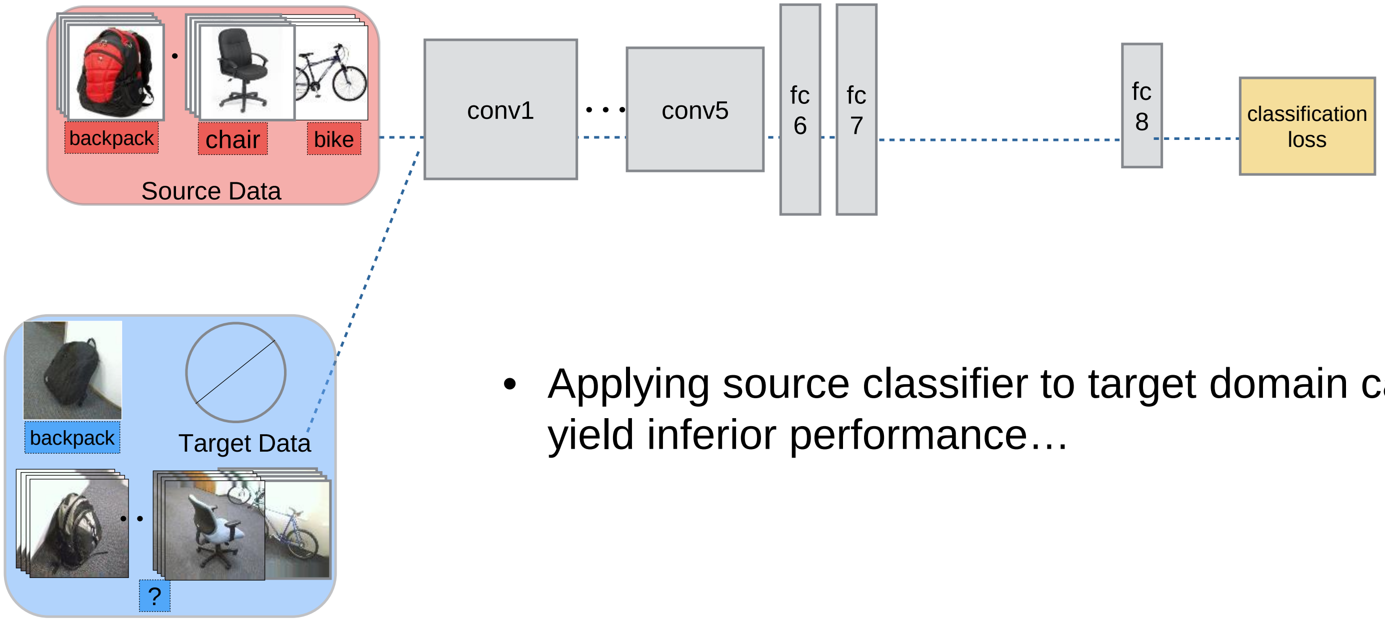
From CAD models to real images



How to adapt a deep network?



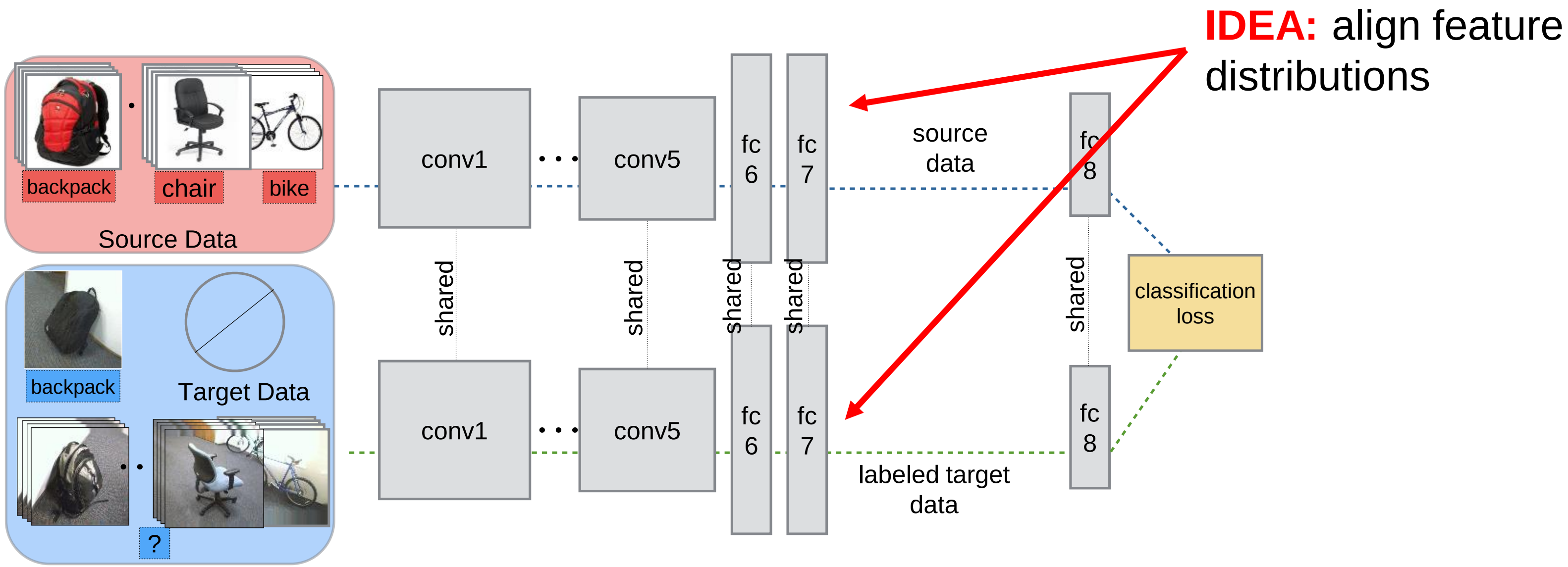
How to adapt a deep network?



- Applying source classifier to target domain can yield inferior performance…

Kate Saenko, Boston University

How to adapt a deep network?



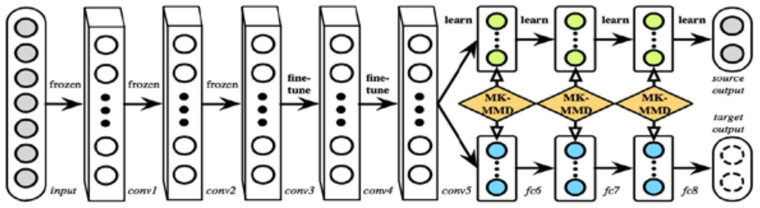
- Fine tune?

  …..Zero or few labels in target domain

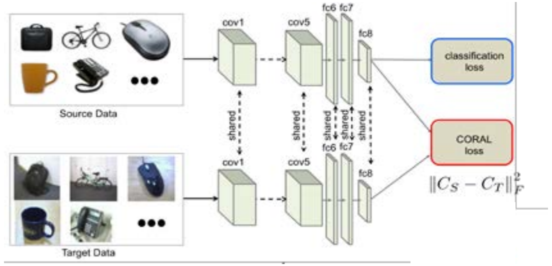
Kate Saenko, Boston University

Solution: align deep feature distributions

- by minimizing **distance** between distributions, e.g.

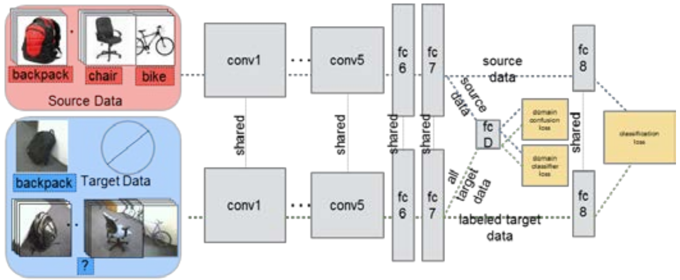


Maximum Mean Discrepancy M. Long, et al. ICML 2015

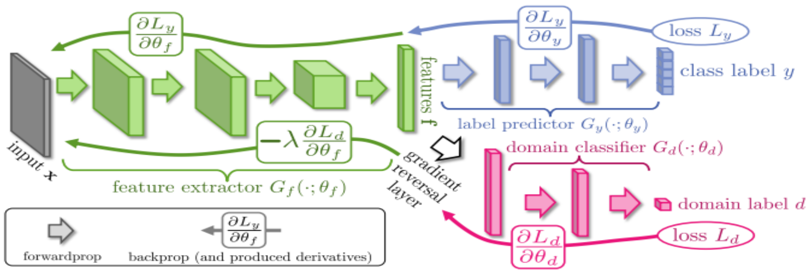


CORrelation ALignment Sun and Saenko, AAAI 2016

- …or by **adversarial** domain alignment, e.g.



Domain Confusion E. Tzeng et al. ICCV 2015



Reverse Gradient Y. Ganin and V. Lempitsky ICML 2015

WHAT IS DOMAIN ADAPTATION

▶ ADVERSARIAL TECHNIQUES

- FEATURE ALIGNMENT
- ADVERSARIAL DROPOUT

OPEN PROBLEMS

WHAT IS DOMAIN ADAPTATION

ADVERSARIAL TECHNIQUES

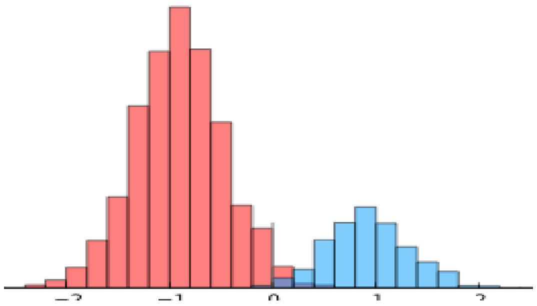
- FEATURE ALIGNMENT
- ADVERSARIAL DROPOUT

OPEN PROBLEMS

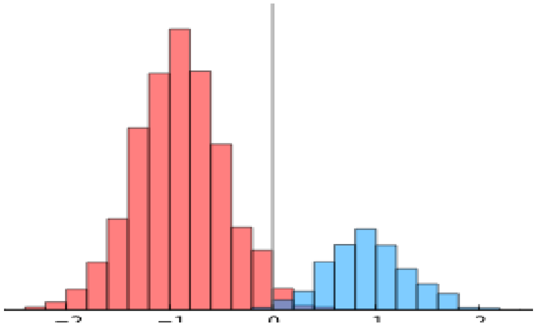
Adversarial networks

Adversarial networks

Encoder *P*　　*Reference* *Q*



P　　*Q*



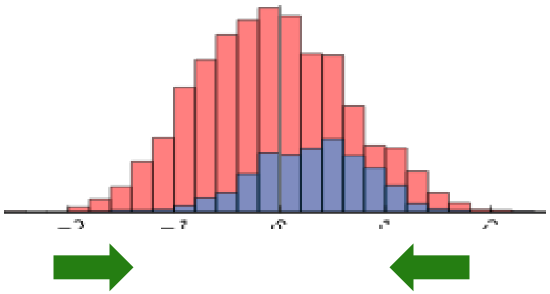
Encoder
Generates features such that their distribution P matches reference distribution Q

Adversary
Tries to discriminate between samples from P and samples from Q

Adversarial networks

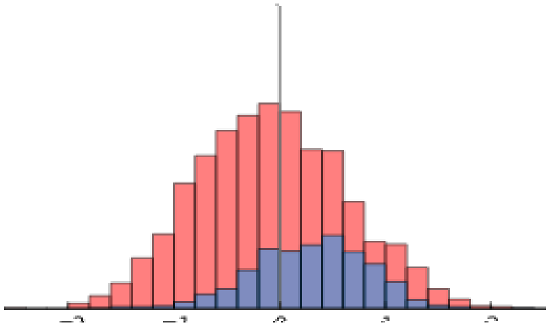
Encoder
P

Reference
Q



P

Q



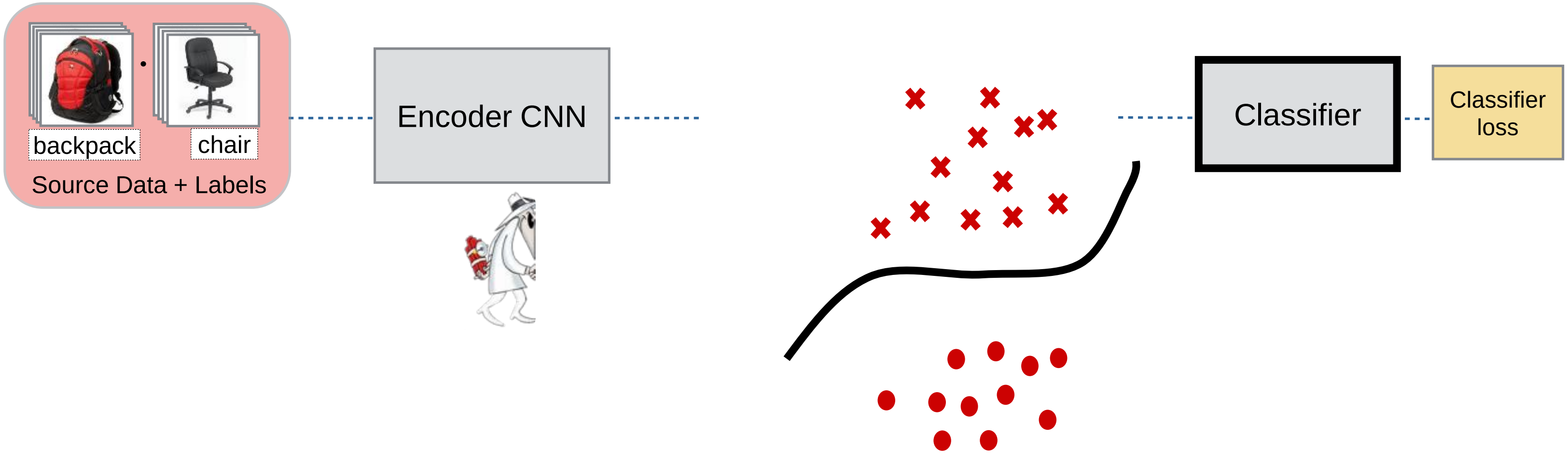
Encoder
Generates features such that their distribution P matches reference distribution Q

fools adversary

Adversary
Tries to discriminate between samples from P and samples from Q

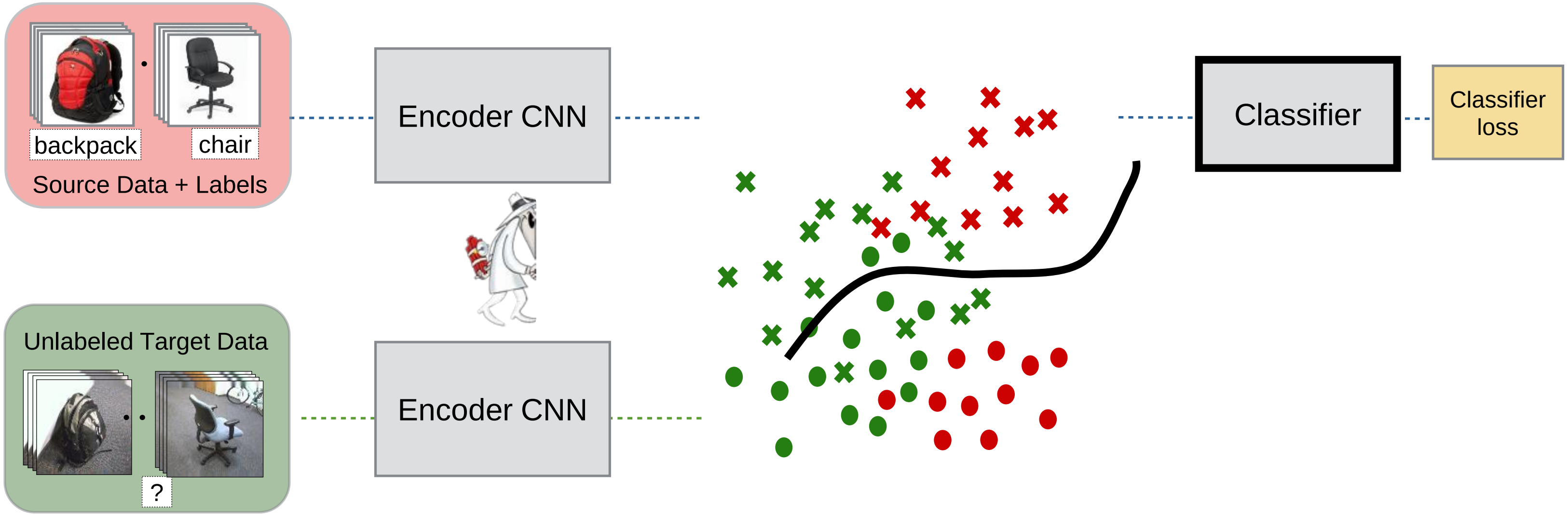
tries harder

Adversarial domain adaptation

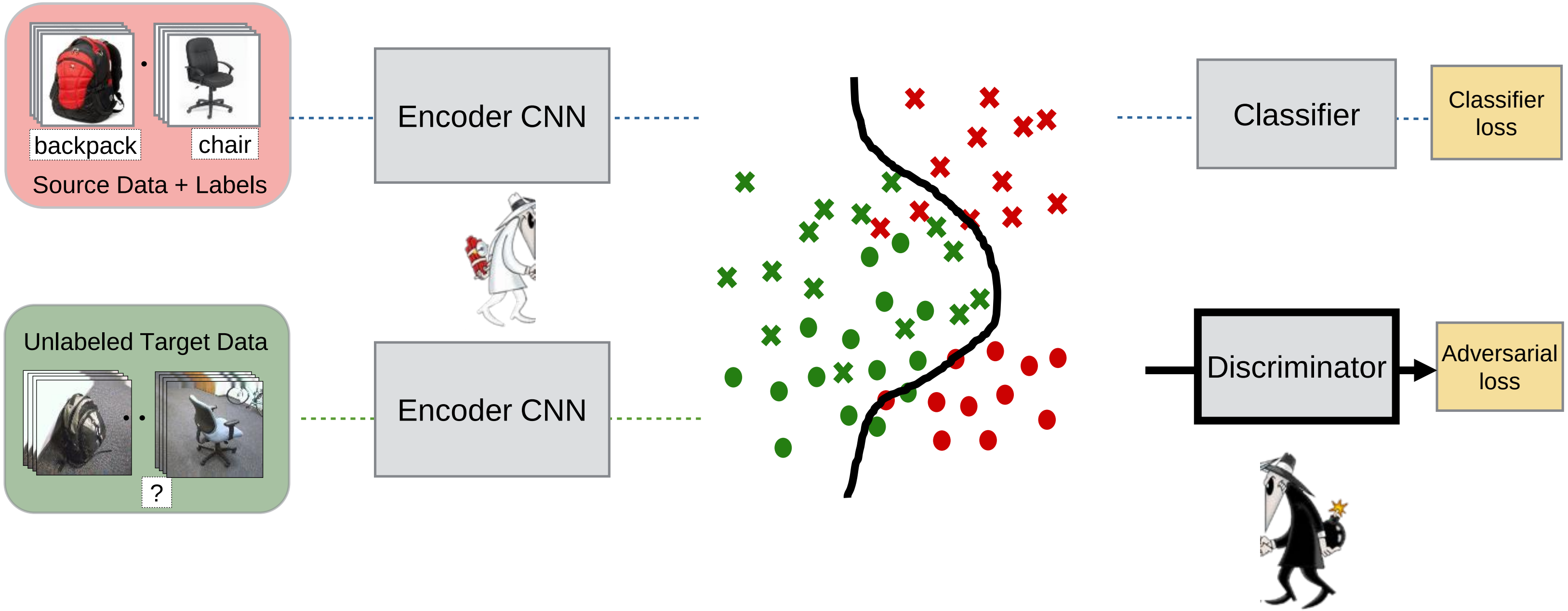


Kate Saenko, Boston University

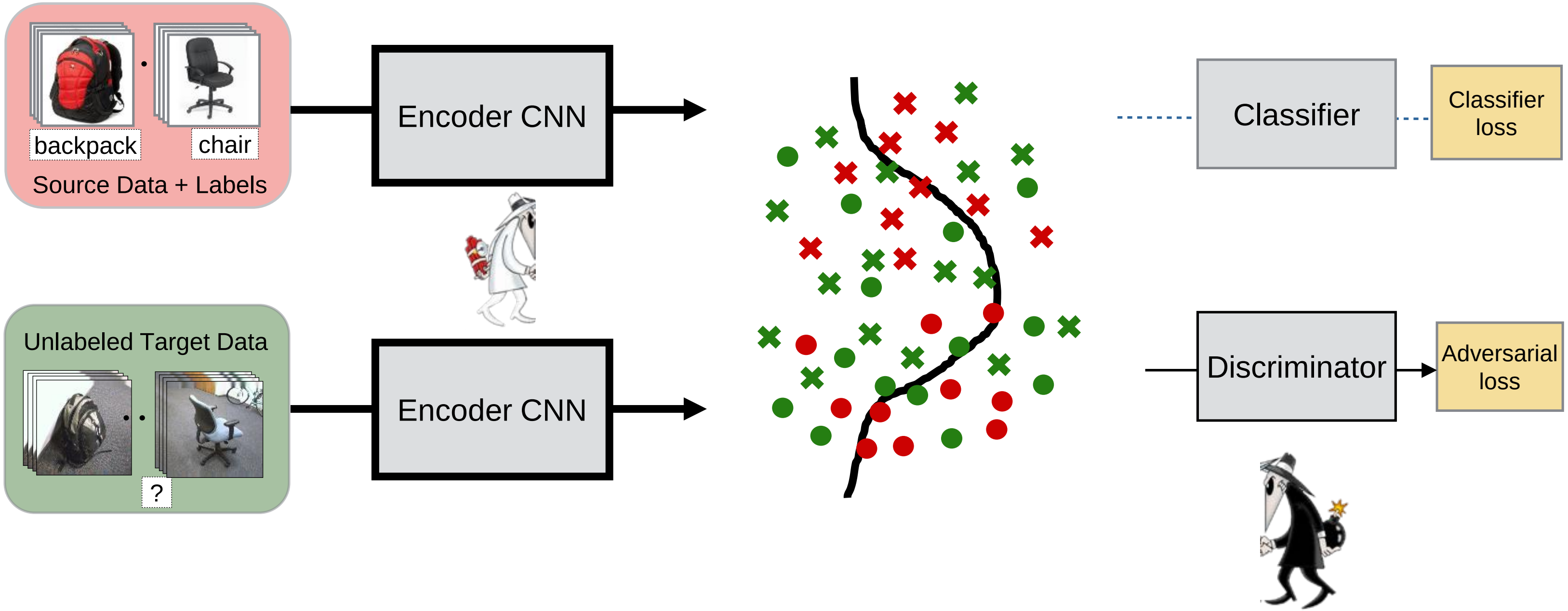
Adversarial domain adaptation



Adversarial domain adaptation

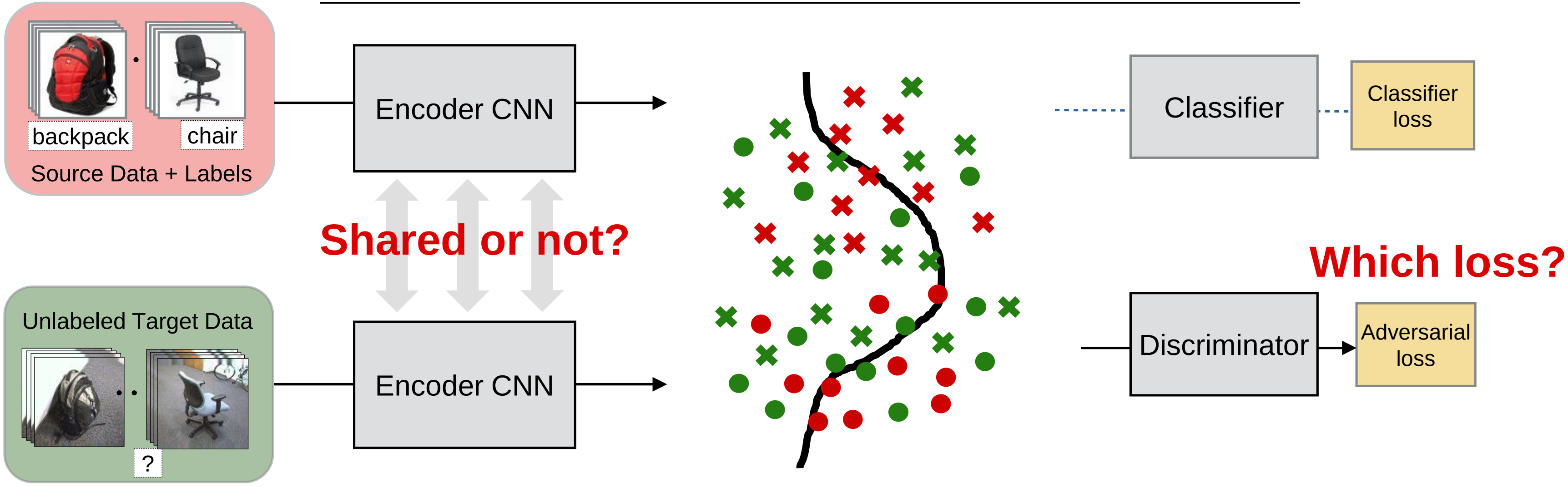


Adversarial domain adaptation



Design choices

Method	Weight sharing	Adversarial loss
Gradient Reversal [Ganin 2015]	shared	minimax
Domain Confusion [Tzeng ICCV15]	shared	confusion
ADDA [Tzeng CVPR17]	unshared	GAN



Kate Saenko, Boston University

ADDA: Adaptation on digits

[Tzeng, Hoffman, Darrell, **Saenko** CVPR17]

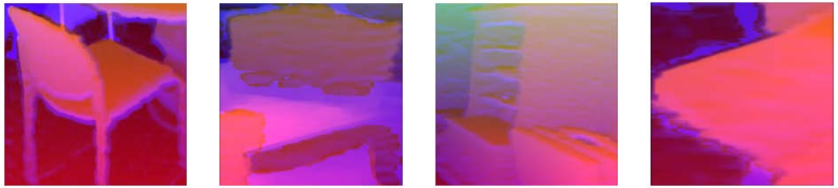
Method	MNIST → USPS	USPS → MNIST	SVHN → MNIST
Source only	0.752 ± 0.016	0.571 ± 0.017	0.601 ± 0.011
Gradient Reversal [Ganin'15]	0.771 ± 0.018	0.730 ± 0.020	0.739 [16]
Domain Confusion [Tzeng'15]	0.791 ± 0.005	0.665 ± 0.033	0.681 ± 0.003
CoGAN	0.912 ± 0.008	0.891 ± 0.008	did not converge
ADDA (ours)	0.894 ± 0.002	0.901 ± 0.008	0.760 ± 0.018

ADDA: Adaptation on RGB-D

[Tzeng, Hoffman, Darrell, **Saenko** CVPR17]

Train on RGB

Test on depth



	bathtub	bed	bookshelf	box	chair	counter	desk	door	dresser	garbage bin	lamp	monitor	night stand	pillow	sink	sofa	table	television	toilet	**overall**
# of instances	19	96	87	210	611	103	122	129	25	55	144	37	51	276	47	129	210	33	17	2401
Source only	0.000	0.010	0.011	0.124	0.188	0.029	0.041	0.047	0.000	0.000	0.069	0.000	0.039	0.587	0.000	0.008	0.010	0.000	0.000	0.139
ADDA (Ours)	0.000	0.146	0.046	0.229	0.344	0.447	0.025	0.023	0.000	0.018	0.292	0.081	0.020	0.297	0.021	0.116	0.143	0.091	0.000	0.211
Train on target	0.105	0.531	0.494	0.295	0.619	0.573	0.057	0.636	0.120	0.291	0.576	0.189	0.235	0.630	0.362	0.248	0.357	0.303	0.647	0.468

Recent work on adversarial deep alignment

- **Learning Transferrable Representations for Unsupervised Domain Adaptation**, Ozan Sener, Hyun Oh Song, Ashutosh Saxena, Silvio Savarese, NIPS 2016

- **Unsupervised Image-to-Image Translation Networks**, Ming-Yu Liu, Thomas Breuel, Jan Kautz, 2017

- **Unsupervised Pixel–Level Domain Adaptation with Generative Adversarial Networks**, Konstantinos Bousmalis, Nathan Silberman, David Dohan, Dumitru Erhan, Dilip Krishnan, CVPR 2017

- **Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks**, Jun-Yan Zhu, Taesung Park, Phillip Isola, Alexei A. Efros

- and more…

WHAT IS DOMAIN ADAPTATION

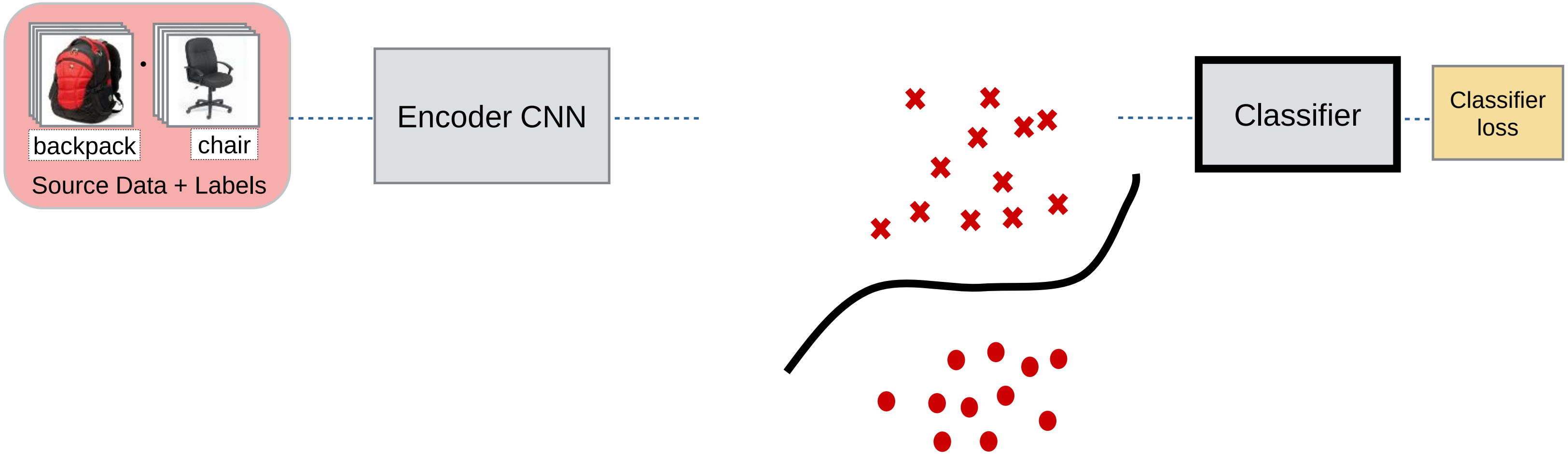
ADVERSARIAL TECHNIQUES

FEATURE ALIGNMENT

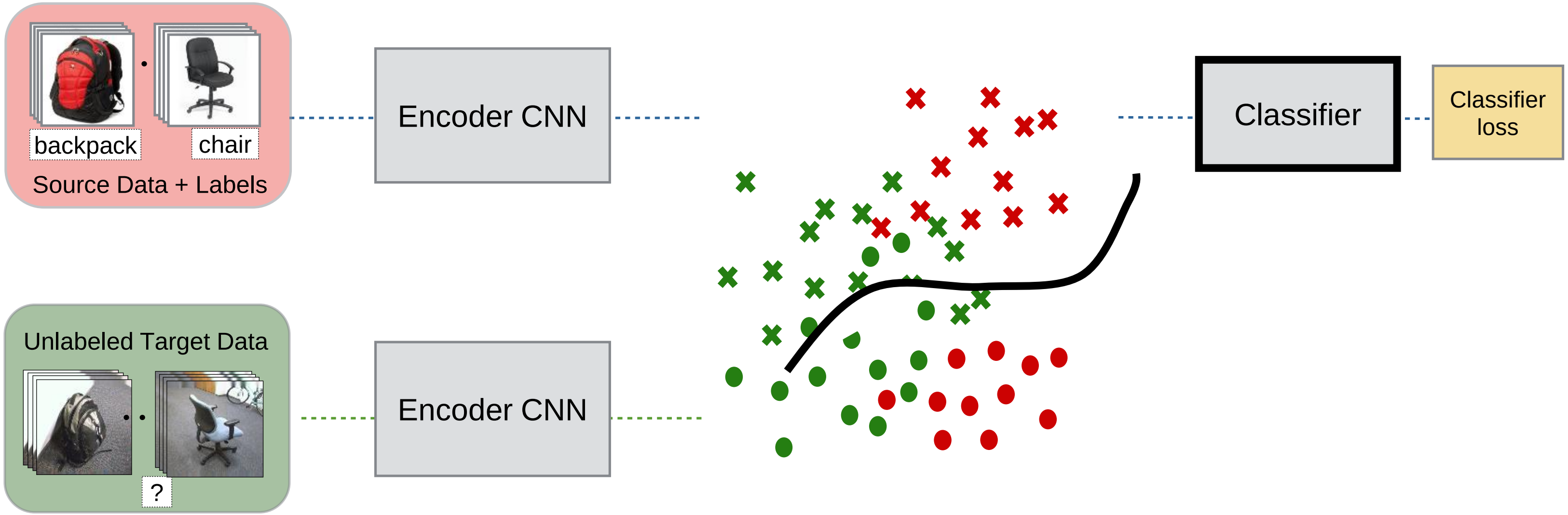
ADVERSARIAL DROPOUT

OPEN PROBLEMS

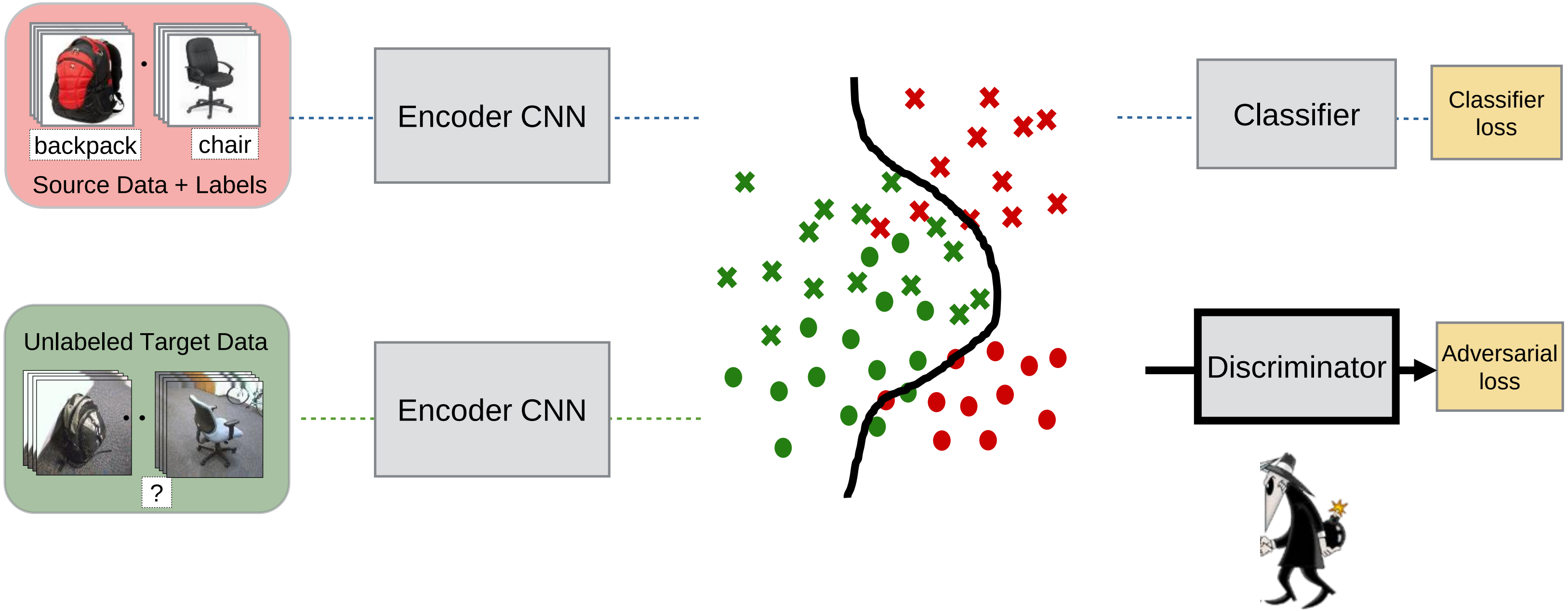
Problem with adversarial alignment



Problem with adversarial alignment

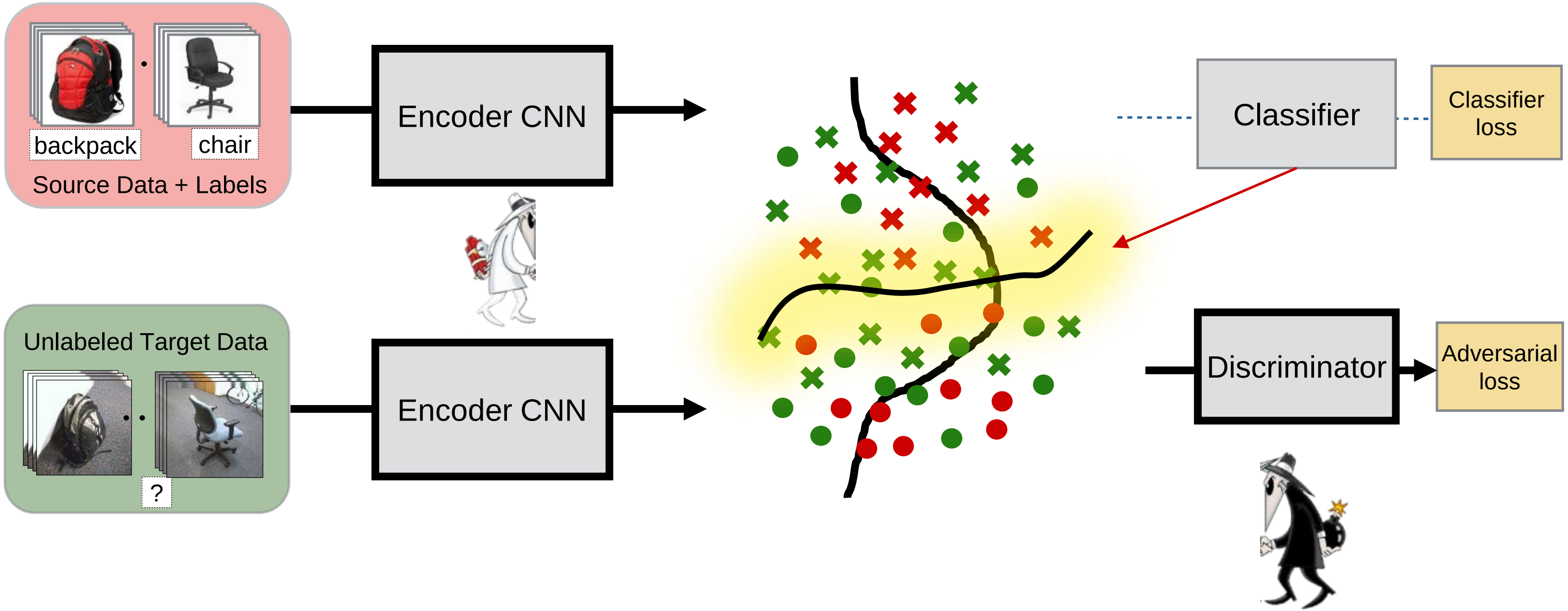


Problem with adversarial alignment



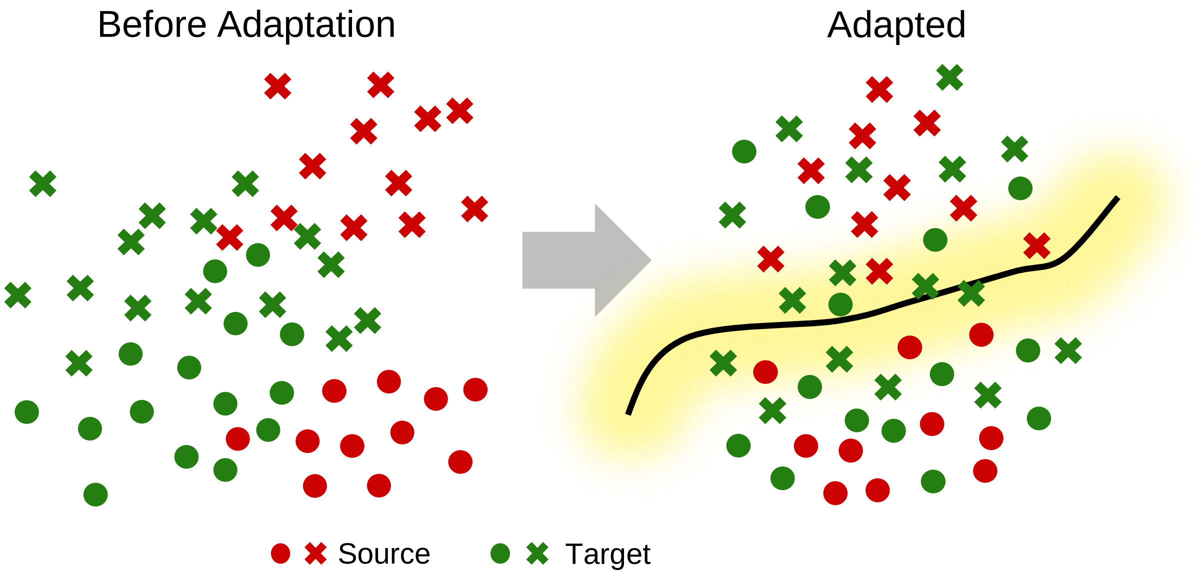
Kate Saenko, Boston University

Problem with adversarial alignment

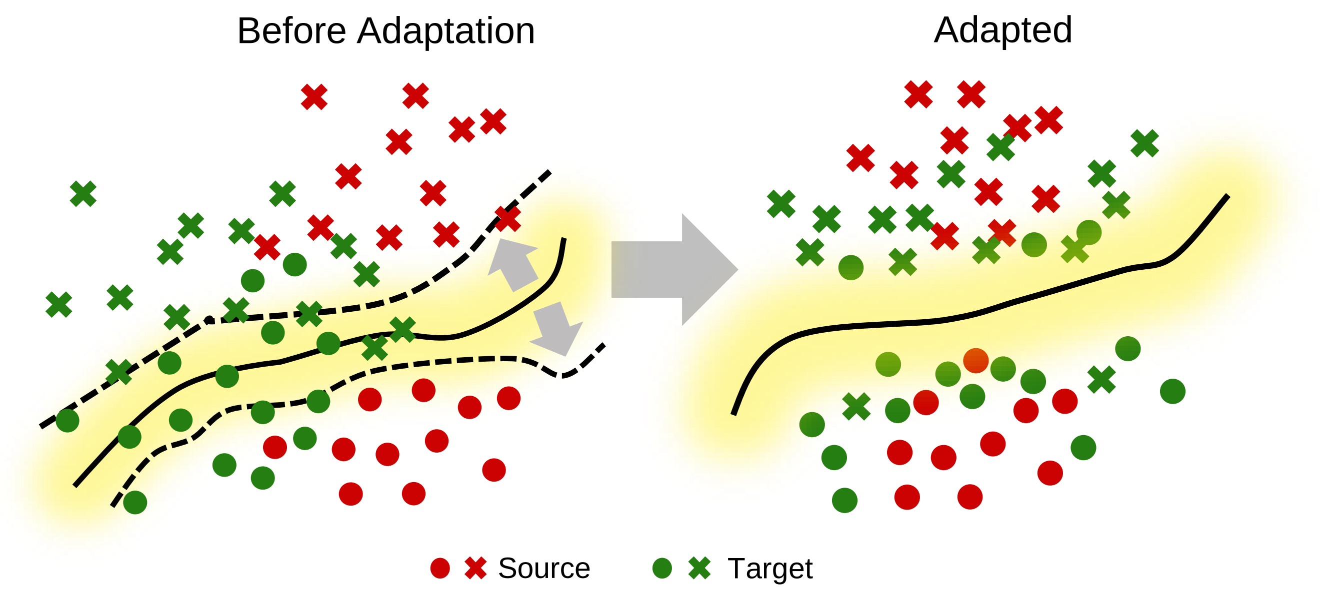


Kate Saenko, Boston University

Problem: ambiguous features



Goal: avoid generating ambiguous features

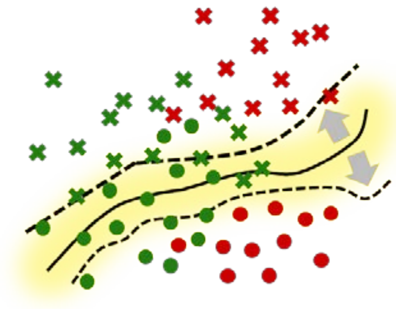


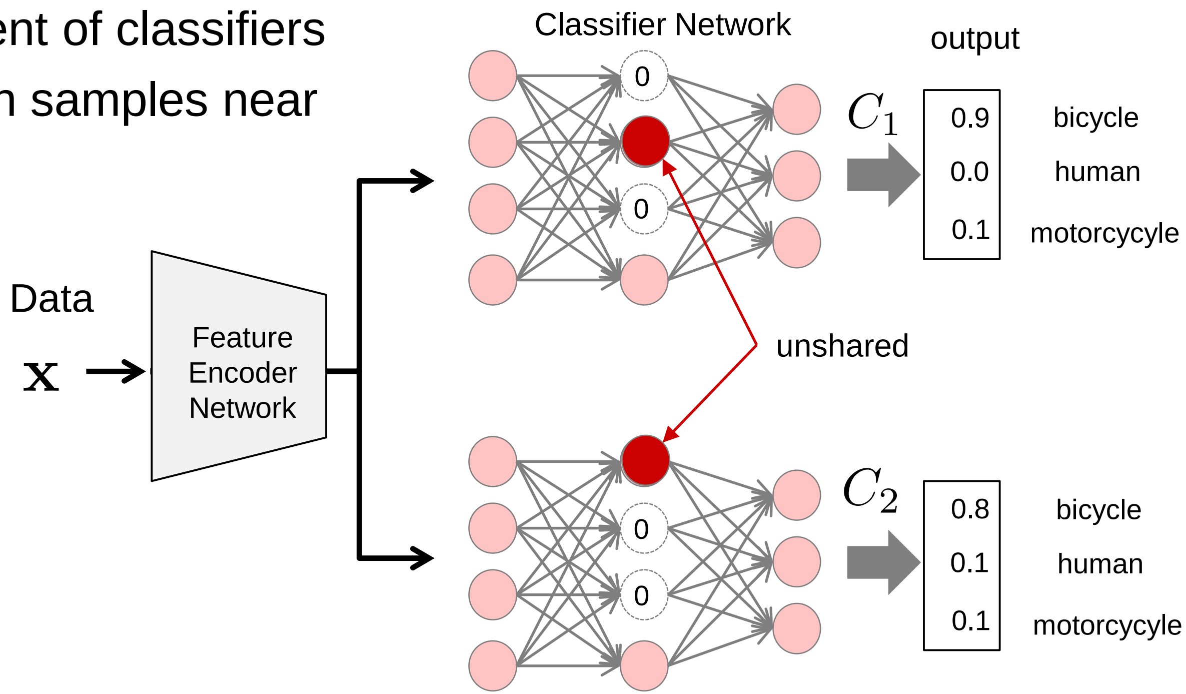
Solution: Train a discriminator sensitive to target samples near decision boundary
Train a generator to fool the critic

Adversarial Dropout Regularization

[Saito, Ushiku, Harada, **Saenko** ICLR18]

- Sample classifiers C_1, C_2 from the same network via dropout
- Maximize disagreement of classifiers
- Most disagreement on samples near boundary!

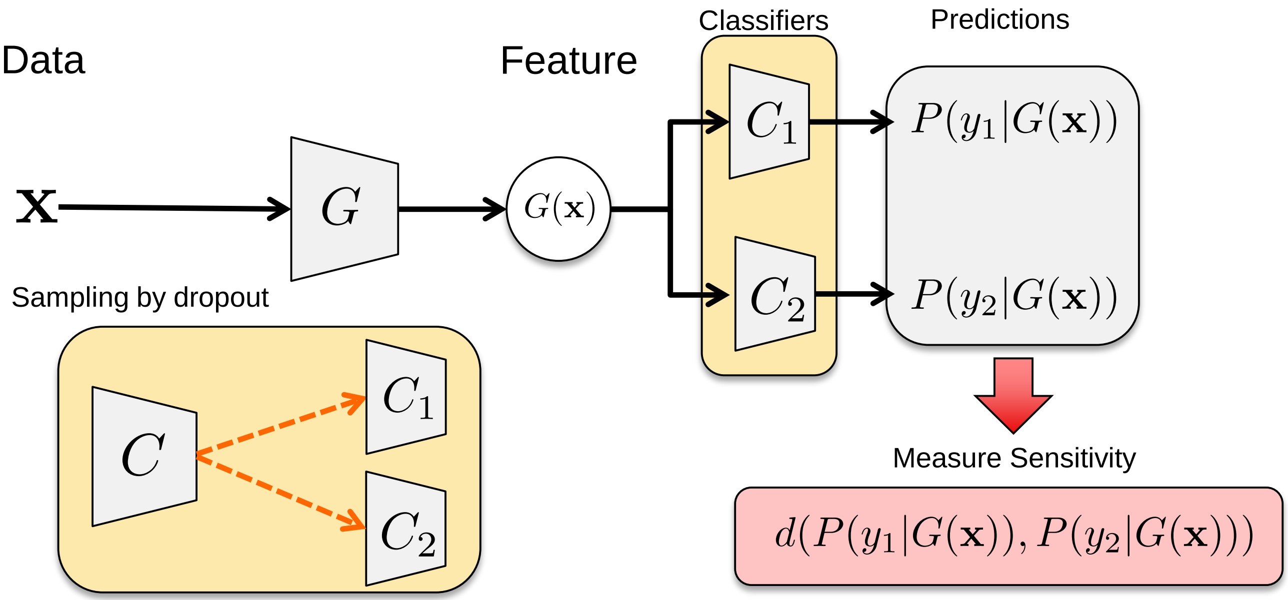




Kate Saenko, Boston University

Adversarial Dropout Regularization

[Saito, Ushiku, Harada, **Saenko** ICLR18]



1. Fix **G** and train **C** to maximize **d(p_1,p_2)** for target samples.
2. Train **G** and **C** to minimize CrossEntropy for source samples.
3. Fix **C** and train **G** to minimize **d(p_1,p_2) for target**.

ADR on Digits Classification

[Saito, Ushiku, Harada, **Saenko** ICLR18]

MNIST

USPS

SVHN

METHOD	SVHN to MNIST	USPS to MNIST	MNIST to USPS
Source Only	67.1	68.1	77.0
ATDA (Saito et al. (2017))	86.2†	-	-
DANN (Ganin & Lempitsky (2014))	73.9	73.0±2.0	77.1±1.8
DoC (Tzeng et al. (2014))	68.1±0.3	66.5±3.3	79.1±0.5
ADDA (Tzeng et al. (2017))	76.0±1.8	90.1±0.8	89.4±0.2
CoGAN (Liu & Tuzel (2016))	did not converge	89.1±0.8	91.2±0.8
DTN (Taigman et al. (2016))	84.7	-	-
Ours	**96.7**±1.85	**91.5**±3.61	**91.3**±0.65

Simulation to reality

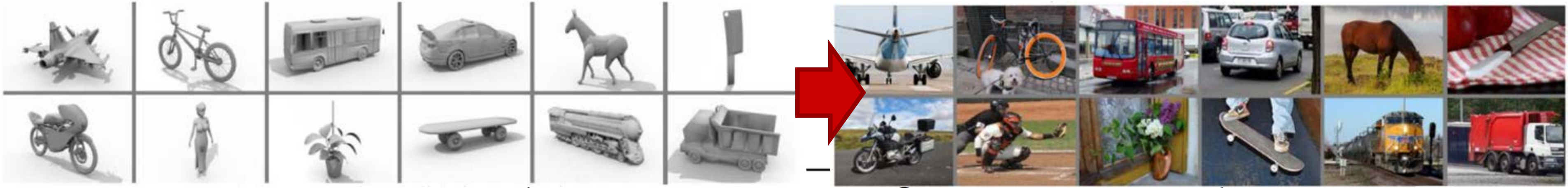
Frame from the Grand Theft Auto game

Kate Saenko, Boston University

ADR Sim2real classification results

[Saito, Ushiku, Harada, **Saenko** ICLR18]

VisDA Challenge 2017 Object Classification



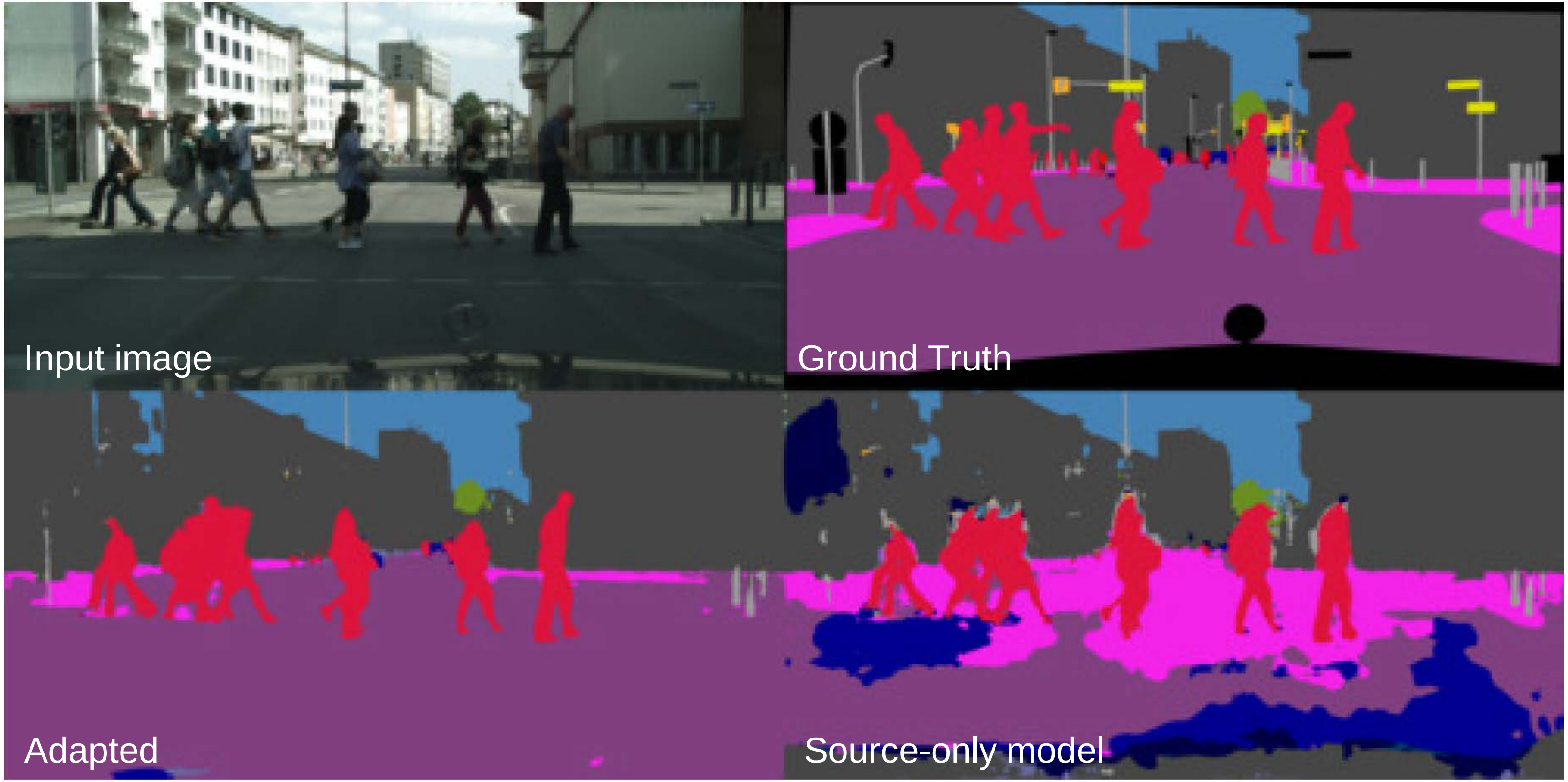
Method	aeroplane	bicycle	bus	car	horse	knife	motorcycl	person	plant	skateboard	train	truck	mean
Finetuning on ResNet101													
Source Only	55.1	53.3	61.9	59.1	80.6	17.9	79.7	31.2	81.0	26.5	73.5	8.5	52.4
MMD	87.1	63.0	76.5	42.0	90.3	42.9	85.9	53.1	49.7	36.3	85.8	20.7	61.1
Ganin & Lempitsky (2014)	81.9	77.7	82.8	44.3	81.2	29.5	65.1	28.6	51.9	54.6	82.8	7.8	57.4
Ours	85.1	77.3	78.9	64.0	91.6	52.7	**91.3**	**75.8**	**88.5**	**60.5**	**87.7**	**33.9**	73.9
Ours (retrain classifer)	**87.8**	**79.5**	**83.7**	**65.3**	**92.3**	**61.8**	88.9	73.2	87.8	60.0	85.5	32.3	**74.8**
Finetuning on ResNeXt													
Source Only	74.3	37.6	61.8	68.2	59.5	10.7	81.4	12.8	61.6	26.0	70.0	5.6	47.4
MMD	90.7	51.1	64.8	65.6	89.9	46.5	91.9	40.1	81.5	24.1	**90.0**	28.5	63.7
Ganin & Lempitsky (2014)	86.0	66.3	60.8	56.0	79.8	53.7	82.3	25.2	58.2	31.0	89.3	26.1	59.6
Ours	93.5	81.1	82.2	68.3	92.0	67.1	90.8	**80.1**	**92.6**	**72.1**	87.6	42.0	79.1
Ours (retrain classifer)	**94.6**	**82.6**	**84.4**	**73.3**	**94.2**	**83.4**	**92.2**	76.1	91.5	55.1	85.2	**43.2**	**79.6**

Sim2real classification: adapted features

[Saito et al. ICLR18]

(a) Pretrained Model (b) Source Only Model (c) Adapted Model

Sim2real transfer for semantic segmentation [Saito et al. ICLR18]



WHAT IS DOMAIN ADAPTATION

ADVERSARIAL TECHNIQUES

FEATURE ALIGNMENT
ADVERSARIAL DROPOUT

OPEN PROBLEMS

Is pixel-to-pixel adaptation better? [Hoffman et al. 2017]

Source only | **CyCADA**

Source: SVHN

→

Target: MNIST

Target Accuracy: 62.3%

Target Accuracy: **86.6%**

CyCADA: Cycle-Consistent Adversarial Domain Adaptation, Judy Hoffman, Eric Tzeng, Taesung Park, Jun-Yan Zhu, Phillip Isola, Kate Saenko, Alexei A. Efros,Trevor Darrell

Many open problems

Is adversarial learning the answer?

- improve stability, e.g. [Usman, Kulis, Saenko ICLRW'18]

Can we handle unknown classes? What about tasks beyond classification, e.g., object detection, pose estimation?

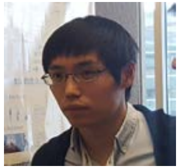
- VisDA: domain adaptation challenge May-Sep 2018

Can DA be applied to robotic manipulation strategies?

- Learn to grasp objects in simulation, transfer to real world

http://ai.bu.edu/visda-2018/

Thanks



Kuniaki Saito

Eric Tzeng

Judy Hoffman

AIR: AI Research Initiative at BU

- Started by a core group of CS/EE faculty doing research on learning for vision, language, robotics

Stan Sclaroff Kate Saenko Margrit Betke Brian Kulis

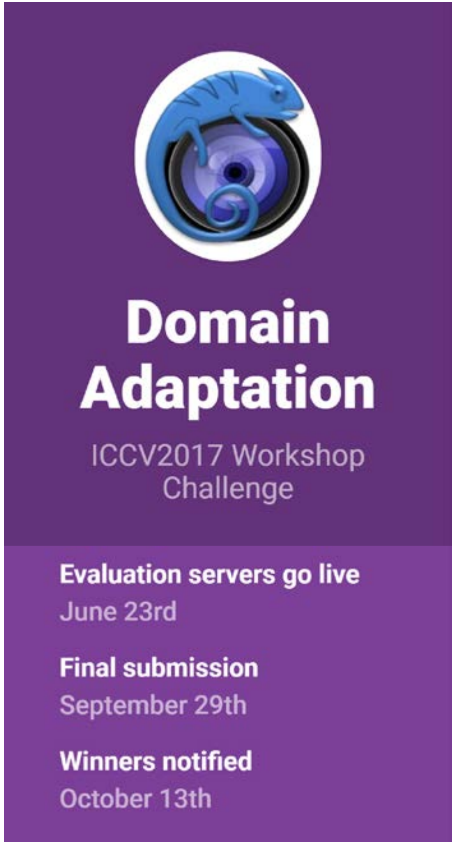
- Promote AI research, seminars, teaching, industry outreach

References

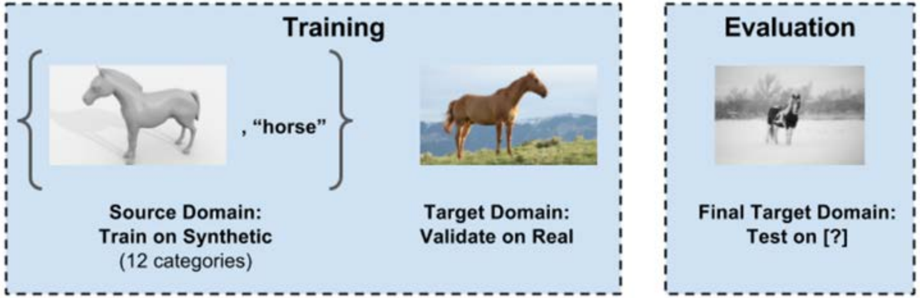
- Eric Tzeng, Judy Hoffman, Trevor Darrell, Kate Saenko, Simultaneous Deep Transfer Across Domains and Tasks, ICCV 2015
- Eric Tzeng, Coline Devin, Judy Hoffman, Chelsea Finn, Pieter Abbeel, Sergey Levine, Kate Saenko, Trevor Darrell, Adapting Deep Visuomotor Representations with Weak Pairwise Constraints, WAFR 2016
- Baochen Sun, Jiashi Feng, Kate Saenko, Return of Frustratingly Easy Domain Adaptation, AAAI 2016
- Baochen Sun, Kate Saenko, Deep CORAL: Correlation Alignment for Deep Domain Adaptation, TASK-CV Workshop at ICCV 2016
- Eric Tzeng, Judy Hoffman, Trevor Darrell, Kate Saenko, Adversarial Discriminative Domain Adaptation, accepted to CVPR 2017
- Synthetic to Real Adaptation with Deep Generative Correlation Alignment Networks, arxiv.org

Kate Saenko, Boston University

VisDA Challenge 2017



Classification Track



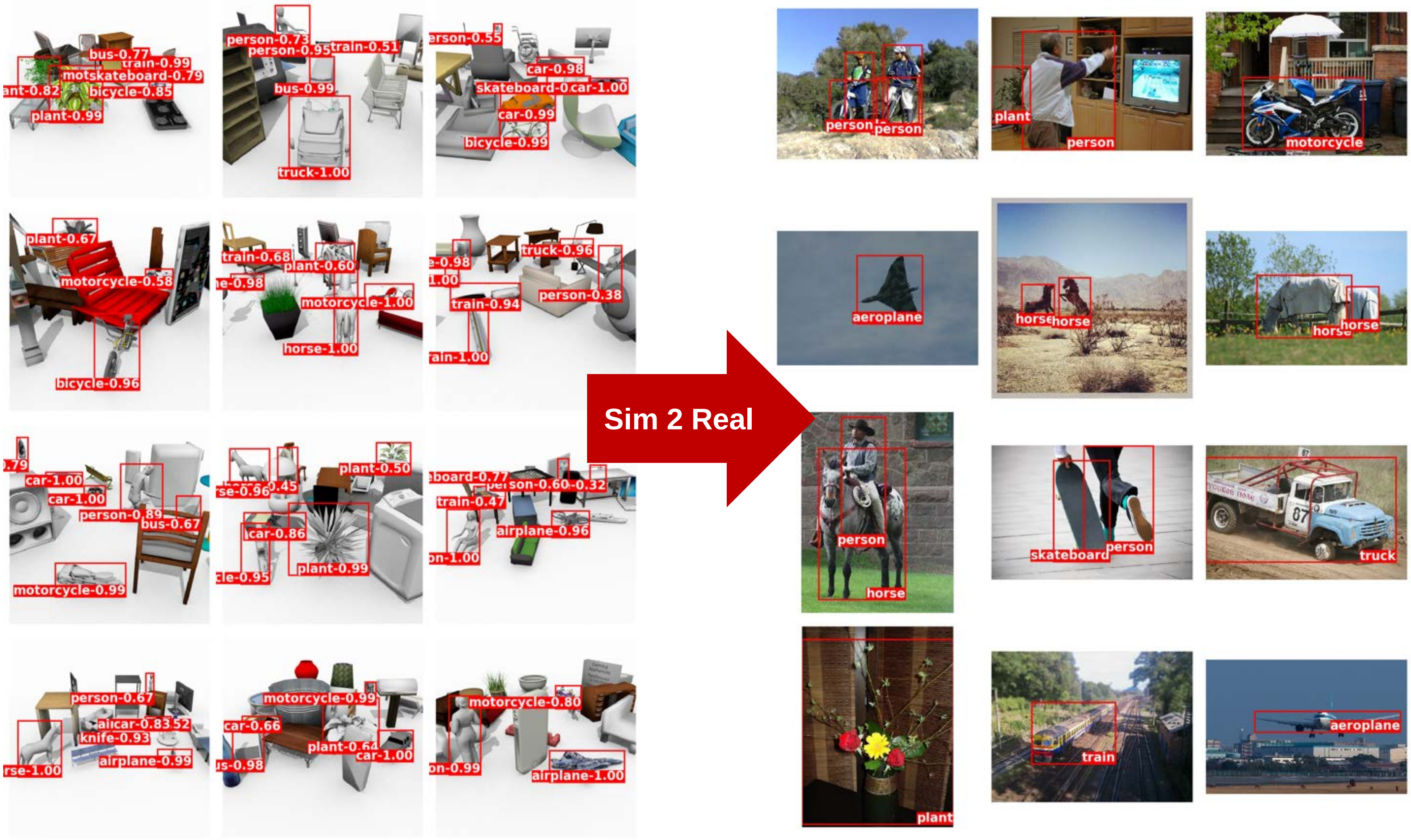
Semantic Segmentation Track

Grand Theft Auto

Source Domain

Real Dashcam Video

Target Domain



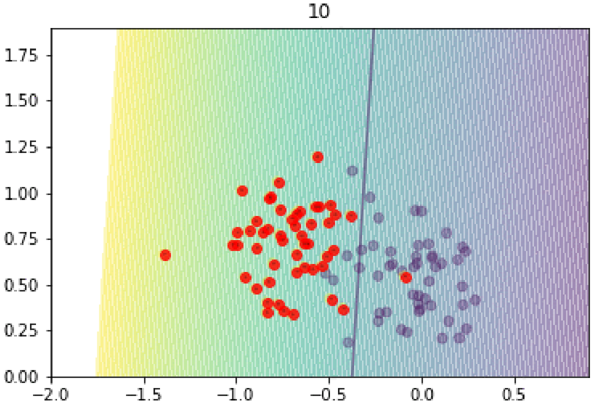
Sim 2 Real
bus-0.77
train-0.99
motskateboard-0.79
ant-0.82
bicycle-0.85
plant-0.99
person-0.73
person-0.95
train-0.51
bus-0.99
truck-1.00
erson-0.55
car-0.98
skateboard-0.car-1.00
car-0.99
bicycle-0.99
plant-0.67
motorcycle-0.58
bicycle-0.96
train-0.68
plant-0.60
ne-0.98
motorcycle-1.00
horse-1.00
e-0.98
1.00
truck-0.96
train-0.94
person-0.38
ain-1.00
.79
car-1.00
car-1.00
person-0.89
bus-0.67
motorcycle-0.99
se-0.96
0.45
plant-0.50
car-0.86
plant-0.99
cle-0.95
board-0.77
person-0.60-0.32
train-0.47
airplane-0.96
on-1.00
person-0.67
aicar-0.83
52
knife-0.93
airplane-0.99
rse-1.00
motorcycle-0.99
car-0.66
plant-0.64
car-1.00
us-0.98
motorcycle-0.80
on-0.99
airplane-1.00
person
person
plant
person
motorcycle
aeroplane
horse
horse
horse
horse
person
horse
skateboard
person
truck
plant
train
aeroplane

Kate Saenko, Boston University

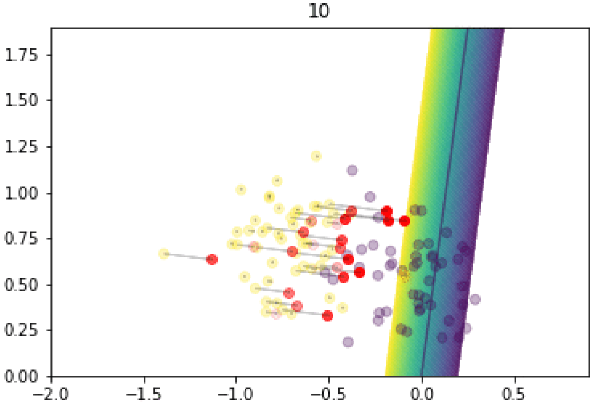
Dual formulation for Stable alignment [Usman et al. 17]

- turn the adversarial min-max problem into a min-min problem by replacing the maximization part with its dual
- empirically improves the quality of the resulting alignment

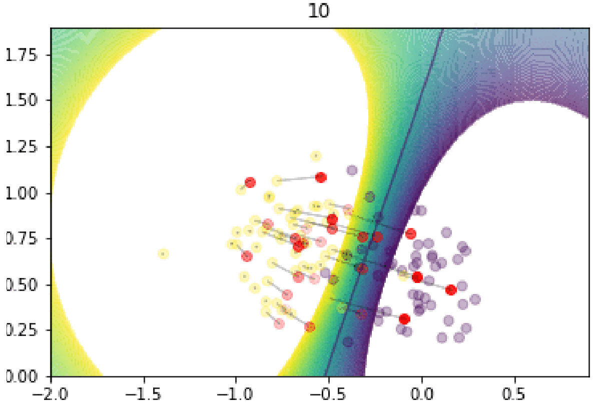
Linear primal



Linear dual



Kernel dual



Usman et al., *Stable Distribution Alignment Using the Dual of the Adversarial Distance.* https://arxiv.org/pdf/1707.04046.pdf