Lecture 14

Representation Learning

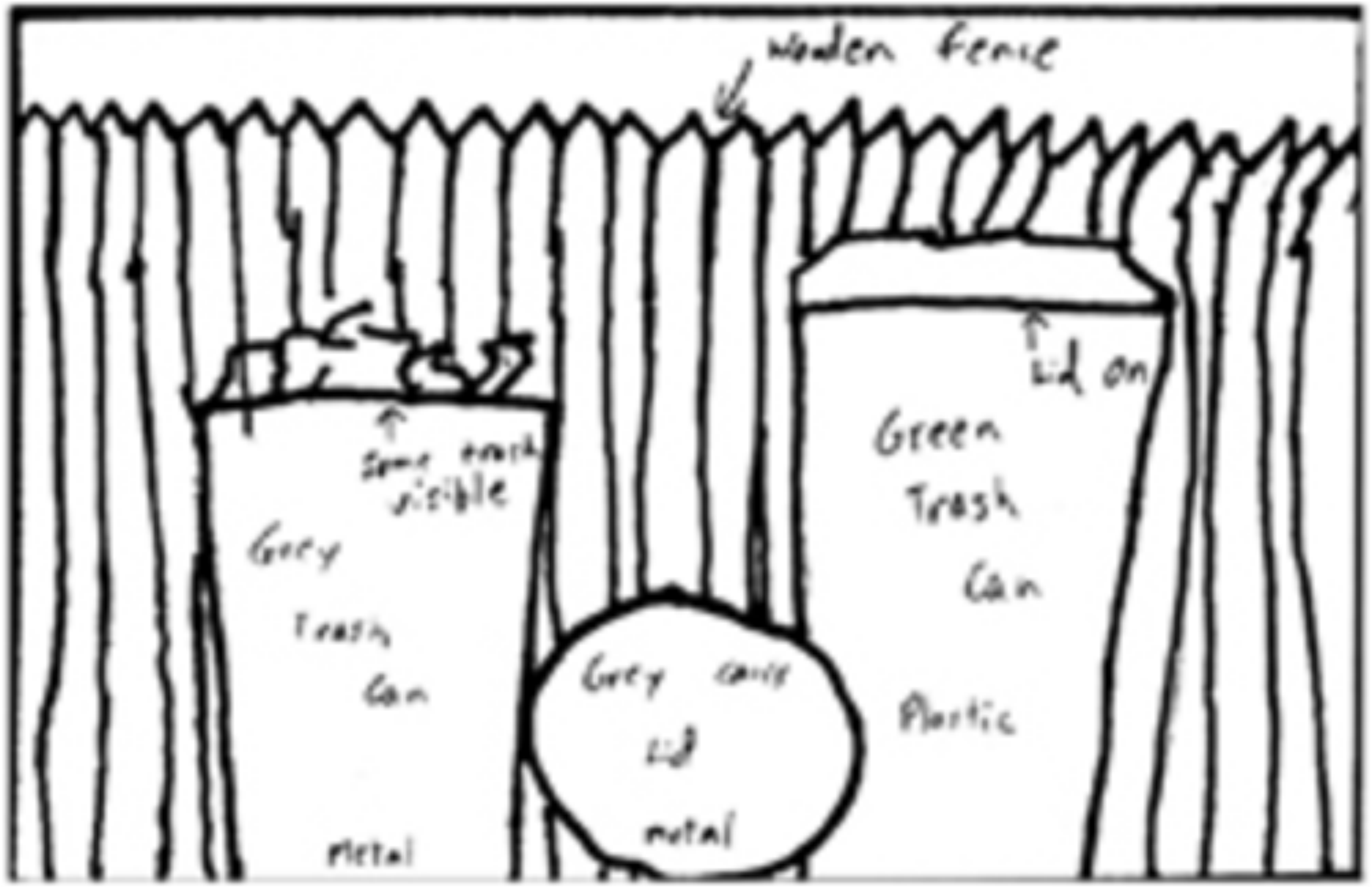
Bill Freeman, Phillip Isola

14. Representation Learning

- Representations in the brain
- What is learned by a deep net?
- Transfer learning and finetuning
- Unsupervised and self-supervised learning

Observed image

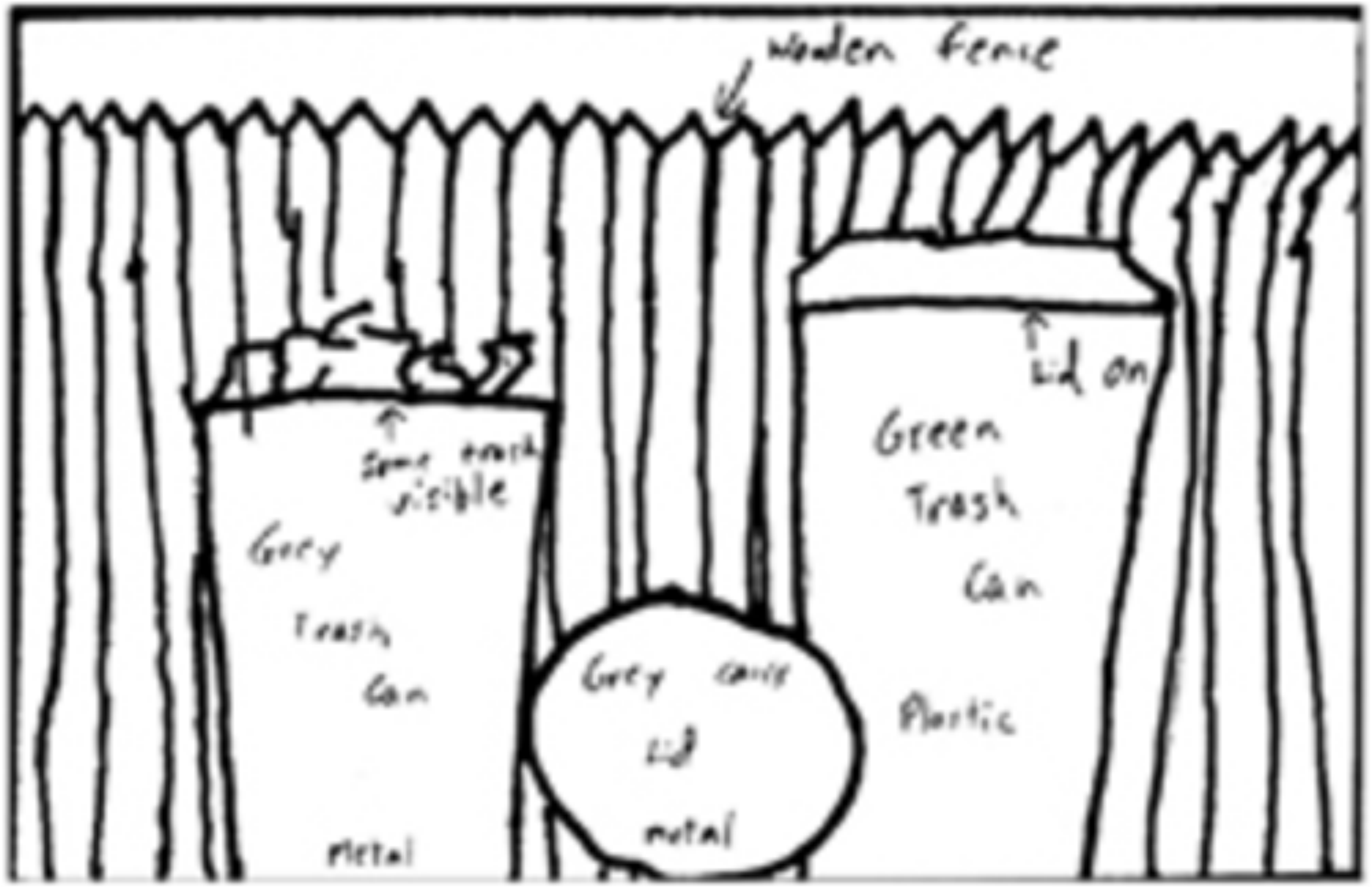
Drawn from memory



[Bartlett, 1932]
[Intraub & Richardson, 1989]

Observed image

Drawn from memory



[Bartlett, 1932]
[Intraub & Richardson, 1989]

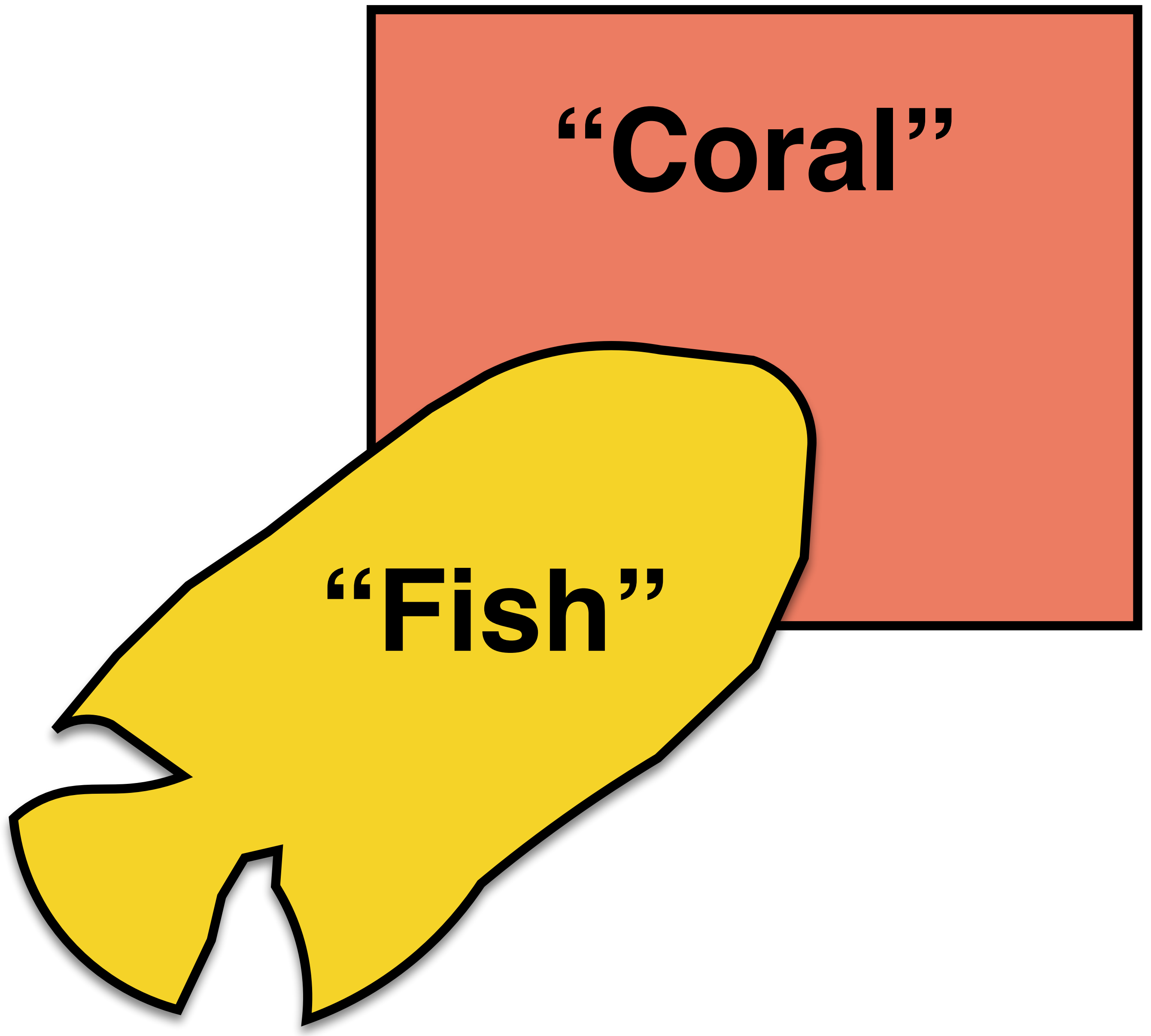
"I stand at the window and see a house, trees, sky. Theoretically I might say there were 327 brightnesses and nuances of colour. Do I have "327"? No. I have sky, house, and trees."

— Max Wertheimer, 1923

Representation learning

$\mathbf{x}$

Image

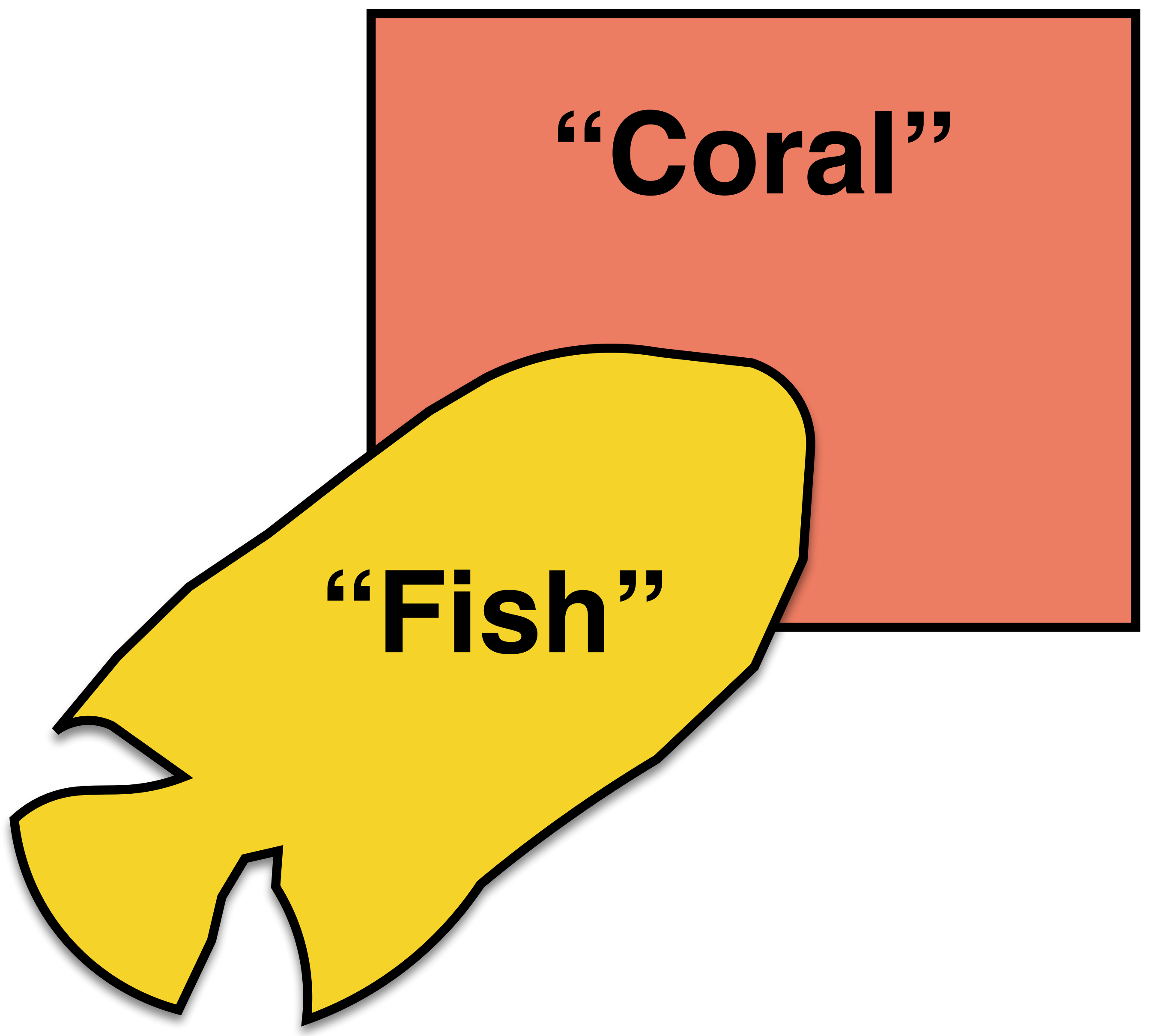


Compact mental representation

Representation learning

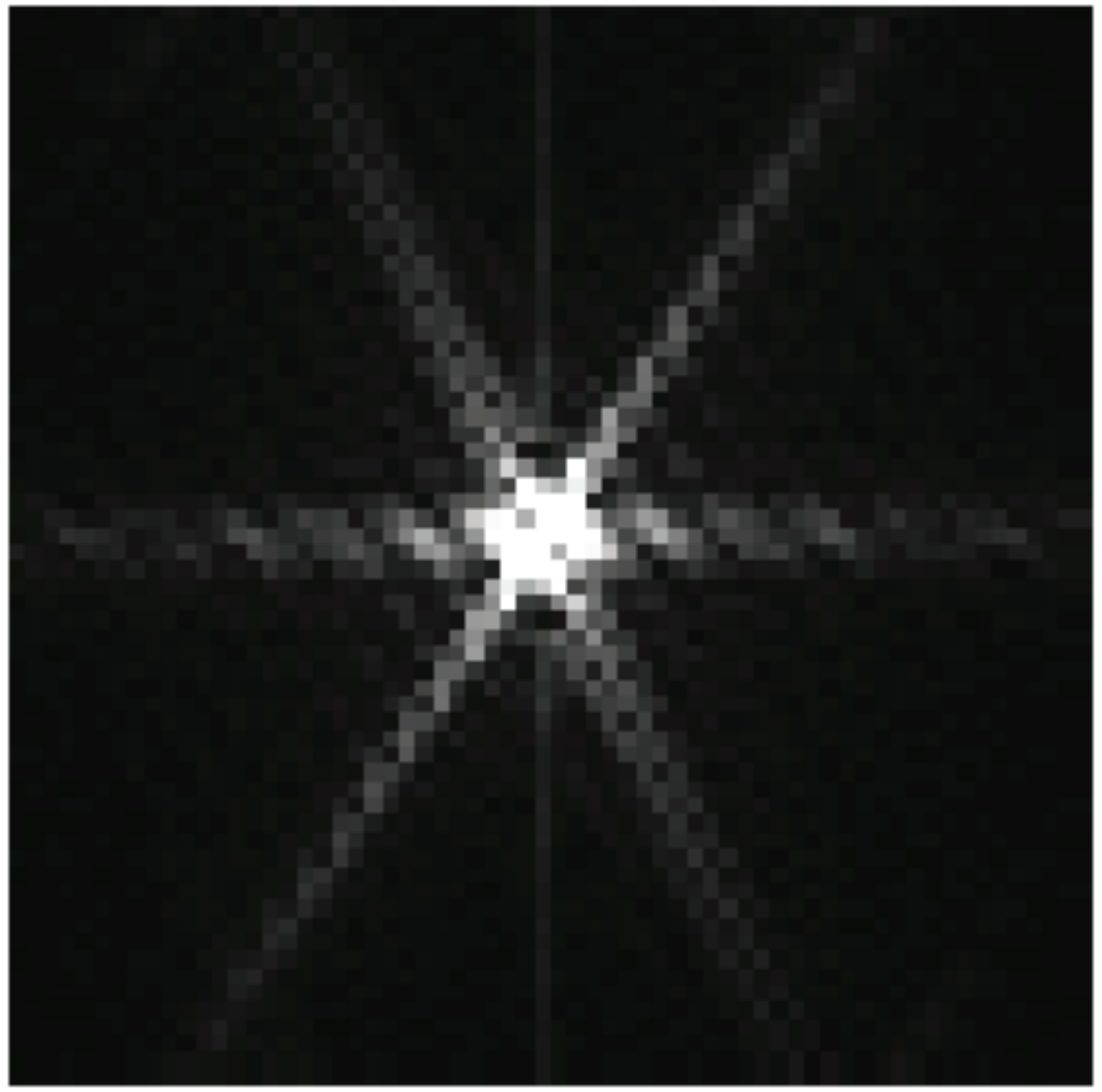
Good representations are:

1. Compact (*minimal*)
2. Explanatory (*sufficient*)
3. Disentangled (*independent factors*)
4. Interpretable
5. *Make subsequent problem solving easy*

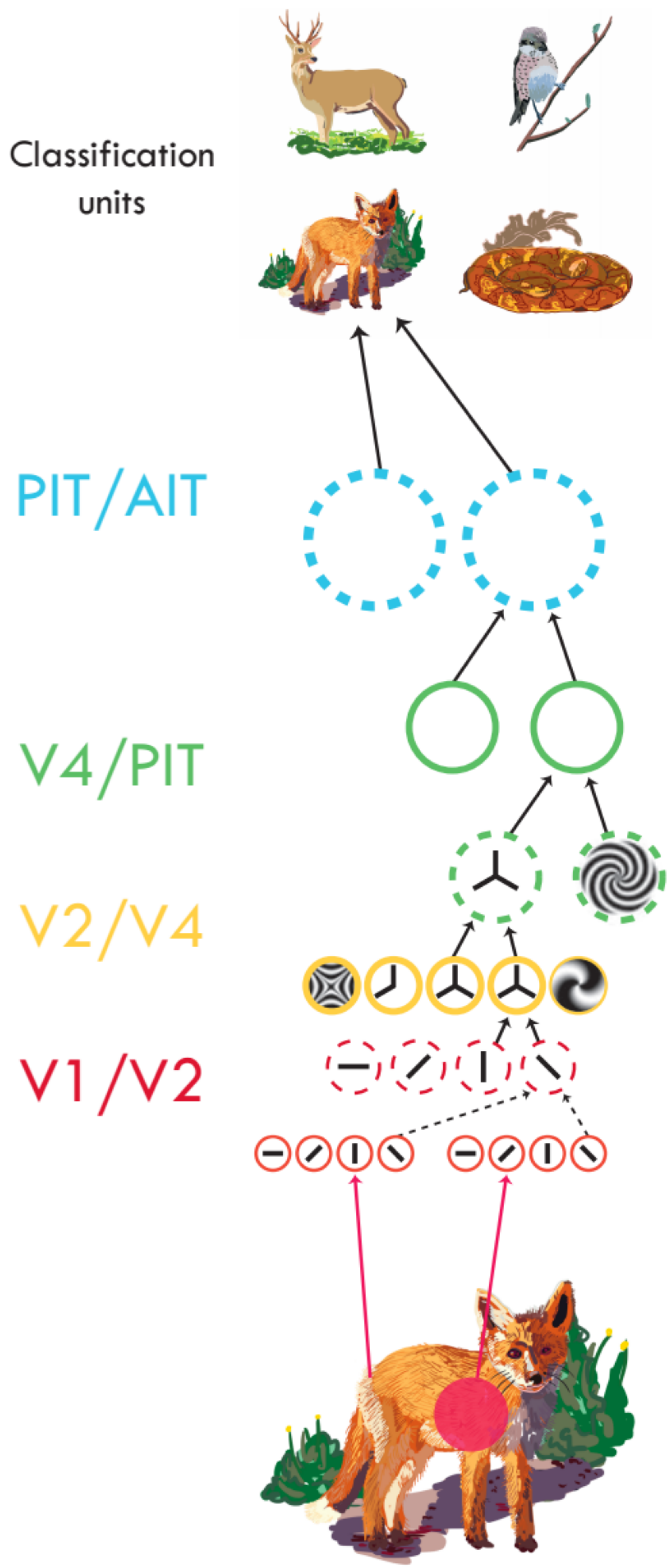


[See “Representation Learning”, Bengio 2013, for more commentary]

Representation learning

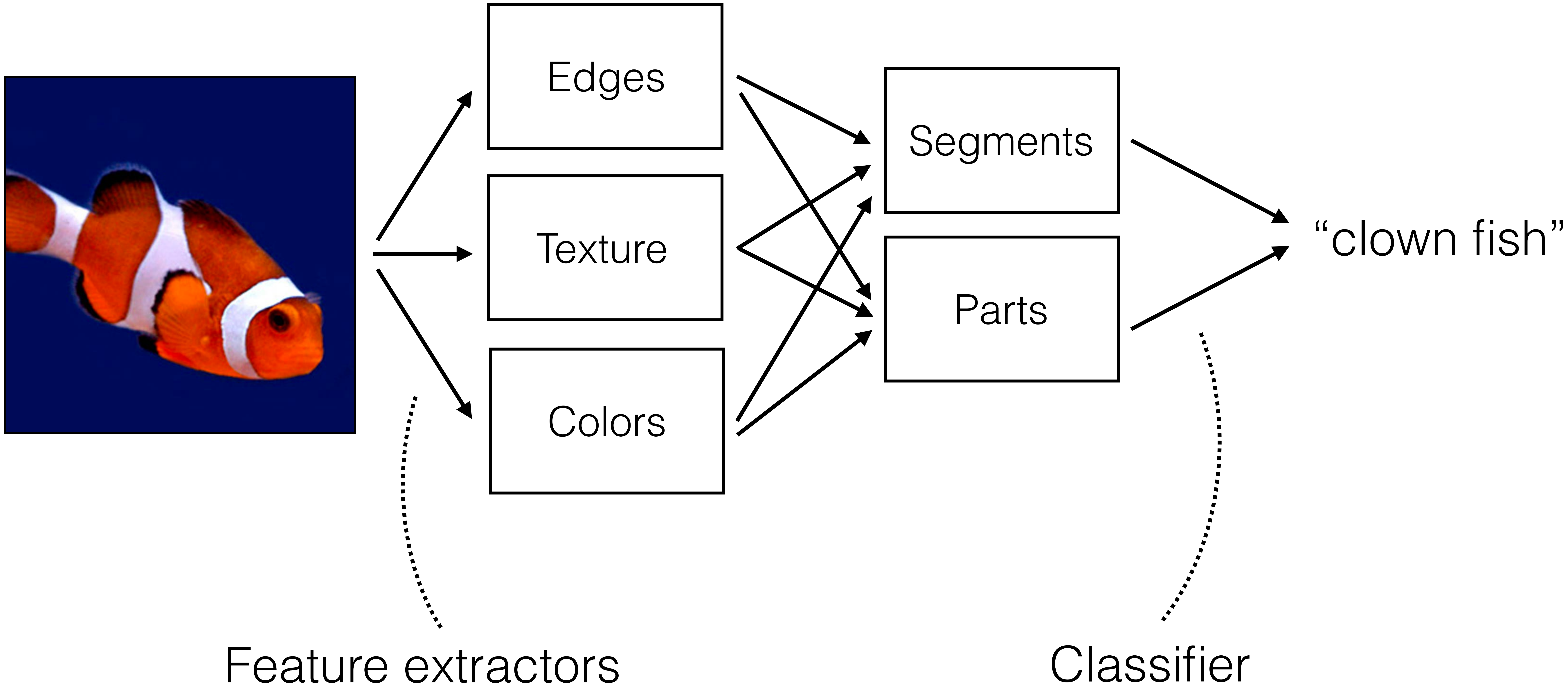


Convolution is pointwise multiplication in the frequency domain.

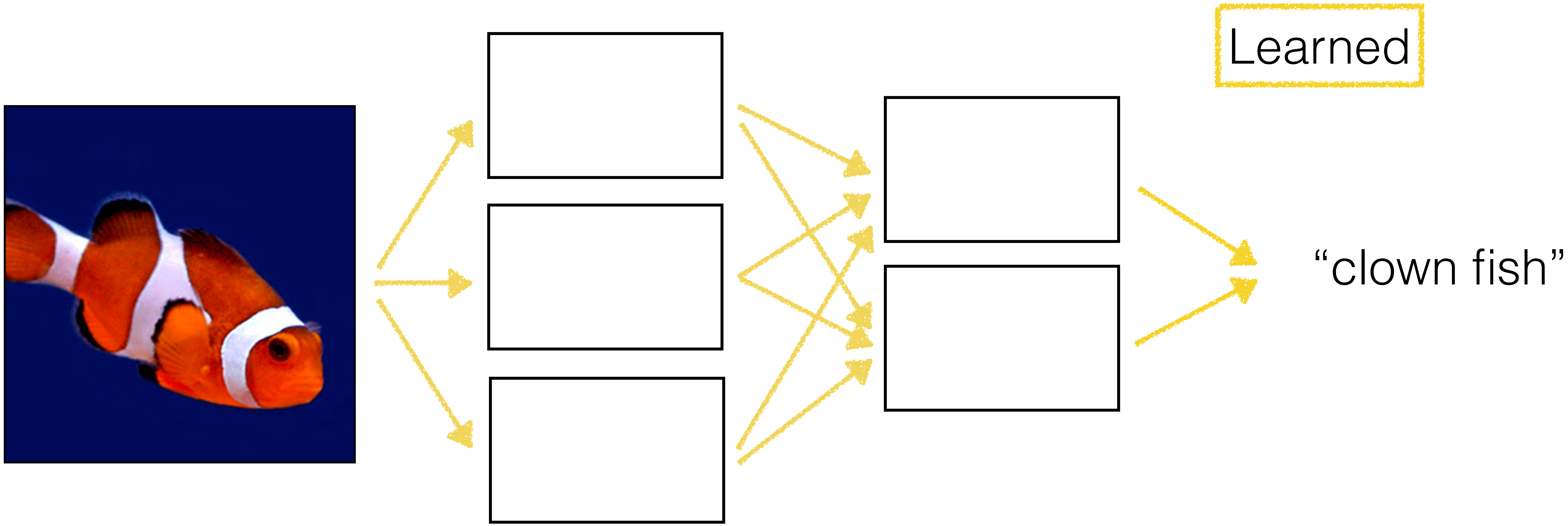


[Serre, 2014]

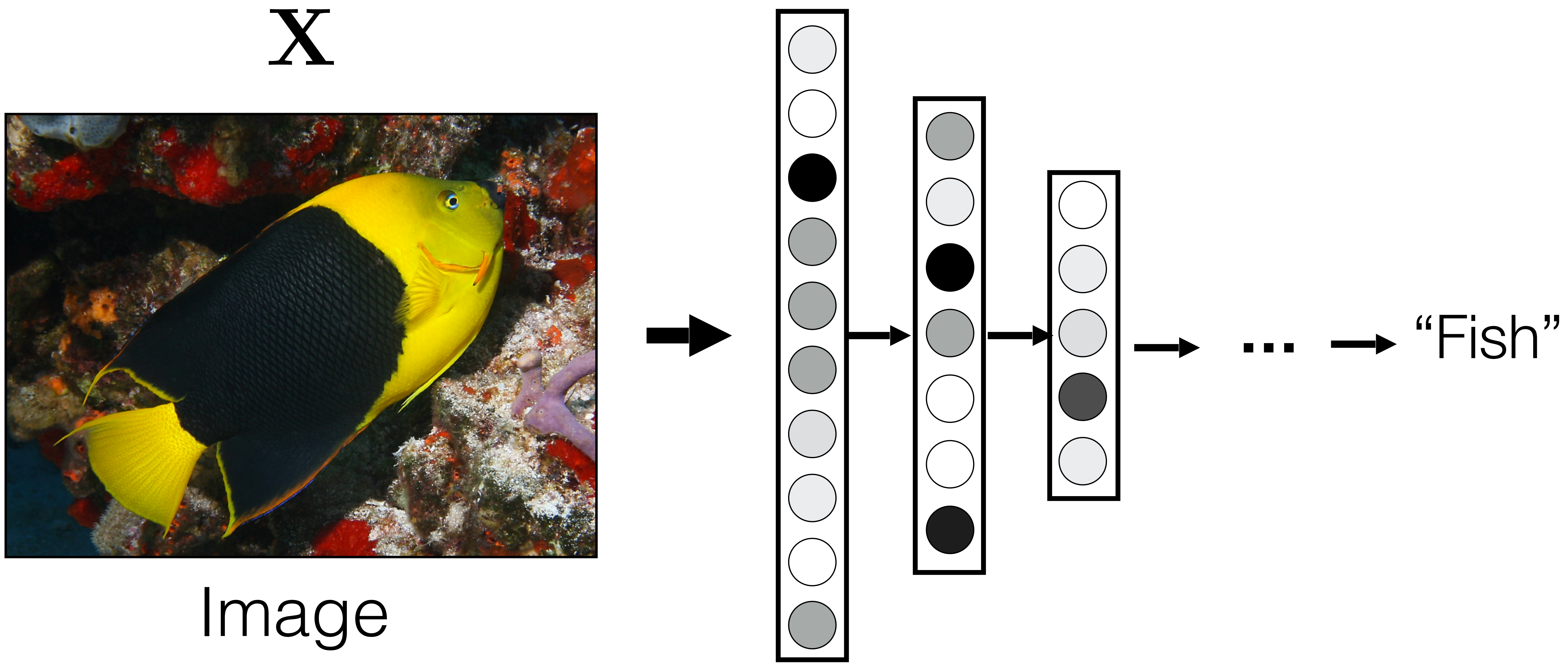
Classical object recognition



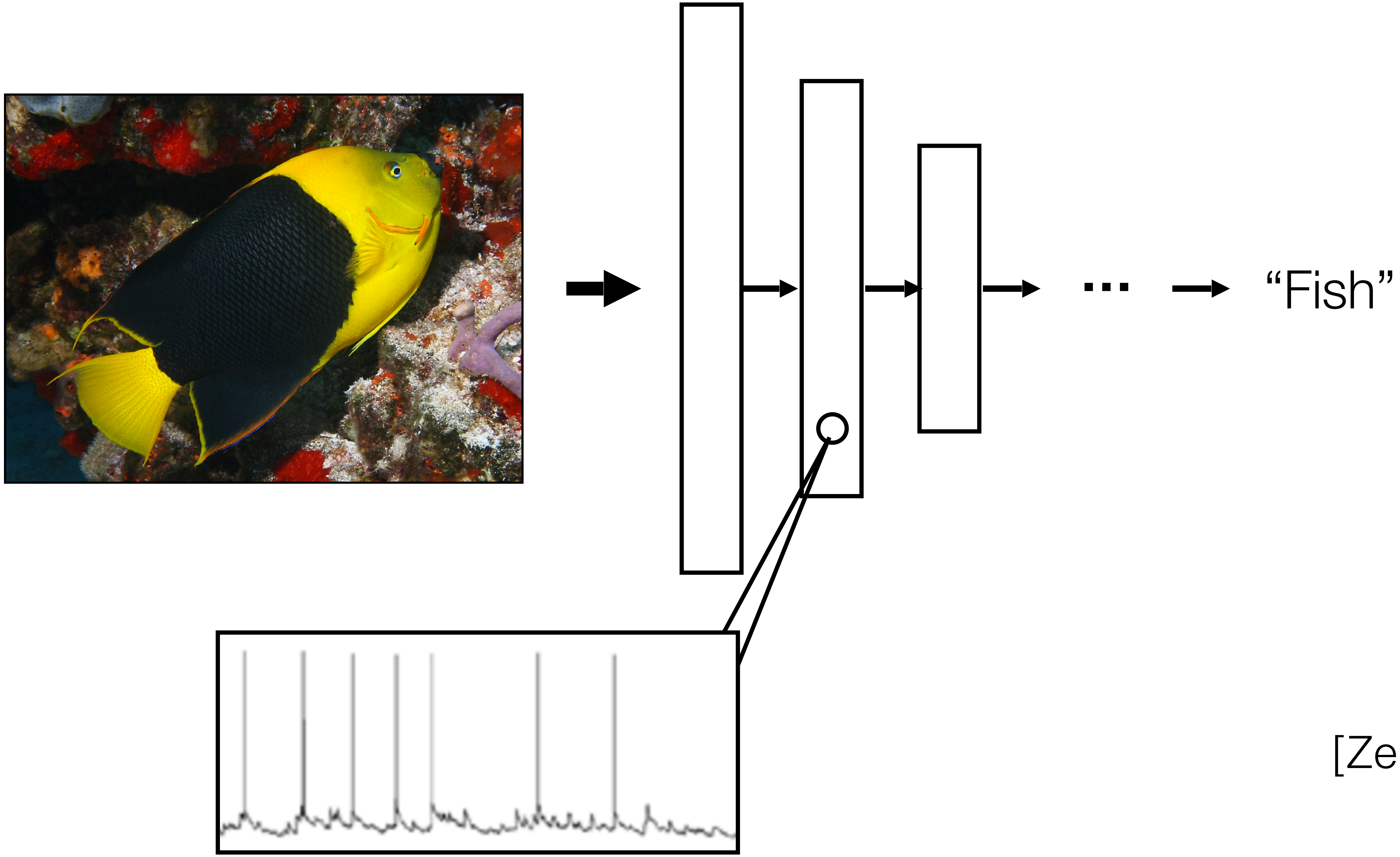
Deep learning



What do deep nets internally learn?



Deep Net "Electrophysiology"



[Zeiler & Fergus, ECCV 2014]
[Zhou et al., ICLR 2015]

Visualizing and Understanding CNNs

[Zeiler and Fergus, 2014]

Gabor-like filters learned by **layer 1**

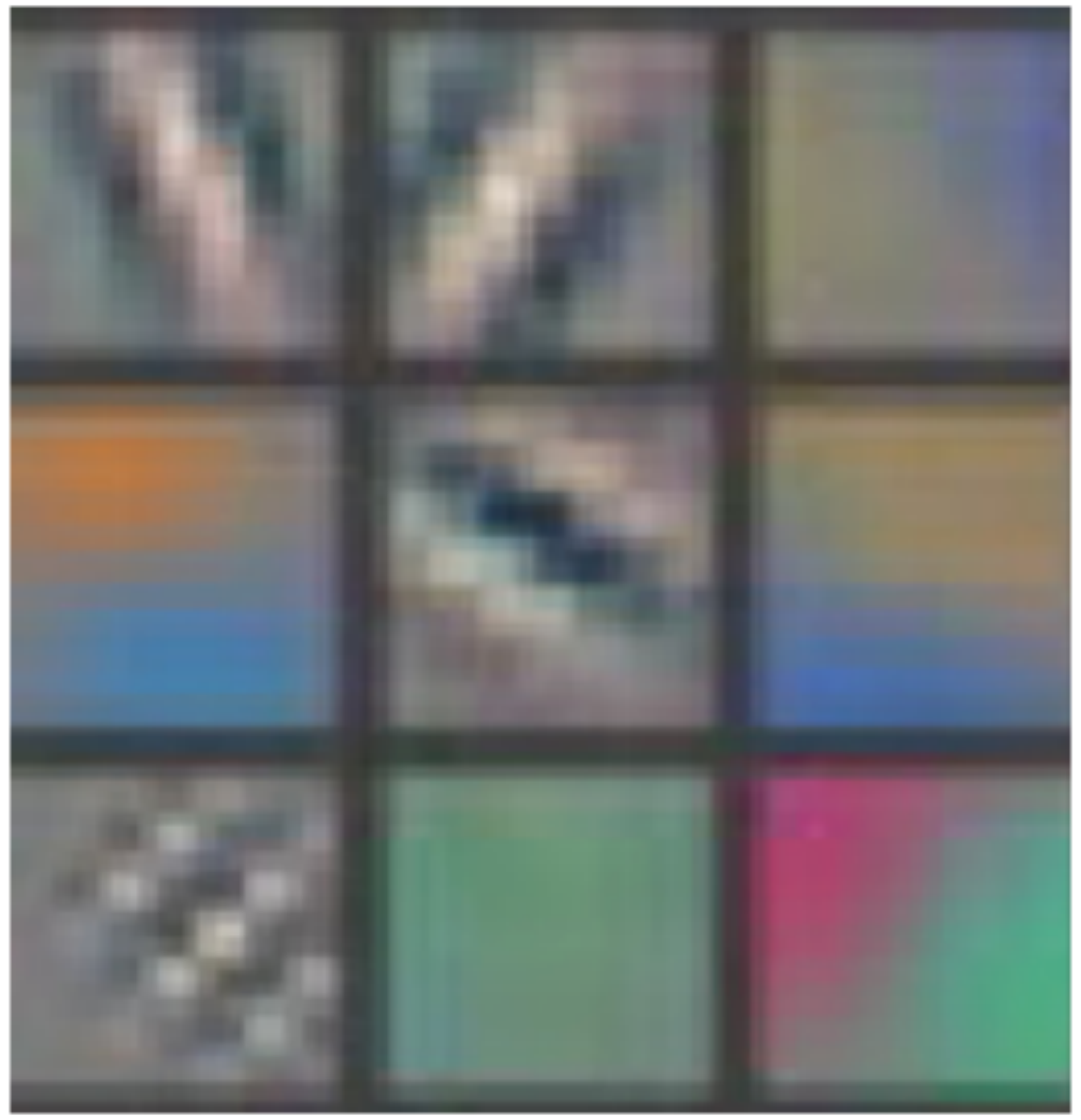
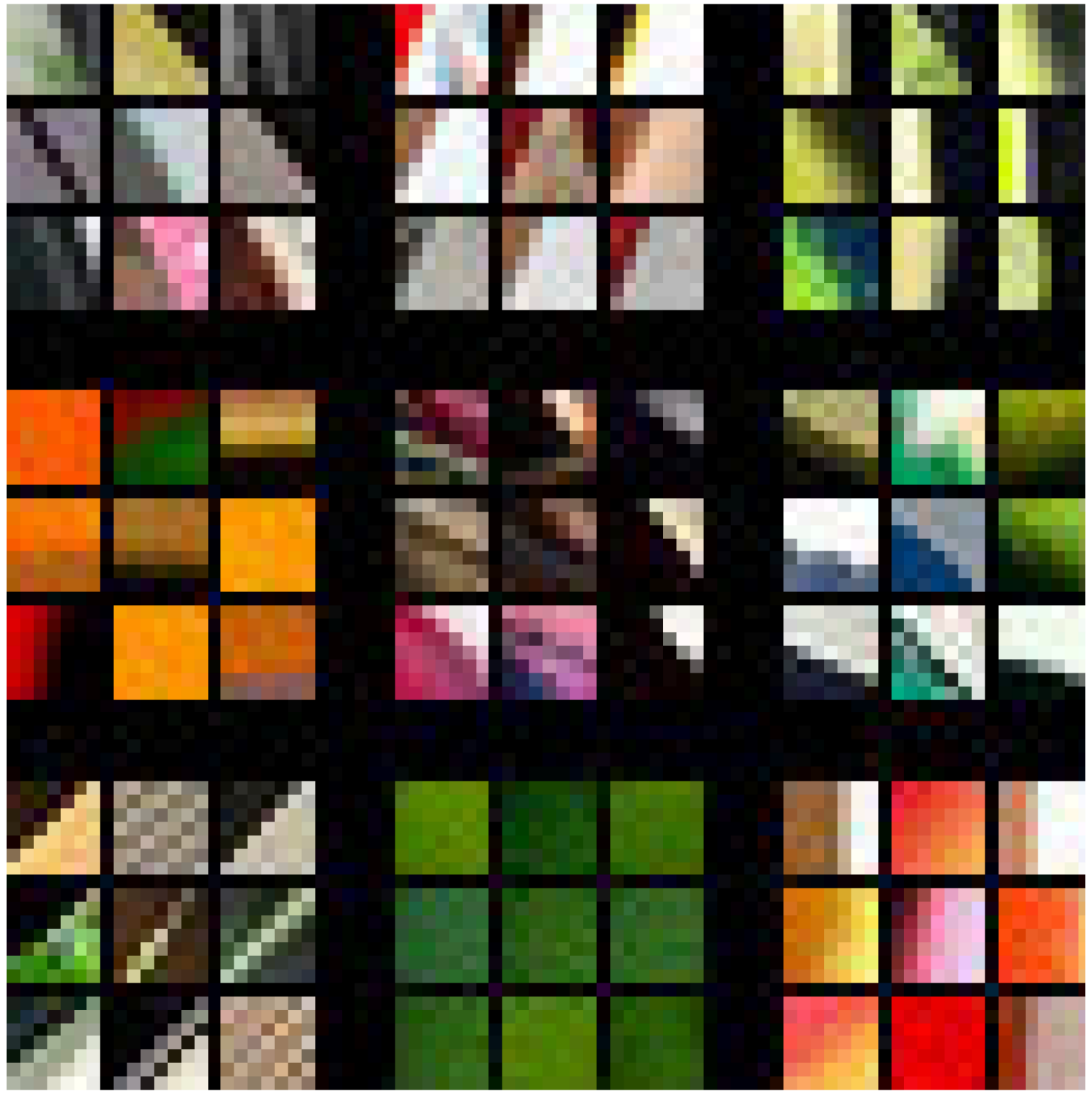
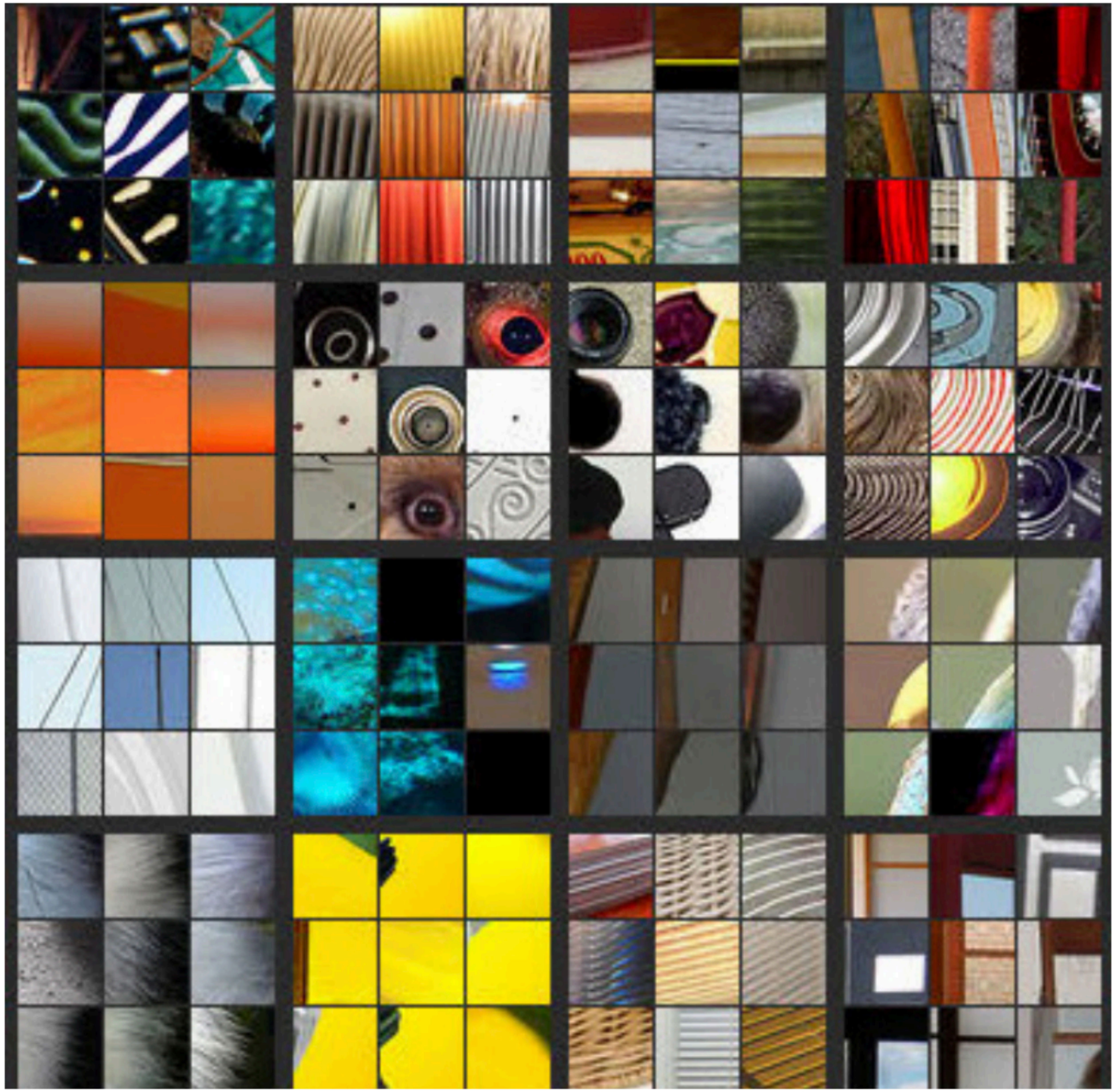


Image patches that activate each of the **layer 1** filters most strongly



[Zeiler and Fergus, 2014]

Image patches that activate several of the **layer 2** neurons most strongly



[Zeiler and Fergus, 2014]

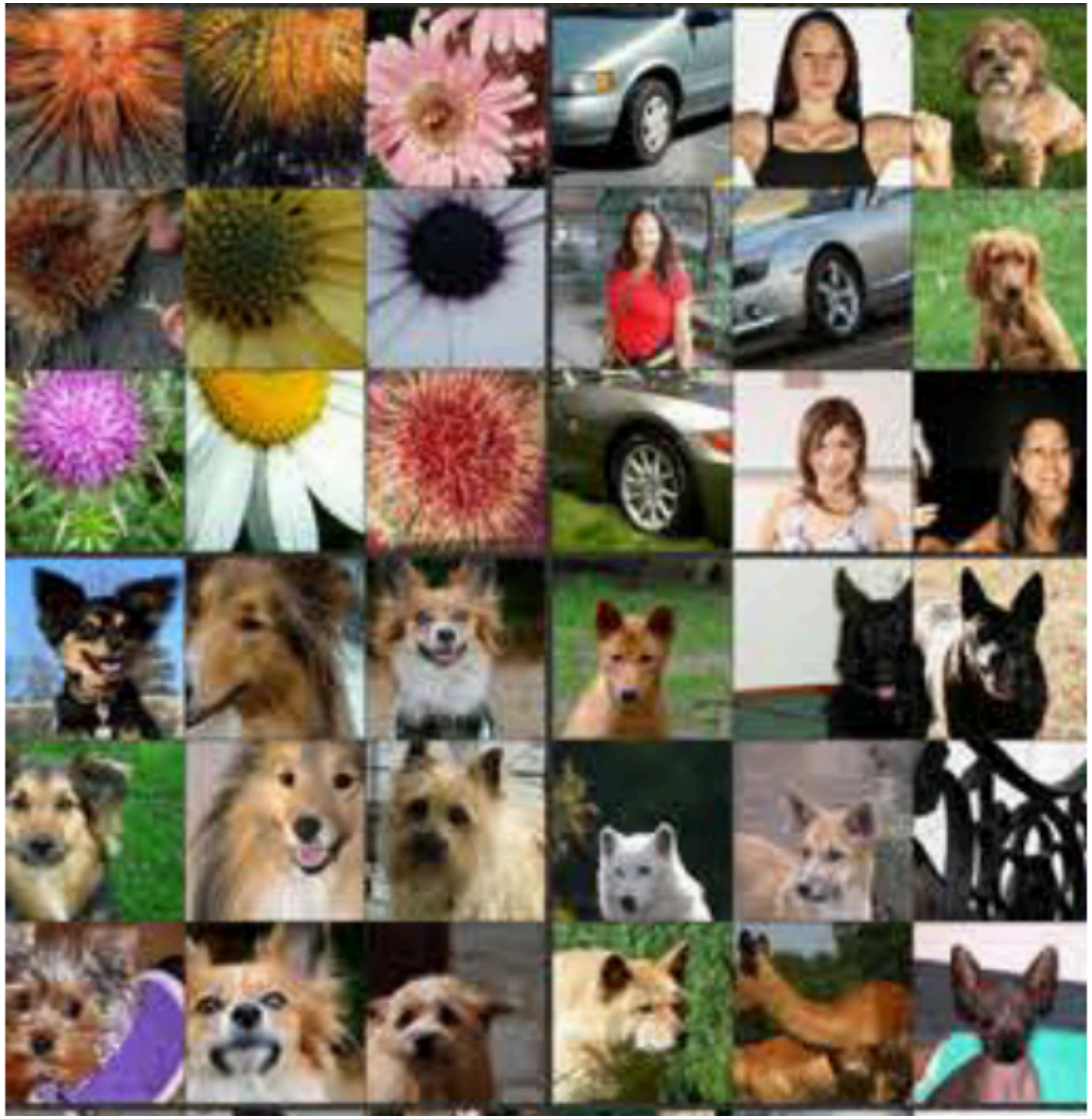
Image patches that activate several of the **layer 3** neurons most strongly

[Zeiler and Fergus, 2014]

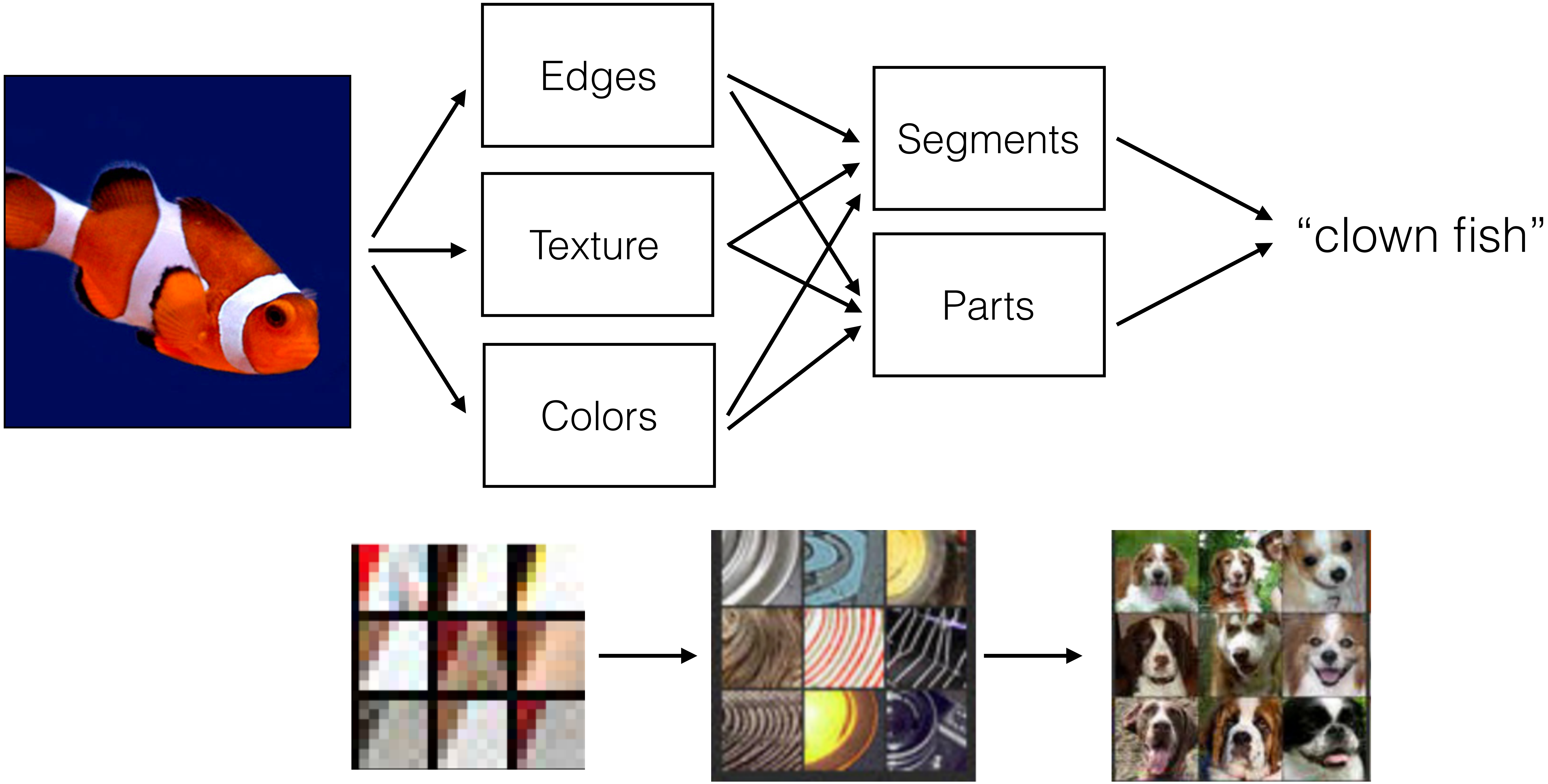
Image patches that activate several of the **layer 4** neurons most strongly

[Zeiler and Fergus, 2014]

Image patches that activate several of the **layer 5** neurons most strongly

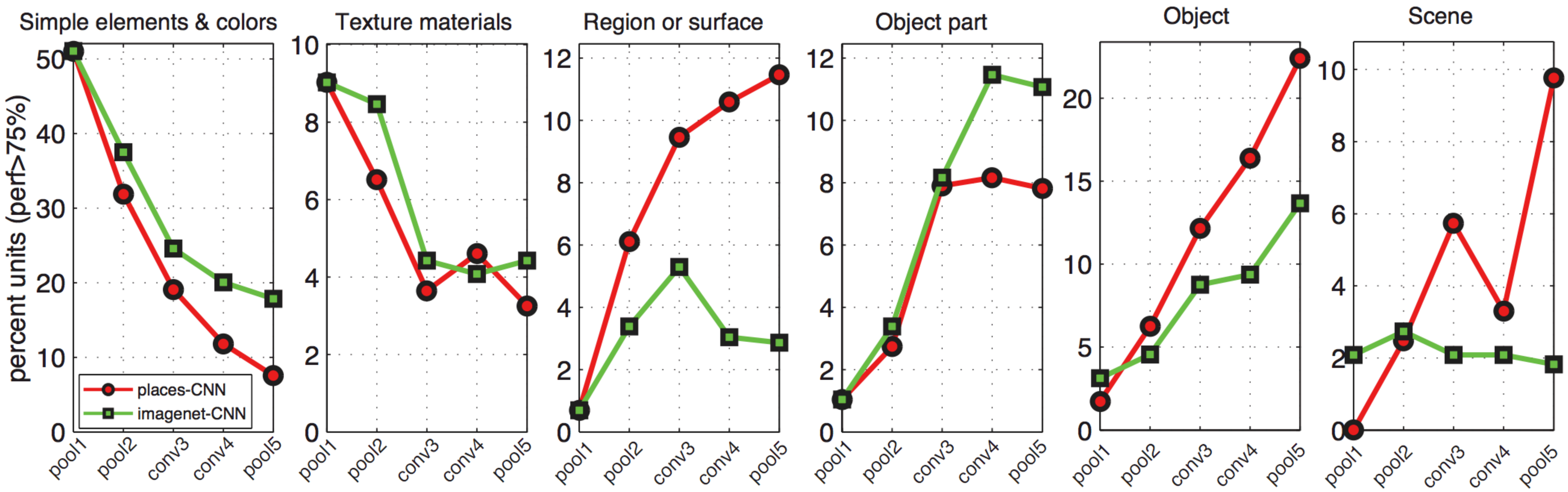


CNNs *learned* the classical visual recognition pipeline!

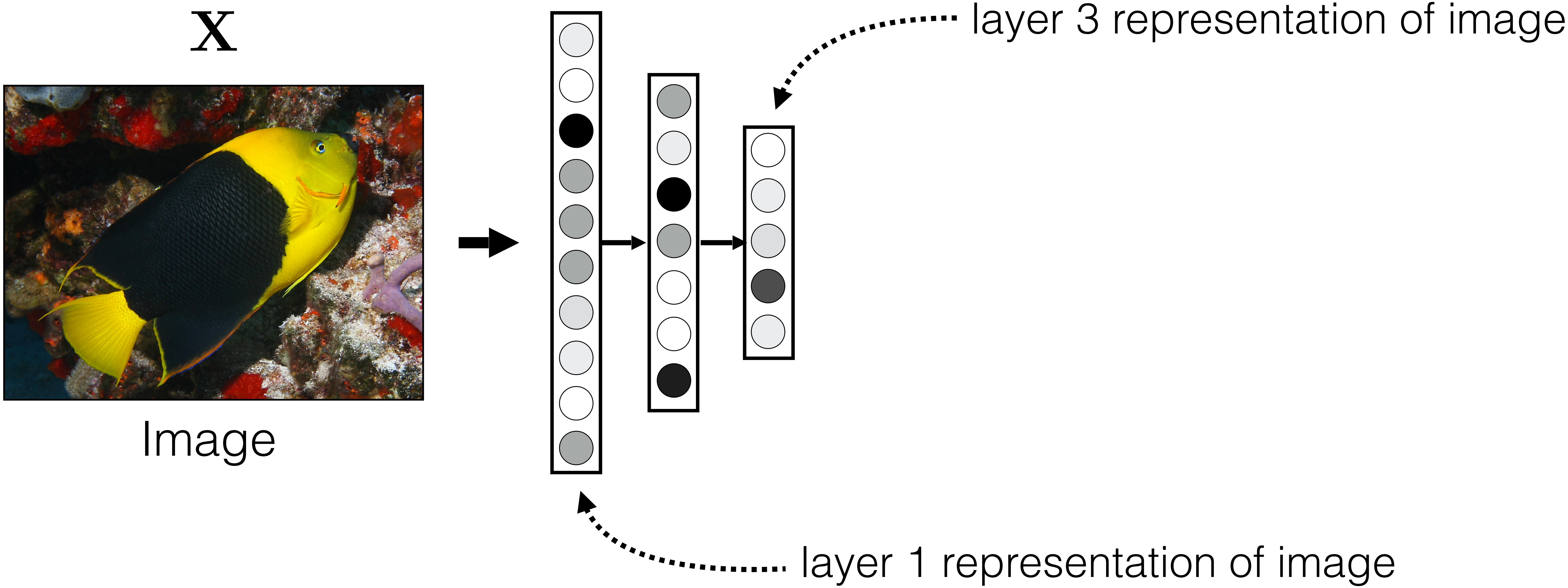


Object Detectors Emergence in Deep Scene CNNs

[Zhou, Khosla, Lapedriza, Oliva, Torralba, ICLR 2015]



im2vec



Represent image as a neural **embedding** — a vector/tensor of neural activations
(perhaps representing a vector of detected texture patterns or object parts)

Investigating a representation via similarity analysis

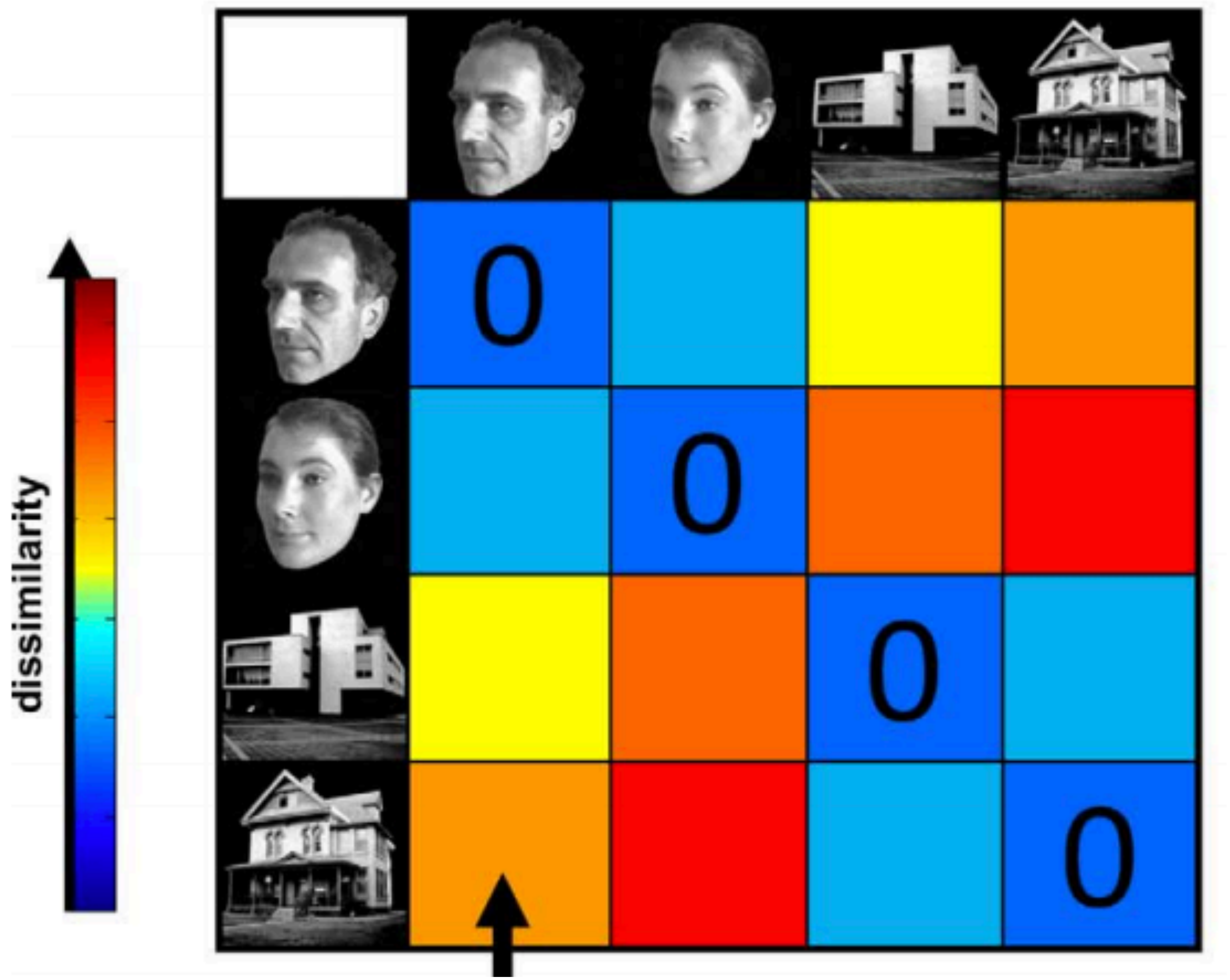
How similar are these two images?

How about these two?

[Kriegeskorte et al. 2008]

Investigating a representation via similarity analysis

Representational Dissimilarity Matrix



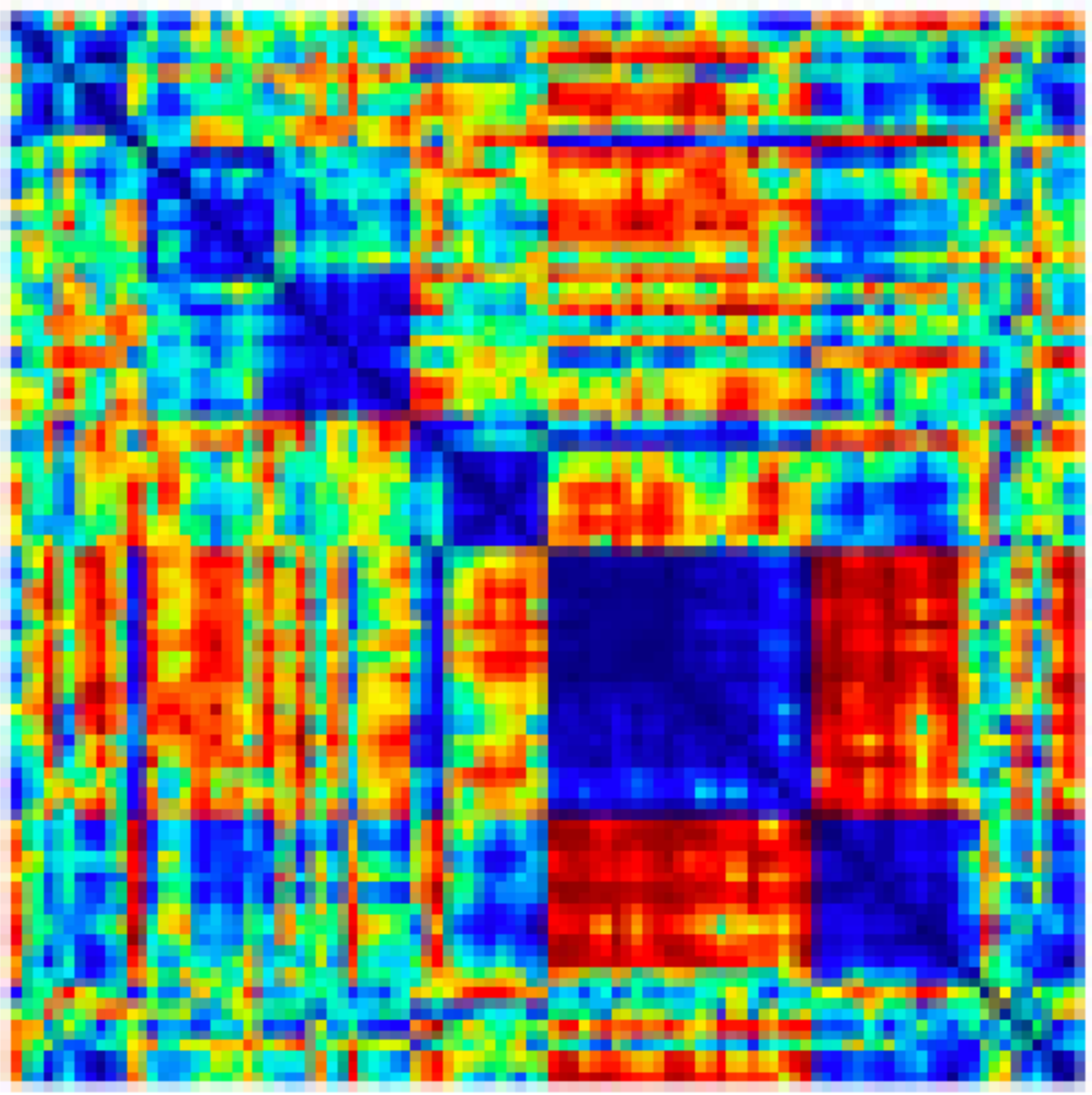
$$\|\mathbf{h}_i - \mathbf{h}_j\|$$

Neural activation vector

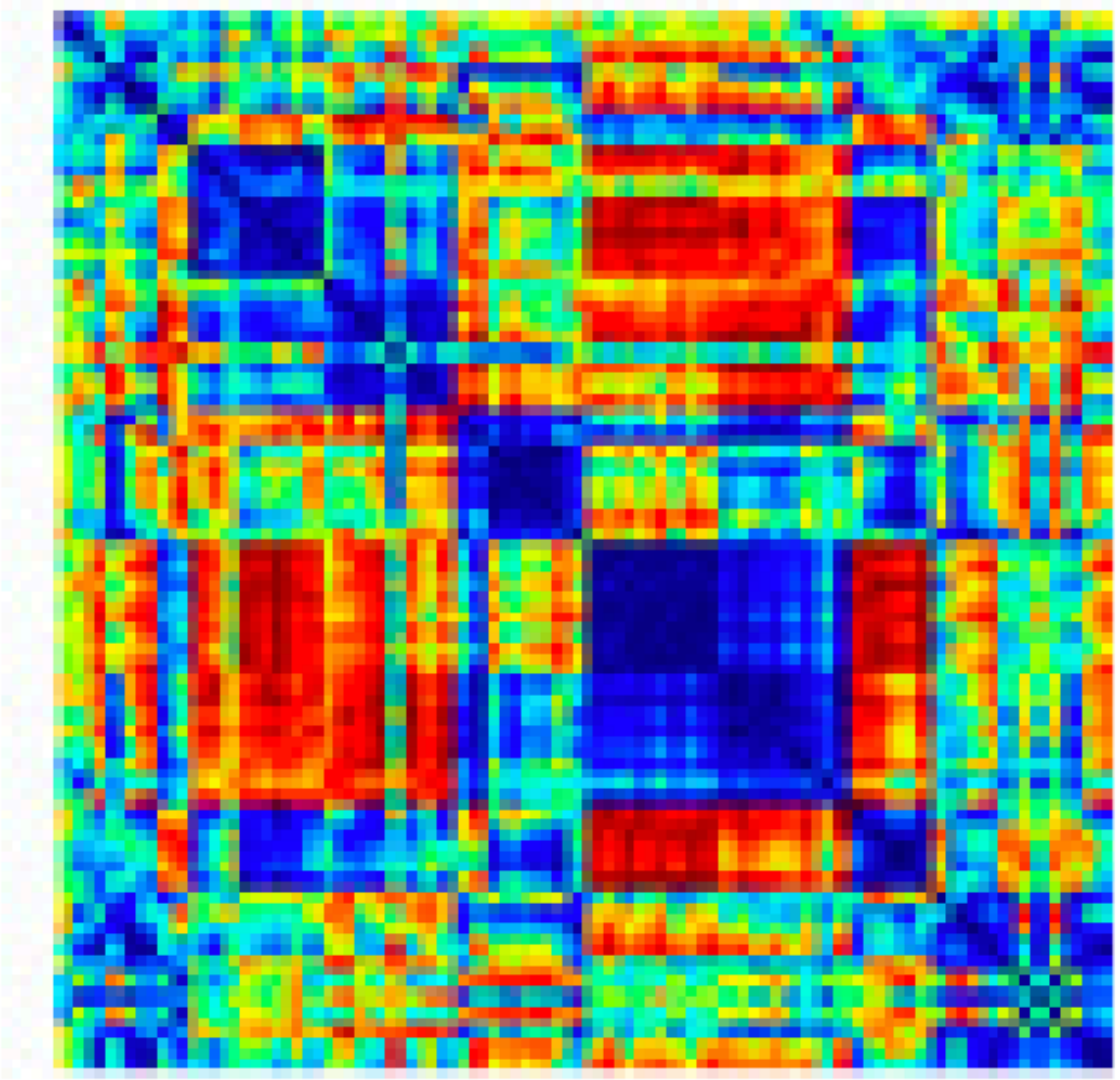
[Kriegeskorte, Mur, Ruff, et al. 2008]

Investigating a representation via similarity analysis

IT Neuronal Units



Deep net (in paricular, HMO)



[Yamins, Hong, Cadieu, Solomon, Seibert, DiCarlo, PNAS 2014]

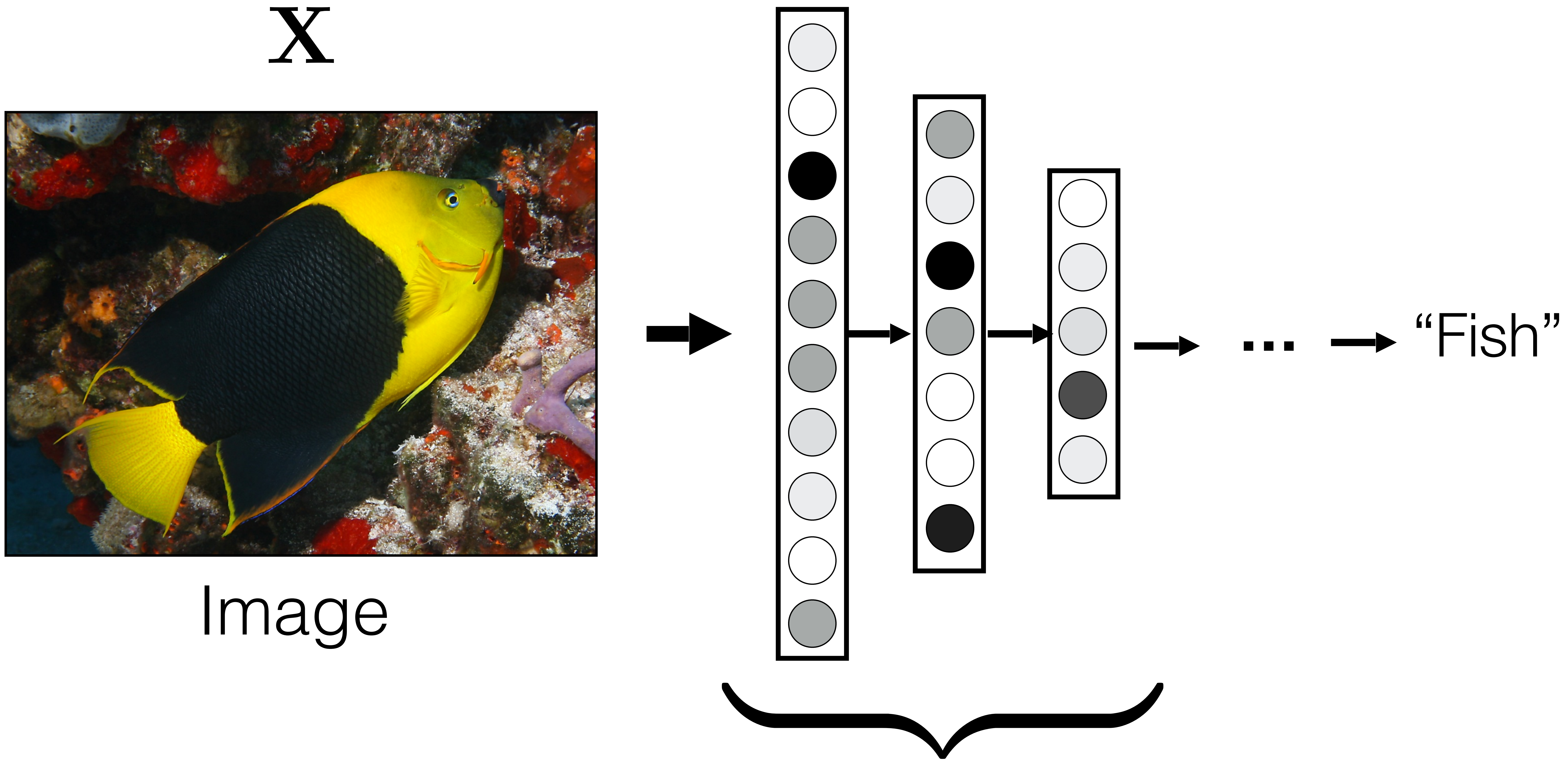
Investigating a representation via similarity analysis

Deep nets and the primate brain both learn similar metric spaces.

Deep nets organize visual information similarly to how our brains do!

[Yamins, Hong, Cadieu, Solomon, Seibert, DiCarlo, PNAS 2014]

What do deep nets internally learn?



Representations!

A CNN is a multiscale, hierarchical representation of data

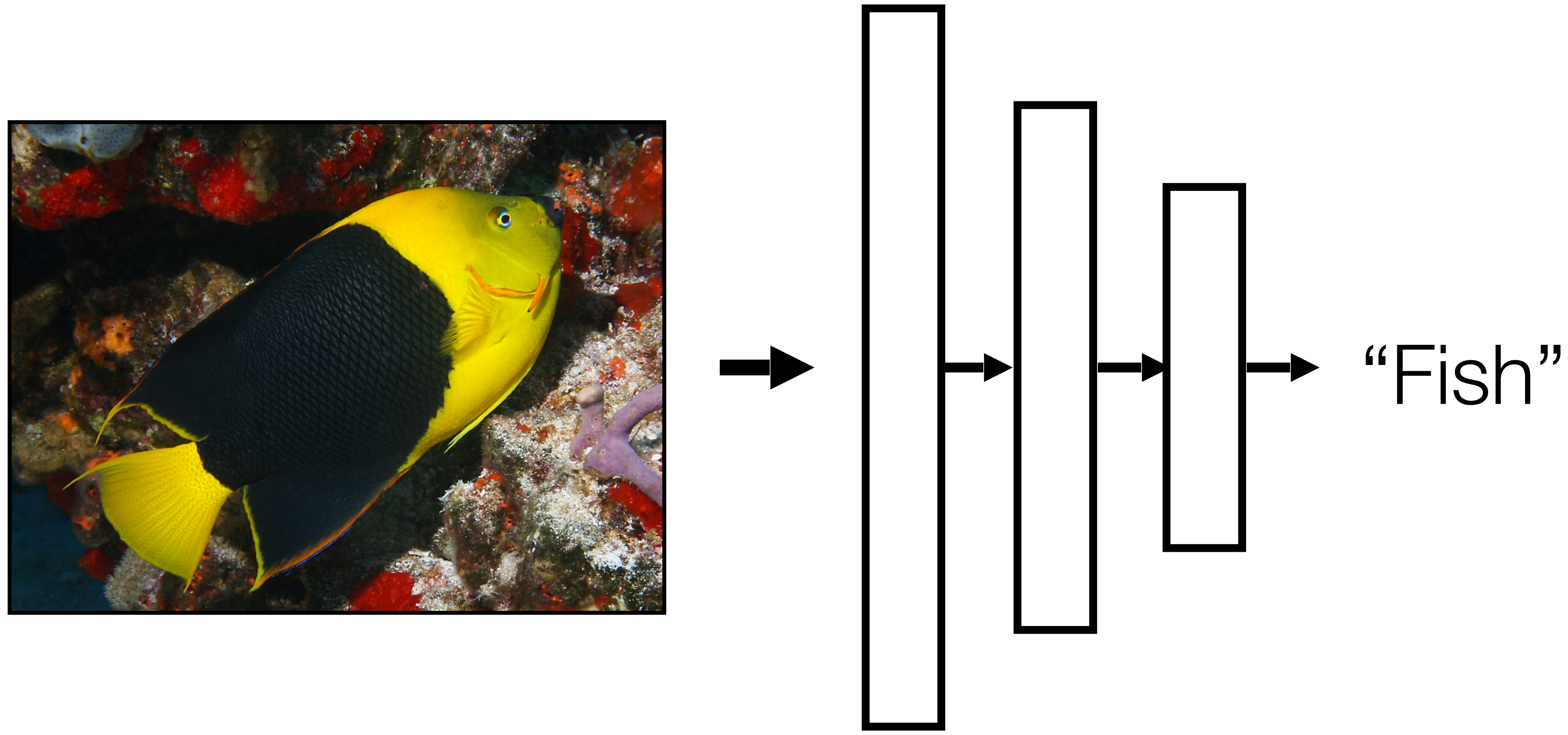
Transfer learning

"Generally speaking, a good representation is one that makes a subsequent learning task easier." — *Deep Learning*, Goodfellow et al. 2016

?

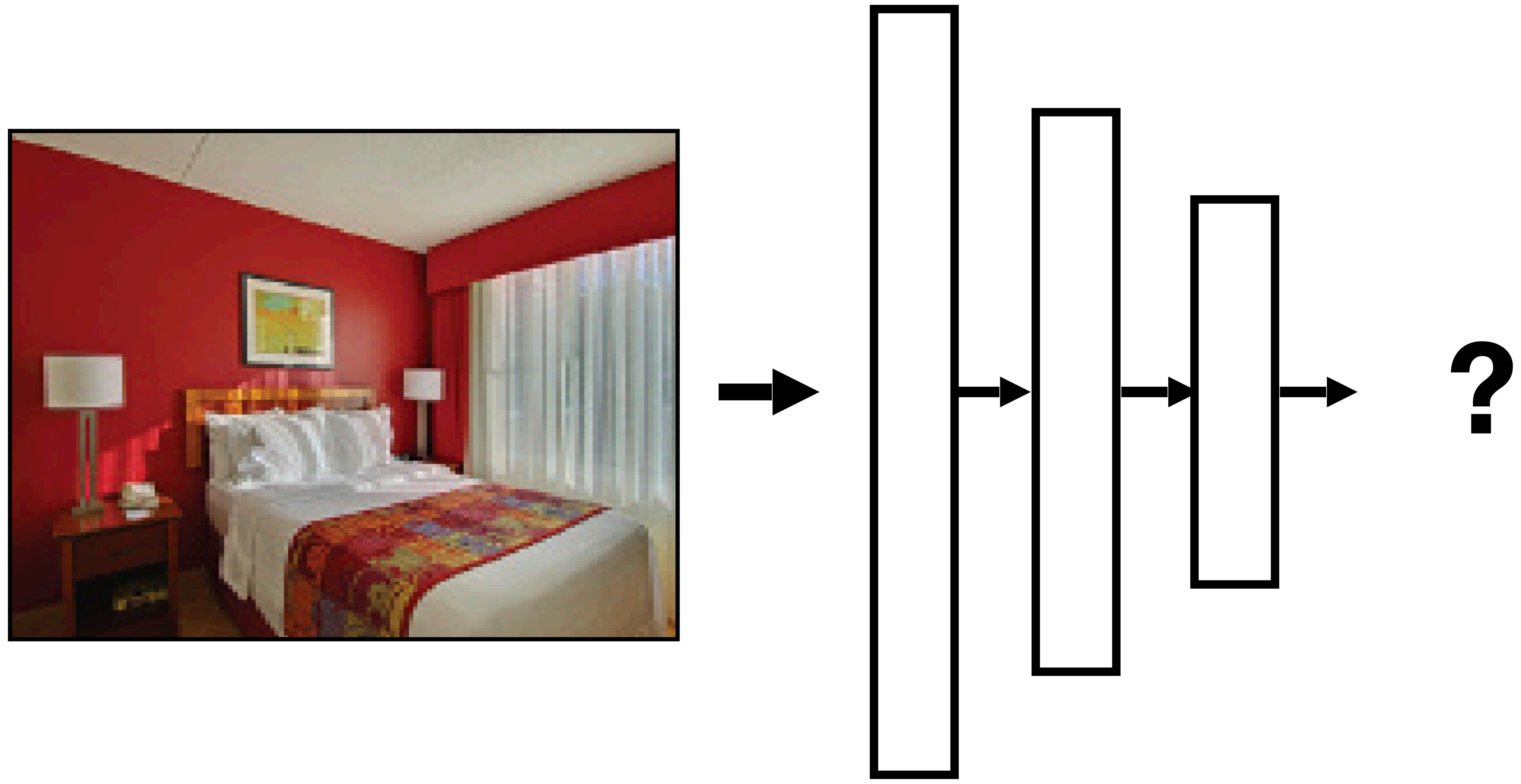
Training

Object recognition

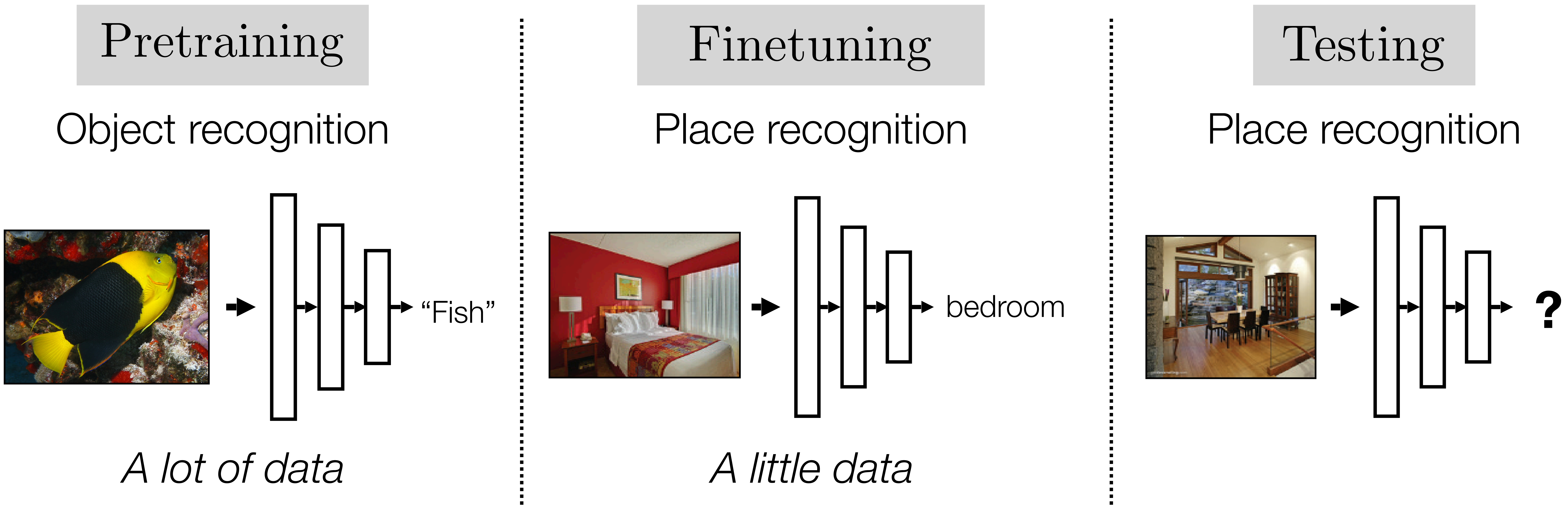


Testing

Place recognition



Often, what we will be "tested" on is to learn to do a new thing.



Finetuning starts with the representation learned on a previous task, and adapts it to perform well on a new task.

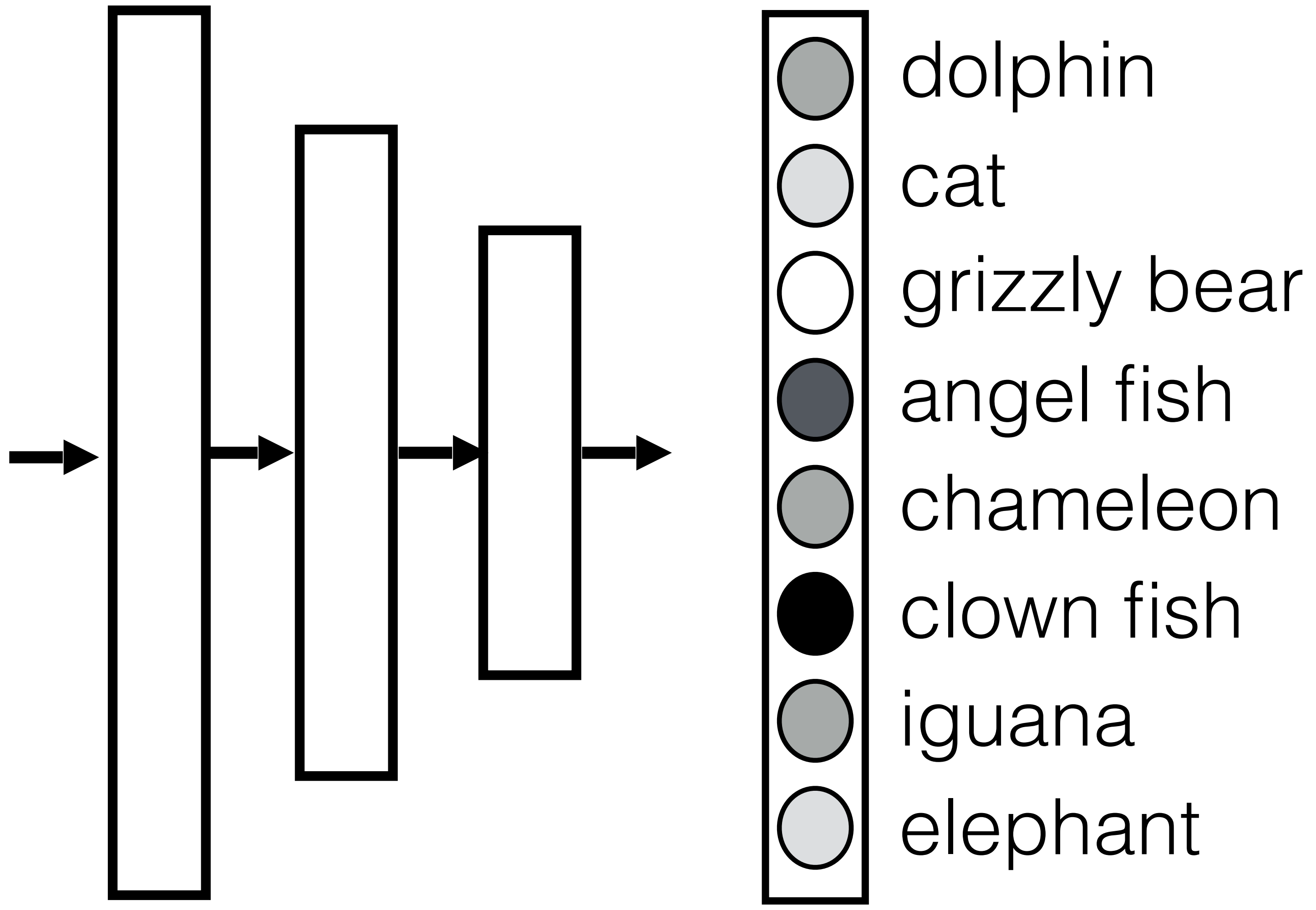
Finetuning in practice

- Pretrain a network on task A (often object recognition), resulting in parameters **W** and **b**
- Initialize a second network with some or all of **W** and **b**
- Train the second network on task B, resulting in parameters **W'** and **b'**

Finetuning in practice

Pretraining

Object recognition



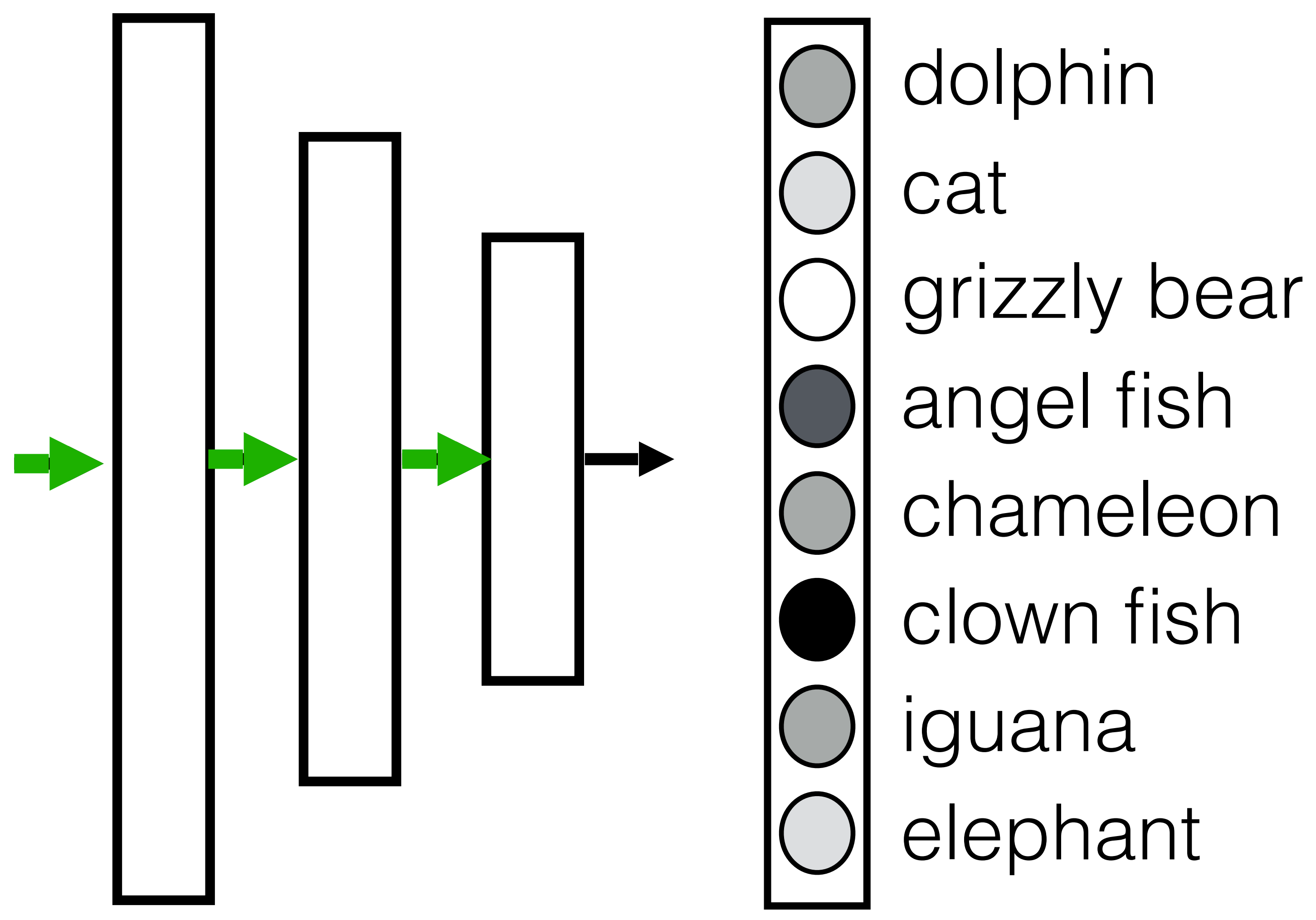
Finetuning

Place recognition

Finetuning in practice

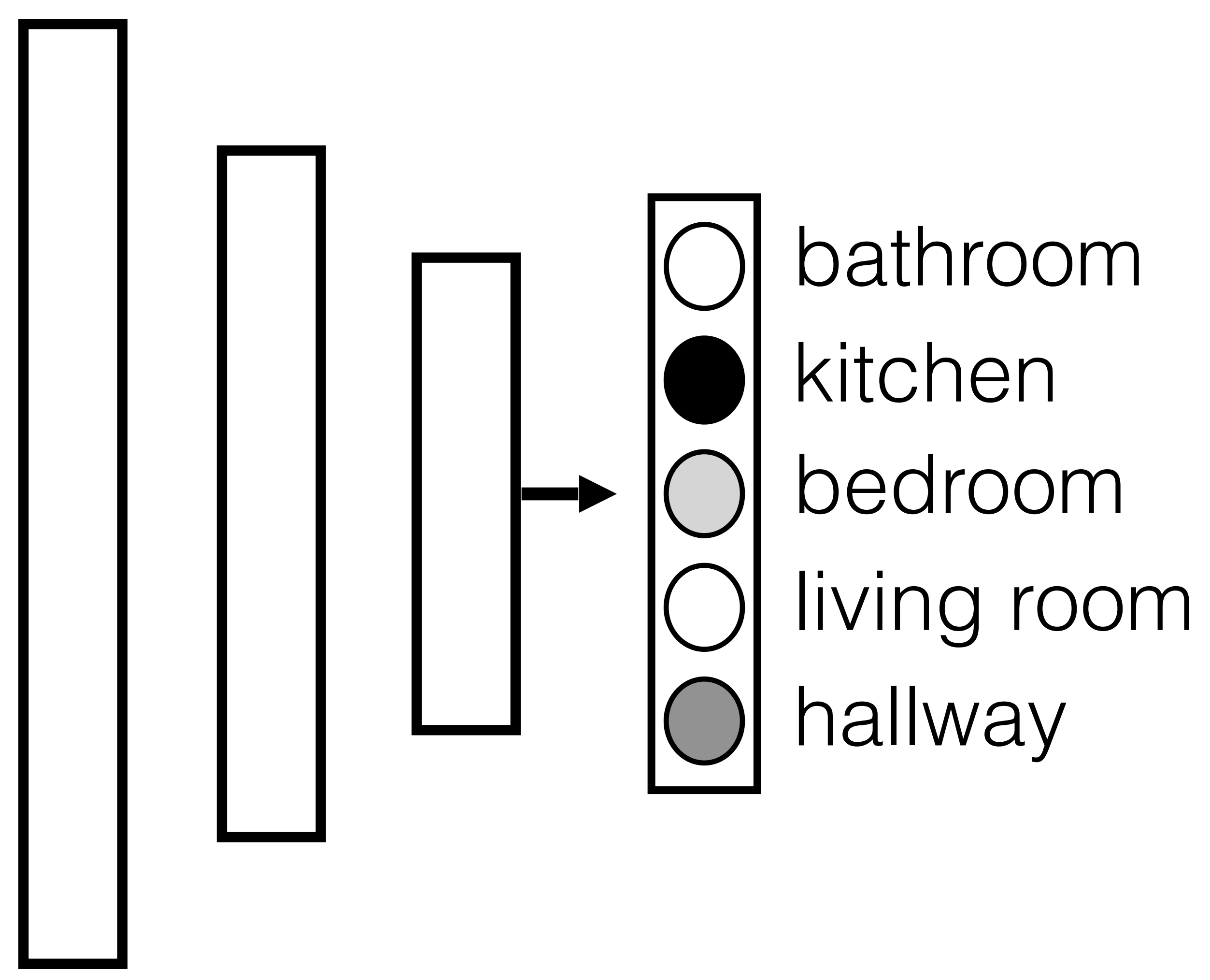
Pretraining

Object recognition



Finetuning

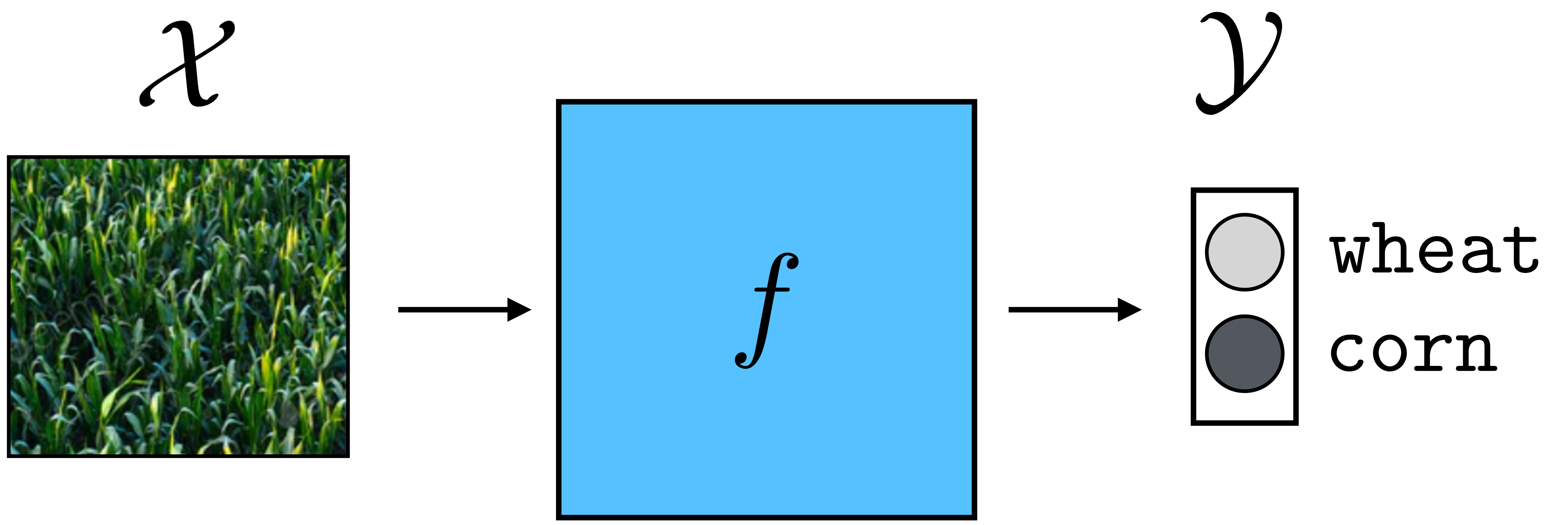
Place recognition



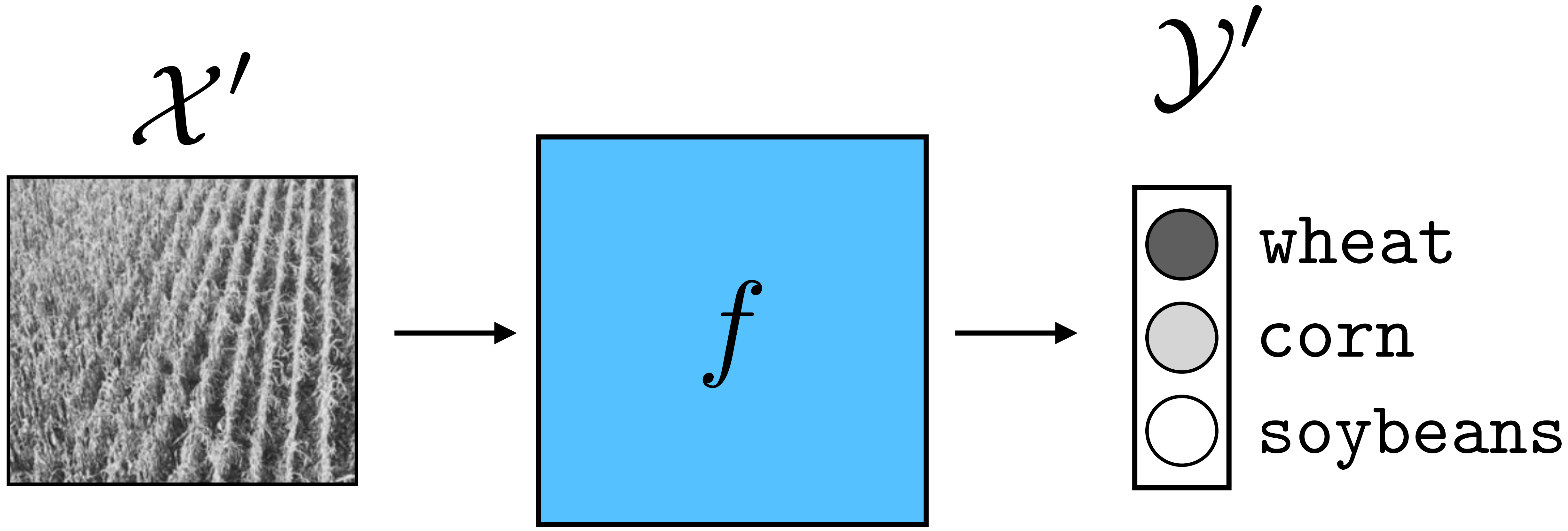
The “learned representation” is just the weights and biases, so that’s what we transfer

What if the input/output dimensions don't match?

Pretraining

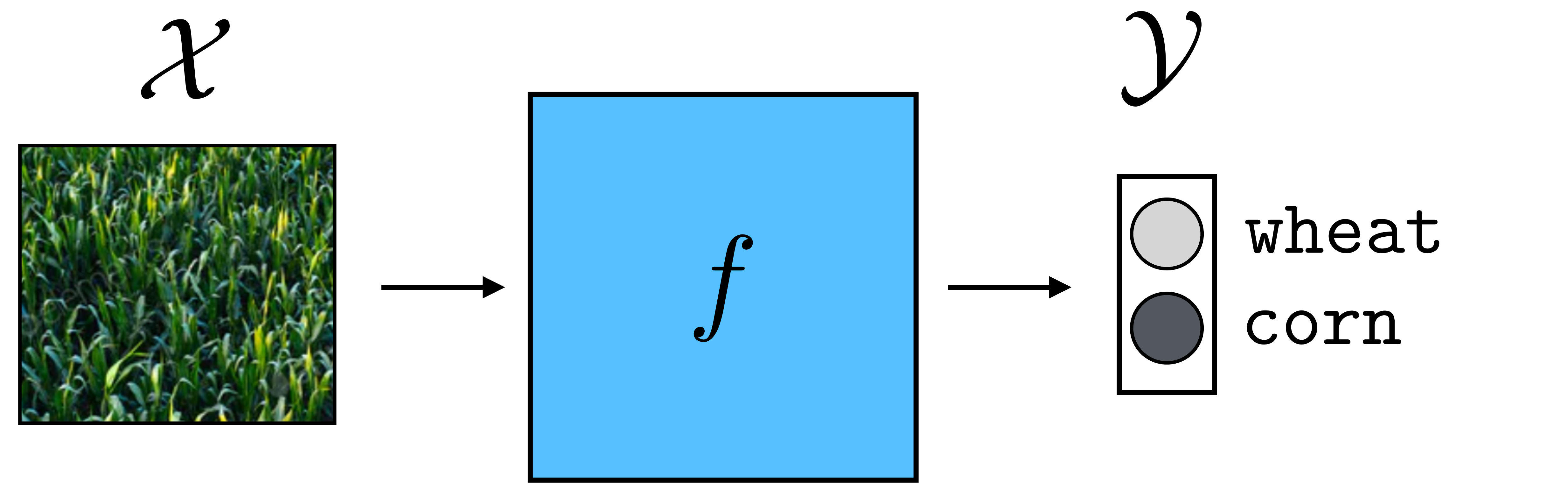


Finetuning

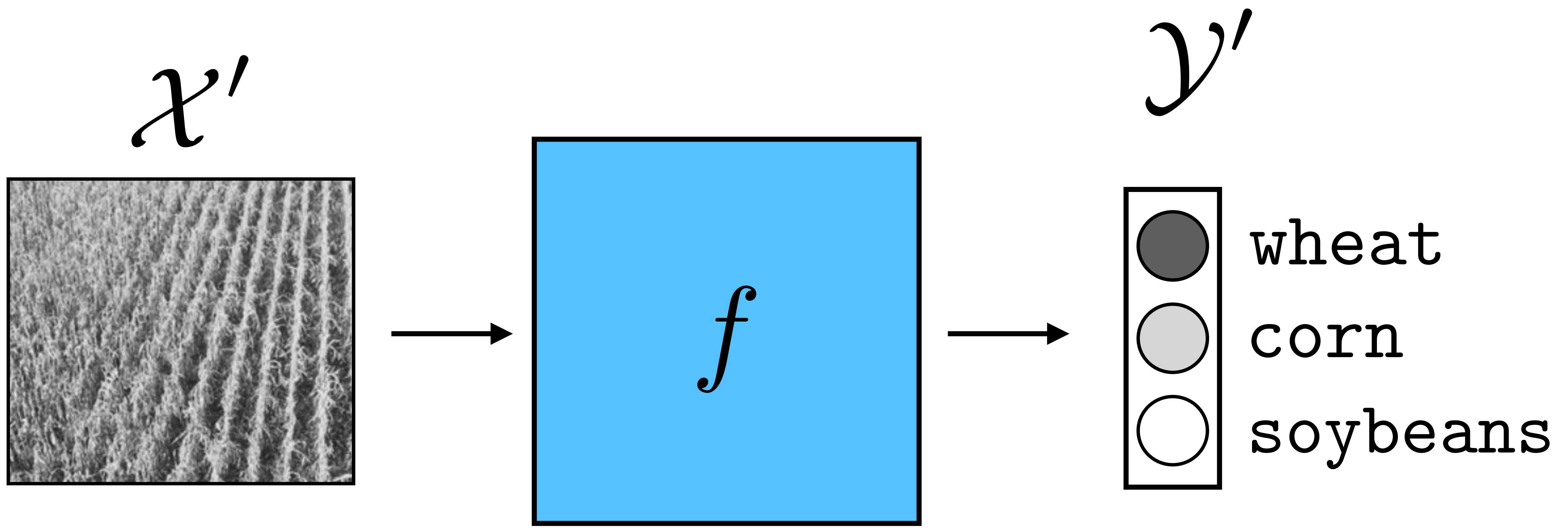


What if the input/output dimensions don't match?

Pretraining

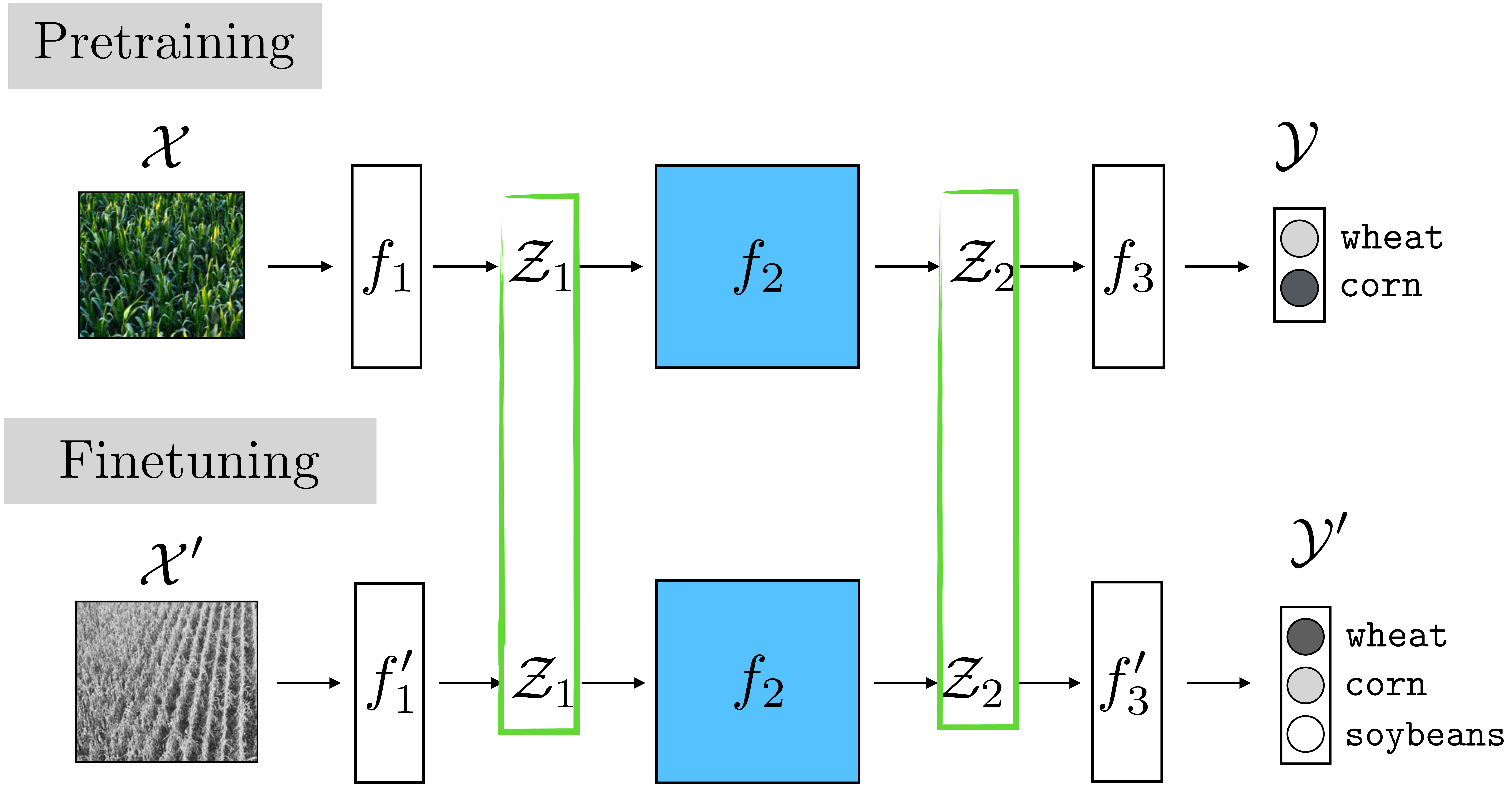


Finetuning

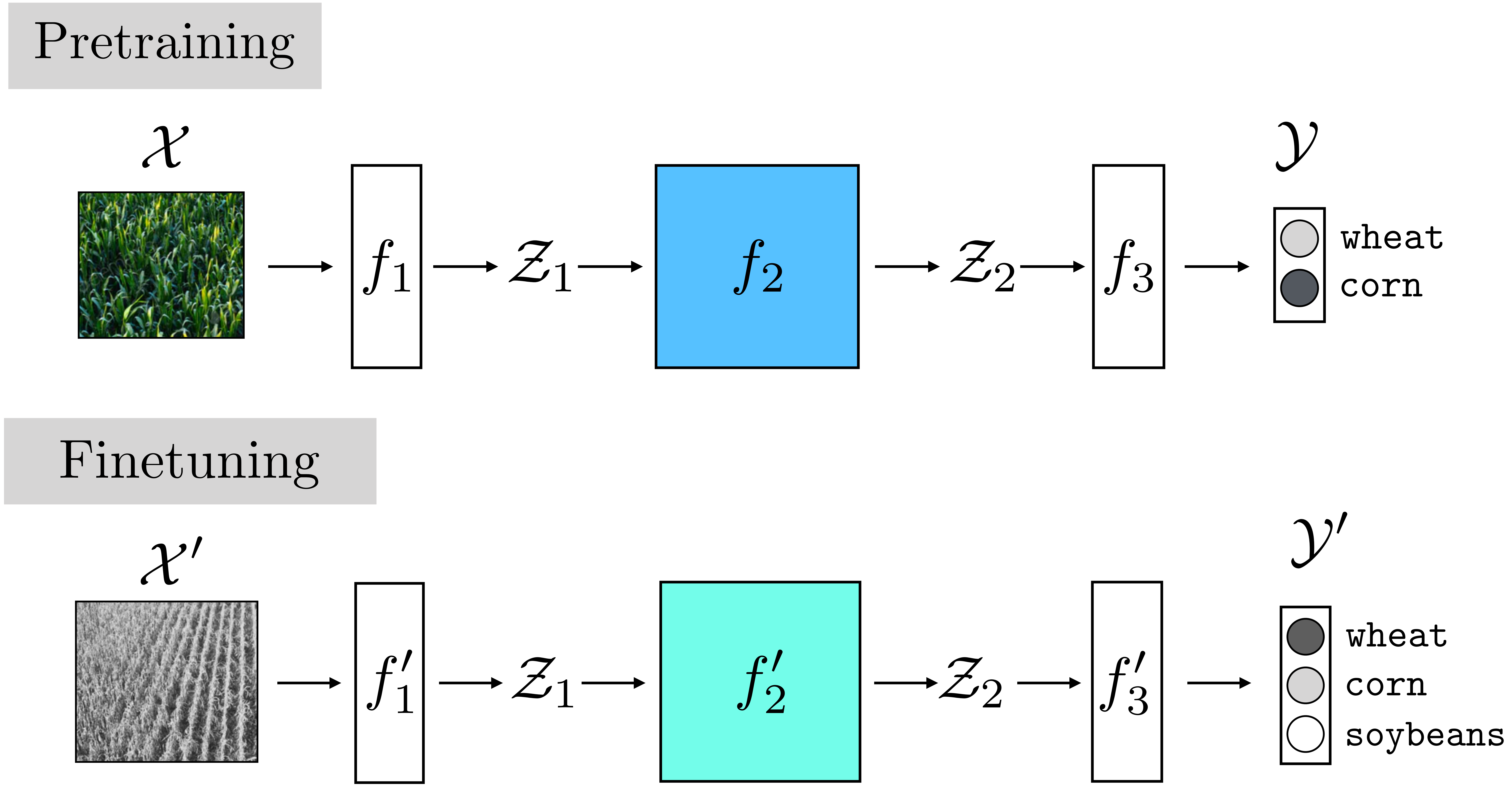


$$\boxed{\begin{array}{c}\mathcal{X}' \neq \mathcal{X} \\ \mathcal{Y}' \neq \mathcal{Y}\end{array}}$$

What if the input/output dimensions don't match?



What if the input/output dimensions don't match?



Supervised object recognition

Learner

"Fish"

image X

label Y

Supervised object recognition

Learner

"Fish"

image X

label Y

Supervised object recognition

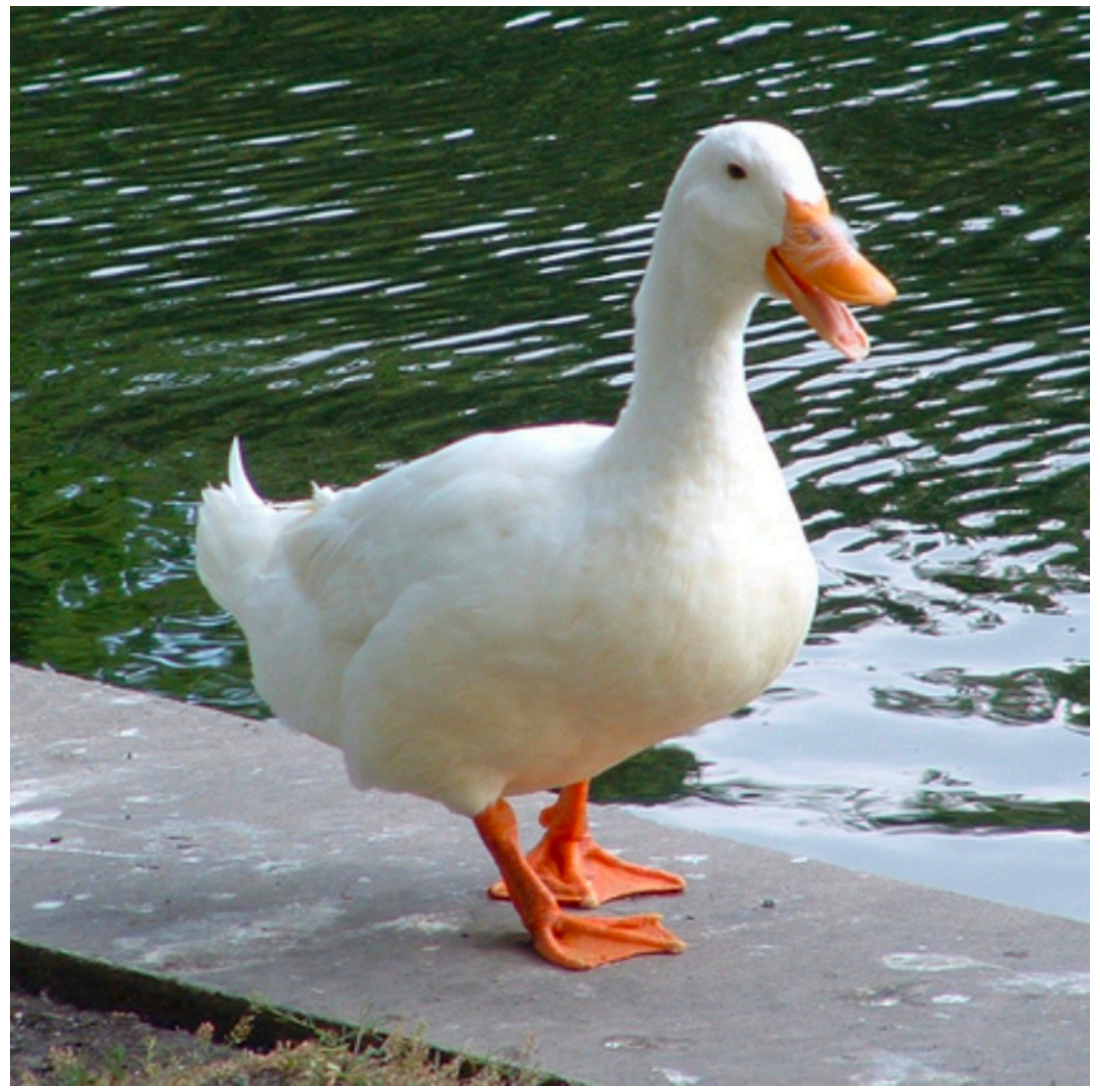
Learner

"Fish"

image X

label Y

Supervised object recognition



⋮

image X

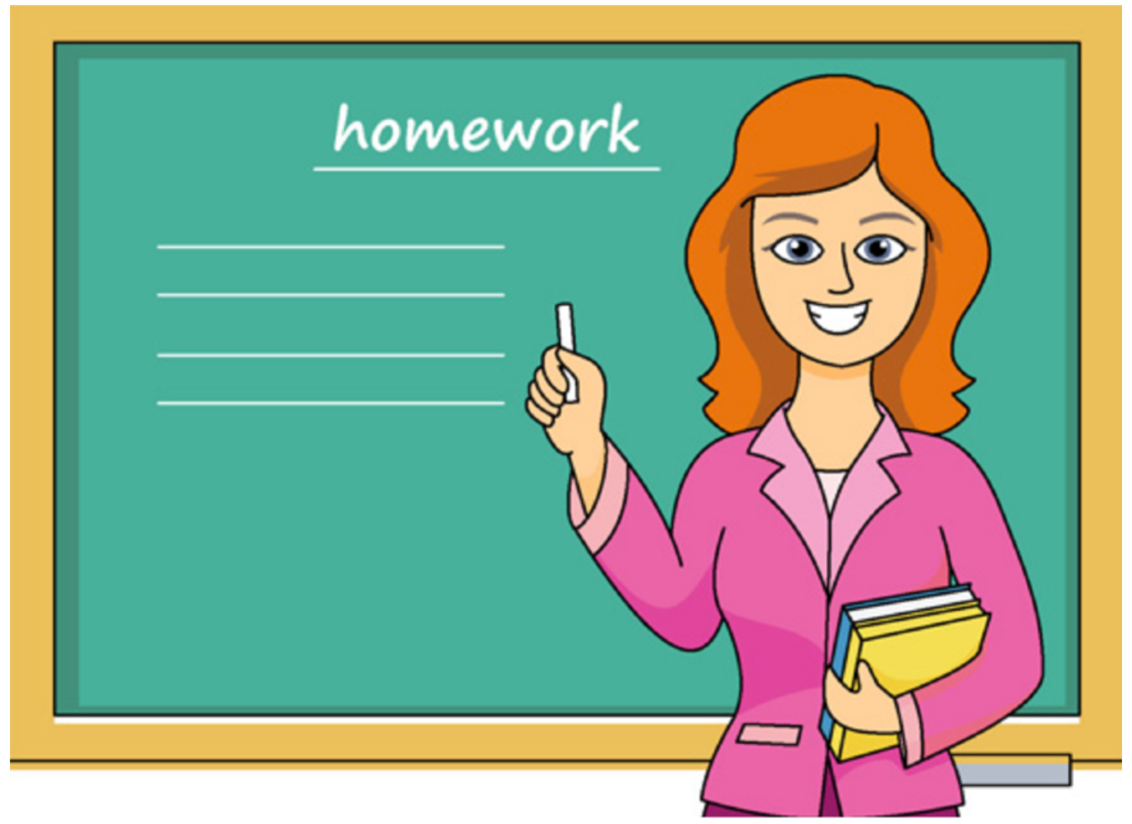
Learner

"Duck"

label Y

Supervised computer vision

Hand-curated training data
+ Informative
- Expensive
- Limited to teacher's knowledge



Vision in nature

Raw unlabeled training data
+ Cheap
- Noisy
- Harder to interpret

Learning from examples

(aka **supervised learning**)

Training data

$$\{x^{(1)}, y^{(1)}\}$$
$$\{x^{(2)}, y^{(2)}\} \quad \rightarrow \quad \boxed{\text{Learner}} \quad \rightarrow \quad f : X \rightarrow Y$$
$$\{x^{(3)}, y^{(3)}\}$$
$$\ldots$$

$$f^* = \arg\min_{f \in \mathcal{F}} \sum_{i=1}^{N} \mathcal{L}(f(\mathbf{x}^{(i)}), \mathbf{y}^{(i)})$$

Learning without examples

(includes **unsupervised learning** and **reinforcement learning**)

Data

$\{x^{(1)}\}$

$\{x^{(2)}\}$ $\rightarrow$ Learner $\rightarrow$?

$\{x^{(3)}\}$

$\dots$

Representation Learning

Data

$\{x^{(1)}\}$

$\{x^{(2)}\}$ $\rightarrow$ Learner $\rightarrow$ Representations

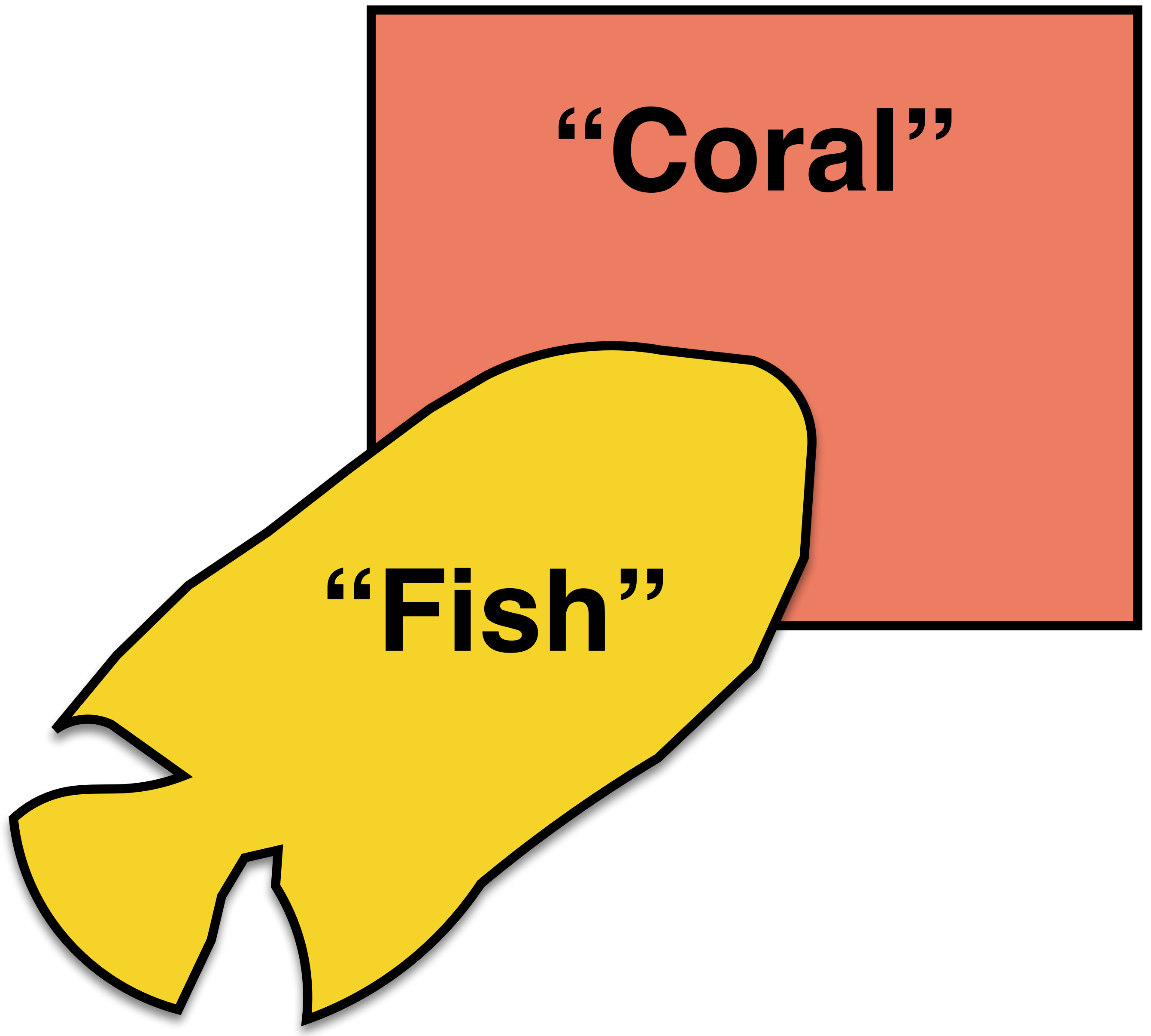
$\{x^{(3)}\}$

$\ldots$

Unsupervised Representation Learning

$\mathbf{x}$

Image

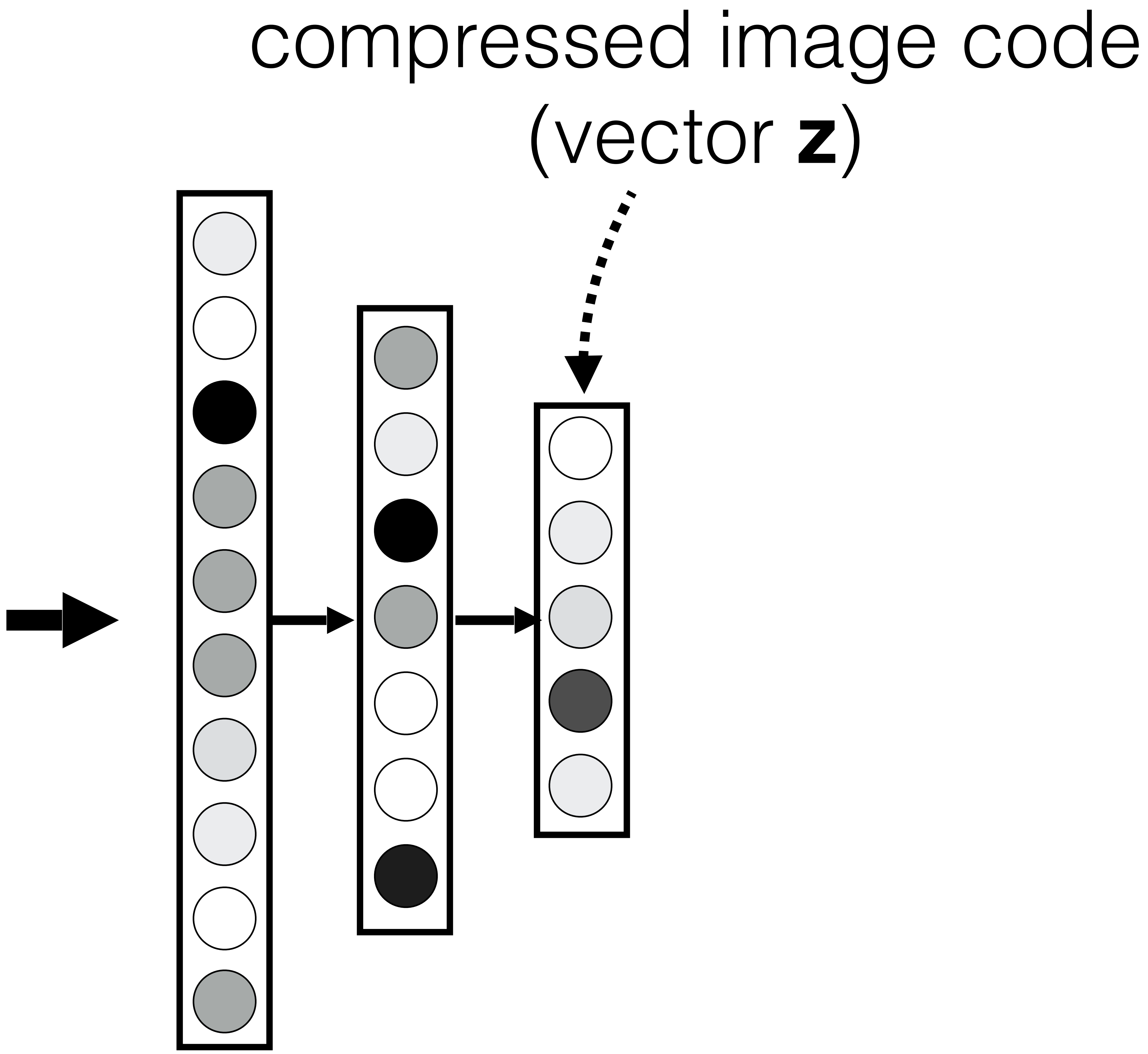


Compact mental representation

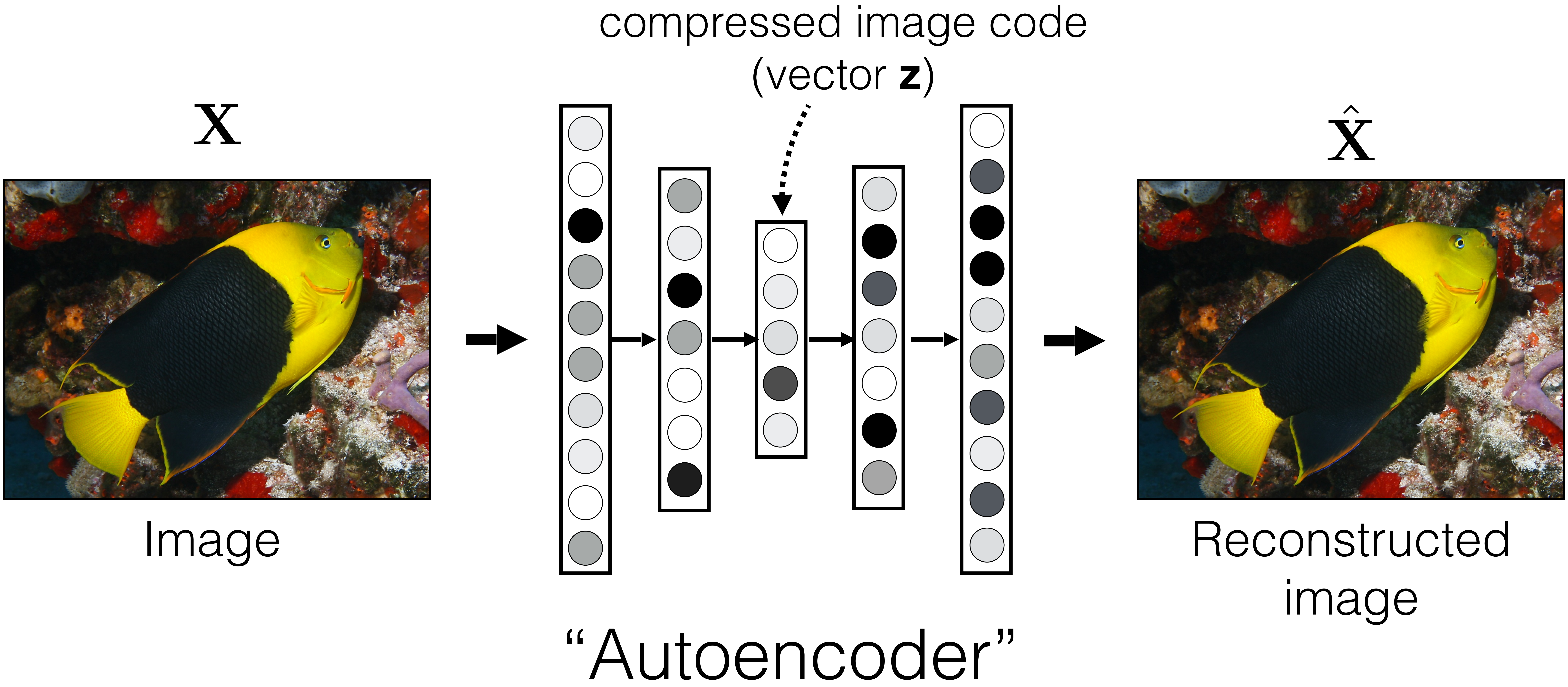
Unsupervised Representation Learning

$\mathbf{x}$

Image

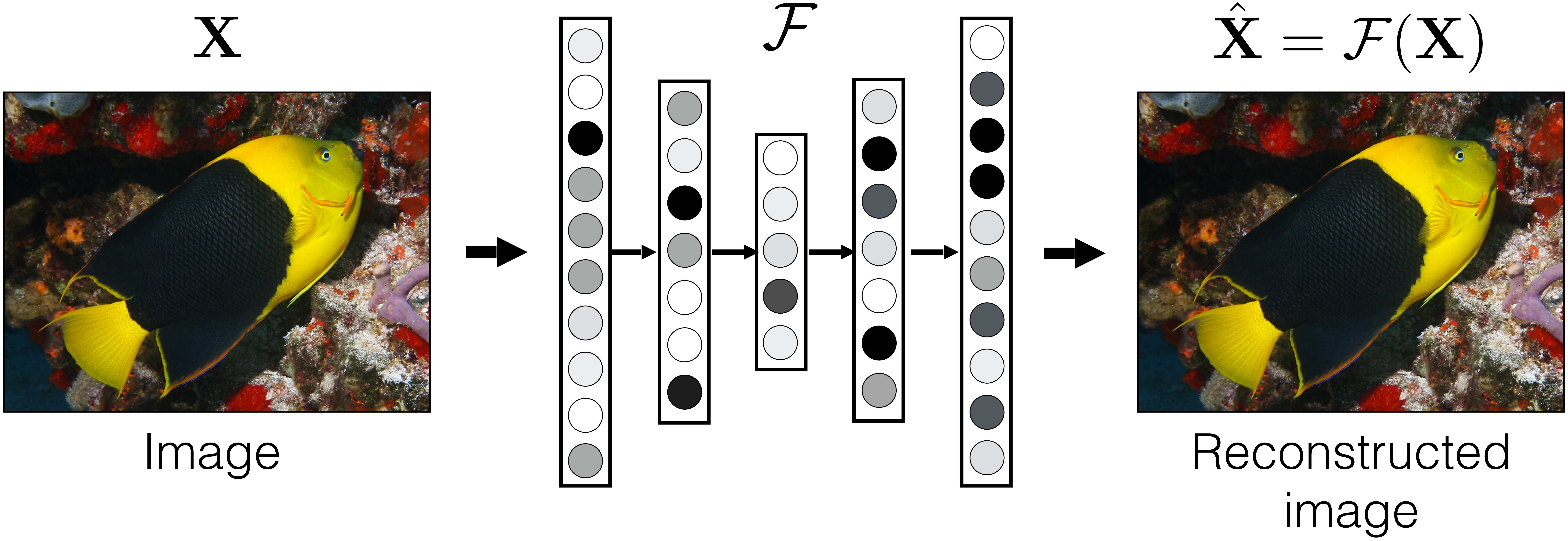


Unsupervised Representation Learning

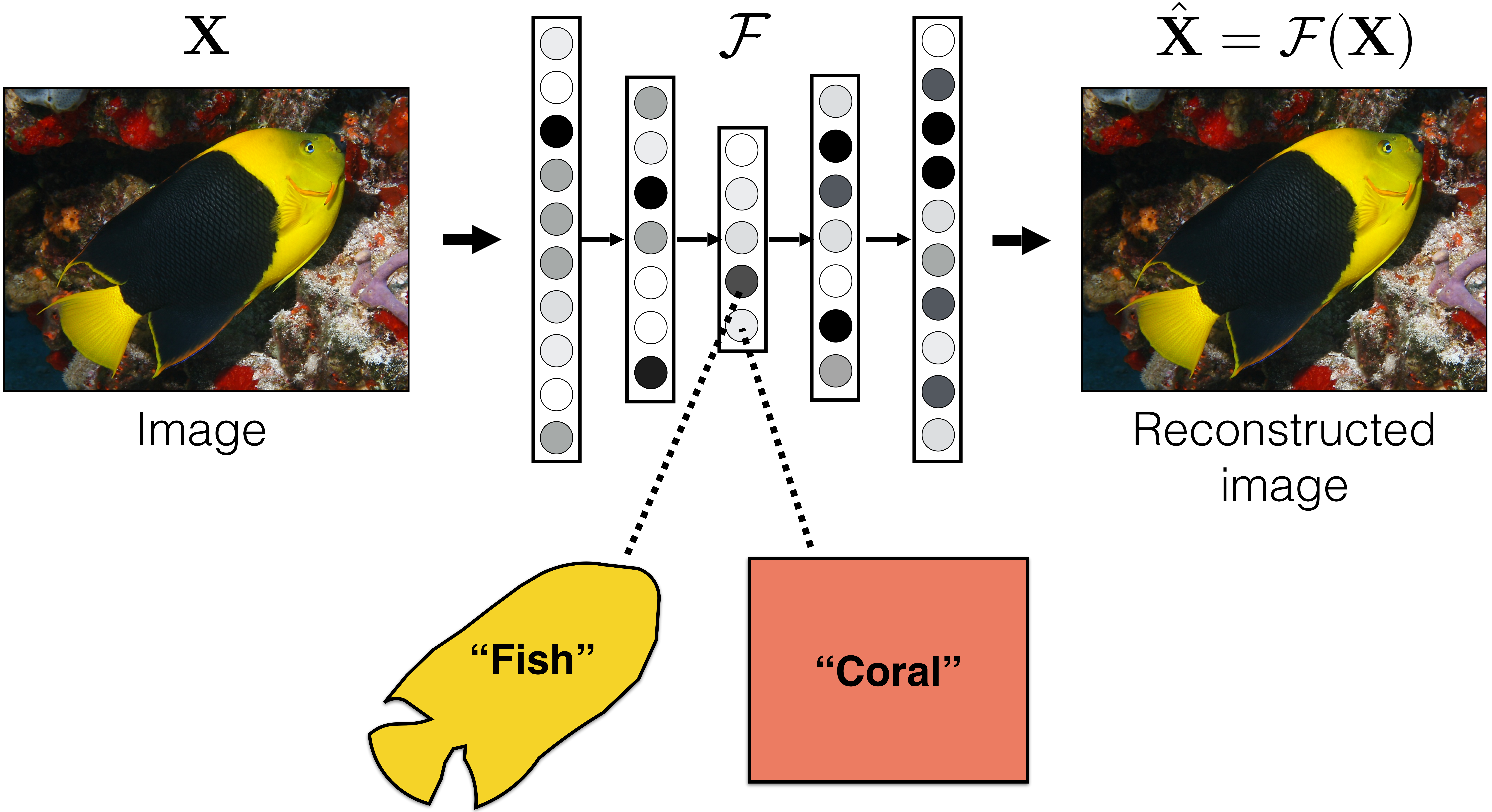


[e.g., Hinton & Salakhutdinov, Science 2006]

Autoencoder



$$\arg\min_{\mathcal{F}} \mathbb{E}_{\mathbf{X}}[||\mathcal{F}(\mathbf{X}) - \mathbf{X}||]$$



$\mathbf{X}$
$\mathcal{F}$
$\hat{\mathbf{X}} = \mathcal{F}(\mathbf{X})$
Image
Reconstructed image
"Fish"
"Coral"

Autoencoder

Data

$\{\mathbf{x}^{(i)}\}_{i=1}^{N} \quad \rightarrow$

Learner

Objective

$\mathcal{L}(f(\mathbf{x}), \mathbf{x}) = \|f(\mathbf{x}) - \mathbf{x}\|_2^2$

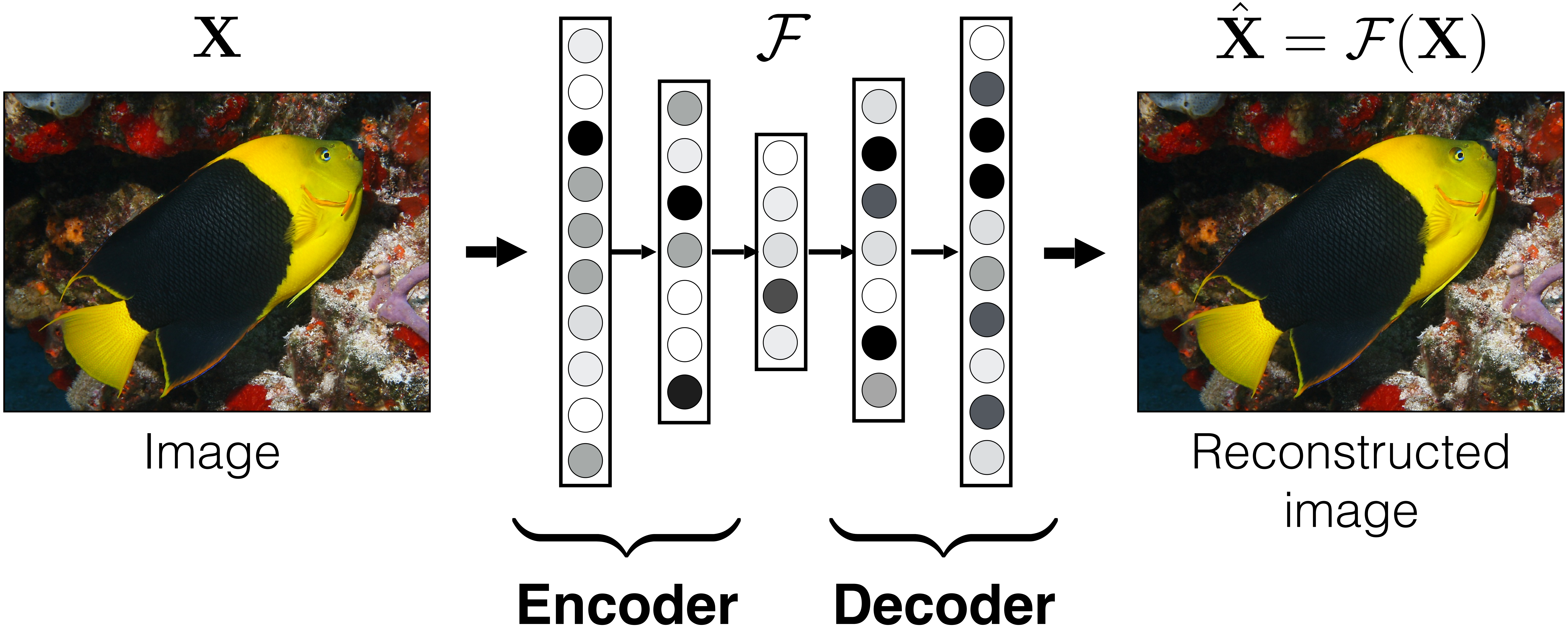
Hypothesis space

Neural net with a bottleneck

Optimizer

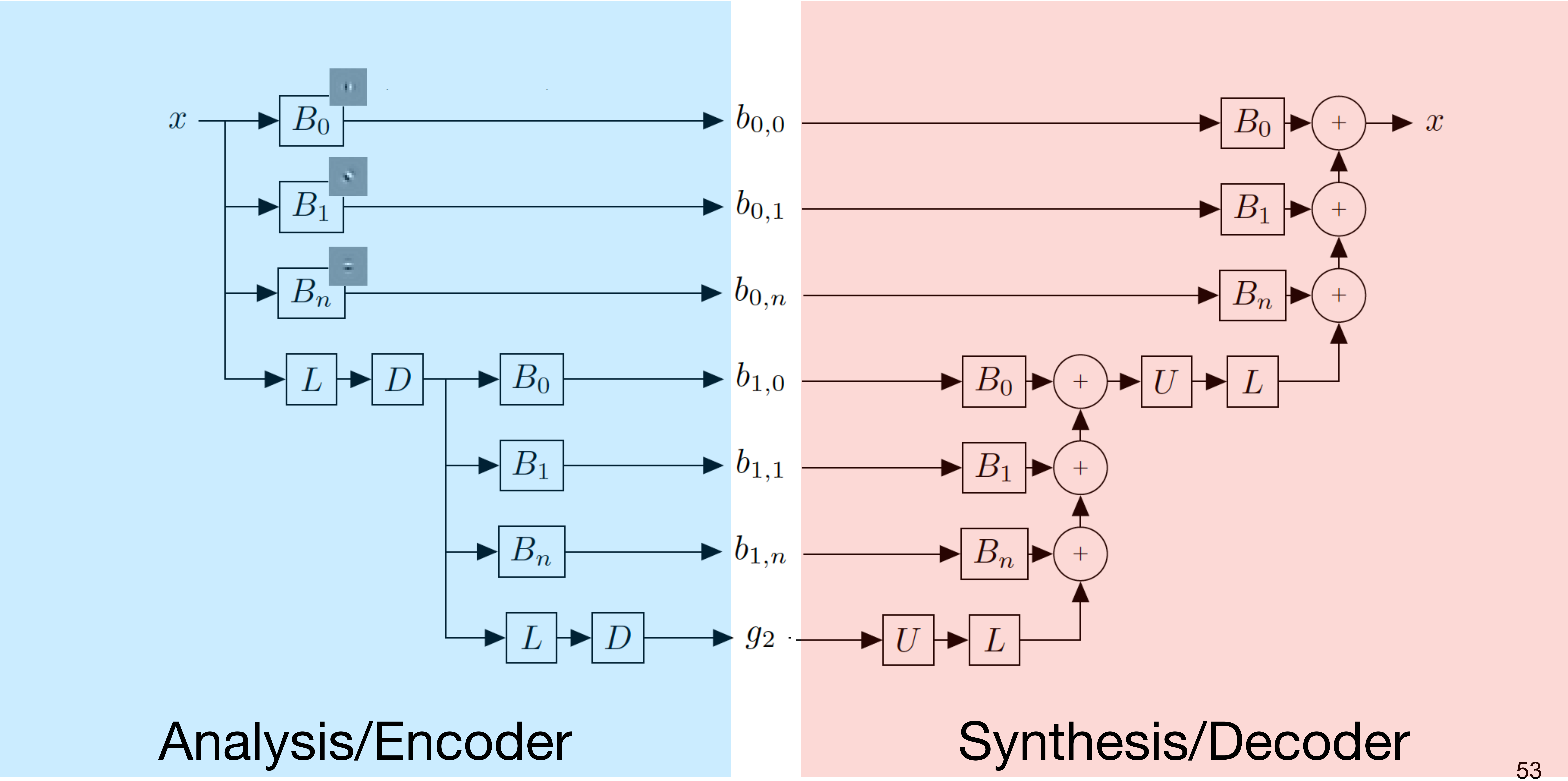
SGD

$\rightarrow f$

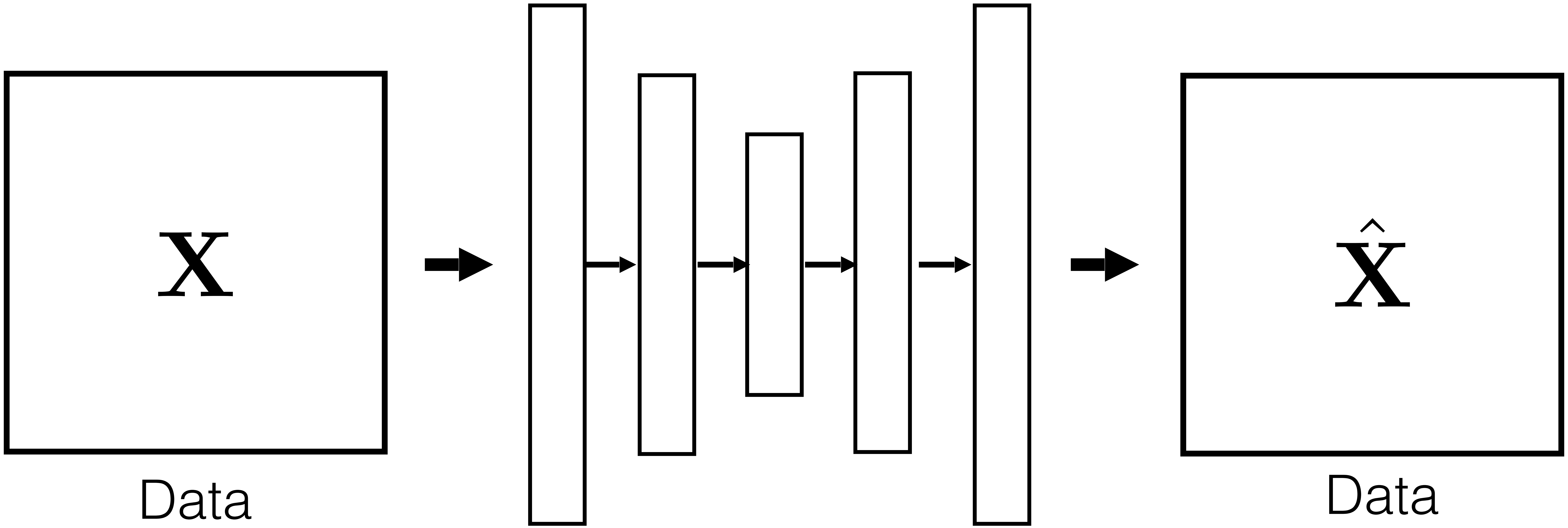


$\mathbf{X}$
$\mathcal{F}$
$\hat{\mathbf{X}} = \mathcal{F}(\mathbf{X})$
Image
Reconstructed image
Encoder
Decoder

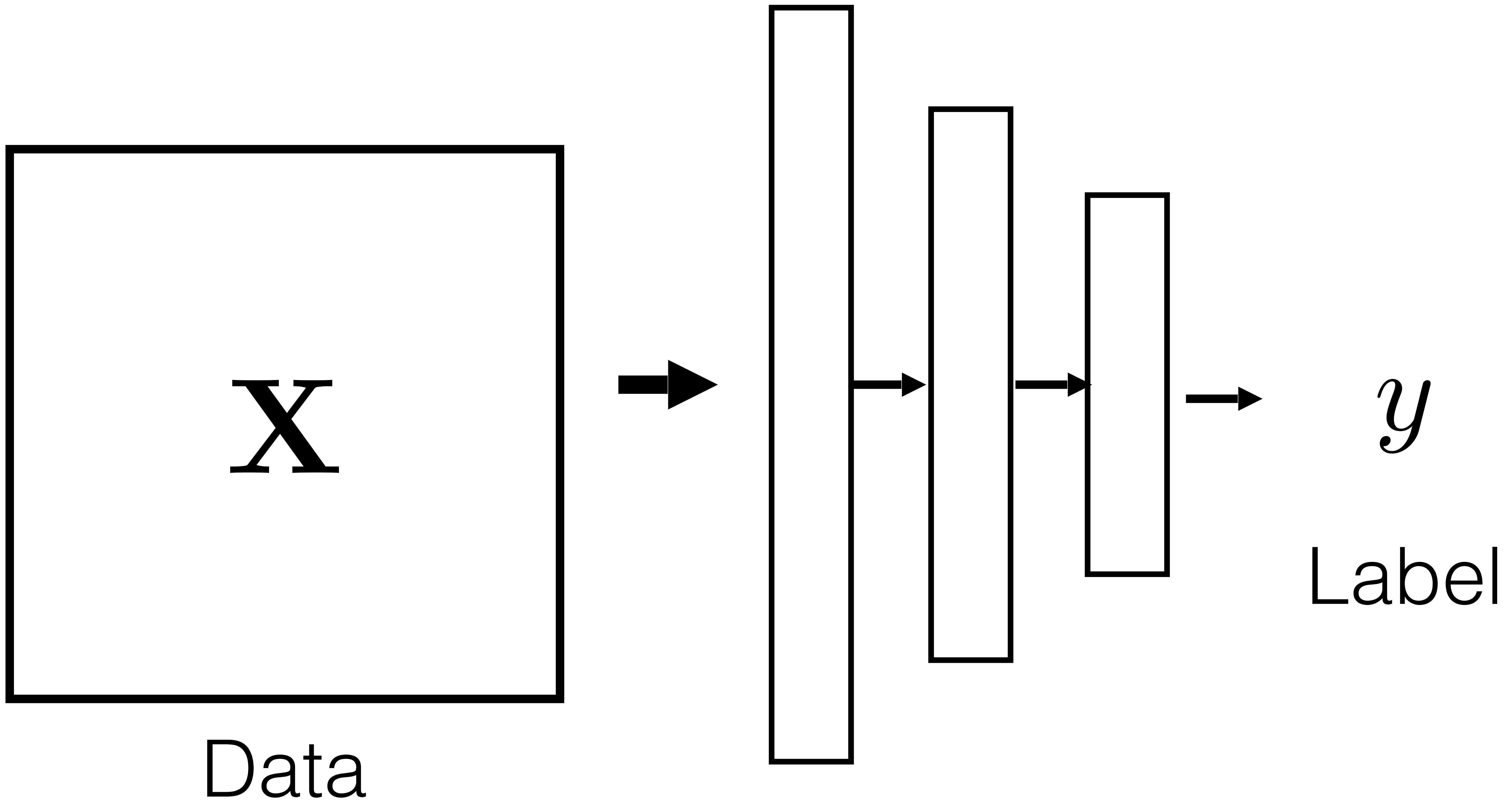
Steerable Pyramid — *A hard-coded autoencoder*



Data compression



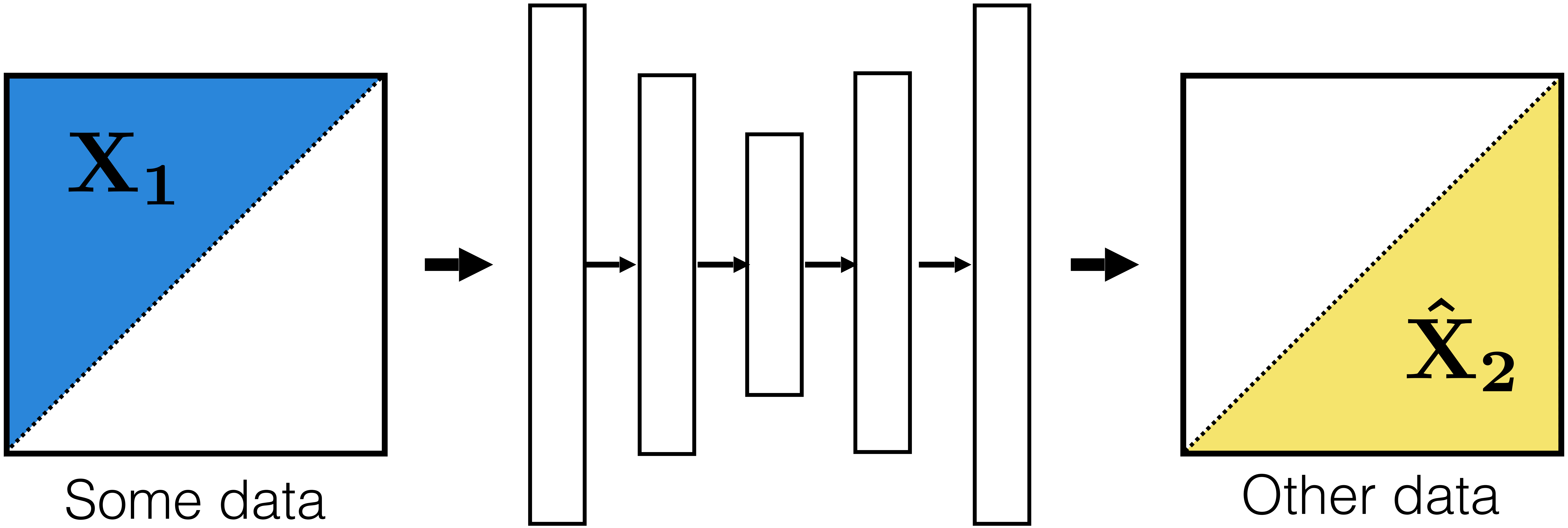
Label prediction

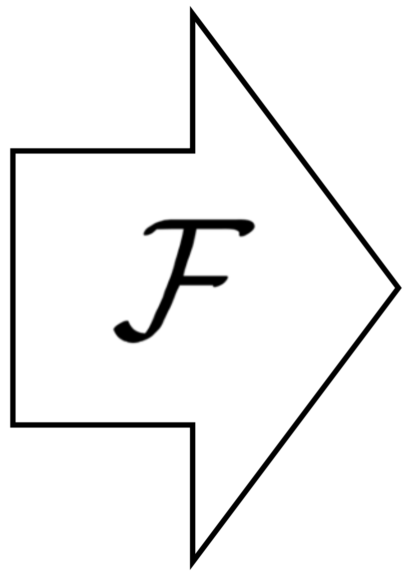


e.g., image classification

Data prediction

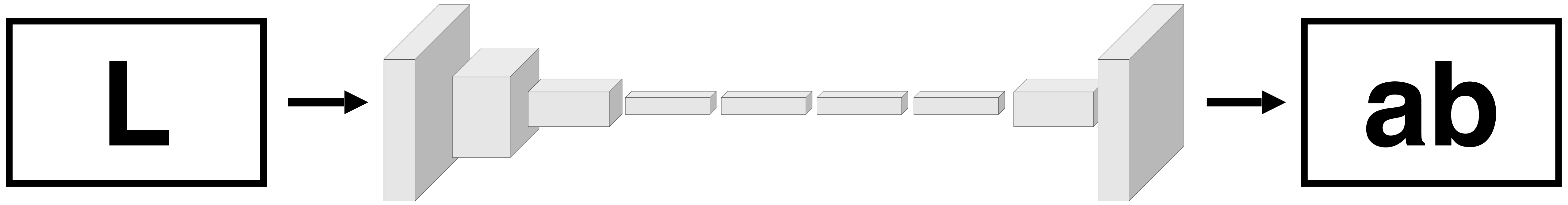
aka “self-supervised learning”





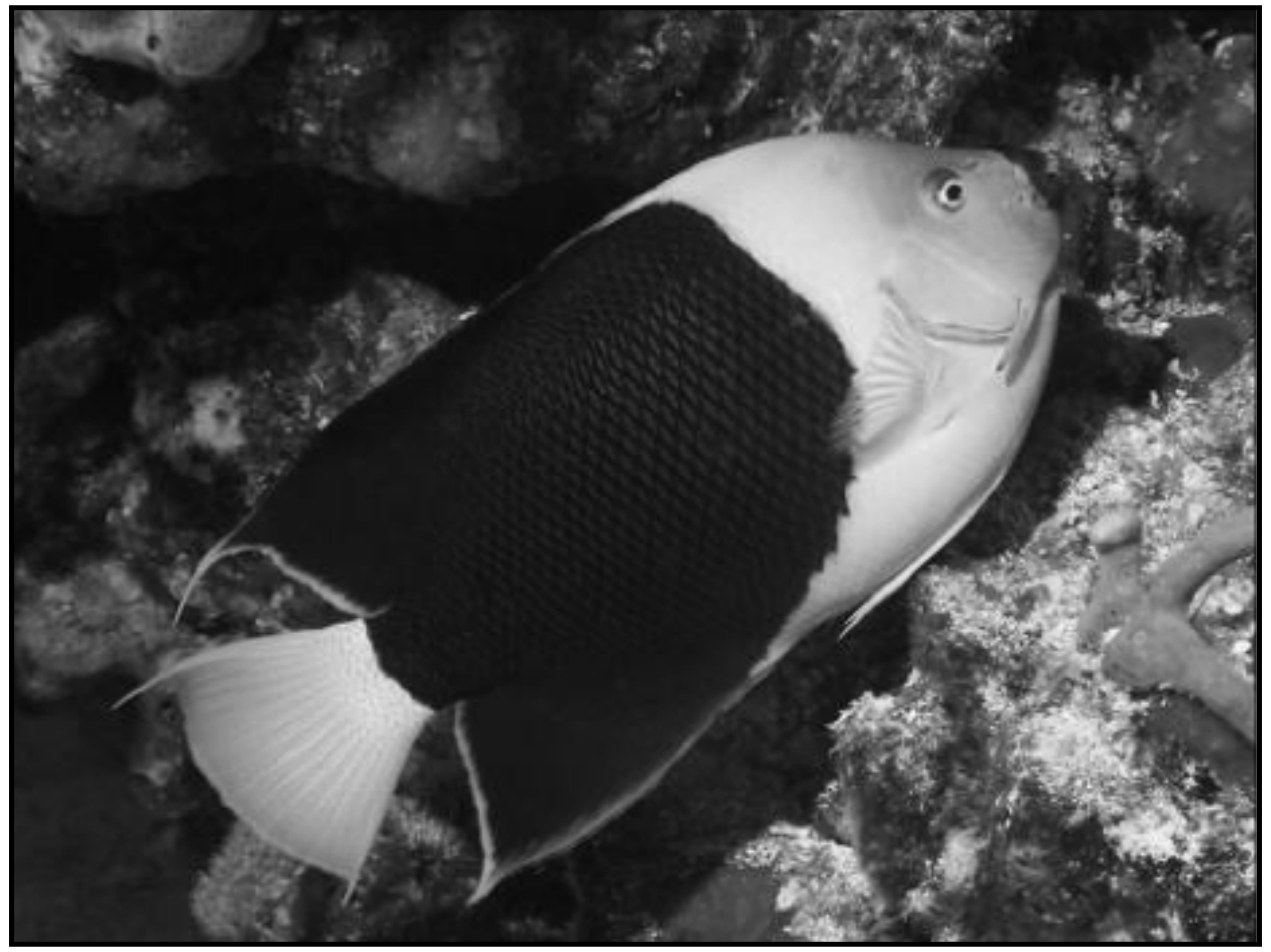
Grayscale image: L channel
$\mathbf{X} \in \mathbb{R}^{H \times W \times 1}$

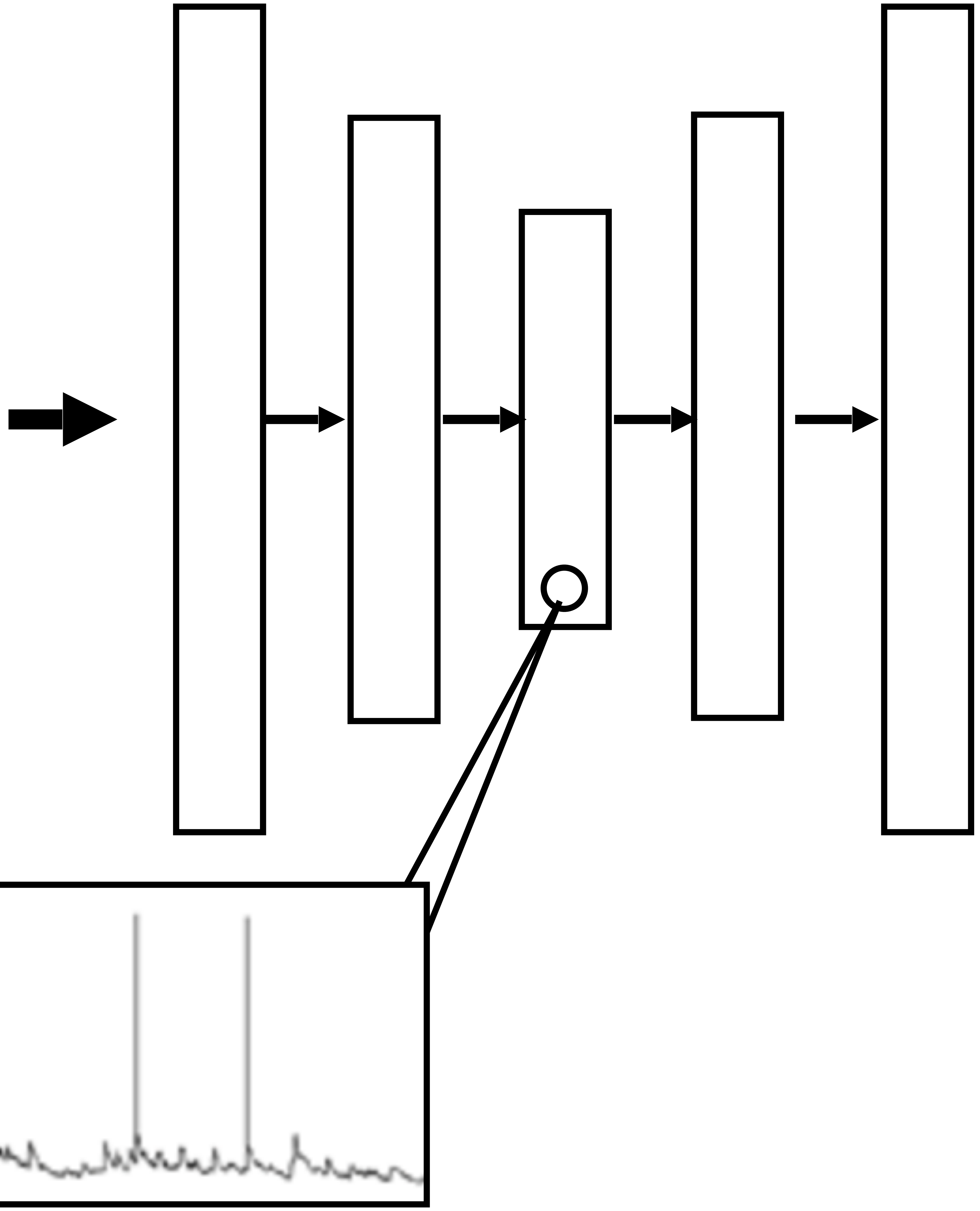
Color information: ab channels
$\widehat{\mathbf{Y}} \in \mathbb{R}^{H \times W \times 2}$



[Zhang, Isola, Efros, ECCV 2016]

Deep Net "Electrophysiology"





[Zeiler & Fergus, ECCV 2014]

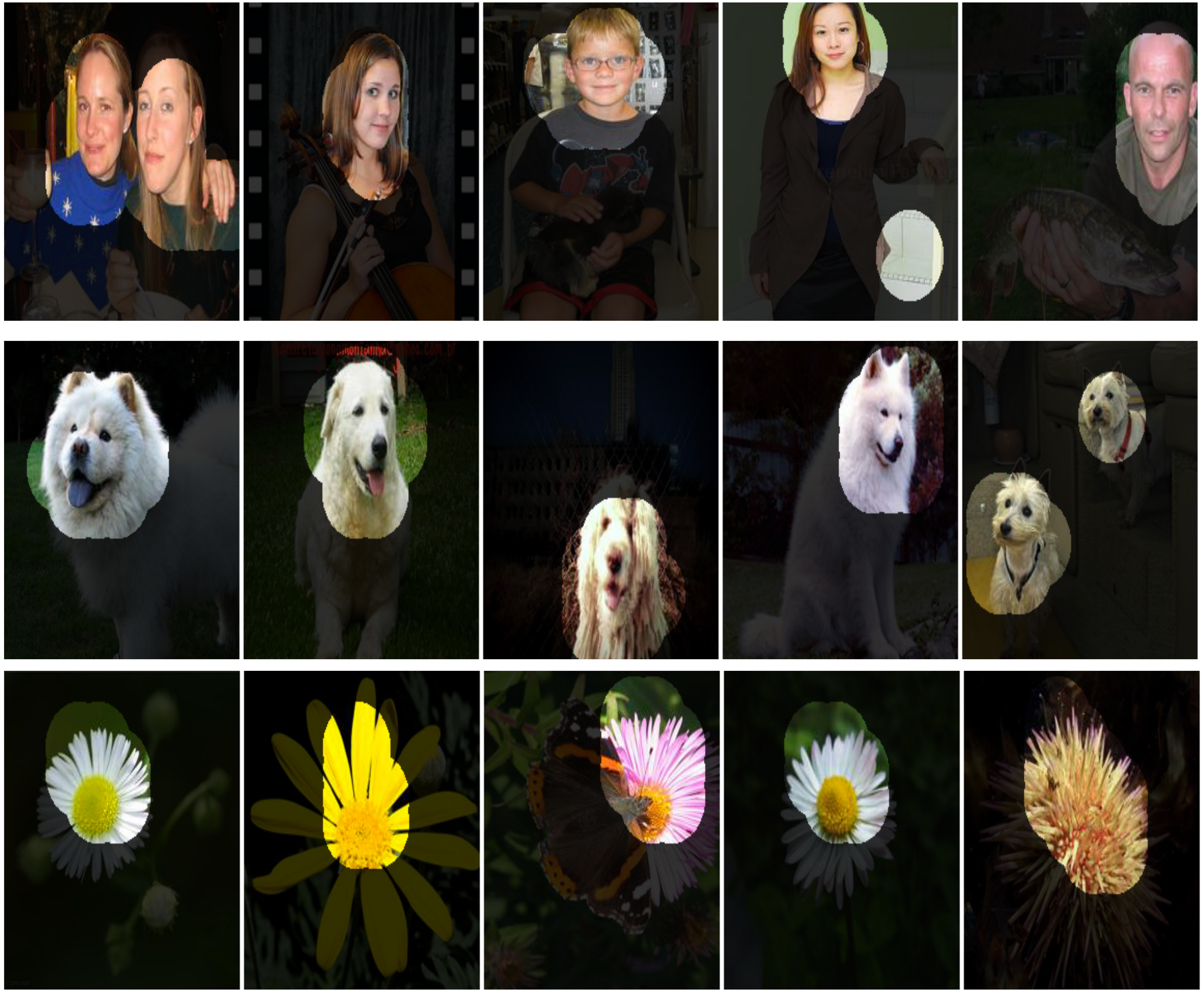
[Zhou et al., ICLR 2015]

Stimuli that drive selected neurons (conv5 layer)

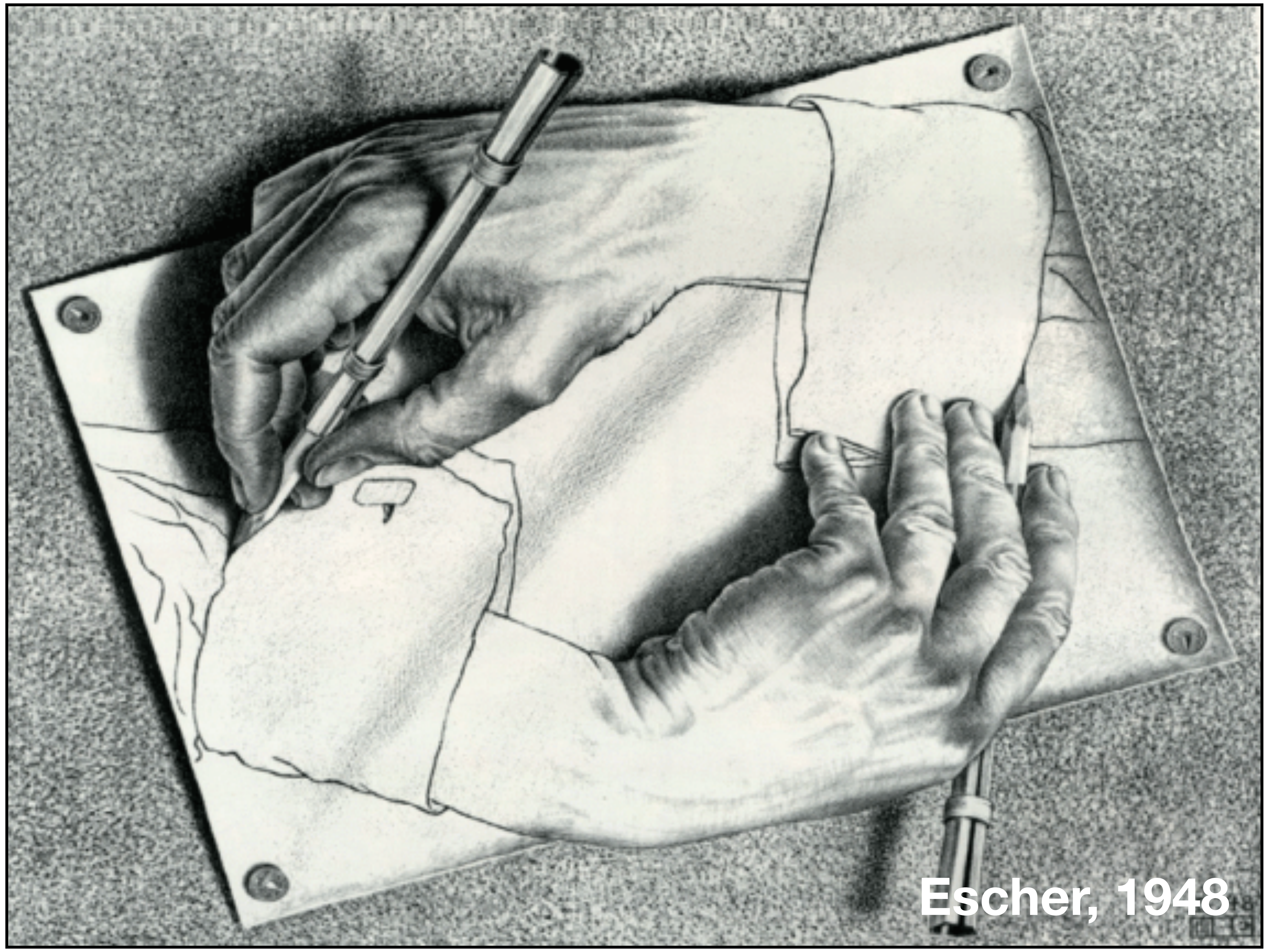
faces

dog faces

flowers



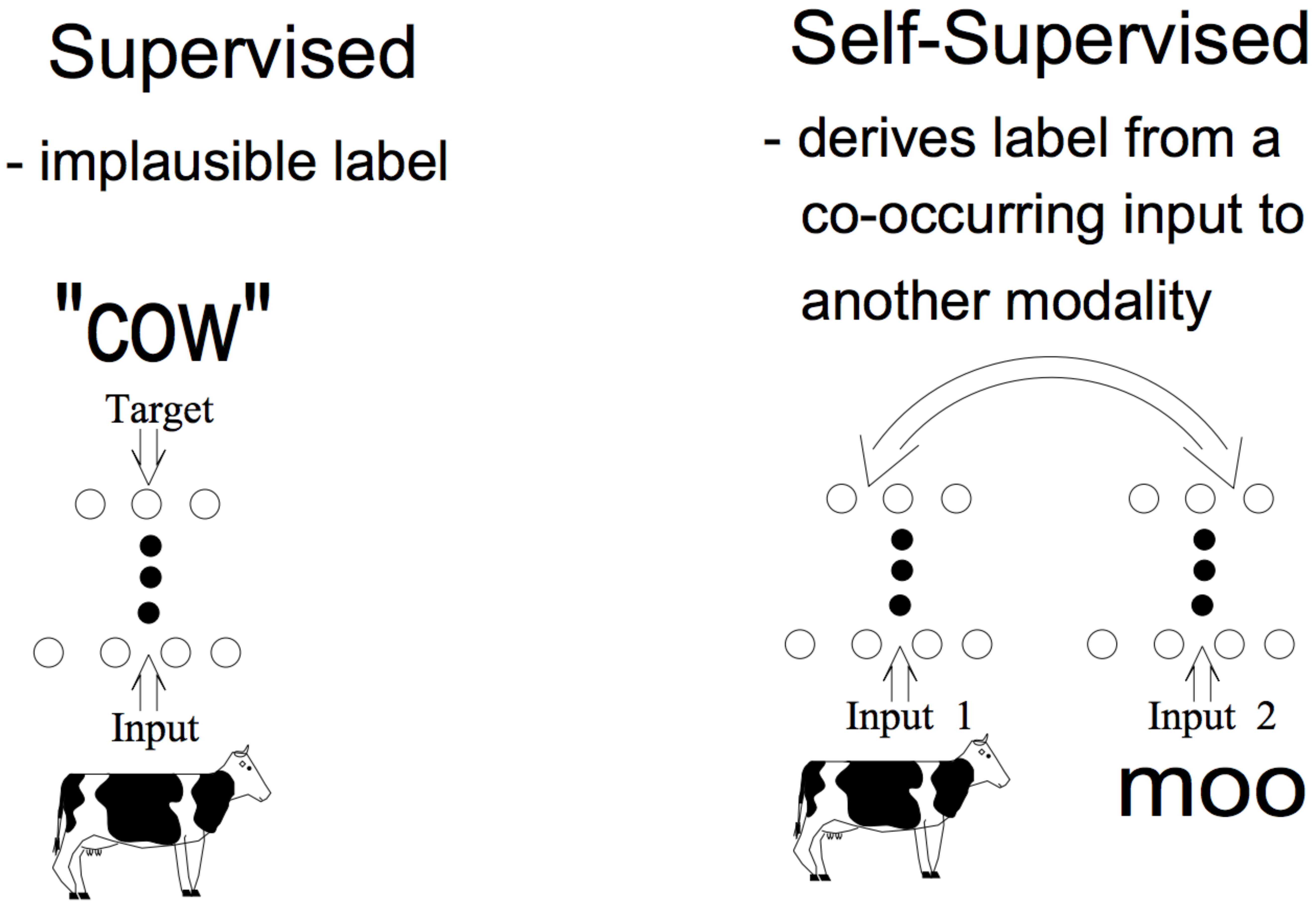
Self-supervised learning



Common trick:

- Convert “unsupervised” problem into “supervised” empirical risk minimization
- Do so by cooking up “labels” (prediction targets) from the raw data itself

Multisensory self-supervision



Virginia de Sa. *Learning Classification with Unlabeled Data*. NIPS 1994.

[see also “Six lessons from babies”, Smith and Gasser 2005]

“Multiview” self-supervised predictive learning

Data

$\{\mathbf{x}^{(i)}\}_{i=1}^{N} \rightarrow$

Learner

Objective

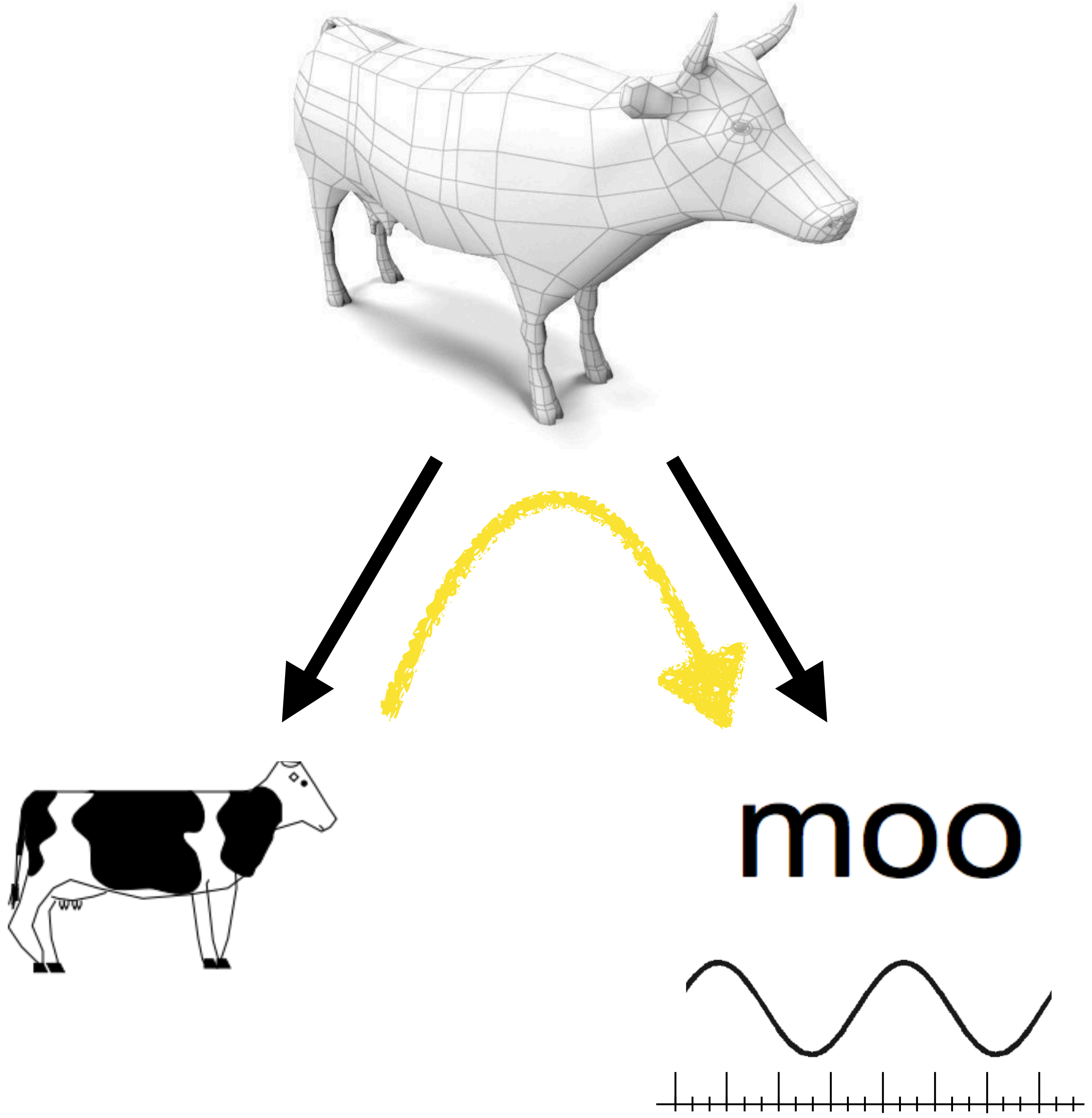
$$\arg\min_{f} \sum_{i} D(f(g(\mathbf{x}^{(i)})), h(\mathbf{x}^{(i)}))$$

$\rightarrow f$

Distance function

g and h are two “**views**” of the data x, e.g., two different sensory channels

The allegory of the cave

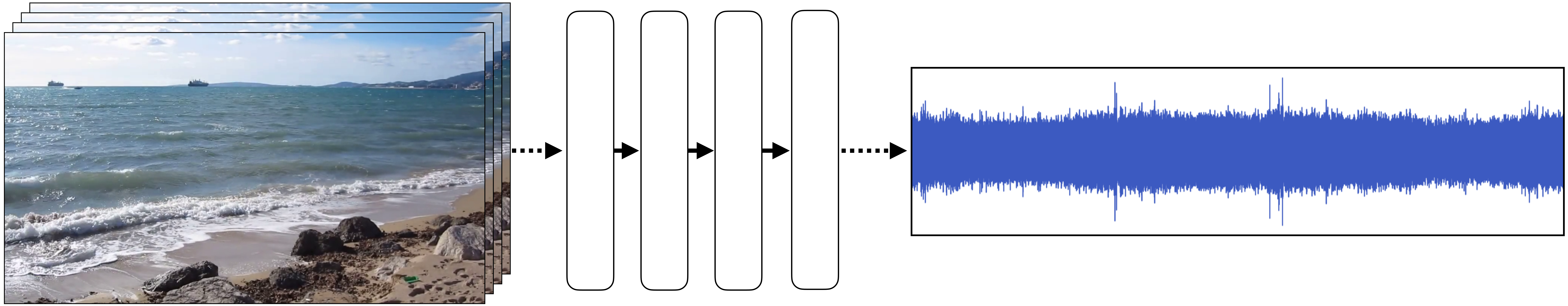


Ambient Sound Provides Supervision for Visual Learning

Andrew Owens Jiajun Wu Josh McDermott
William Freeman Antonio Torralba

[Slide credit: Andrew Owens]

Predicting ambient sound



[Slide credit: Andrew Owens]

What did the model learn?

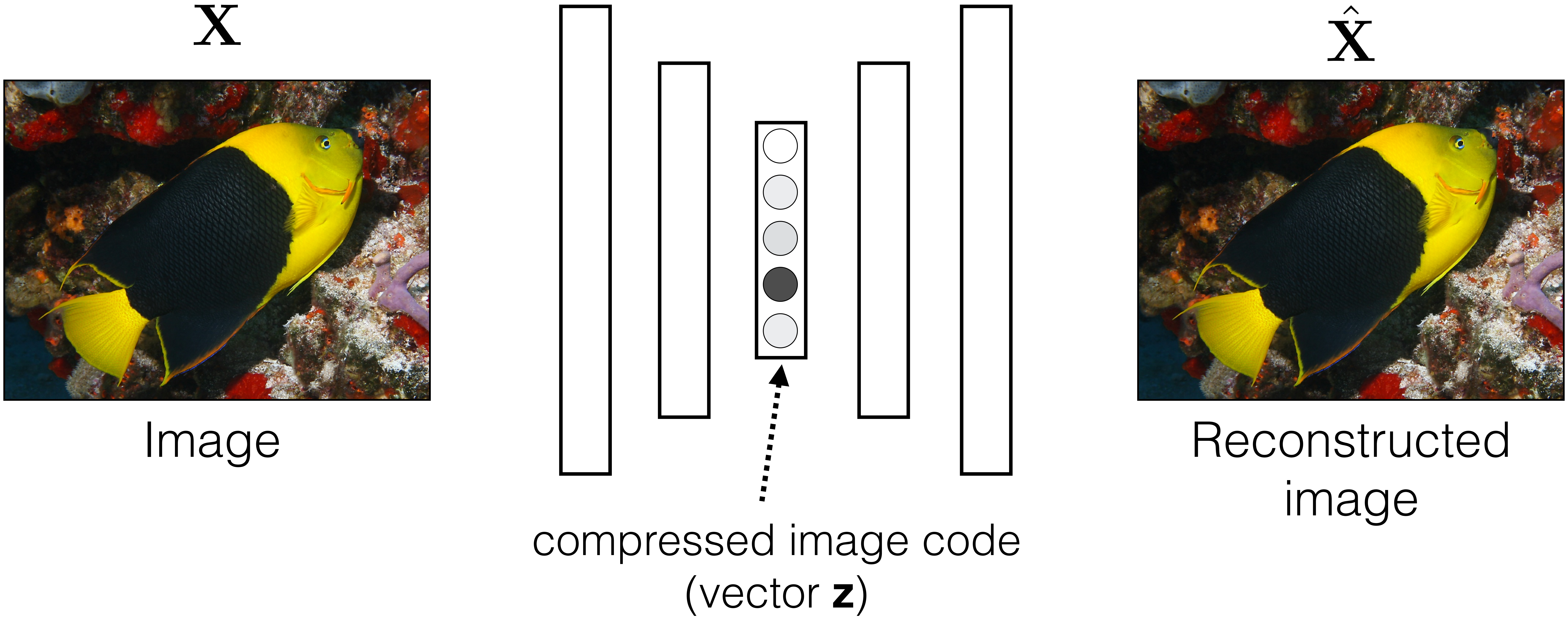
Unit #90 of 256

Strongest responses in dataset

Visualization method from (Zhou 2015)

[Slide credit: Andrew Owens]

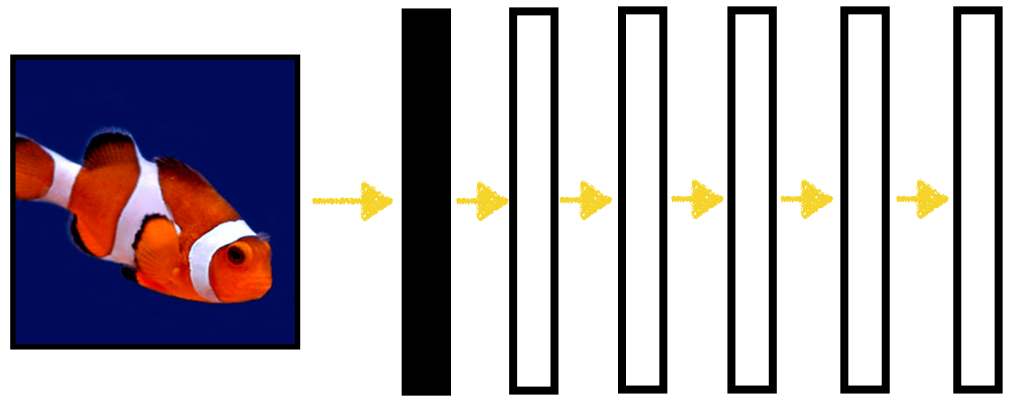
[Slide credit: Andrew Owens]



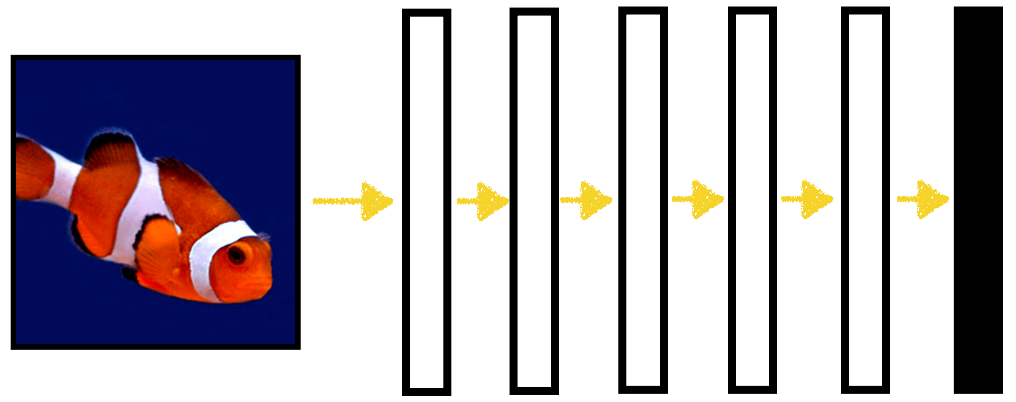
Is the code informative about object class y?

Logistic regression:

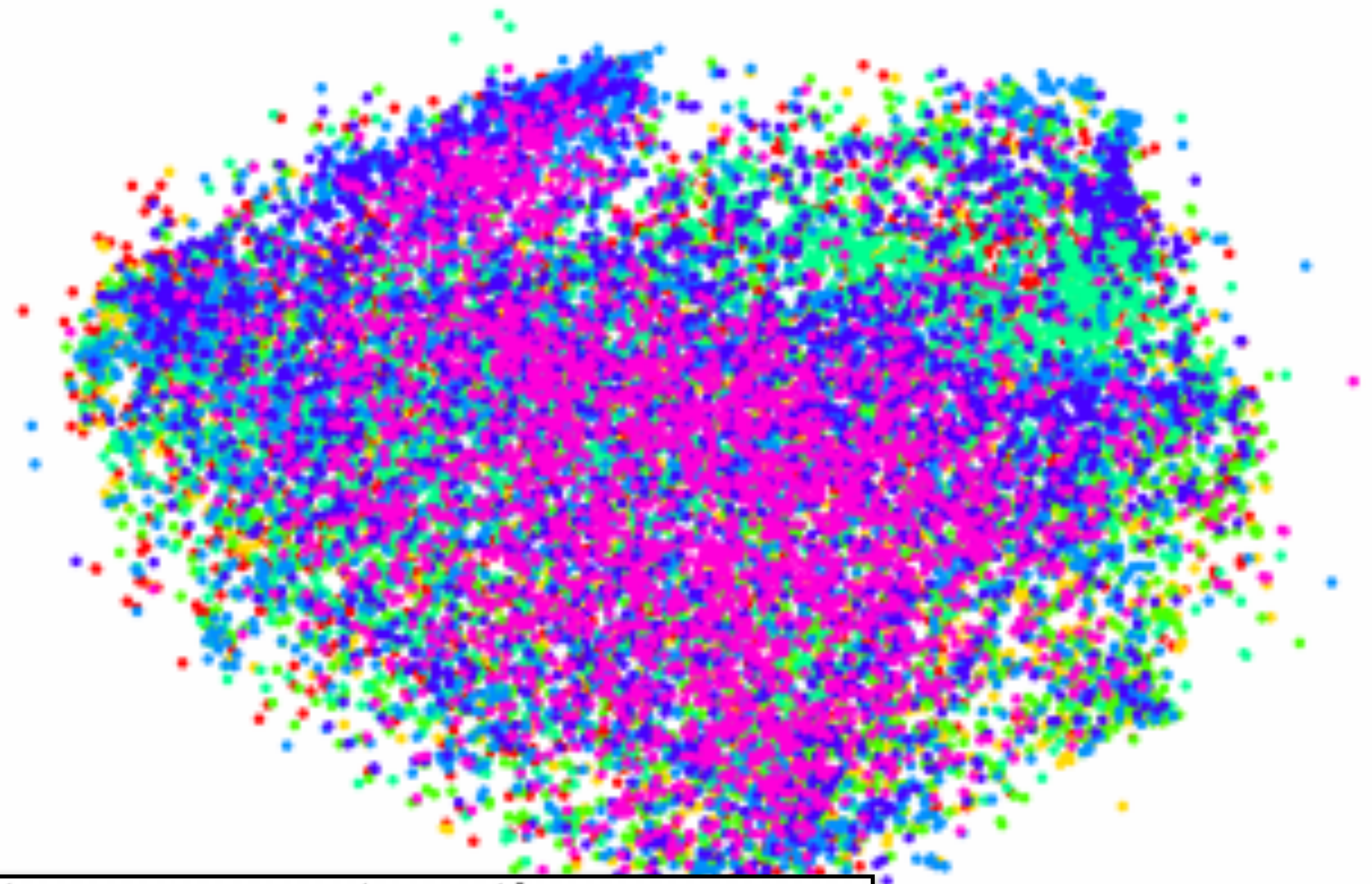
$$y = \sigma(\mathbf{W}\mathbf{z} + \mathbf{b})$$

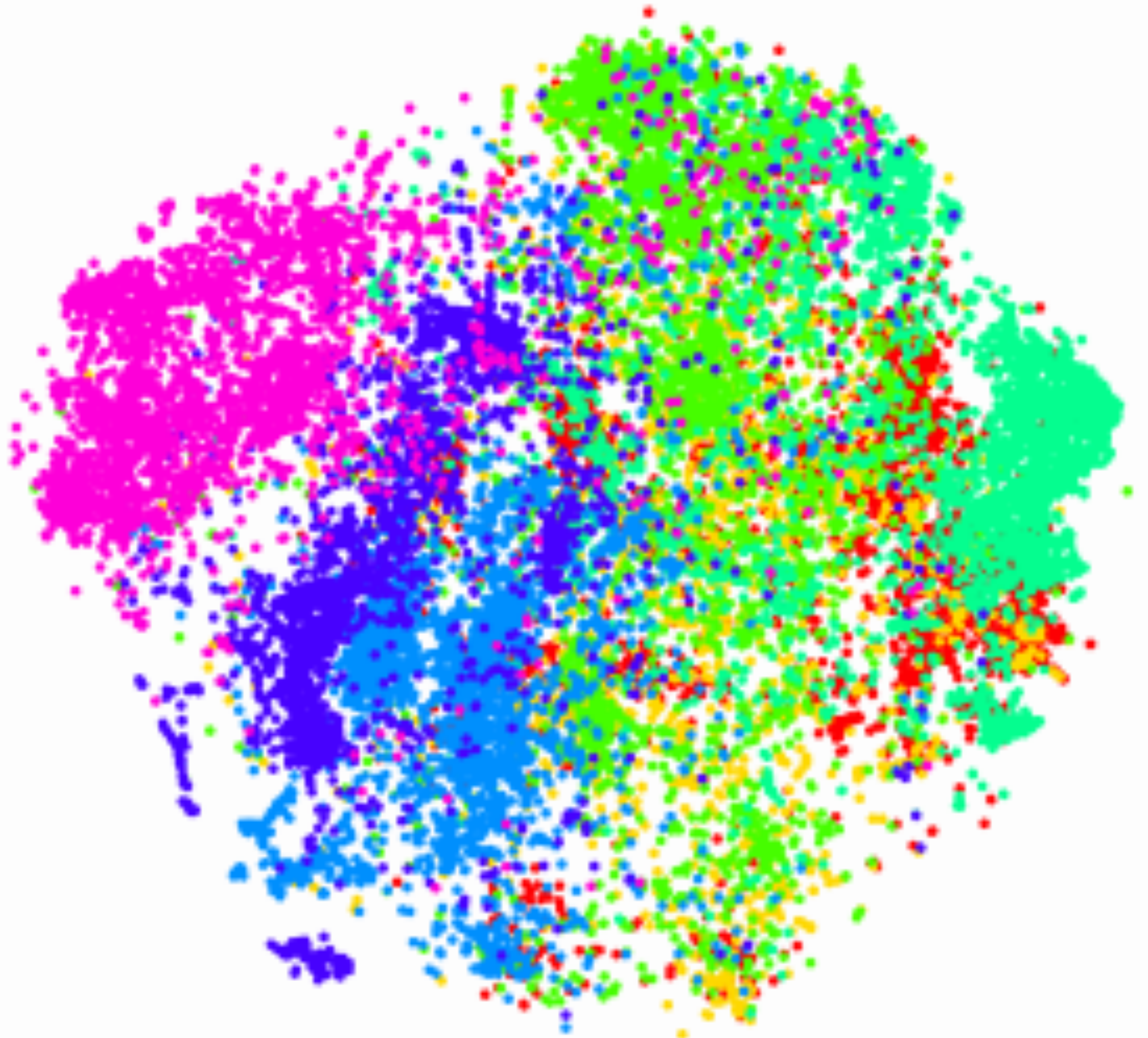


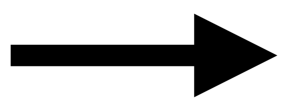
Layer 1 representation



Layer 6 representation



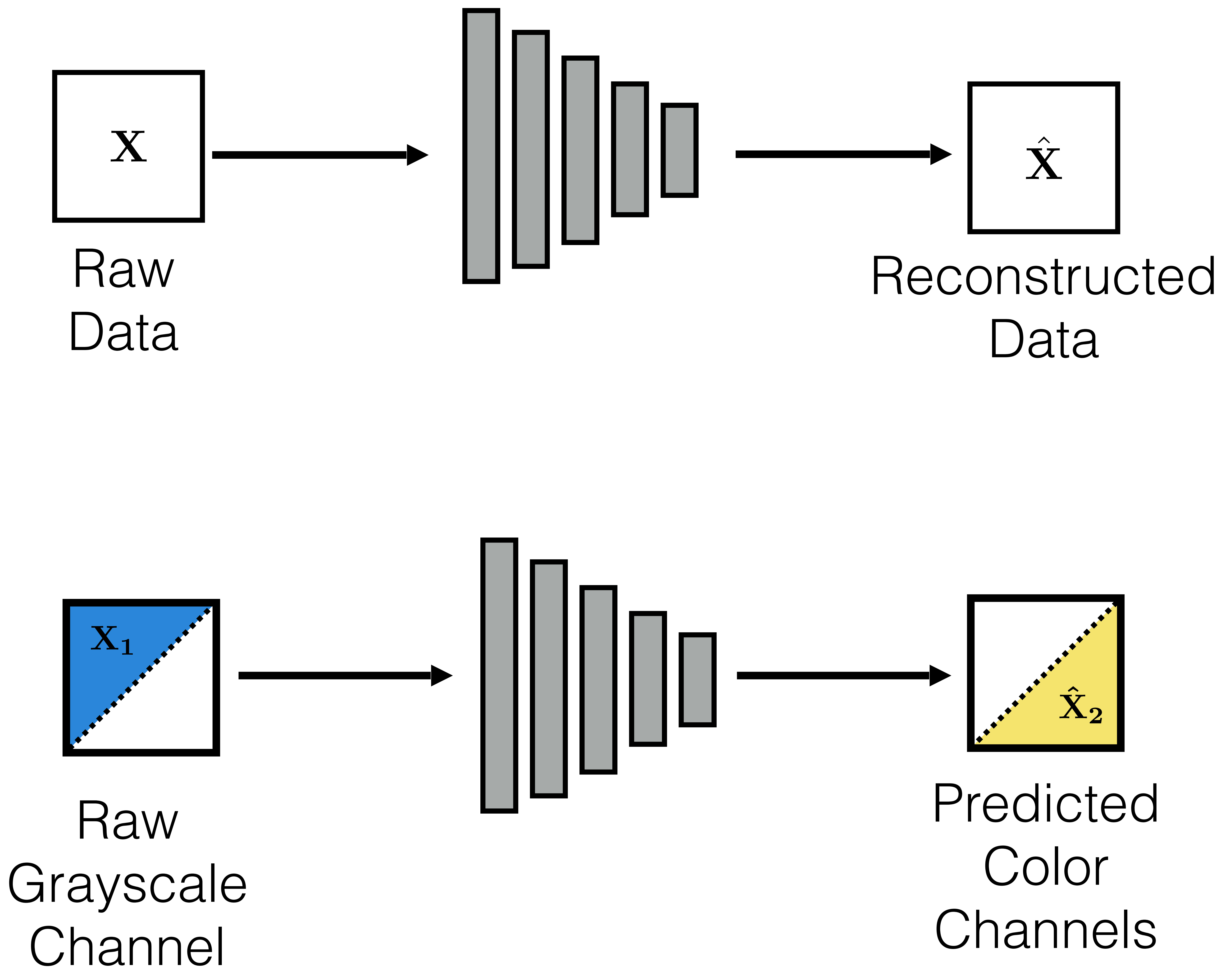


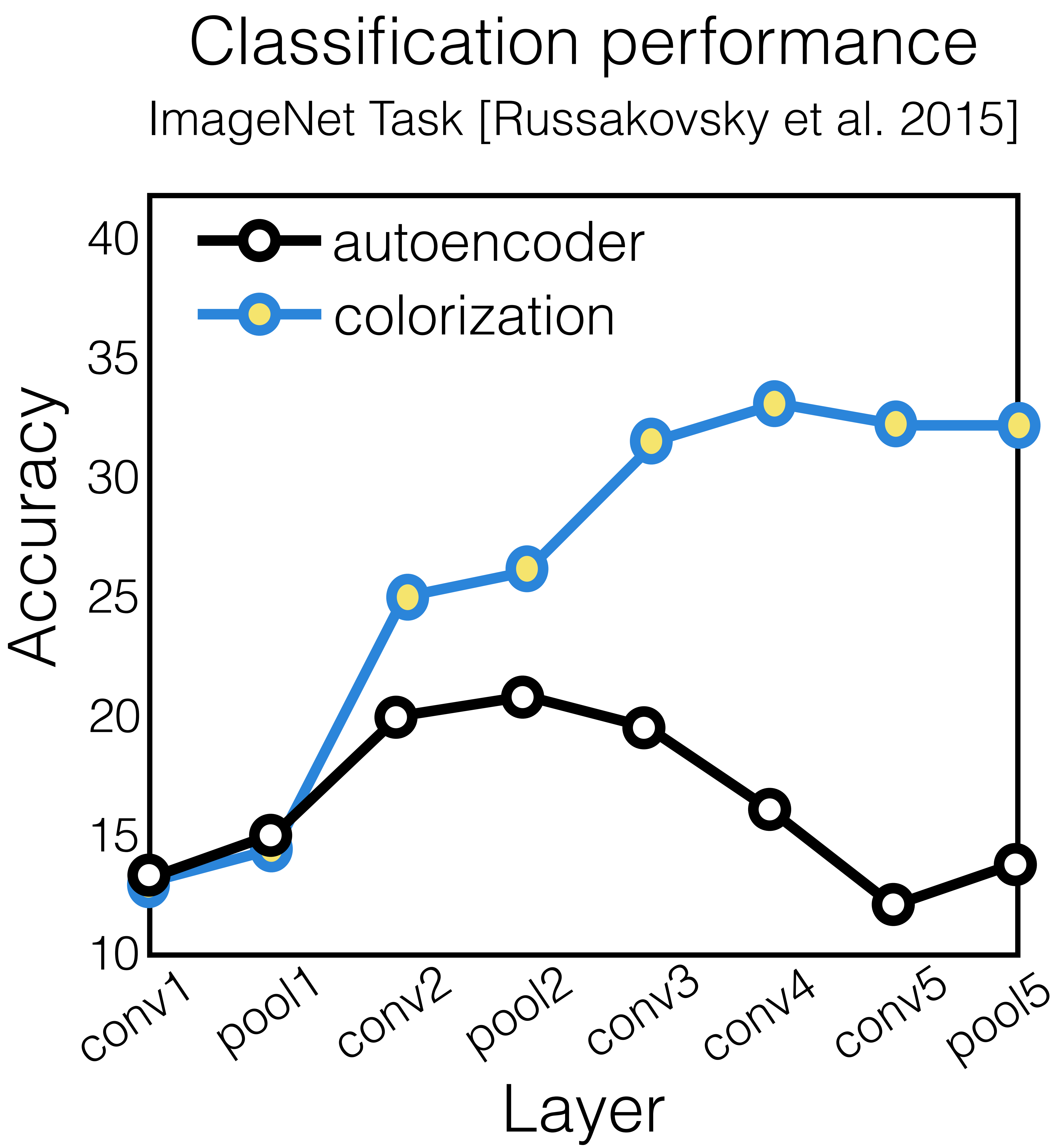


- structure, construction
- covering
- commodity, trade good, good
- conveyance, transport
- invertebrate
- bird
- hunting dog

[DeCAF, Donahue, Jia, et al. 2013]

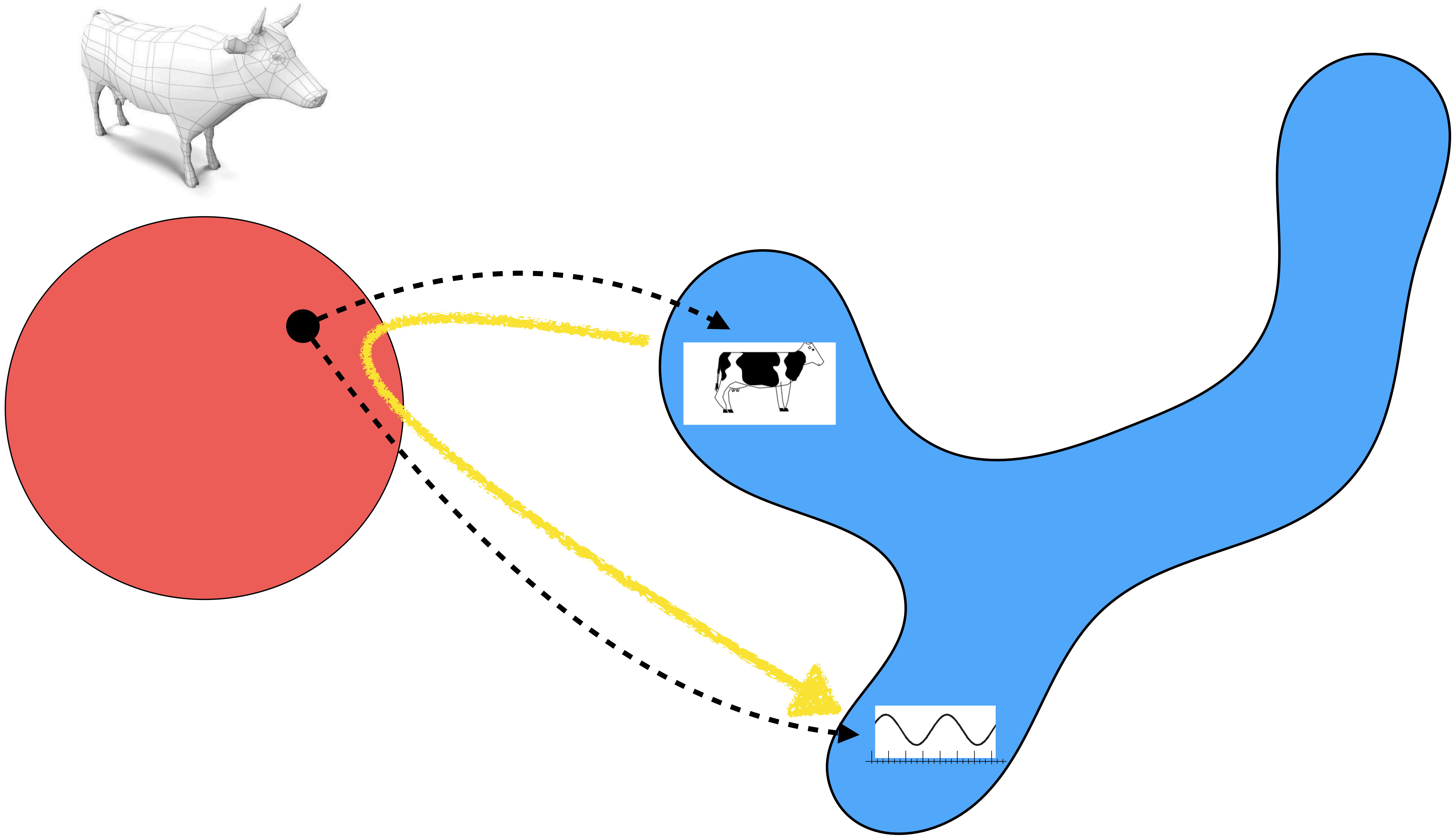
[Visualization technique : t-sne, van der Maaten & Hinton, 2008]

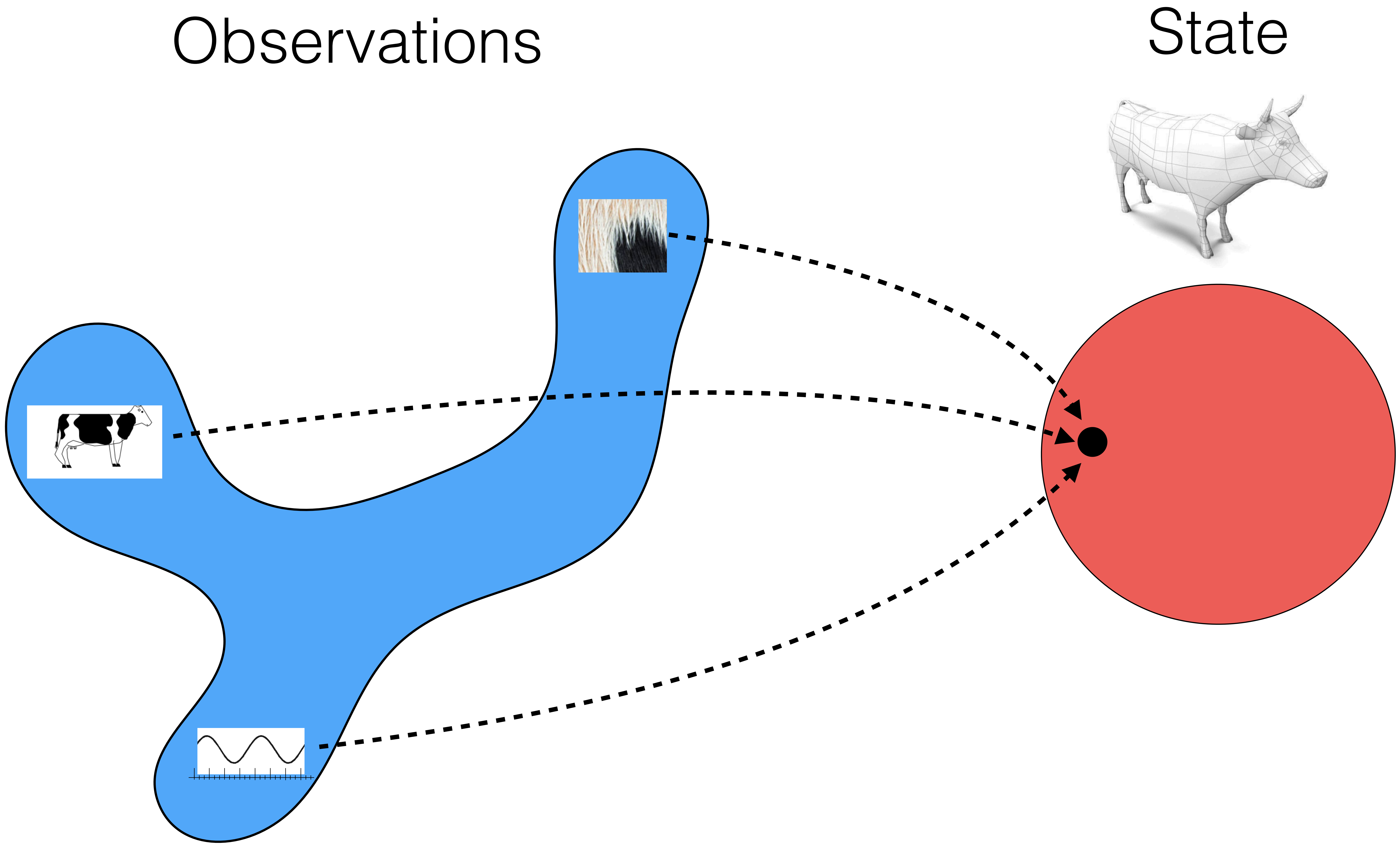

x
Raw
Data
x̂
Reconstructed
Data
x_1
Raw
Grayscale
Channel
$\hat{x}_2$
Predicted
Color
Channels


Classification performance
ImageNet Task [Russakovsky et al. 2015]
autoencoder
colorization
Accuracy
40
35
30
25
20
15
10
conv1
pool1
conv2
pool2
conv3
conv4
conv5
pool5
Layer

State

Observations

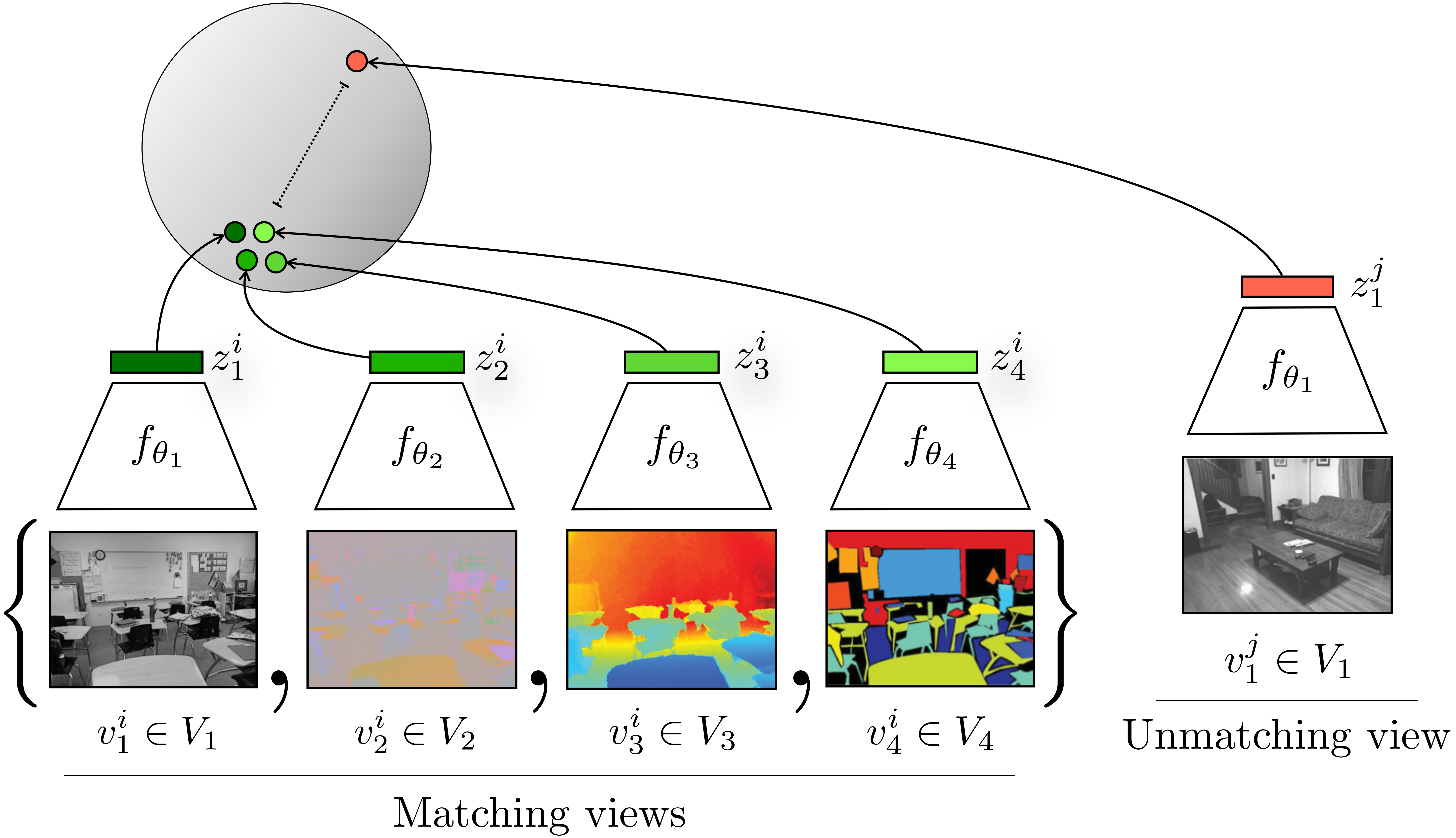




The way you measure the world does not change the underlying state

Contrastive Multiview Coding

[Tian, Krishnan, Isola, ECCV 2020]



"Multiview" self-supervised **contrastive** learning

Data

$\{\mathbf{x}^{(i)}\}_{i=1}^{N} \quad \rightarrow$

Learner

Objective

$$\arg\min_{f} \sum_{i,j} D(f_1(g(\mathbf{x}^{\boxed{(i)}})), f_2(h(\mathbf{x}^{\boxed{(i)}}))+$$

$$-D(f_1(g(\mathbf{x}^{\boxed{(i)}})), f_2(h(\mathbf{x}^{\boxed{(j)}})))$$

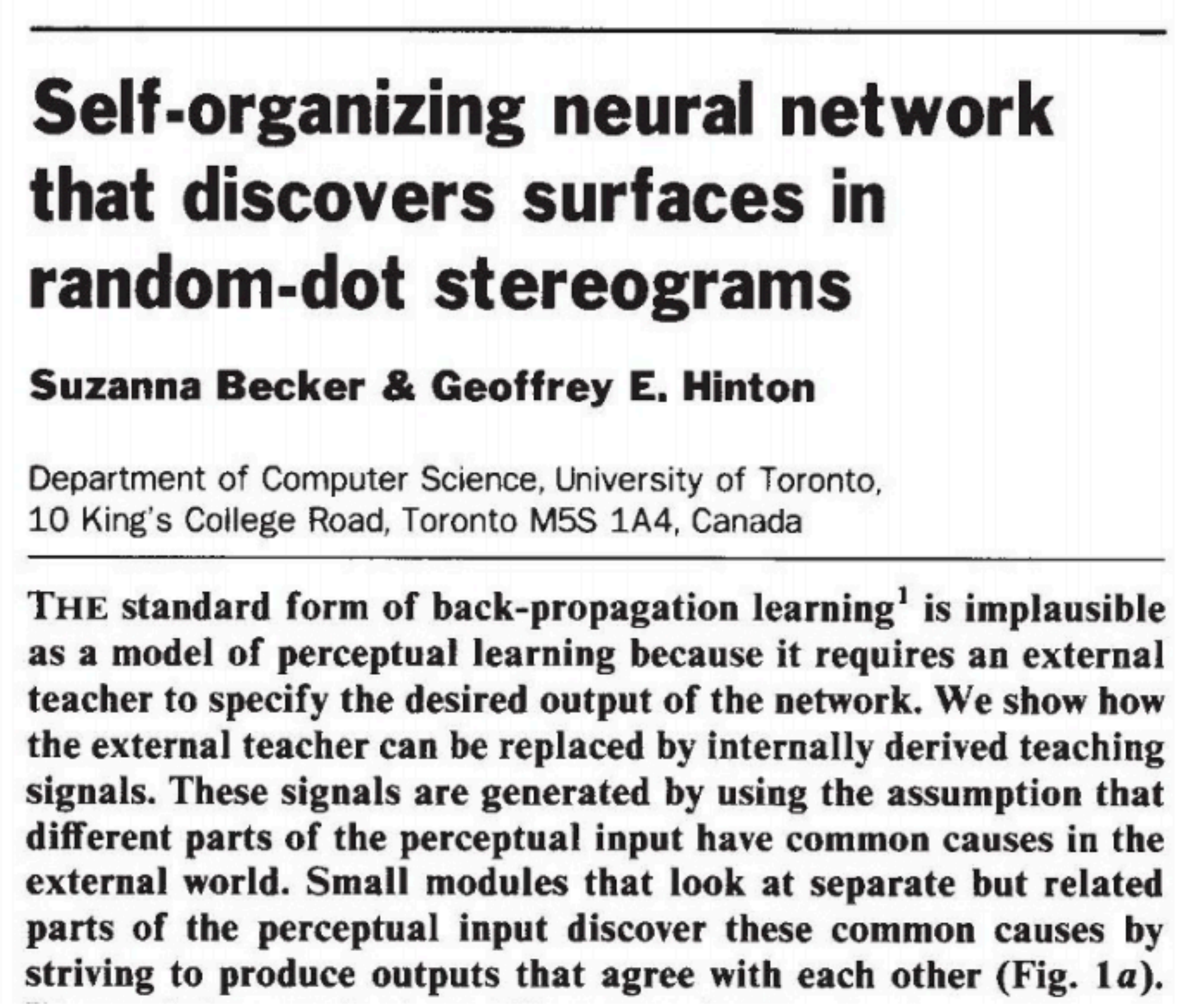
$\rightarrow \quad f_1, f_2$

Distance function

g and h are two "**views**" of the data x,
e.g., two different sensory channels

SimCLR

[Chen, Kornblith, Norouzi, Hinton, ICML 2020]



Self-organizing neural network that discovers surfaces in random-dot stereograms

Suzanna Becker & Geoffrey E. Hinton

Department of Computer Science, University of Toronto,
10 King's College Road, Toronto M5S 1A4, Canada

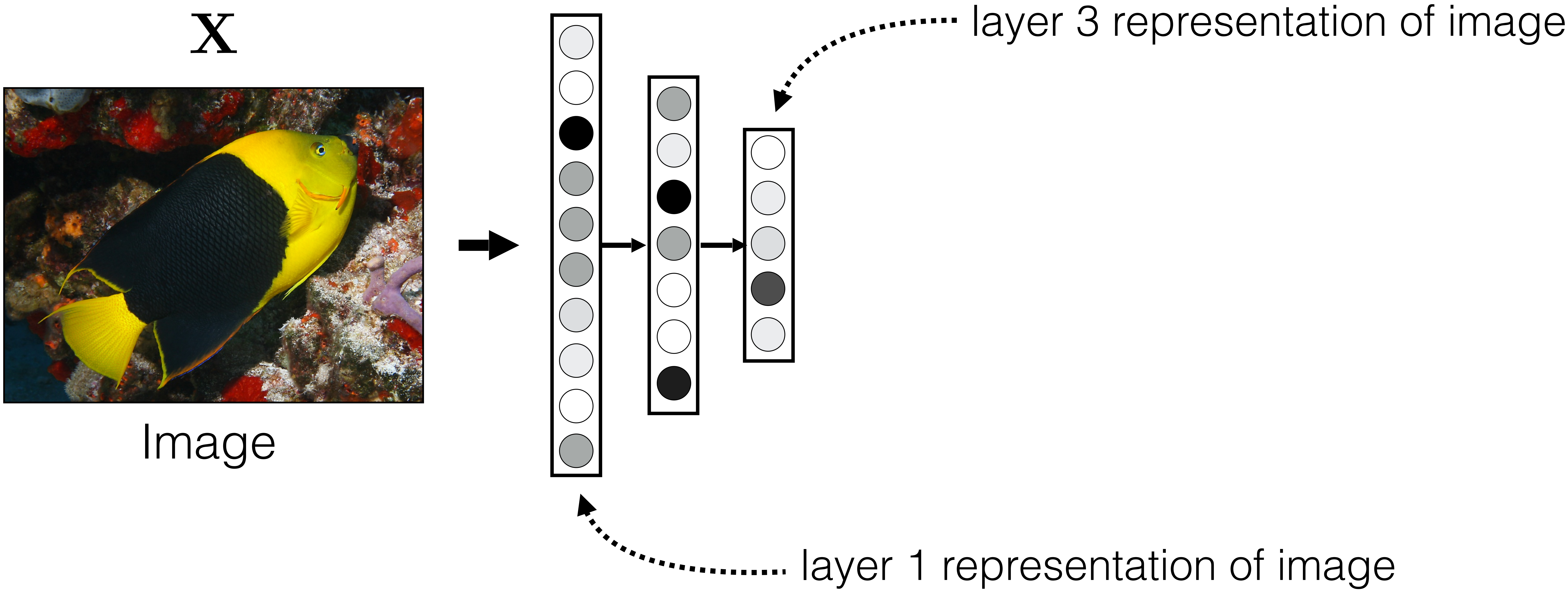
THE standard form of back-propagation learning[1] is implausible as a model of perceptual learning because it requires an external teacher to specify the desired output of the network. We show how the external teacher can be replaced by internally derived teaching signals. These signals are generated by using the assumption that different parts of the perceptual input have common causes in the external world. Small modules that look at separate but related parts of the perceptual input discover these common causes by striving to produce outputs that agree with each other (Fig. 1*a*).

[c.f. Becker & Hinton, Nature 1992]

How to represent words as numbers

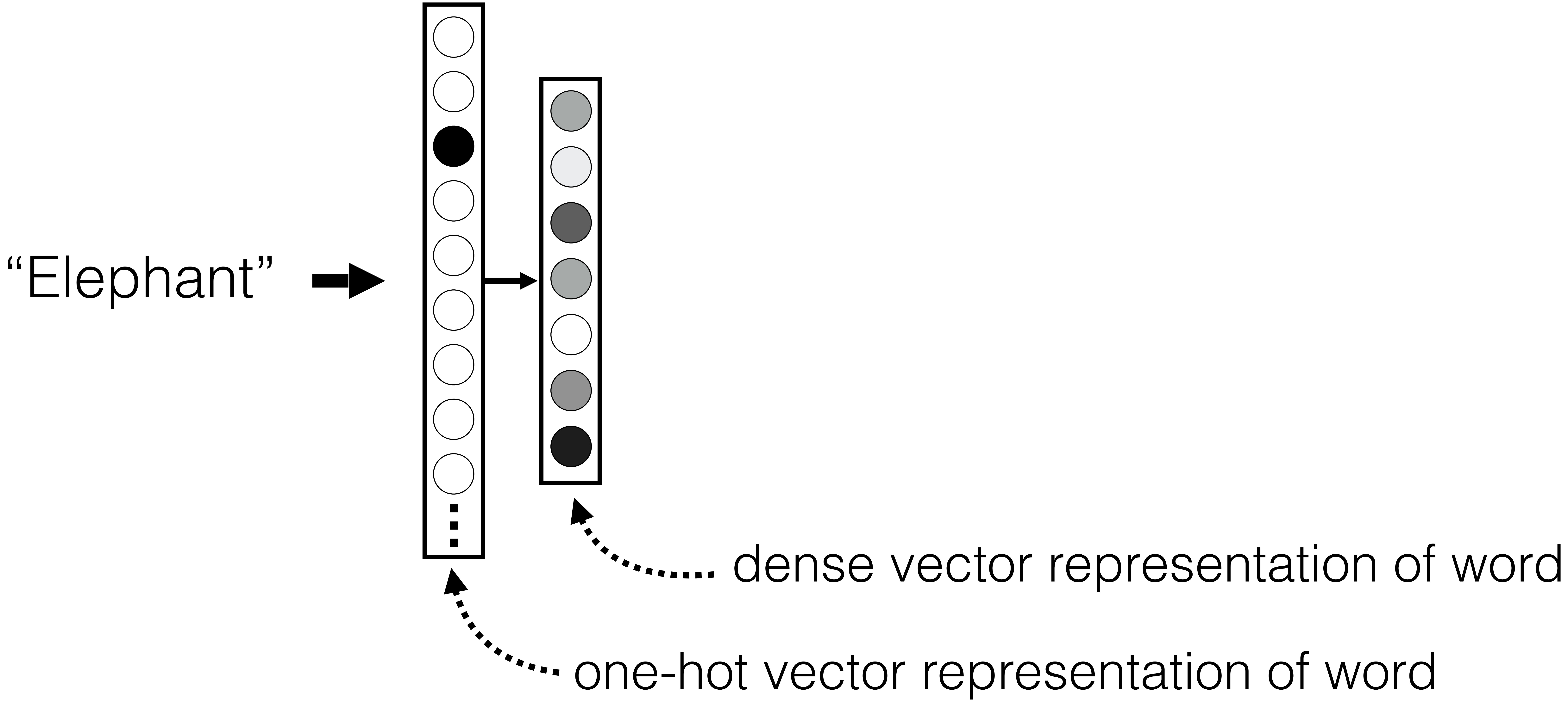
"Fish"	$\longrightarrow$	[**1**,0,0,0,0,0,0,…]
"Shark"	$\longrightarrow$	[0,**1**,0,0,0,0,0,…]
"Whale"	$\longrightarrow$	[0,0,**1**,0,0,0,0,…]
"Water"	$\longrightarrow$	[0,0,0,**1**,0,0,0,…]
"Cat"	$\longrightarrow$	[0,0,0,0,**1**,0,0,…]
"Couch"	$\longrightarrow$	[0,0,0,0,0,**1**,0,…]
"Sun"	$\longrightarrow$	[0,0,0,0,0,0,**1**,…]

im2vec

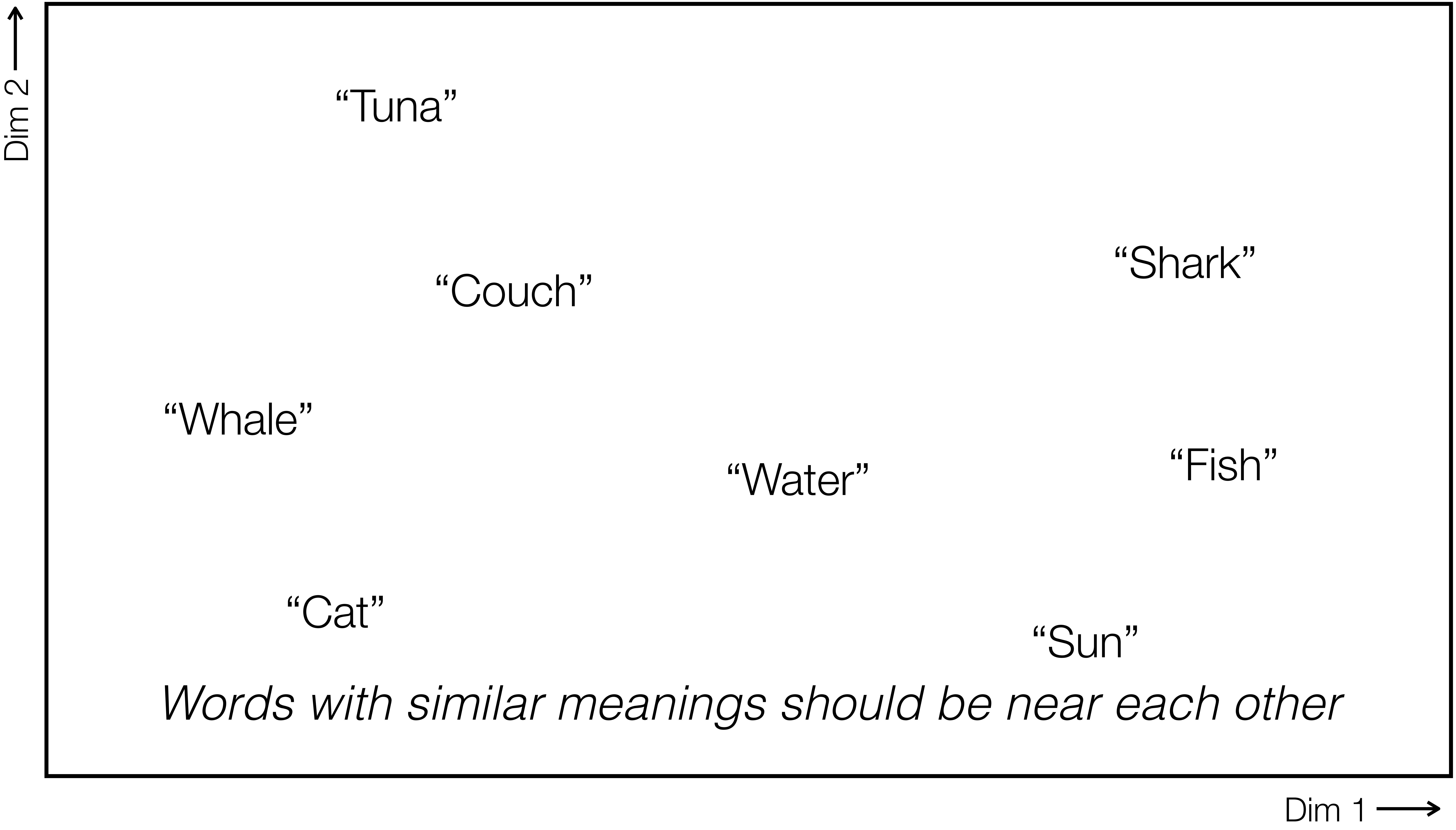


Represent image as a neural **embedding** — a vector/tensor of neural activations
(perhaps representing a vector of detected texture patterns or object parts)

word2vec



X2vec methods are also called embeddings of X, e.g., a **word embedding**


Dim 2
“Tuna”
“Shark”
“Couch”
“Whale”
“Fish”
“Water”
“Cat”
“Sun”
Words with similar meanings should be near each other
Dim 1

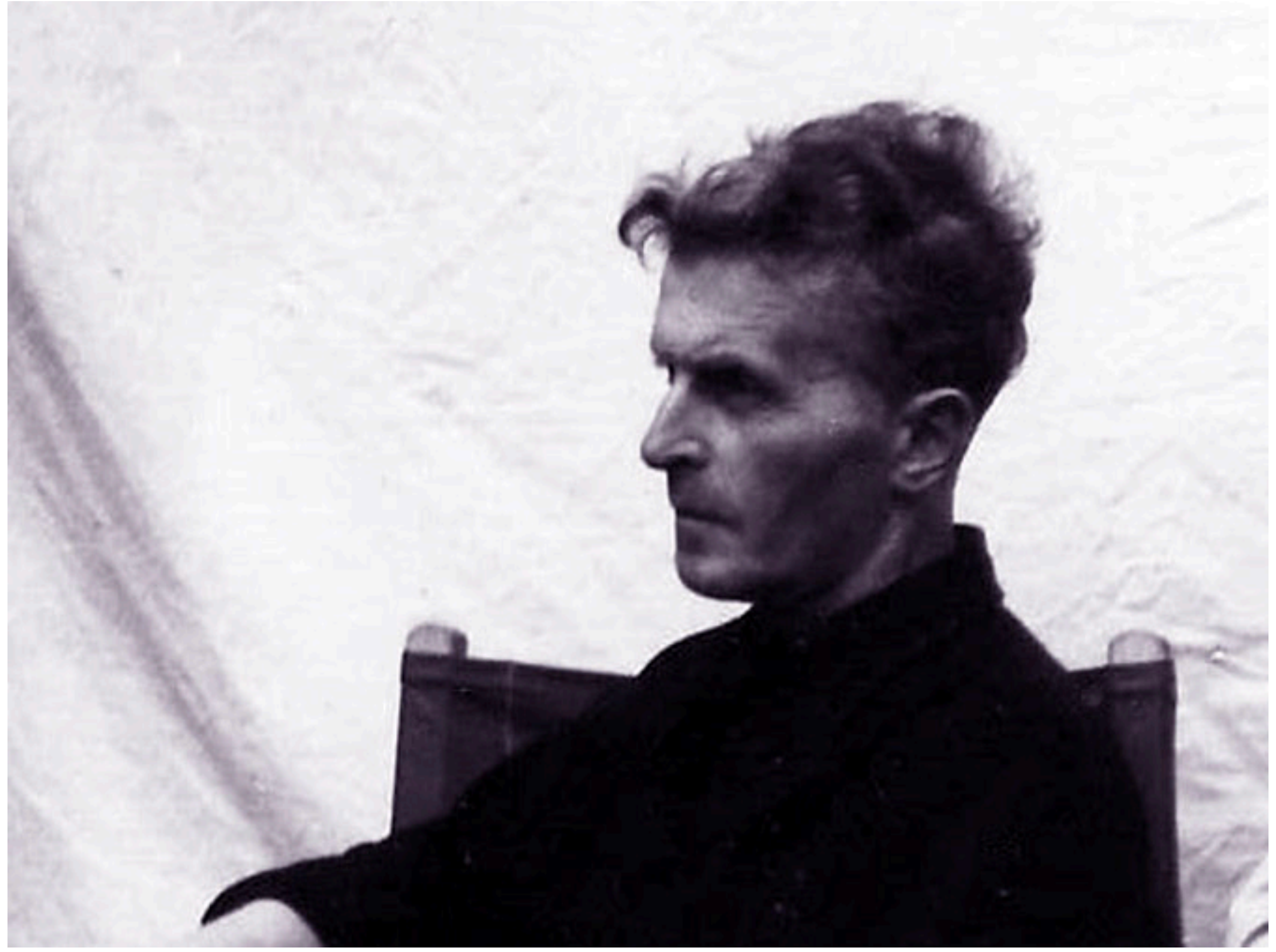
word2vec

Words with similar meanings should be near each other

Proxy: words that are used in the same context tend to have similar meanings

words with similar contexts should be near each other

“Meaning is use” — Wittgenstein



Next to the 'sofa' is a desk, and a 'person' is sitting behind it.

'armchair'
'bench'
'chair'
'deck chair'
'ottoman'
'seat'
'stool'
'swivel chair'
'loveseat'
...

'man'
'woman'
'child'
'teenager'
'girl'
'boy'
'baby'
'daughter'
'son'
...

word2vec

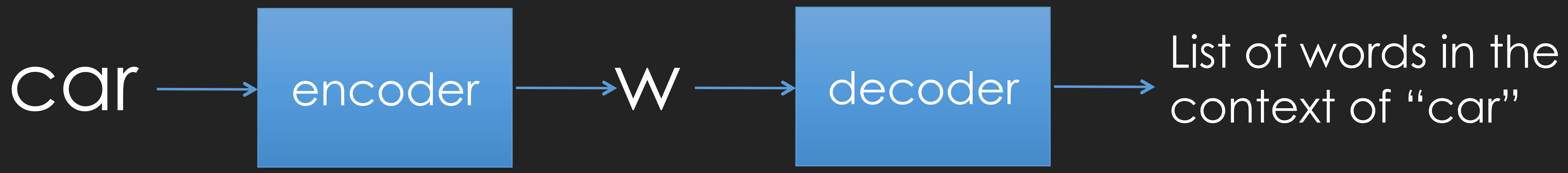
I parked the car in a nearby street. It is a red car with two doors, ...

I parked the vehicle in a nearby street...

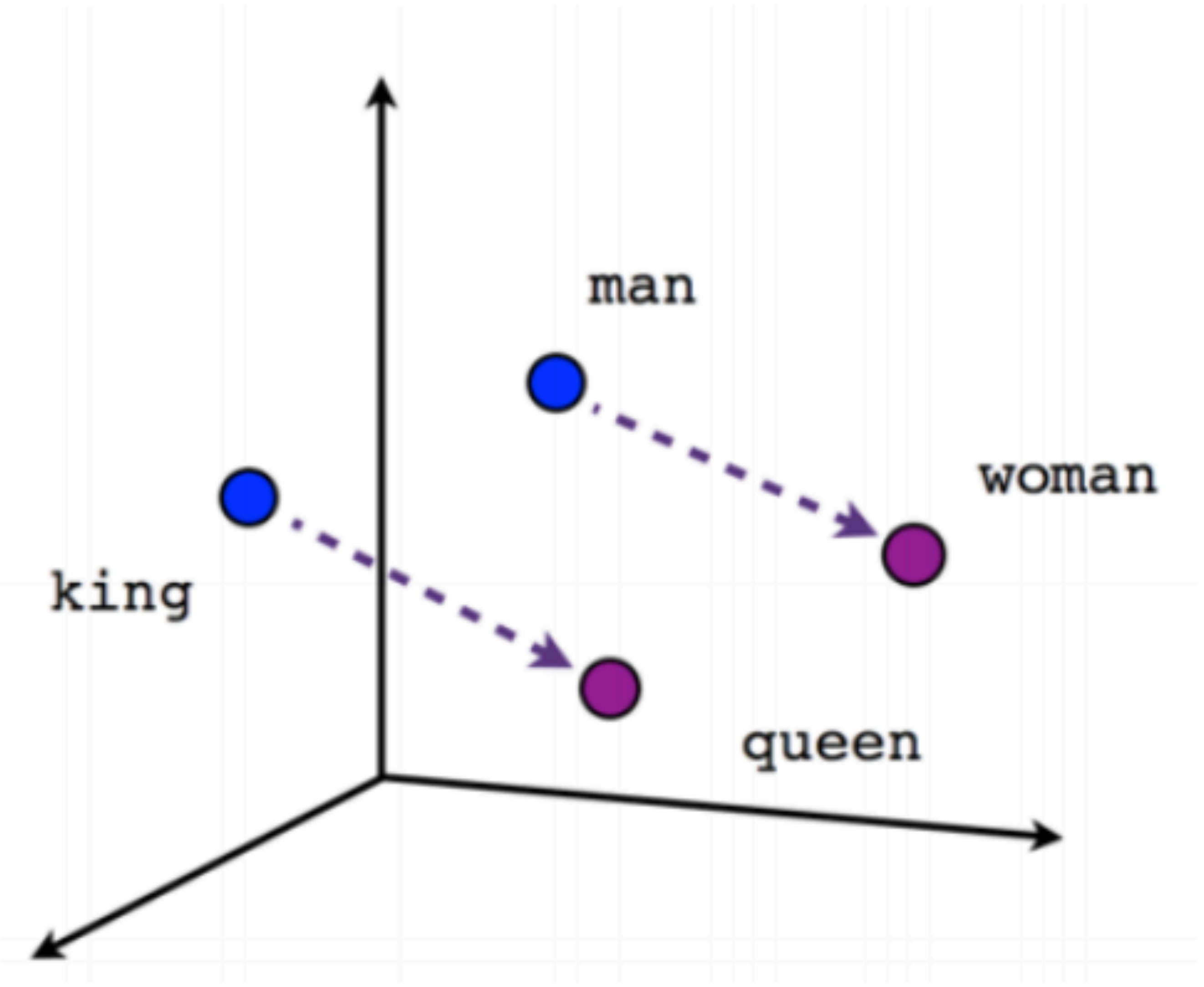
T. Mikolov, K. Chen, G. Corrado, J. Dean. Efficient Estimation of Word Representations in Vector Space. arXiv:1301.3781, 2013

word2vec

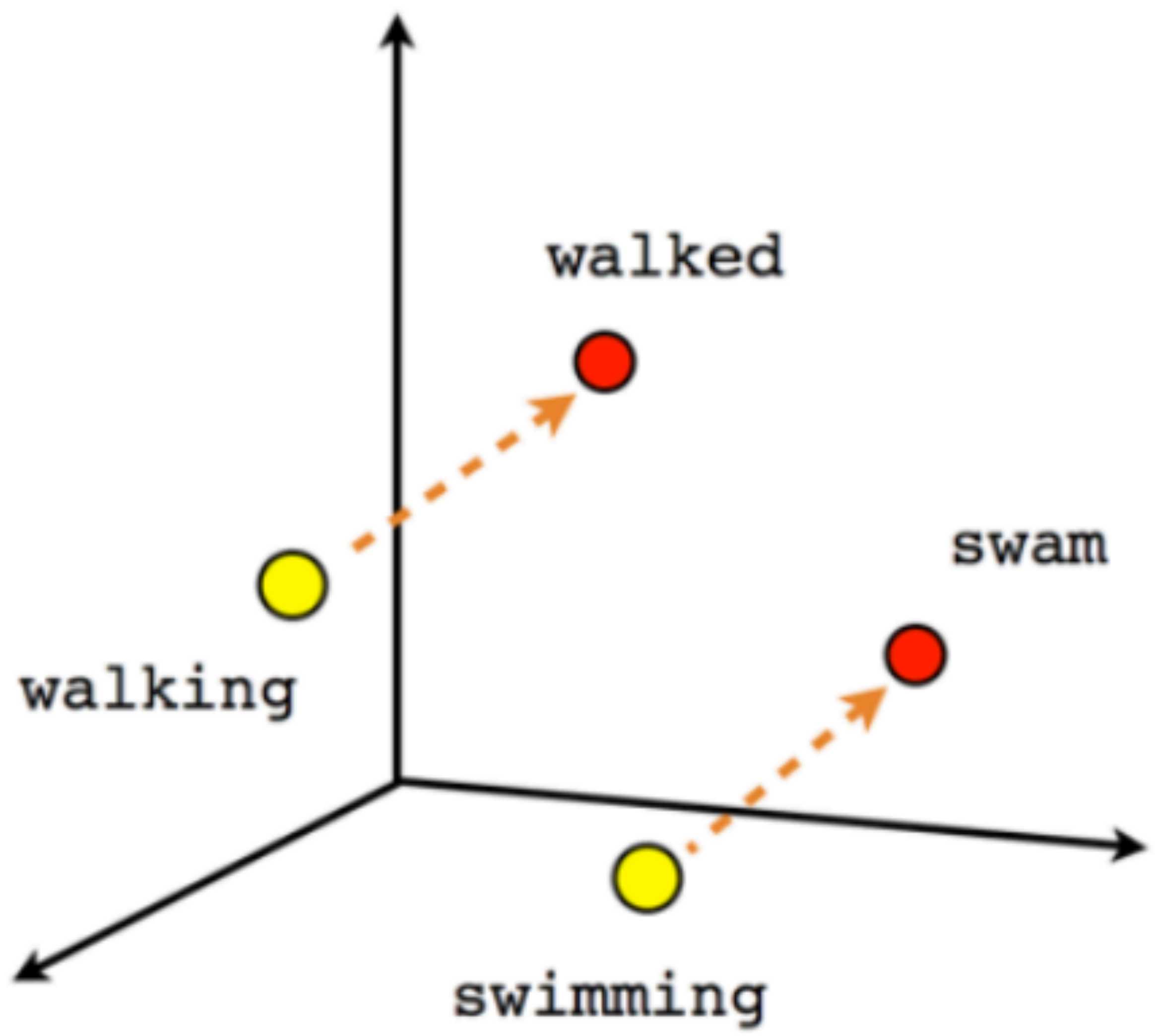
I parked the car in a nearby street. It is a red car with two doors, …



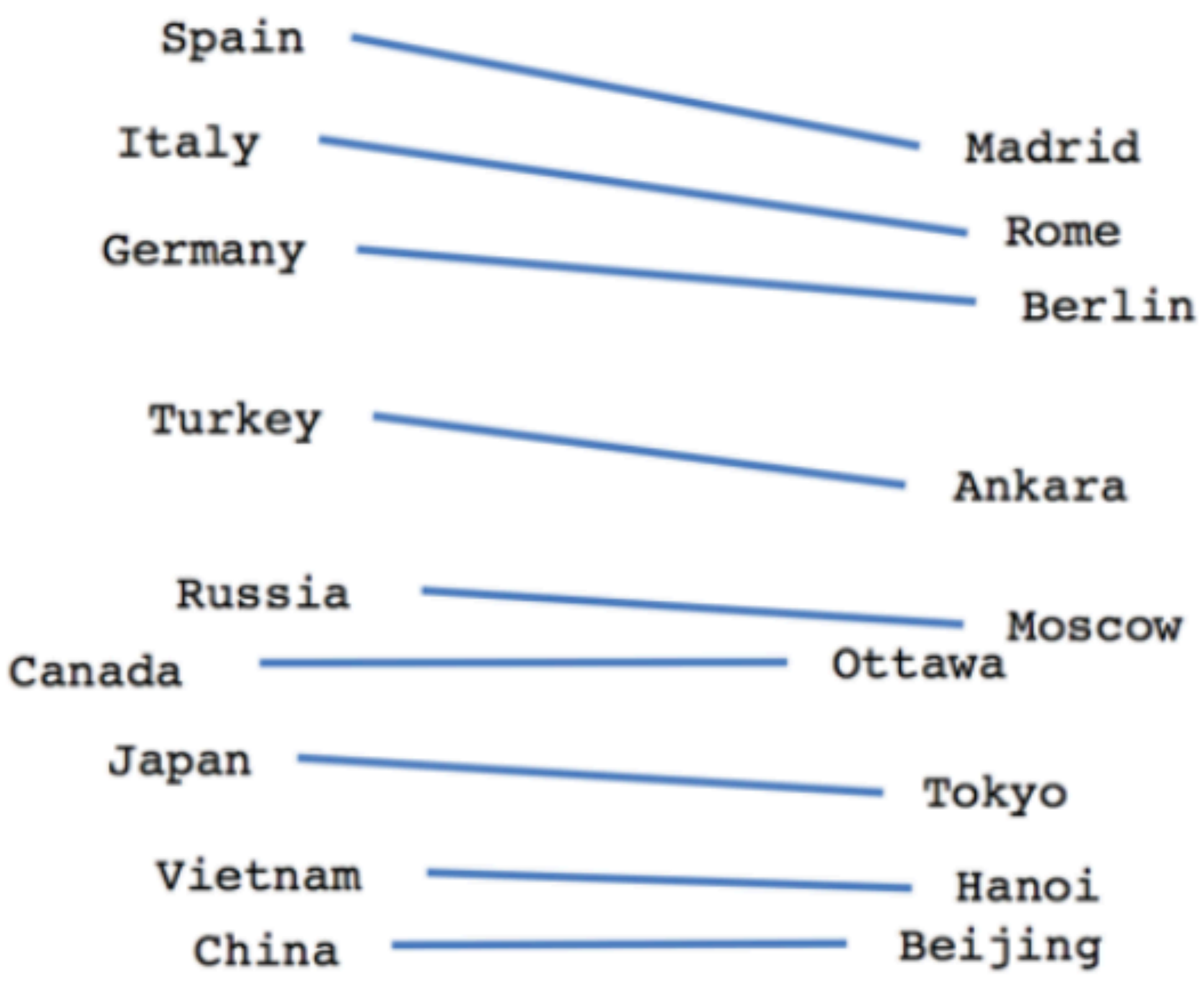
T. Mikolov, K. Chen, G. Corrado, J. Dean. Efficient Estimation of Word Representations in Vector Space. arXiv:1301.3781, 2013



Male-Female



Verb tense



Country-Capital

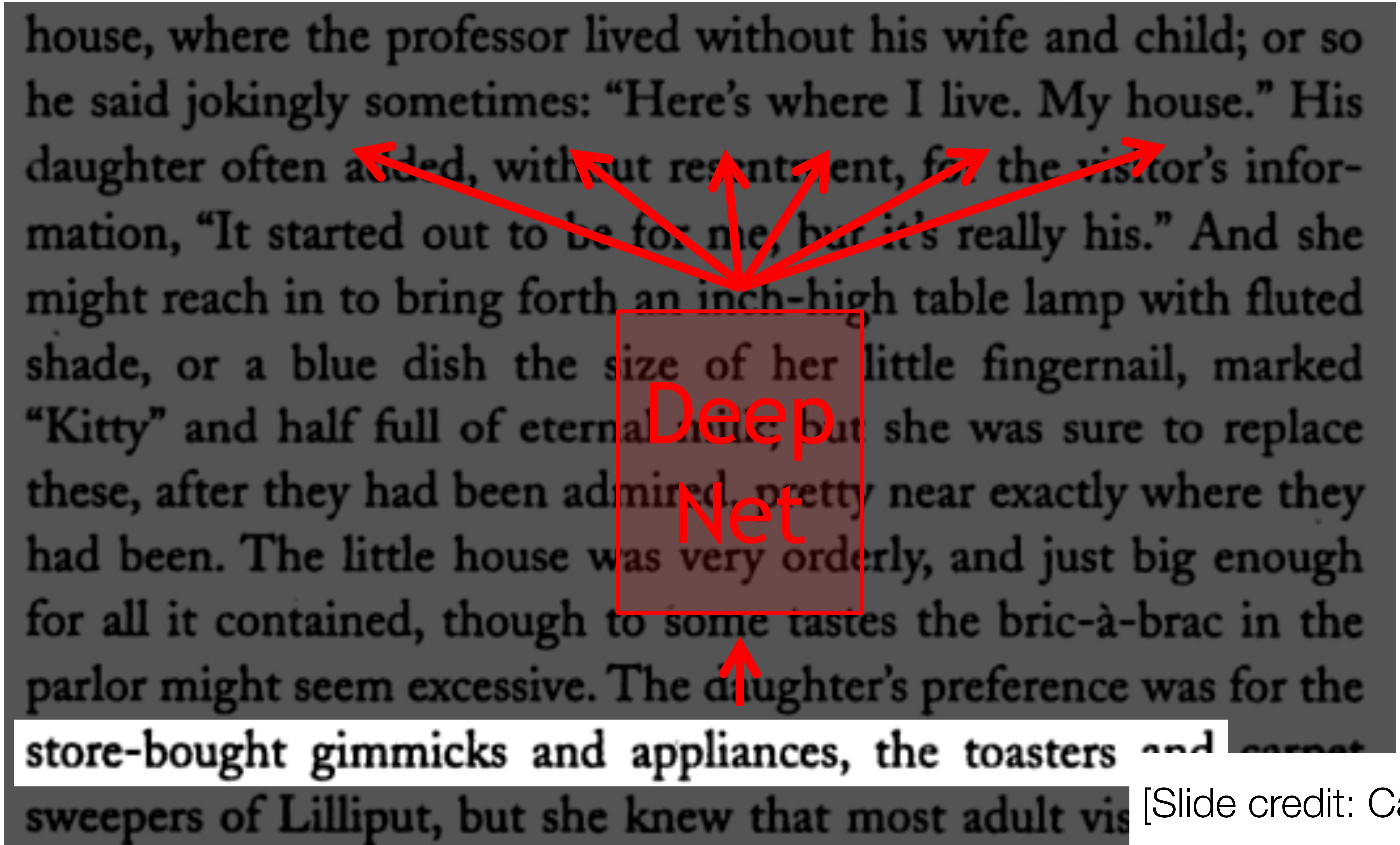
Examples from https://www.tensorflow.org/tutorials/representation/word2vec

Unsupervised visual representation learning by context prediction

[Carl Doersch, Abhinav Gupta, Alexei A. Efros, ICCV 2015]

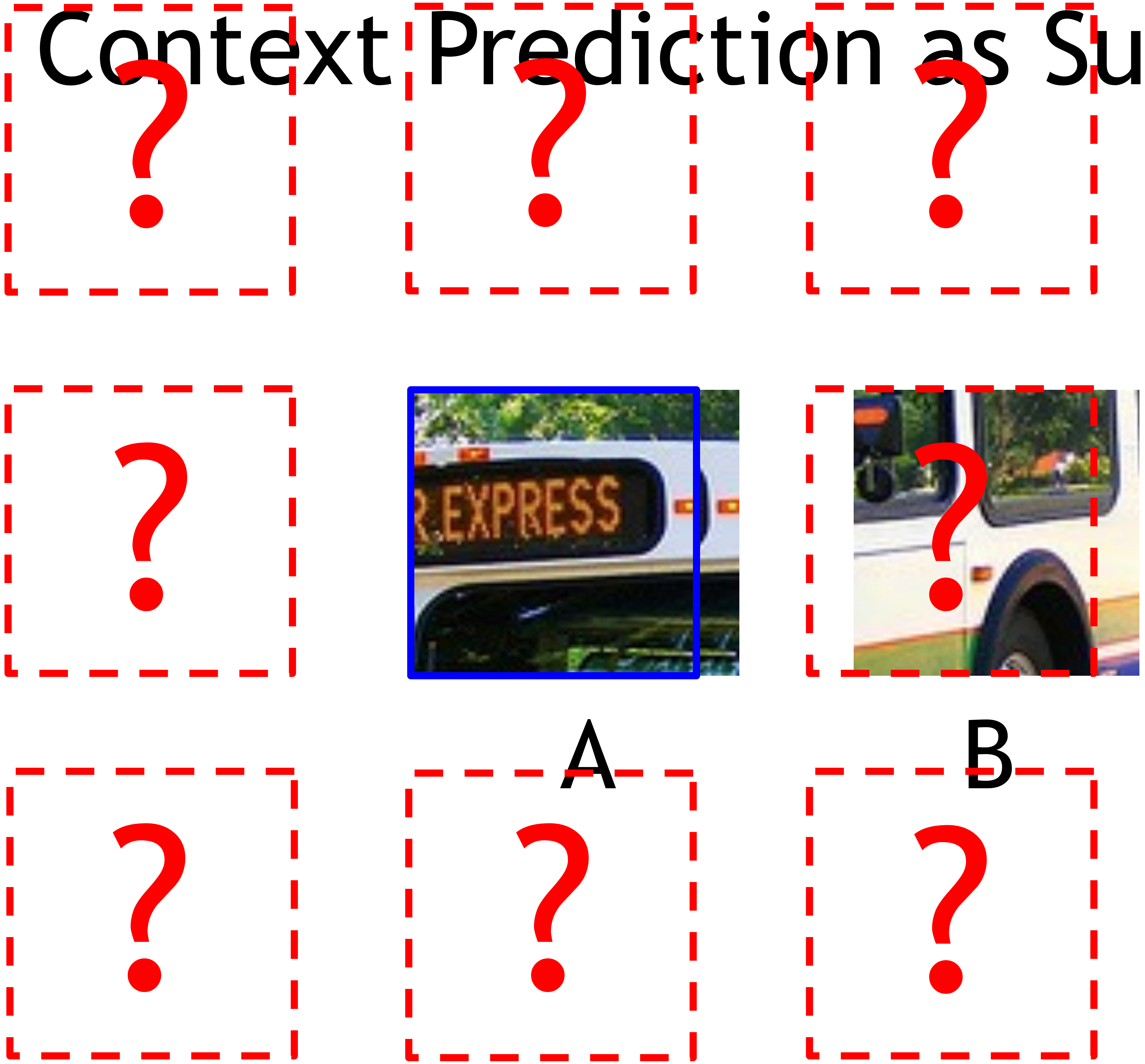
Context as Supervision

[Collobert & Weston 2008; Mikolov et al. 2013]



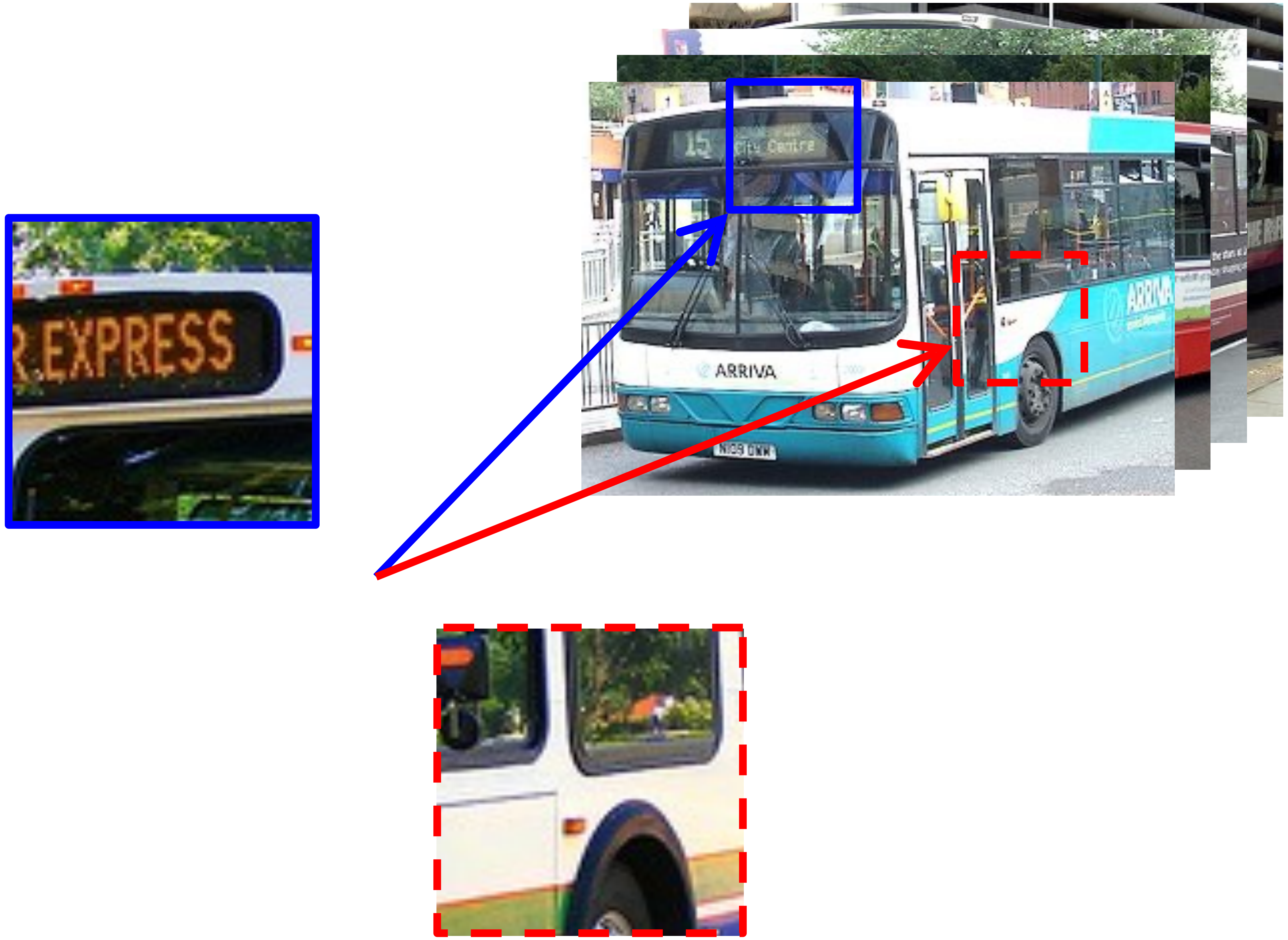
[Slide credit: Carl Doersch]

Context Prediction as Supervision



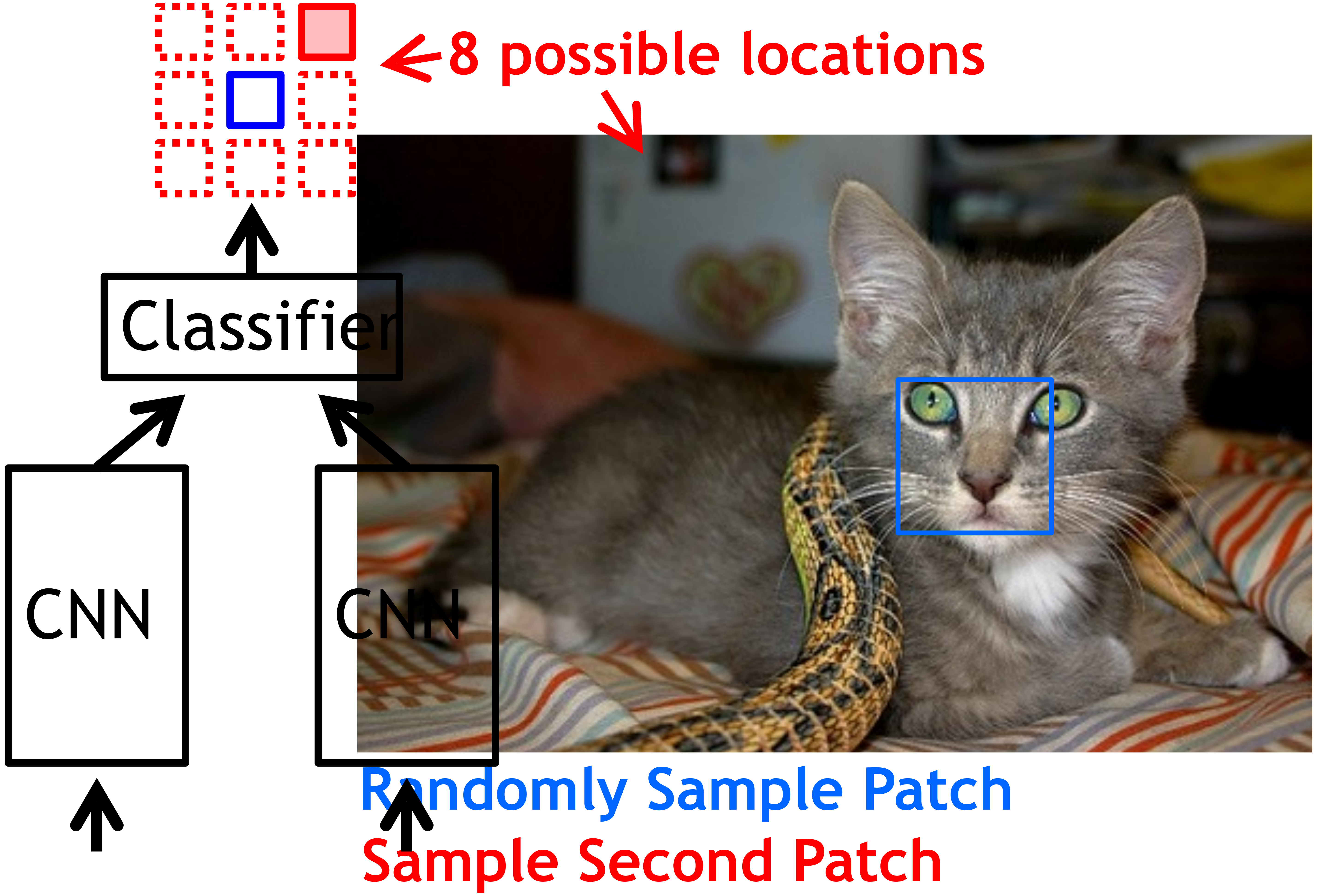
[Slide credit: Carl Doersch]

Semantics from a non-semantic task

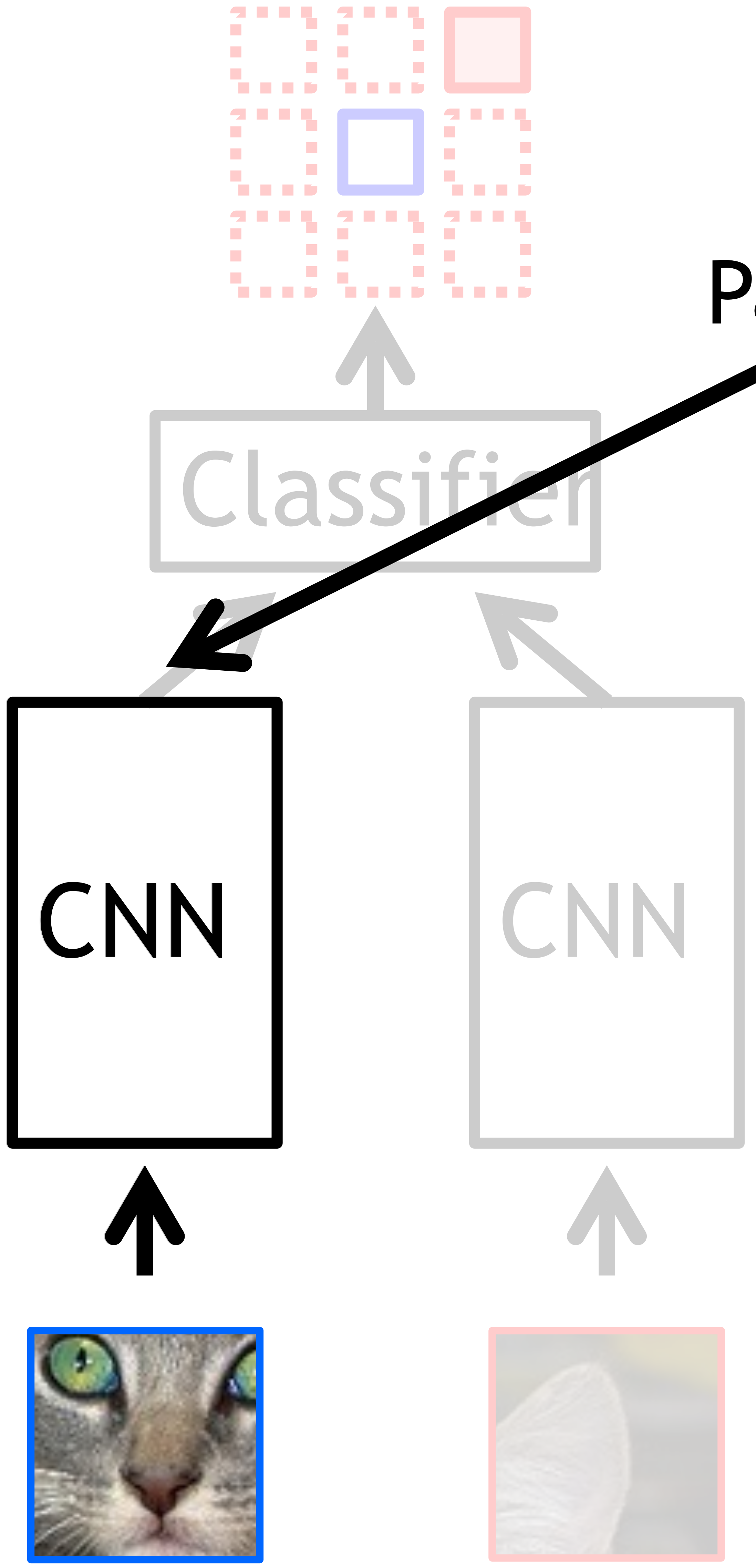


[Slide credit: Carl Doersch]

Relative Position Task



[Slide credit: Carl Doersch]



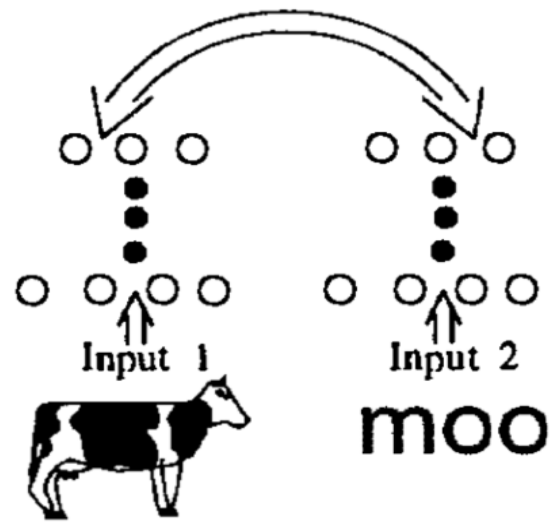
Patch Embedding (representation)

Input Nearest Neighbors

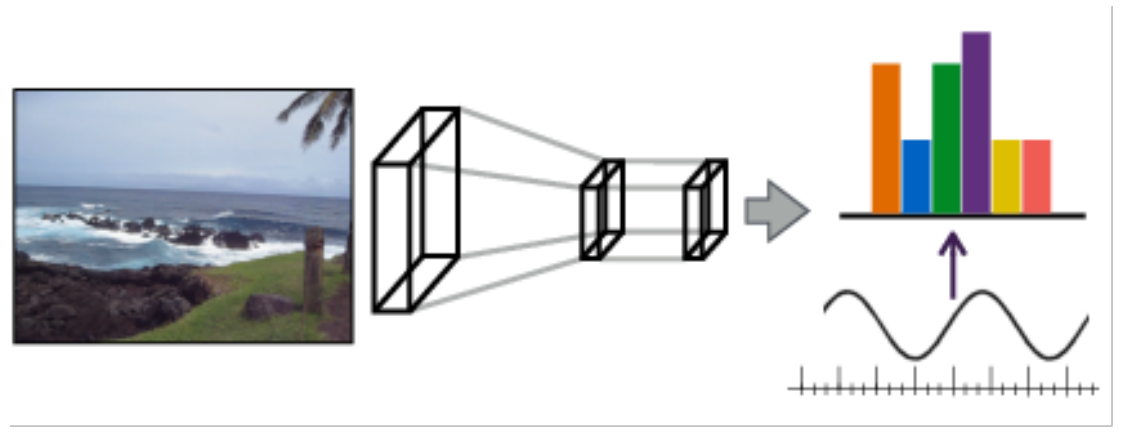
Note: connects ***across*** instances!

[Slide credit: Carl Doersch]

Audio

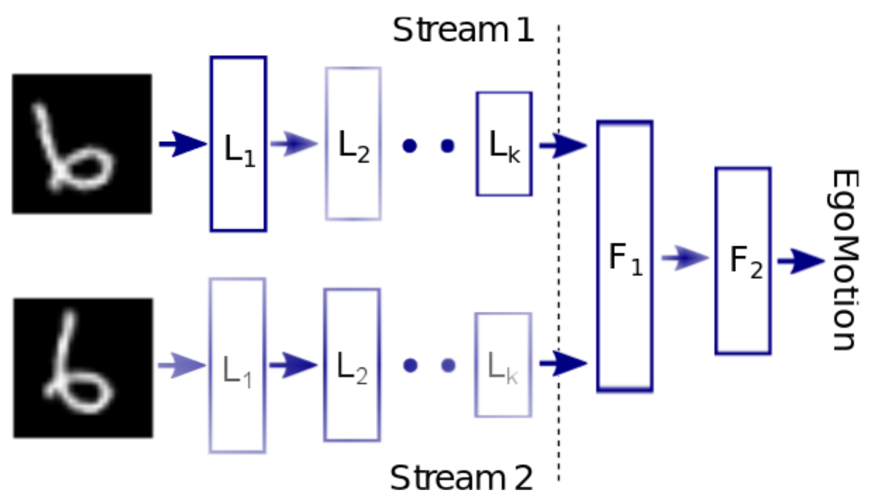


de Sa. NIPS 1994.



Owens et al. ECCV 2016.

Egomotion

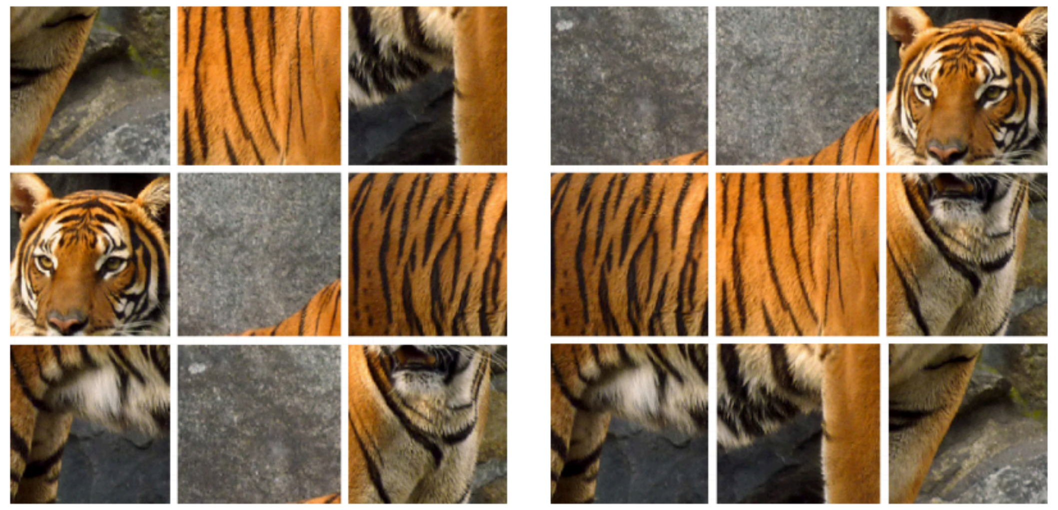


Agrawal et al. ICCV 2015.

Jayaraman et al. ICCV 2015.

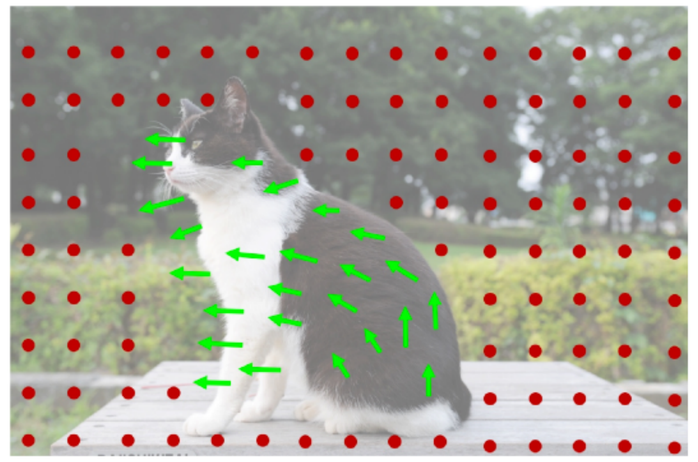
Context

Pathak et al. CVPR 2016.



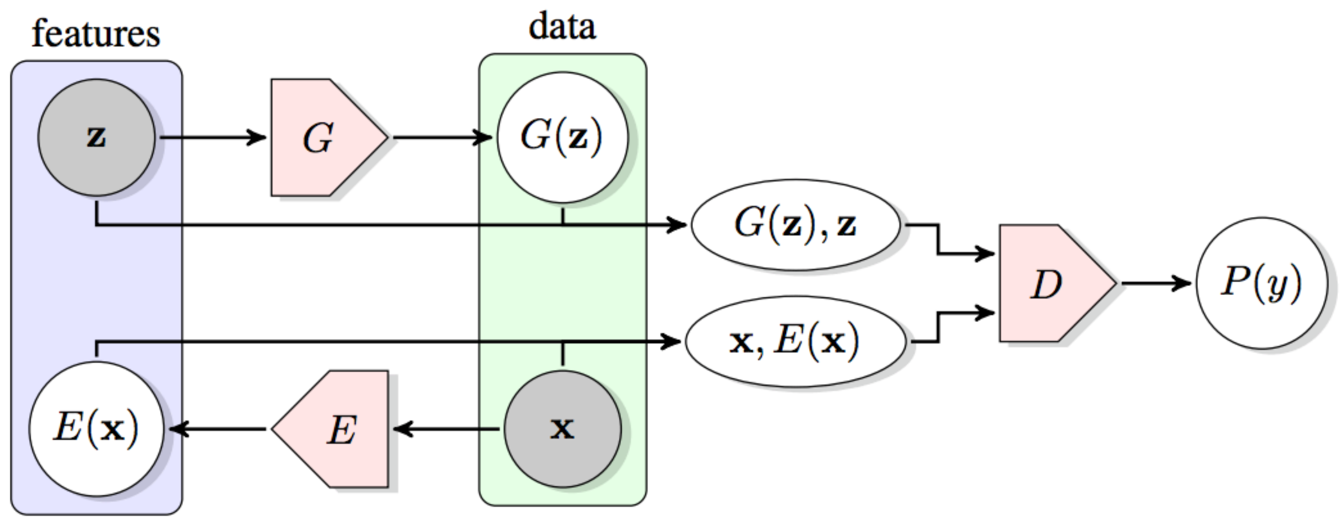
Noroozi and Favaro. ECCV 2016.
Doersch et al. ICCV 2015.

Video



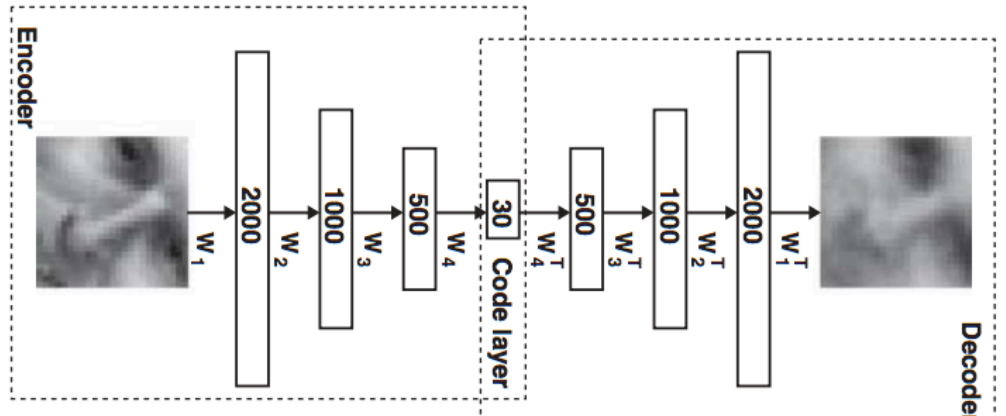
Wang et al. ICCV 2015. Pathak et al. CVPR 2017.
Misra et al. ECCV 2016.

Generative Modeling



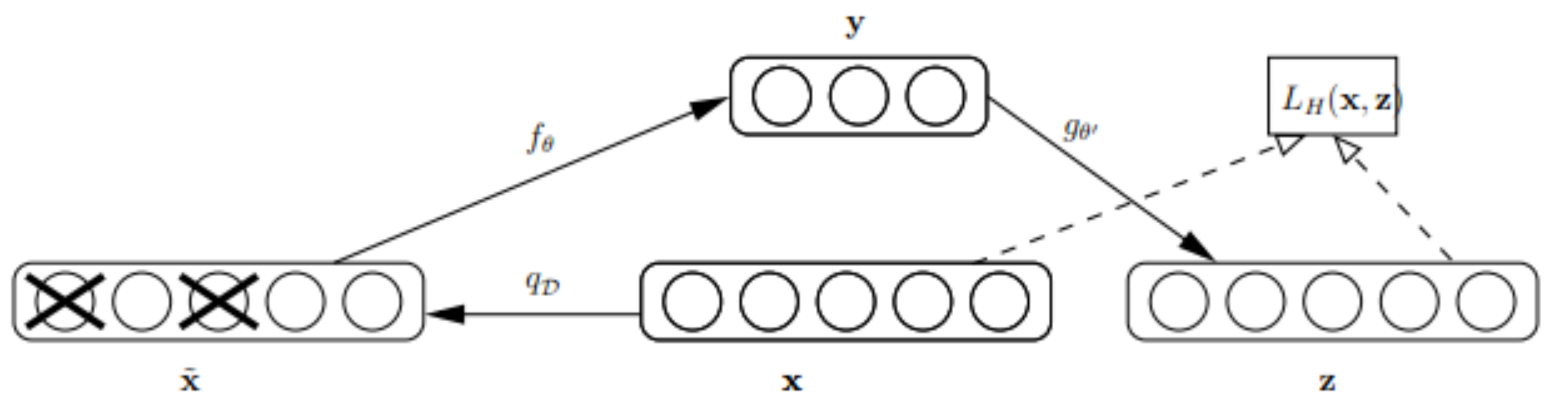
Donahue et al. Dumoulin et al. ICLR 2017.

Autoencoders



Hinton & Salakhutdinov. Science 2006.

Denoising Autoencoders



Vincent et al. ICML 2008.

Goal: Set up a pre-training scheme to induce a "useful" representation

[Slide credit: Richard Zhang]

How Much Information is the Machine Given during Learning?

- **"Pure" Reinforcement Learning (cherry)**
  - The machine predicts a scalar reward given once in a while.
  - **A few bits for some samples**
- **Supervised Learning (icing)**
  - The machine predicts a category or a few numbers for each input
  - Predicting human-supplied data
  - **10→10,000 bits per sample**
- **Self-Supervised Learning (cake génoise)**
  - The machine predicts any part of its input for any observed part.
  - Predicts future frames in videos
  - **Millions of bits per sample**

[Slide Credit: Yann LeCun]

Summary

1. Deep nets learn *representations*, just like our brains do

2. This is useful because representations transfer — they act as prior knowledge that enables quick learning on new tasks

3. Representations can also be learned without labels, which is great since labels are expensive and limiting

4. Without labels there are many ways to learn representations. We saw:

    1. representations as compressed codes

    2. representations that are shared across sensory modalities

    3. representations that are predictive of their context