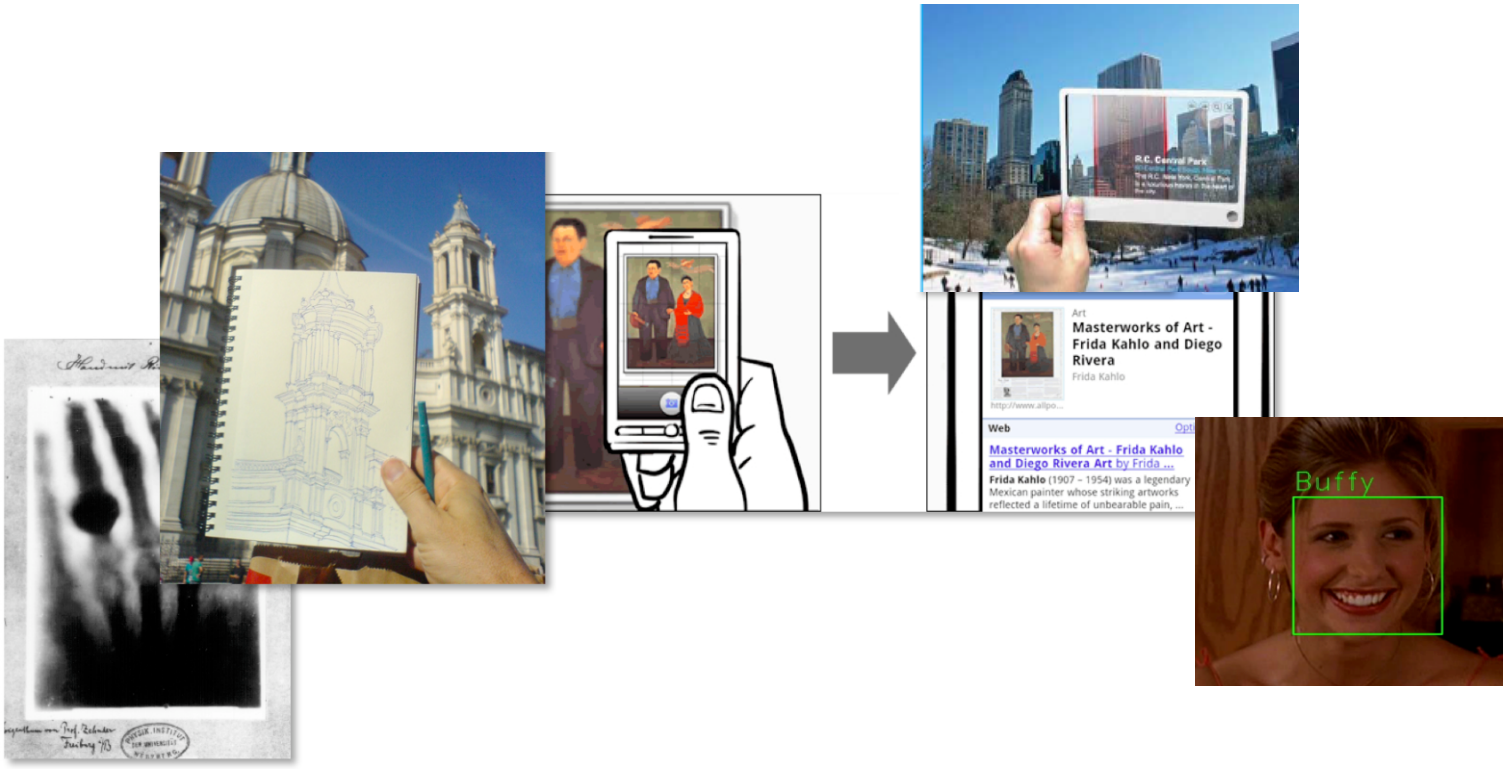


Deep (4)

Matthieu Cord -- SU

Outline

ConvNets as Deep Neural Networks for Vision

1. Neural Nets
2. Deep Convolutional Neural Networks
3. Modern Deep Architectures
4. **Beyond ImageNet**
    1. **Fully Convolutional Networks (FCNs)**
    2. Transfer

From ImageNet to complex scenes

- ImageNet: huge dataset (1.2M training images) with labels ... but centered objects

ImageNet

- How to apply/adapt/modify learning strategies to deal with:

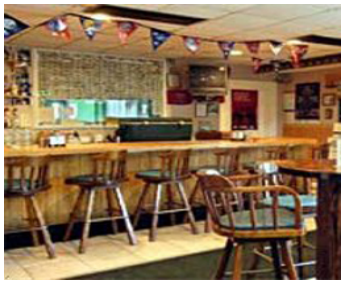
VOC 2012

MS COCO

From ImageNet to complex scenes?

- Working on datasets with complex scenes (large and cluttered background), not centered objects, variable size, ...

VOC07/12



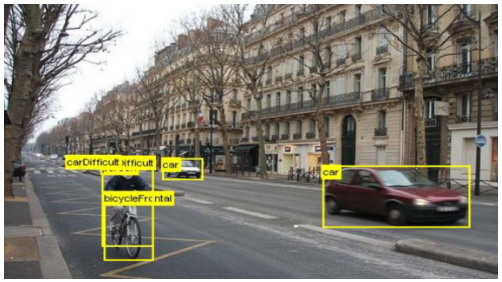
MIT67

15 Scene

COCO

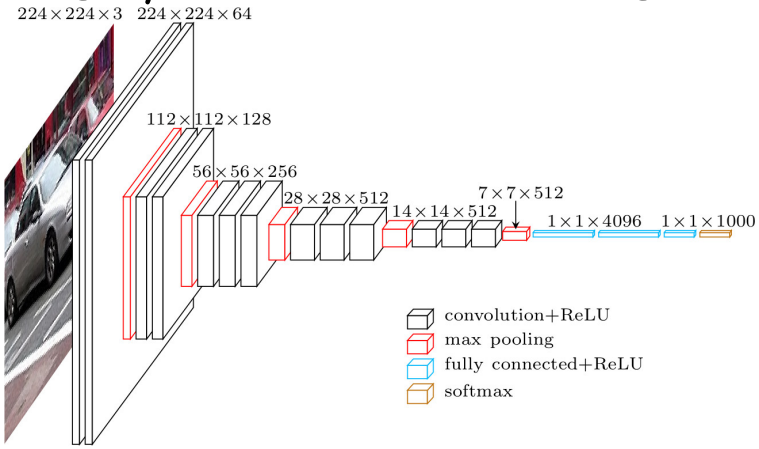
VOC12 Action

- Select relevant regions $\rightarrow$ better prediction



- Full annotations expensive $\Rightarrow$ training with weak supervision

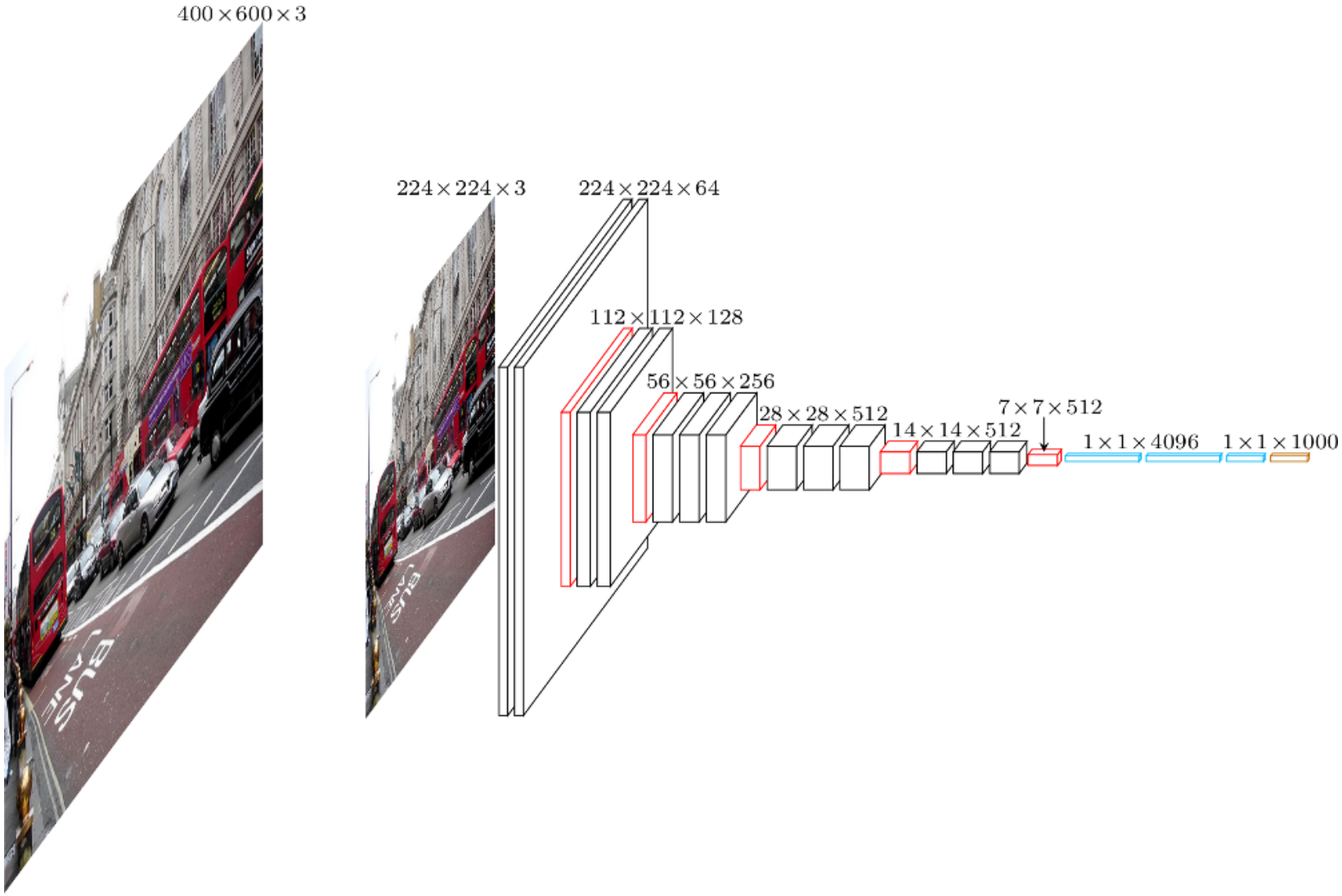
How to adapt VGG16 archi. for large/complex images?



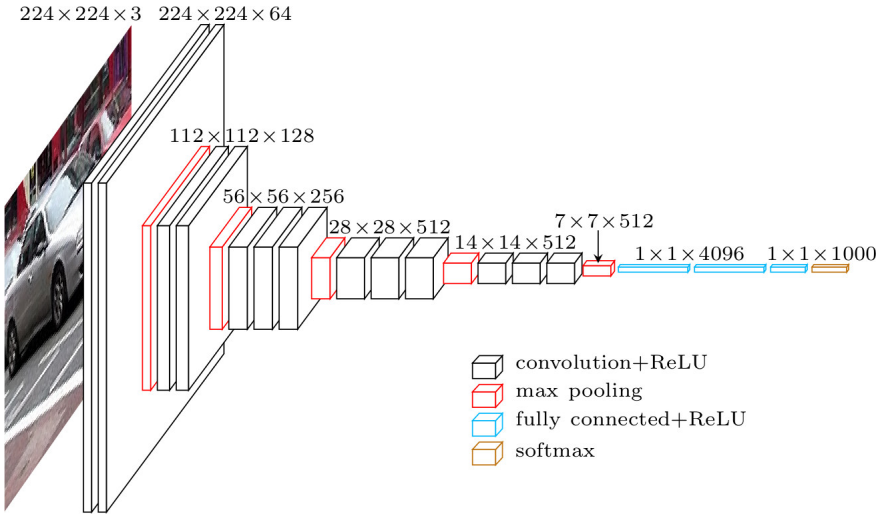
?

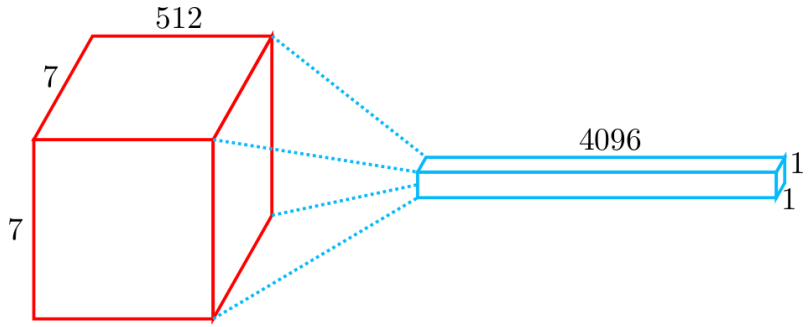
Naïve approach: brut transfer (next Section)

- Resize the image



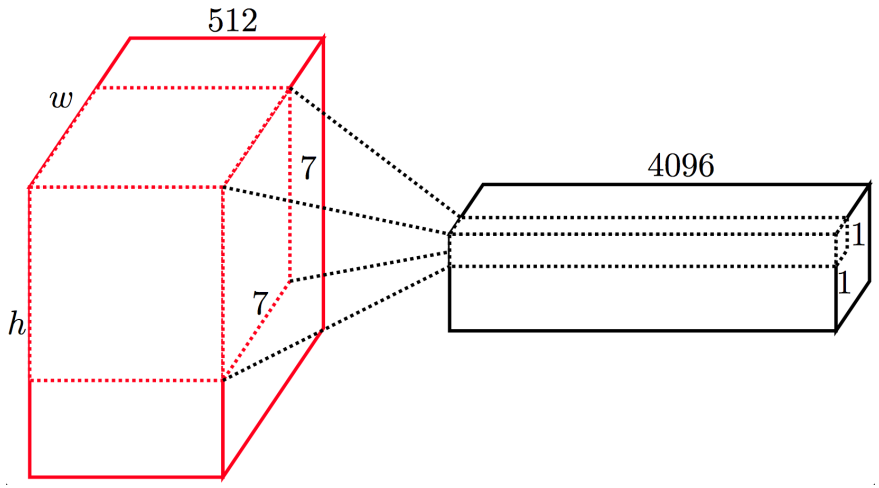
Sliding window ⇒ convolutional layers



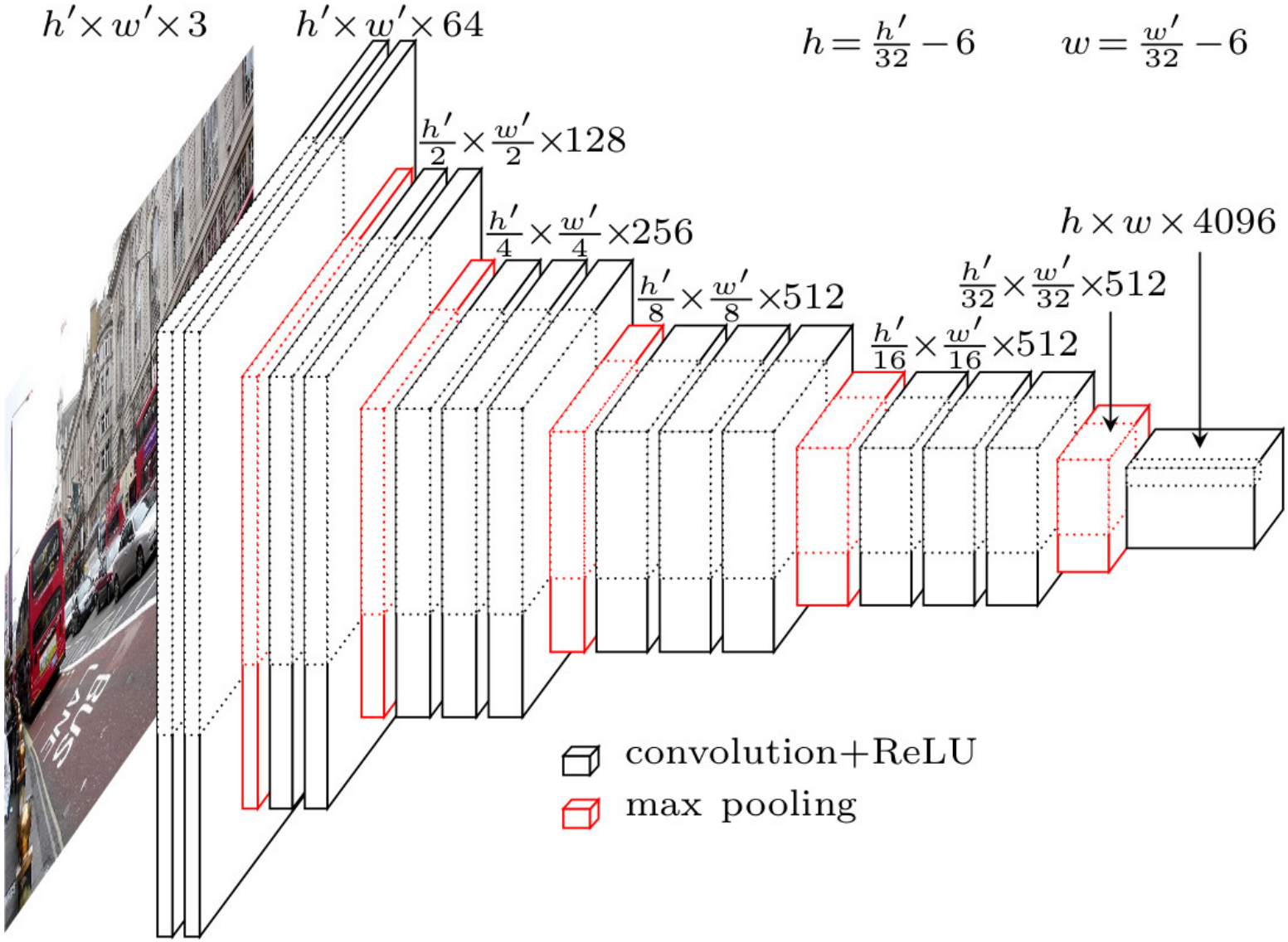


Sliding window ⇒ convolutional layers

- Fully connected as convolutional layer (here 4096 conv. filters 7x7x512)



Sliding window $\Rightarrow$ convolutional layers



Transfer – Pooling – Classification

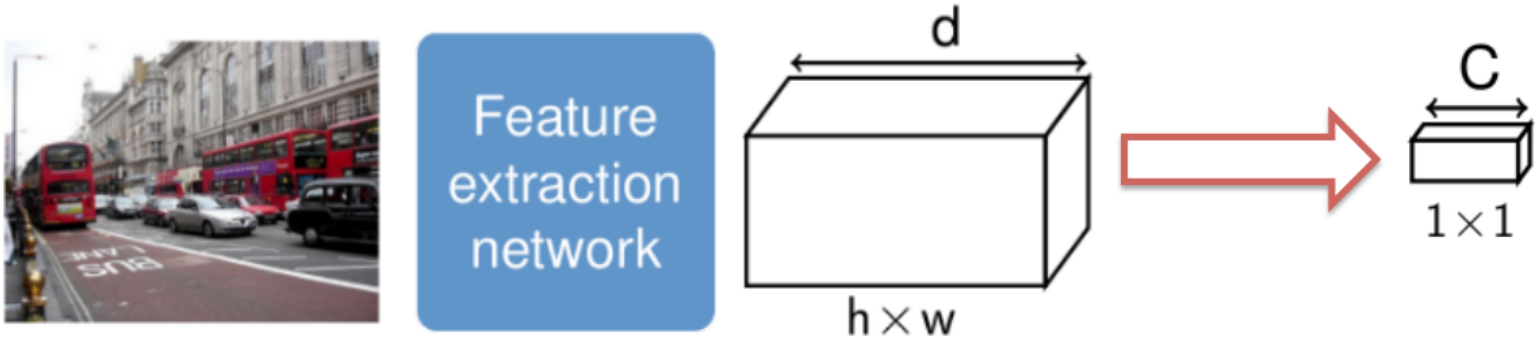
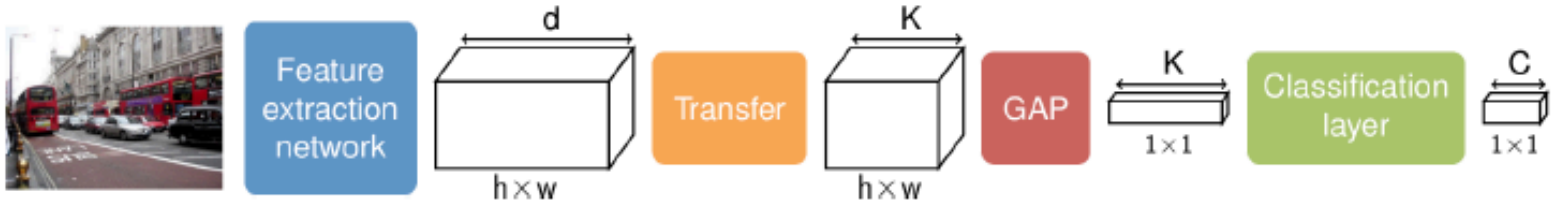


Image-based strategy

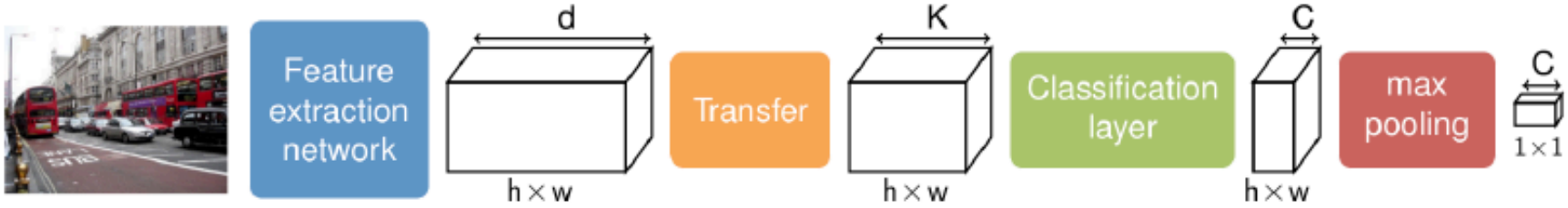
- Global Average Pooling (GoogLeNet, ResNet)



Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba
Learning Deep Features for Discriminative Localization.
In *CVPR*, 2016.

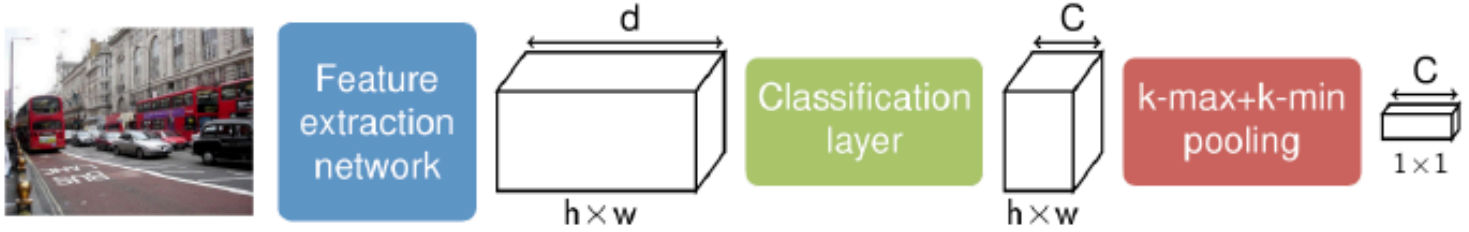
Region-based strategy

- Deep MIL



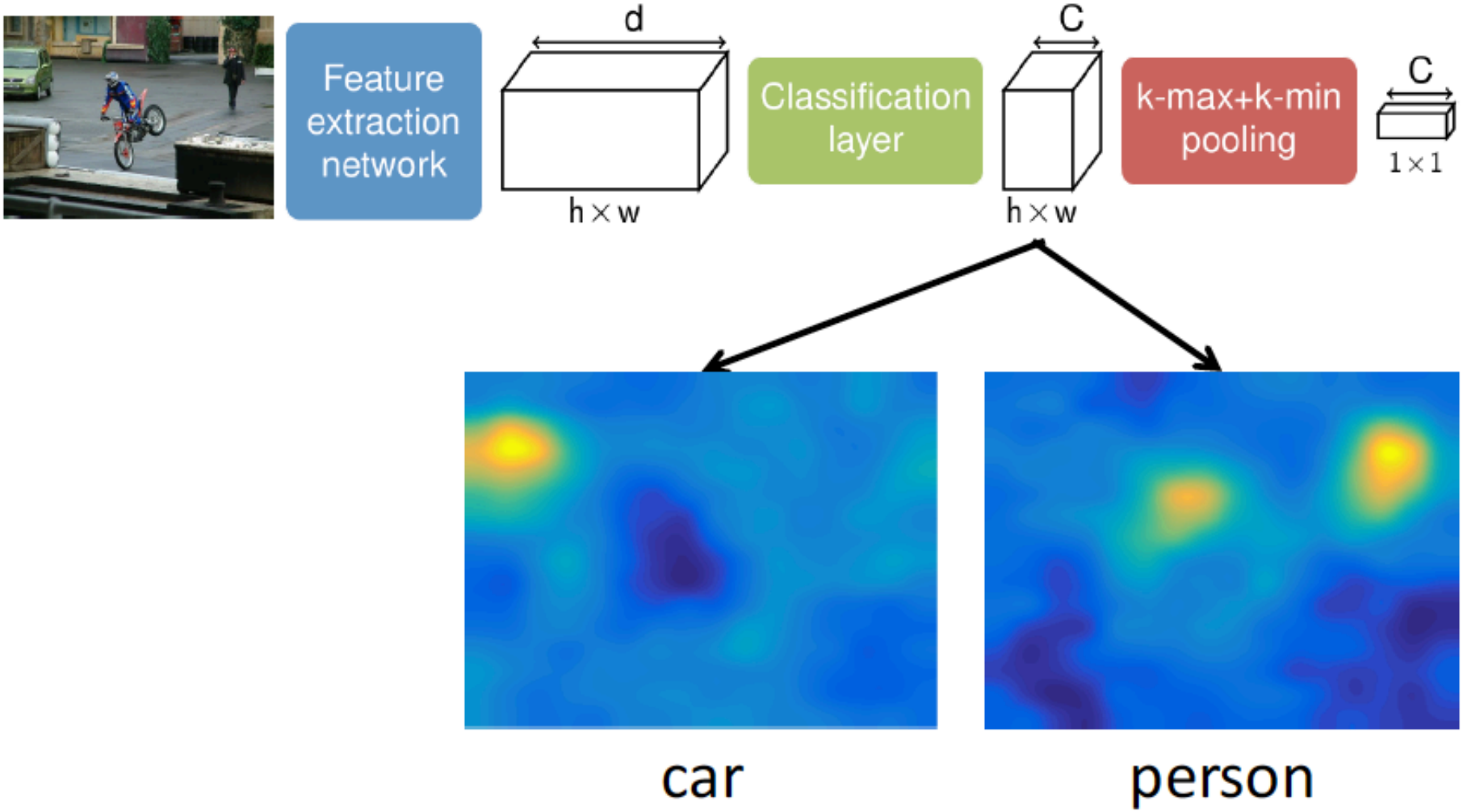
Maxime Oquab, Léon Bottou, Ivan Laptev and Josef Sivic
Is object localization for free? – Weakly-supervised learning with CNNs.
In *CVPR*, 2015.

- WELDON and ProNet [Sun, CVPR16]

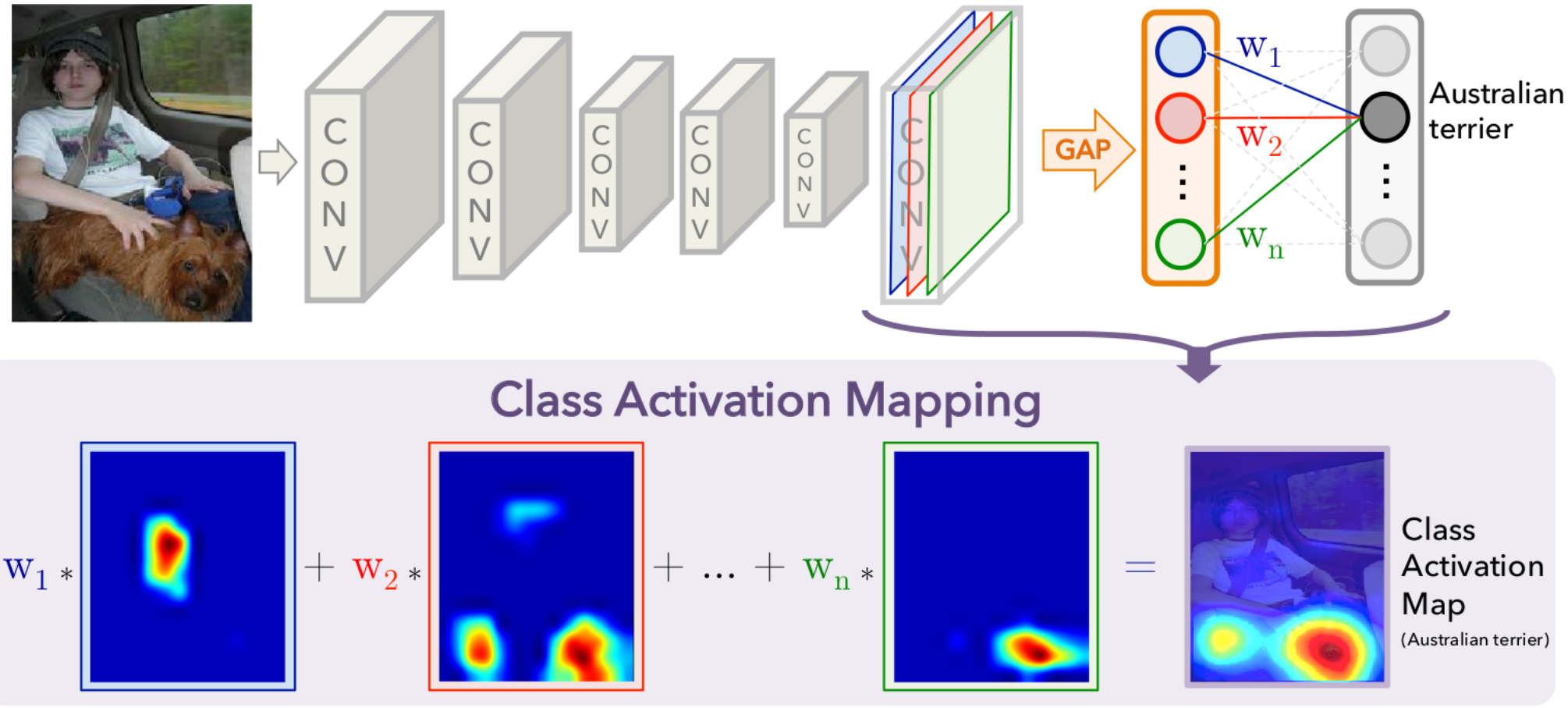


Thibaut Durand, Nicolas Thome, and Matthieu Cord
WELDON: Weakly Supervised Learning of Deep ConvNets.
In *CVPR*, 2016.

Pixel contribution to the classification

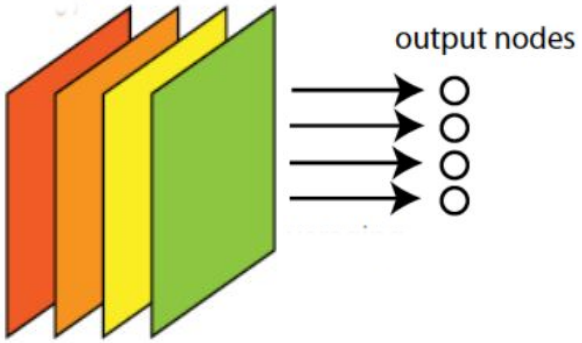


Pixel contribution to the classification



Pooling schemes

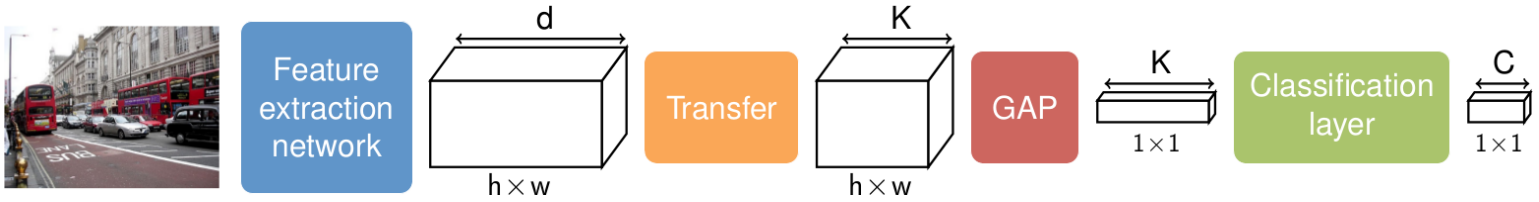
- Max [Oquab, CVPR15]



$$y^c = \max_{i,j} z_{ij}^c$$

- GAP [Zhou, CVPR16]

$$y^c = \frac{1}{N} \sum_{i,j} z_{ij}^c$$



- LSE [Pinheiro, CVPR15] / SPLeap [Kulkarni, ECCV16]

$$y^c = \frac{1}{\beta} \log \left(\frac{1}{N} \sum_{i,j} \exp(\beta \cdot z_{ij}^c) \right)$$

Max pooling limitation

- Classifying only with the max scoring region

- Loss of contextual information

Max pooling limitation

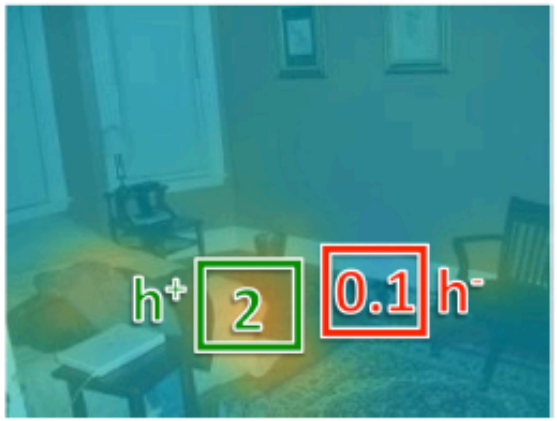
- Classifying only with the max scoring region

- Loss of contextual information

WELDON: `max+min` pooling

- $\mathbf{h}^+$: presence of the class $\rightarrow$ high $\mathbf{h}^+$
- $\mathbf{h}^-$: localized evidence of the absence of class

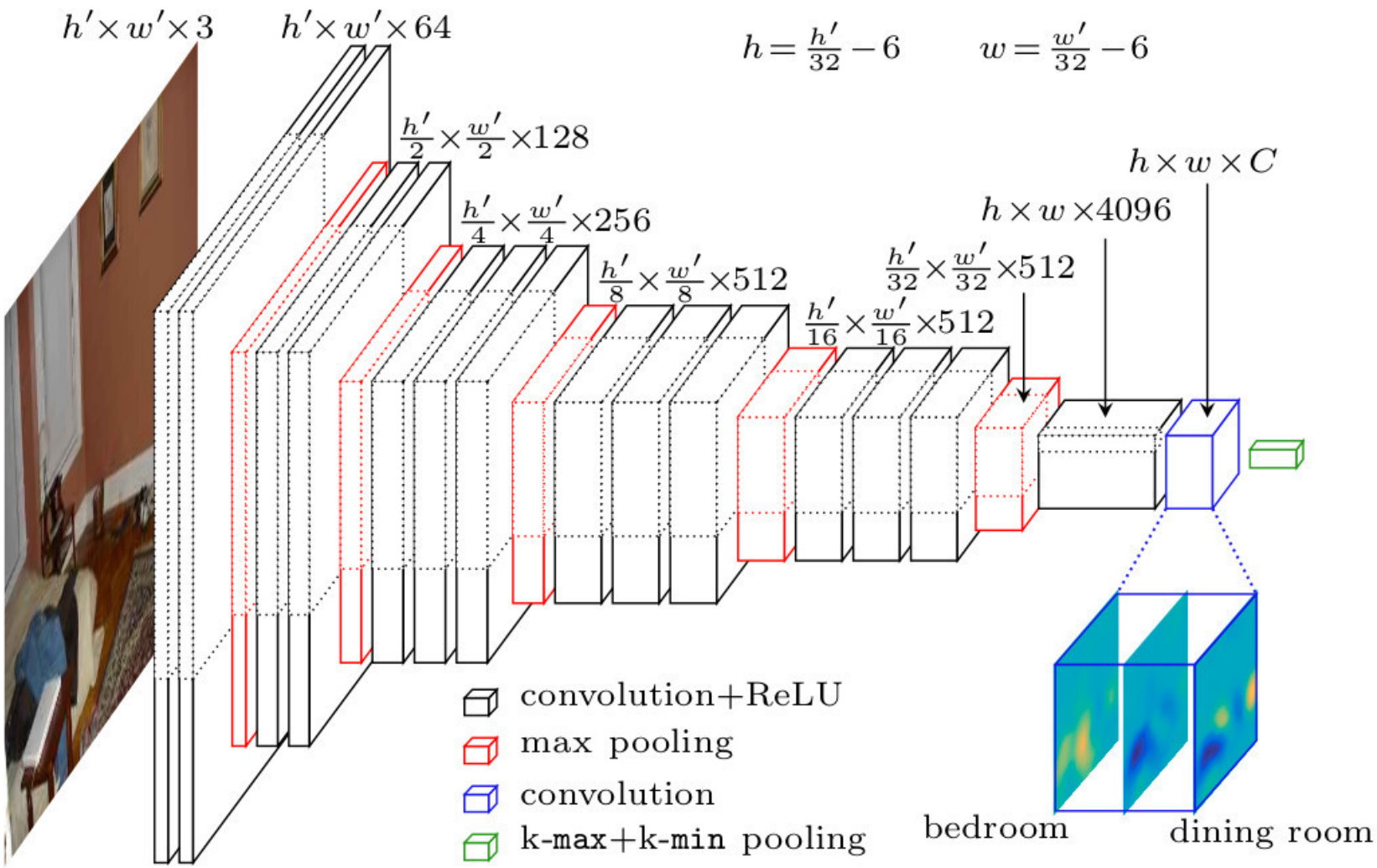
original image



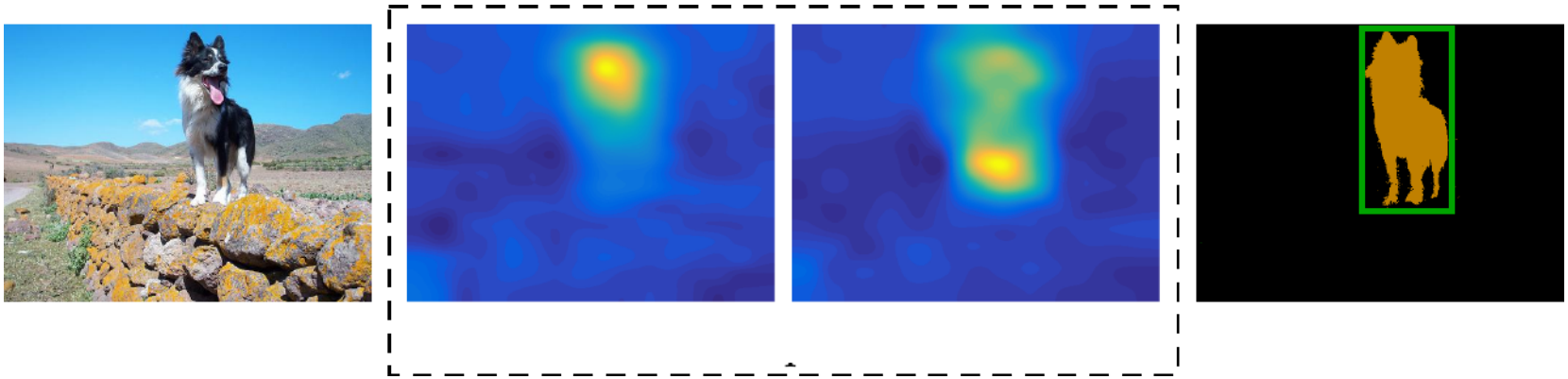
bedroom

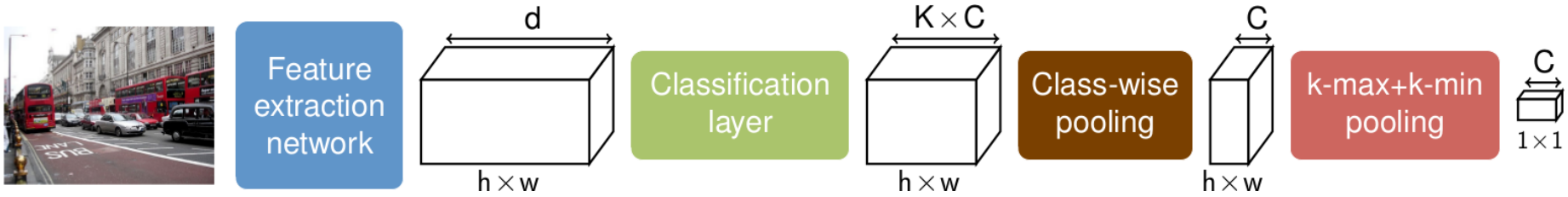
airport inside

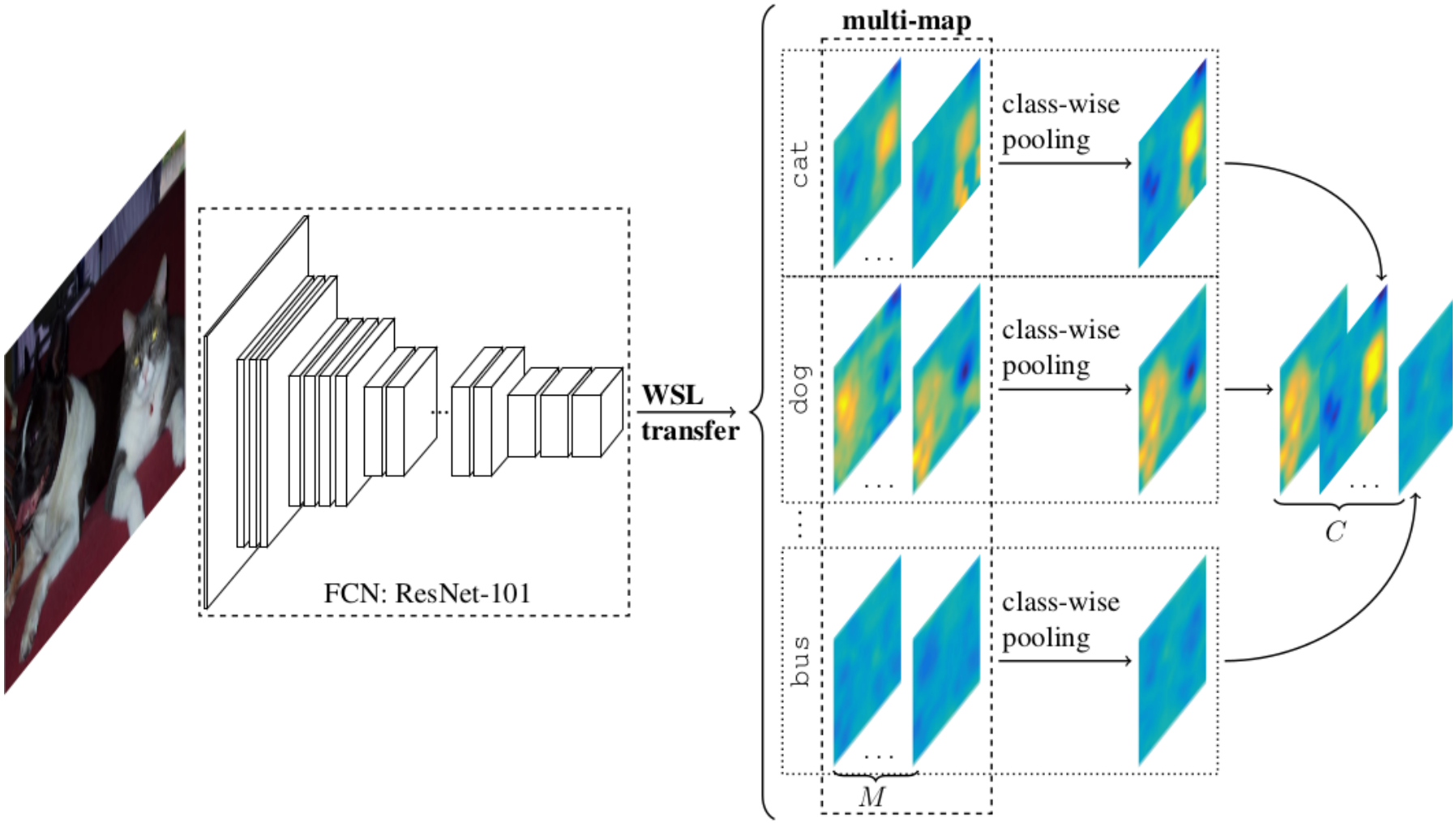
dining room


$h' \times w' \times 3$
$h' \times w' \times 64$
$h = \frac{h'}{32} - 6$
$w = \frac{w'}{32} - 6$
$\frac{h'}{2} \times \frac{w'}{2} \times 128$
$\frac{h'}{4} \times \frac{w'}{4} \times 256$
$\frac{h'}{8} \times \frac{w'}{8} \times 512$
$\frac{h'}{16} \times \frac{w'}{16} \times 512$
$\frac{h'}{32} \times \frac{w'}{32} \times 512$
$h \times w \times 4096$
$h \times w \times C$
convolution+ReLU
max pooling
convolution
k-max+k-min pooling
bedroom
dining room

- Generalization to K models per class
- Catch multiple class-related modalities

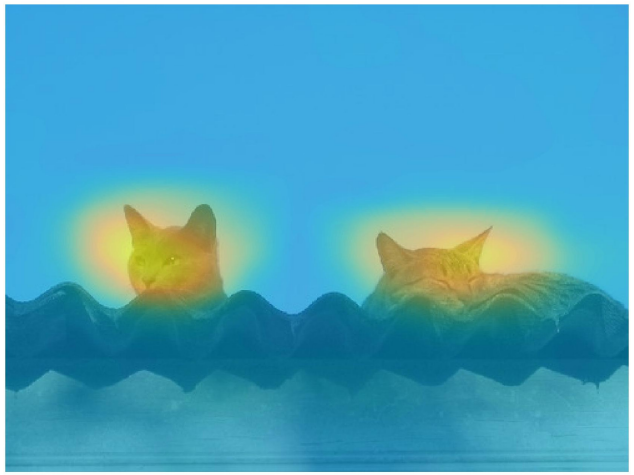





multi-map
FCN: ResNet-101
WSL
transfer
cat
dog
bus
class-wise
pooling
class-wise
pooling
class-wise
pooling
C
M

Class activation maps

bus



cat

horse

aeroplane

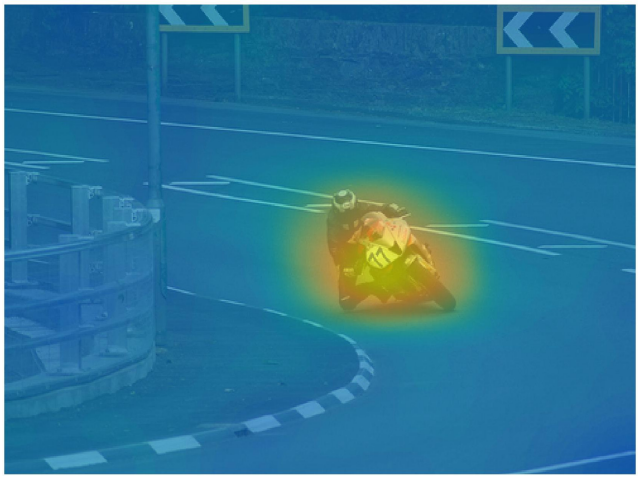
bottle

bicycle

Class activation maps

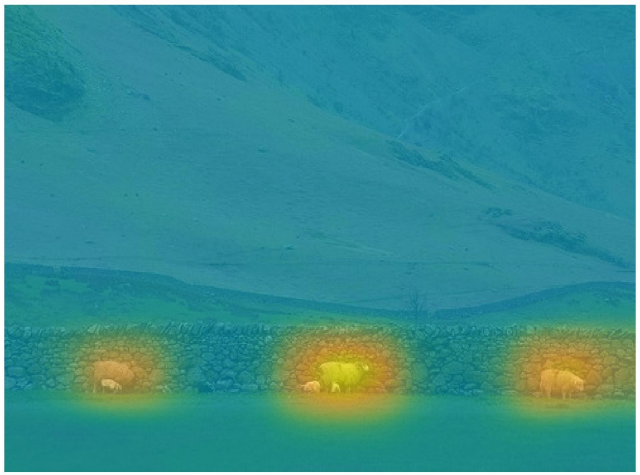
bicycle

bird



motorbike

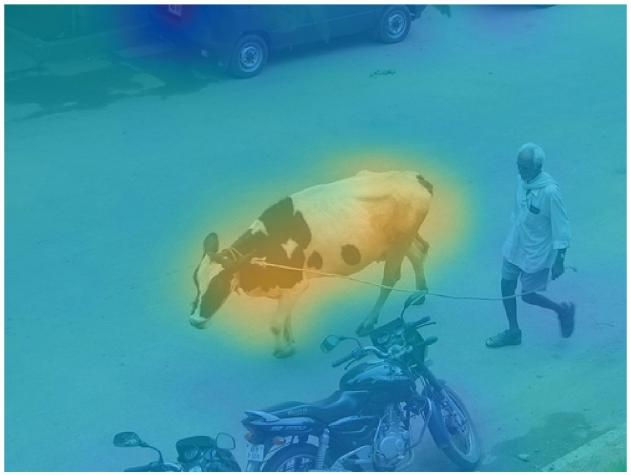
person



sheep

bird

Class activation maps



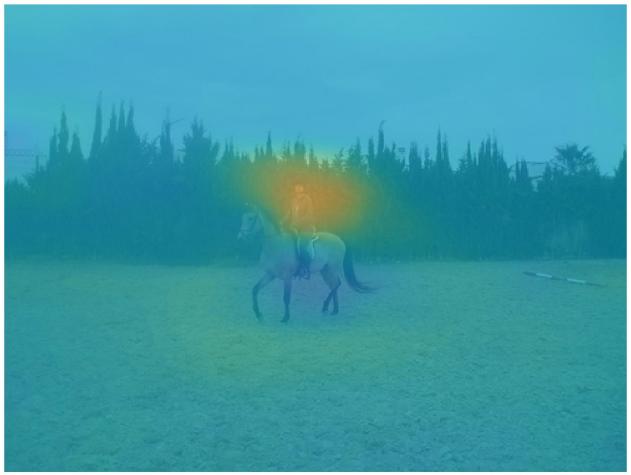
cow

motorbike

horse

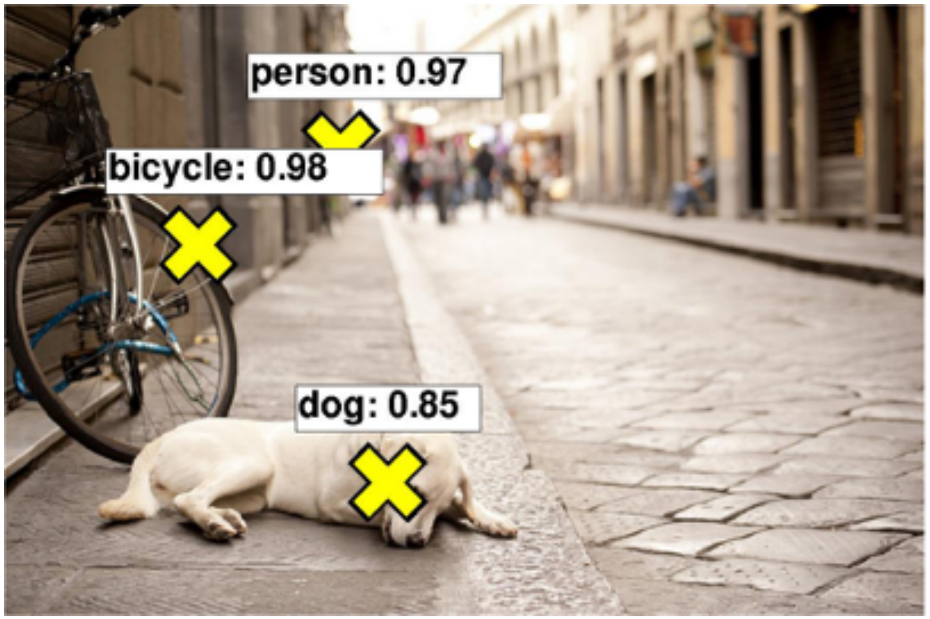
person

car



person

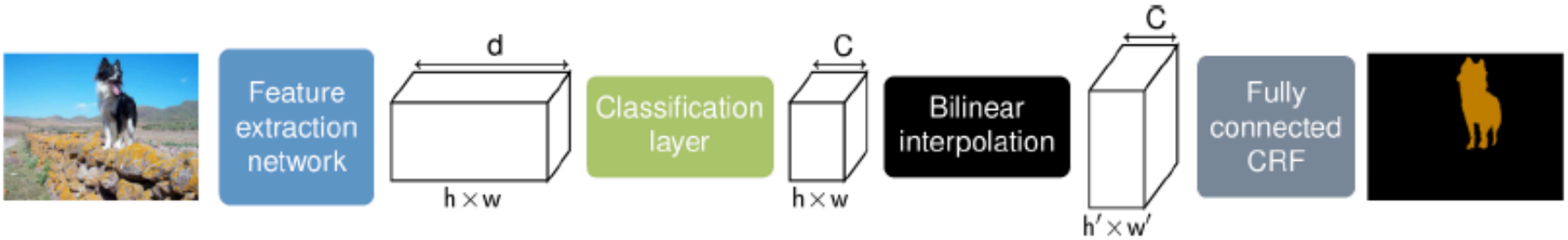
Visual recognition task: localization



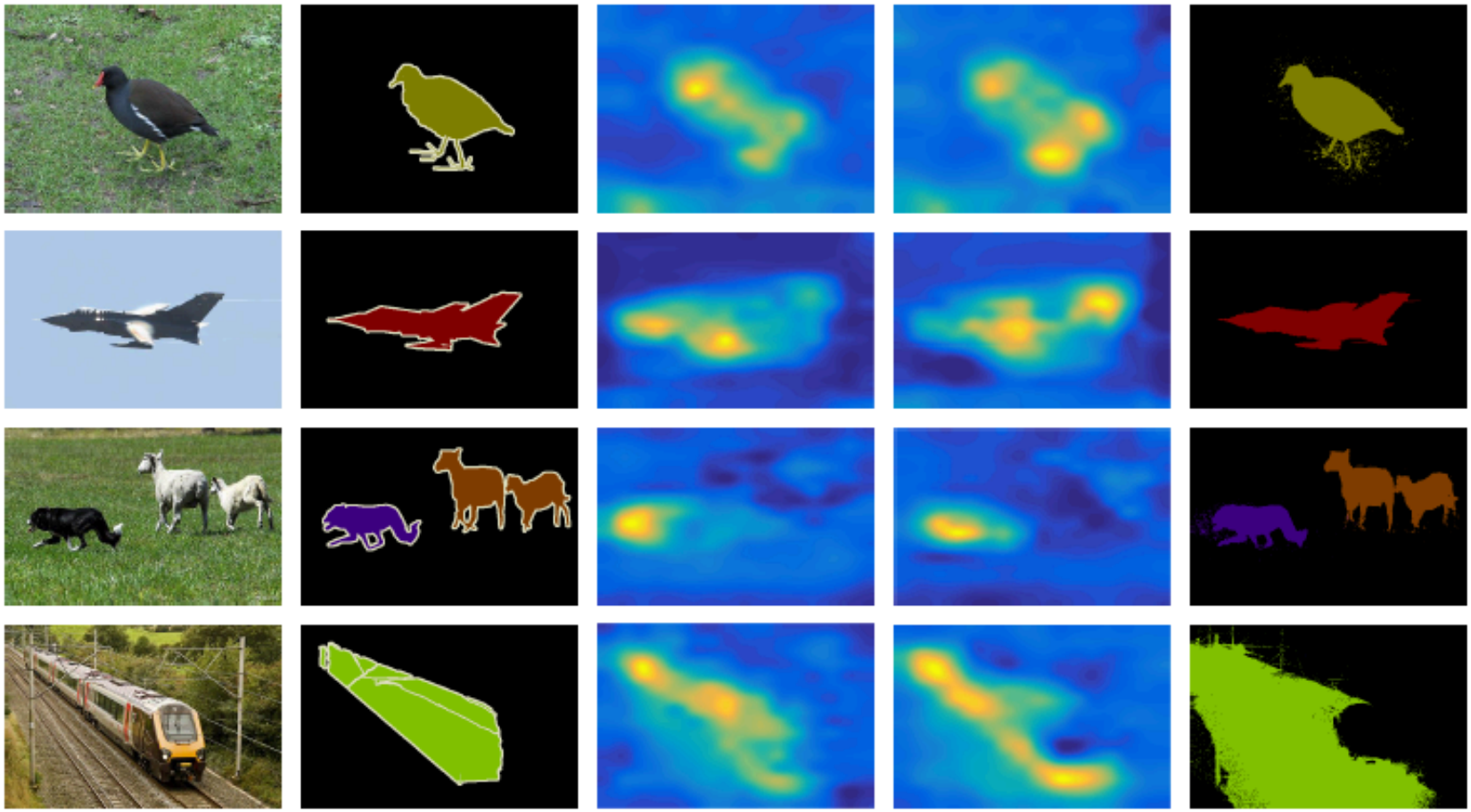
Method	VOC 2012	MS COCO
Deep MIL	74.5	41.2
ProNet	77.7	46.4
WSLocalization	79.7	49.2

In preview Segmentation

- WSL segmentation framework
  - Learning with image-level labels (presence/absence of the class)
  - Difficult task: no information about location and extend of objects
- Localized features in spatial maps
- Deep + fully connected CRFs



In preview Segmentation



Outline

ConvNets as Deep Neural Networks for Vision

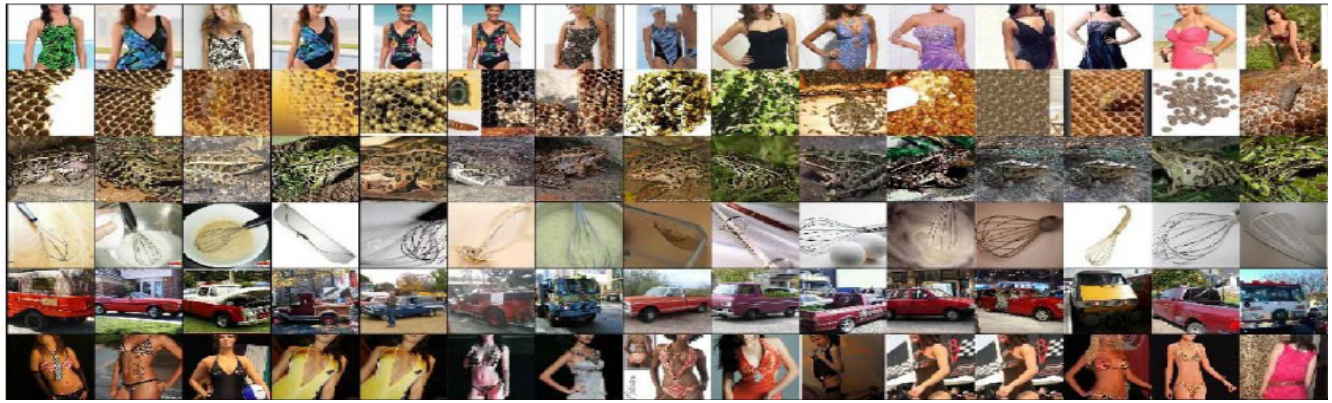
1. Neural Nets
2. Deep Convolutional Neural Networks
3. Beyond ImageNet
    1. Fully Convolutional Networks (FCNs)
    2. **Transfer**

Transfer from ImageNet

Transfer as **generic features**

Brut Deep features (learned from ImageNet)

Retrieval



Transfer learning

Frozen features + SVM => solution to small datasets

Frozen features + Deep

Fine tuning not easy in that case (small datasets)

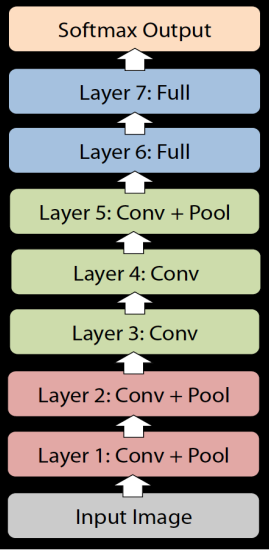
Transfer from ImageNet

Source: ImageNet => AlexNet trained

Target: Chopped AlexNet (layer i) + SVM trained and test on Cal-101 and Cal-256:

Architecture of Krizhevsky et al.

- 8 layers total
- Trained on Imagenet dataset [Deng et al. CVPR'09]
- 18.2% top-5 error
- Our reimplementation: 18.1% top-5 error



Tapping off Features at each Layer

Plug features from each layer into linear SVM or soft-max

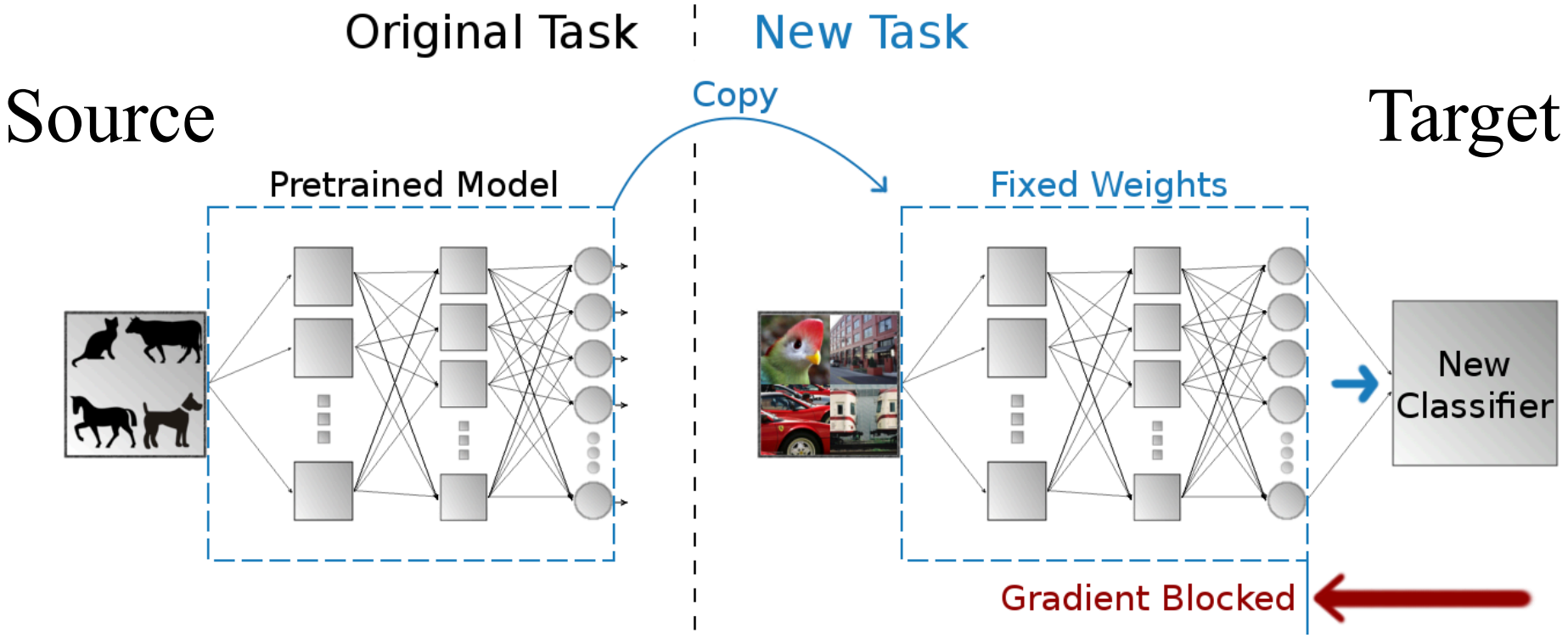
	Cal-101 (30/class)	Cal-256 (60/class)
SVM (1)	44.8 ± 0.7	24.6 ± 0.4
SVM (2)	66.2 ± 0.5	39.6 ± 0.3
SVM (3)	72.3 ± 0.4	46.0 ± 0.3
SVM (4)	76.6 ± 0.4	51.3 ± 0.1
SVM (5)	$\mathbf{86.2 \pm 0.8}$	65.6 ± 0.3
SVM (7)	$\mathbf{85.5 \pm 0.4}$	$\mathbf{71.7 \pm 0.2}$
Softmax (5)	82.9 ± 0.4	65.7 ± 0.5
Softmax (7)	$\mathbf{85.4 \pm 0.4}$	$\mathbf{72.6 \pm 0.1}$

=> Results better than SoA CV methods on Cal-101!

Transfer: fine-tuning of a deep model

Train a deep (AlexNet) on source (ImageNet)
Keep the deep params. for target and complete with a small deep on top (fully trained on target)
Fine-tune the whole model on target data

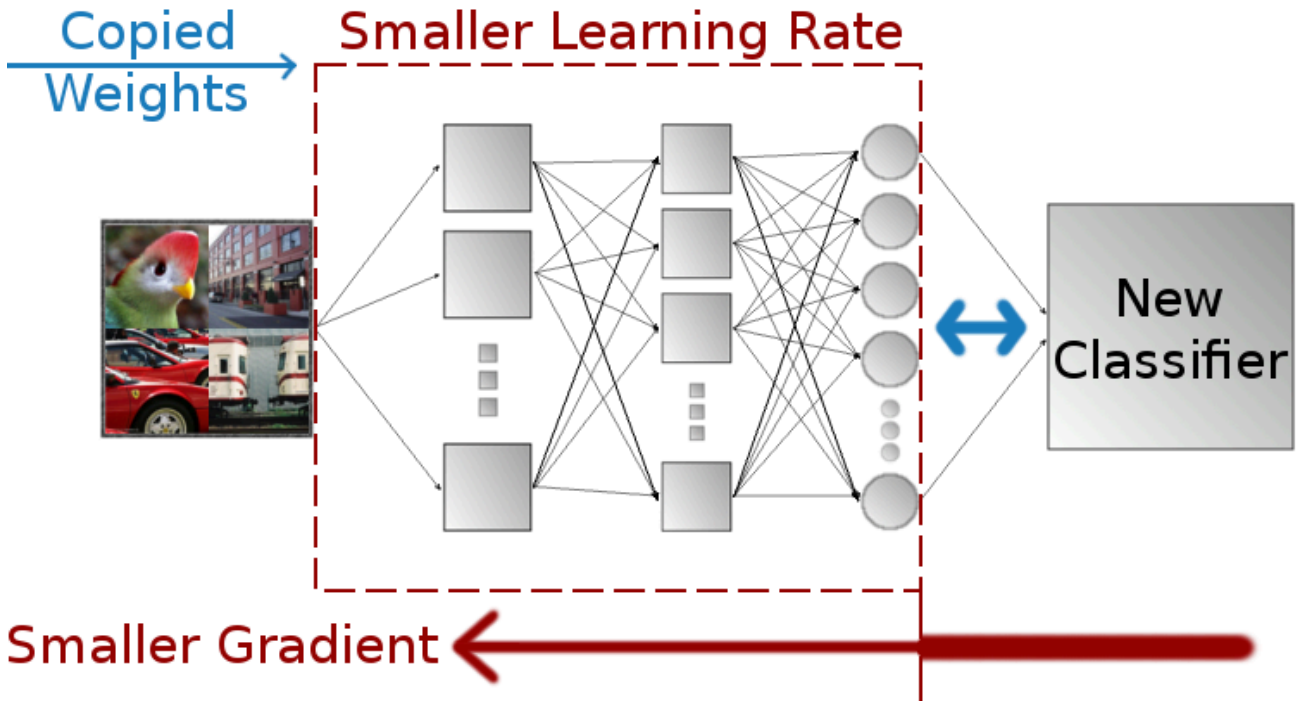
- Challenge: only limited target data, careful about overfitting
- Solution: Freeze the gradient's update for AlexNet part



Transfer: fine-tuning of a deep model

Train a deep (AlexNet) on source (ImageNet)
Keep the deep params. for target and complete with a small deep on top (fully trained on target)
Fine-tune the whole model on target data

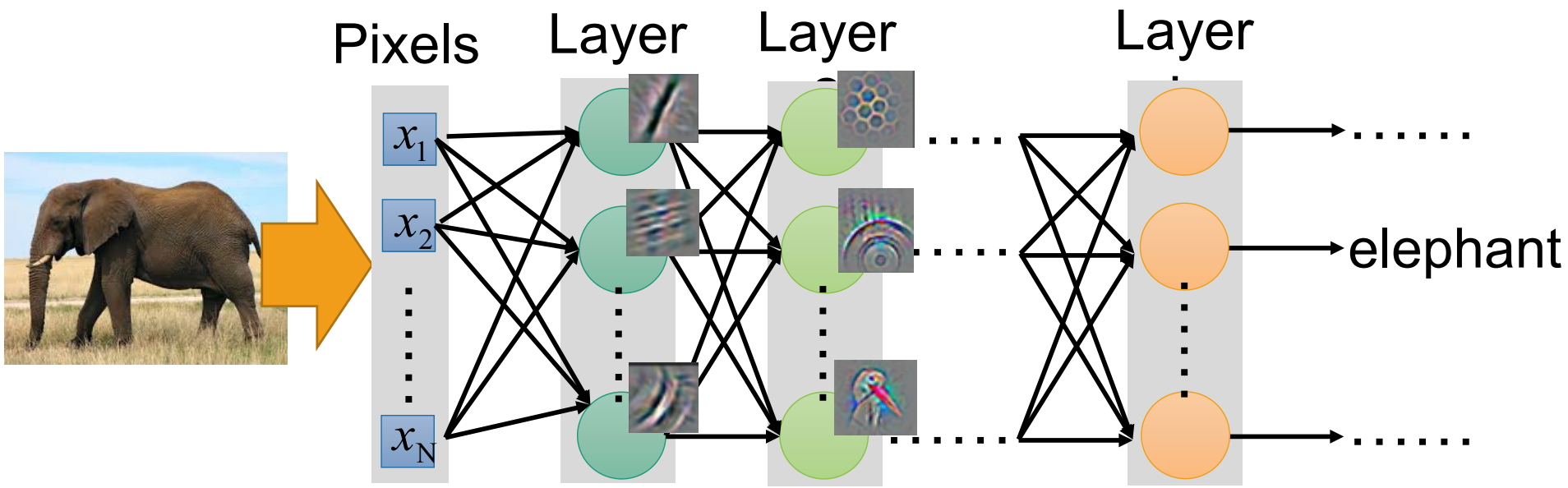
- Challenge: only limited target data, careful about overfitting
- Solution: Freeze the gradient's update for AlexNet part
- Other solution: use smaller gradient's update for AlexNet part



Layer Transfer

Which layer can be transferred (copied)?

- Speech: usually copy the last few layers
- Image: usually copy the first few layers



Transfer: fine-tuning of a deep model

- Task description
  - Source data: (x^s, y^s) ⇐ A large amount
  - Target data: (x^t, y^t) ⇐ Very little

Rq: One-shot learning: only a few examples in target domain

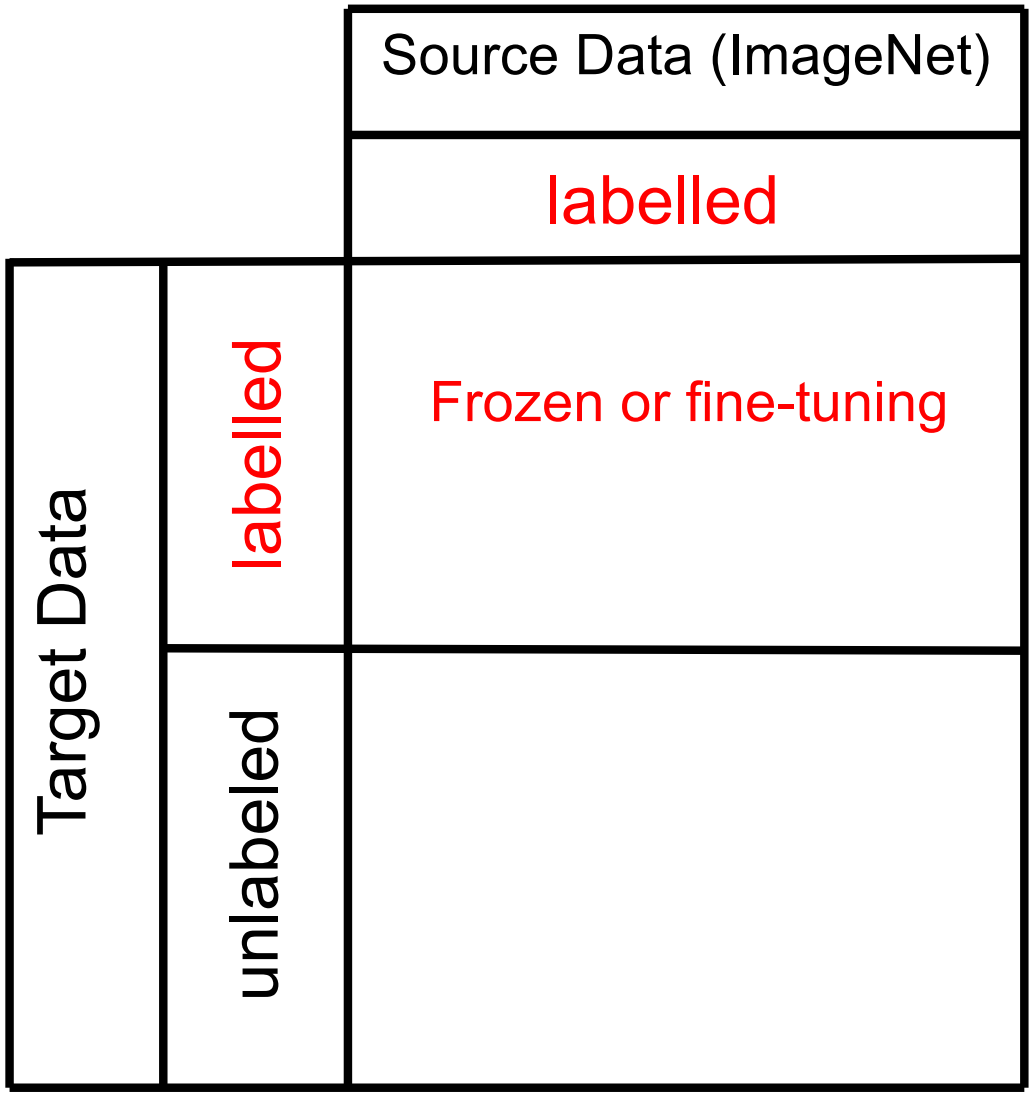
Many different contexts:

In vision: from ImageNet to small datasets

In speech: (supervised) speaker adaption

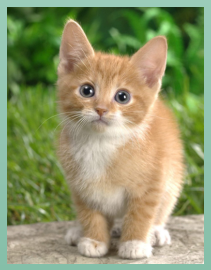
- Source data: audio data and transcriptions from many speakers
- Target data: audio data and its transcriptions of specific user

More on transfer framework



General Framework for Transfer Learning

Dog/Cat Classifier



cat

dog

Data ***not directly related to*** the task considered

ImageNet: Similar domain, different tasks (1000 classes)

General Framework for Transfer Learning

Dog/Cat Classifier

cat

dog

Data ***not directly related to*** the task considered

elephant

tiger

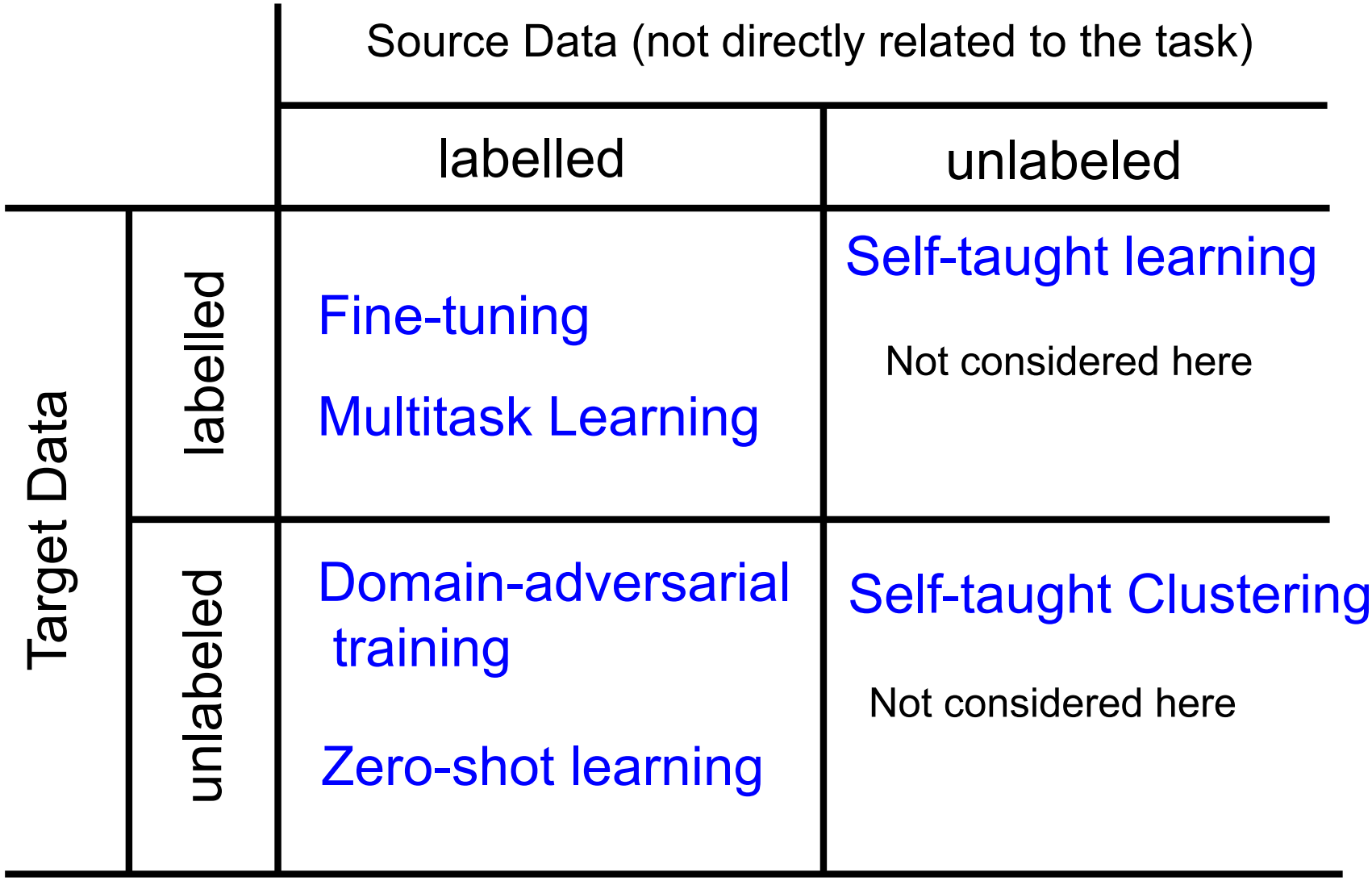
Similar domain, completely different tasks

dog

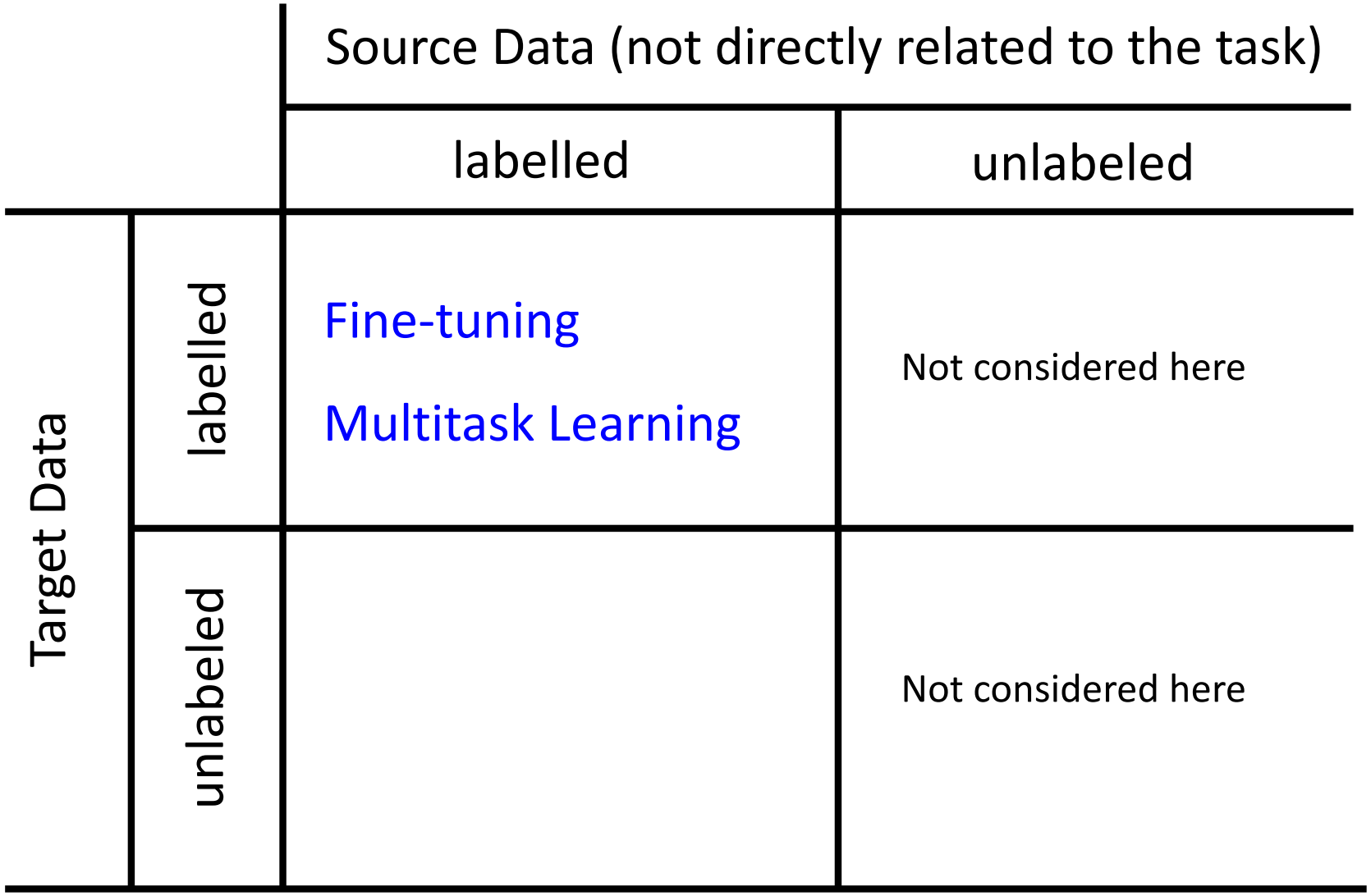
cat

Different domains, same task

General Framework for Transfer Learning

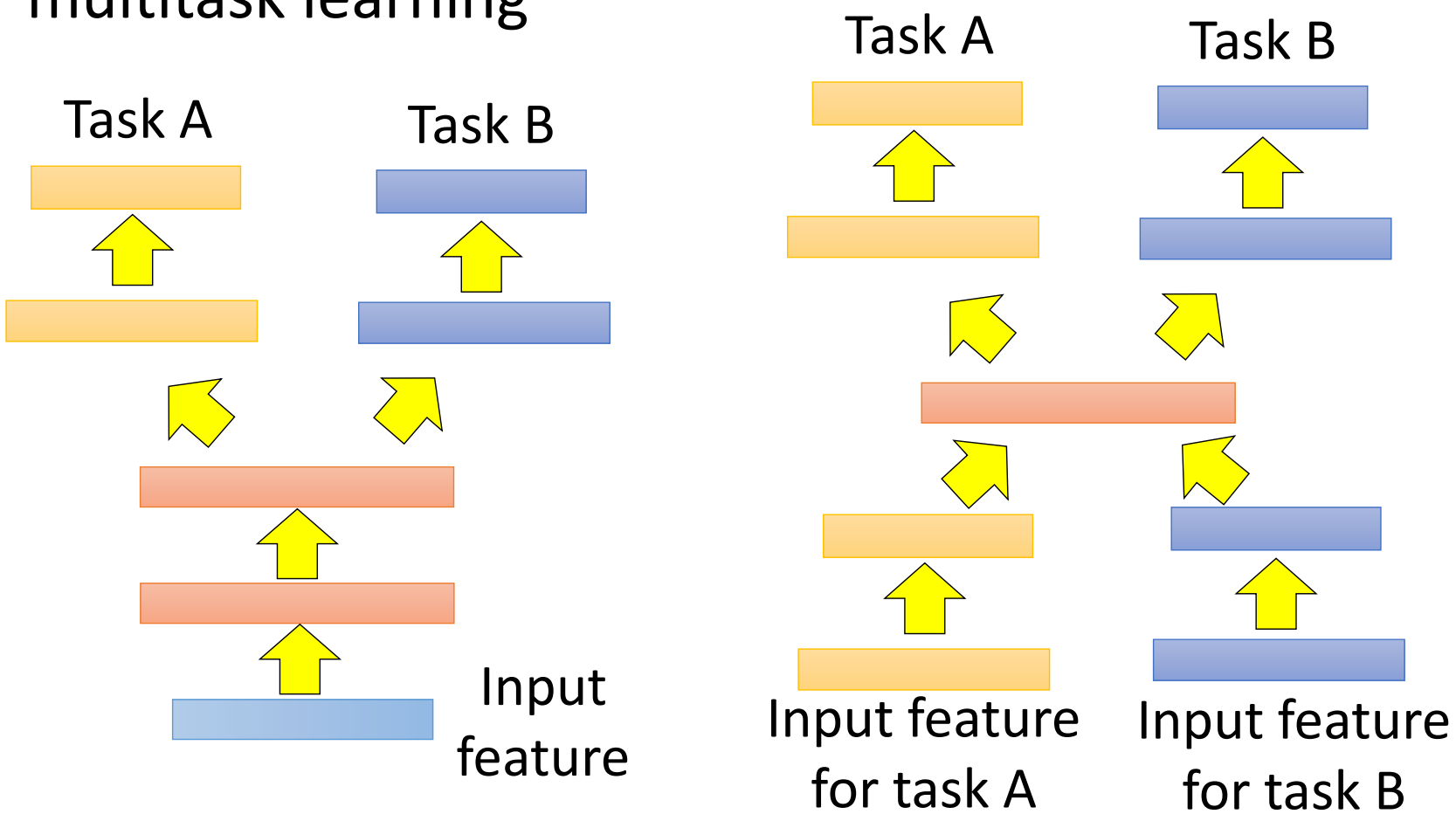


Transfer Learning - Overview

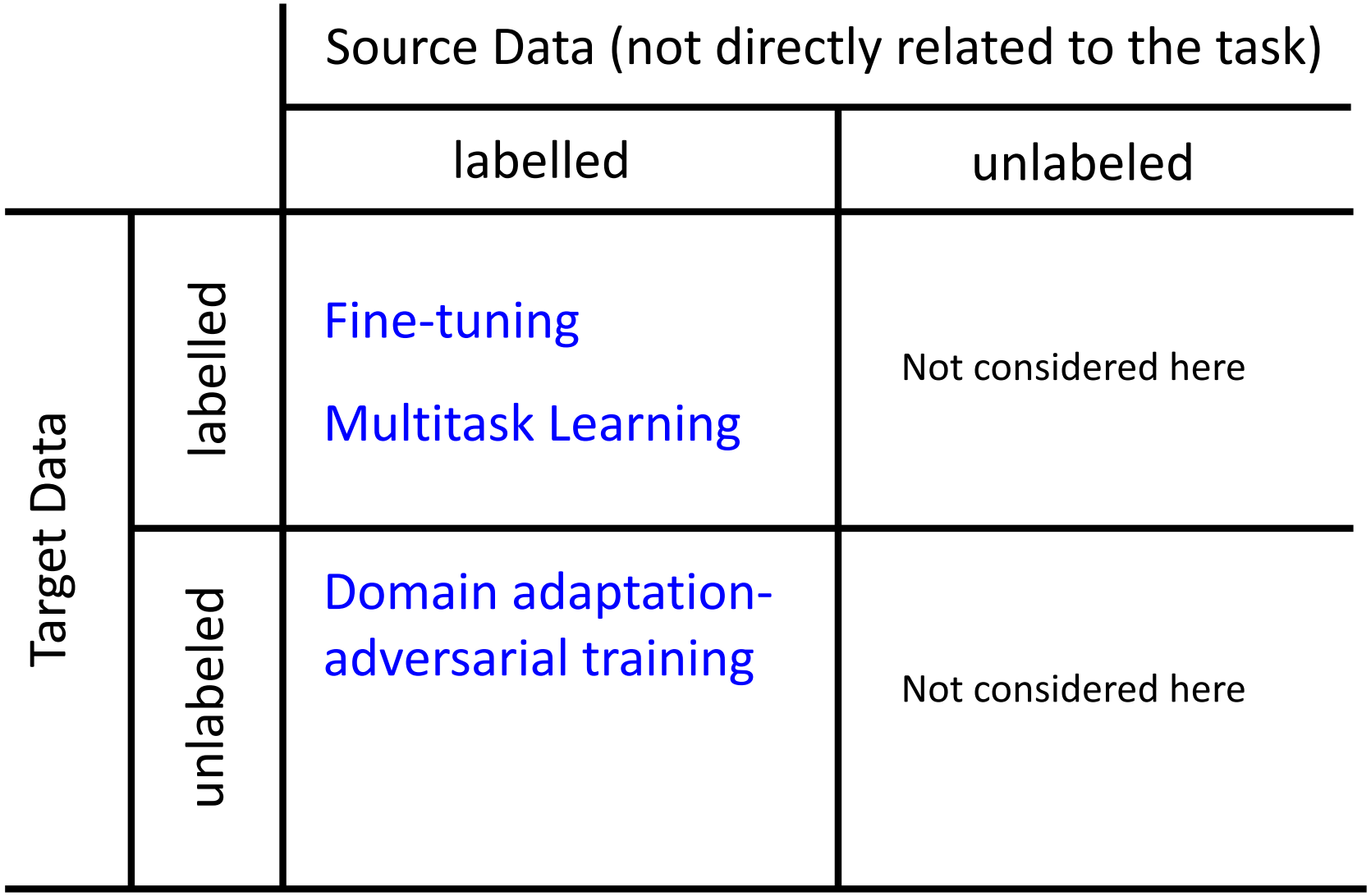


Multitask Learning

- The multi-layer structure makes NN suitable for multitask learning



Transfer Learning - Overview



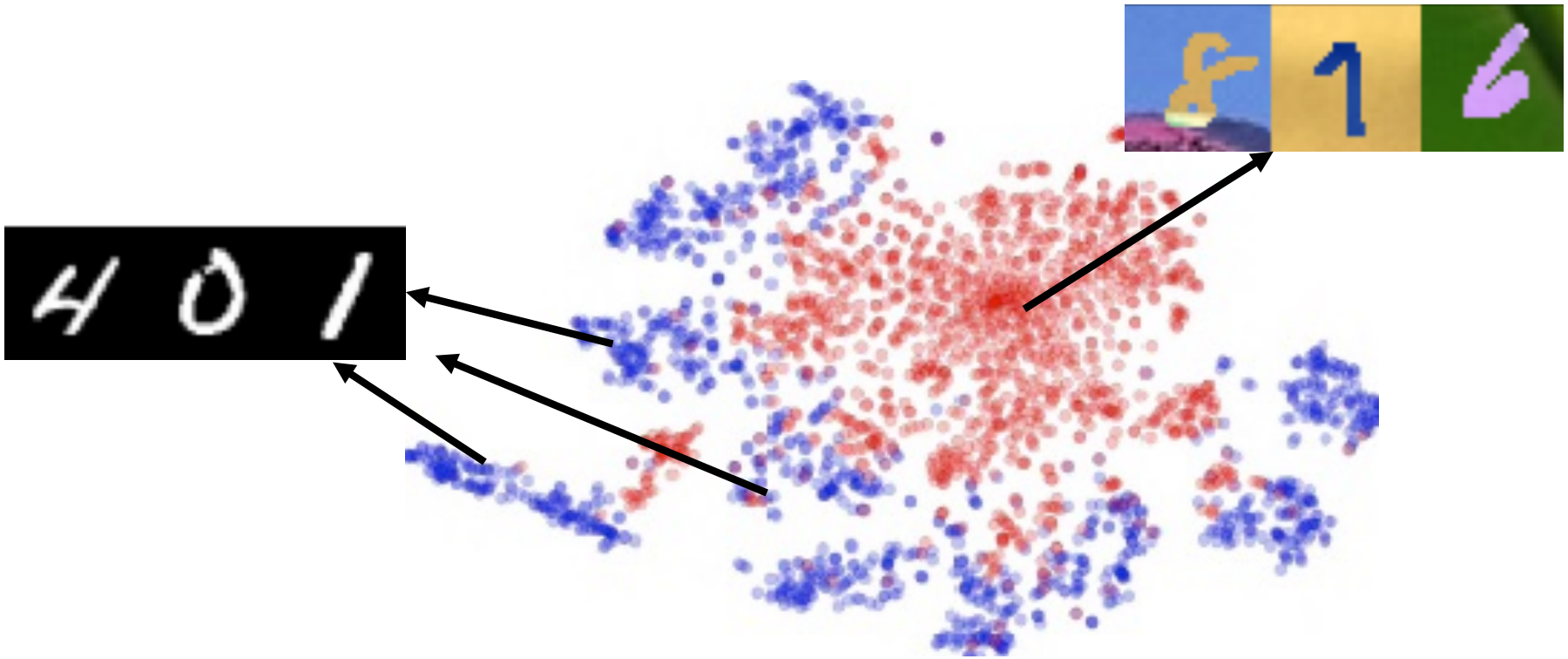
Task description

- Source data: (x^s, y^s) → Training data
- Target data: (x^t) → Testing data

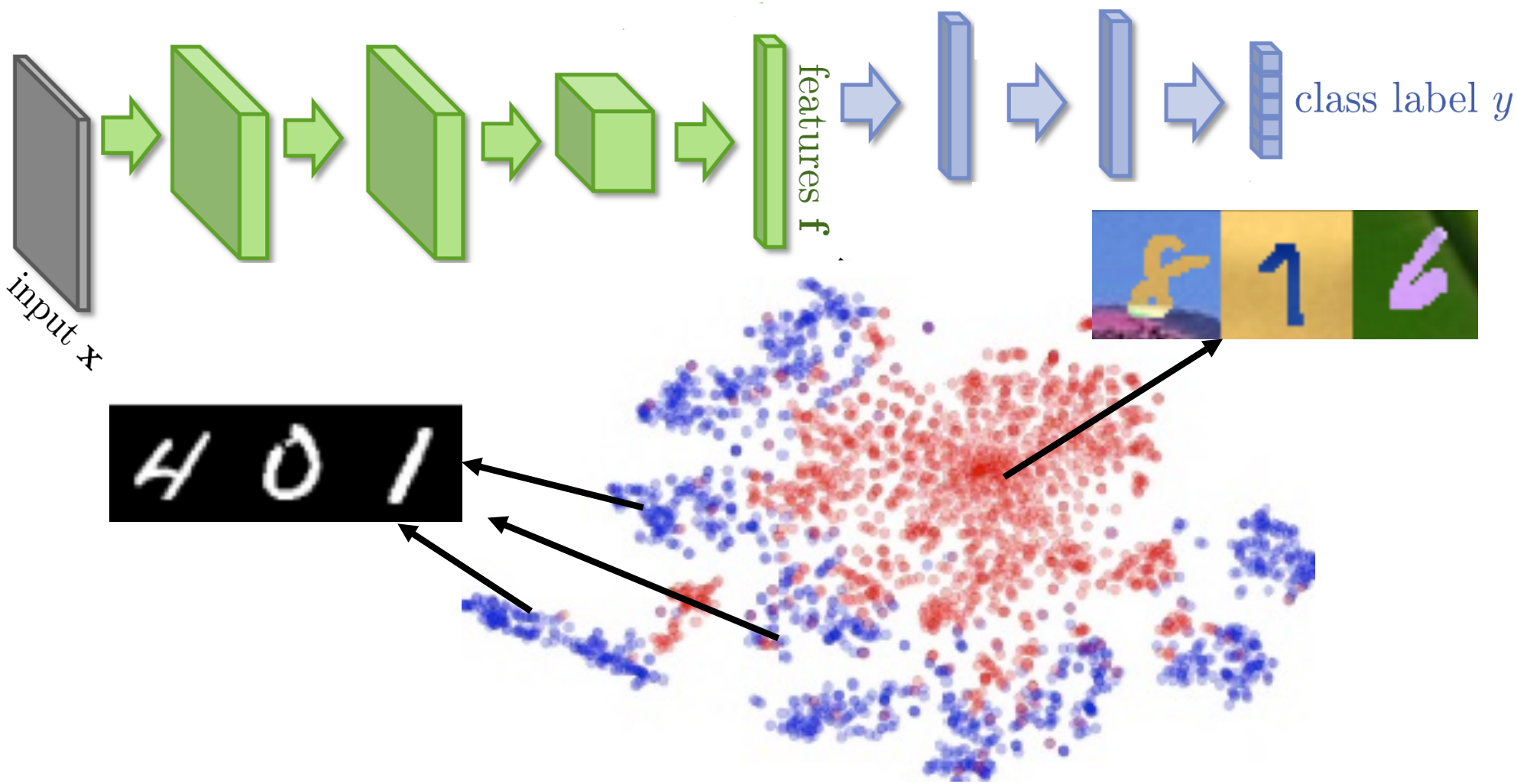
} Same task, mismatch

Domain adaptation

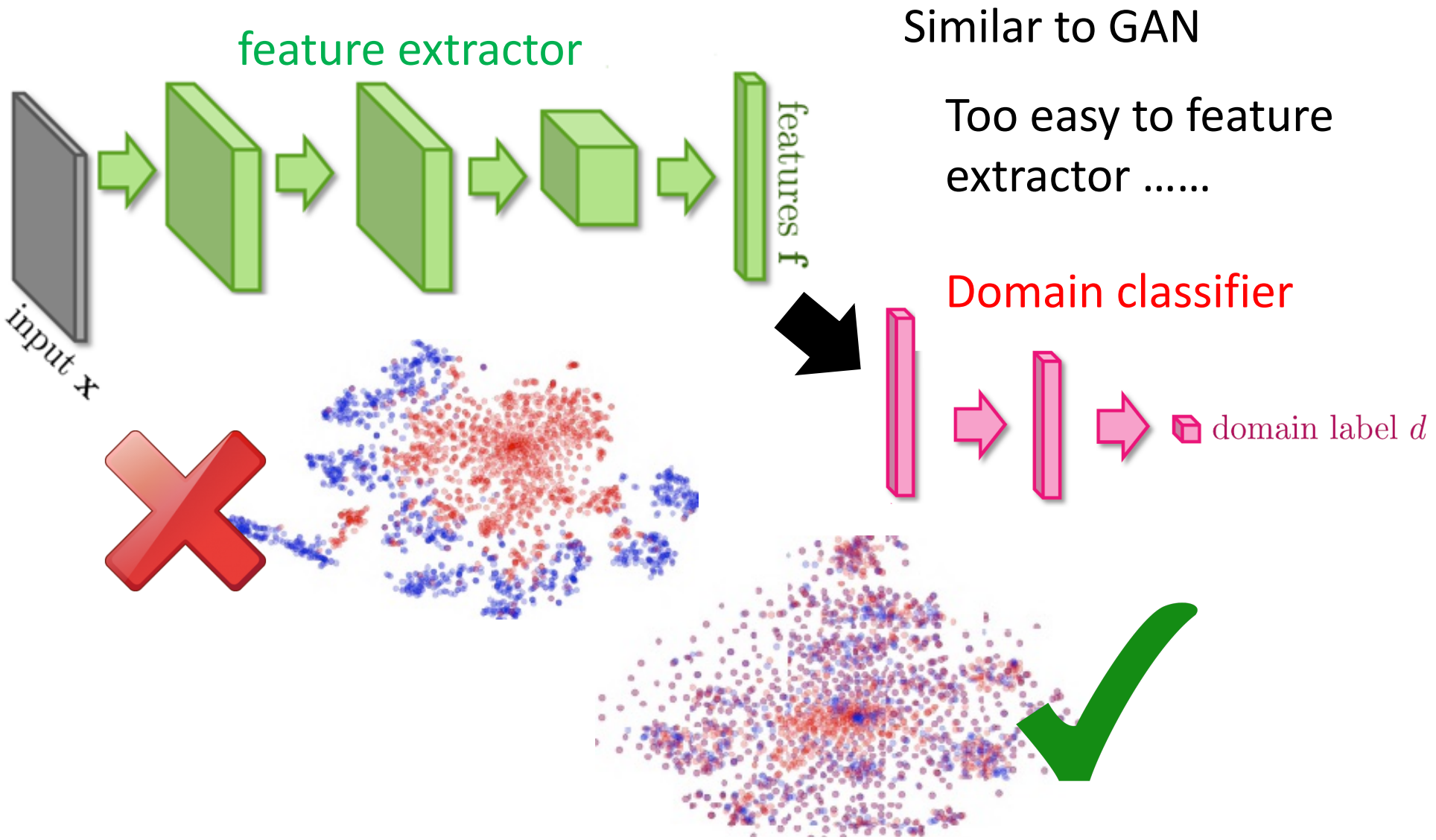
Main principle: diminish the domain shift in the learned features, encourage domain confusion



Domain-adversarial training



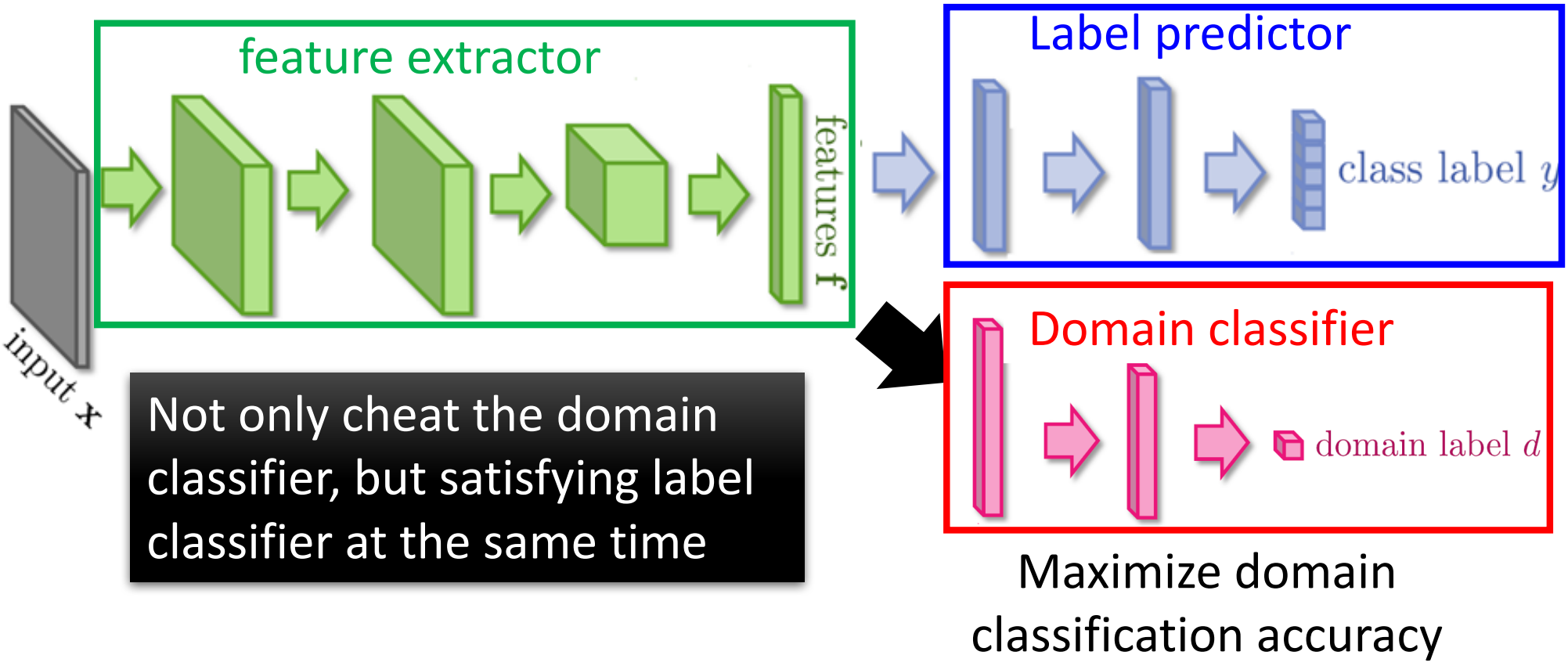
Domain-adversarial training



Domain-adversarial training

Maximize label classification accuracy + minimize domain classification accuracy

Maximize label classification accuracy

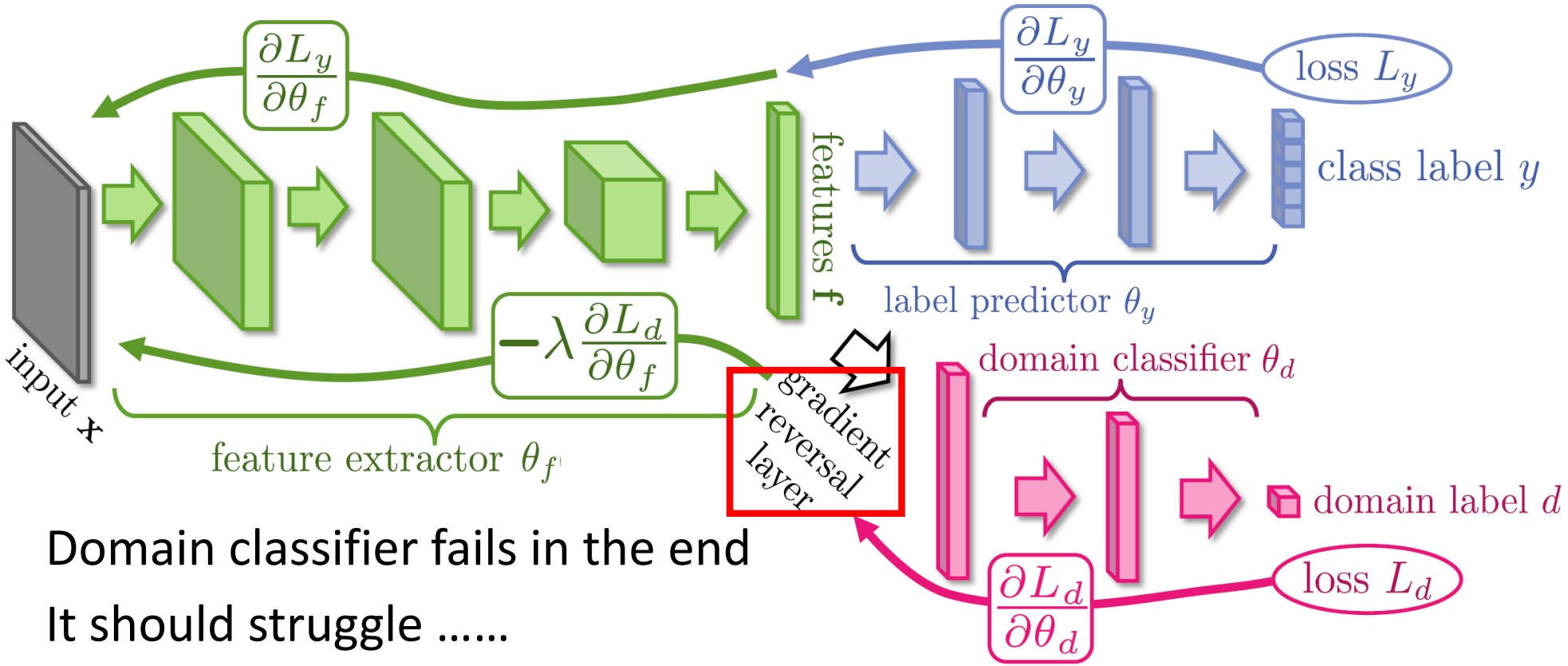


Not only cheat the domain classifier, but satisfying label classifier at the same time

Maximize domain classification accuracy

This is a big network, but different parts have different goals.

Domain-adversarial training



Domain classifier fails in the end

It should struggle ……

Yaroslav Ganin, Victor Lempitsky, Unsupervised Domain Adaptation by Backpropagation, ICML, 2015

Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, Domain-Adversarial Training of Neural Networks, JMLR, 2016

Domain-adversarial training

METHOD	SOURCE / TARGET	MNIST / MNIST-M	SYN NUMBERS / SVHN	SVHN / MNIST	SYN SIGNS / GTSRB
SOURCE ONLY		.5749	.8665	.5919	.7400
SA (FERNANDO ET AL., 2013)		.6078 (7.9%)	.8672 (1.3%)	.6157 (5.9%)	.7635 (9.1%)
PROPOSED APPROACH		**.8149** (57.9%)	**.9048** (66.1%)	**.7107** (29.3%)	**.8866** (56.7%)
TRAIN ON TARGET		.9891	.9244	.9951	.9987

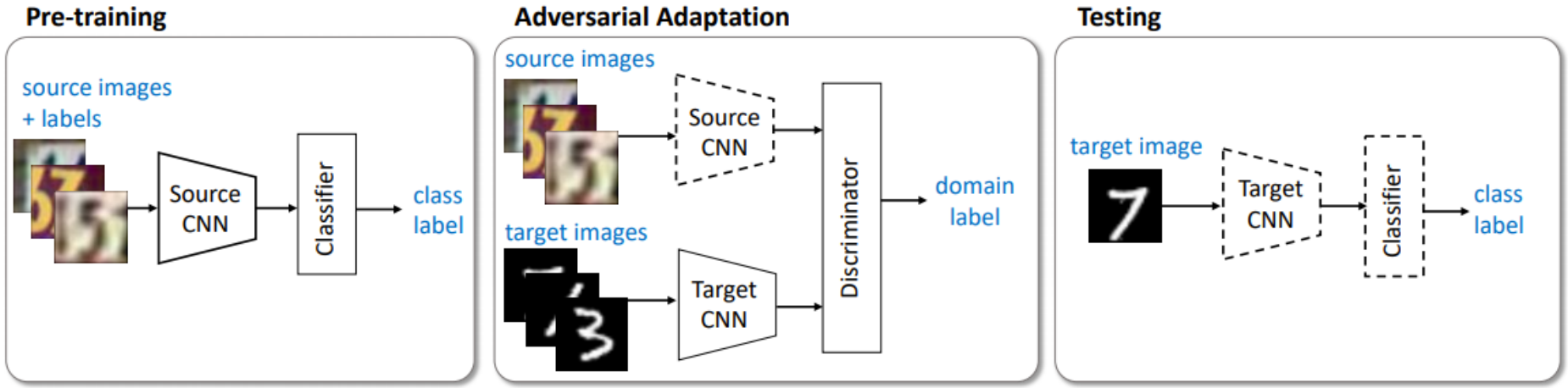
Yaroslav Ganin, Victor Lempitsky, Unsupervised Domain Adaptation by Backpropagation, ICML, 2015

Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, Domain-Adversarial Training of Neural Networks, JMLR, 2016

Domain adaptation

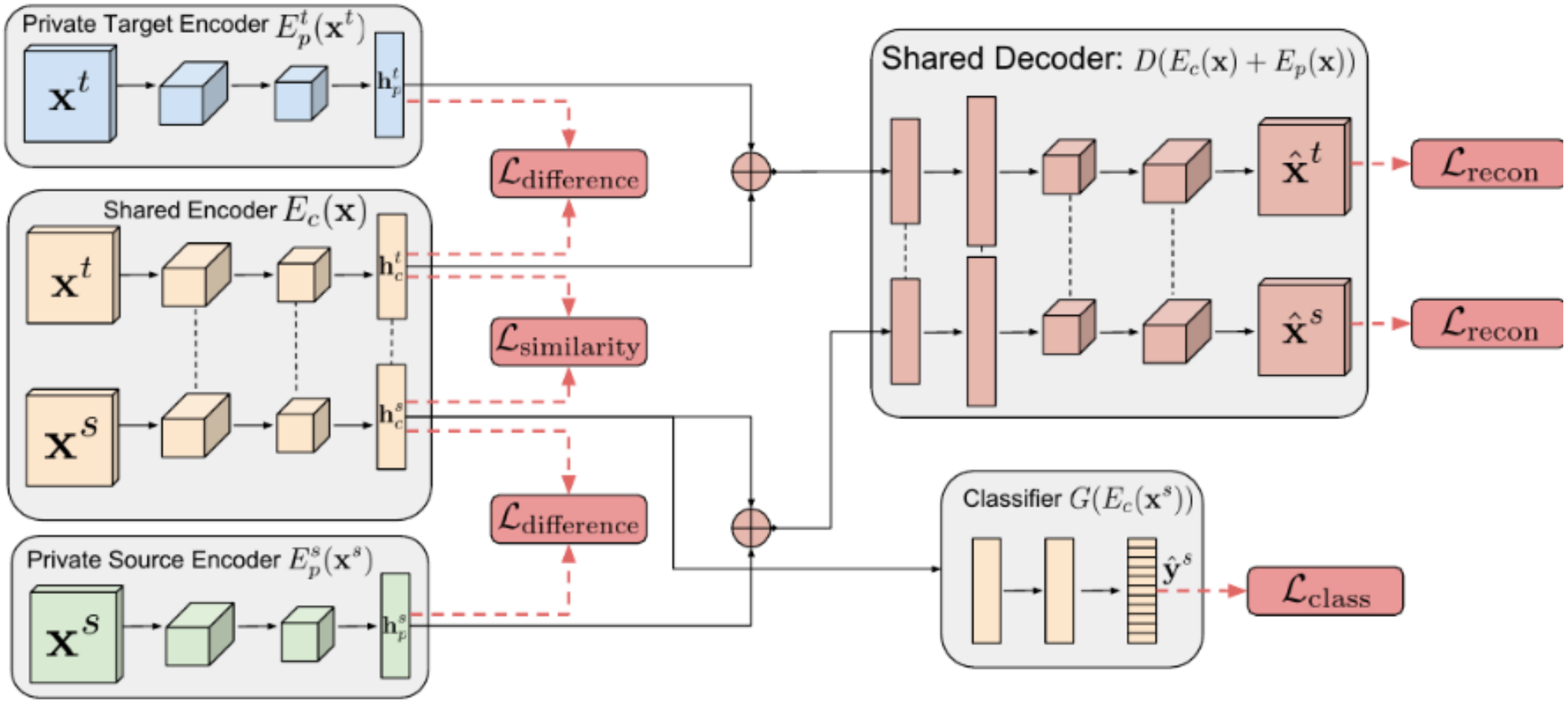
Main principle: diminish the domain shift in the learned features, encourage domain confusion

Another example: Adversarial Discriminative Domain Adaptation [Tzeng et al. 2017]



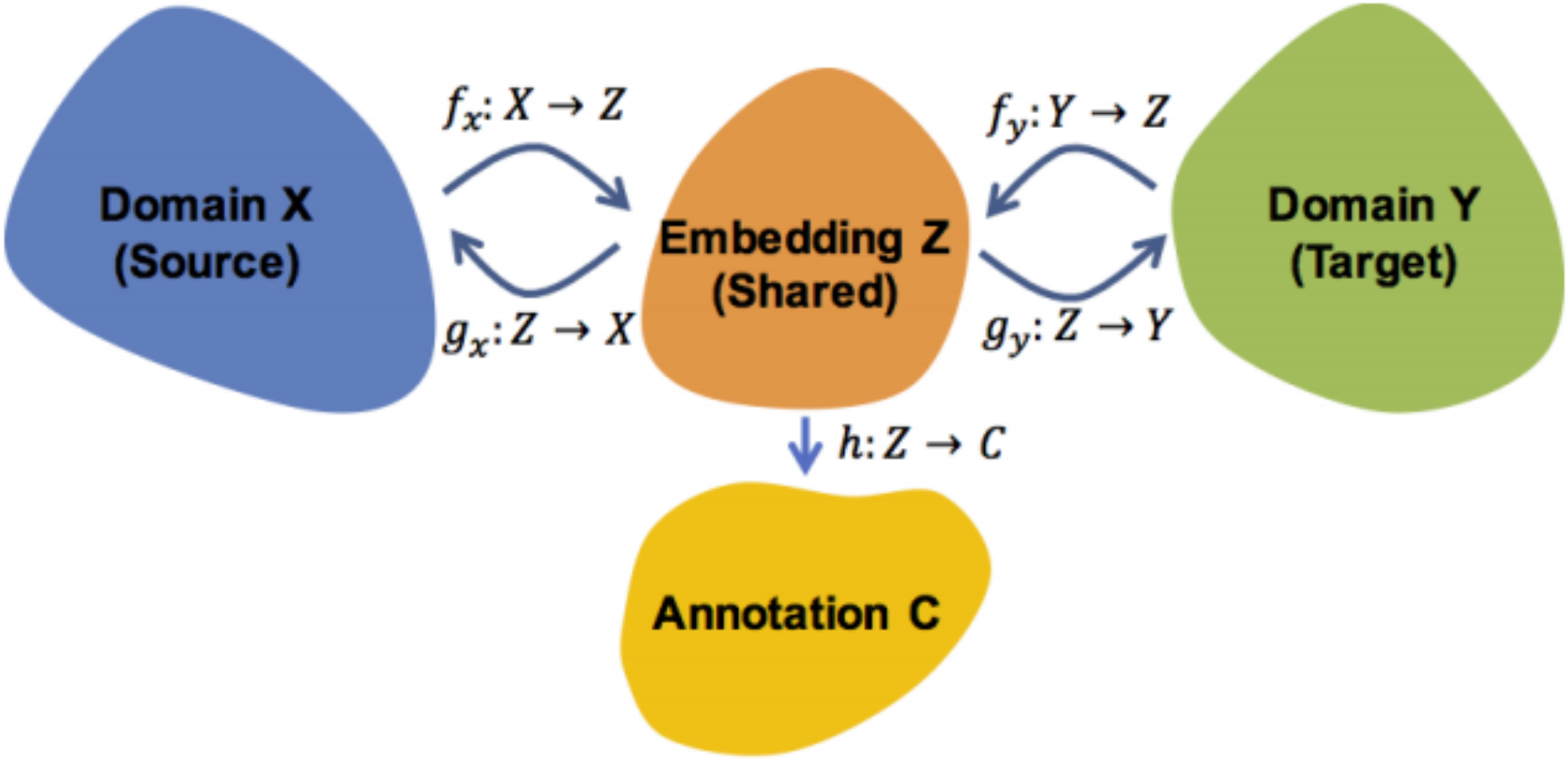
Domain adaptation

Other architecture



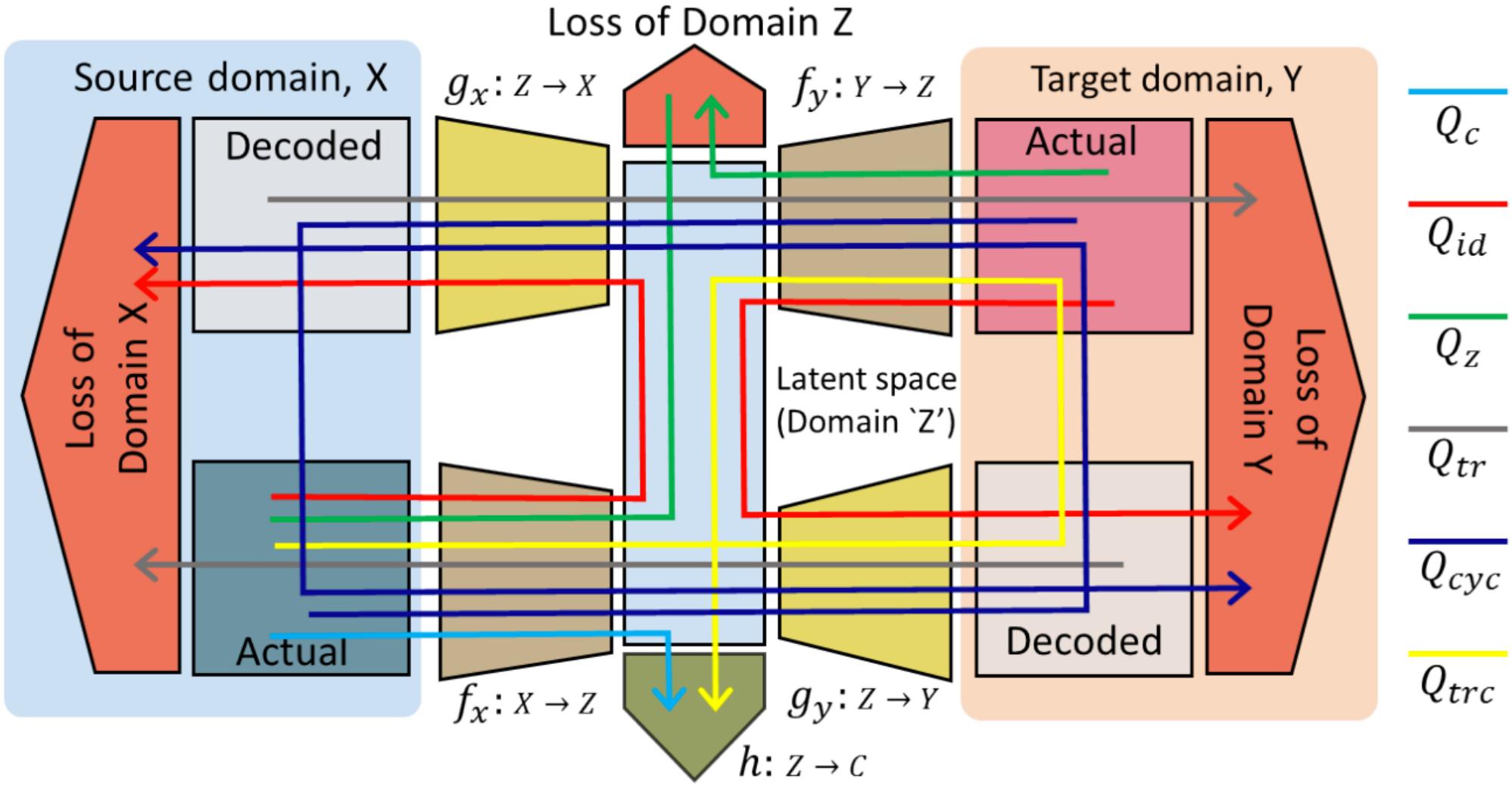
Domain adaptation

Other architecture: Image translation for Domain adaptation [Murez 2017]

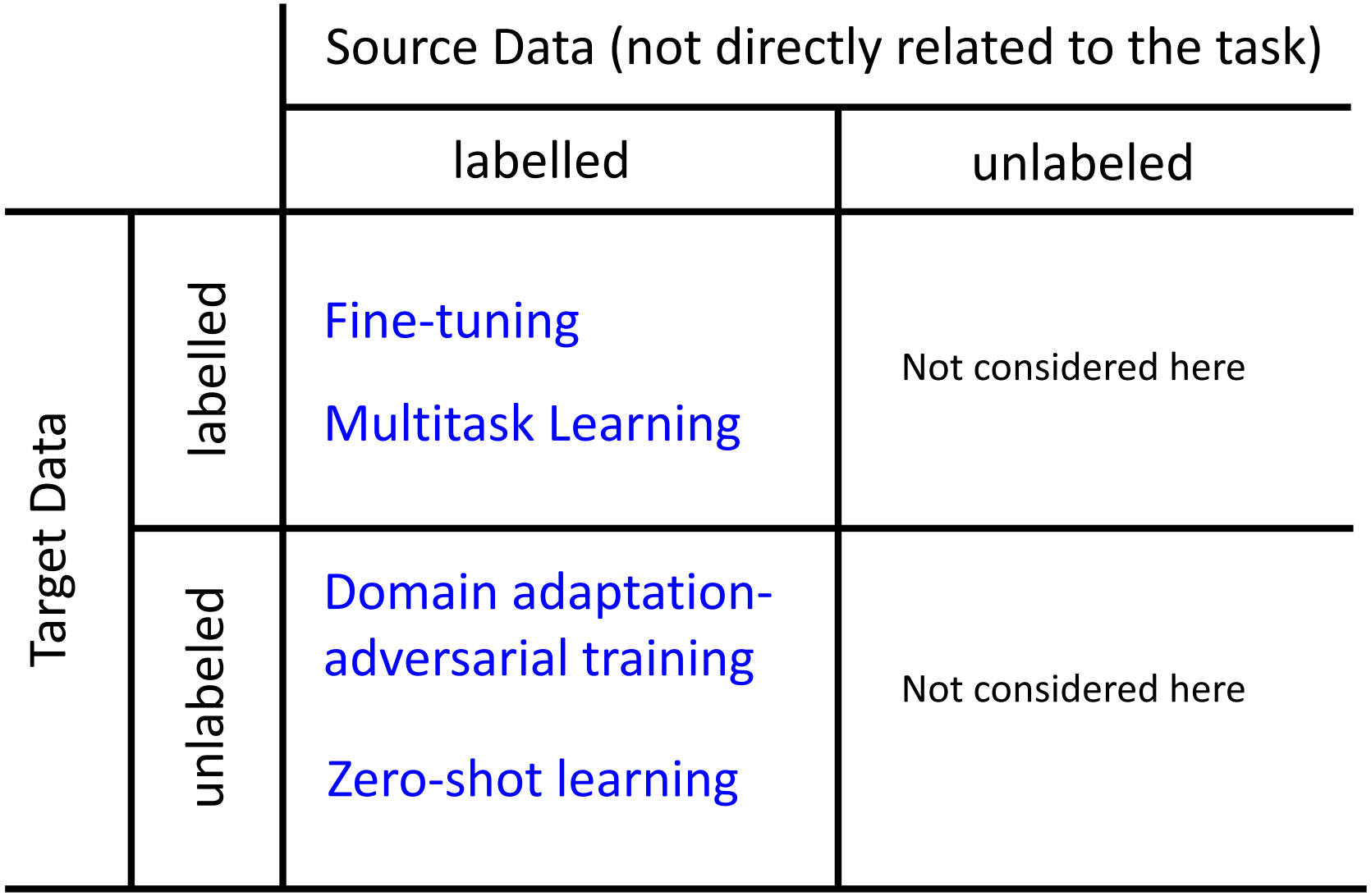


Domain adaptation

Other architecture: Image translation for Domain adaptation [Murez 2017]



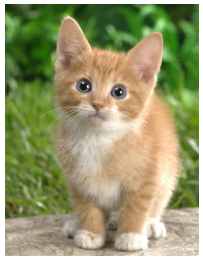
Transfer Learning - Overview



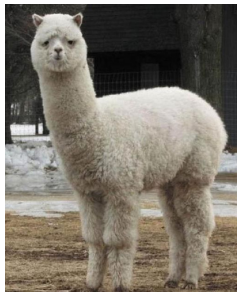
Zero-shot Learning

- Source data: (x^s, y^s) → Training data
- Target data: (x^t) → Testing data

Different tasks

x^s:   ……

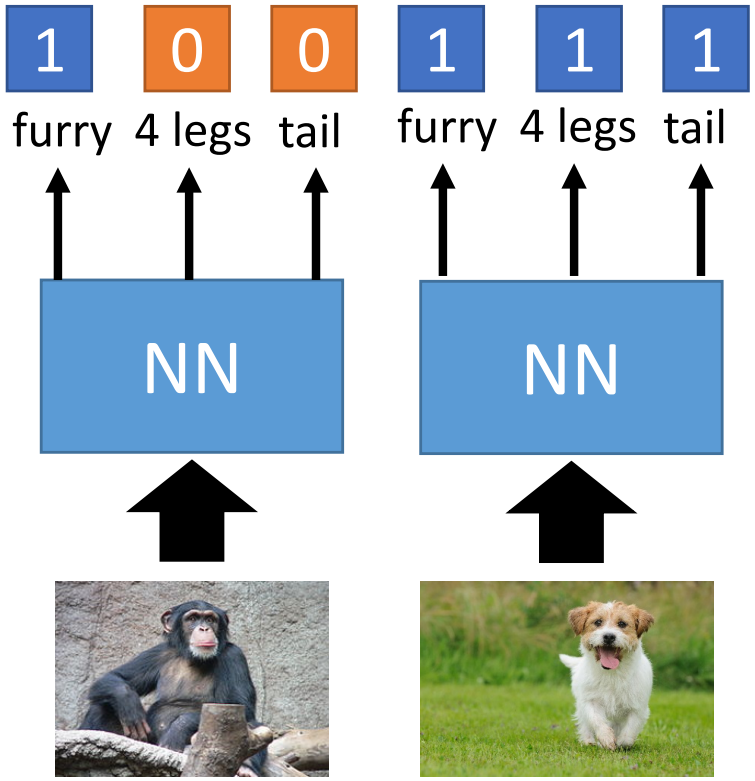
y^s: cat dog ……

x^t :  

Zero-shot Learning

- Representing each class by its attributes

Training



Database

attributes

class

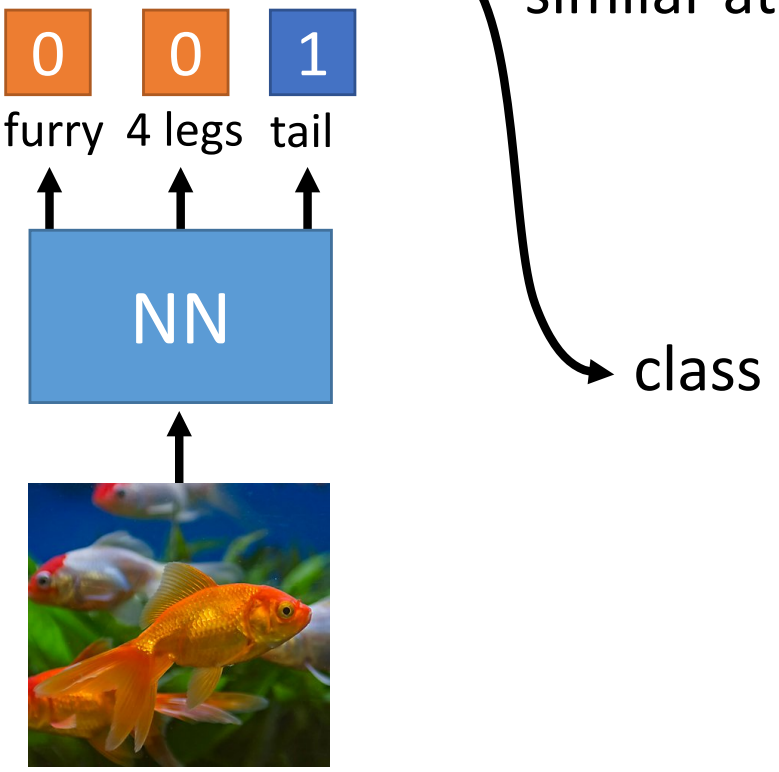
	furry	4 legs	tail	…
Dog	O	O	O	
Fish	X	X	O	
Chimp	O	X	X	
…				

sufficient attributes for one to one mapping

Zero-shot Learning

- Representing each class by its attributes

Testing



Find the class with the most similar attributes

attributes

	furry	4 legs	tail	…
Dog	O	O	O	
Fish	X	X	O	
Chimp	O	X	X	
…				

sufficient attributes for one to one mapping

Zero-shot Learning

What if we don't have attribute database

- Attribute embedding + word embedding

Zero-shot Learning

$f(*)$ and $g(*)$ can be NN.

Training target:

$f(x^n)$ and $g(y^n)$ as close as possible

- Attribute embedding