“What you saw is not what you get”

Domain adaptation for deep learning

Kate Saenko

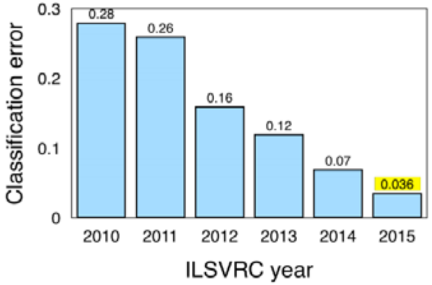
BOSTON UNIVERSITY

Successes of Deep Learning in AI

The New York Times

A Learning Advance in Artificial Intelligence Rivals Human Abilities

FLOWER
FIELD
SKY
CLOUDS



IMAGENET

Deep Learning for self-driving cars

Google's DeepMind Masters Atari Games

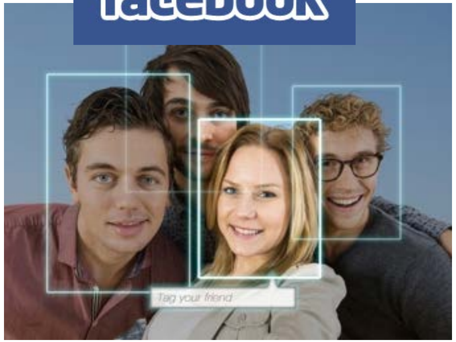
Google Translate

English | Russian | Chinese (Simplified)

Time flies like an arrow

时间过得很快像箭

facebook



Face Recognition

So is AI solved?

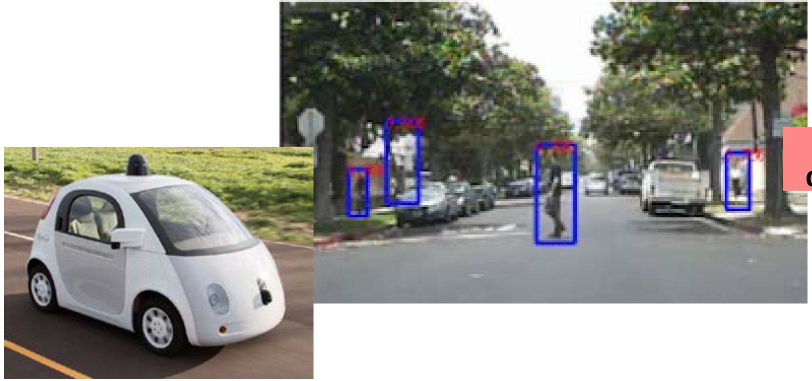
pedestrian detection FAIL

https://www.youtube.com/watch?v=w2pwxv8rFkU

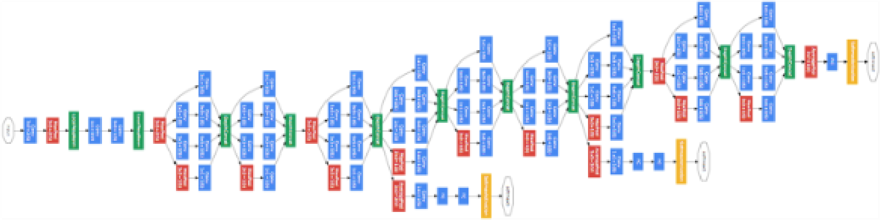
Major limitation of deep learning

Not data efficient: Learning requires **millions** of labeled examples,

models do not generalize well to new domains; not like humans!



New domain



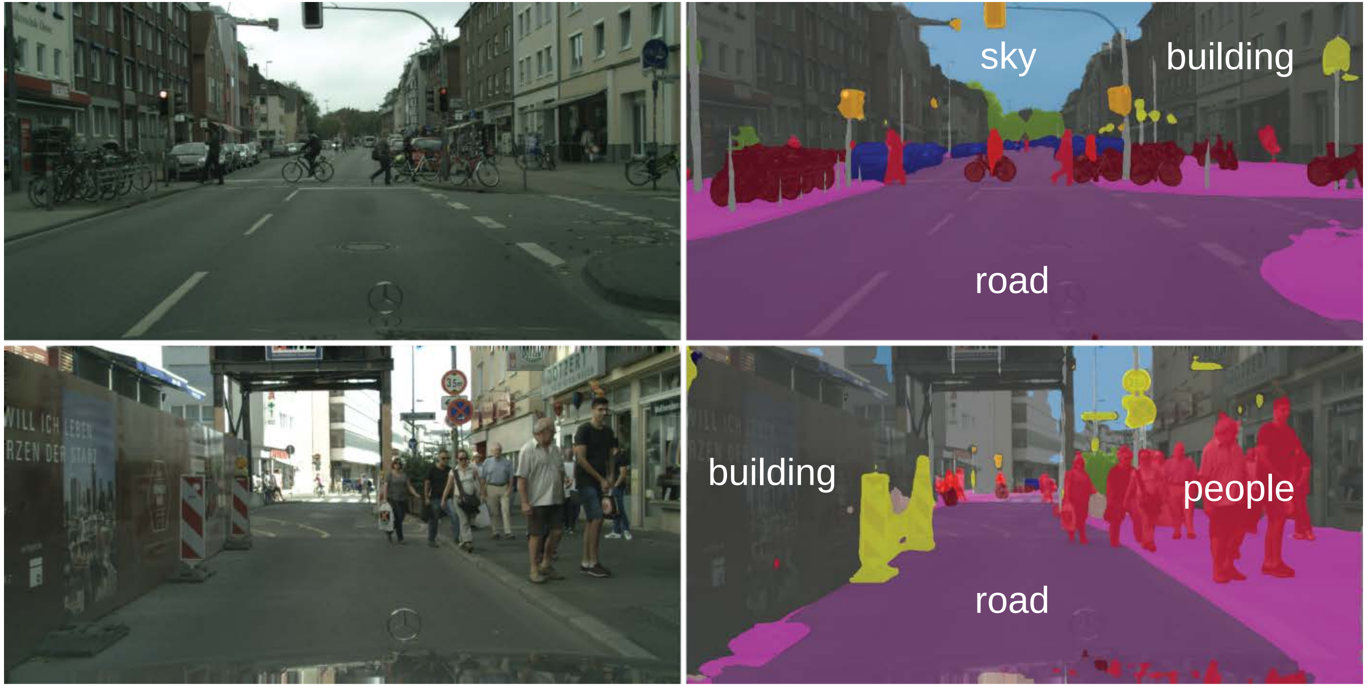
“What you saw is not what you get”

What your net is trained on

What it’s asked to label

“Dataset Bias”
“Domain Shift”
“Domain Adaptation”
“Domain Transfer”

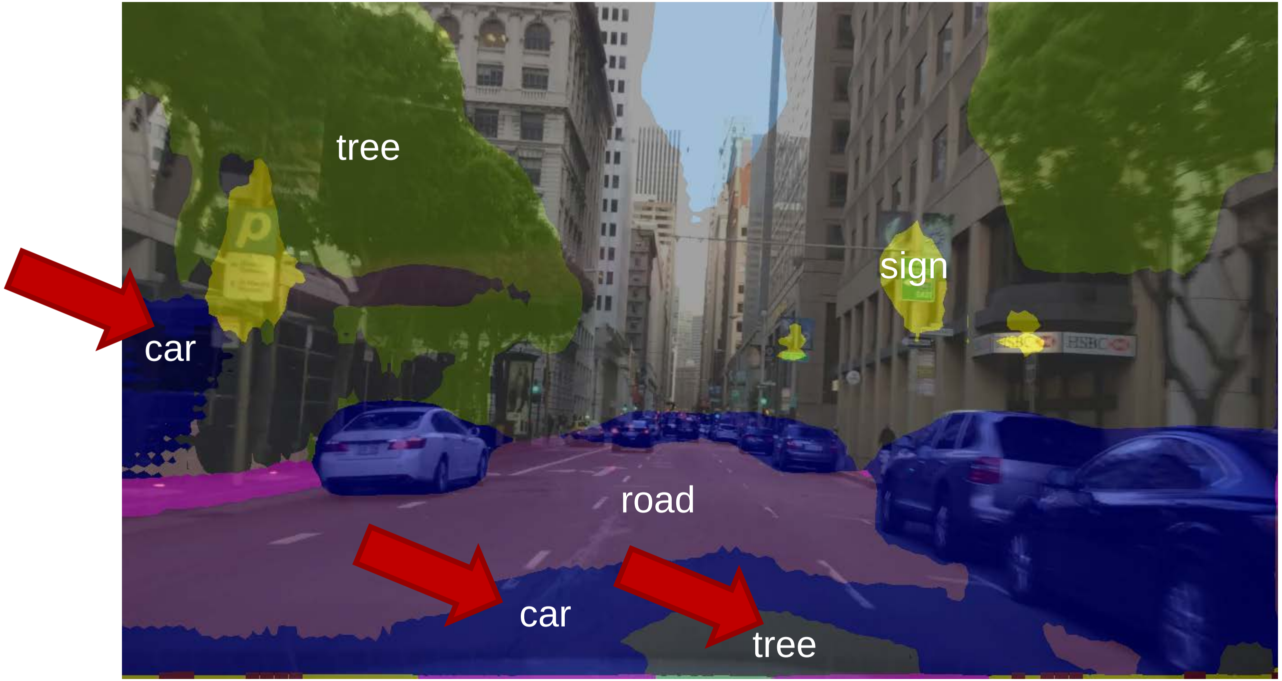
Example: scene segmentation



Train on Cityscapes, Test on Cityscapes

FCNs in the Wild: Pixel-level Adversarial and Constraint-based Adaptation, Judy Hoffman, Dequan Wang, Fisher Yu, Trevor Darrell, Arxiv 2016

Domain shift: Cityscapes to SF



Train on Cityscapes, Test on San Francisco Dashcam

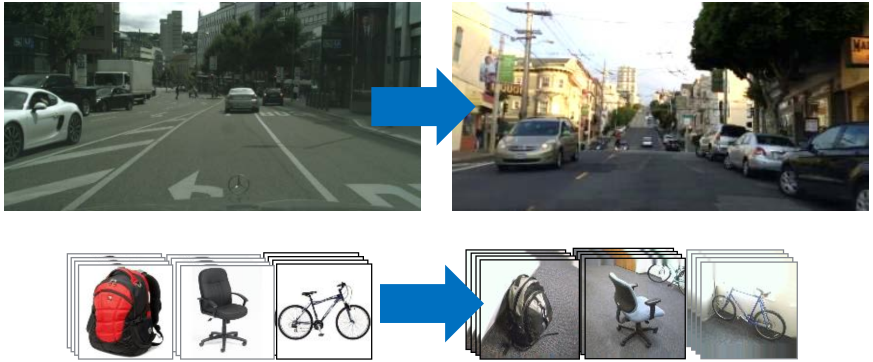
FCNs in the Wild: Pixel-level Adversarial and Constraint-based Adaptation, Judy Hoffman, Dequan Wang, Fisher Yu, Trevor Darrell, Arxiv 2016

No tunnels in CityScapes?...

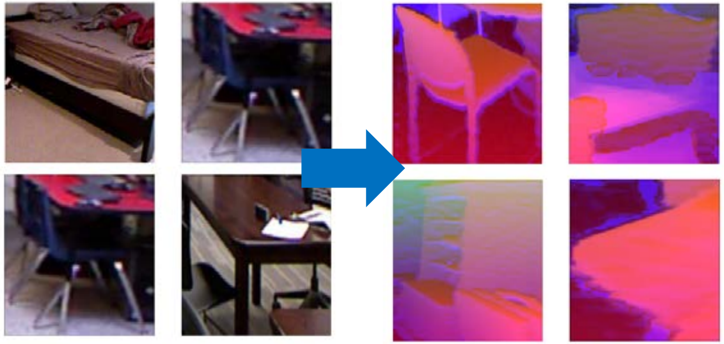
FCNs in the Wild: Pixel-level Adversarial and Constraint-based Adaptation, Judy Hoffman, Dequan Wang, Fisher Yu, Trevor Darrell, Arxiv 2016

Applications to different types of domain shift

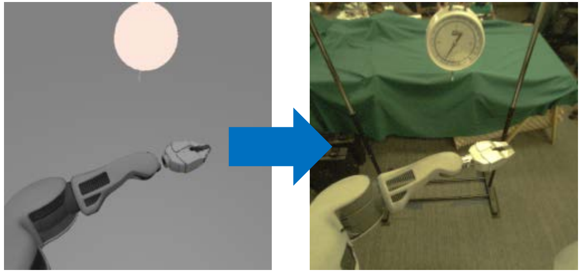
From dataset to dataset



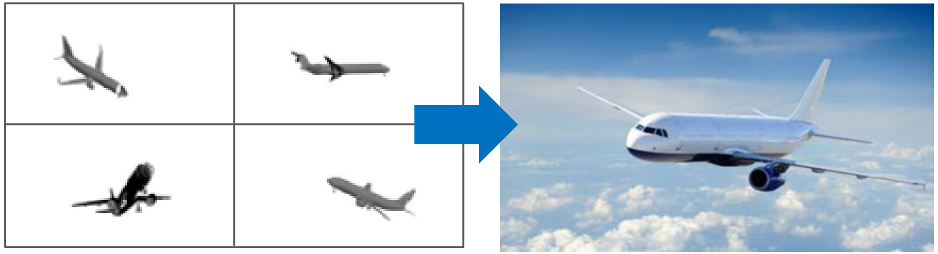
From RGB to depth



From simulated to real control

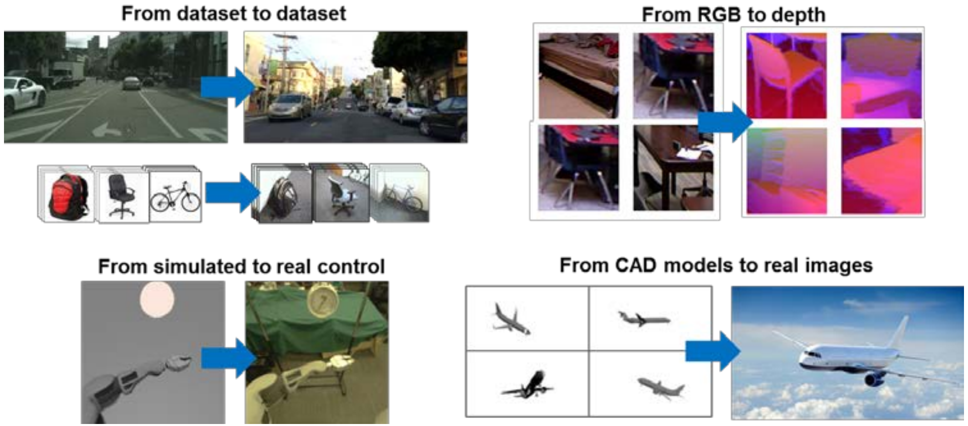


From CAD models to real images

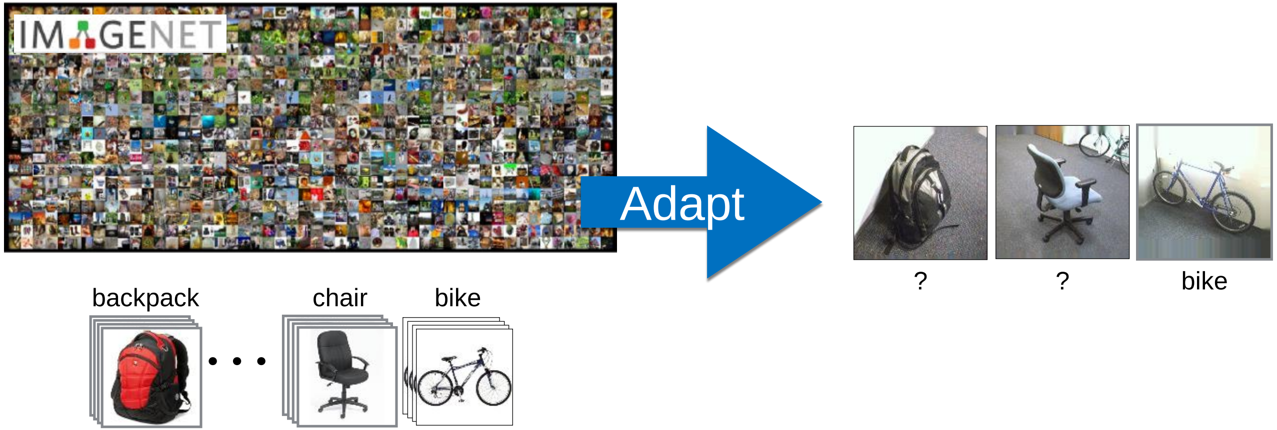


Today

- Show that deep models can be adapted without labels
- Propose two deep adaptation methods:
  - adversarial alignment
  - correlation alignment
- Show applications



Background: Domain Adaptation from source to target distribution



Source Domain $\sim P_S(X, Y)$ $\neq$ Target Domain $\sim P_T(Z, H)$

lots of **labeled** data

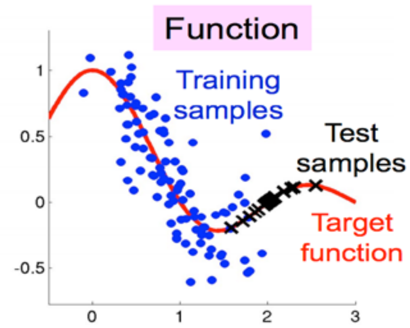
unlabeled or limited labels

$$D_S = \{(\mathbf{x}_i, y_i), \forall i \in \{1, \dots, N\}\}$$

$$D_T = \{(\mathbf{z}_j, ?), \forall j \in \{1, \dots, M\}\}$$

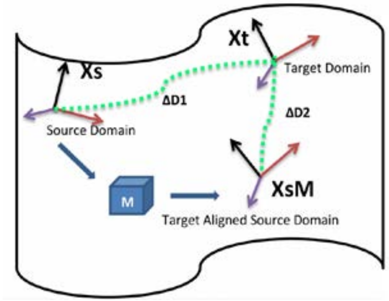
Background: unsupervised domain adaptation

Sample re-weighting

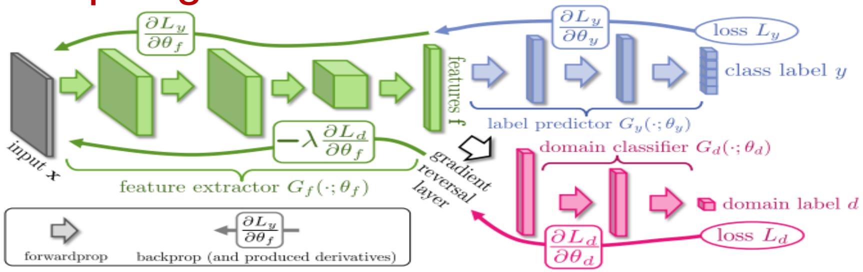


- NO labels in target domain
- Roughly, three categories of methods
  - Sample re-weighting
  - Subspace matching
  - Deep methods

Subspace alignment

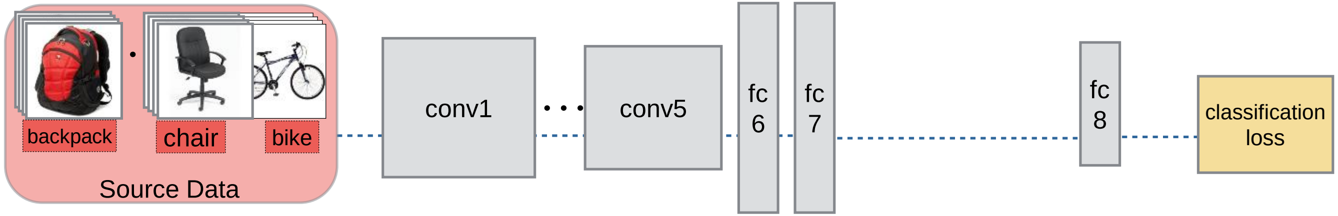


Deep alignment

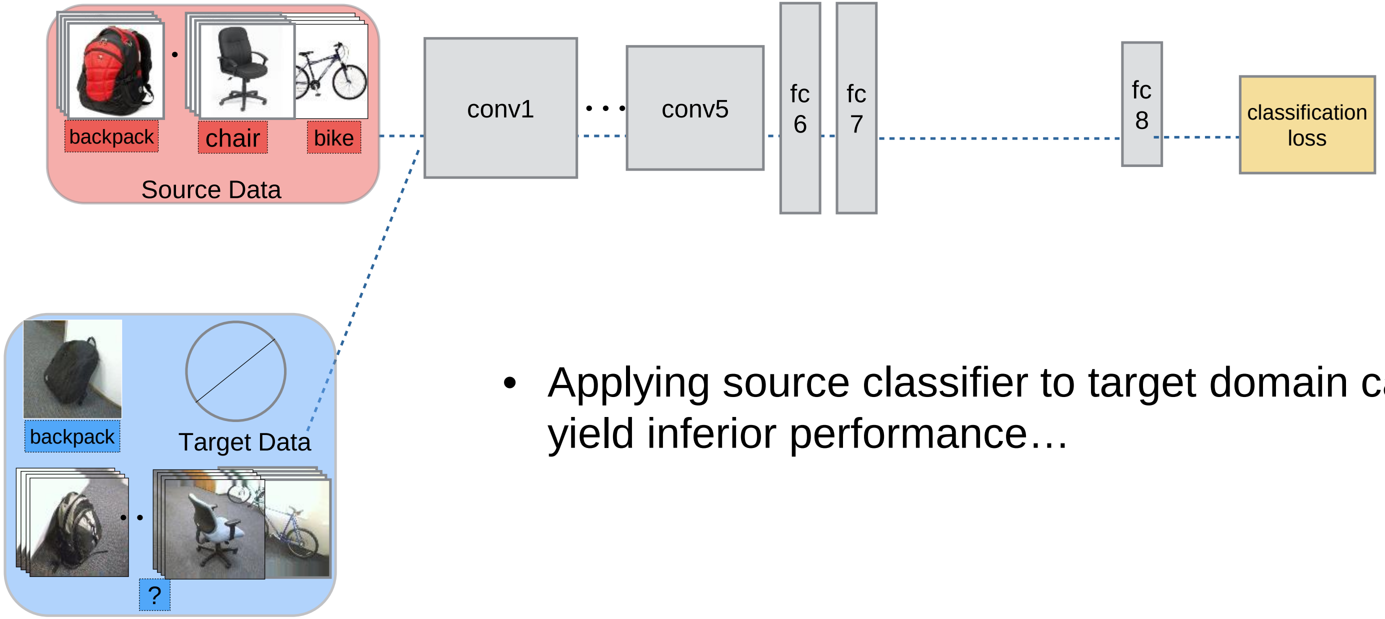


B. Femando, A. Habrard, M. Sebban, and T. Tuytelaars. Unsupervised visual domain adaptation using subspace alignment. In ICCV, 2013.
B. Gong, Y. Shi, F. Sha, and K. Grauman. Geodesic flow kernel for unsupervised domain adaptation. In CVPR, 2012
Y. Ganin and V. Lempitsky. Unsupervised domain adaptation by back propagation. In ICML 2015

How to adapt a deep network?

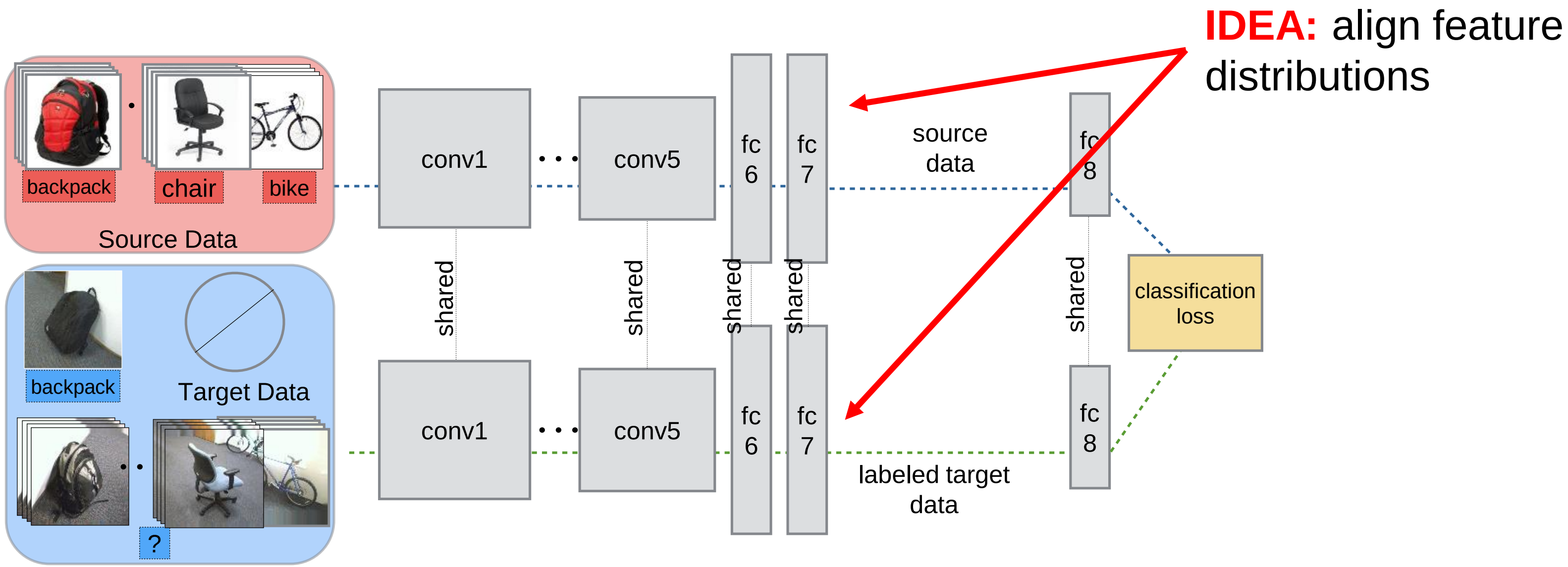


How to adapt a deep network?



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

How to adapt a deep network?

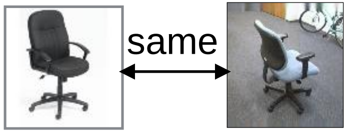


- Fine tune?

  …..Zero or few labels in target domain

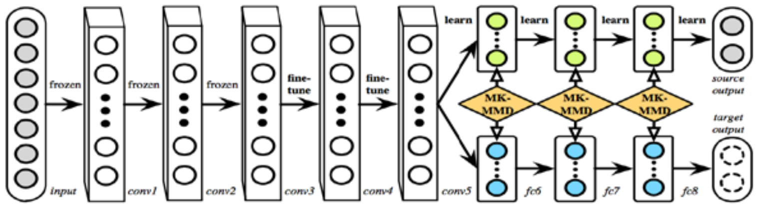
- Siamese network?

  …..No paired / aligned instance examples!

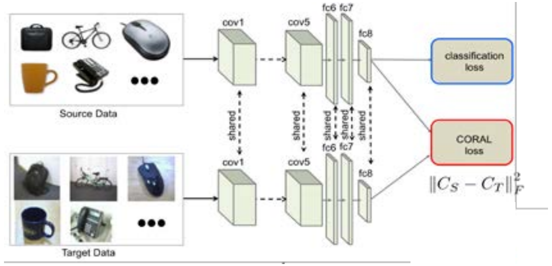


Deep distribution alignment

- by minimizing distance between distributions, e.g.

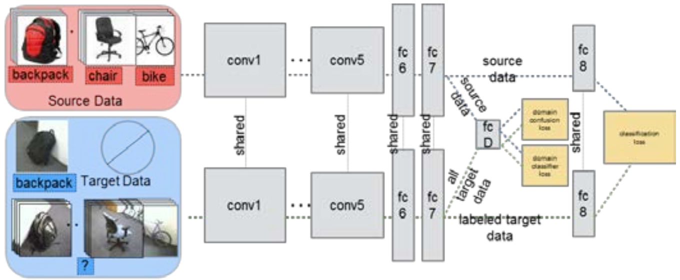


Maximum Mean Discrepancy M. Long, et al. ICML 2015

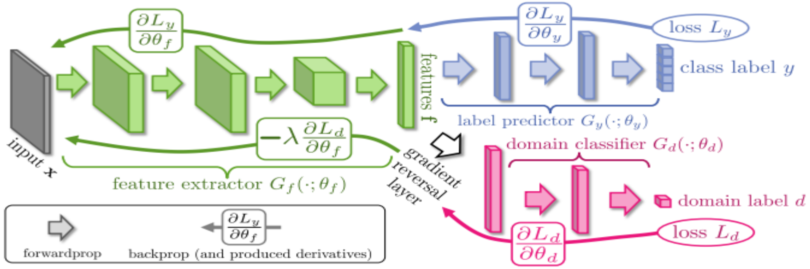


CORrelation ALignment Sun and Saenko, AAAI 2016

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



Domain Confusion E. Tzeng et al. ICCV 2015



Reverse Gradient Y. Ganin and V. Lempitsky ICML 2015

Adversarial Domain Adaptation

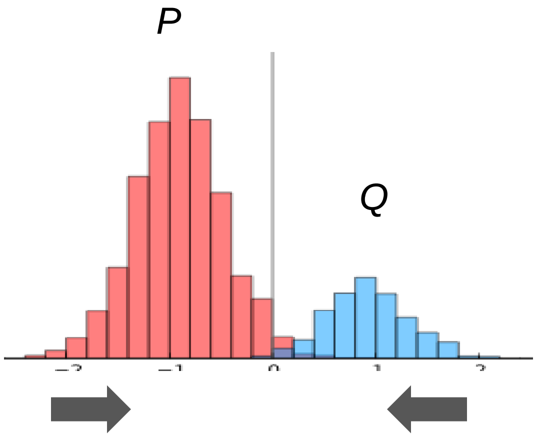
Eric
Tzeng
UC Berkeley

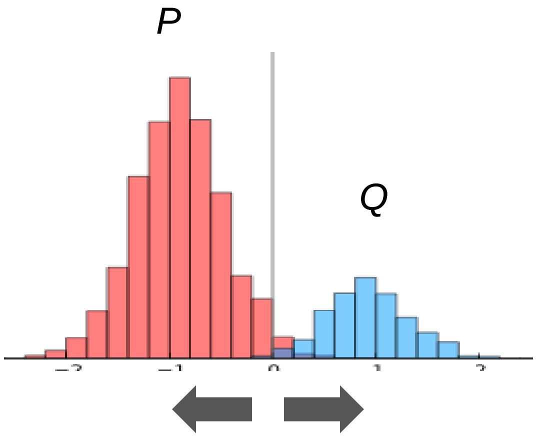
Judy
Hoffman
UC Berkeley

Trevor
Darrell
UC Berkeley

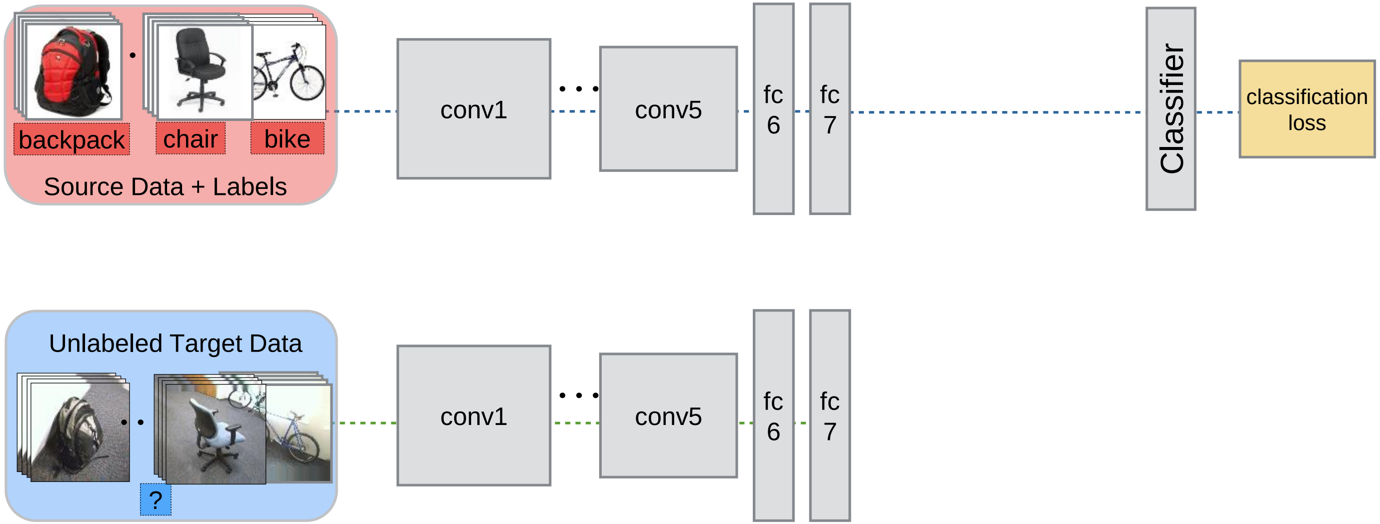
Adversarial networks

Adversarial networks

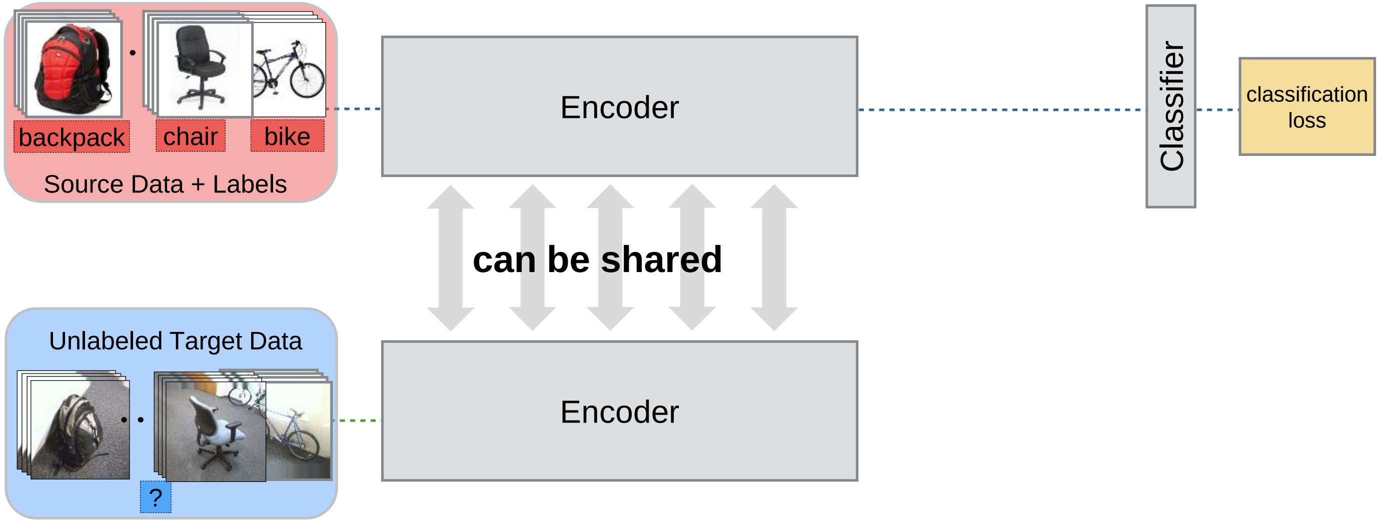




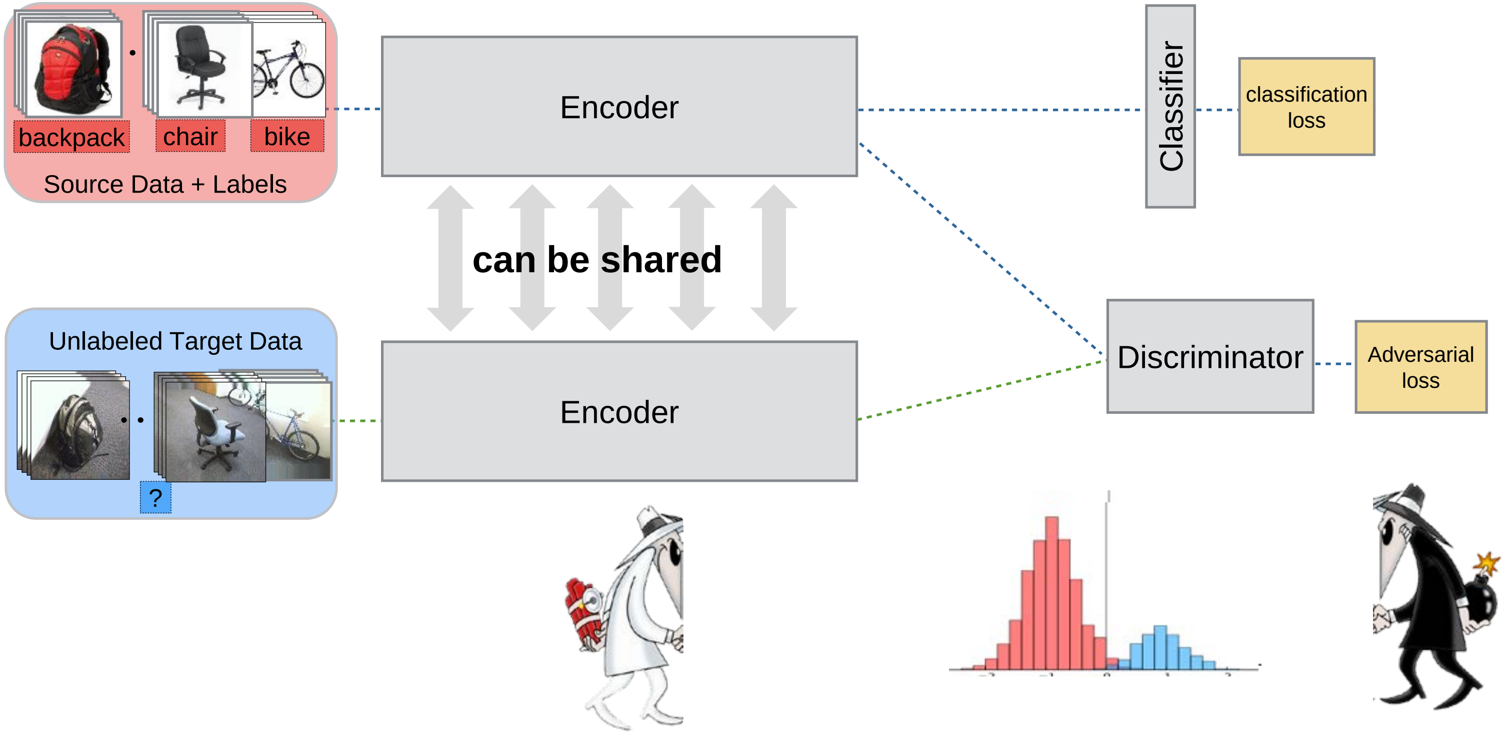
Adversarial domain adaptation



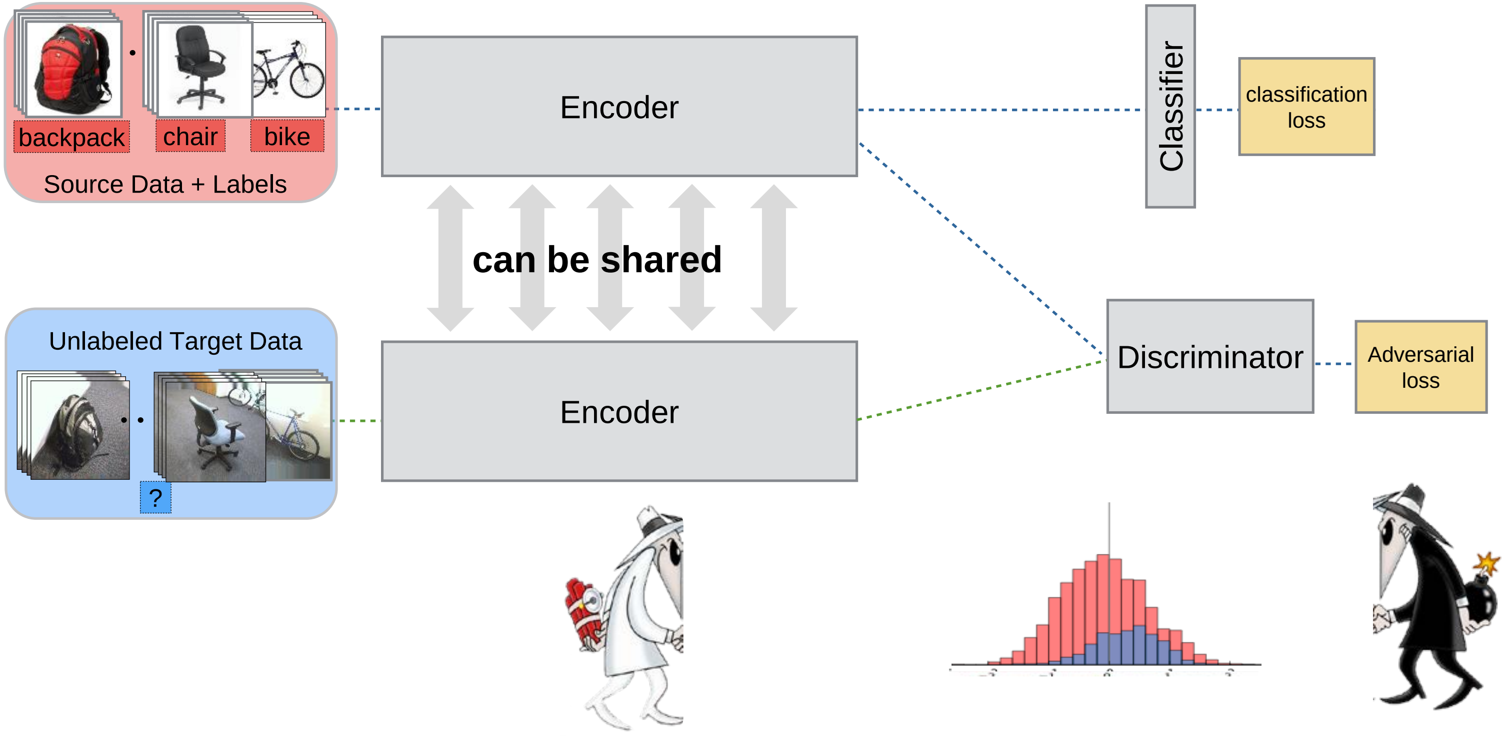
Adversarial domain adaptation



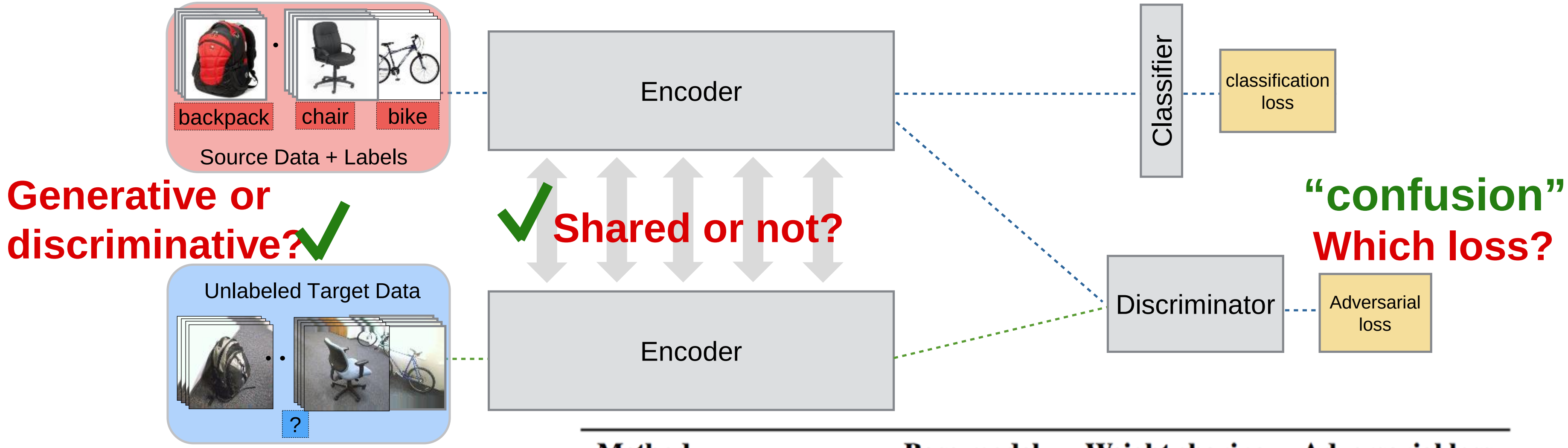
Adversarial domain adaptation



Adversarial domain adaptation



Design choices in adversarial adaptation

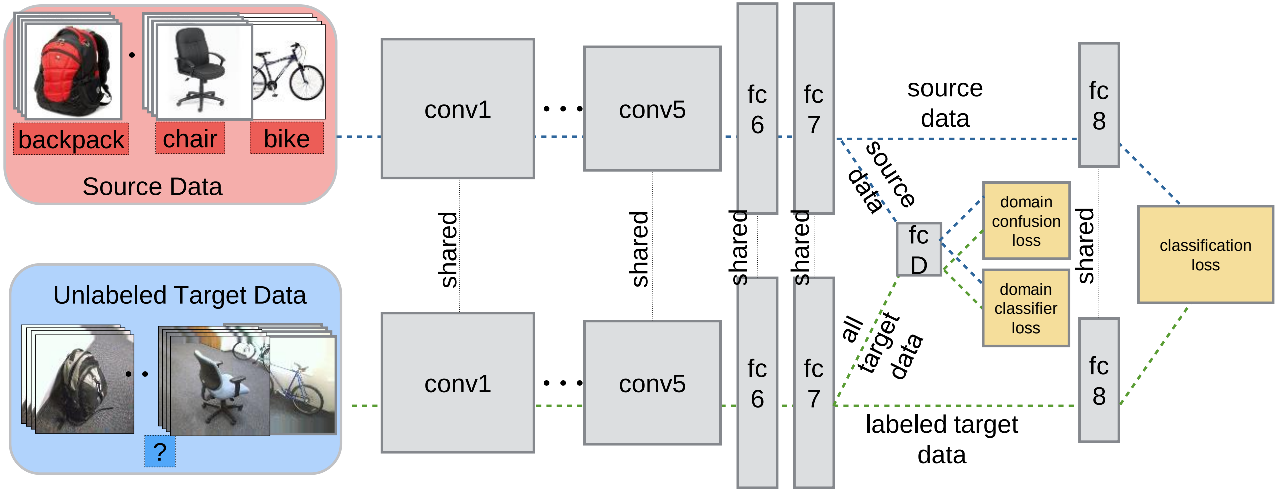


Method	Base model	Weight sharing	Adversarial loss
Gradient reversal [16]	discriminative	shared	minimax
Domain confusion [12]	discriminative	shared	confusion
CoGAN [13]	generative	unshared	GAN

Ming-Yu Liu and Oncel Tuzel. Coupled generative adversarial networks, NIPS 2016

Deep domain confusion

[Tzeng ICCV15]



*Adversarial Training of domain label predictor and **domain confusion** loss:*

$$\min_{\theta_D} \mathcal{L}_D(x_S, x_T, \theta_{\text{repr}}; \theta_D)$$

$$\min_{\theta_{\text{repr}}} \mathcal{L}_{\text{conf}}(x_S, x_T, \theta_D; \theta_{\text{repr}}).$$

$$\mathcal{L}_D(x_S, x_T, \theta_{\text{repr}}; \theta_D) = -\sum_d \mathbb{1}[y_D = d] \log q_d$$

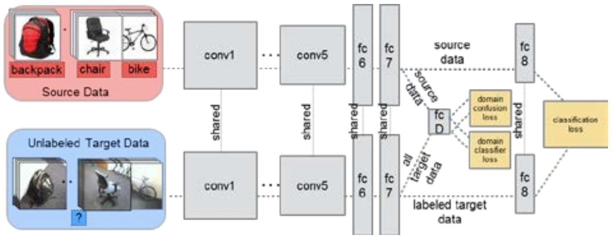
$$\mathcal{L}_{\text{conf}}(x_S, x_T, \theta_D; \theta_{\text{repr}}) = -\sum_d \frac{1}{D} \log q_d.$$

Domain Label Cross-entropy with uniform distribution

Deep domain confusion

[Tzeng ICCV15]

Train a network to minimize classification loss AND confuse two domains



source inputs | target inputs | network parameters (fixed) | domain classifier (learn) | domain classifier loss

$$\mathcal{L}_D(x_S, x_T, \theta_{\text{repr}}; \theta_D) = -\sum_d \mathbb{1}[y_D = d] \log q_d$$

domain classifier prediction

$$q = \text{softmax}(\theta_D^T f(x; \theta_{\text{repr}})) = p(y_D = 1|x)$$

iterate

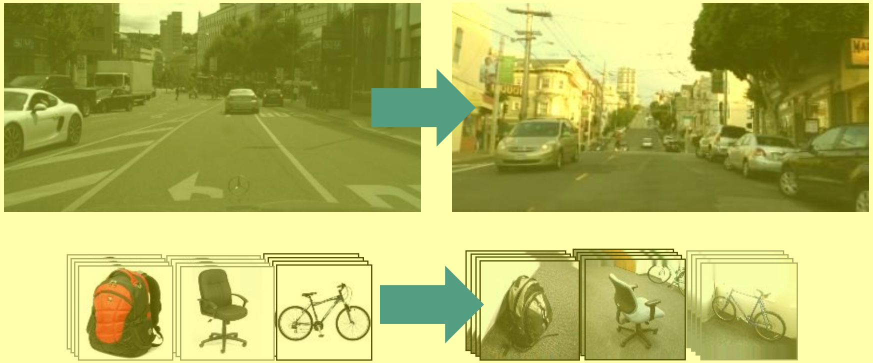
domain classifier (fixed) | network parameters (learn) | domain confusion loss

$$\mathcal{L}_{\text{conf}}(x_S, x_T, \theta_D; \theta_{\text{repr}}) = -\sum_d \frac{1}{D} \log q_d$$

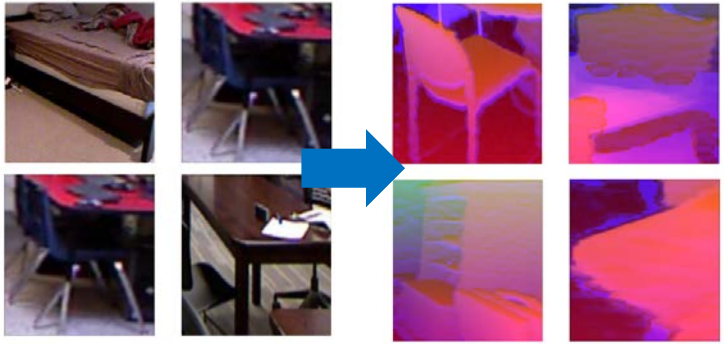
(cross-entropy with uniform distribution)

Applications to different types of domain shift

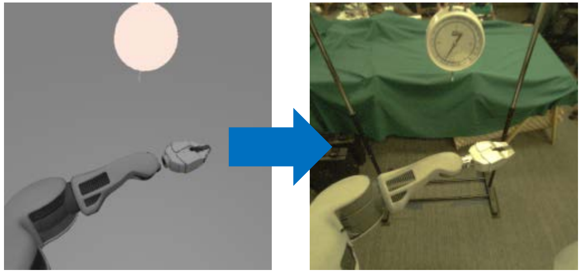
From dataset to dataset



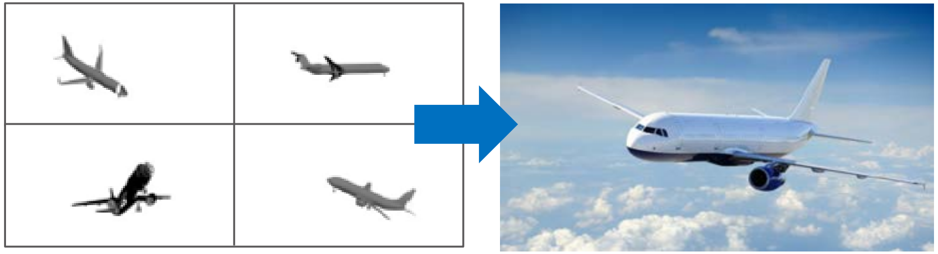
From RGB to depth



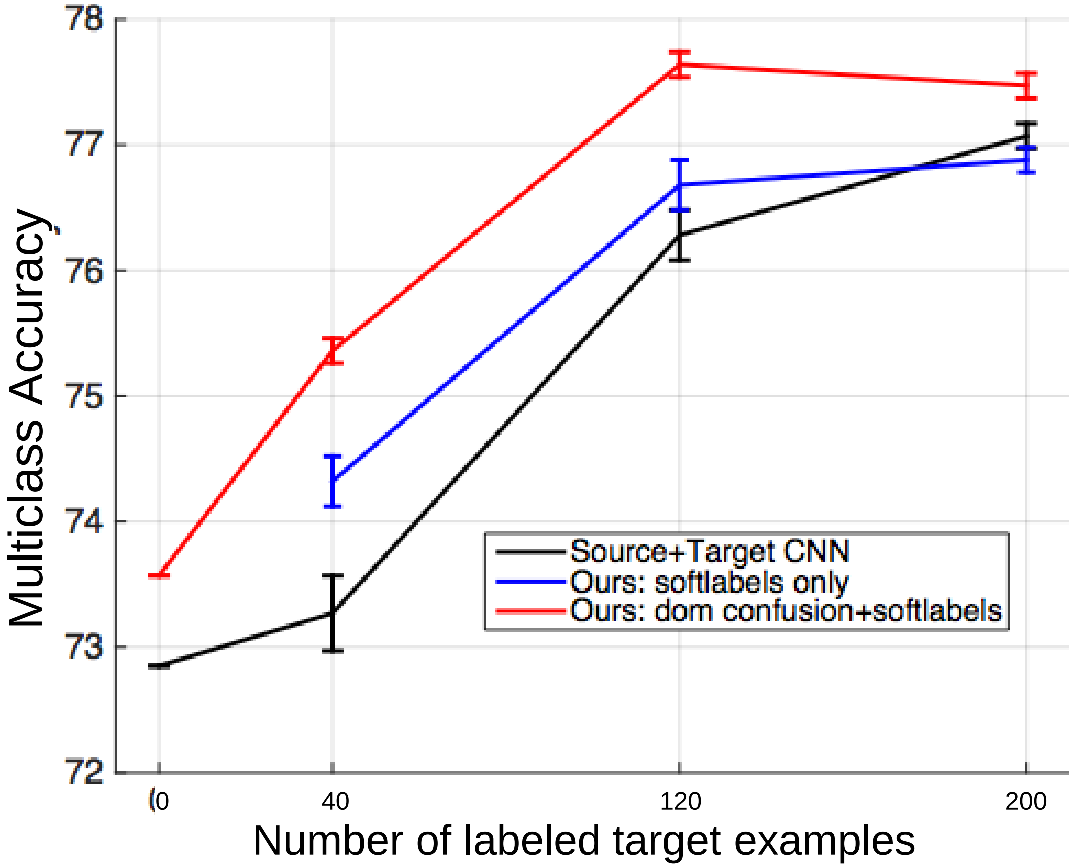
From simulated to real control



From CAD models to real images



ImageNet adapted to Caltech [Tzeng ICCV15]



Results on Cityscapes to SF adaptation

Before domain confusion

After domain confusion

FCNs in the Wild: Pixel-level Adversarial and Constraint-based Adaptation, Judy Hoffman, Dequan Wang, Fisher Yu, Trevor Darrell, Arxiv 2016

Adversarial Loss Functions

Confusion loss [Tzeng 2015]

$$\max_{D} \mathbb{E}_{\mathbf{x}\sim p_S(\mathbf{x})}\left[\log D(M_S(\mathbf{x}))\right] + \mathbb{E}_{\mathbf{x}\sim p_T(\mathbf{x})}\left[\log(1 - D(M_T(\mathbf{x})))\right]$$

$$\max_{M_S, M_T} \sum_{d\in\{S,T\}} \mathbb{E}_{\mathbf{x}\sim p_d(\mathbf{x})}\left[\frac{1}{2}\log D(M_d(\mathbf{x})) + \frac{1}{2}\log(1 - D(M_d(\mathbf{x})))\right]$$

Minimax loss [Ganin 2015]

$$\min_{M_S, M_T} \max_{D} V(D, M_S, M_T) = \mathbb{E}_{\mathbf{x}\sim p_S(\mathbf{x})}[\log D(M_S(\mathbf{x}))] + \mathbb{E}_{\mathbf{x}\sim p_T(\mathbf{x})}[\log(1 - D(M_T(\mathbf{x})))]$$

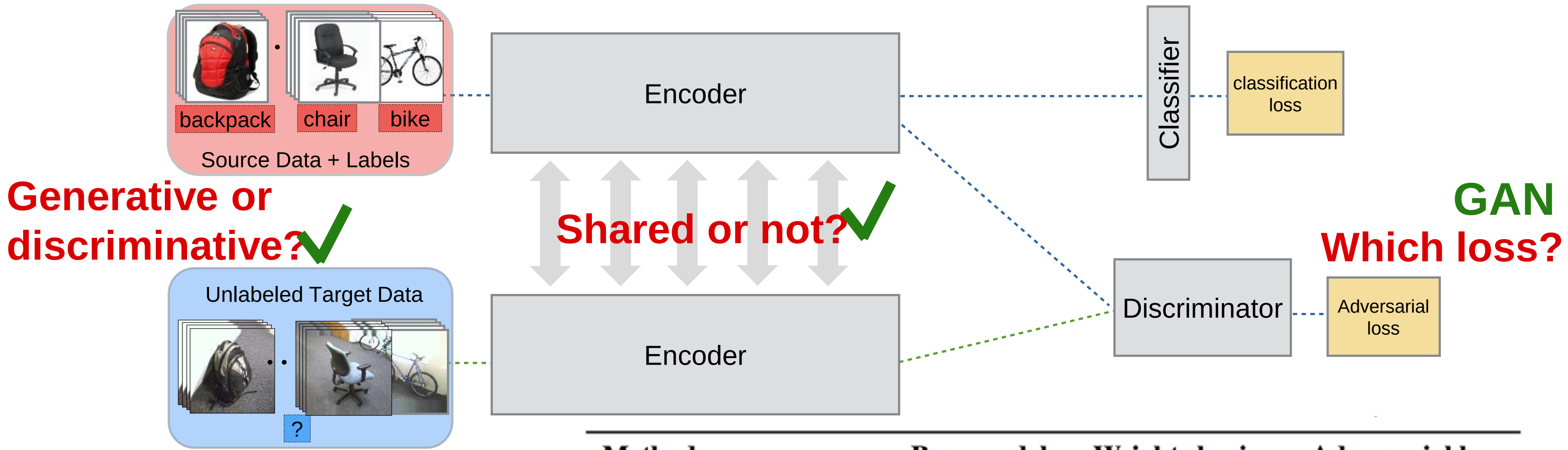
GAN loss [Goodfellow 2014]

$$\max_{D} \mathbb{E}_{\mathbf{x}\sim p_S(\mathbf{x})}[\log D(M_S(\mathbf{x}))] + \mathbb{E}_{\mathbf{x}\sim p_T(\mathbf{x})}[\log(1 - D(M_T(\mathbf{x})))]$$

"stronger gradients"

$$\max_{M_T} \mathbb{E}_{\mathbf{x}\sim p_T(\mathbf{x})}[\log D(M_T(\mathbf{x}))].$$

Adversarial Discriminative Domain Adaptation (ADDA) (in submission)



Method	Base model	Weight sharing	Adversarial loss
Gradient reversal [16]	discriminative	shared	minimax
Domain confusion [12]	discriminative	shared	confusion
CoGAN [13]	generative	unshared	GAN
ADDA	discriminative	unshared	GAN

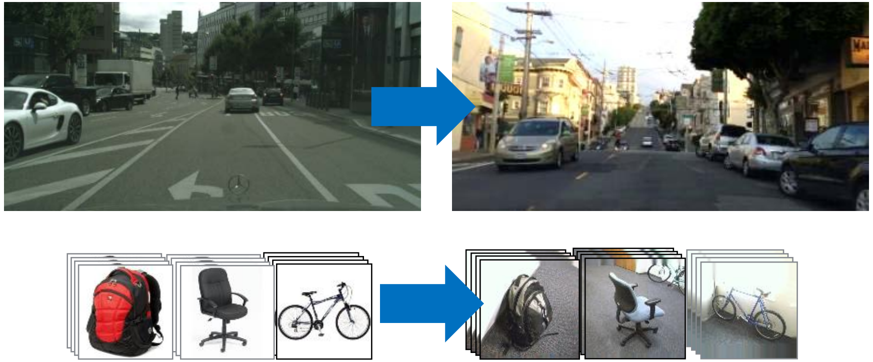
ADDA: Adaptation on digits

(in submission)

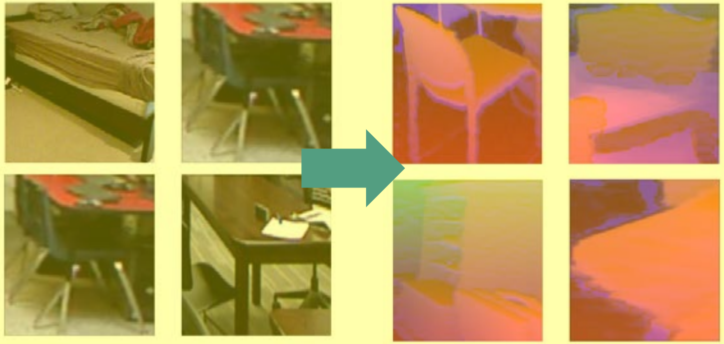
Method	MNIST → USPS	USPS → MNIST	SVHN → MNIST
Source only	0.752 ± 0.016	0.571 ± 0.017	0.601 ± 0.011
Gradient reversal	0.771 ± 0.018	0.730 ± 0.020	0.739 [16]
Domain confusion	0.791 ± 0.005	0.665 ± 0.033	0.681 ± 0.003
CoGAN	0.912 ± 0.008	0.891 ± 0.008	did not converge
ADDA (Ours)	0.894 ± 0.002	0.901 ±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

Applications to different types of domain shift

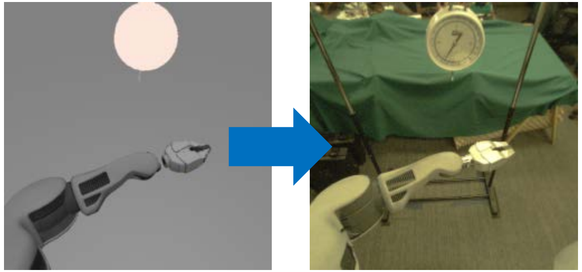
From dataset to dataset



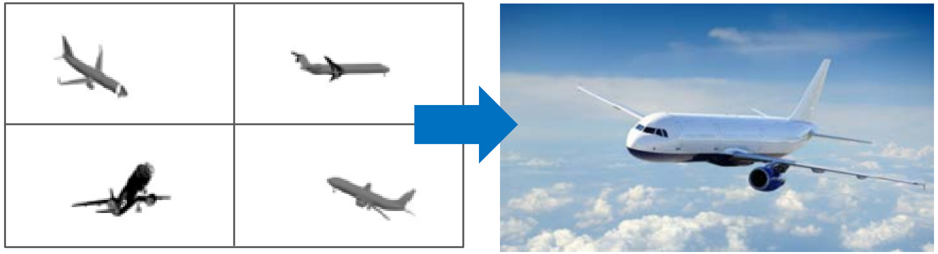
From RGB to depth



From simulated to real control



From CAD models to real images

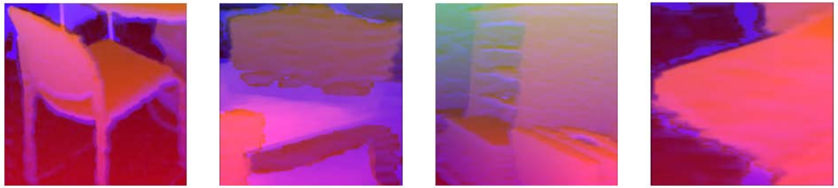


ADDA: Adaptation on RGB-D

(in submission)

Train on RGB

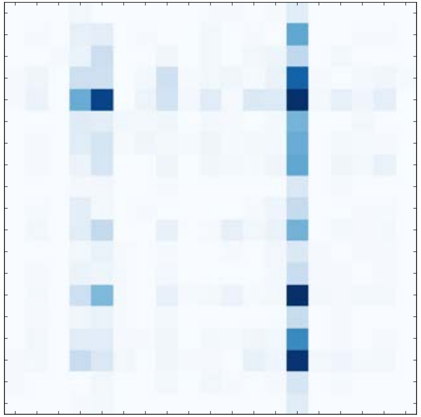
Test on depth



	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

ADDA: Adaptation on RGB-D

(in submission)



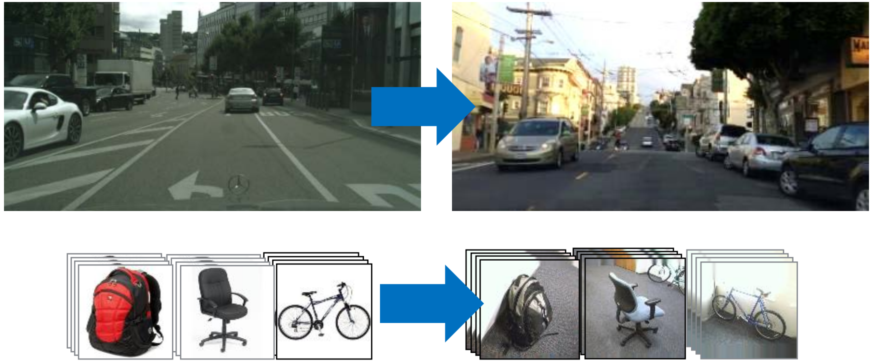
Train on target

True label

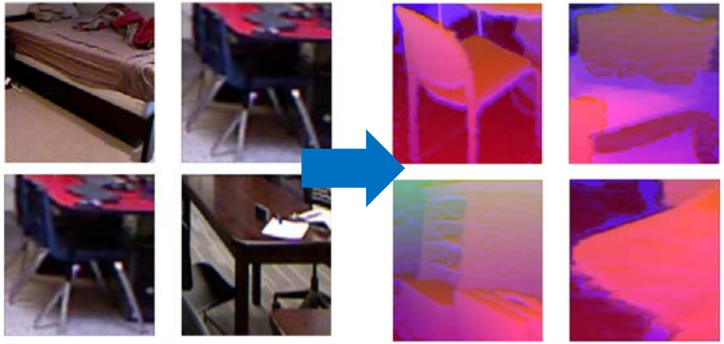
stand
pillow
sink
sofa
table
television
toilet

Applications to different types of domain shift

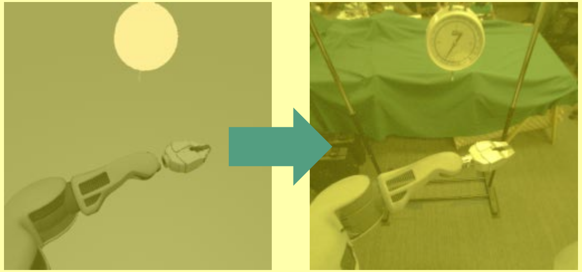
From dataset to dataset



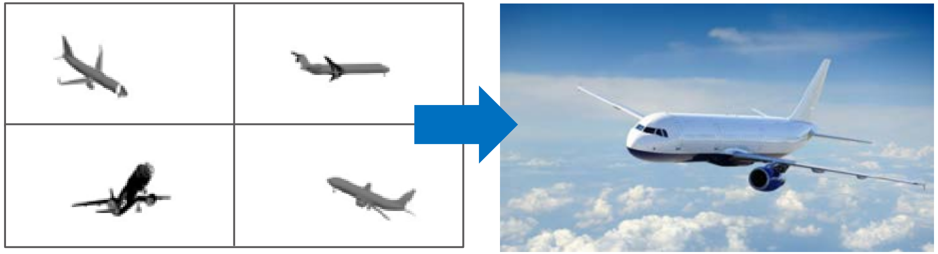
From RGB to depth



From simulated to real control



From CAD models to real images



Adapting Deep Visuomotor Representations with Weak Pairwise Constraints

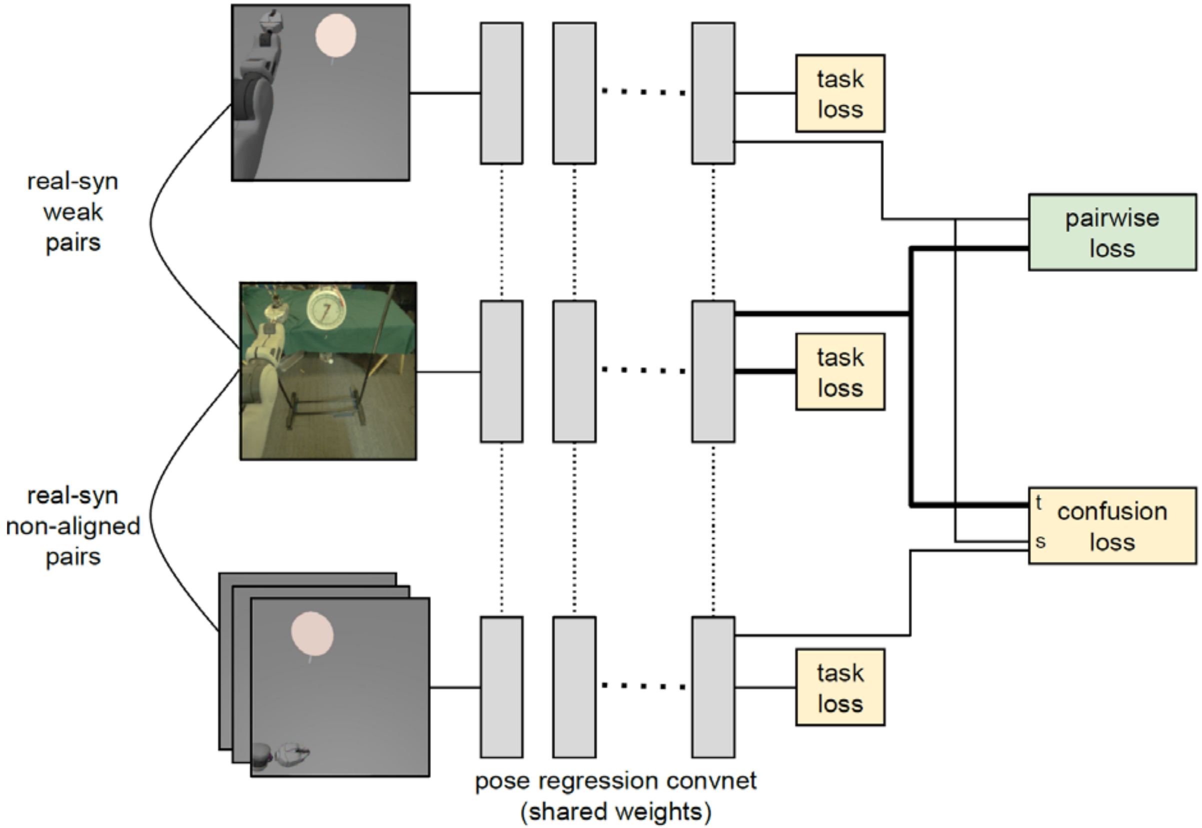
Eric Tzeng[1], Coline Devin[1], Judy Hoffman[1], Chelsea Finn[1],
Pieter Abbeel[1], Sergey Levine[1], Kate Saenko[2], Trevor Darrell[1]

[1] University of California, Berkeley
[2] Boston University

From simulation to real world control [Tzeng, Devin, et al 16]

Weak pairwise constraints

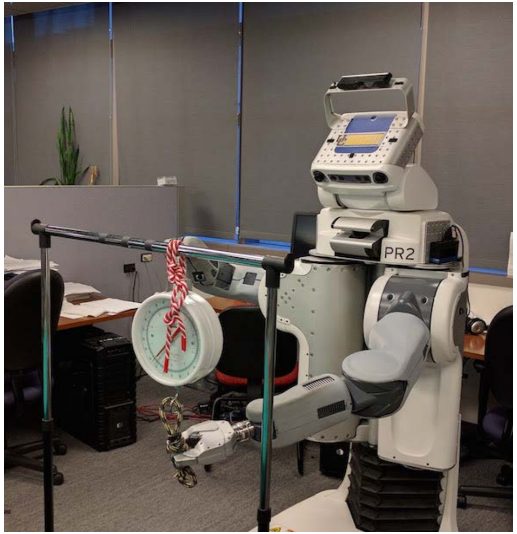
[Tzeng, Devin, et al 16]



Robotic task: place rope on scale

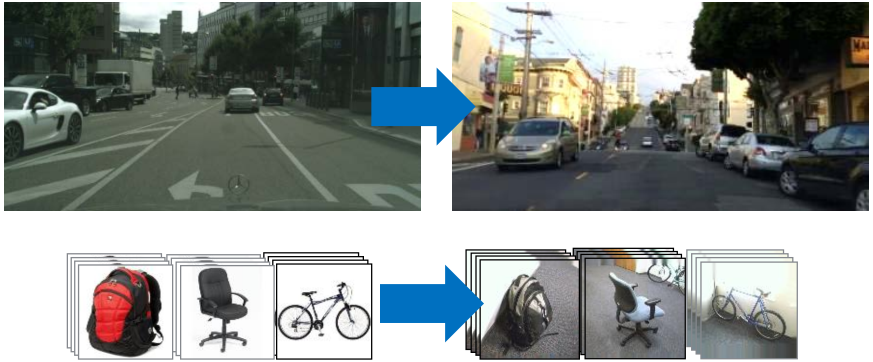
[Tzeng, Devin, et al 16]

Method	# Sim	# Real (unlabeled)	Success rate
Synthetic only	4000	0	38.1% ± 8%
Autoencoder (100)	0	100	28.6% ± 25%
Autoencoder (500)	0	500	33.2% ±15%
Domain alignment with randomly assigned pairs	4000	100	33.3% ±16%
Domain alignment with weakly supervised pairwise constraints	4000	100	**76.2% ± 16%**
Oracle	0	500 (labeled)	71.4% ± 14%

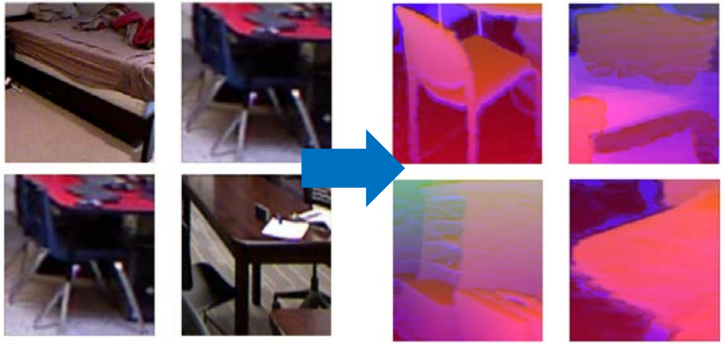


Applications to different types of domain shift

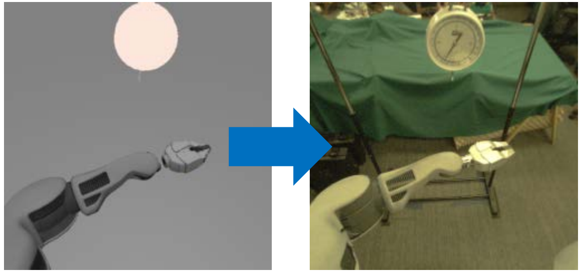
From dataset to dataset



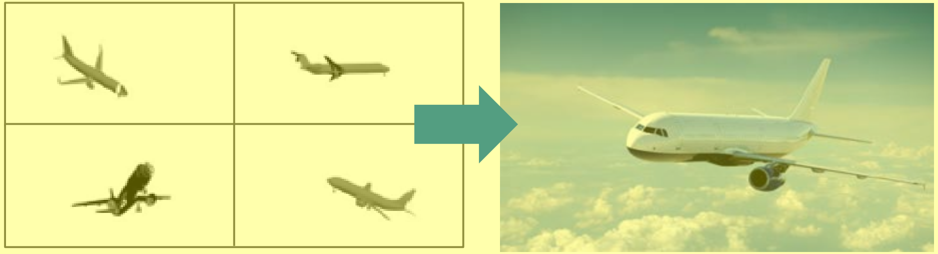
From RGB to depth



From simulated to real control



From CAD models to real images



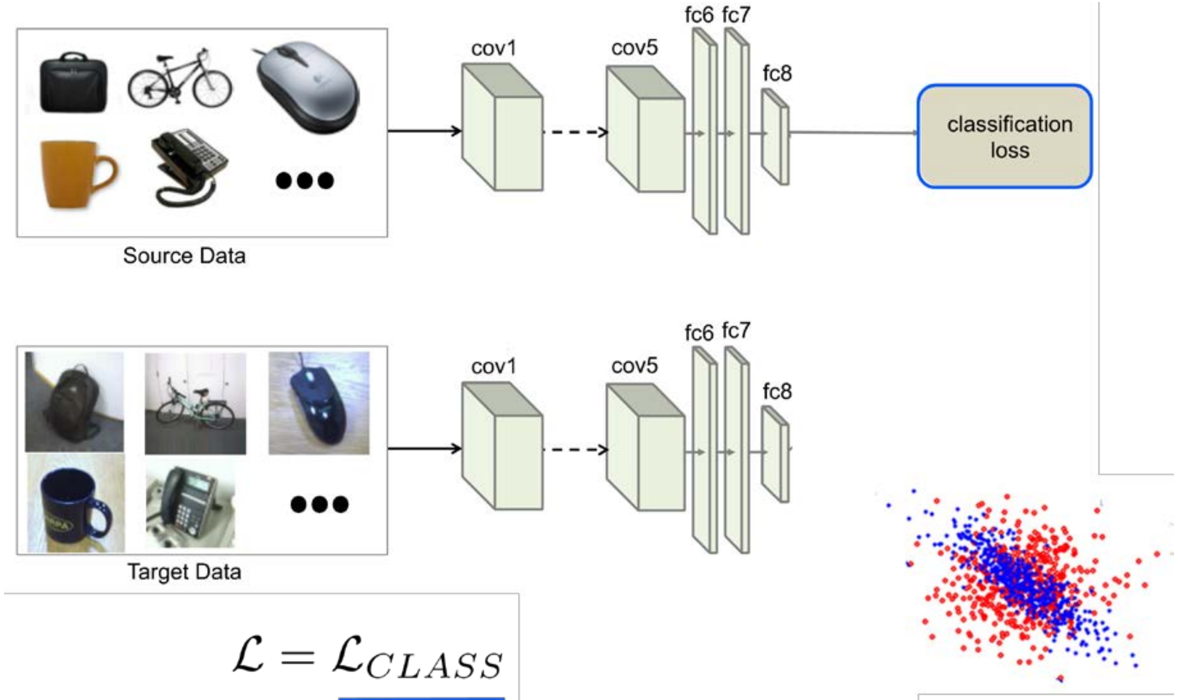
Domain Adaptation via Correlation Alignment

Baochen Sun
Microsoft

Xingchao Peng
Boston University

Deep CORAL: Correlation Alignment for Deep Domain Adaptation

[Sun 2016]

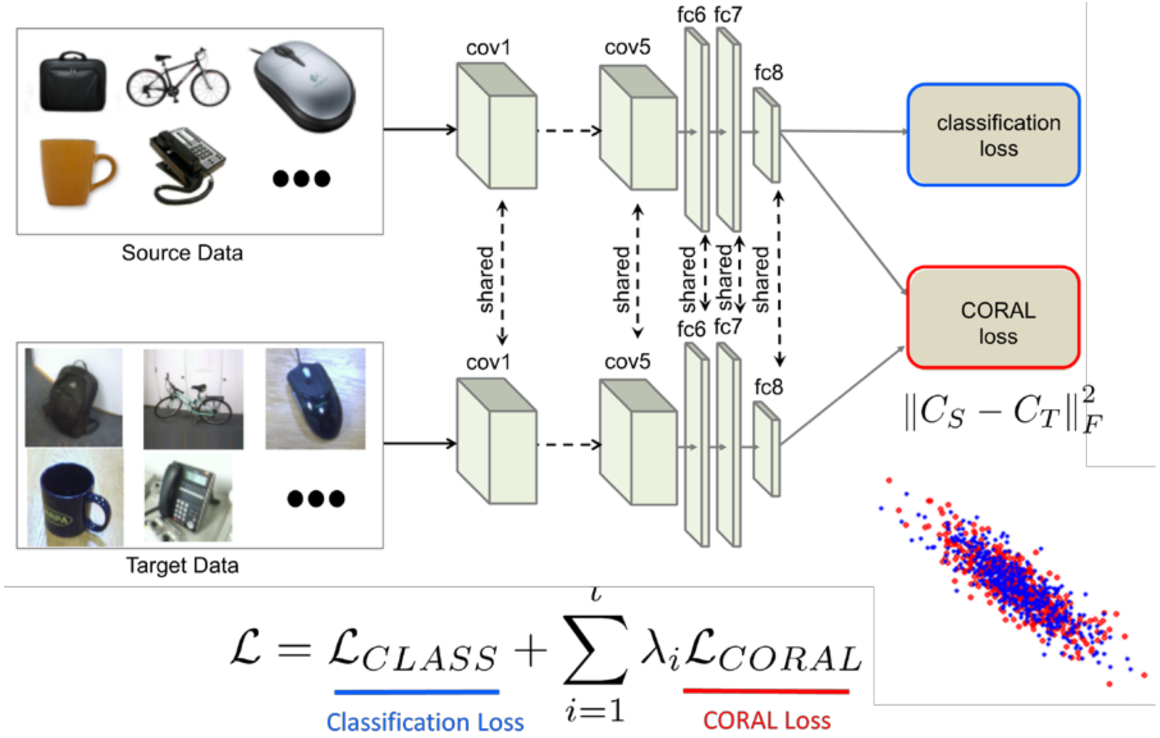


$$\mathcal{L} = \underbrace{\mathcal{L}_{CLASS}}$$

Classification Loss

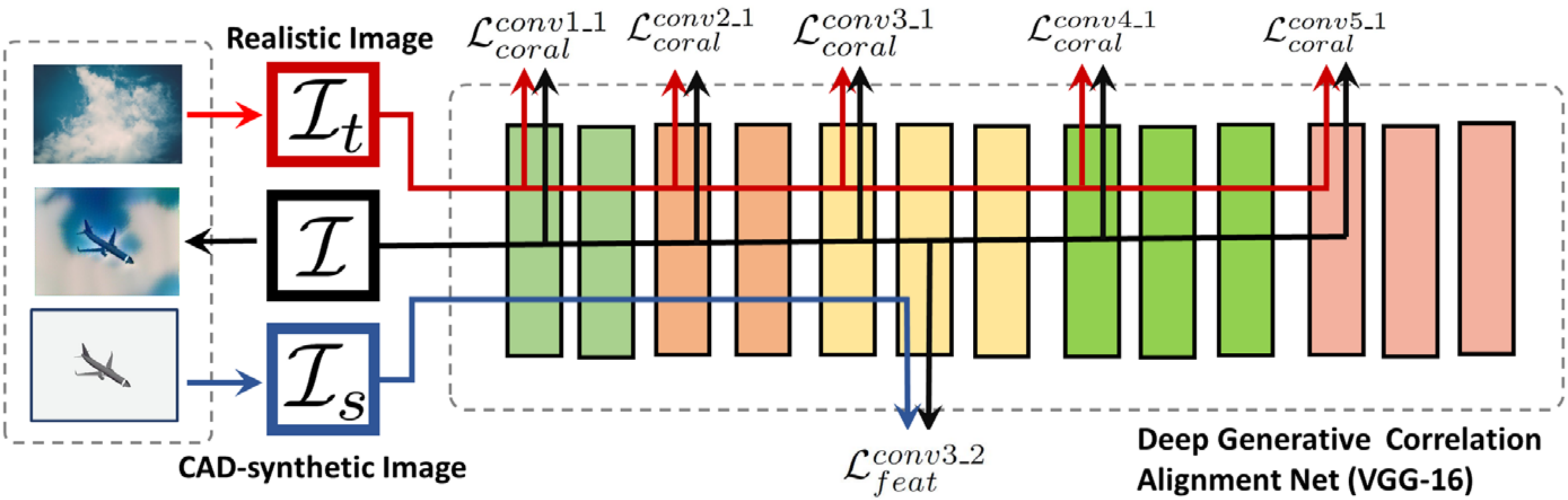
Deep CORAL: Correlation Alignment for Deep Domain Adaptation

[Sun 2016]



Generative CORAL Network

(in submission)

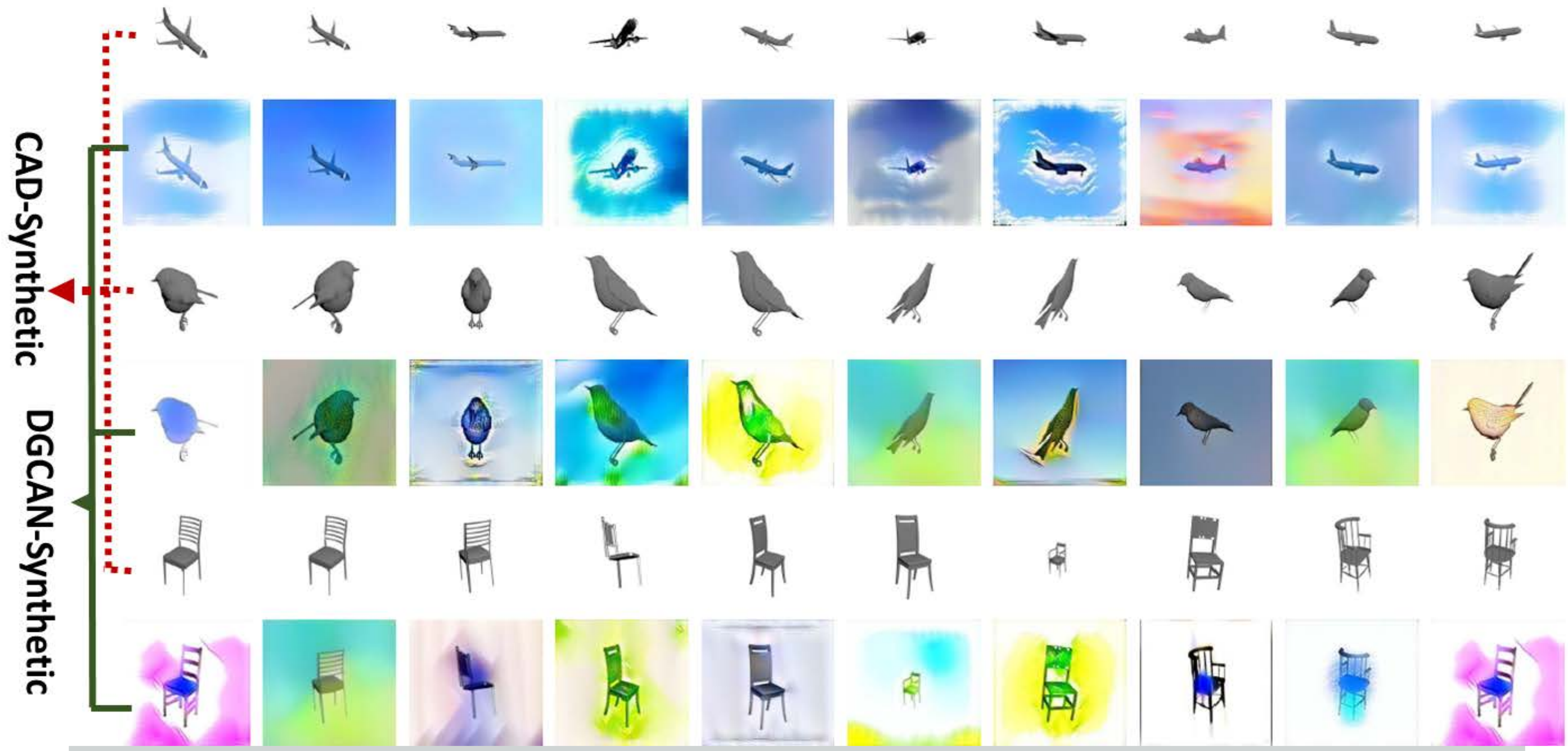


Synthetic to real adaptation for object recognition (in submission)

Train on synthetic

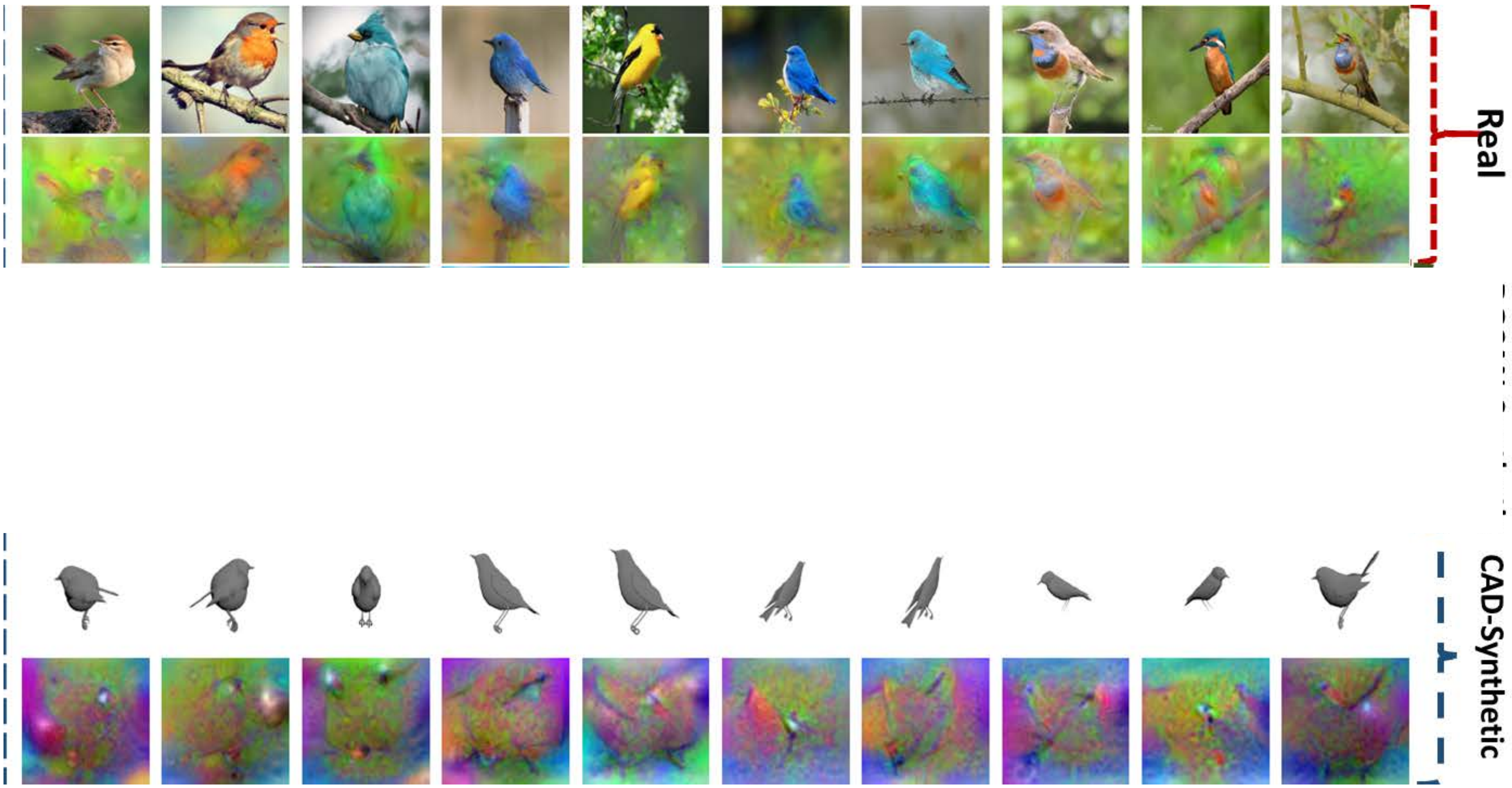
Test on real

Synthetic to Real Adaptation with Deep Generative Correlation Alignment Networks (in submission)



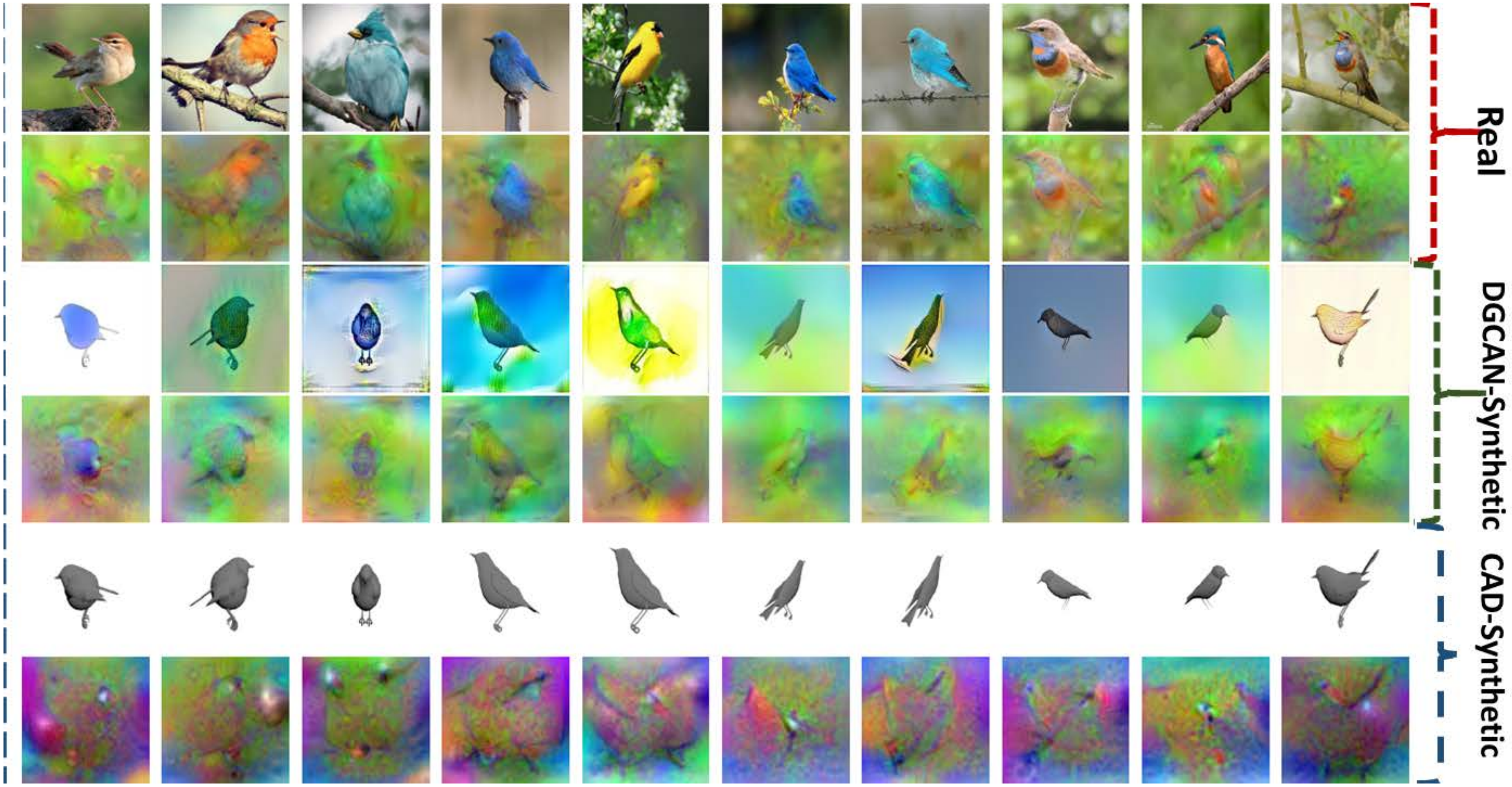
Synthetic to Real Adaptation with Deep Generative Correlation Alignment Networks

(in submission)



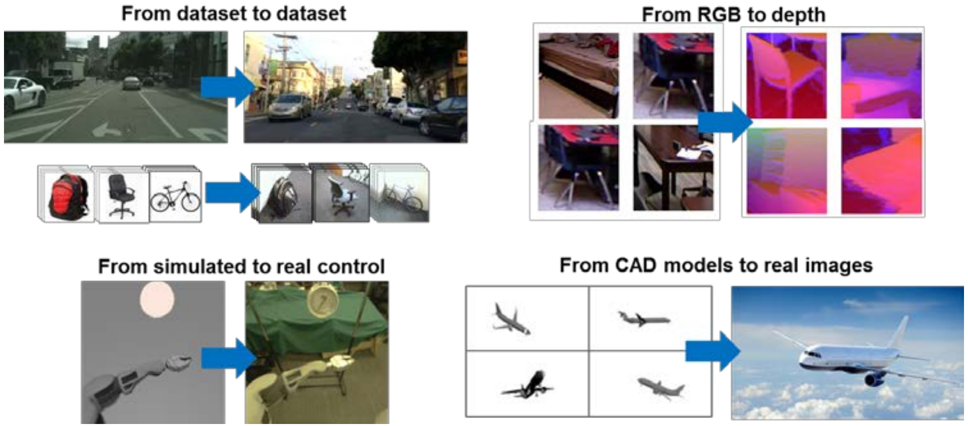
Synthetic to Real Adaptation with Deep Generative Correlation Alignment Networks

(in submission)



Summary

- Deep models can be adapted to new domains without labels
- Proposed two deep feature alignment methods:
  - adversarial alignment
  - correlation alignment
- Many potential applications



Thank you

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
- Adversarial Discriminative Domain Adaptation, in submission
- Synthetic to Real Adaptation with Deep Generative Correlation Alignment Networks, in submission