AIDA Course

Domain Adaptation and Generalization

Vittorio Murino, Pietro Morerio

April 8, 2022

Credits

- Tutorial by Pietro Morerio and Massimiliano Mancini
- Some slides are courtesy of Prof. Elisa Ricci and Dr. Riccardo Volpi
- Other material is referred in the corresponding slides

Outline

Session 1 - Introduction (1h)

- What is domain adaptation and why do we need it?
- The domain shift issue in vision
- Domain shift - formal statement
- Common Domain Adaptation scenarios
- Classical methods and benchmarks

Session 2 - Recent Methods (Deep learning) (1h)

- Adversarial DA
- Image translation methods
- Feature alignment/confusion
- Batchnorm-based methods
- Pseudo-labeling

Outline

Session 3 - Beyond Domain Adaptation (1h)

- Source-Free UDA
- Domain Discovery
- Continuous DA
- Predictive DA
- Validation issues in Unsupervised Domain adaptation

Session 4 - Domain generalization (1h)

- A more challenging problem
- Single source domain generalization
- Wrap-up and conclusions

Outline

Session 1 - Introduction (1h)

- What is domain adaptation and why do we need it?
- The domain shift issue in vision
- Domain shift - formal statement
- Common Domain Adaptation scenarios
- Classical methods and benchmarks

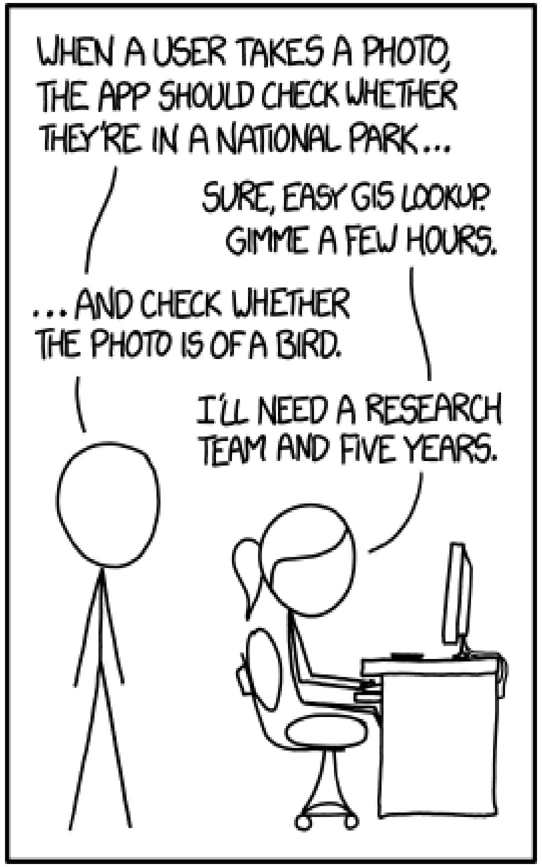
Session 2 - Recent Methods (Deep learning) (1h)

- Adversarial DA
- Image translation methods
- Feature alignment/confusion
- Batchnorm-based methods
- Pseudo-labeling (TODO)

Session 1

Introduction to Domain Adaptation

Is there a bird?



IN CS, IT CAN BE HARD TO EXPLAIN THE DIFFERENCE BETWEEN THE EASY AND THE VIRTUALLY IMPOSSIBLE.

https://xkcd.com/1425/

Is there a bird?

Is there a bird?





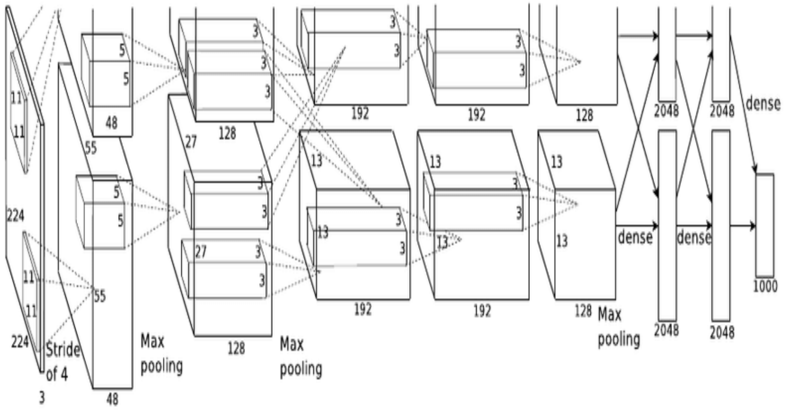


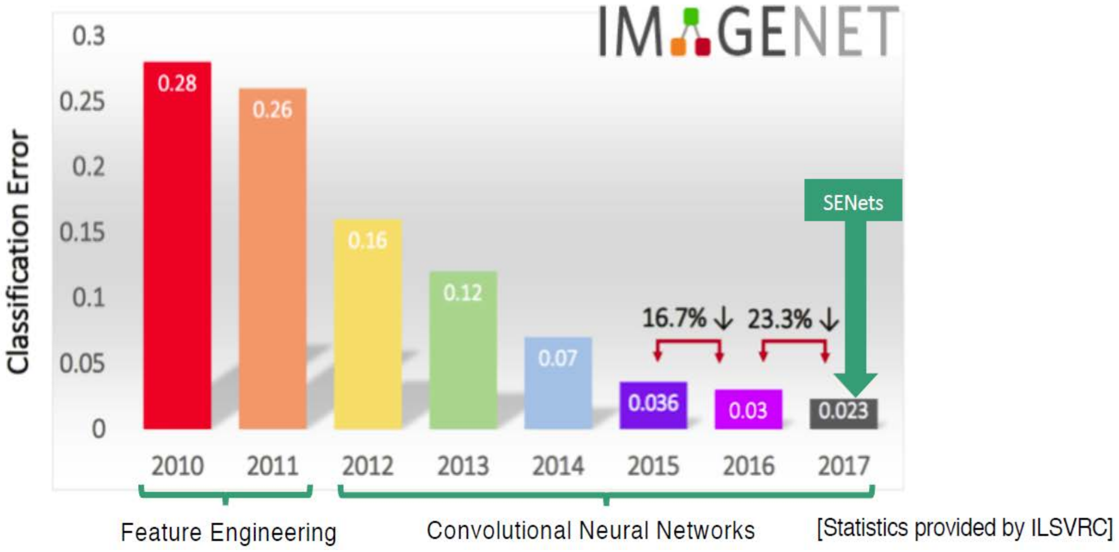





Why are we so good?

Deep Learning Revolution

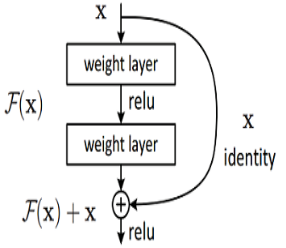


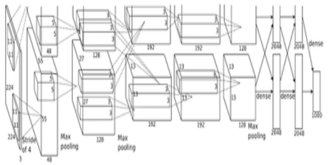


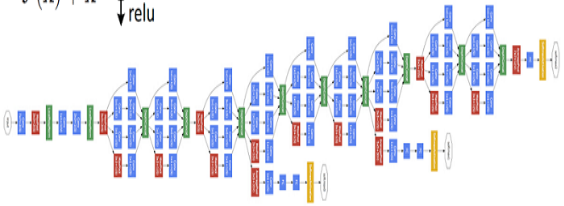
Krizhevsky, A., Sutskever, I. & Hinton, G. E. Imagenet Classification with Deep Convolutional Neural Networks. *NIPS*, 2012.

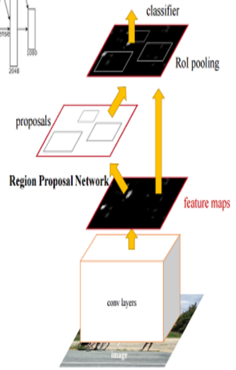
Why are we so good?

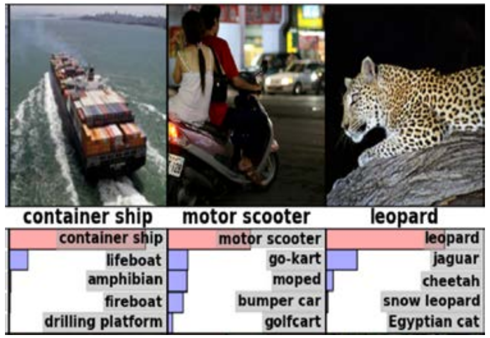
Convolutional Neural Networks

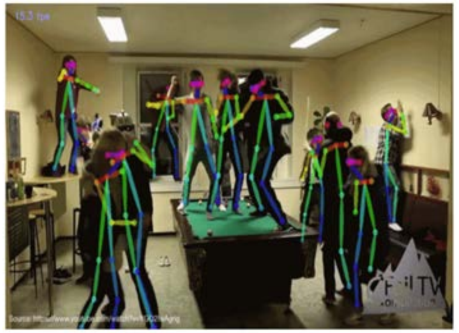


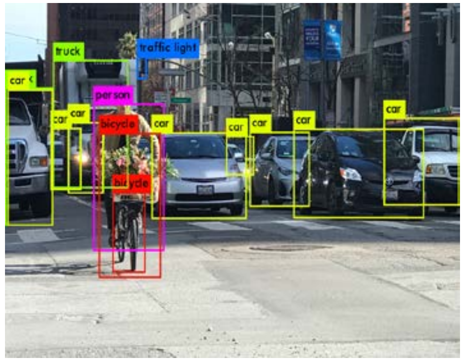




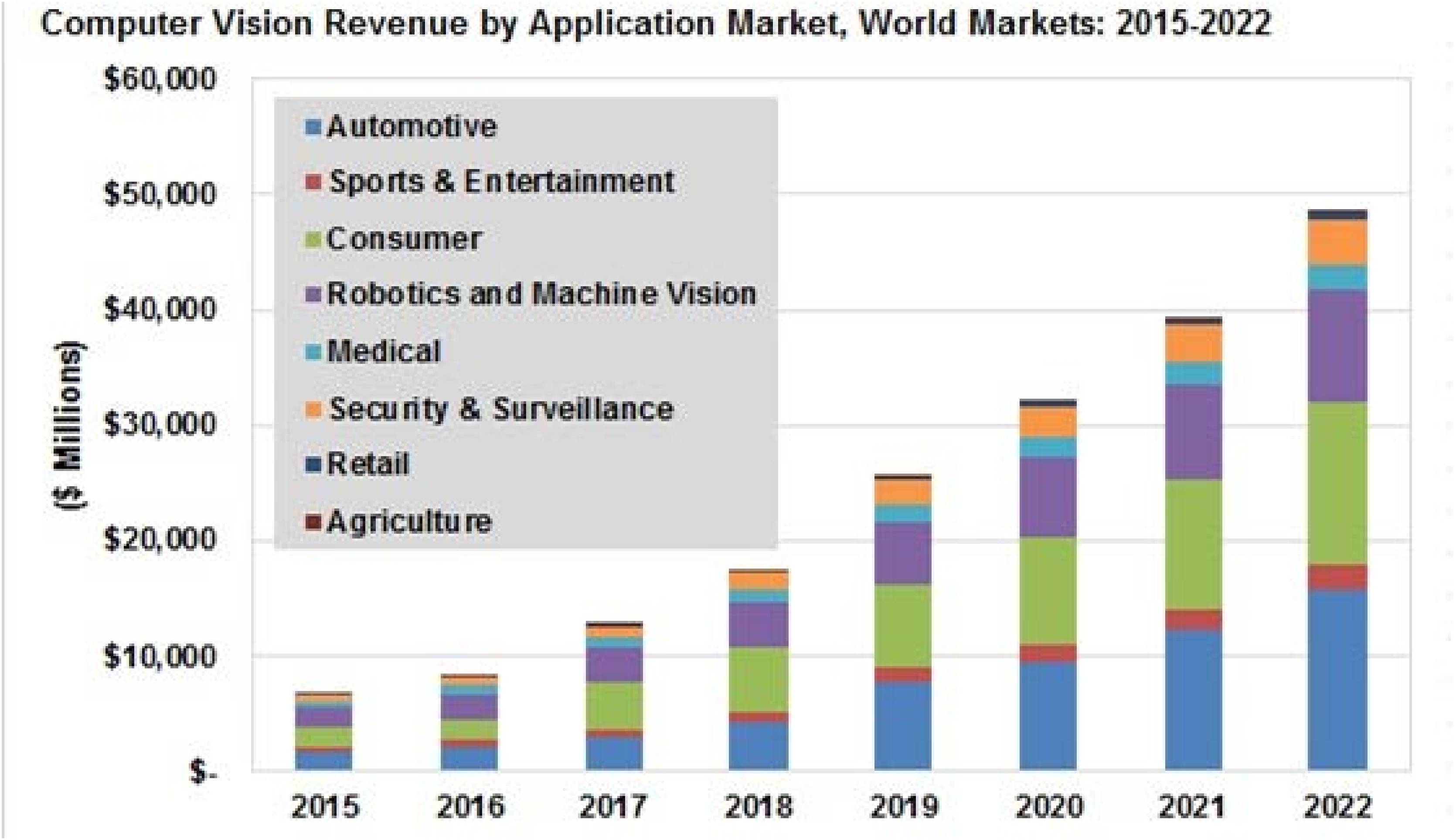






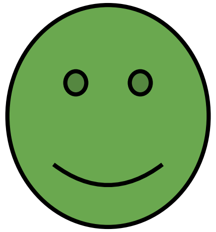


Why are we so good?

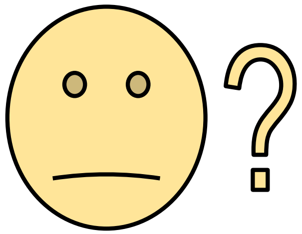


Source: [Tractica]

…back to birds

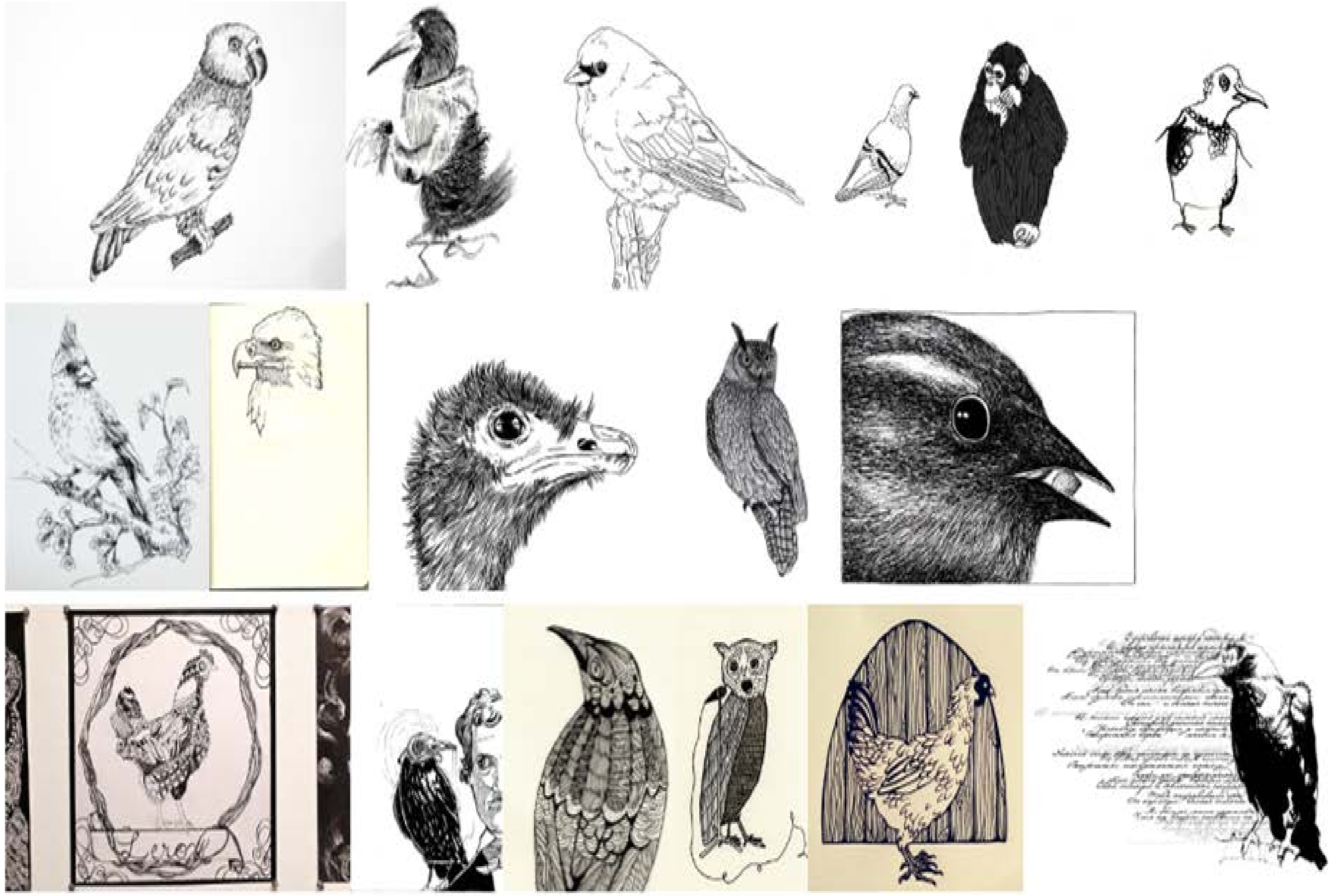


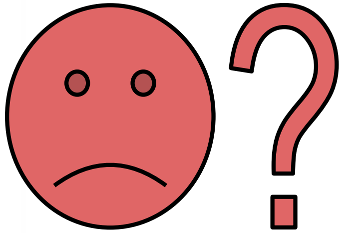
Is there a bird?



https://bam-dataset.org/

Is there a bird?





https://bam-dataset.org/

Domain Shift

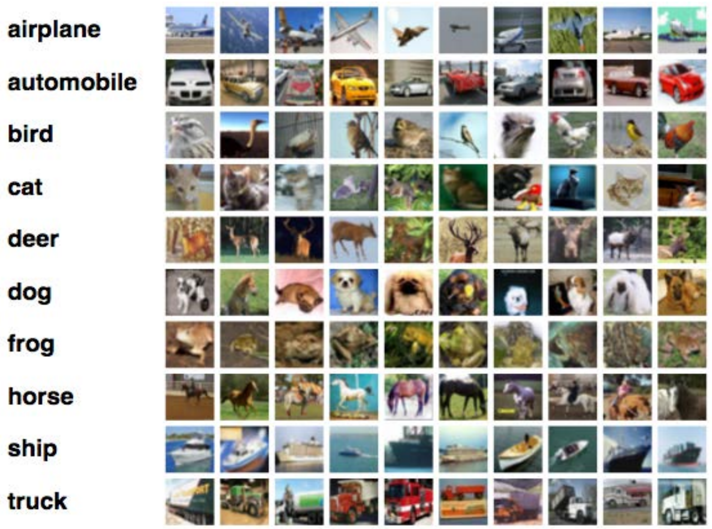
Visual appearance changes degrade the performance of visual recognition systems.

B. Kulis, K. Saenko, and T. Darrell, "What You Saw is Not What You Get: Domain Adaptation Using Asymmetric Kernel Transforms" CVPR, 2011.

Domain Shift

Visual appearance changes degrade the performance of visual recognition systems.

Accuracy: 85%



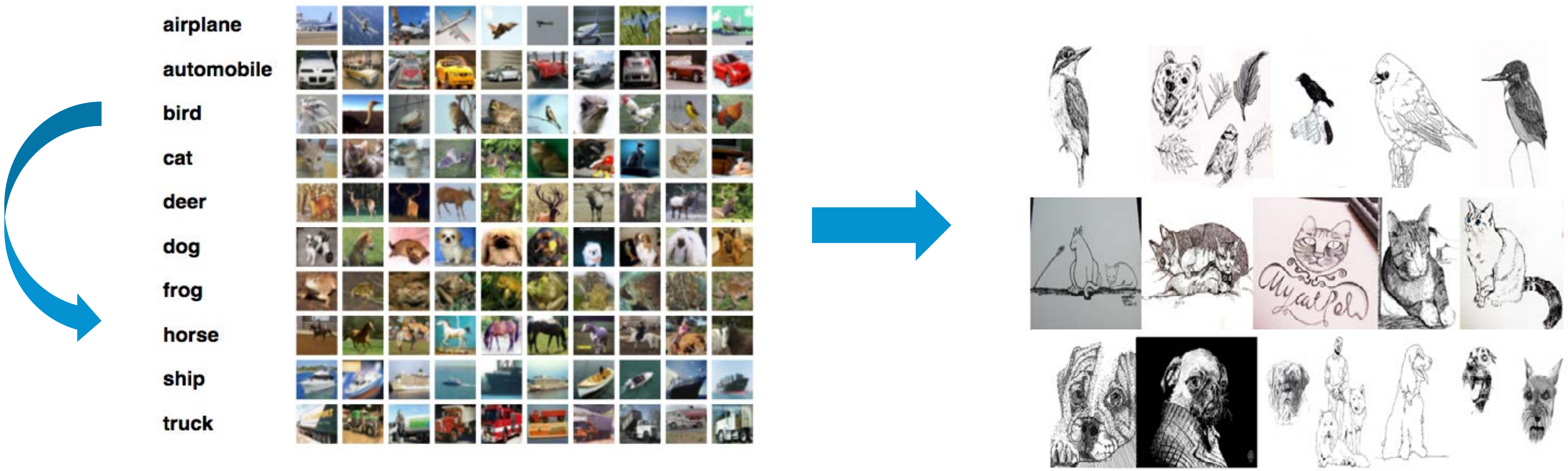
B. Kulis, K. Saenko, and T. Darrell, "What You Saw is Not What You Get: Domain Adaptation Using Asymmetric Kernel Transforms" CVPR, 2011.

Domain Shift

Visual appearance changes degrade the performance of visual recognition systems.

Accuracy: 85%

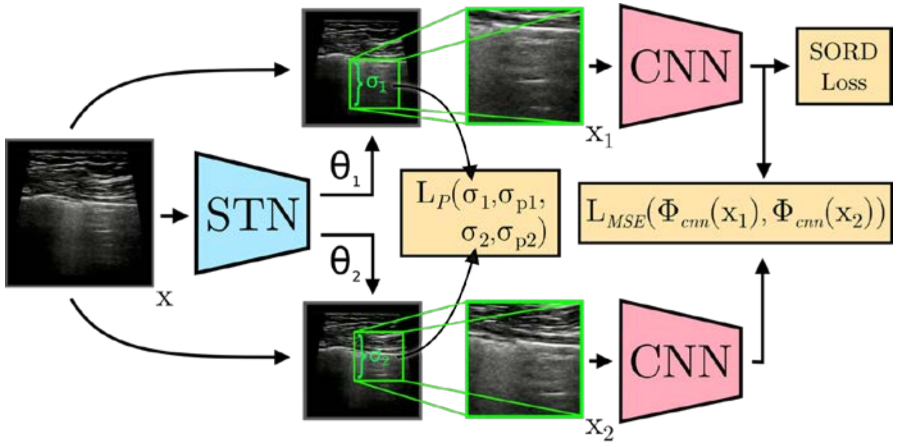
Accuracy: 55%

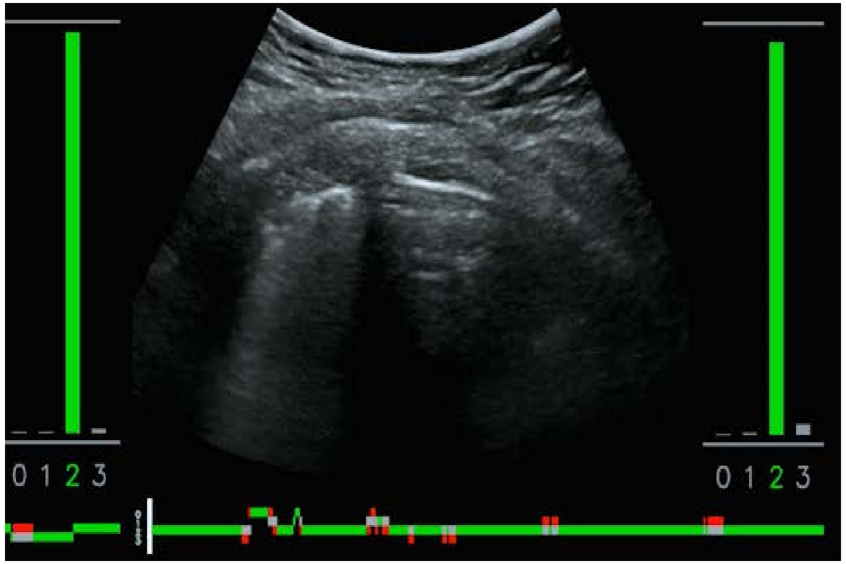


B. Kulis, K. Saenko, and T. Darrell, 'What You Saw is Not What You Get: Domain Adaptation Using Asymmetric Kernel Transforms" CVPR, 2011.

A current real-world example

- Prediction of COVID-19 markers in lung ultrasonography images.
- Data from different hospitals (and different sensors) in Italy
- Train/test data on same hospital ≥85% accuracy.
- Performance drops of even 20% considering data from different hospitals.
- Issues in having annotated data from all the hospitals.





S. Roy et al., *Deep learning for classification and localization of COVID-19 markers in point-of-care lung ultrasound*, IEEE Trans. on Medical Imaging 2020.

Domain Shift: why do we care about?

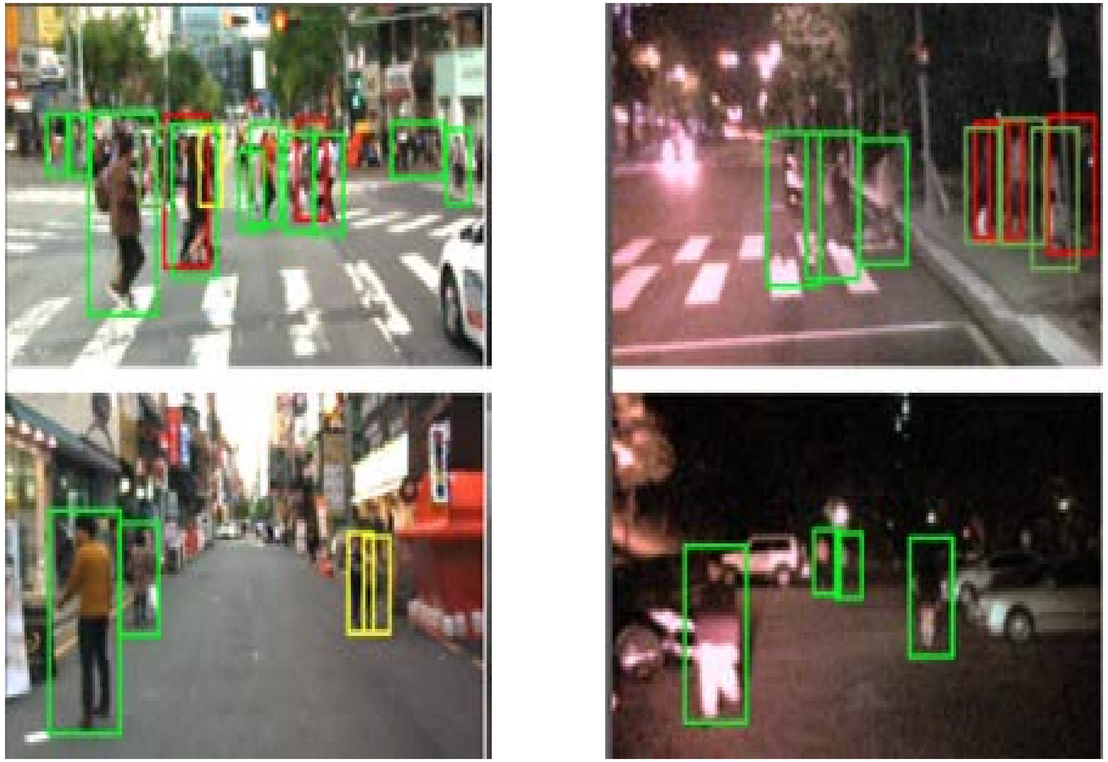
Domain Shift: why do we care about?

Appearance changes due to seasonal and time changes.

Z. Chen et al., "Deep Learning Features at Scale for Visual Place Recognition". ICRA 2017.

Domain Shift: why do we care about?

Appearance changes due to illumination conditions



S. Hwang et al., "Multispectral Pedestrian Detection: Benchmark Dataset and Baseline", CVPR 2015.

Domain Shift: why do we care about?

Appearance changes due to different sensors.

S. Hwang et al., "Multispectral Pedestrian Detection: Benchmark Dataset and Baseline", CVPR 2015.

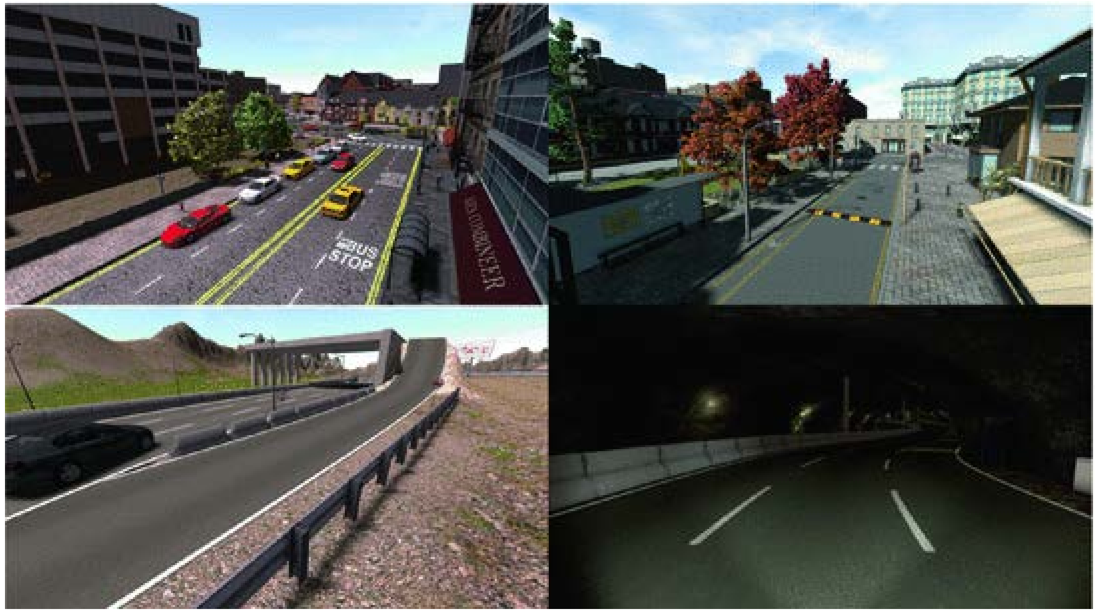
Domain Shift: why do we care about?

Overcoming costly/unfeasible data collection

https://www.zdnet.com/article/robots-to-the-rescue-searching-for-survivors-checking-on-structural-damage-in-japan/

Domain Shift: why do we care about?

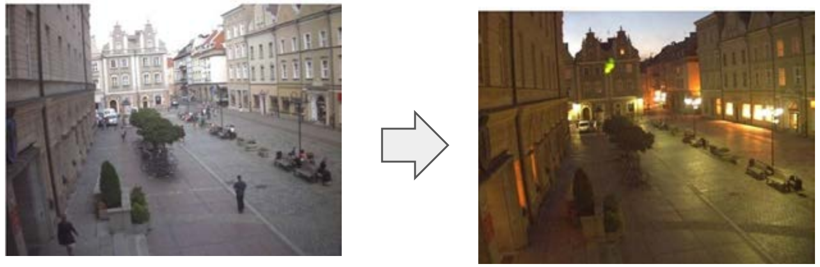
Use of synthetic data



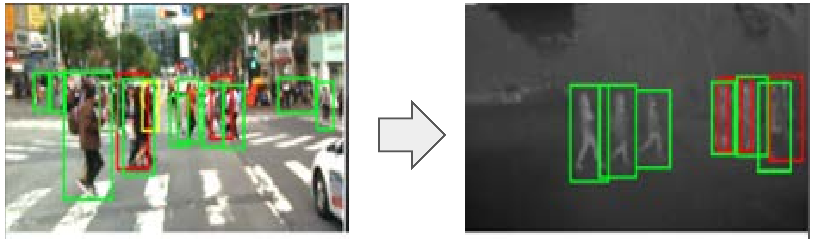
http://synthia-dataset.net/

Domain Shift is *ubiquitous* …

Time & Environmental Changes



Different Modalities



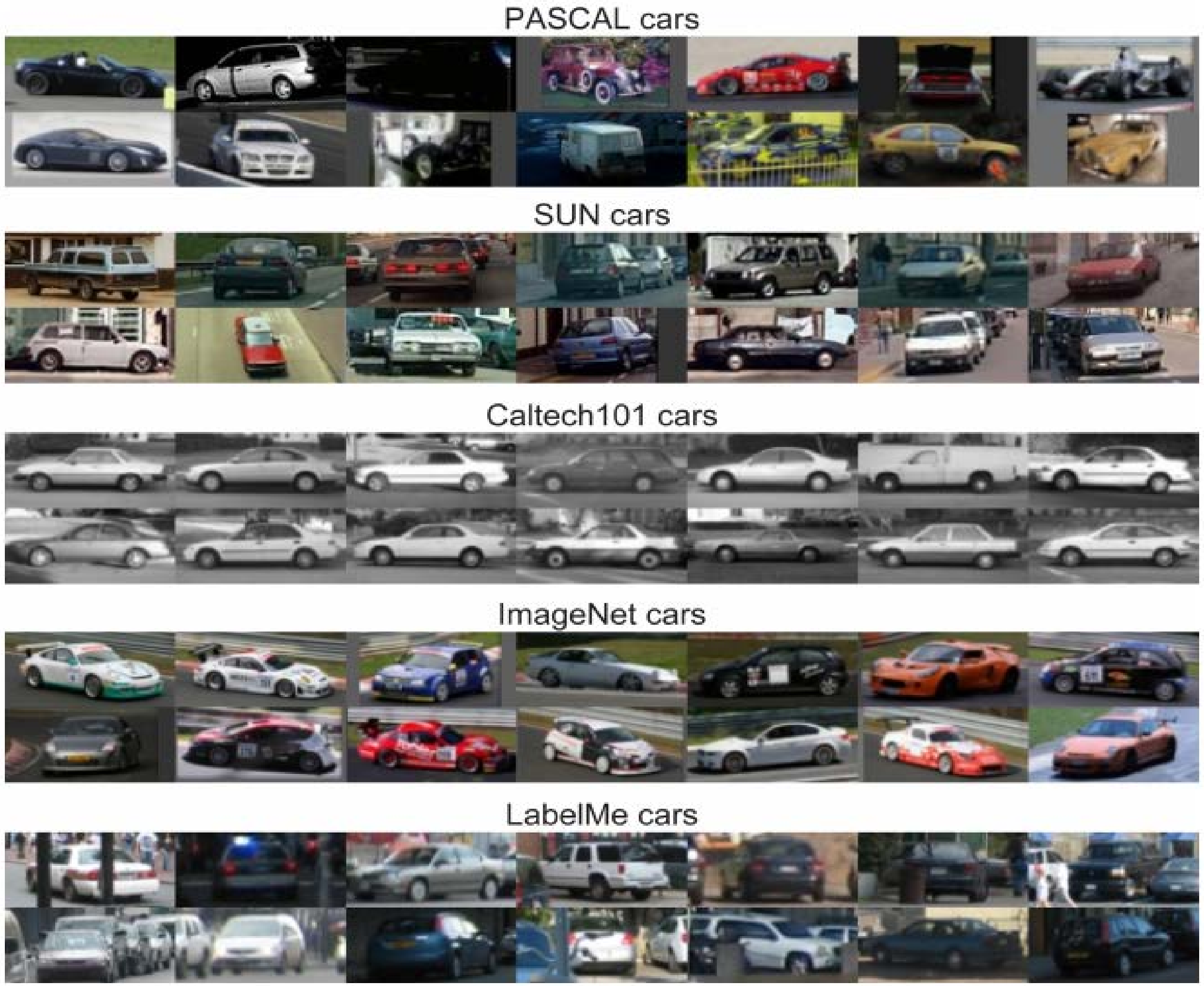
Synthetic to Real Images

CAD to Real Images

Domain shift and dataset bias

Caltech101 ___	Tiny ___	LabelMe ___	15 Scenes ___
MSRC ___	Corel ___	COIL–100 ___	Caltech256 ___
UIUC ___	PASCAL 07 ___	ImageNet ___	SUN09 ___

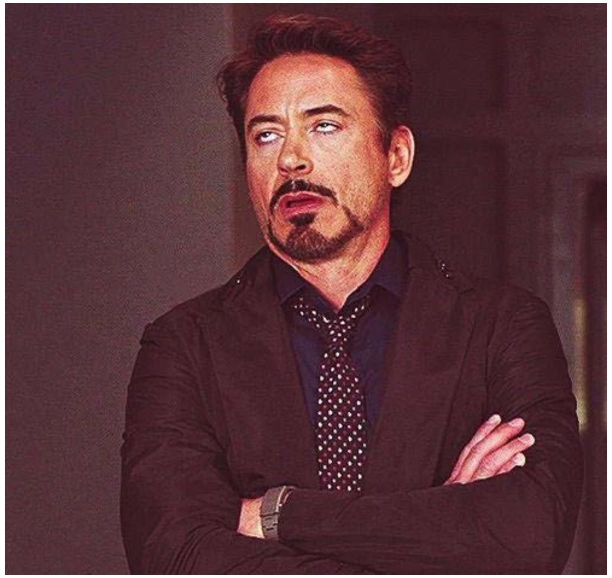
Figure 1. Name That Dataset: Given three images from twelve popular object recognition datasets, can you match the images with the dataset? (answer key below)

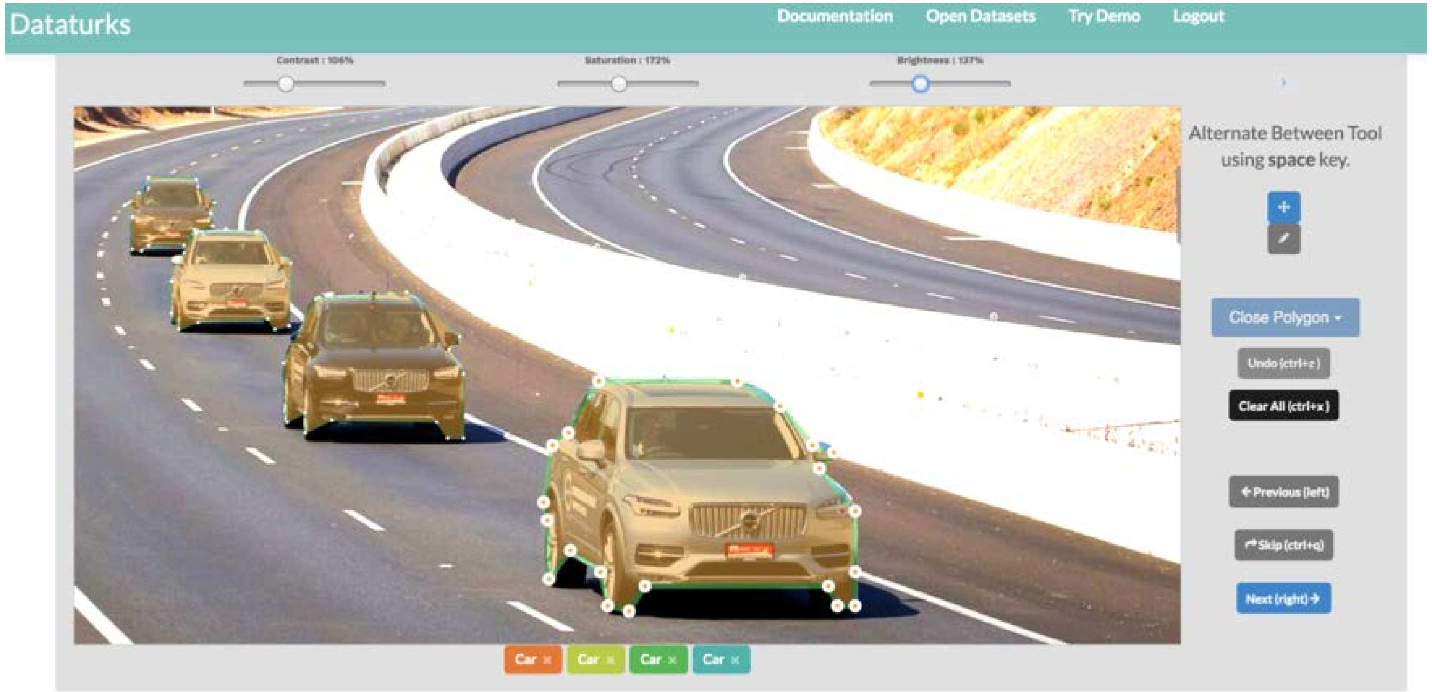


Torralba and Efros ,"Unbiased look at dataset bias, CVPR 2011

Domain Shift: how do we solve it?

Annotate target data …

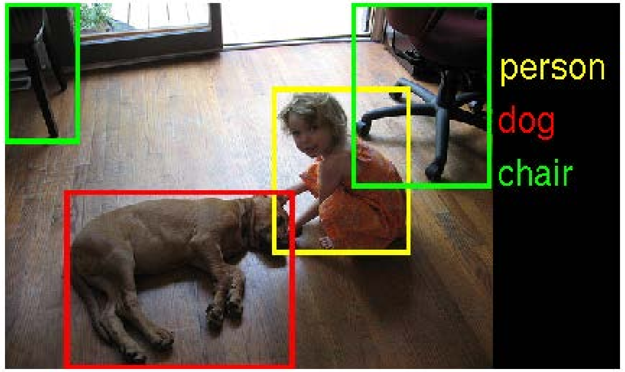




[Source image](#)

Domain Shift & Data Annotation

Data collection is costly



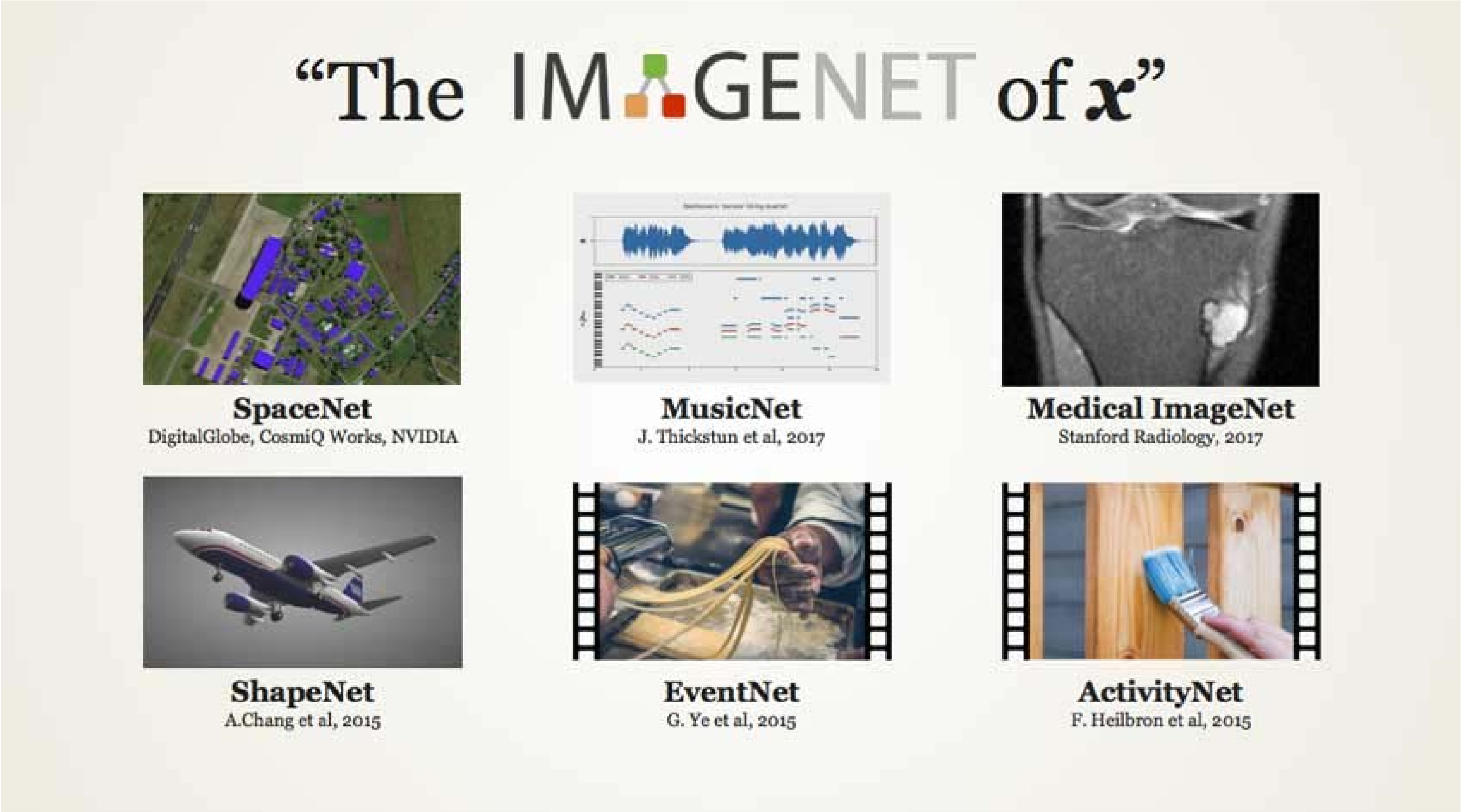
- about hand-annotated 14M images
- about 1M with bounding-boxes
- more than 20,000 categories

J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li and L. Fei-Fei, ImageNet: A Large-Scale Hierarchical Image Database. CVPR) 2009.

PAVIS

Domain Shift & Data Annotation

We cannot annotate everything



Slide credit: Fei-Fei Li

Domain Shift & Data Annotation

Sometimes collecting data is impossible

https://www.zdnet.com/article/robots-to-the-rescue-searching-for-survivors-checking-on-structural-damage-in-japan/

Domain Shift

A bit of notation

$\mathcal{X}$	Input space
$\mathcal{Y}$	Output space
$X \in \mathcal{X}$	Input variable
$Y \in \mathcal{Y}$	Output variable

$D = \{\mathcal{X}, P(X)\}$	Domain
$T = \{\mathcal{Y}, P(Y\|X)\}$	Task

F. Orabona & T. Tommasi. Tutorial on Domain Adaptation and Transfer Learning. In ECCV 2014.

The Domain Adaptation (DA) Problem

Source Domain

$$D^s = \{\mathcal{X}^s, P(X^s)\}$$
$$T^s = \{\mathcal{Y}^s, P(Y^s|X^s)\}$$

Target Domain

$$D^t = \{\mathcal{X}^t, P(X^t)\}$$
$$T^t = \{\mathcal{Y}^t, P(Y^t|X^t)\}$$

DA problem

$$D^s \neq D^t \qquad T^s = T^t$$

F. Orabona & T. Tommasi. Tutorial on Domain Adaptation and Transfer Learning. In ECCV 2014.

The Domain Adaptation (DA) Problem

Source Domain

$$D^s = \{\mathcal{X}^s, P(X^s)\}$$
$$T^s = \{\mathcal{Y}^s, P(Y^s|X^s)\}$$

Target Domain

$$D^t = \{\mathcal{X}^t, P(X^t)\}$$
$$T^t = \{\mathcal{Y}^t, P(Y^t|X^t)\}$$

DA problem

$$\mathcal{X}^s \neq \mathcal{X}^t$$
$$P(X^s) \neq P(X^t)$$

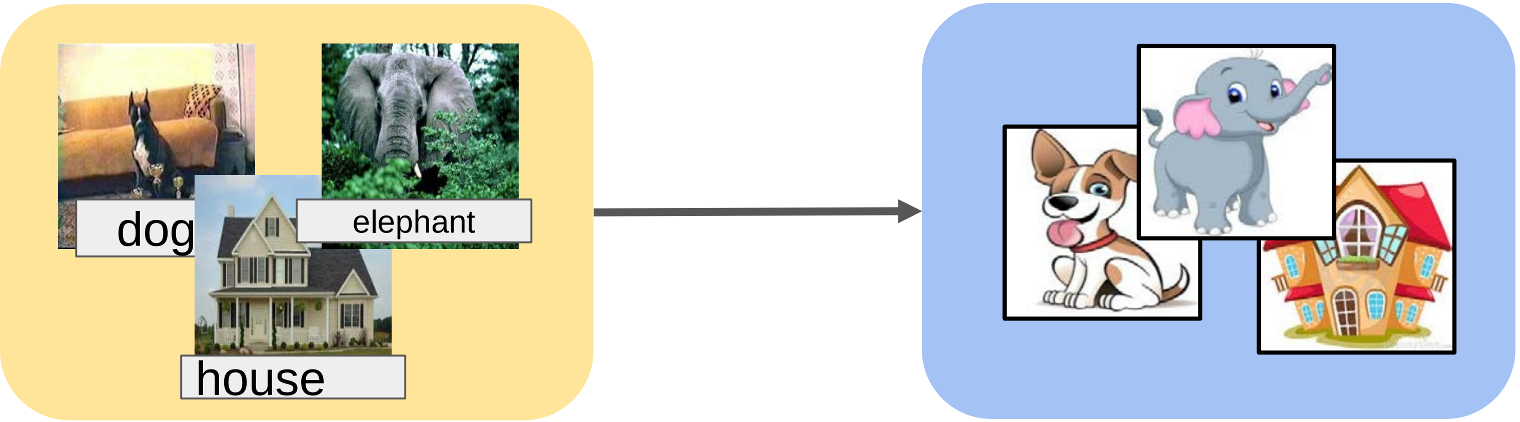
$$\mathcal{Y}^s = \mathcal{Y}^t$$

PAVIS

DA Landscape: unsupervised DA

Labelled
Source Domain

Unlabelled
Target Domain

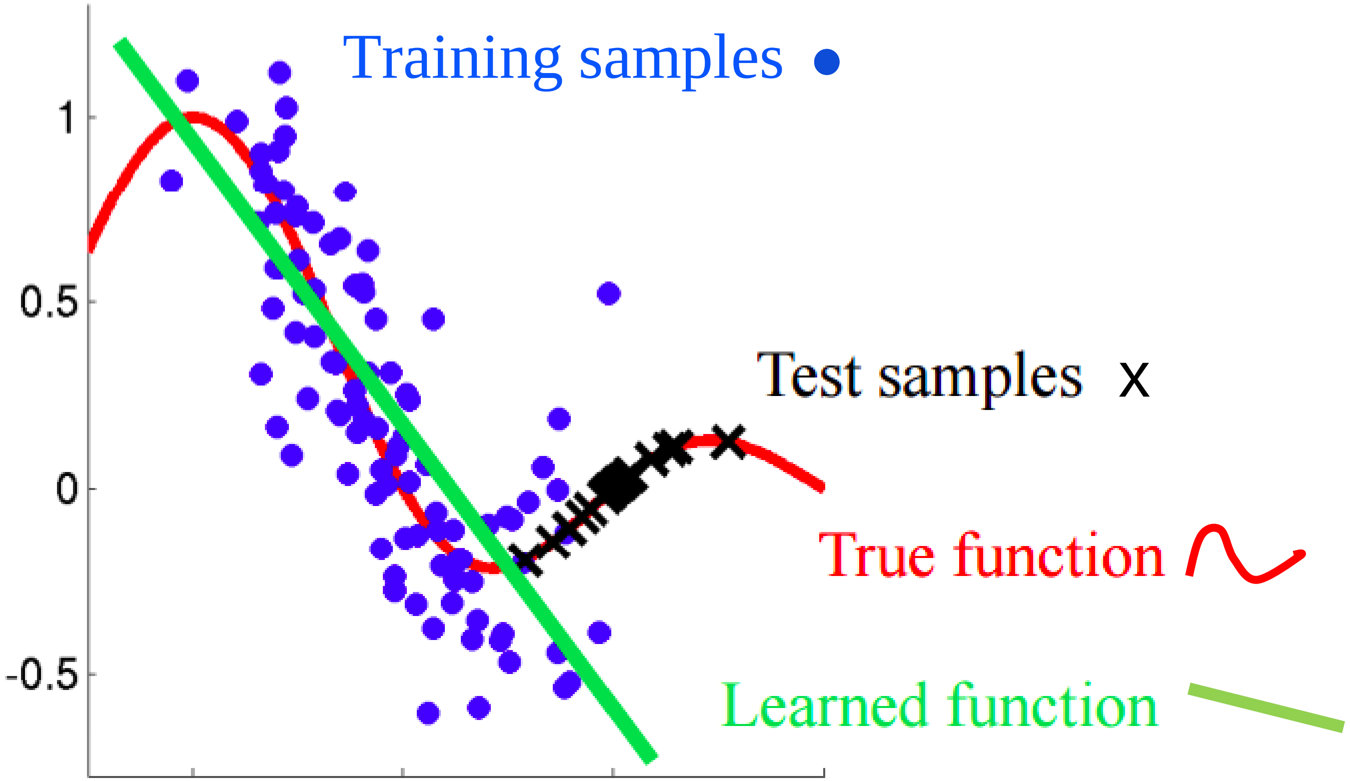


$$P(X^s) \neq P(X^t)$$
$$\mathcal{Y}^s = \mathcal{Y}^t$$

Real images vs Cartoons
{elephant, dog, house}

http://www.eecs.qmul.ac.uk/~dl307/project_iccv2017

Covariate Shift Assumption

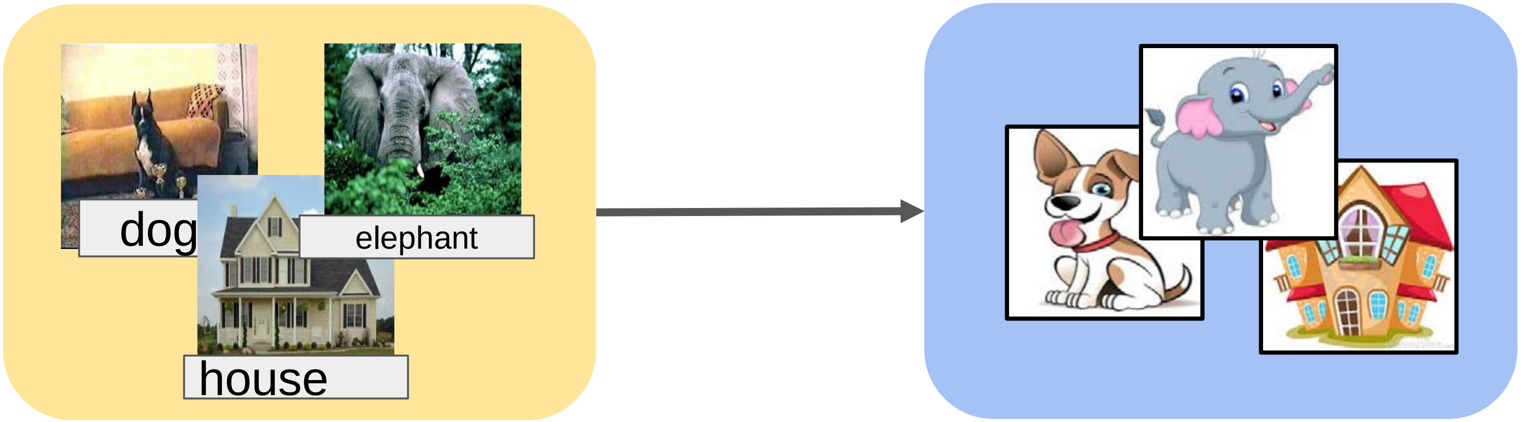


DA Scenarios

DA Landscape: unsupervised DA

Labelled
Source Domain

Unlabelled
Target Domain



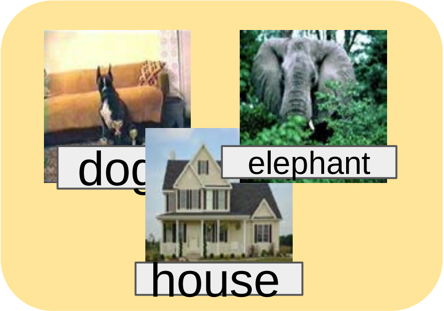
$$P(X^s) \neq P(X^t)$$
$$\mathcal{Y}^s = \mathcal{Y}^t$$

Real images vs
Cartoons
{elephant, dog, house}

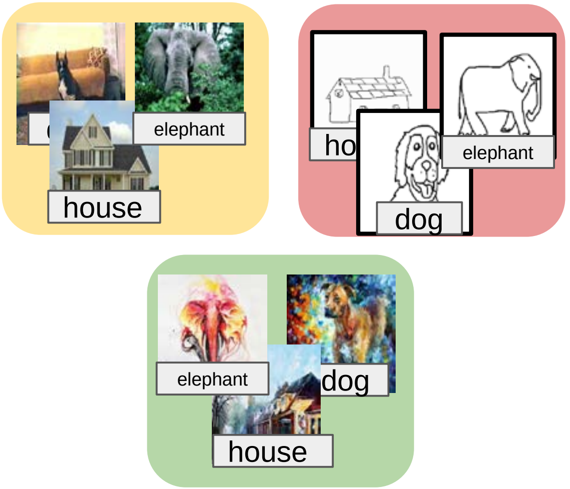
http://www.eecs.qmul.ac.uk/~dl307/project_iccv2017

DA Landscape: number of *Source* domains

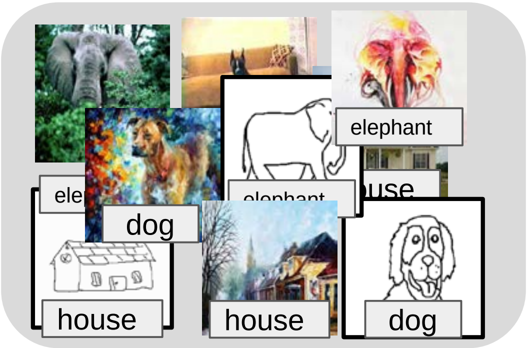
Single



Multi

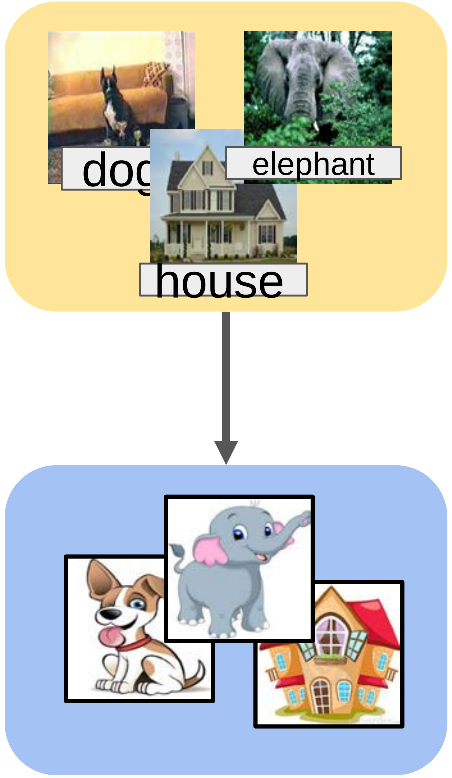


Mixed

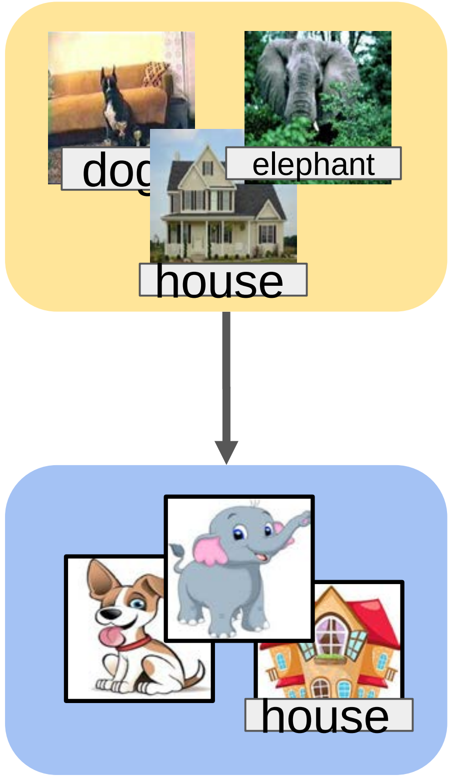


DA Landscape: the target labels

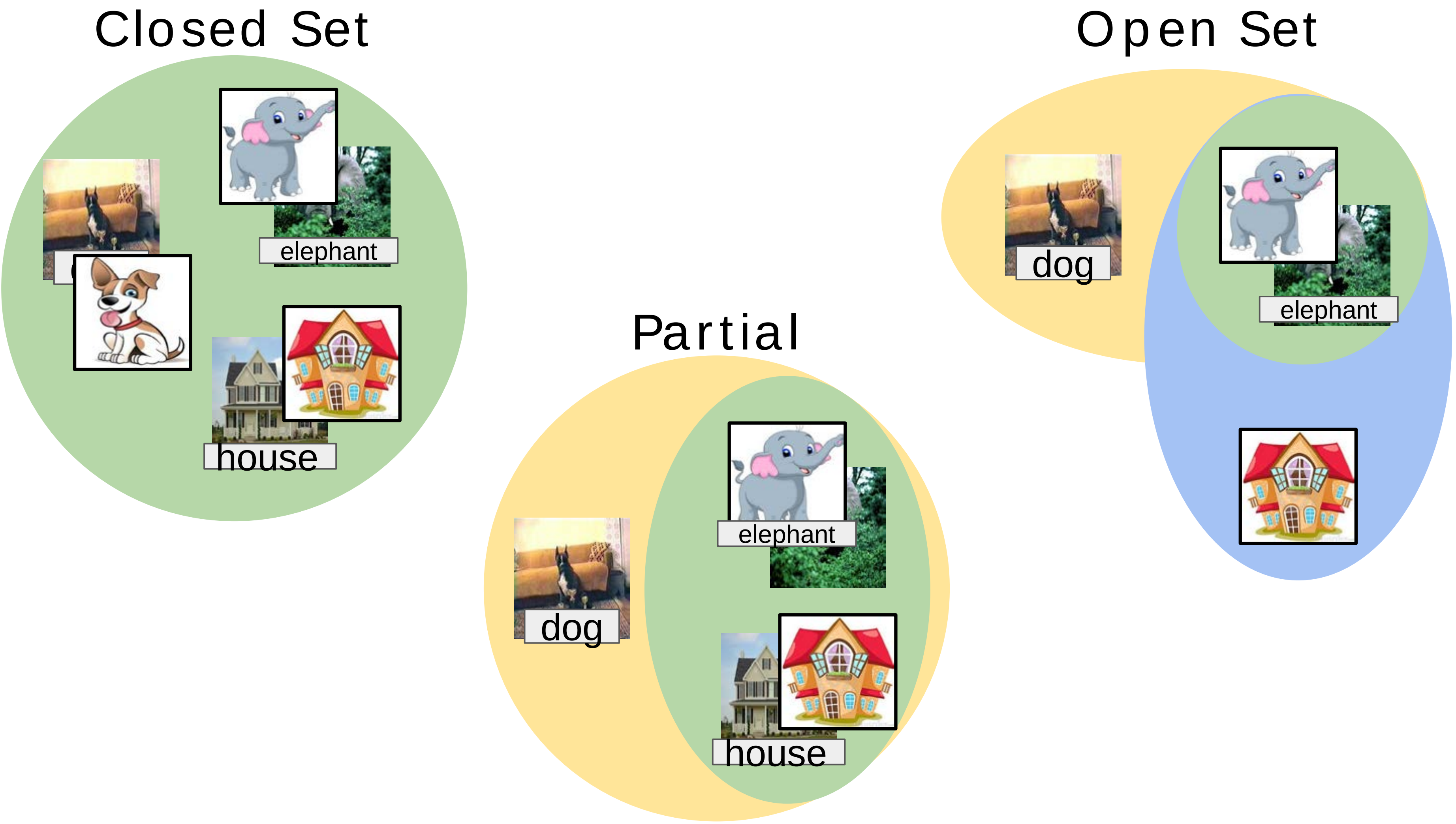
Unsupervised



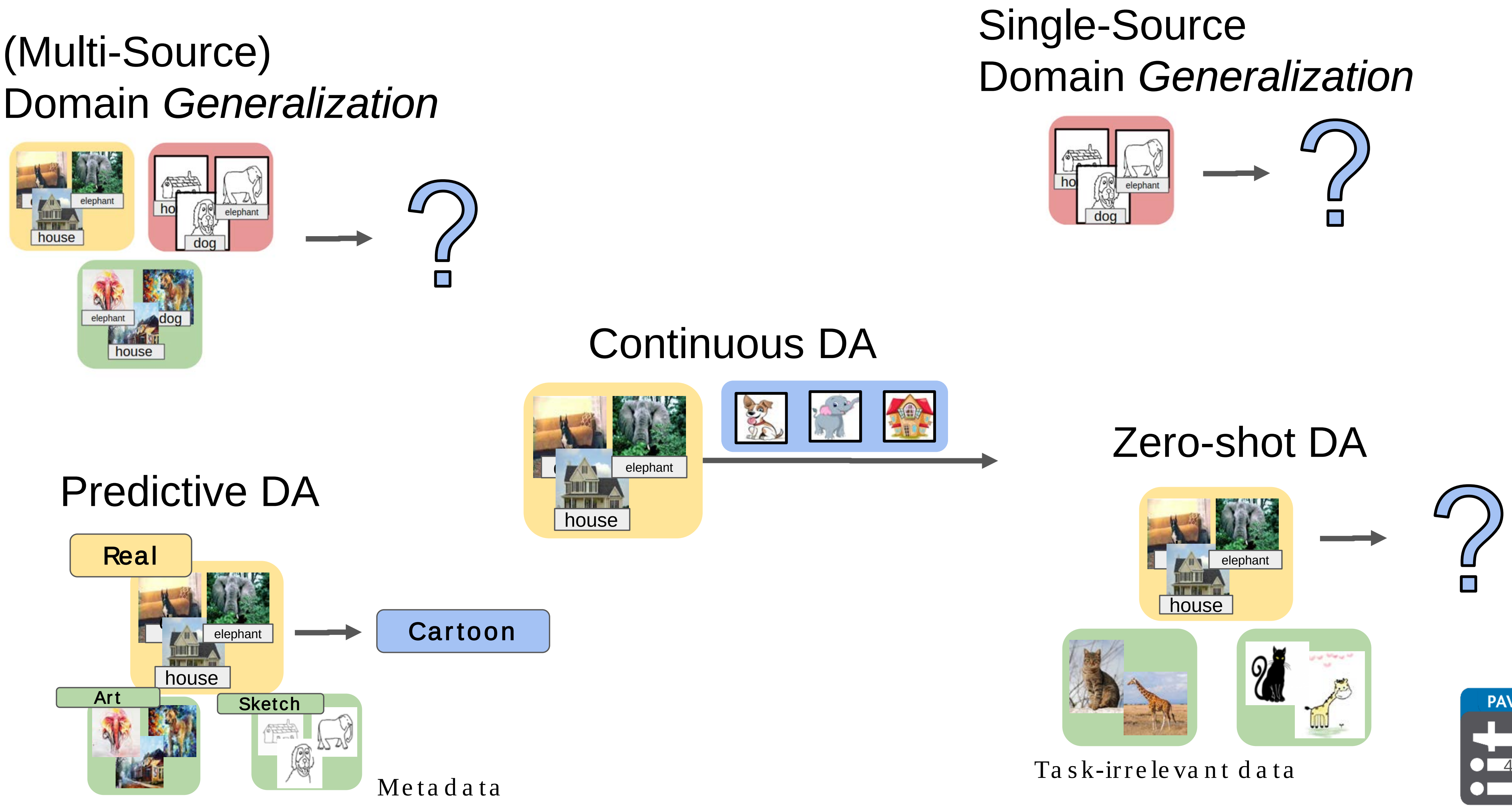
Semi-Supervised



DA Landscape: the label space



DA Scenarios: <u>without</u> target data!



A bit of history ...

Benchmarks: Office31

Saenko, K., et al. "Adapting visual category models to new domains." In ECCV 2010.
Gong, B., et al. "Geodesic flow kernel for unsupervised domain adaptation." In CVPR 2012.

A bit of history …
DA before the Deep Learning Revolution

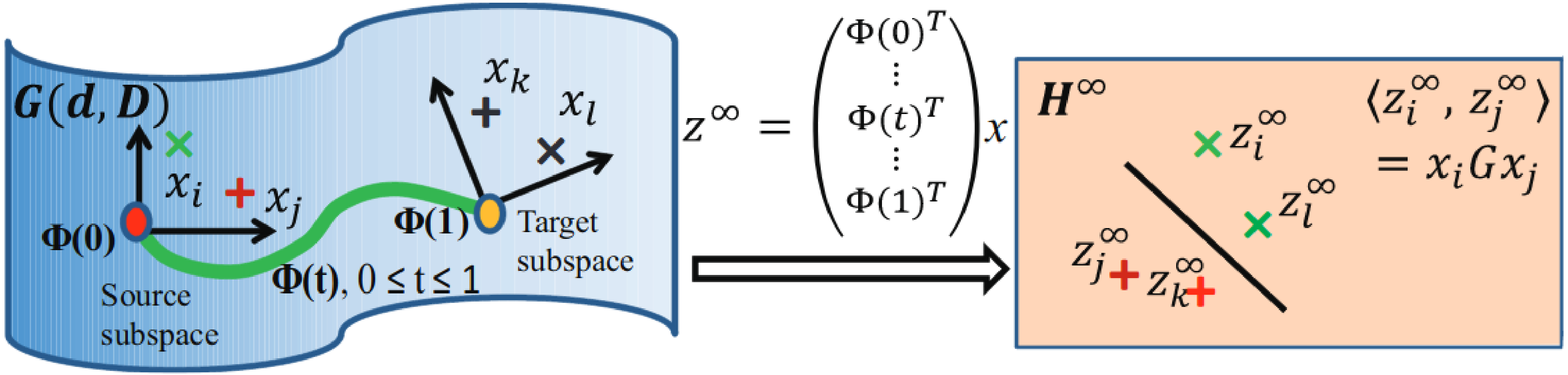
Traditional approaches based on hand-crafted features:

- Instance re-weighting (boosting): utilize training instances from the source domain which are more similar to target samples to build target model [Dai et al, ICML 2007]
- Parameter transfer: assume some parameters shared between source and target model [Bruzzone et al., TPAMI 2010]
- Feature alignment: identifying similar feature subspaces that can be used to align source and target domains [Fernando et al., ICCV 2013] [Sun et al. AAAI 2016]

PAVIS

A bit of history ...

DA before the Deep Learning Revolution : Geodesic Flow Kernel (GFK)



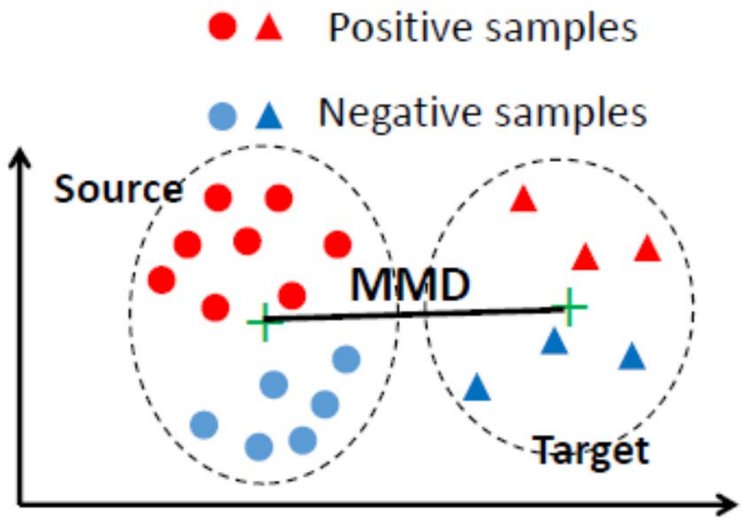
The geodesic flow kernel tackles domain shift by integrating an infinite number of subspaces that characterize changes in geometric and statistical properties from the source to the target domain

Gong, B., et al. "Geodesic flow kernel for unsupervised domain adaptation". CVPR 2012.

A bit of history ...
DA before the Deep Learning Revolution : Maximum Mean Discrepancy

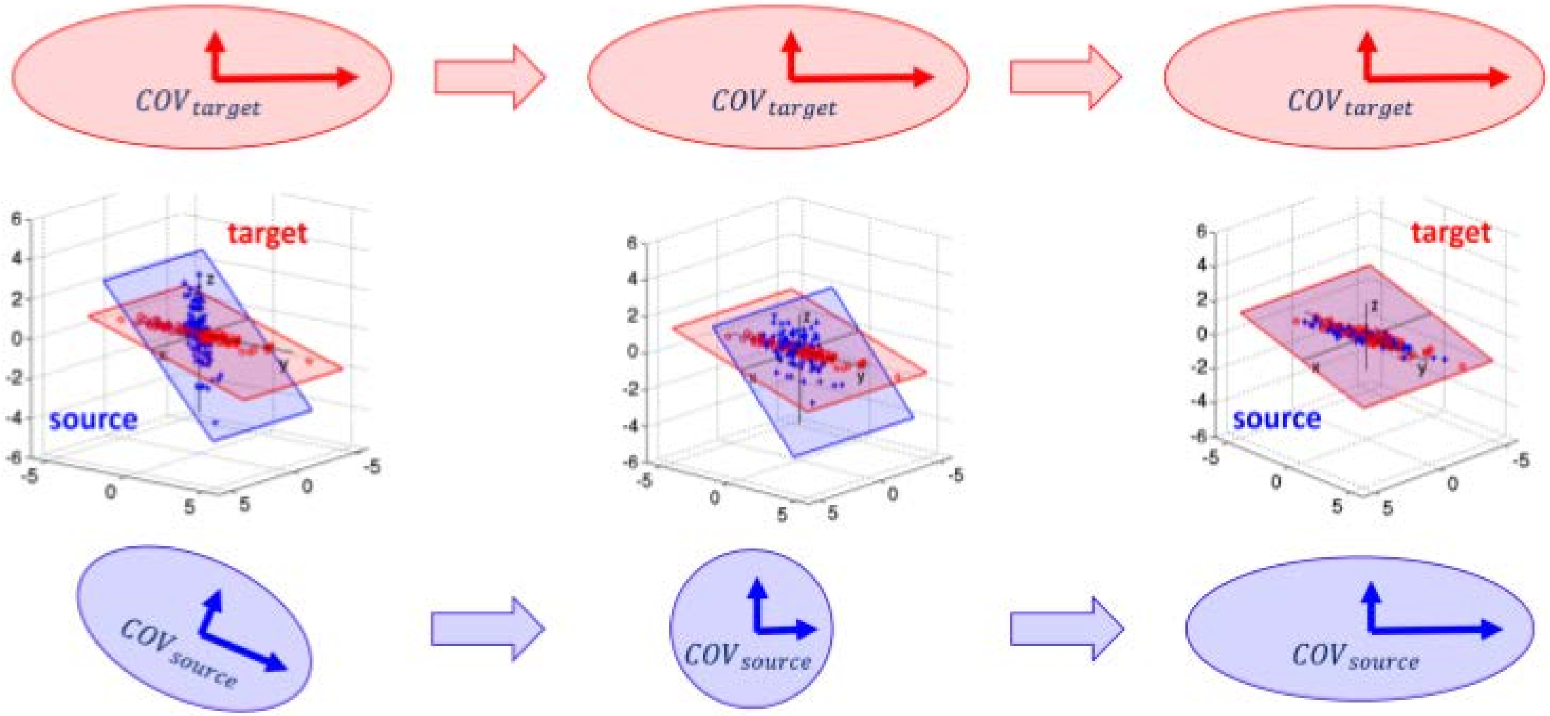
Distance between embeddings of the probability distributions in a reproducing kernel Hilbert space.

$$\mathrm{MMD}^2 = \left\| \frac{1}{n^s} \sum_{i=1}^{n^s} \phi(x_i^s) - \frac{1}{n^t} \sum_{i=1}^{n^t} \phi(x_i^t) \right\|^2$$



Gretton, A., et al. . "A kernel method for the two-sample-problem". In NIPS. 2007.
F. Orabona & T. Tommasi. Tutorial on Domain Adaptation and Transfer Learning. In ECCV 2014.

A bit of history…
DA before the Deep Learning Revolution :
CORrelation ALignment, CORAL



Sun, B., et al. "Correlation alignment for unsupervised domain adaptation". In Domain Adaptation in Computer Vision Applications (pp. 153-171), 2017..

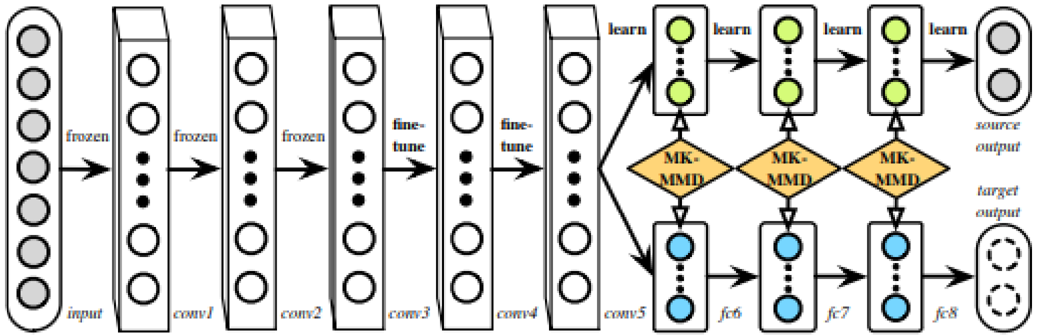
A bit of history ...
Deep Domain Adaptation

- Three main categories (not necessarily ultimate):
  - <u>Discrepancy-based</u>: fine-tuning the deep network with labeled or unlabeled target data to narrow down the domain shift
  - <u>Adversarial-based</u>: using domain discriminators to encourage domain confusion through an adversarial objective
  - <u>Reconstruction-based</u>: using the data reconstruction as an auxiliary task to ensure feature invariance

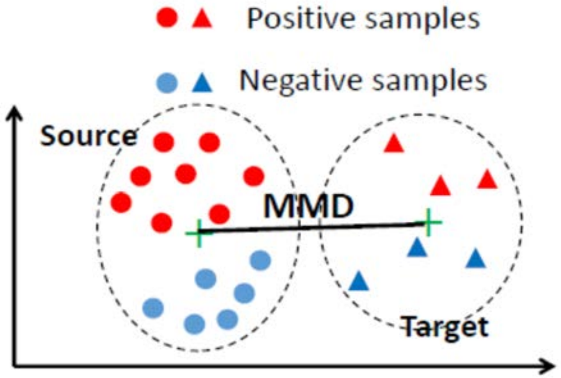
G. Csurka, "Domain adaptation for visual applications: A comprehensive survey". arXiv preprint arXiv:1702.05374, 2017

A bit of history …
Deep DA : Maximum Mean Discrepancy

- Distance between embeddings of the probability distributions in a reproducing kernel Hilbert space.
- Idea: perform a re-mapping of the feature representation in a common embedding so that such bias is removed. Such re-mapping is made at different layers and is induced by a MMD distance over a kernel function



Domain Adaptive Network, DAN

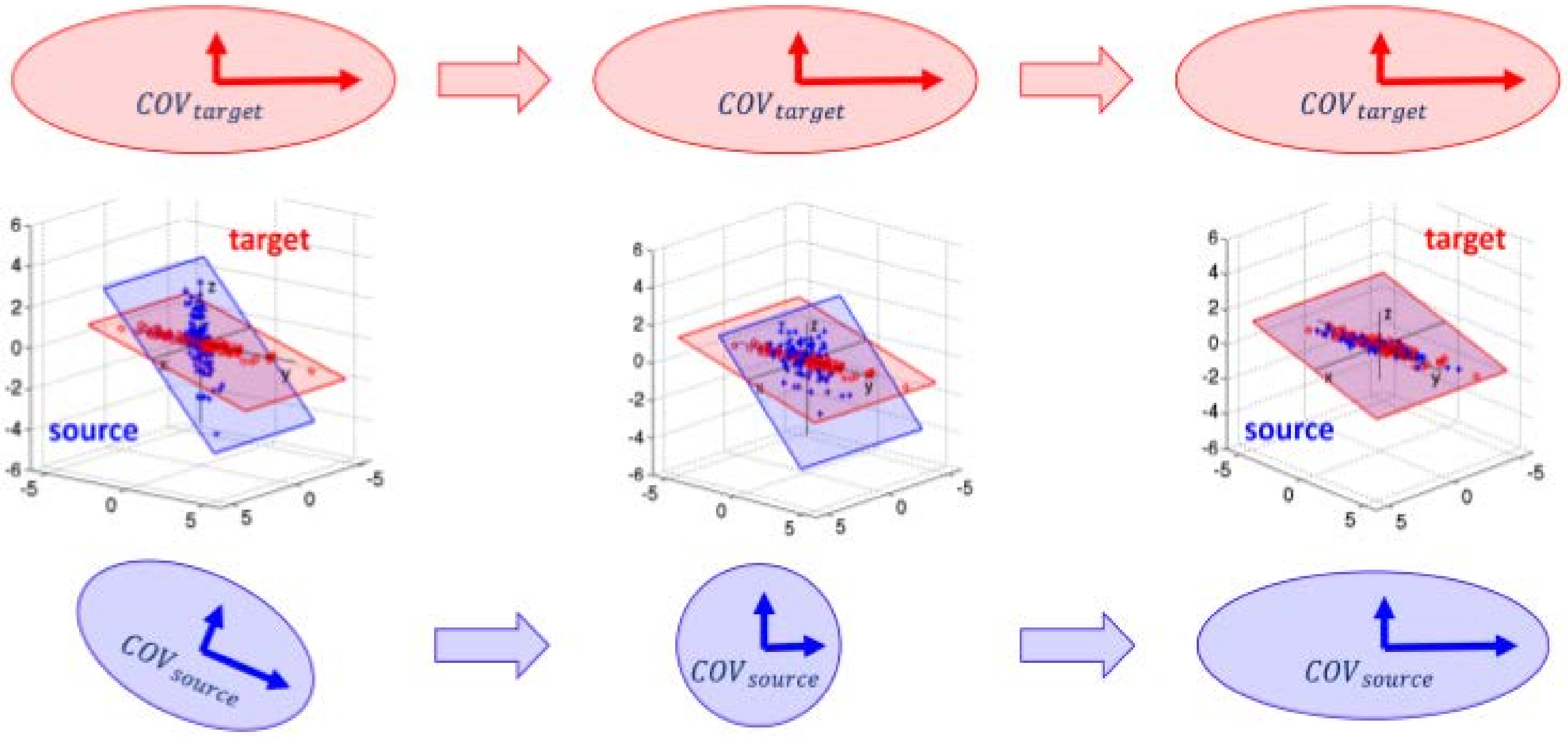


$$\mathrm{MMD}^2 = \left\| \frac{1}{n^s} \sum_{i=1}^{n^s} \phi(x_i^s) - \frac{1}{n^t} \sum_{i=1}^{n^t} \phi(x_i^t) \right\|^2$$

Gretton, A. et al.. "A kernel method for the two-sample-problem". In NIPS. 2007.
Long, M. et al., "Learning Transferable Features with Deep Adaptation Networks". ICML 2015.

PAVIS

A bit of history ...
Deep DA : deep CORAL



Sun, B., et al. "Correlation alignment for unsupervised domain adaptation". In Domain Adaptation in Computer Vision Applications, 2017.

Office31and Office-Caltech results (2015)

Table 1. Accuracy on *Office-31* dataset with standard unsupervised adaptation protocol (Gong et al., 2013).

Method	$A \rightarrow W$	$D \rightarrow W$	$W \rightarrow D$	$A \rightarrow D$	$D \rightarrow A$	$W \rightarrow A$	Average
TCA	21.5 ± 0.0	50.1 ± 0.0	58.4 ± 0.0	11.4 ± 0.0	8.0 ± 0.0	14.6 ± 0.0	27.3
GFK	19.7 ± 0.0	49.7 ± 0.0	63.1 ± 0.0	10.6 ± 0.0	7.9 ± 0.0	15.8 ± 0.0	27.8

Table 2. Accuracy on *Office-10* + *Caltech-10* dataset with standard unsupervised adaptation protocol (Gong et al., 2013).

Method	$A \rightarrow C$	$W \rightarrow C$	$D \rightarrow C$	$C \rightarrow A$	$C \rightarrow W$	$C \rightarrow D$	Average
TCA	42.7 ± 0.0	34.1 ± 0.0	35.4 ± 0.0	54.7 ± 0.0	50.5 ± 0.0	50.3 ± 0.0	44.6
GFK	41.4 ± 0.0	26.4 ± 0.0	36.4 ± 0.0	56.2 ± 0.0	43.7 ± 0.0	42.0 ± 0.0	41.0

Long, M., et al. "Learning Transferable Features with Deep Adaptation Networks". ICML 2015..

Office31 and Office-Caltech results (2015)

Table 1. Accuracy on *Office-31* dataset with standard unsupervised adaptation protocol (Gong et al., 2013).

Method	A → W	D → W	W → D	A → D	D → A	W → A	Average
TCA	21.5 ± 0.0	50.1 ± 0.0	58.4 ± 0.0	11.4 ± 0.0	8.0 ± 0.0	14.6 ± 0.0	27.3
GFK	19.7 ± 0.0	49.7 ± 0.0	63.1 ± 0.0	10.6 ± 0.0	7.9 ± 0.0	15.8 ± 0.0	27.8
CNN	61.6 ± 0.5	95.4 ± 0.3	99.0 ± 0.2	63.8 ± 0.5	51.1 ± 0.6	49.8 ± 0.4	70.1

Table 2. Accuracy on *Office-10* + *Caltech-10* dataset with standard unsupervised adaptation protocol (Gong et al., 2013).

Method	A → C	W → C	D → C	C → A	C → W	C → D	Average
TCA	42.7 ± 0.0	34.1 ± 0.0	35.4 ± 0.0	54.7 ± 0.0	50.5 ± 0.0	50.3 ± 0.0	44.6
GFK	41.4 ± 0.0	26.4 ± 0.0	36.4 ± 0.0	56.2 ± 0.0	43.7 ± 0.0	42.0 ± 0.0	41.0
CNN	83.8 ± 0.3	76.1 ± 0.5	80.8 ± 0.4	91.1 ± 0.2	83.1 ± 0.3	89.0 ± 0.3	84.0

Long, M., et al. "Learning Transferable Features with Deep Adaptation Networks". ICML 2015..

Office31 and Office-Caltech results (2015)

Table 1. Accuracy on *Office-31* dataset with standard unsupervised adaptation protocol (Gong et al., 2013).

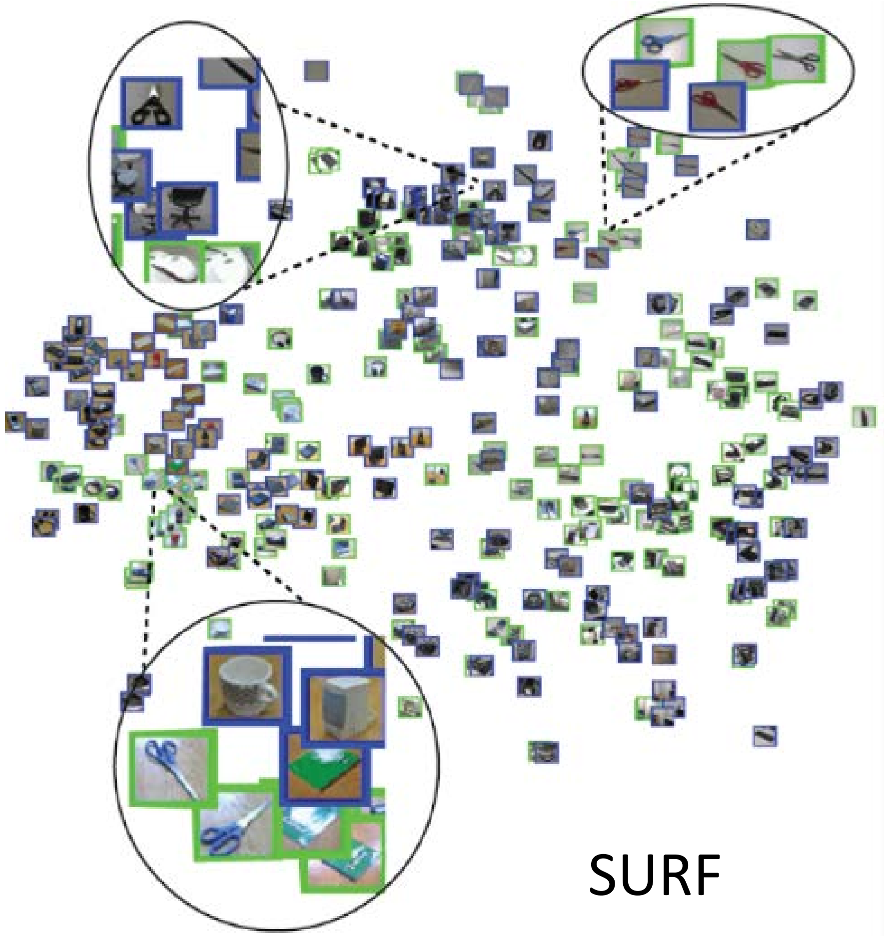
Method	A → W	D → W	W → D	A → D	D → A	W → A	Average
TCA	21.5 ± 0.0	50.1 ± 0.0	58.4 ± 0.0	11.4 ± 0.0	8.0 ± 0.0	14.6 ± 0.0	27.3
GFK	19.7 ± 0.0	49.7 ± 0.0	63.1 ± 0.0	10.6 ± 0.0	7.9 ± 0.0	15.8 ± 0.0	27.8
CNN	61.6 ± 0.5	95.4 ± 0.3	99.0 ± 0.2	63.8 ± 0.5	51.1 ± 0.6	49.8 ± 0.4	70.1
LapCNN	60.4 ± 0.3	94.7 ± 0.5	**99.1** ± 0.2	63.1 ± 0.6	51.6 ± 0.4	48.2 ± 0.5	69.5
DDC	61.8 ± 0.4	95.0 ± 0.5	98.5 ± 0.4	64.4 ± 0.3	52.1 ± 0.8	52.2 ± 0.4	70.6
DAN_7	63.2 ± 0.2	94.8 ± 0.4	98.9 ± 0.3	65.2 ± 0.4	52.3 ± 0.4	52.1 ± 0.4	71.1
DAN_8	63.8 ± 0.4	94.6 ± 0.5	98.8 ± 0.6	65.8 ± 0.4	52.8 ± 0.4	51.9 ± 0.5	71.3
DAN_{SK}	63.3 ± 0.3	95.6 ± 0.2	99.0 ± 0.4	65.9 ± 0.7	53.2 ± 0.5	52.1 ± 0.4	71.5
DAN	**68.5** ± 0.4	**96.0** ± 0.3	99.0 ± 0.2	**67.0** ± 0.4	**54.0** ± 0.4	**53.1** ± 0.3	**72.9**

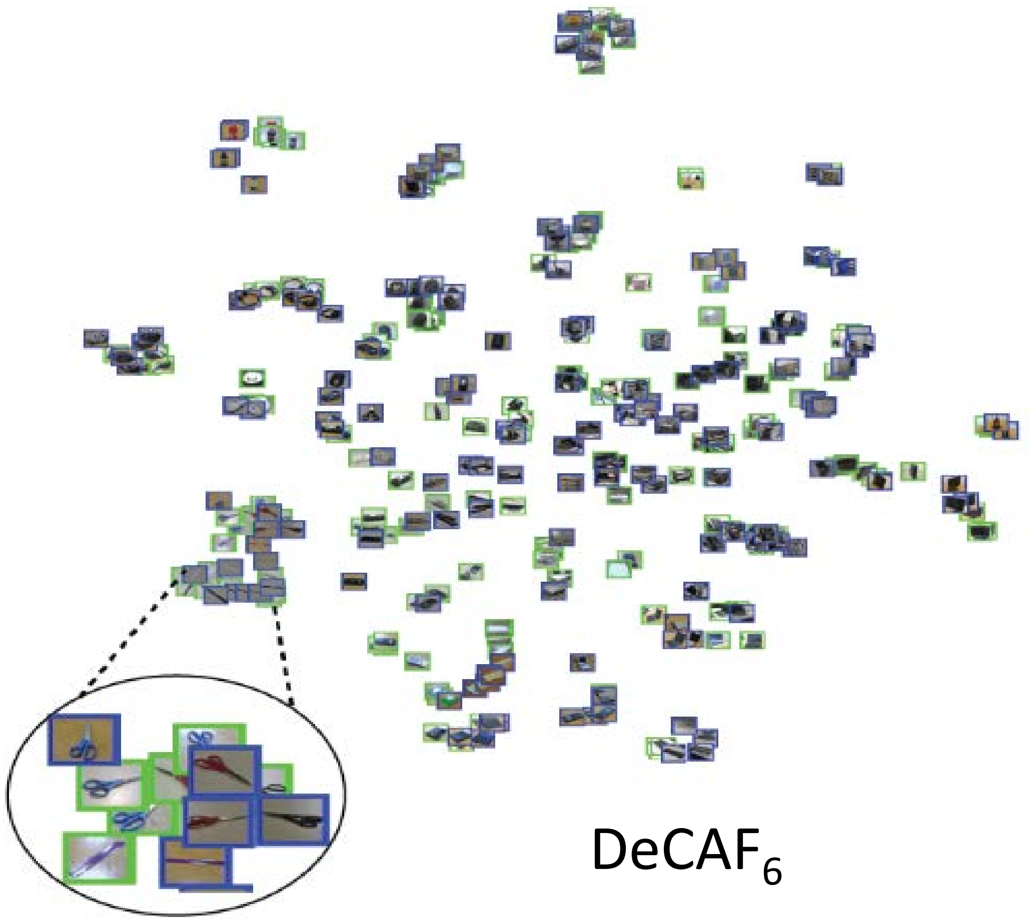
Table 2. Accuracy on *Office-10* + *Caltech-10* dataset with standard unsupervised adaptation protocol (Gong et al., 2013).

Method	A → C	W → C	D → C	C → A	C → W	C → D	Average
TCA	42.7 ± 0.0	34.1 ± 0.0	35.4 ± 0.0	54.7 ± 0.0	50.5 ± 0.0	50.3 ± 0.0	44.6
GFK	41.4 ± 0.0	26.4 ± 0.0	36.4 ± 0.0	56.2 ± 0.0	43.7 ± 0.0	42.0 ± 0.0	41.0
CNN	83.8 ± 0.3	76.1 ± 0.5	80.8 ± 0.4	91.1 ± 0.2	83.1 ± 0.3	89.0 ± 0.3	84.0
LapCNN	83.6 ± 0.6	77.8 ± 0.5	80.6 ± 0.4	**92.1** ± 0.3	81.6 ± 0.4	87.8 ± 0.4	83.9
DDC	84.3 ± 0.5	76.9 ± 0.4	80.5 ± 0.2	91.3 ± 0.3	85.5 ± 0.3	89.1 ± 0.3	84.6
DAN_7	84.7 ± 0.3	78.2 ± 0.5	81.8 ± 0.3	91.6 ± 0.4	87.4 ± 0.3	88.9 ± 0.5	85.4
DAN_8	84.4 ± 0.3	80.8 ± 0.4	81.7 ± 0.2	91.7 ± 0.3	90.5 ± 0.4	89.1 ± 0.4	86.4
DAN_{SK}	84.1 ± 0.4	79.9 ± 0.4	81.1 ± 0.5	91.4 ± 0.3	86.9 ± 0.5	89.5 ± 0.3	85.5
DAN	**86.0** ± 0.5	**81.5** ± 0.3	**82.0** ± 0.4	92.0 ± 0.3	**92.0** ± 0.4	**90.5** ± 0.2	**87.3**

Long, M., et al. "Learning Transferable Features with Deep Adaptation Networks". ICML 2015..

CNN & Universal Representations





Donahue, J., et al.. Decaf: A deep convolutional activation feature for generic visual recognition. ICML 2014.

Office31 results (2017)

Table 1. Classification accuracy (%) on *Office-31* dataset for unsupervised domain adaptation (AlexNet and ResNet)

Method	A → W	D → W	W → D	A → D	D → A	W → A	Avg
AlexNet (Krizhevsky et al., 2012)	61.6±0.5	95.4±0.3	99.0±0.2	63.8±0.5	51.1±0.6	49.8±0.4	70.1
TCA (Pan et al., 2011)	61.0±0.0	93.2±0.0	95.2±0.0	60.8±0.0	51.6±0.0	50.9±0.0	68.8
GFK (Gong et al., 2012)	60.4±0.0	95.6±0.0	95.0±0.0	60.6±0.0	52.4±0.0	48.1±0.0	68.7
DDC (Tzeng et al., 2014)	61.8±0.4	95.0±0.5	98.5±0.4	64.4±0.3	52.1±0.6	52.2±0.4	70.6
DAN (Long et al., 2015)	68.5±0.5	96.0±0.3	99.0±0.3	67.0±0.4	54.0±0.5	53.1±0.5	72.9
RTN (Long et al., 2016)	73.3±0.3	**96.8**±0.2	**99.6**±0.1	71.0±0.2	50.5±0.3	51.0±0.1	73.7
RevGrad (Ganin & Lempitsky, 2015)	73.0±0.5	96.4±0.3	99.2±0.3	72.3±0.3	53.4±0.4	51.2±0.5	74.3
JAN (ours)	74.9±0.3	96.6±0.2	99.5±0.2	71.8±0.2	**58.3**±0.3	55.0±0.4	76.0
JAN-A (ours)	**75.2**±0.4	96.6±0.2	**99.6**±0.1	**72.8**±0.3	57.5±0.2	**56.3**±0.2	**76.3**
ResNet (He et al., 2016)	68.4±0.2	96.7±0.1	99.3±0.1	68.9±0.2	62.5±0.3	60.7±0.3	76.1
TCA (Pan et al., 2011)	72.7±0.0	96.7±0.0	99.6±0.0	74.1±0.0	61.7±0.0	60.9±0.0	77.6
GFK (Gong et al., 2012)	72.8±0.0	95.0±0.0	98.2±0.0	74.5±0.0	63.4±0.0	61.0±0.0	77.5
DDC (Tzeng et al., 2014)	75.6±0.2	96.0±0.2	98.2±0.1	76.5±0.3	62.2±0.4	61.5±0.5	78.3
DAN (Long et al., 2015)	80.5±0.4	97.1±0.2	99.6±0.1	78.6±0.2	63.6±0.3	62.8±0.2	80.4
RTN (Long et al., 2016)	84.5±0.2	96.8±0.1	99.4±0.1	77.5±0.3	66.2±0.2	64.8±0.3	81.6
RevGrad (Ganin & Lempitsky, 2015)	82.0±0.4	96.9±0.2	99.1±0.1	79.7±0.4	68.2±0.4	67.4±0.5	82.2
JAN (ours)	85.4±0.3	**97.4**±0.2	**99.8**±0.2	84.7±0.3	68.6±0.3	70.0±0.4	84.3
JAN-A (ours)	**86.0**±0.4	96.7±0.3	99.7±0.1	**85.1**±0.4	**69.2**±0.4	**70.7**±0.5	**84.6**

Long, M., et al. "Deep transfer learning with joint adaptation networks". ICML 2017.

Office31and Office-Caltech results (2017)

Table 1. Classification accuracy (%) on *Office-31* dataset for unsupervised domain adaptation (AlexNet and ResNet)

Method	A → W	D → W	W → D	A → D	D → A	W → A	Avg
AlexNet (Krizhevsky et al., 2012)	61.6±0.5	95.4±0.3	99.0±0.2	63.8±0.5	51.1±0.6	49.8±0.4	70.1
TCA (Pan et al., 2011)	61.0±0.0	93.2±0.0	95.2±0.0	60.8±0.0	51.6±0.0	50.9±0.0	68.8
GFK (Gong et al., 2012)	60.4±0.0	95.6±0.0	95.0±0.0	60.6±0.0	52.4±0.0	48.1±0.0	68.7
DDC (Tzeng et al., 2014)	61.8±0.4	95.0±0.5	98.5±0.4	64.4±0.3	52.1±0.6	52.2±0.4	70.6
DAN (Long et al., 2015)	68.5±0.5	96.0±0.3	99.0±0.3	67.0±0.4	54.0±0.5	53.1±0.5	72.9
RTN (Long et al., 2016)	73.3±0.3	**96.8**±0.2	**99.6**±0.1	71.0±0.2	50.5±0.3	51.0±0.1	73.7
RevGrad (Ganin & Lempitsky, 2015)	73.0±0.5	96.4±0.3	99.2±0.3	72.3±0.3	53.4±0.4	51.2±0.5	74.3
JAN (ours)	74.9±0.3	96.6±0.2	99.5±0.2	71.8±0.2	**58.3**±0.3	55.0±0.4	76.0
JAN-A (ours)	**75.2**±0.4	96.6±0.2	**99.6**±0.1	**72.8**±0.3	57.5±0.2	**56.3**±0.2	**76.3**
ResNet (He et al., 2016)	68.4±0.2	96.7±0.1	99.3±0.1	68.9±0.2	62.5±0.3	60.7±0.3	76.1
TCA (Pan et al., 2011)	72.7±0.0	96.7±0.0	99.6±0.0	74.1±0.0	61.7±0.0	60.9±0.0	77.6
GFK (Gong et al., 2012)	72.8±0.0	95.0±0.0	98.2±0.0	74.5±0.0	63.4±0.0	61.0±0.0	77.5
DDC (Tzeng et al., 2014)	75.6±0.2	96.0±0.2	98.2±0.1	76.5±0.3	62.2±0.4	61.5±0.5	78.3
DAN (Long et al., 2015)	80.5±0.4	97.1±0.2	99.6±0.1	78.6±0.2	63.6±0.3	62.8±0.2	80.4
RTN (Long et al., 2016)	84.5±0.2	96.8±0.1	99.4±0.1	77.5±0.3	66.2±0.2	64.8±0.3	81.6
RevGrad (Ganin & Lempitsky, 2015)	82.0±0.4	96.9±0.2	99.1±0.1	79.7±0.4	68.2±0.4	67.4±0.5	82.2
JAN (ours)	85.4±0.3	**97.4**±0.2	**99.8**±0.2	84.7±0.3	68.6±0.3	70.0±0.4	84.3
JAN-A (ours)	**86.0**±0.4	96.7±0.3	99.7±0.1	**85.1**±0.4	**69.2**±0.4	**70.7**±0.5	**84.6**

Long, M., et al. "Deep transfer learning with joint adaptation networks". In ICML 2017.

Office31and Office-Caltech results (2017)

Table 1. Accuracy on *Office-31* dataset with standard unsupervised adaptation protocol (Gong et al., 2013).

Method	A → W	D → W	W → D	A → D	D → A	W → A	Average
TCA	21.5 ± 0.0	50.1 ± 0.0	58.4 ± 0.0	11.4 ± 0.0	8.0 ± 0.0	14.6 ± 0.0	27.3
GFK	19.7 ± 0.0	49.7 ± 0.0	63.1 ± 0.0	10.6 ± 0.0	7.9 ± 0.0	15.8 ± 0.0	27.8

ResNet (He et al., 2016)	68.4±0.2	96.7±0.1	99.3±0.1	68.9±0.2	62.5±0.3	60.7±0.3	76.1
TCA (Pan et al., 2011)	72.7±0.0	96.7±0.0	99.6±0.0	74.1±0.0	61.7±0.0	60.9±0.0	77.6
GFK (Gong et al., 2012)	72.8±0.0	95.0±0.0	98.2±0.0	74.5±0.0	63.4±0.0	61.0±0.0	77.5

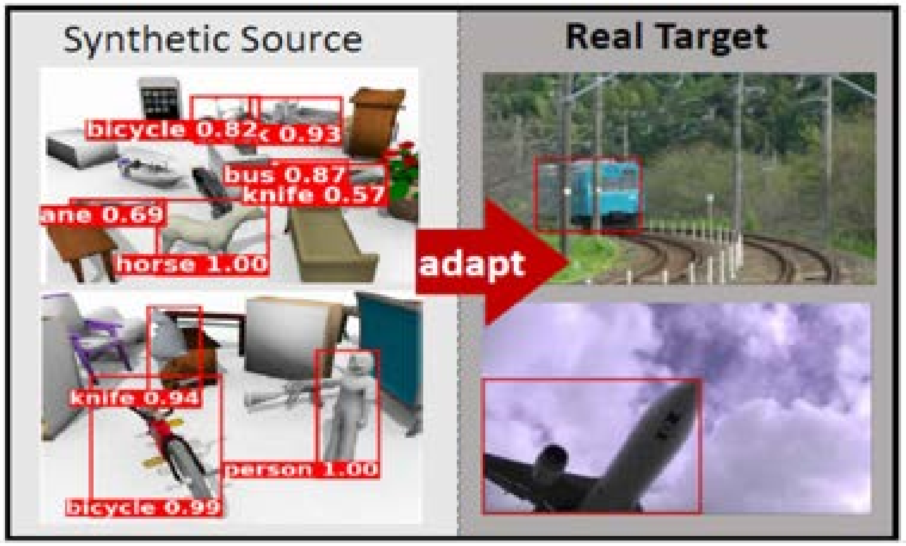
Long, M., et al. "Learning Transferable Features with Deep Adaptation Networks". In ICML. 2015.
Long, M., et al. "Deep transfer learning with joint adaptation networks". In ICML 2017.

Benchmarks

DomainNet - VisDA Challenges

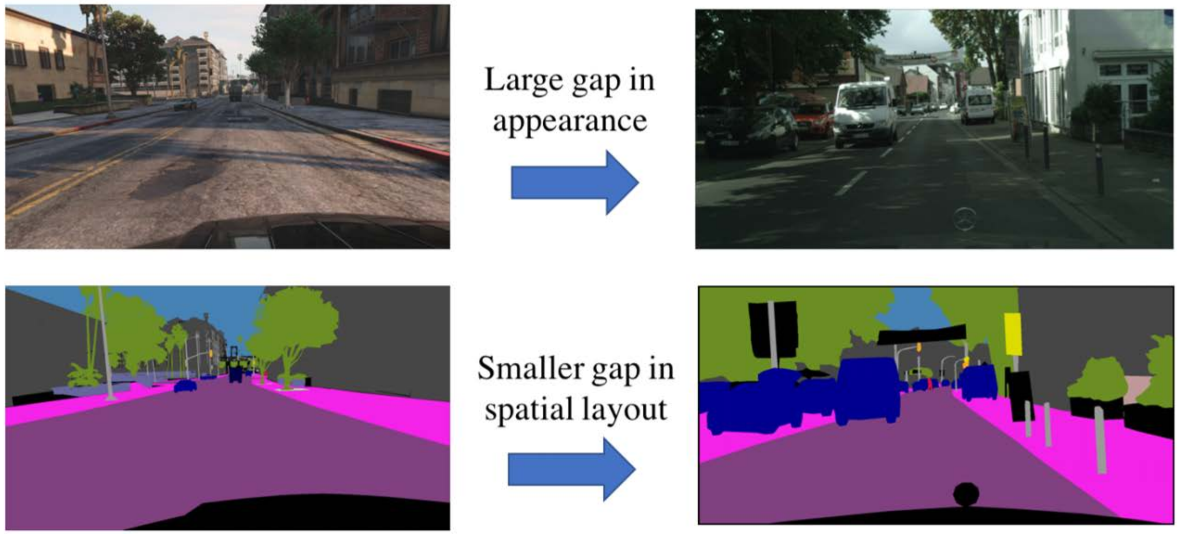
https://ai.bu.edu/visda-2019/

Benchmarks



Object detection

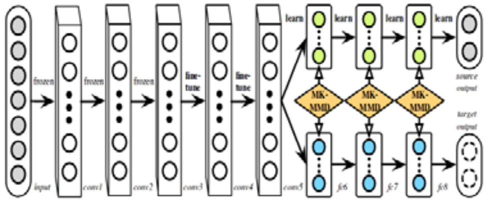
Semantic segmentation



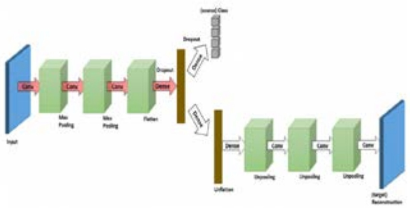
https://ai.bu.edu/visda-2018/

Tsai, Y. H. et al. "Learning to adapt structured output space for semantic segmentation". In CVPR 2018.

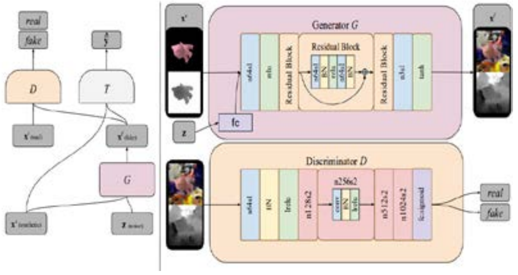
Deep DA state of the art



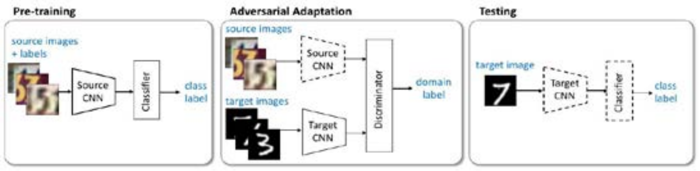
[Long et al. ICML 2015]



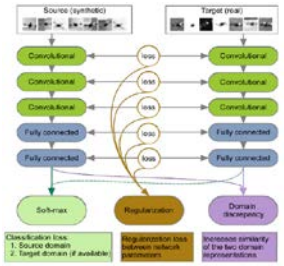
[Ghifary et al. ECCV 2016]



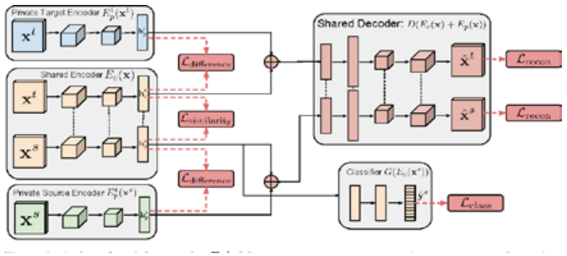
[Bousmalis et al. CVPR 2017]



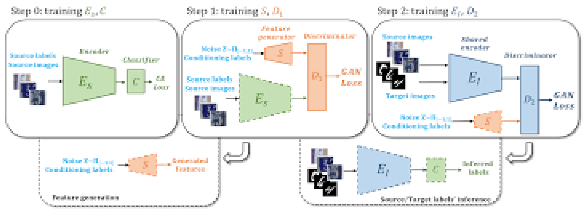
[Tzeng et al. CVPR 2017]



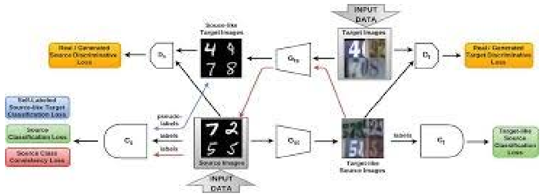
[Rozantsev et al. TPAMI 2019]



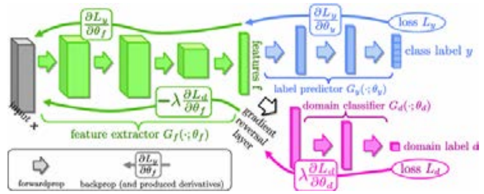
[Bousmalis et al. NIPS 2016]



[Volpi et al. CVPR 2018]



[Carlucci et al. CVPR 2018]



[Ganin et al. JMLR 2016]

..and many, many more methods

Thanks for the attention