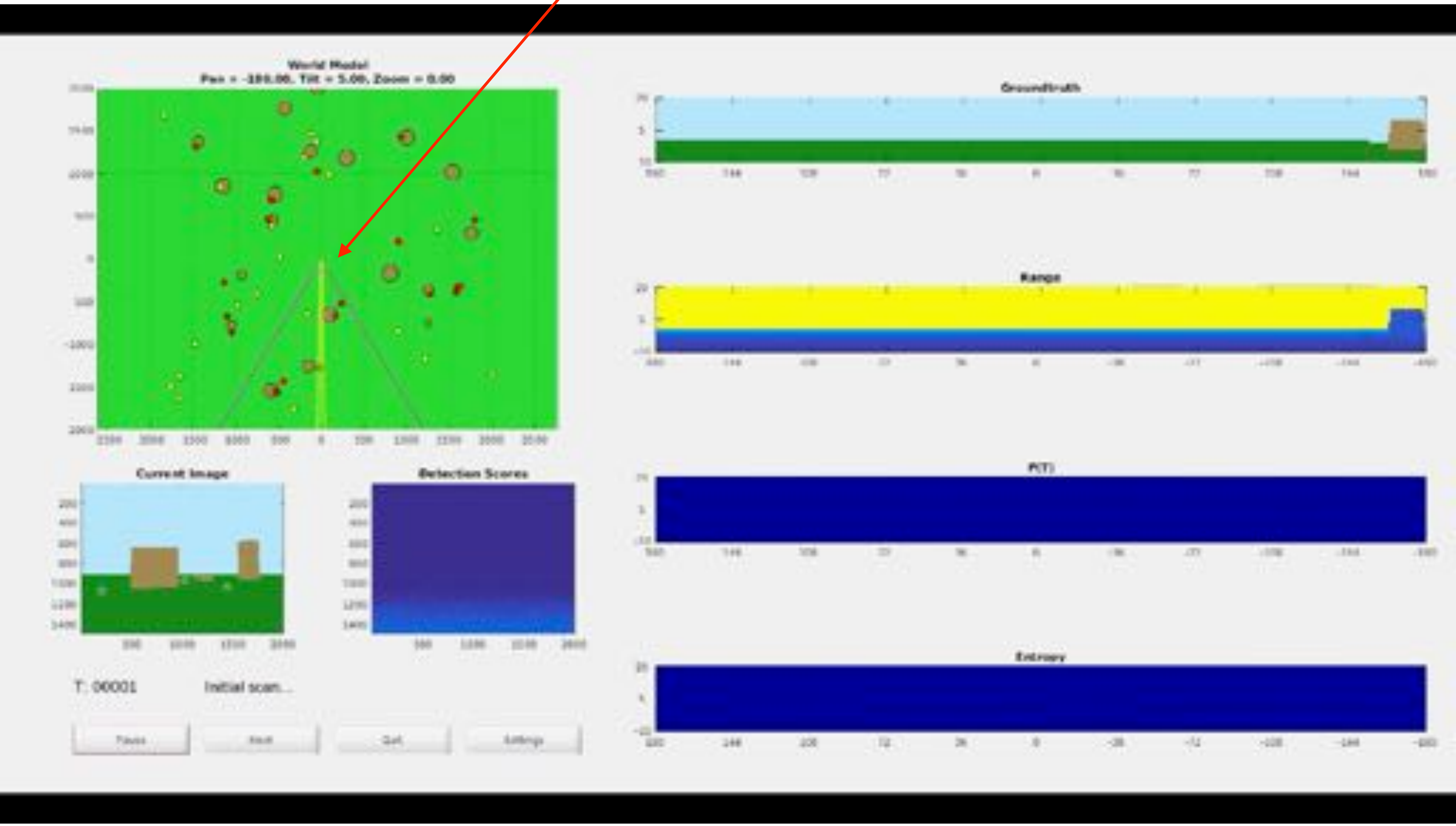
Domain Adaptation for *Robust Visual Recognition*

Boqing Gong
bgong@crcv.ucf.edu

UCF Center for Research in Computer Vision

An intelligent robot



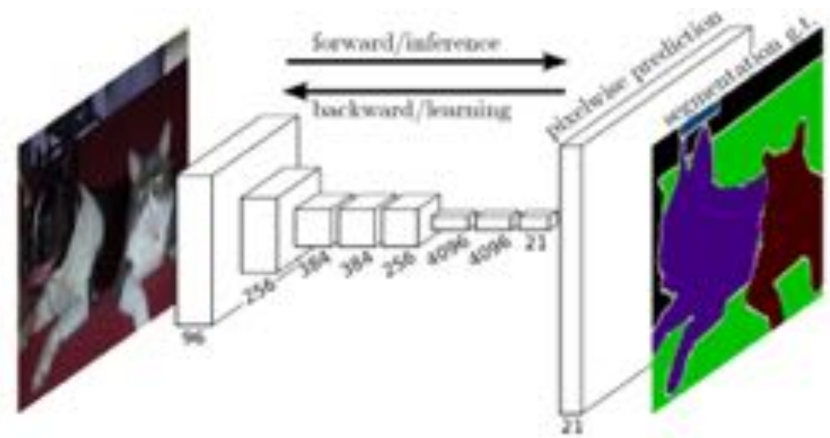
Semantic segmentation of urban scenes

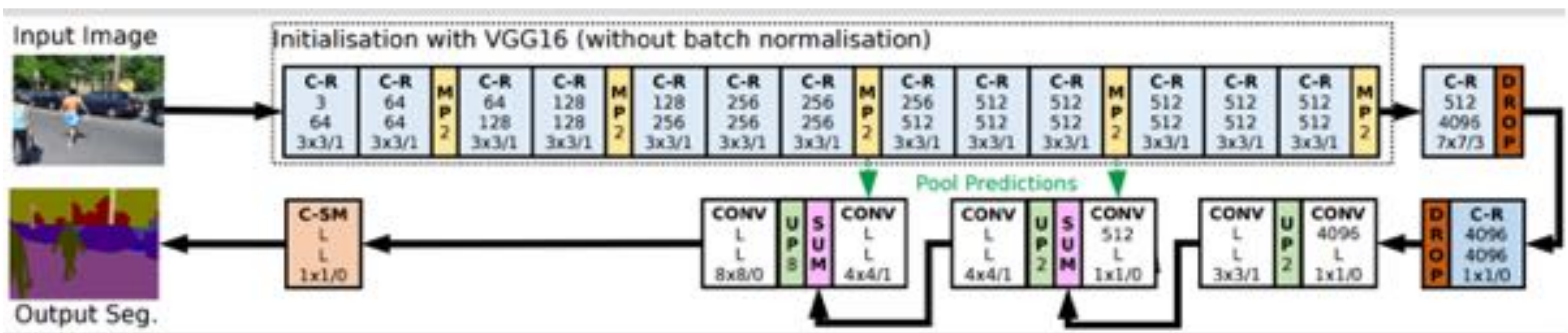
Assign each pixel a semantic label

An appealing application: **self-driving**

Image credit: https://www.cityscapes-dataset.com/

Triumphal approach: CNNs
convolutional neural networks





Long, J., Shelhamer, E., & Darrell, T. (2015). Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*.

To teach/train CNNs to segment images and videos

About 1.5 hrs to label one such image!

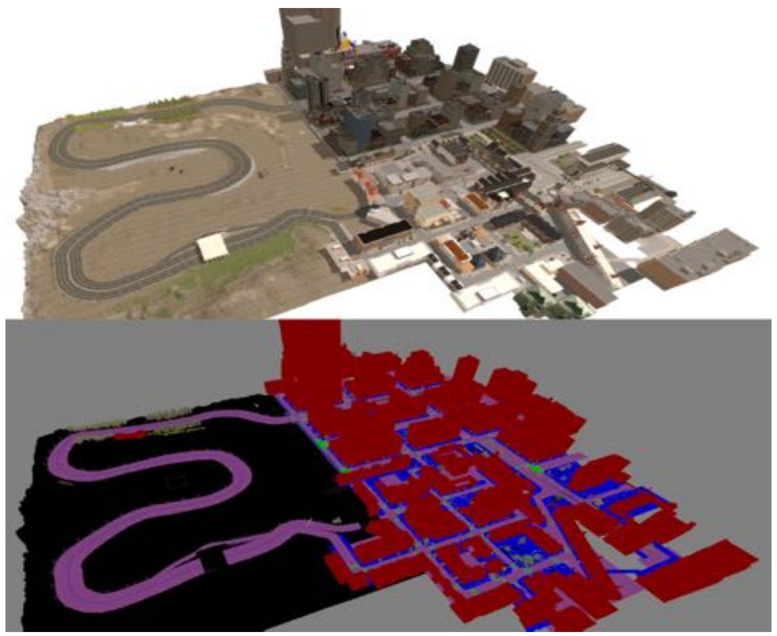
Cityscapes: largest publicly available dataset

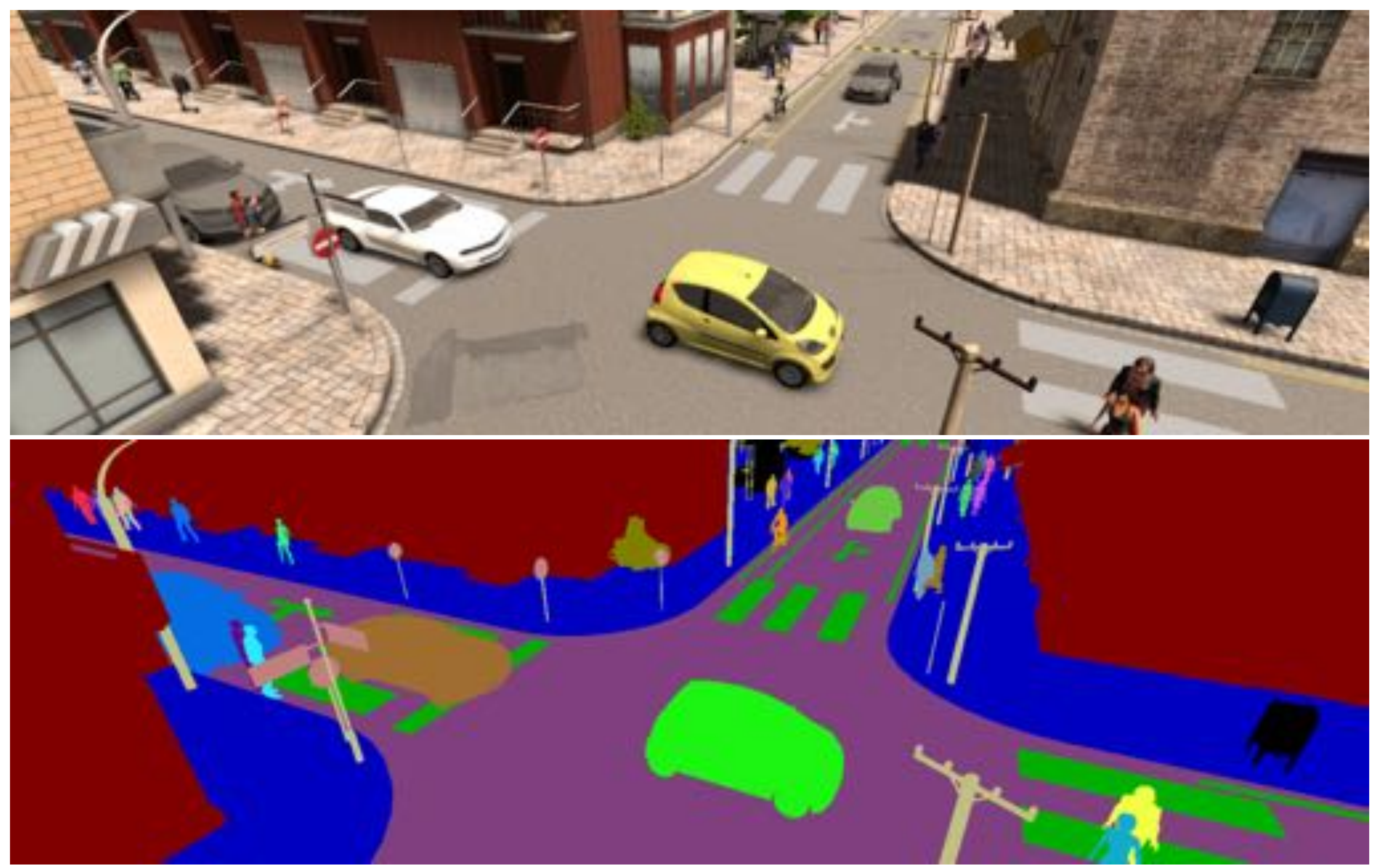
30k images captured from 50 cities

Only 5k are well labeled thus far

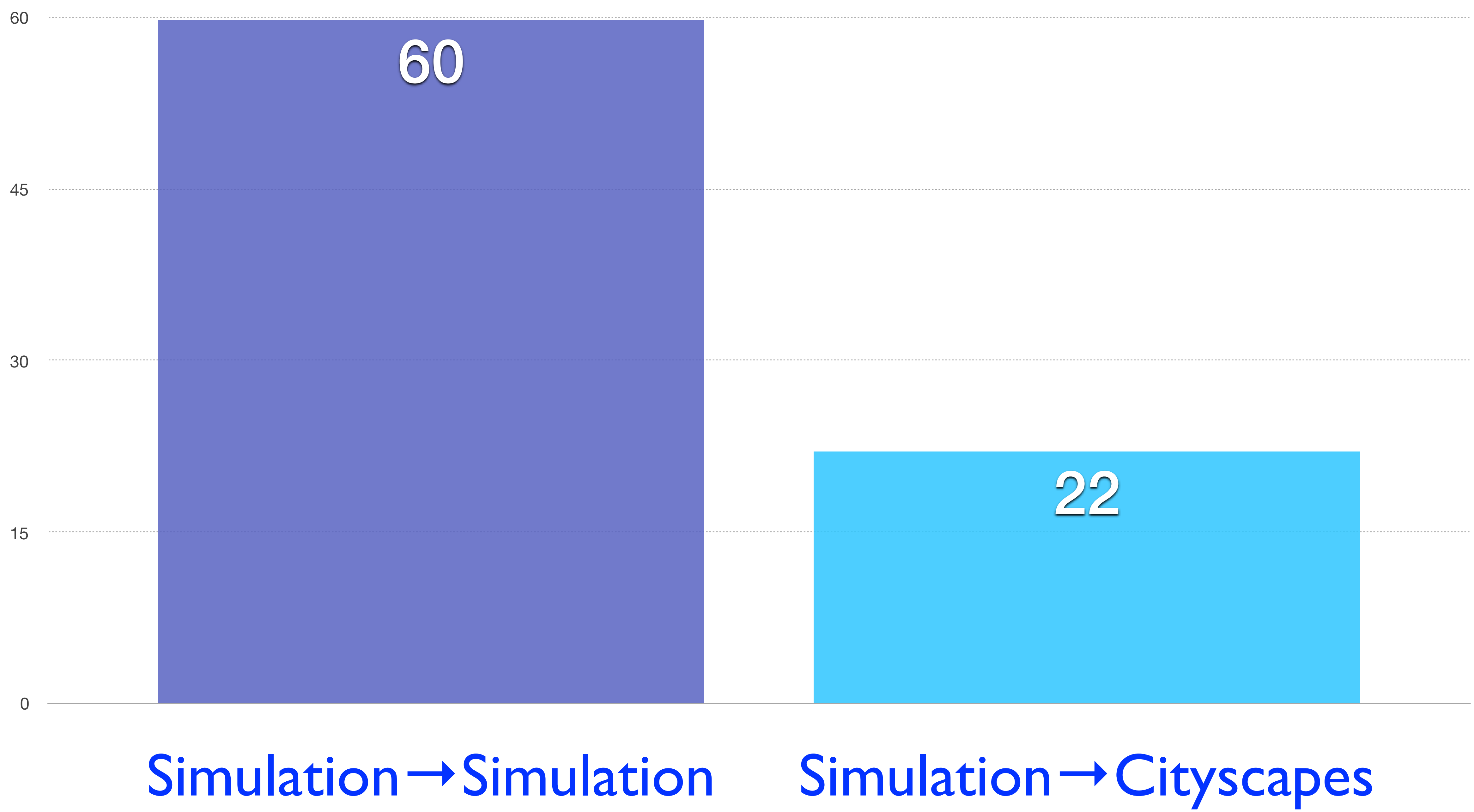
Image credit: https://www.cityscapes-dataset.com/

Labeling-free training data by simulation





Simulation to real world: catastrophic performance drop



The perils of mismatched domains

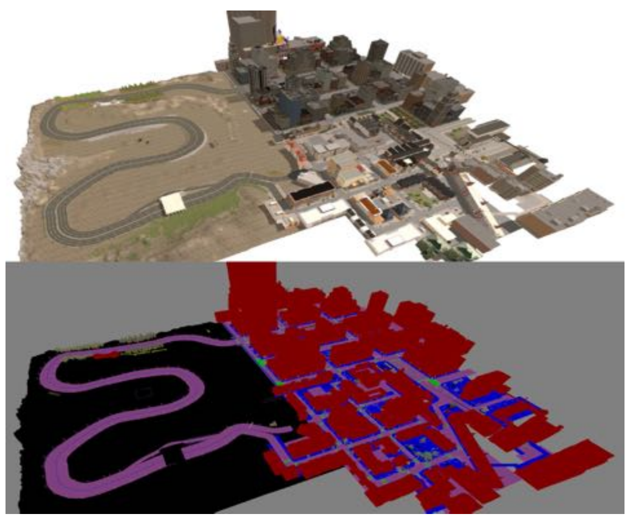
Cause: standard assumption in machine learning

Same underlying distribution for training and testing

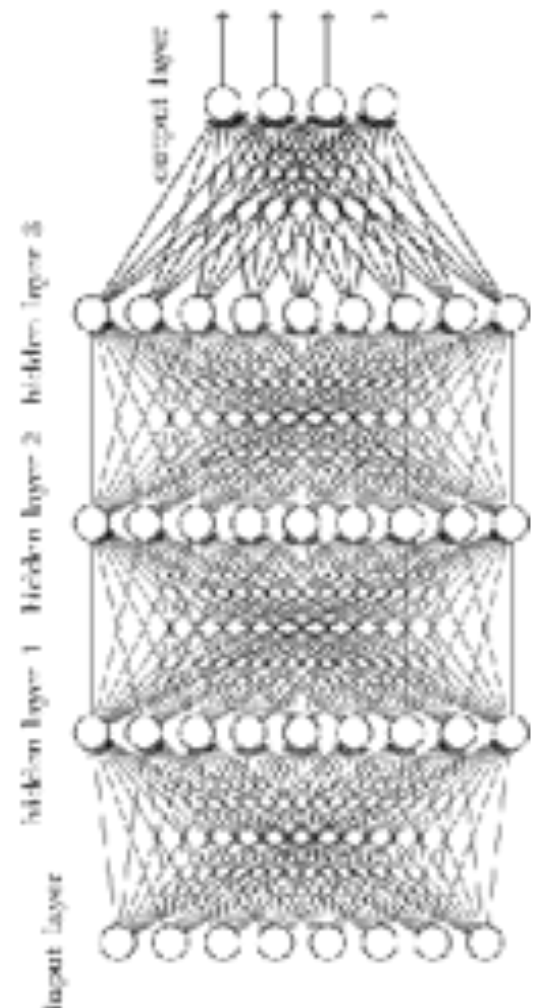
Consequence:

Poor cross-domain generalization

Brittle systems in dynamic and changing environment

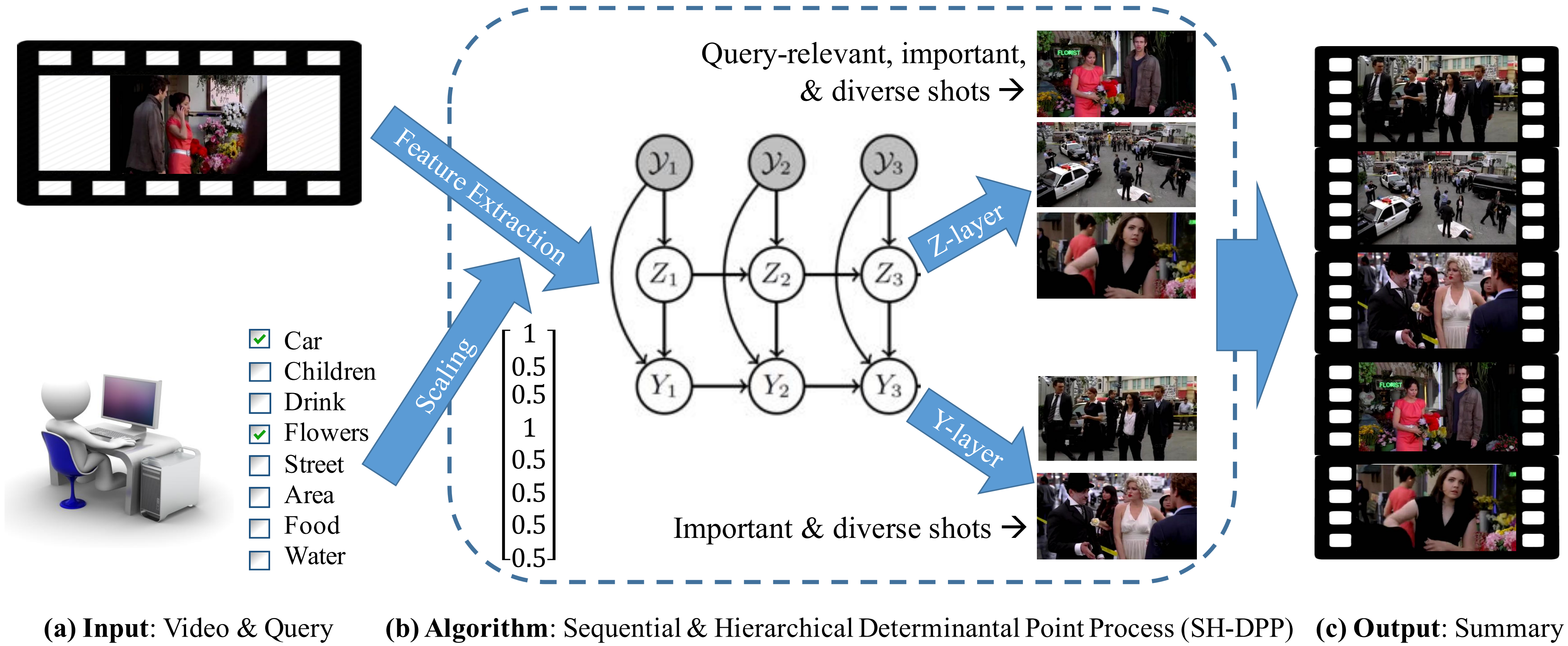


Synthetic imagery → Real photos

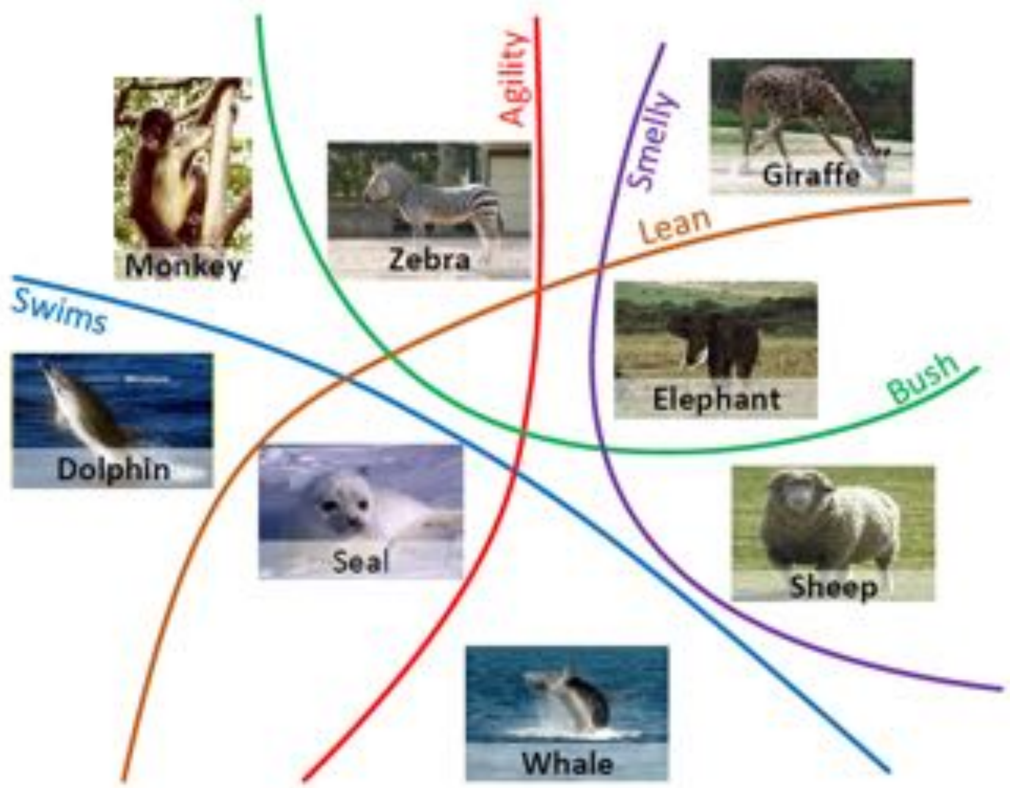


Webly supervised learning

Adapting face detector to a user’s album



Personalization of video summarizers



Middle-level concepts to describe objects, faces, etc.

Shared by different categories

Attribute detection

Abstract form: *unsupervised* domain adaptation (DA)

Setup

Source domain (with labeled data)

$$D_S = \{(x_m, y_m)\}_{m=1}^{\mathsf{M}} \sim \boxed{P_S(X,Y)}$$

Target domain (no labels for training)

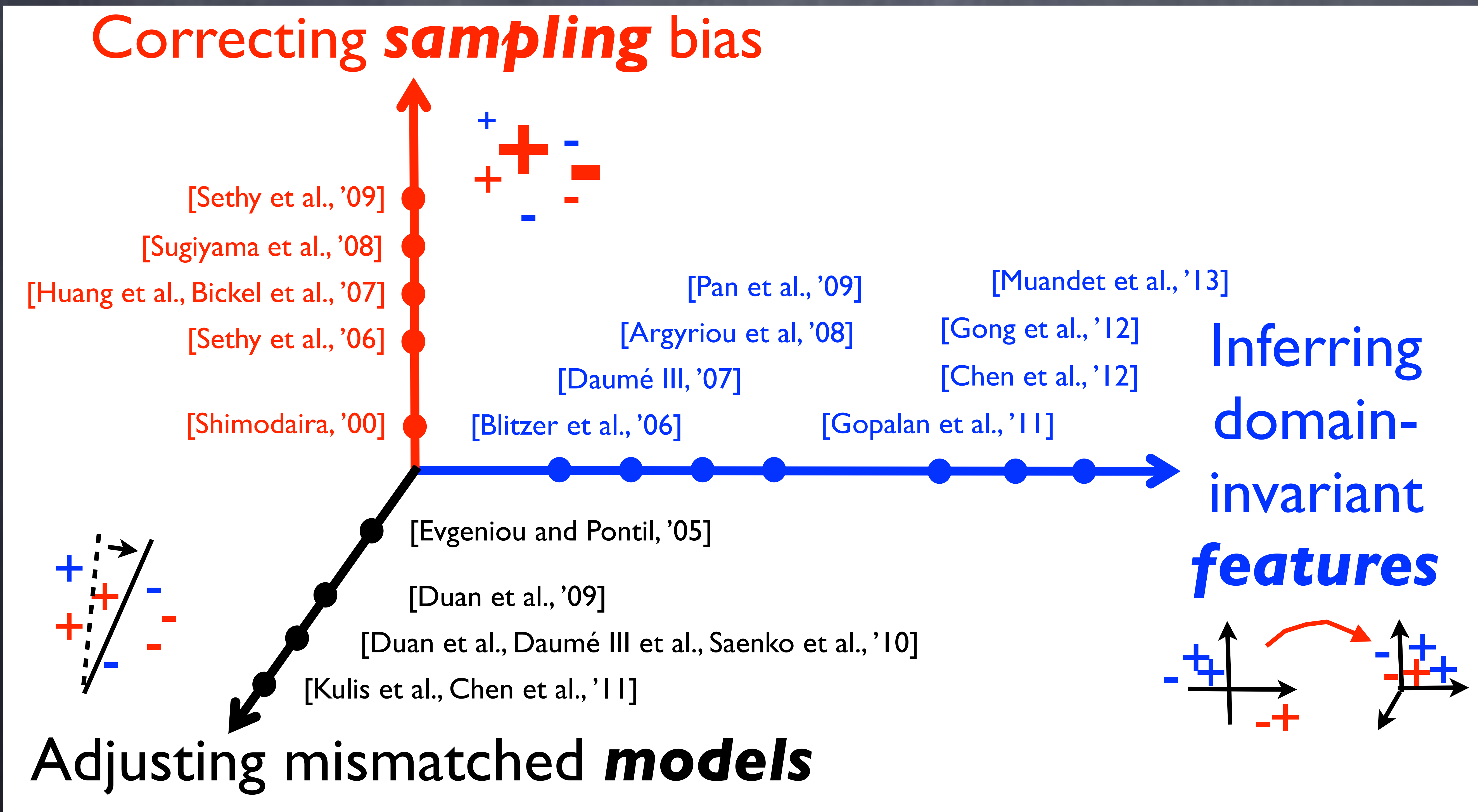
$$D_T = \{(x_n, \textbf{?})\}_{n=1}^{\mathsf{N}} \sim \boxed{P_T(X,Y)}$$

Different distributions

Objective

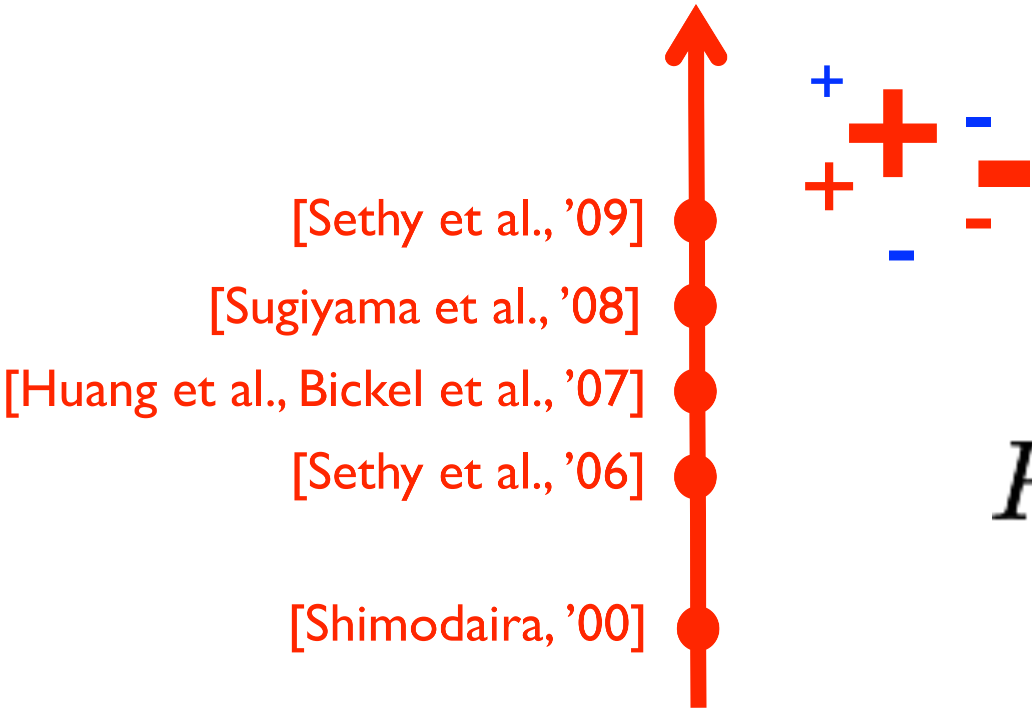
Learn models to work well on target

This talk


Correcting *sampling* bias
[Sethy et al., '09]
[Sugiyama et al., '08]
[Huang et al., Bickel et al., '07]
[Sethy et al., '06]
[Shimodaira, '00]
[Pan et al., '09]
[Argyriou et al, '08]
[Daumé III, '07]
[Blitzer et al., '06]
[Muandet et al., '13]
[Gong et al., '12]
[Chen et al., '12]
[Gopalan et al., '11]
Inferring domain-invariant *features*
[Evgeniou and Pontil, '05]
[Duan et al., '09]
[Duan et al., Daumé III et al., Saenko et al., '10]
[Kulis et al., Chen et al., '11]
Adjusting mismatched *models*

This talk

Correcting ***sampling*** bias



$$P_{\mathcal{L}}(\text{landmarks}) \approx P_{\mathcal{T}}(\text{target})$$

$$\min_{\text{landmarks}} \quad d(P_{\mathcal{L}}, P_{\mathcal{T}})$$

Selecting most adaptable source instances

Landmarks are labeled source instances distributed similarly to the target domain.

Source

Target

[ICML'13]

Selecting most adaptable source instances

Landmarks are labeled source instances distributed similarly to the target domain.

Source

Identifying landmarks:

$$P_{\mathcal{L}}(\text{landmarks}) \approx P_{\mathcal{T}}(\text{target})$$

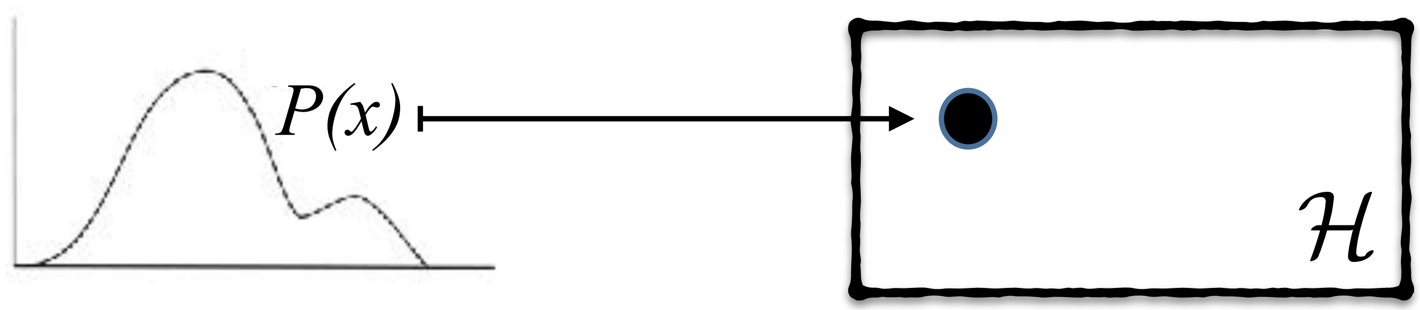
$$\min_{\text{landmarks}} d(P_{\mathcal{L}}, P_{\mathcal{T}})$$

Target

[ICML'13]

Kernel embedding of distributions

$$\mu[P] \triangleq \mathbb{E}_x[\phi(x)]$$



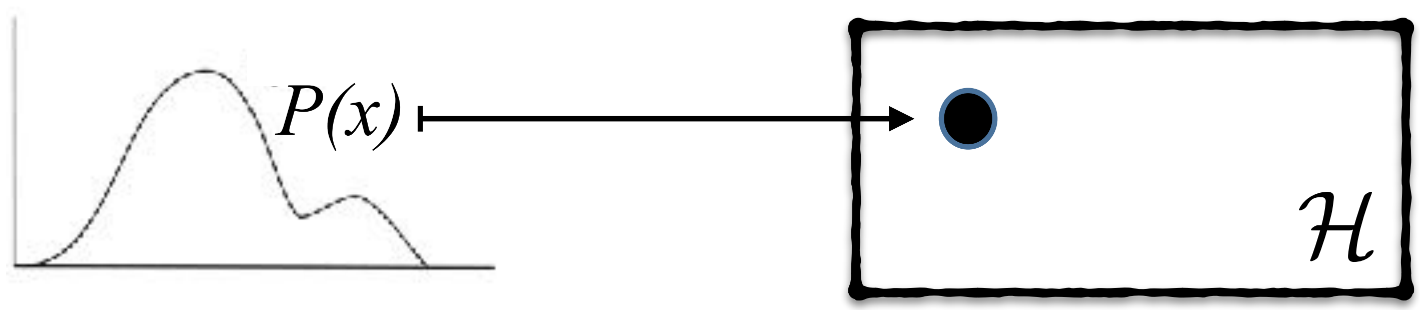
μ maps distribution P to Reproducing Kernel Hilbert Space

μ is injective if $\phi(\cdot)$ is characteristic

[Müller'97,Gretton et al.'07,Sriperumbudur et al.'10]

Kernel embedding of distributions

$$\mu[P] \triangleq \mathbb{E}_x[\phi(x)]$$



Empirical kernel embedding:

$$\hat{\mu}[P] = \frac{1}{\mathsf{n}} \sum_{i=1}^{\mathsf{n}} \phi(x_i), \quad x_i \sim P$$

Identifying landmarks by matching kernel embeddings

Integer programming

$$\min_{\{\alpha_m\}} \left\| \frac{1}{\sum_i \alpha_i} \sum_{m=1}^{\mathsf{M}} \alpha_m \phi(x_m) - \frac{1}{\mathsf{N}} \sum_{n=1}^{\mathsf{N}} \phi(x_n) \right\|_{\mathcal{H}}^2$$

where

$$\alpha_m = \begin{cases} 1 & \text{if } x_m \text{ is a } \textbf{landmark} \text{ wrt target} \\ 0 & \text{else} \end{cases}$$

$$m = 1, 2, \cdots, \mathsf{M}$$

Solving by relaxation

Convex relaxation

$$\min_{\{\alpha_m\}} \left\| \frac{1}{\sum_i \alpha_i} \sum_{m=1}^{M} \alpha_m \phi(x_m) - \frac{1}{N} \sum_{n=1}^{N} \phi(x_n) \right\|_{\mathcal{H}}^2$$

$$\beta_m = \frac{\alpha_m}{\sum_i \alpha_i} \rightarrow \text{Quadratic programming}$$

$$\min_{\beta} \quad \beta^T K^s \beta - \frac{2}{N} \beta^T K^{st} \mathbf{1}$$

Other details

Class balance constraint

Recovering $\alpha_m^\star$ from $\beta_m^\star (= \frac{\alpha_m}{\sum_i \alpha_i})$

Multi-scale analysis

(See [Gong et al., ICML'13, IJCV'14] for details)

Experimental study

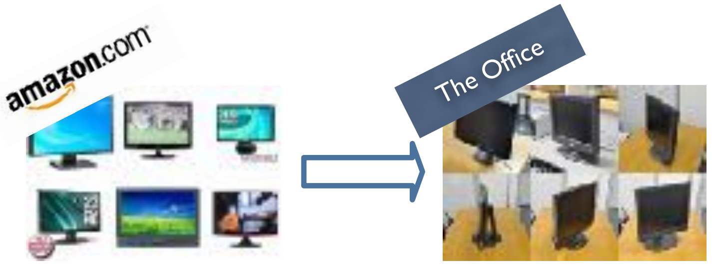
Four vision datasets/domains on visual object recognition

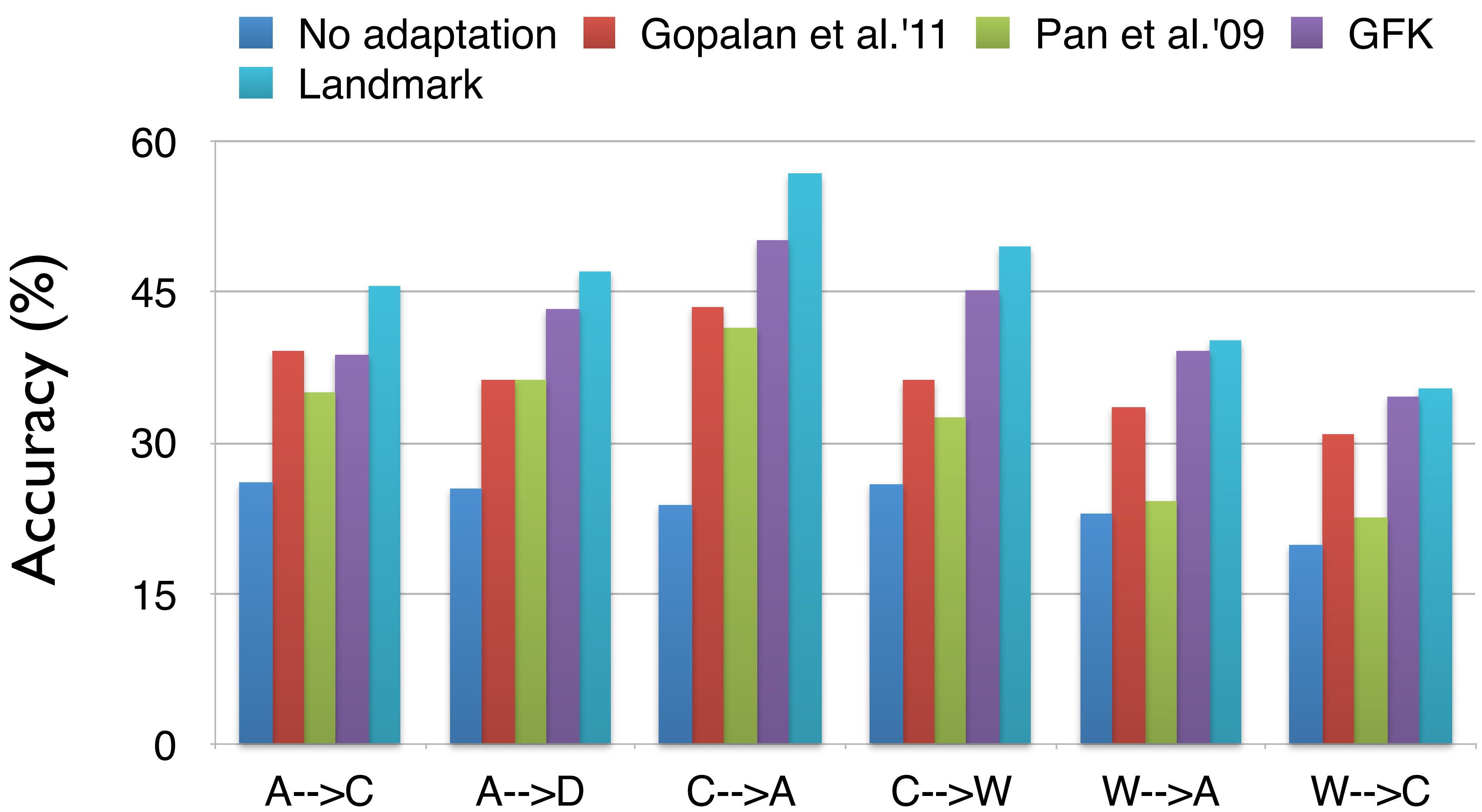
[Griffin et al. '07, Saenko et al. 10']

Four types of product reviews on sentiment analysis

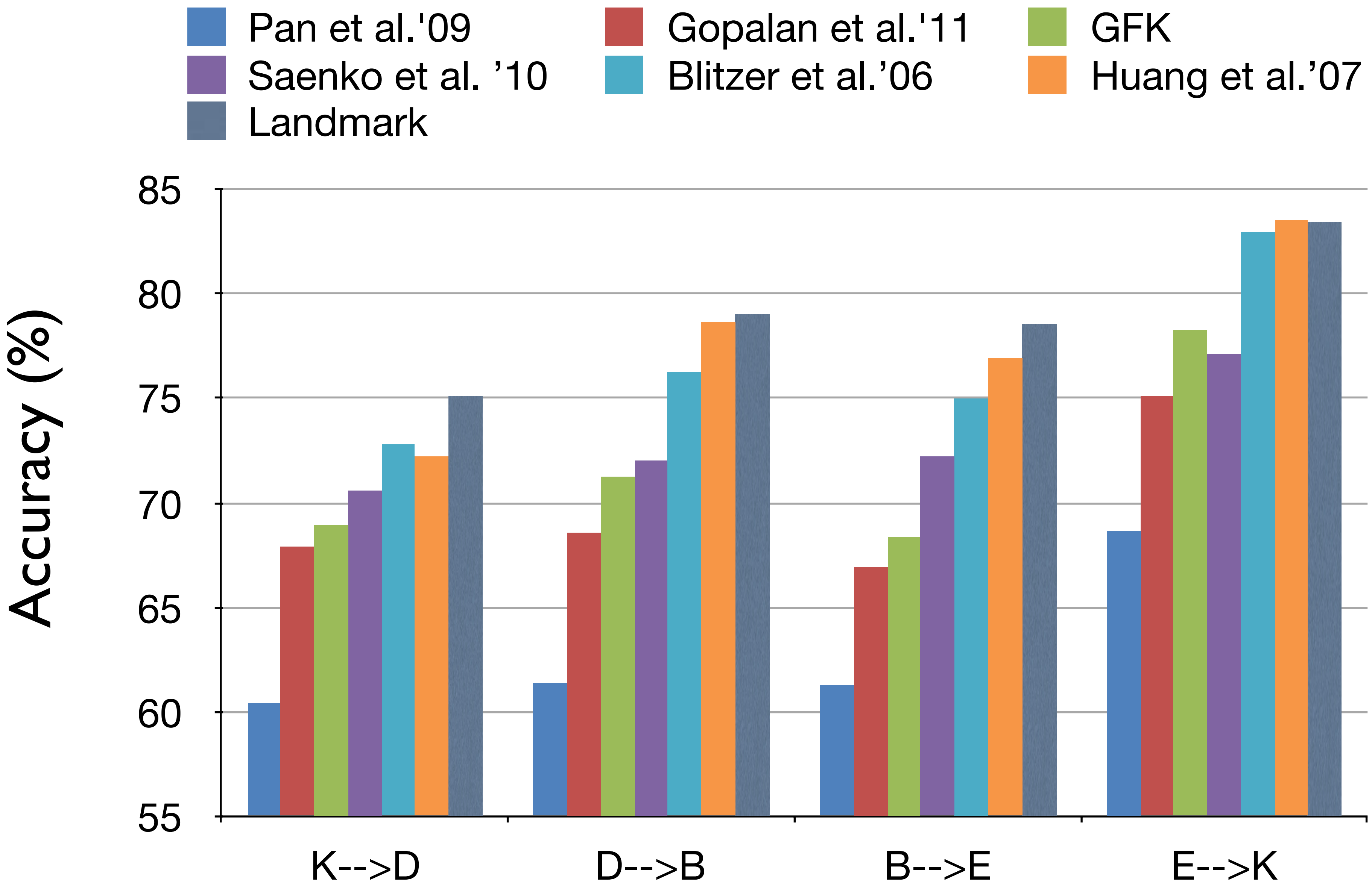
Books, DVD, electronics, kitchen appliances [Biltzer et al. '07]

Comparison results: object recognition





Comparison results: sentiment analysis



What do landmarks look like?

Summary - Landmarks

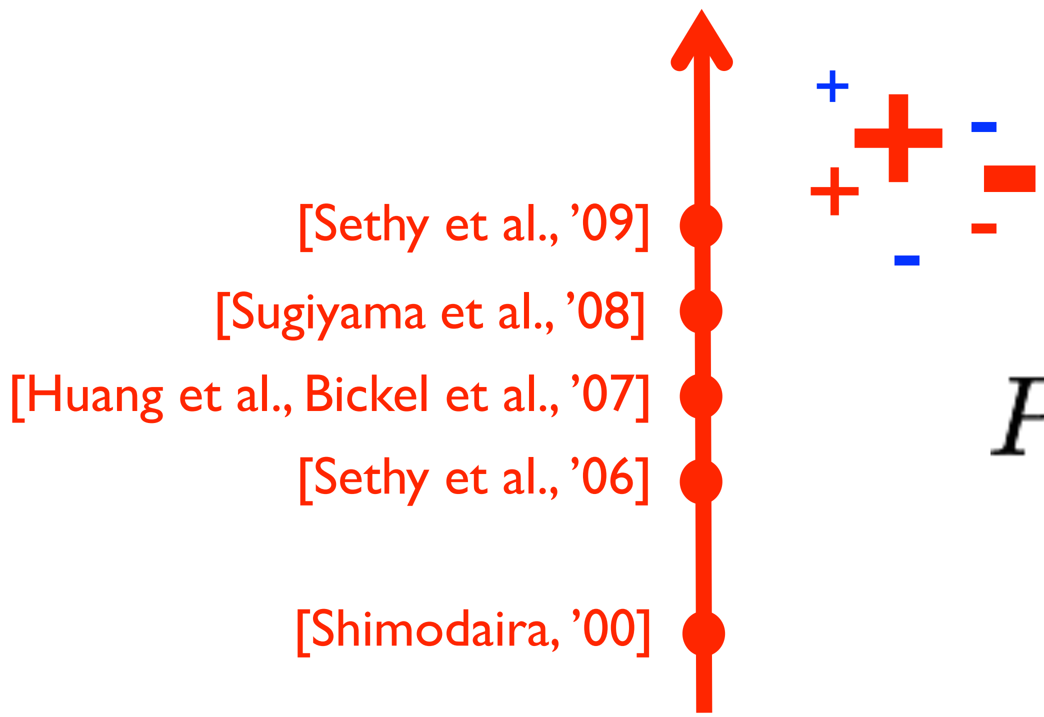
Landmarks

[Gong *et al.*, ICML'13]

- *Labeled source instances, distributed similarly to target*
- *Better approximation of discriminative loss of target*
- *Automatically identifying landmarks*
- *Benefiting other adaptation methods*

Snags in landmarks

Correcting ***sampling*** bias



$$P_{\mathcal{L}}(\text{landmarks}) \approx P_{\mathcal{T}}(\text{target})$$

$$\min_{\text{landmarks}} \quad d(P_{\mathcal{L}}, P_{\mathcal{T}})$$

No sufficient data of the target domain?

Solution: *landmarks of multiple source domains are also shared by the target domain*

No sufficient data of the target domain?

E.g., human activity recognition on the fly

Gan, C., Sun, C., Duan, L., & Gong, B. (2016). Webly-supervised Video Recognition by Mutually Voting for Relevant Web Images and Web Video Frames. In *European Conference on Computer Vision.*

Web videos are often redundant, *sometimes misleading*

Bench Press

Pizza Tossing

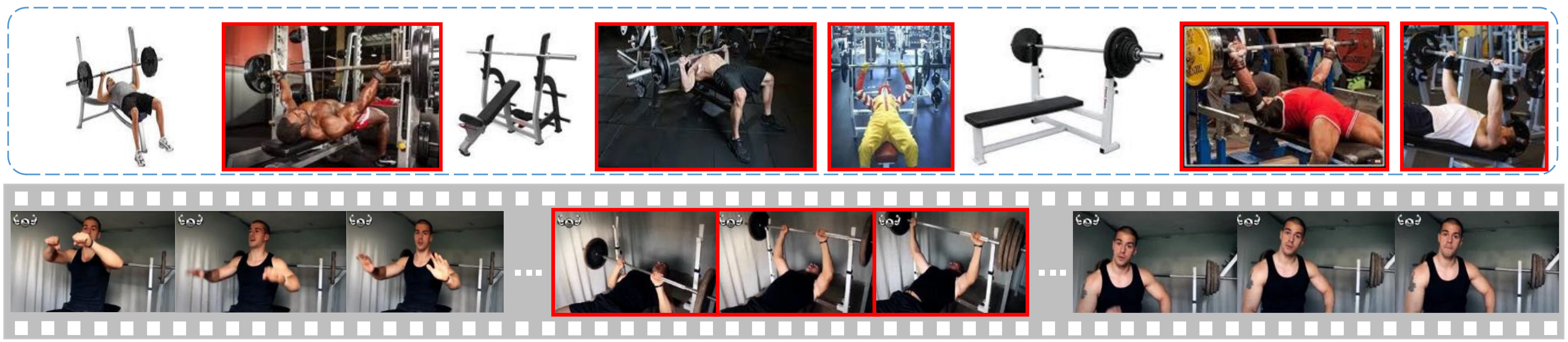
Web images are informative for activity recognition, *and noisy*

Bench Press

Pizza Tossing

Mutually voting for landmarks!

(a) Basketbal Dunk



(b) Bench Press

(c) Pizza Tossing

Experimental results on UCF101

Table 1: Comparison results on UCF101.

Method	Accuracy (%)
Karpathy et al. [20]	65.4
LRCN [7]	71.1
Spatial stream net. [29]	73.0
Ours	69.3

Sophisticated model learned from ***manually pruned and labeled*** training videos.

SVM trained from ***auto-pruned*** Web images & Web videos.

Experimental results on UCF101

Table 1: Comparison results on UCF101.

Method	Accuracy (%)
Karpathy et al. [20]	65.4
LRCN [7]	71.1
Spatial stream net. [29]	73.0
LSTM composite [34]	75.8
C3D [40]	82.3
IDT + FV [41]	87.9
Ours	69.3

Sophisticated model learned from ***manually pruned and labeled*** training videos.

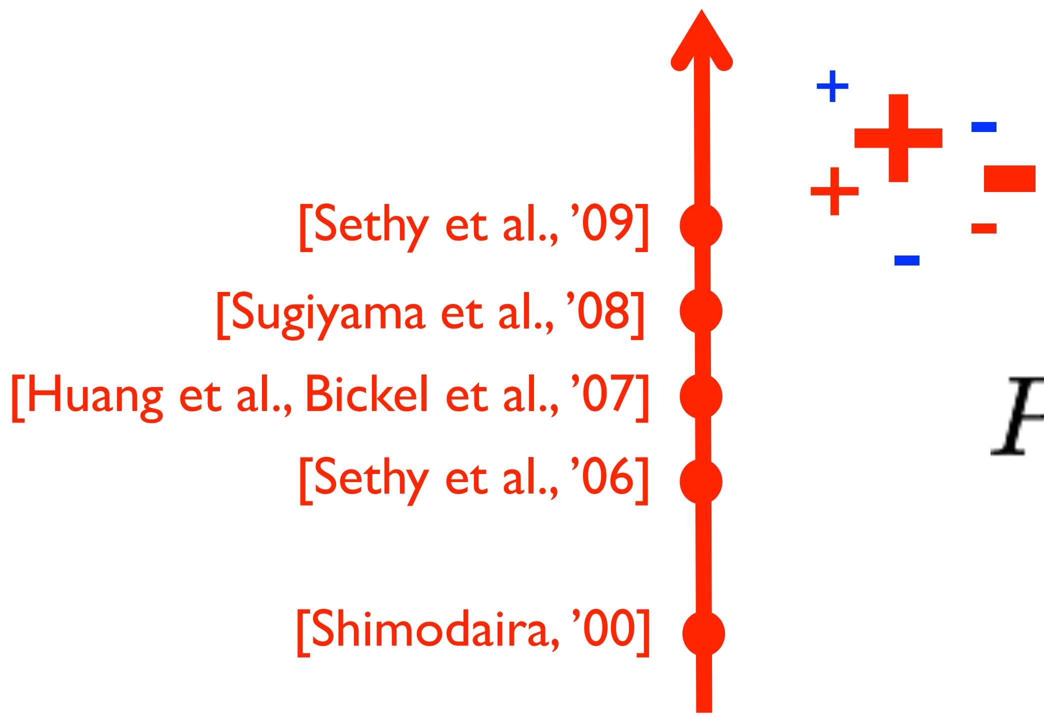
Motion, or temporal features

SVM trained from ***auto-pruned*** Web images & Web videos.

[ECCV'16]

Snags in Landmarks

Correcting ***sampling*** bias



$$P_{\mathcal{L}}(\text{landmarks}) \approx P_{\mathcal{T}}(\text{target})$$

$$\min_{\text{landmarks}} \quad d(P_{\mathcal{L}}, P_{\mathcal{T}})$$

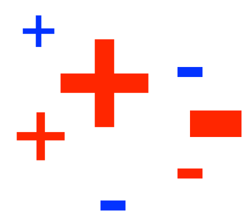
No sufficient data of the target domain?

Solution: *landmarks of multiple source domains are also shared by the target domain*

Snags in Landmarks

Correcting ***sampling*** bias

[Sethy et al., '09]
[Sugiyama et al., '08]
[Huang et al., Bickel et al., '07]
[Sethy et al., '06]
[Shimodaira, '00]

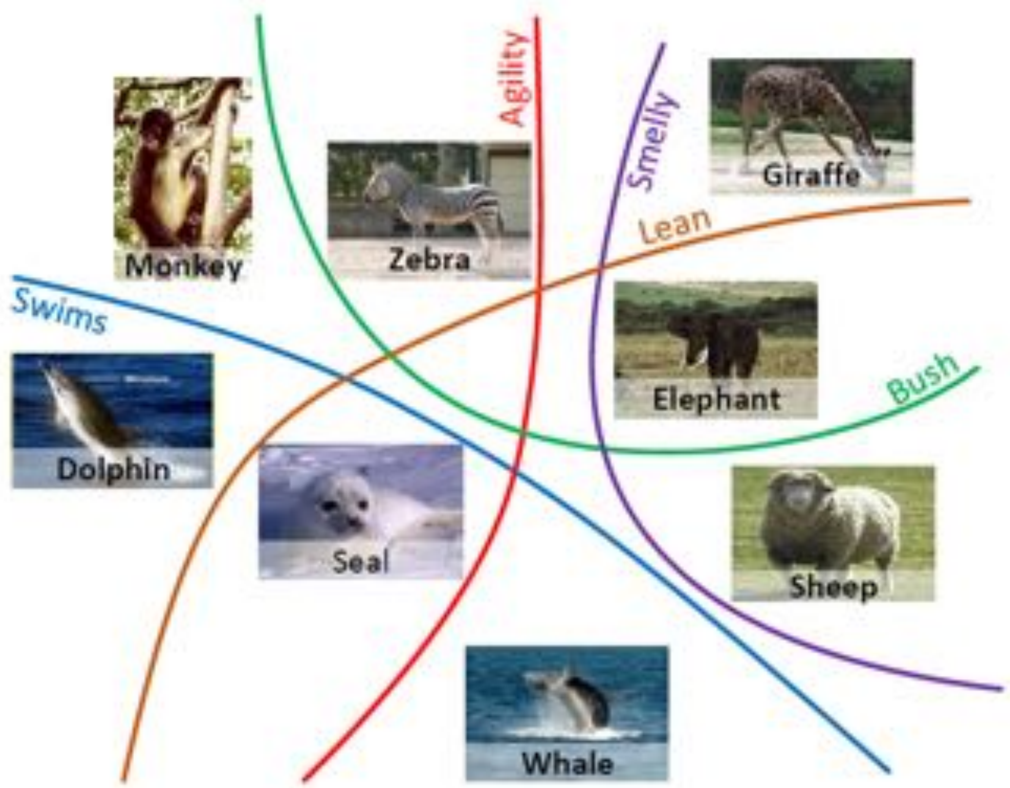


$$P_{\mathcal{L}}(\text{landmarks}) \approx P_{\mathcal{T}}(\text{target})$$

$$\min_{\text{landmarks}} \quad d(P_{\mathcal{L}}, P_{\mathcal{T}})$$

No sufficient data of the target domain?

Large inter-domain discrepancy (*seal vs whale*)?



Middle-level concepts to describe objects, faces, etc.

Shared by different categories

Attribute detection

Visual attributes

What are visual attributes?

Middle-level concepts to describe objects, faces, etc.

Examples: *four-legged, smiley, outdoor, crowded, etc.*

Properties

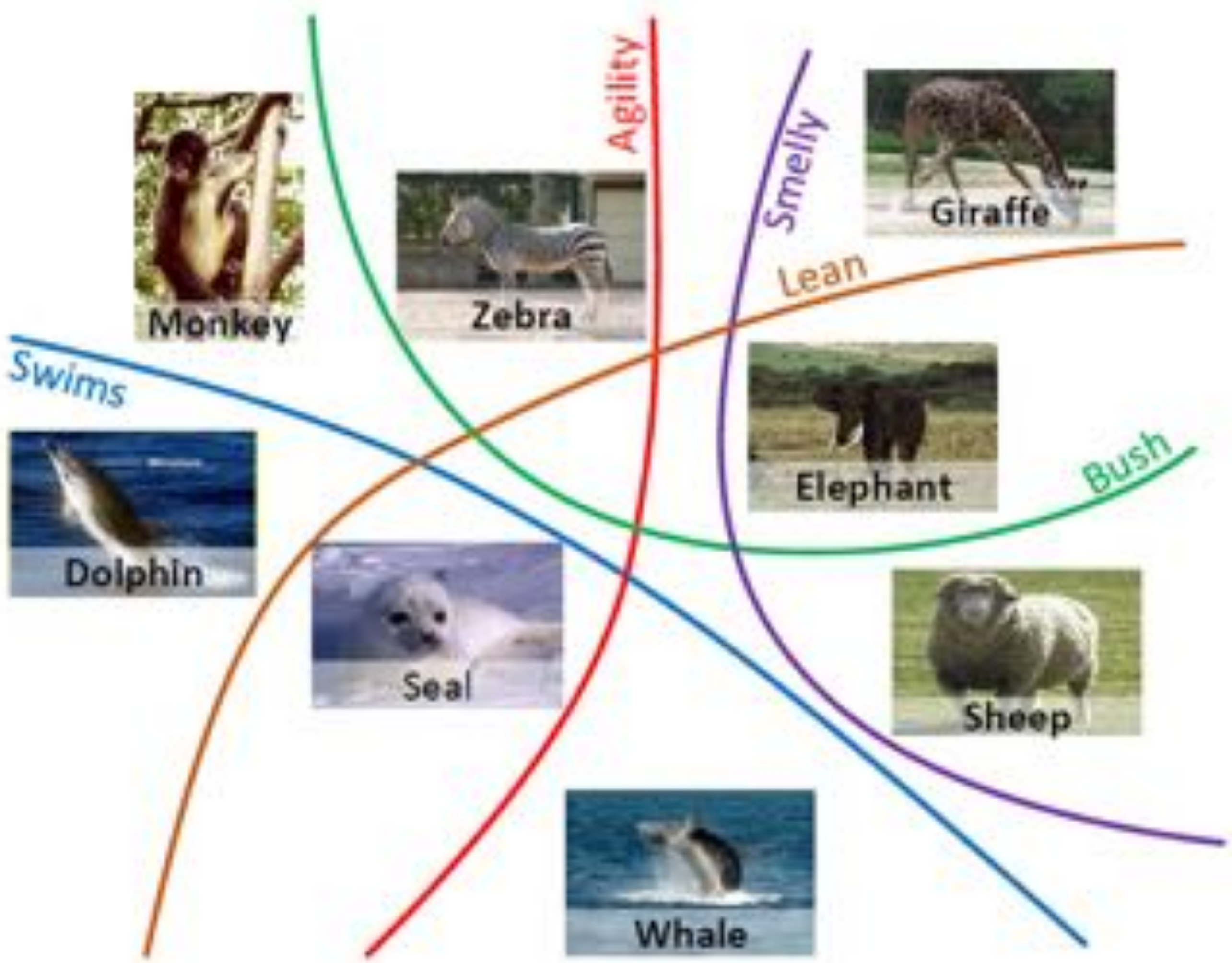
Human-nameable & machine-detectable

Shared by different categories

Applications

Zero-shot learning, image search, HCI, etc.

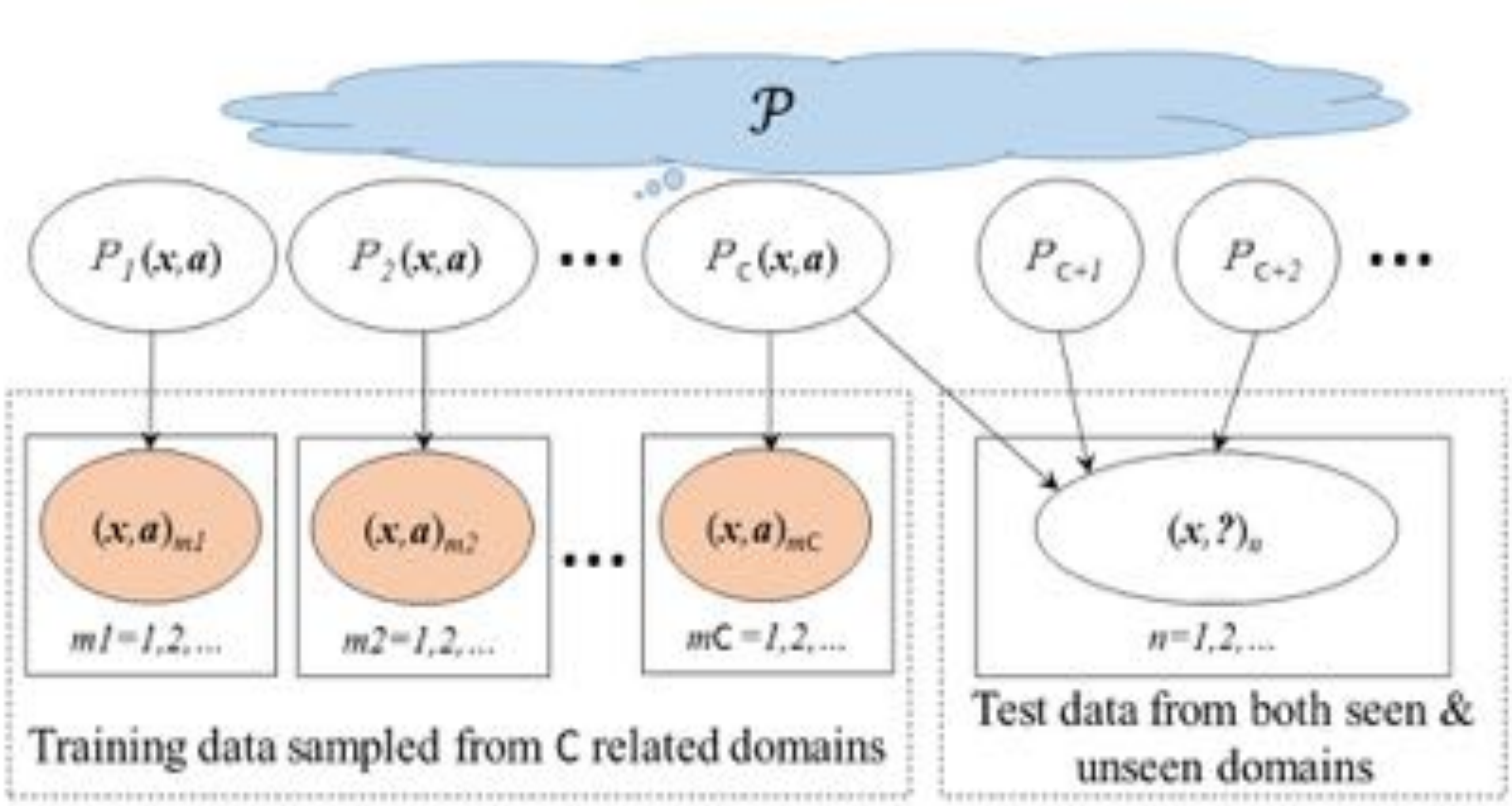
What makes a good attribute detector?



Effective, efficient, … and *generalize well across different activity categories*, including previously unseen ones.

Boundaries between middle-level attributes and high-level object classes cross each other.

Domain adaptation for attribute detection

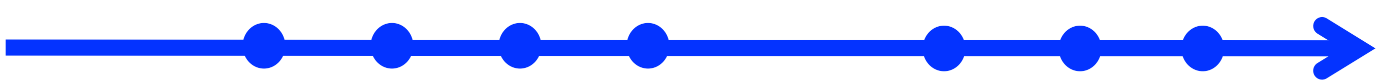


This talk

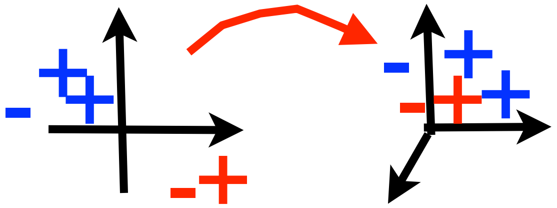
$$\mathbf{x} \mapsto \mathbf{z}, \quad \text{s.t.}$$

$$P_{\mathcal{S}}(z,y) \approx P_{\mathcal{T}}(z,y)$$

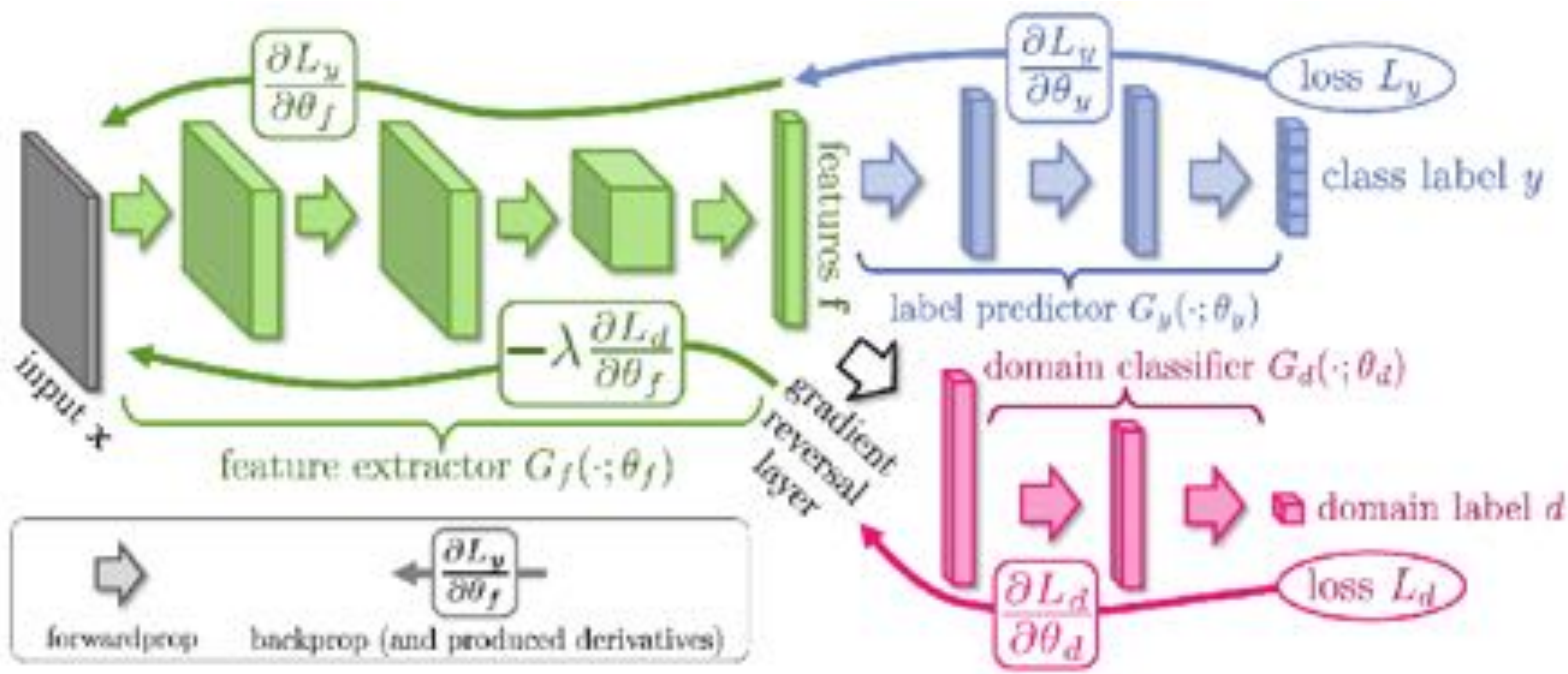
[Pan et al., '09]
[Argyriou et al, '08]
[Daumé III, '07]
[Blitzer et al., '06]
[Gopalan et al., '11]
[Chen et al., '12]
[Gong et al., '12]
[Muandet et al., '13]



Inferring domain-invariant ***features***



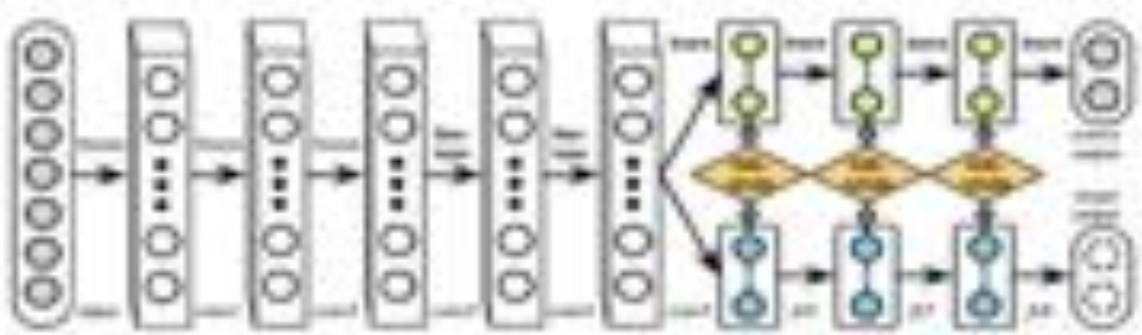
Review: maximizing the domain classification loss



Ganin, Y., & Lempitsky, V. (2014). Unsupervised domain adaptation by backpropagation. *International Conference on Machine Learning*.

Review

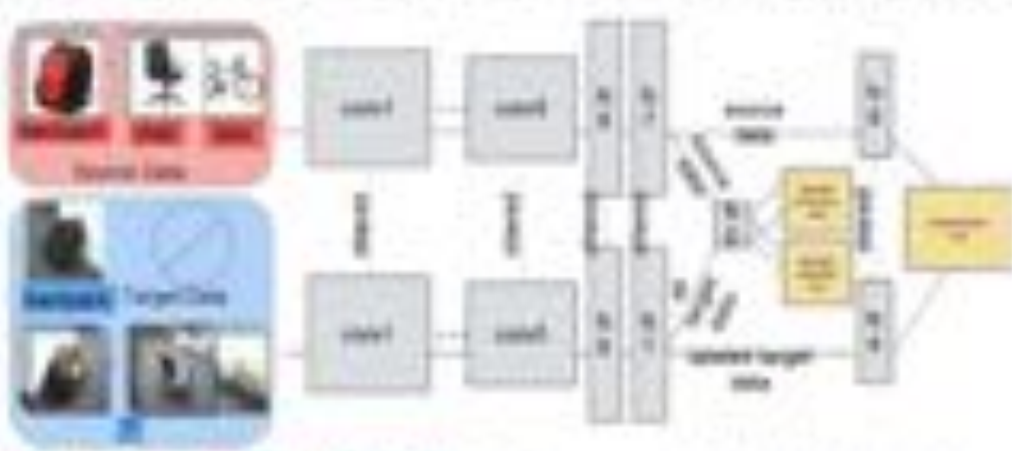
- by minimizing distance between distributions, e.g.



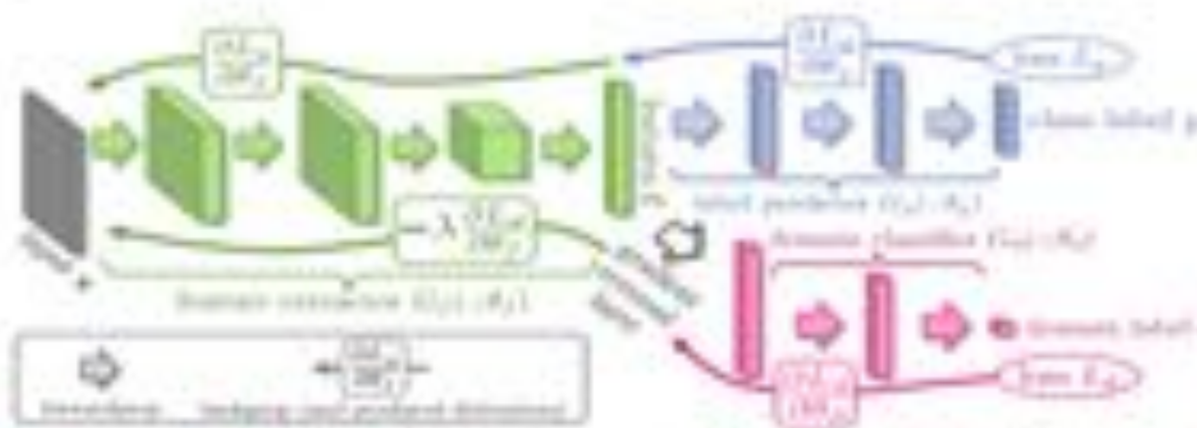
Maximum Mean Discrepancy M. Long, et al. ICML 2015

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

Attribute detection results

Approcahes	AWA	CUB	a-Yahoo	UCF101
IAP [44]	74.0/79.2*	74.9*	–	–
ALE [1]	65.7	60.3	–	–
HAP [12]	74.0/79.1*	68.5/74.1*	58.2*	72.1 ± 1.1
CSHAP_G [12]	74.3/79.4*	62.7/74.6*	58.2*	72.3 ± 1.0
CSHAP_H [12]	74.0/79.0*	68.5/73.4*	65.2*	72.4 ± 1.1
DAP [44]	72.8/78.9*	61.8/72.1*	77.4*	71.8 ± 1.2
UDICA (Ours)	**83.9**	**76.0**	**82.3**	$\mathbf{74.3 \pm 1.3}$
KDICA (Ours)	**84.4**	**76.4**	**84.7**	$\mathbf{75.5 \pm 1.1}$

[CVPR'16]

Boosting zero-shot learning and image retrieval

Approcahes	AWA	CUB	UCF101
ALE [1]	37.4	18.0	–
HLE [1]	39.0	12.1	–
AHLE [1]	43.5	17.0	–
DA [35]	30.6	–	–
MLA [19]	41.3	–	–
ZSRF [34]	48.7	–	–
SM [20]	66.0	–	–
Embedding [2]	60.1	29.9	-
IAP [44]	42.2/49.4*	4.6/34.9*	–
HAP [12]	45.0/55.6*	17.5/40.7*	–
CSHAP_G [12]	45.0/54.5*	17.5/38.7*	–
CSHAP_H [12]	45.6/53.3*	17.5/36.9*	–
DAP [44]	41.2/58.9*	10.5/39.8*	26.8 ± 1.1
UDCIA (Ours)	**63.6**	**42.4**	$\mathbf{29.6 \pm 1.2}$
KDCIA (Ours)	**73.8**	**43.7**	$\mathbf{31.1 \pm 0.8}$

query	VGG	UDICA	KDICA
single	78.9	83.9	**84.4**
double	77.2	79.5	**81.0**
triple	76.1	78.6	**79.4**

query	VGG	UDICA	KDICA
single	76.3	78.5	**79.2**
double	75.9	76.1	76.1
triple	75.5	75.6	**75.8**

[CVPR'16]

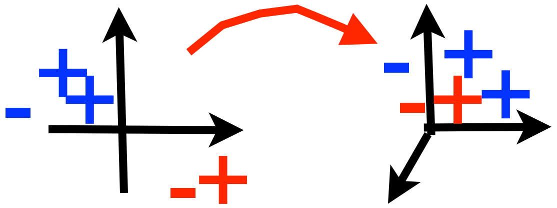
Pros: effective for large inter-domain discrepancy

$$\mathbf{x} \mapsto \mathbf{z}, \quad \text{s.t.}$$

$$P_{\mathcal{S}}(z, y) \approx P_{\mathcal{T}}(z, y)$$

[Blitzer et al., '06]
[Daumé III, '07]
[Argyriou et al, '08]
[Pan et al., '09]
[Gopalan et al., '11]
[Chen et al., '12]
[Gong et al., '12]
[Muandet et al., '13]

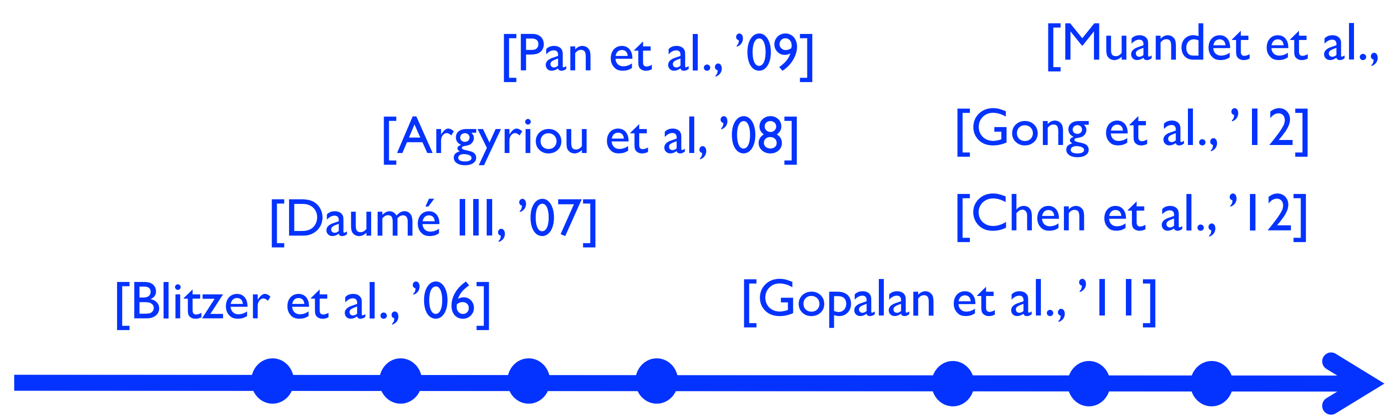
Inferring domain-invariant ***features***



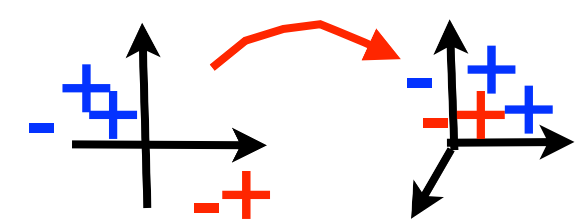
Cons: not discriminative enough for fine-grained tasks

$$\mathbf{x} \mapsto \mathbf{z}, \quad \text{s.t.}$$

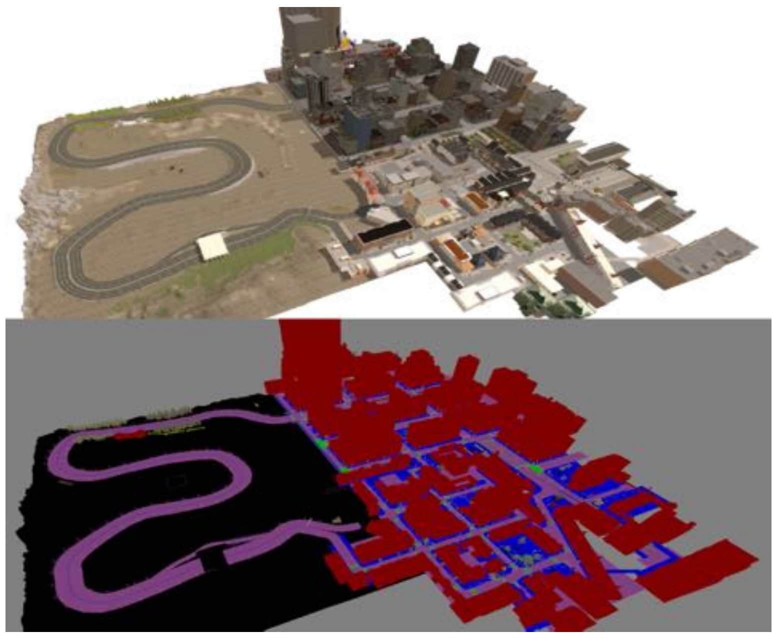
$$P_{\mathcal{S}}(z, y) \approx P_{\mathcal{T}}(z, y)$$



Inferring domain-invariant ***features***

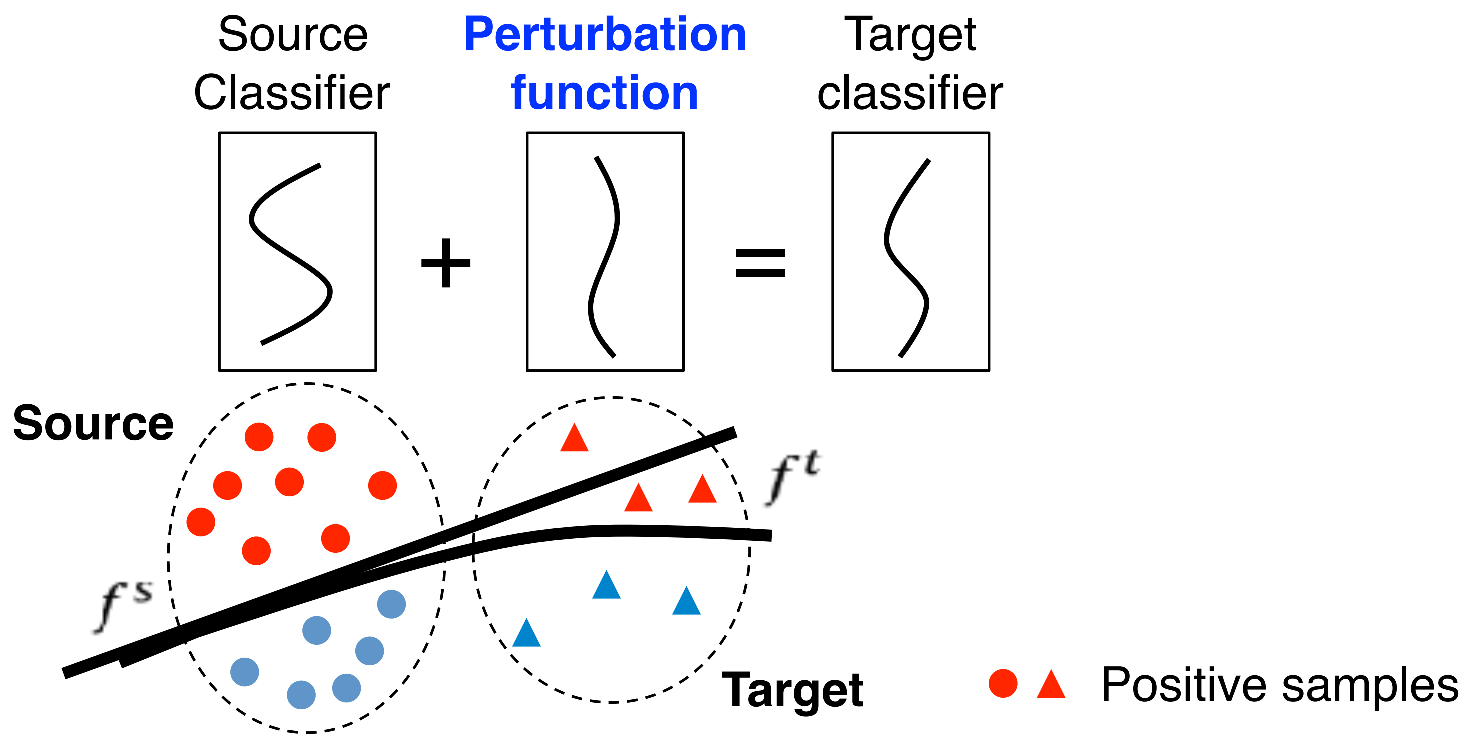


Cons: not discriminative enough for fine-grained tasks



E.g., semantic segmentation

Directly adapt classifiers/models



[Evgeniou and Pontil, ’05]

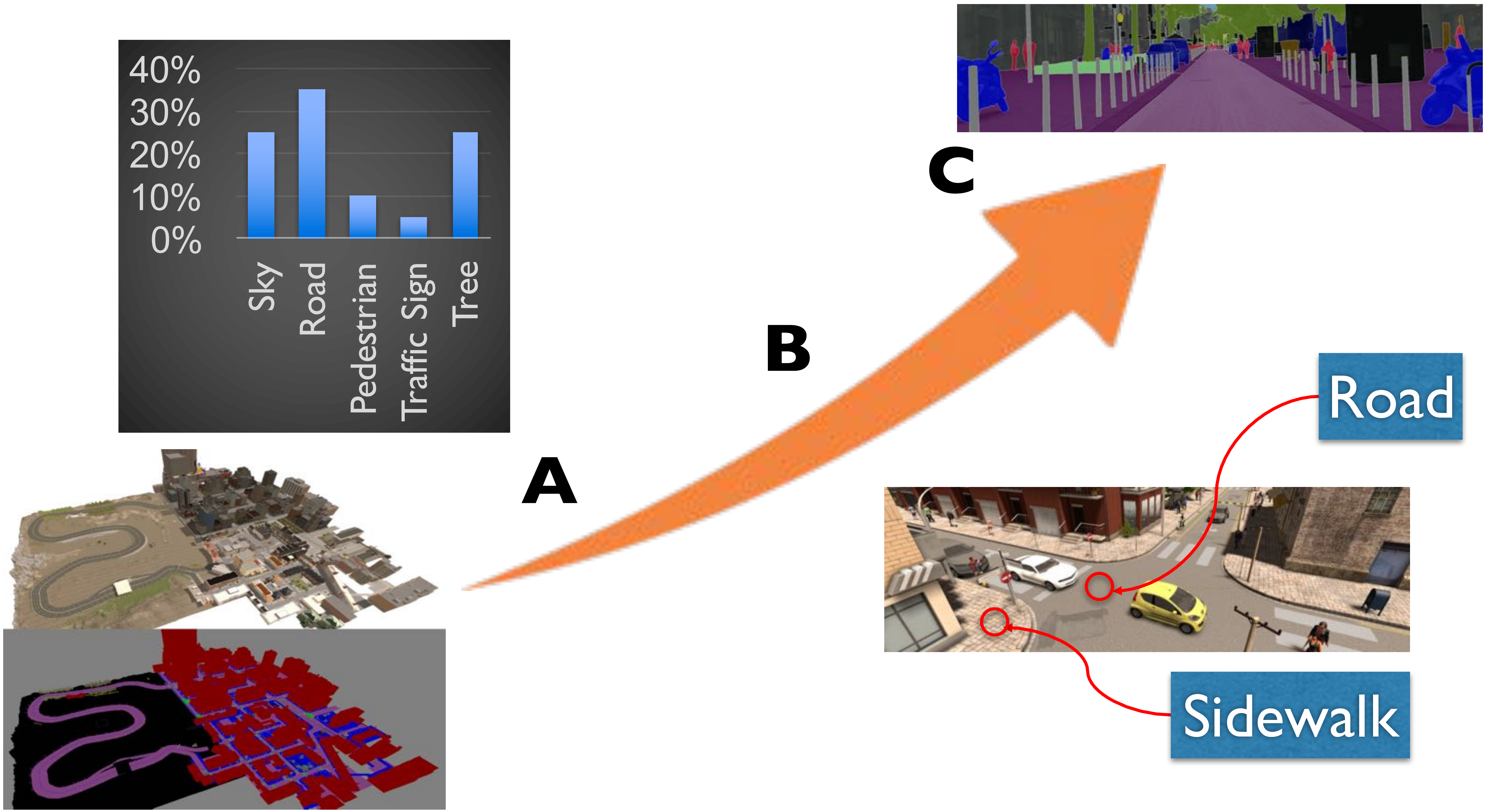
[Duan et al., ’09]

[Duan et al., Daumé III et al., Saenko et al., ’10]

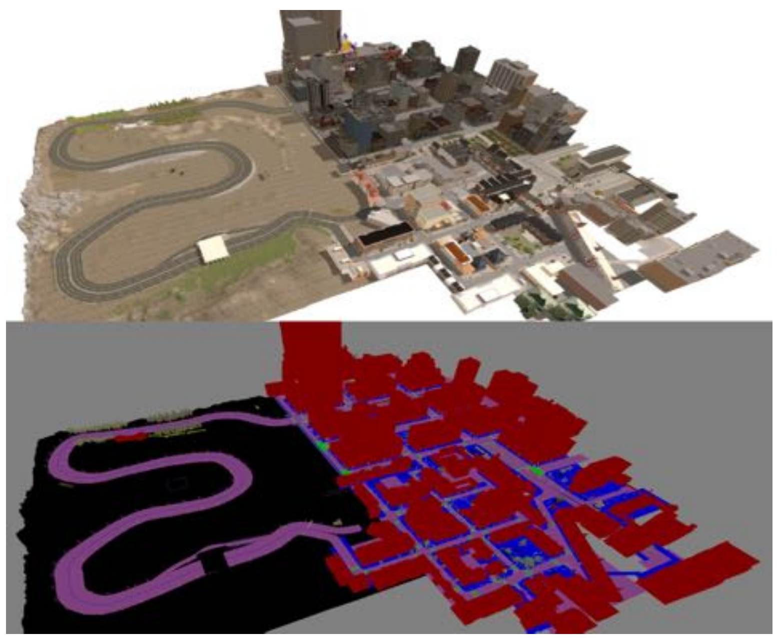
[Kulis et al., Chen et al., ’11]

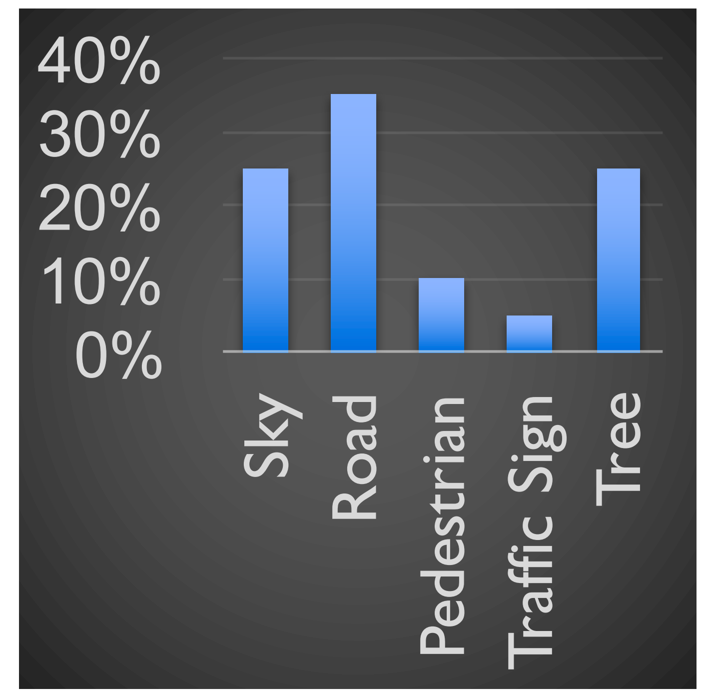
Adjusting mismatched ***models***

Curriculum domain adaptation



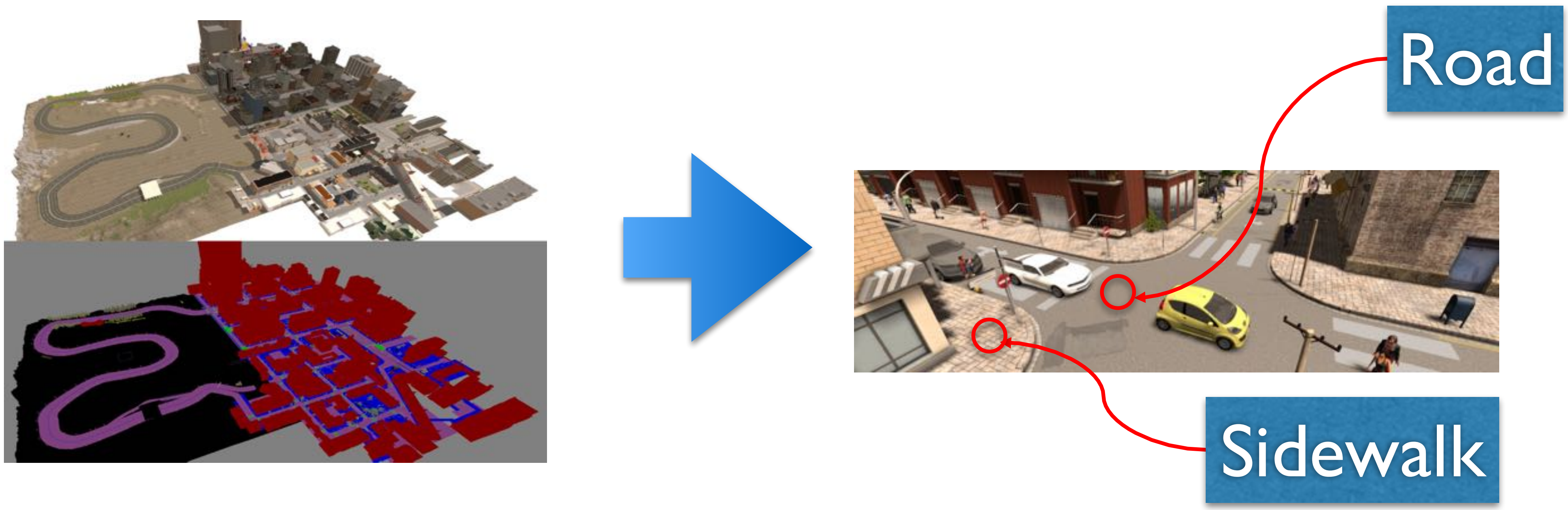
Perturbation functions for semantic segmentation (1)





Input: An urban scene image
Algorithm: Logistic regression
Output: Label distributions

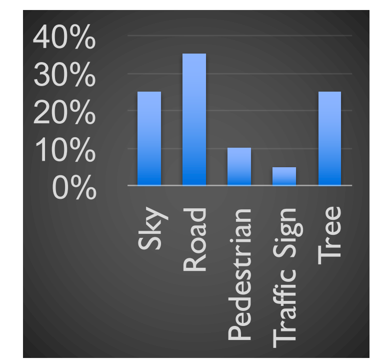
Perturbation functions for semantic segmentation (2)



Input: An urban scene image
Algorithm: Super-pixel + Logistic regression
Output: Labels of some super-pixels

Curriculum domain adaptation *for training CNNs*

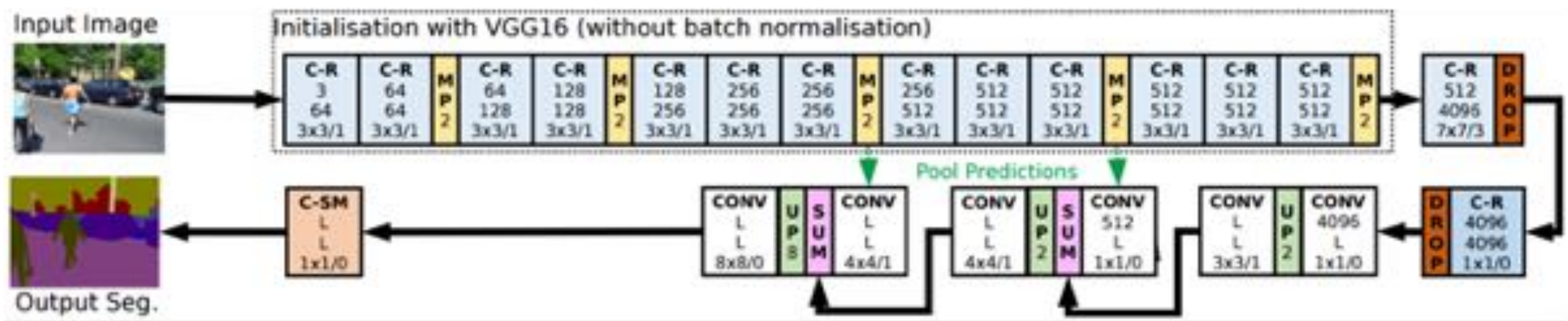
$$\min_{\Theta} \; \mathcal{L}(Y_s, \widehat{Y}_s) + d(p_t, p_t(\widehat{Y}_t))$$



s : Source, t : Target

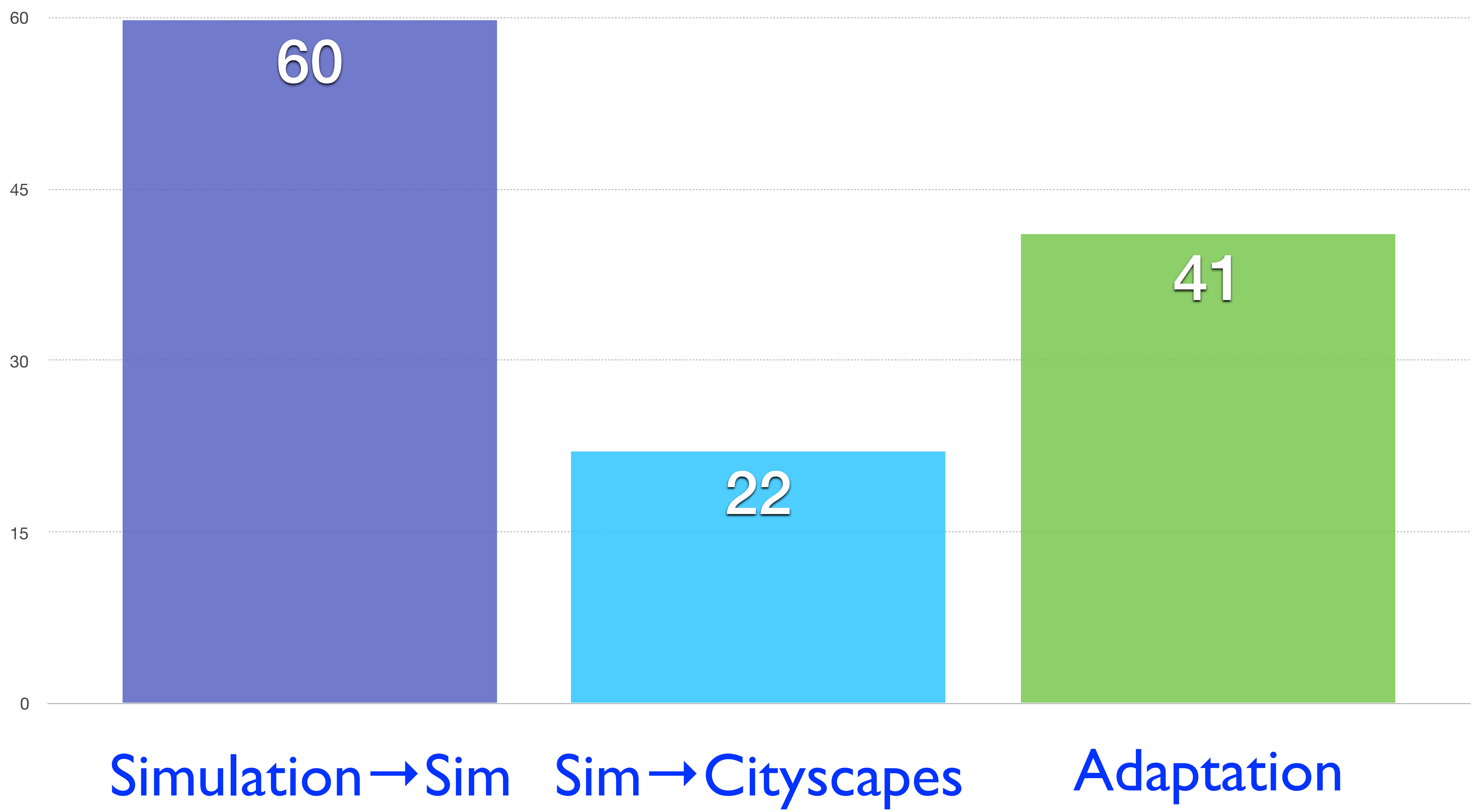
p_t : Perturbation function

$\widehat{Y}$



[ICCV'17]

Simulation to real world: ~~catastrophic~~ performance drop



[ICCV'17]

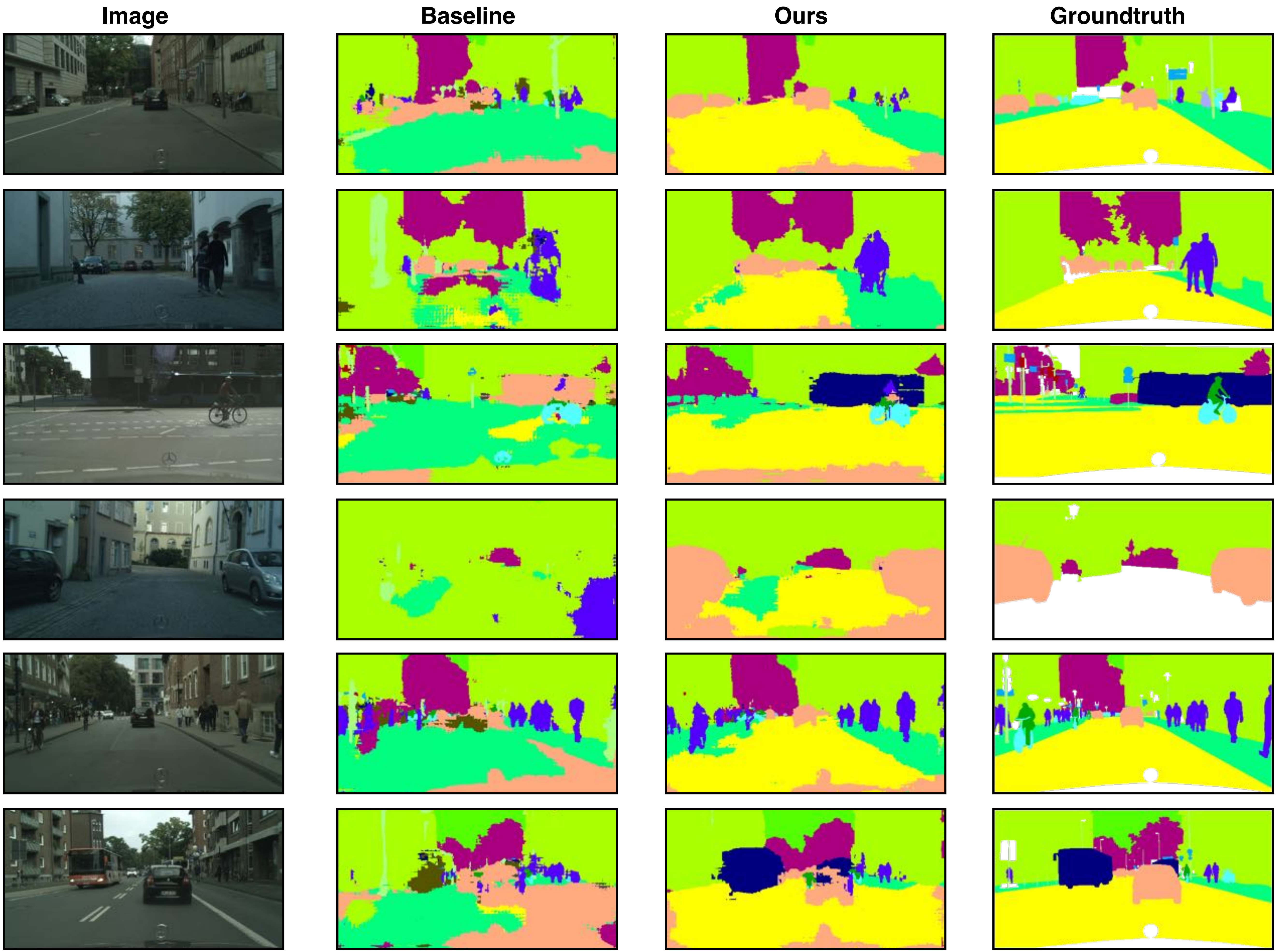
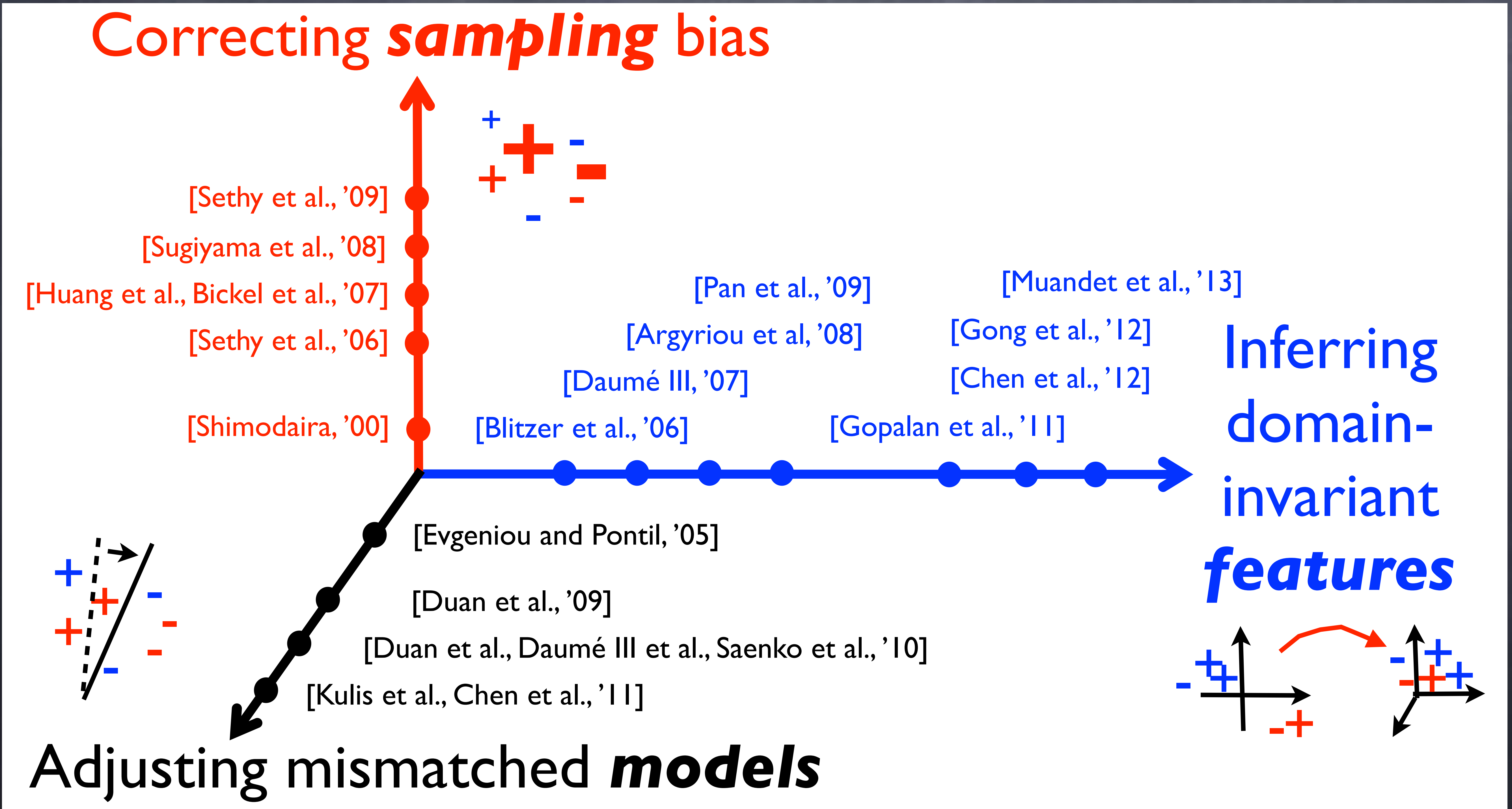


Image
Baseline
Ours
Groundtruth

This talk



Correcting *sampling* bias
[Sethy et al., '09]
[Sugiyama et al., '08]
[Huang et al., Bickel et al., '07]
[Sethy et al., '06]
[Shimodaira, '00]
[Pan et al., '09]
[Argyriou et al, '08]
[Daumé III, '07]
[Blitzer et al., '06]
[Muandet et al., '13]
[Gong et al., '12]
[Chen et al., '12]
[Gopalan et al., '11]
Inferring domain-invariant *features*
[Evgeniou and Pontil, '05]
[Duan et al., '09]
[Duan et al., Daumé III et al., Saenko et al., '10]
[Kulis et al., Chen et al., '11]
Adjusting mismatched *models*

Abstract form: *unsupervised* domain adaptation (DA)

Setup

Source domain (with labeled data)

$$D_S = \{(x_m, y_m)\}_{m=1}^{\mathsf{M}} \sim \boxed{P_S(X,Y)}$$

Target domain (no labels for training)

$$D_T = \{(x_n, \textbf{?})\}_{n=1}^{\mathsf{N}} \sim \boxed{P_T(X,Y)}$$

Different distributions

Objective

Learn models to work well on target

A realistic obstacle for autonomous systems

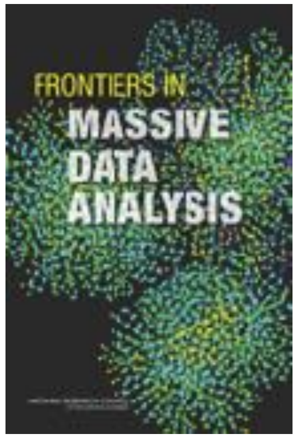
Systems often deployed to new environment, not lab reproducible

Expensive to collect training data from each type of target environment

Systems naturally degrade; environment dynamically evolves

Sampling bias & heterogeneity

“(training) Data may have been collected according to a certain criterion …, but (testing) the inferences and decisions may refer to a different sampling criterion.”



National Academies Report

Domain adaptation (DA) & related

2012	**We started the journey**
2011	**Computer vision:** **datasets are biased** *[Torralba & Efros]*
2009, 10	**Computer vision: classification**
2000s	Machine learning, NLP: DA, covariate shift, transfer learning, etc.
1990s	Speech: speaker adaptation
1970s	Statistics & econometrics: sampling bias

Summary

$$D_S = \{(x_m, y_m)\}_{m=1}^{M} \sim \boxed{P_S(X,Y)}$$

$$D_T = \{(x_n, y_n)\}_{n=1}^{N} \sim \boxed{P_T(X,Y)}$$

Different distributions

Find good domains for target tasks

- Landmarks: a source domain distilled for target
- Merging & reshaping datasets to domains

To reduce domain discrepancy

- Learning domain-invariant features
- Crafting perturbation functions to tune the models

$$D_S = \{(x_m, y_m)\}_{m=1}^{M} \sim \boxed{P_S(X,Y)}$$

$$D_T = \{(x_n, y_n)\}_{n=1}^{N} \sim \boxed{P_T(X,Y)}$$

Different distributions

Summary

Domain adaptation in computer vision

- Learning from Web images & Web videos
- Visual attribute recognition
- Semantic segmentation of urban scenes
- Object recognition, human activity recognition, video summarization, etc.

Acknowledgements

U. Southern California: Fei Sha

U. Texas at Austin: Kristen Grauman

U. Central Florida

Yang Zhang, Aidean Sharghi, Abdullah Jamal

Tsinghua U.: Chuang Gan

Google: Chen Sun

U. Iowa: Tianbao Yang