Universal, Unsupervised, and Understandable Image Representations

Dr Andrea Vedaldi

Medical Imaging Summer School

Favignana, August 2018

How can AI impact people?

Horizontal problems

Autonomous driving

Internet search

Social networks

Medical imaging (?)

Vertical problems

Measuring plants

Matching manuscripts

Matching galaxies

Counting penguins

Recognizing flowers

Measuring condensation

Tracking crystals

Measuring astrolabes

Searching greek vases

Comparing 19th century paintings

Matching Mesopotamian clay rolls

...

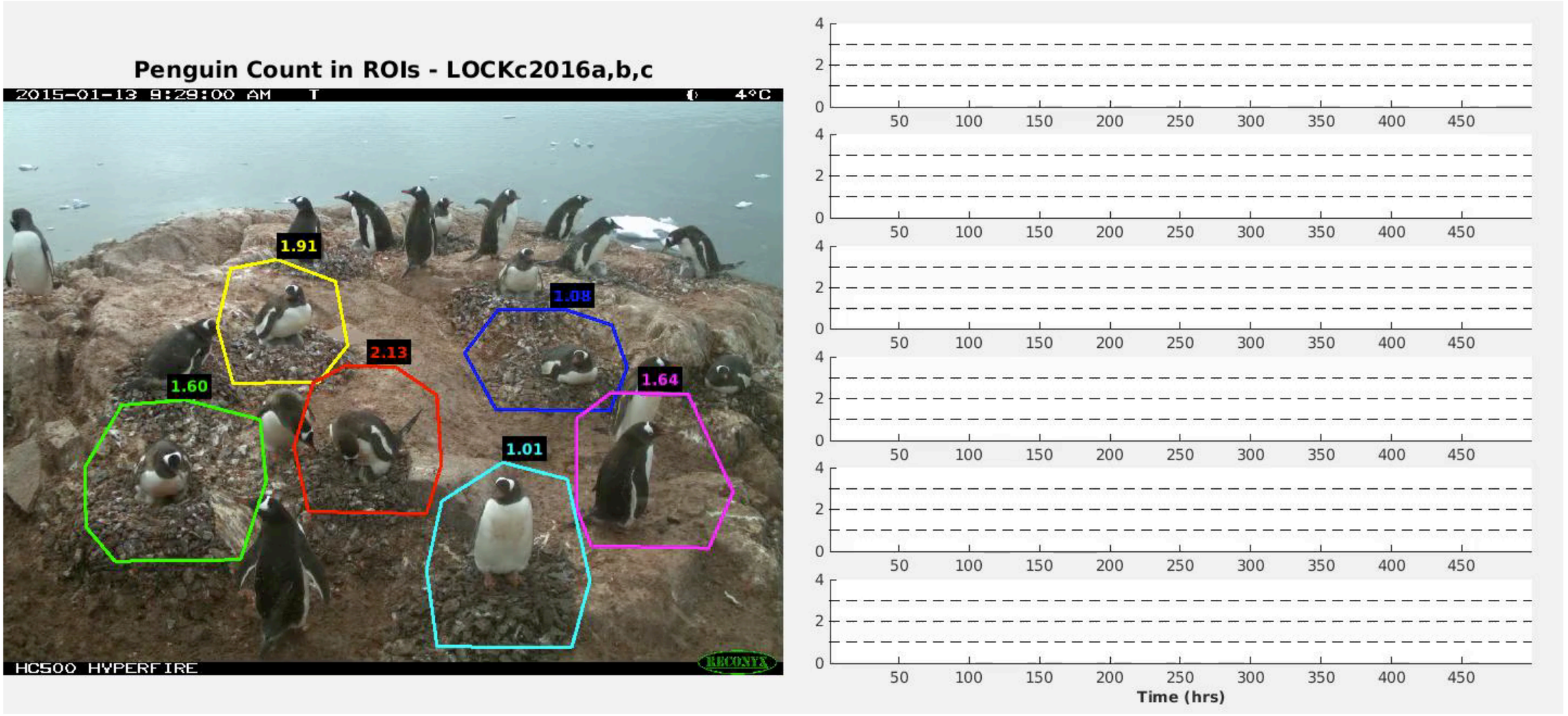
Vertical AI problems

Archeology, bibliography, art historians, etc.

Image matching, search, comparison, recognition

Vertical AI problems

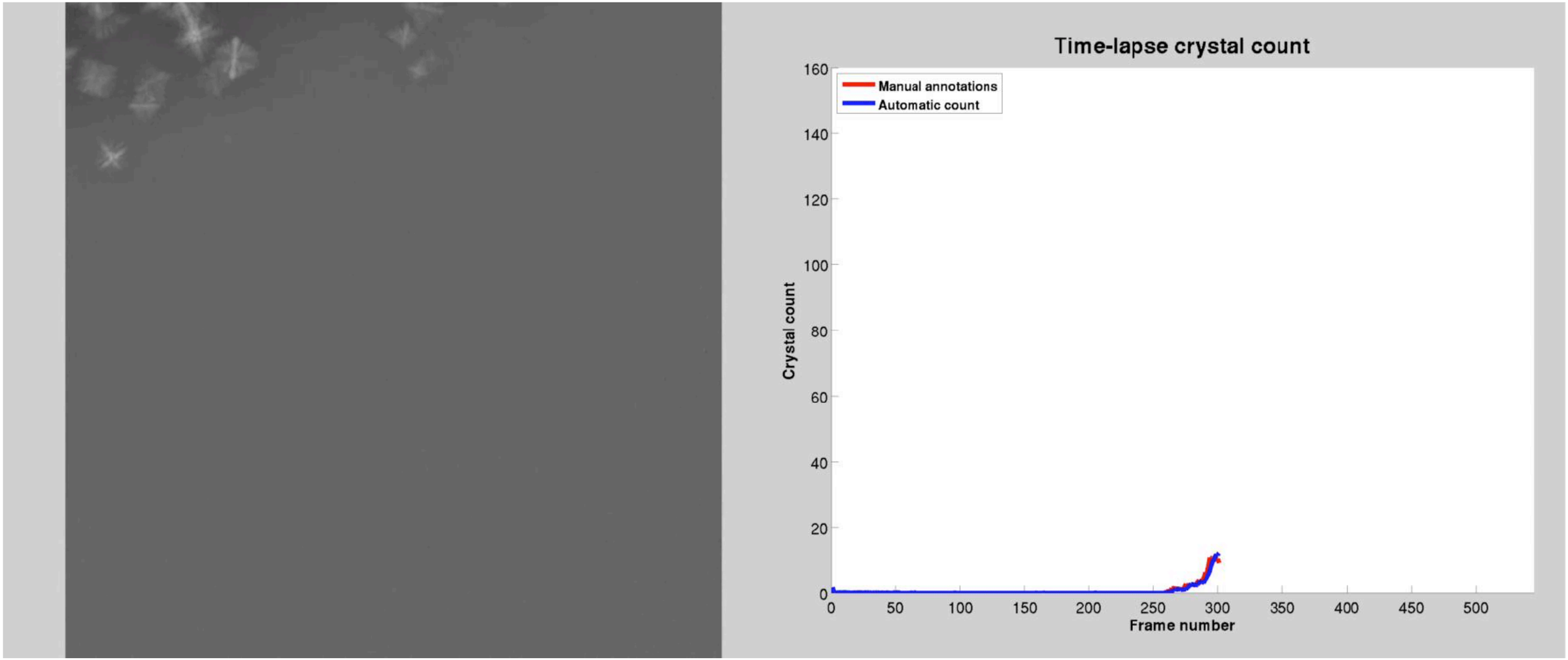
Biology, zoology, medical imaging, material science, etc.



Counting penguins

Vertical AI problems

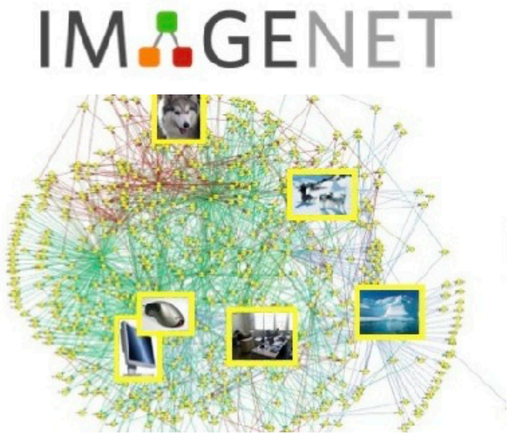
Biology, zoology, areas of medical imaging, material science, etc.



Counting crystals in thousands of videos

The need for (too) big data

One of the most significant bottlenecks of deep learning



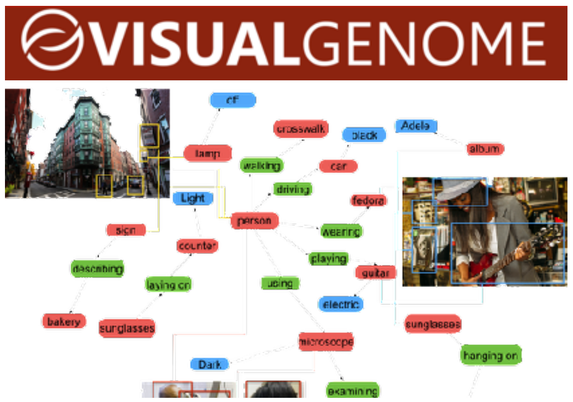
ImageNet ILSVRC

1.2M images

ADE 20K

20K images

Dense annotations



Visual Genome

100K images

4M annotations

Open Images

9M images

28M annotations

ImageNet: A Large-Scale Hierarchical Image Database. Deng, Dong, Socher, Li, Li, Fei-Fei. CVPR, 2009.
Scene parsing through ADE20K dataset. Zhou, Zhao, Puig, Fidler, Barriuso, Torralba. CVPR, 2017.
Visual genome: Connecting language and vision using crowdsourced dense image annotations. Krishna et al., 2016.
OpenImages: A public dataset for large-scale multi-label and multi-class image classification. Krasin et al.. 2017

Will he label 1M images?

Three key challenges of deep learning

Universal Representations

Fewer models to train

Unsupervised Representations

Less effort to train new models

Understandable Representations

Trust, safety, and usability

Three key challenges of deep learning

Universal Representations

Fewer models to train

Unsupervised Representations

Less effort to train new models

Understandable Representations

Trust, safety, and usability

Narrow representations

Object recognition

w_1

FC1

Medical diagnosis

w_2

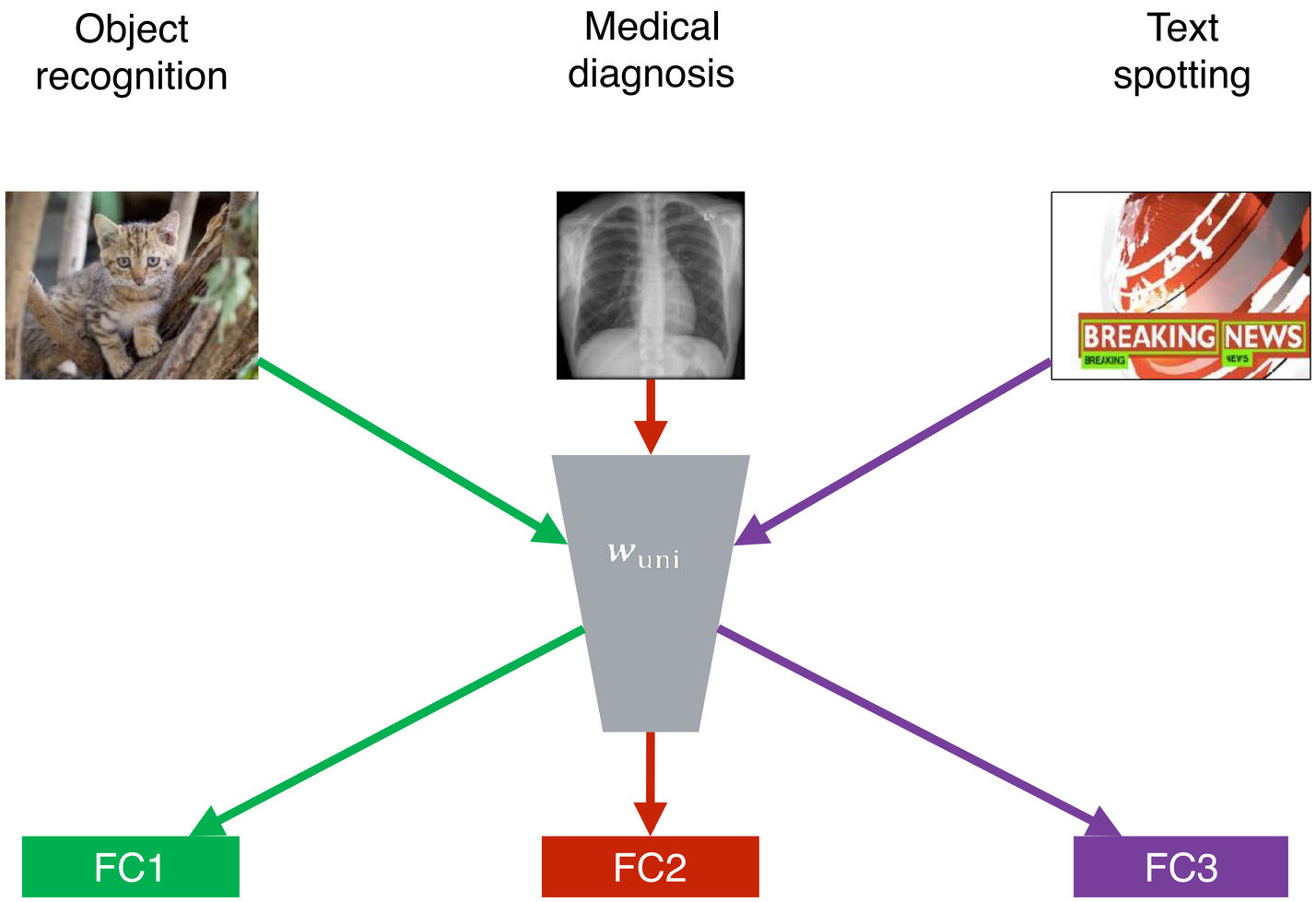
FC2

Text spotting

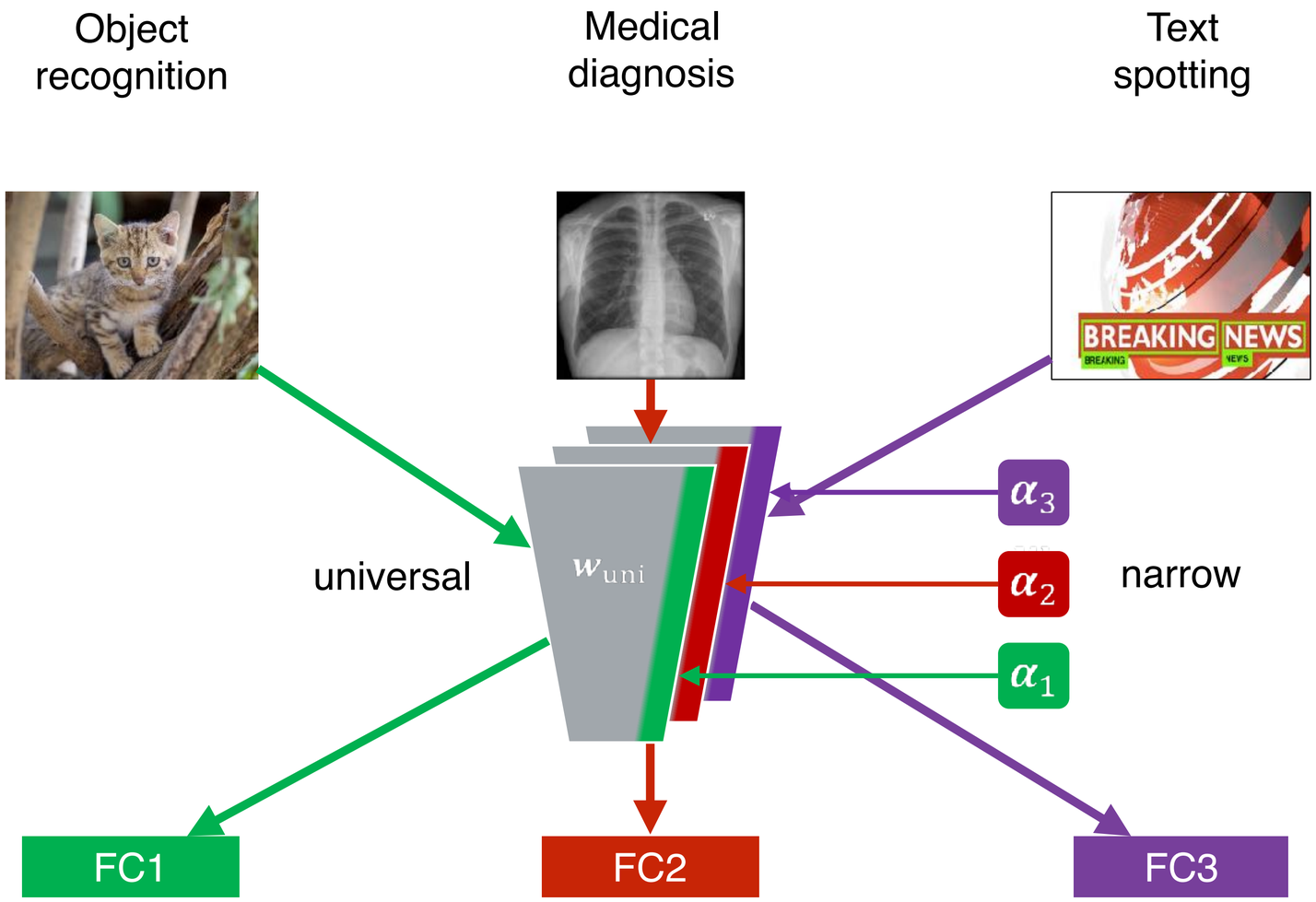
w_3

FC3

Universal representation

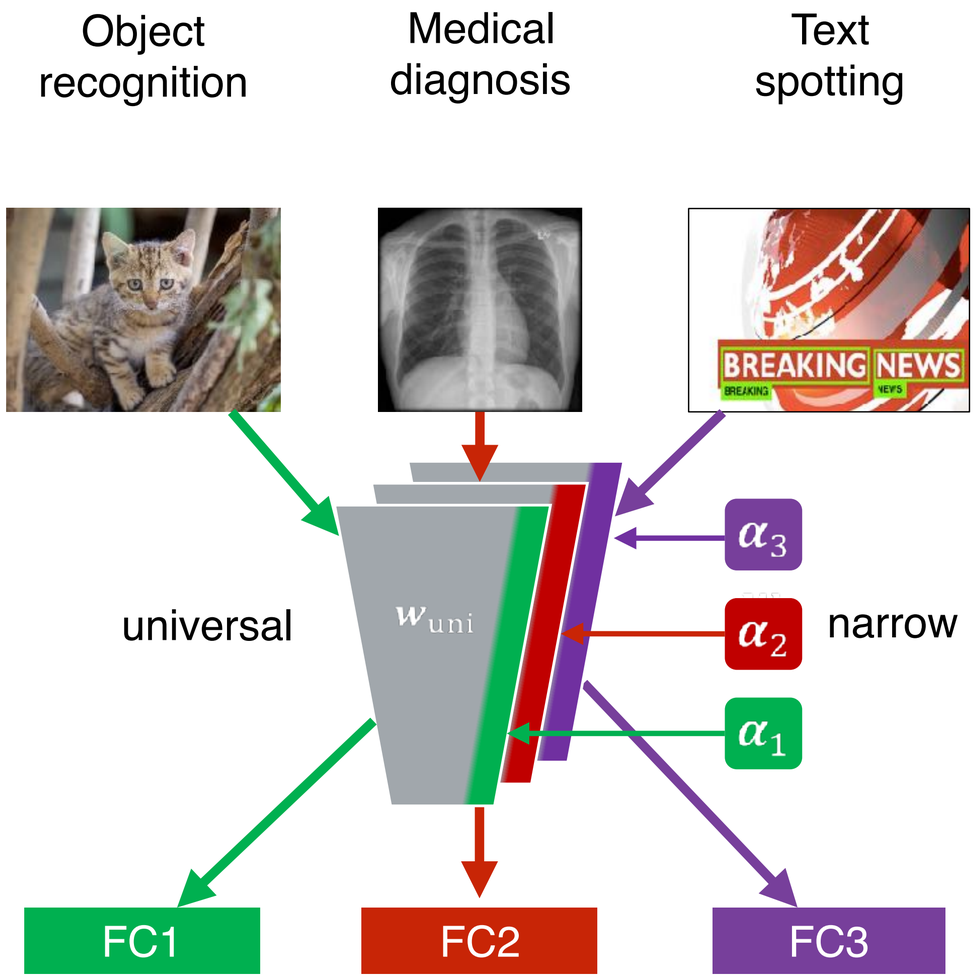


Universal representation family



Learning multiple visual domains with residual adapters, Rebuffi, Bilen, Vedaldi, NIPS 2017

Universal representation families



Preview

- > 90% of parameters are **shared**
- **Same** or **better performance** than narrow models
- **No forgetting**

Applications

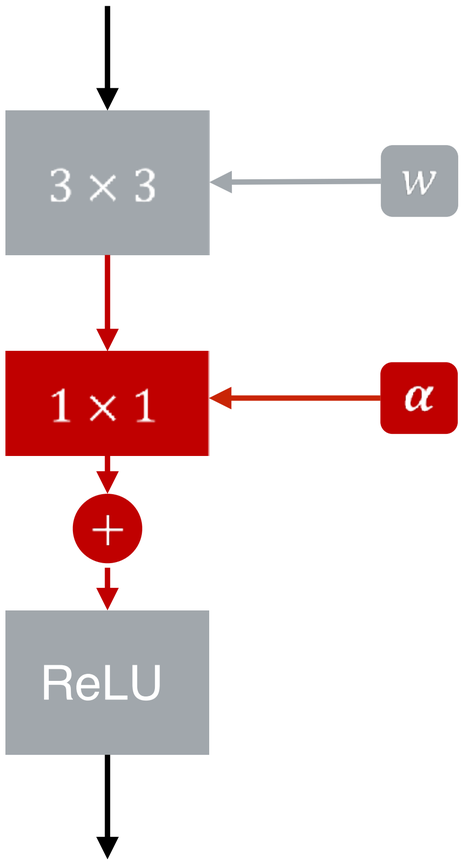
- Better than standard pre-training, especially for small dataset
- Efficient model storage, transmission, updating

Learning multiple visual domains with residual adapters. Rebuffi, Bilen, Vedaldi. NIPS, 2017.

Residual adapters

An efficient parametrization

typical building block



Adapters

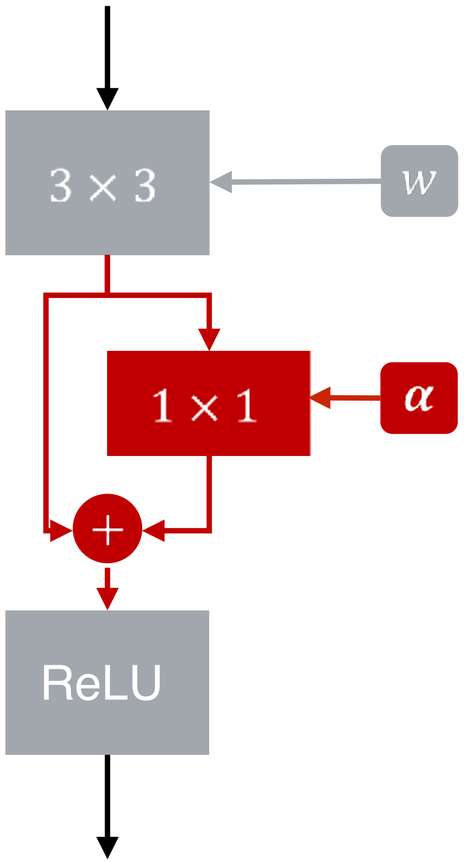
- Tweak a fixed neural network block
- Interleaved with standard convolutions
- 1 x 1 filter bank
- Only ~10% of the parameters

Learning multiple visual domains with residual adapters. Rebuffi, Bilen, Vedaldi. NIPS, 2017.

Residual adapters

An efficient parametrization

typical building block

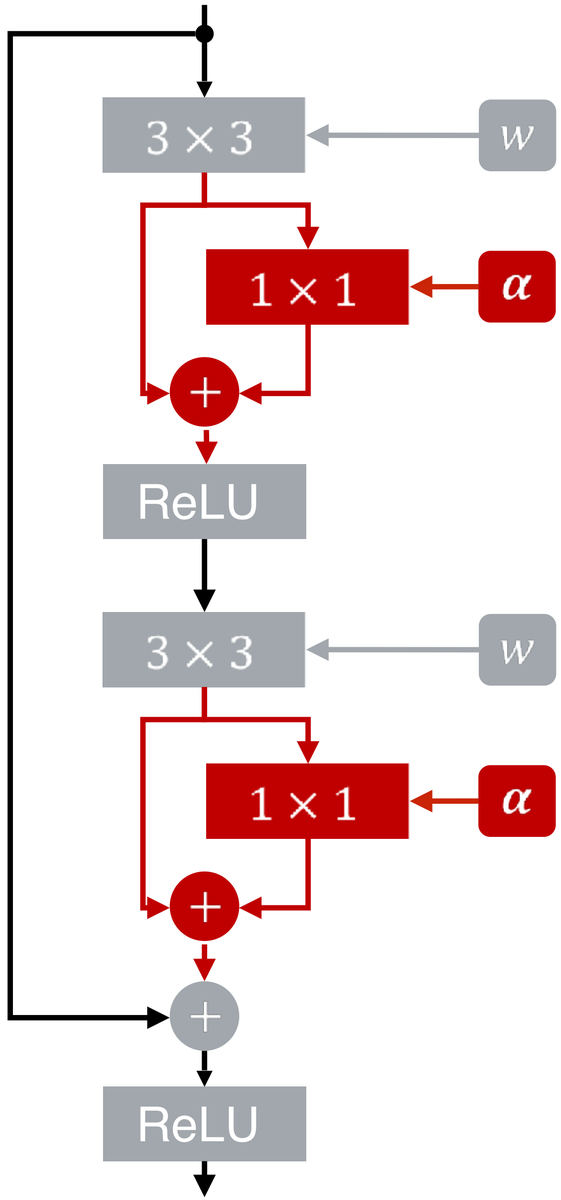


Residual Adapters

- Shrinkage controls the amount of adaptation
- Better generalization when adapting to small target datasets

Learning multiple visual domains with residual adapters. Rebuffi, Bilen, Vedaldi. NIPS, 2017.

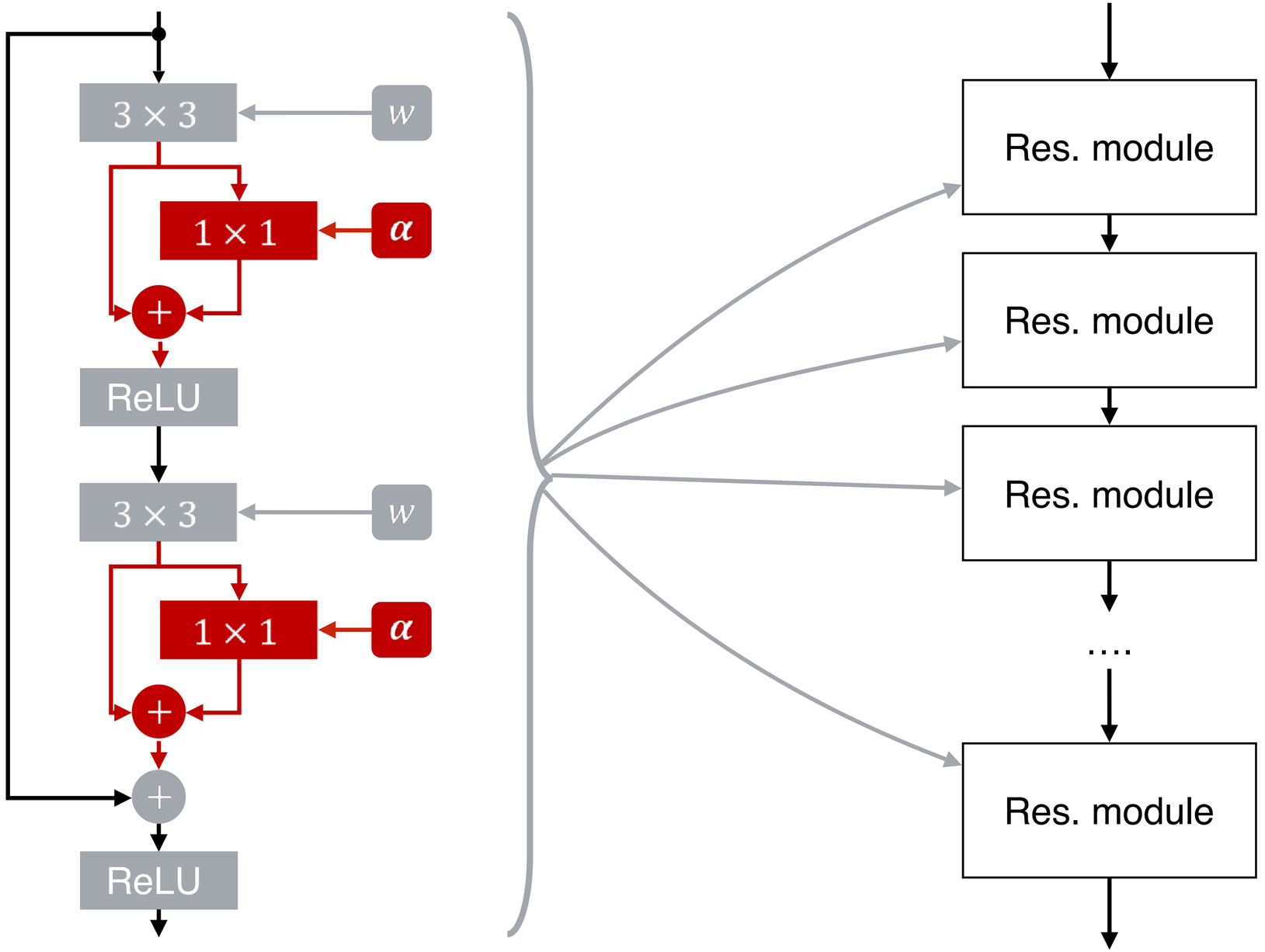
Adapters: plug & play



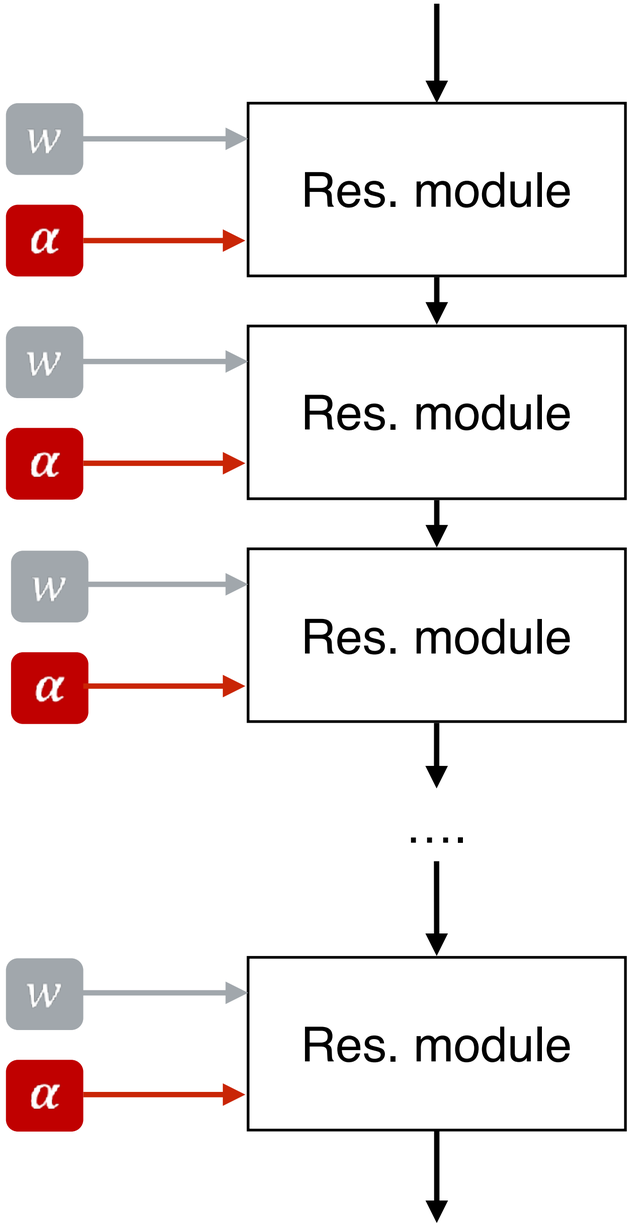
Residual Adapters

- Can be added to any off-the-shelf network
- Example: adapted ResNet [He et al. 2016]

Head-to-foot adaptation



Head-to-foot adaptation



Dynamic adaptation parameters

LearNet

w

α

Res. module

w

α

Res. module

w

α

Res. module

....

w

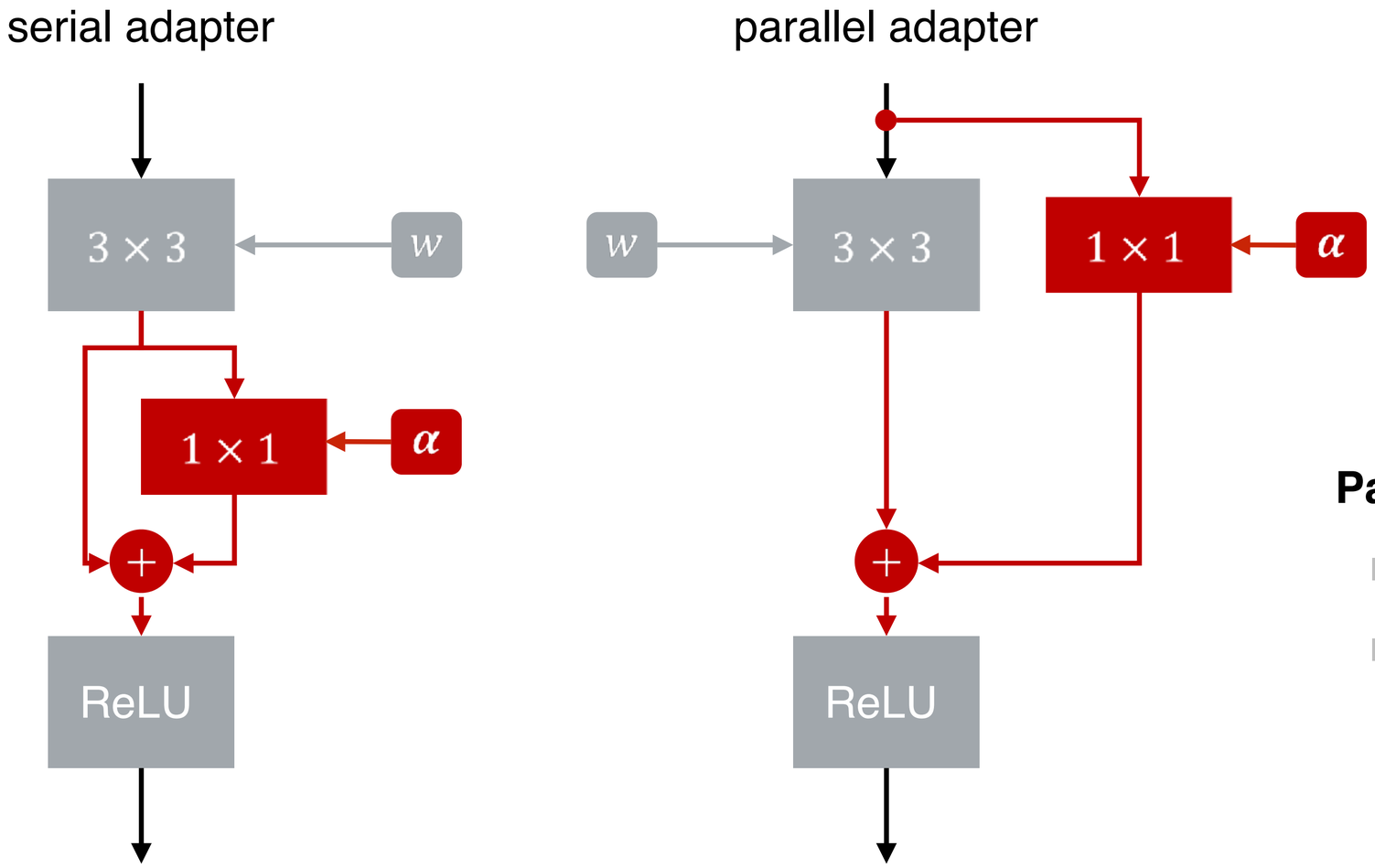
α

Res. module

Also known as: dynamic weights, metanets, hypernets [Schmidhuber 92, Bromley et al. 93, Yang et al. 2015, Noh et al. 2016, **Bertinetto et al. 2016**, Ha et al. 2016]

Parallel residual adapters

A better variant



Parallel Residual Adapters

- Completely plug-and-play
- Obtains better results than standard fine-tuning

Efficient parametrization of multi-domain deep neural networks, Rebuffi, Bilen, Vedaldi, CVPR 2018

Visual Domain Decathlon Challenge

A new benchmark

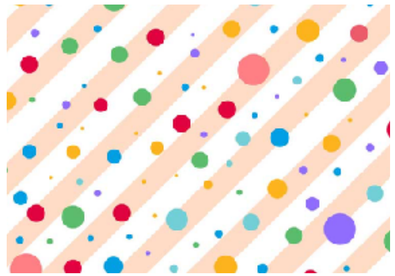
Goal: learn a single model that performs as well as possible on 10 very different visual domains

Aircrafts

CIFAR-100

Daimler Pedestrians

Describable Textures



German Street Signs

ImageNet

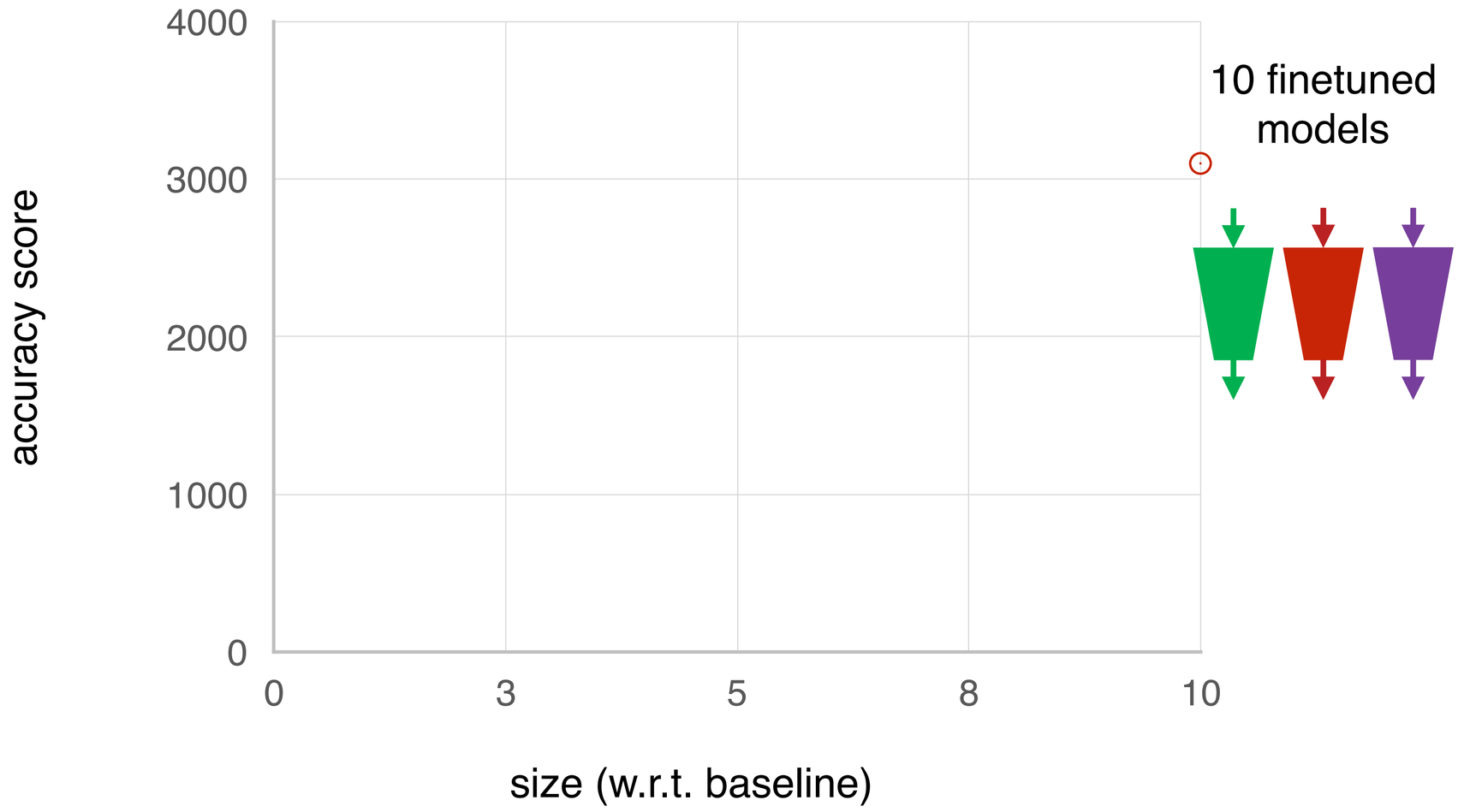
VGG Flowers

Omniglot

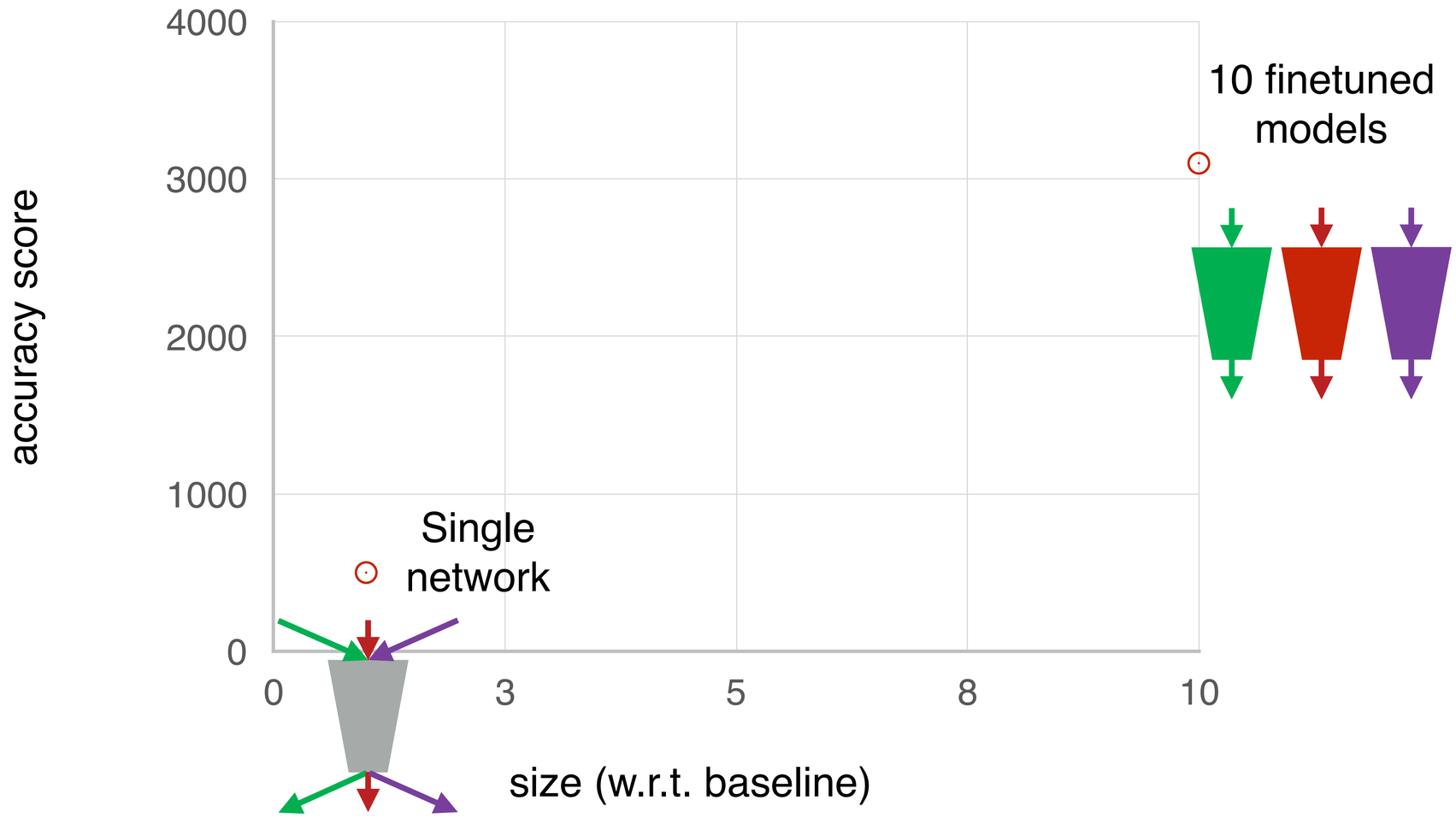
SVHN

UCF101 Dyn. Images

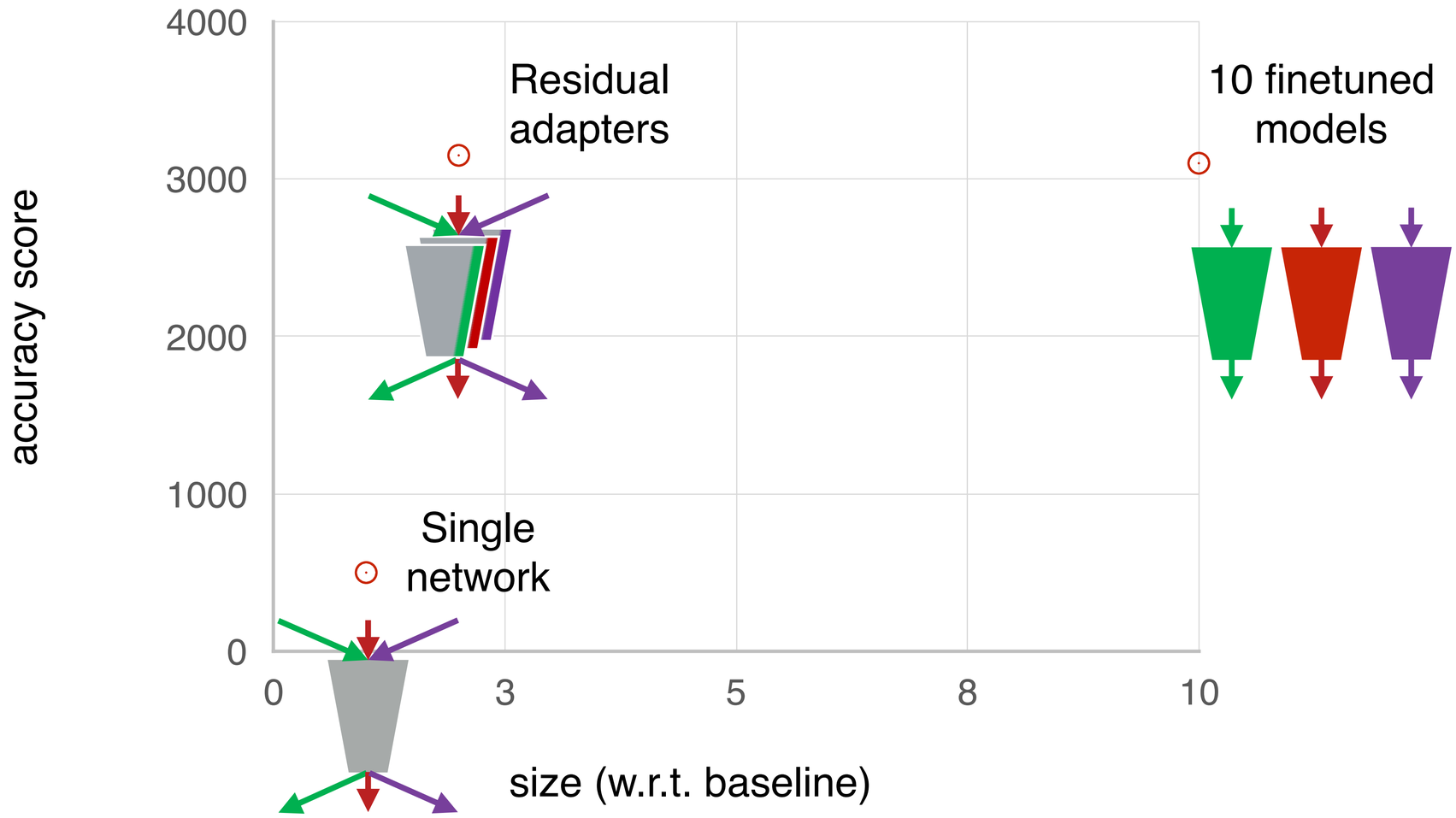
Visual Decathlon results



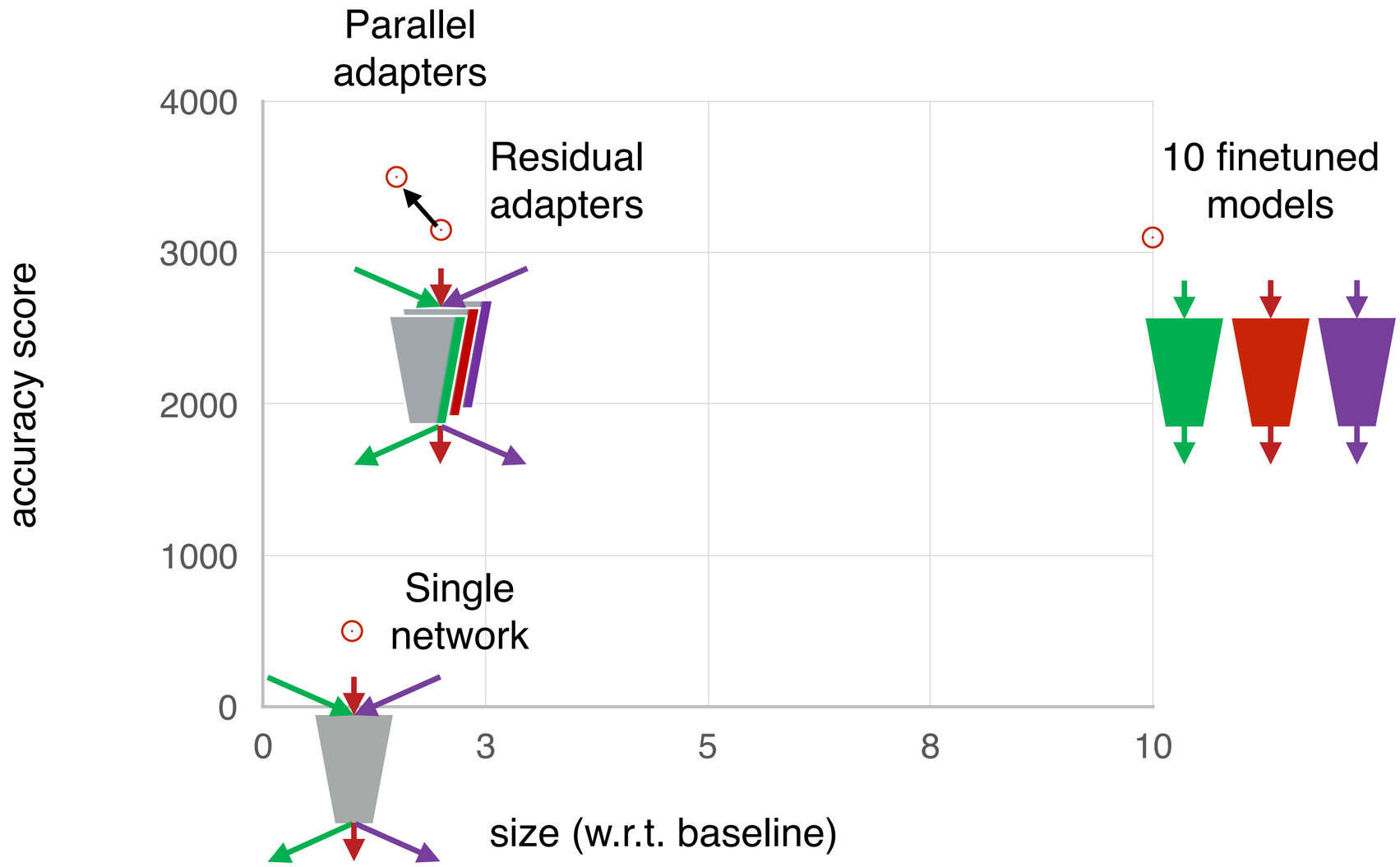
Visual Decathlon results



Visual Decathlon results



Visual Decathlon results

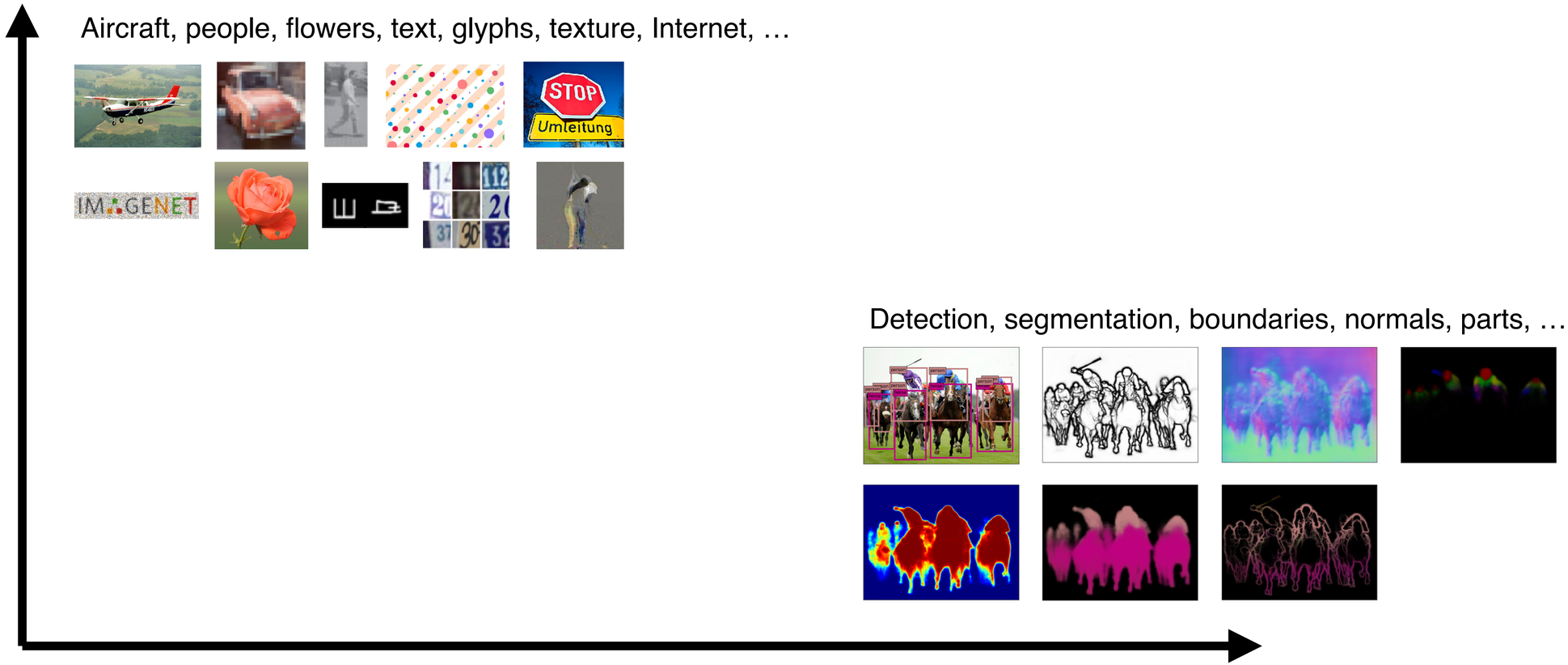


Efficient parametrization of multi-domain deep neural networks, Rebuffi, Bilen, Vedaldi, CVPR 2018

Universal Families *vs* UberNets

Complementary problems

Universal family: more domains



UberNet: Training a universal convolutional neural network for low-, mid-, and high-level vision using diverse datasets and limited memory. Kokkinos. CVPR, 2017

Three key challenges of deep learning

Universal Representations

Fewer models to train

Unsupervised Representations

Less effort to train new models

Understandable Representations

Trust, safety, and usability

Can we drop annotations?

Images are cheap and abundant

However, manual annotations are extremely expensive

Scene parsing through ADE20K dataset. Zhou, Zhao, Puig, Fidler, Barriuso, Torralba. CVPR, 2017.

Self-supervised learning

Can we drop annotations?

Images are cheap and abundant

Self-supervison

Extract a supervisory signal from the raw data alone

Scene parsing through ADE20K dataset. Zhou, Zhao, Puig, Fidler, Barriuso, Torralba. CVPR, 2017.

Self-supervised features

Self-supervised structure

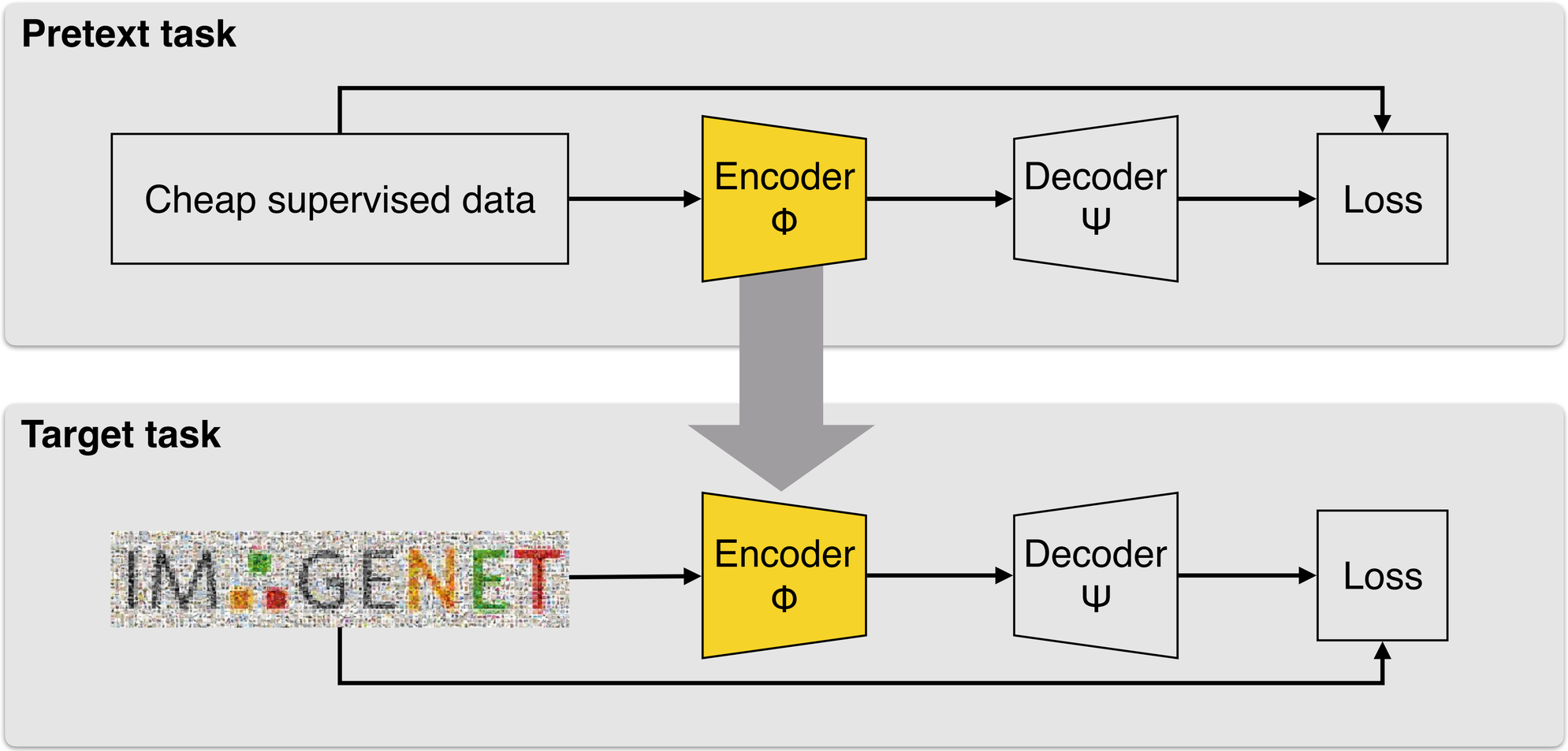
Deep image prior

Self-supervised features

Self-supervised structure

Deep image prior

Self-supervised features: transfer learning



Find a **pretext task** to **pre-train** a model Φ

The pretext tasks comes with **cheap supervision**

Fine-tune the model for a **target task**

Far less annotations are now required

Where to get supervision from

Self-supervised features

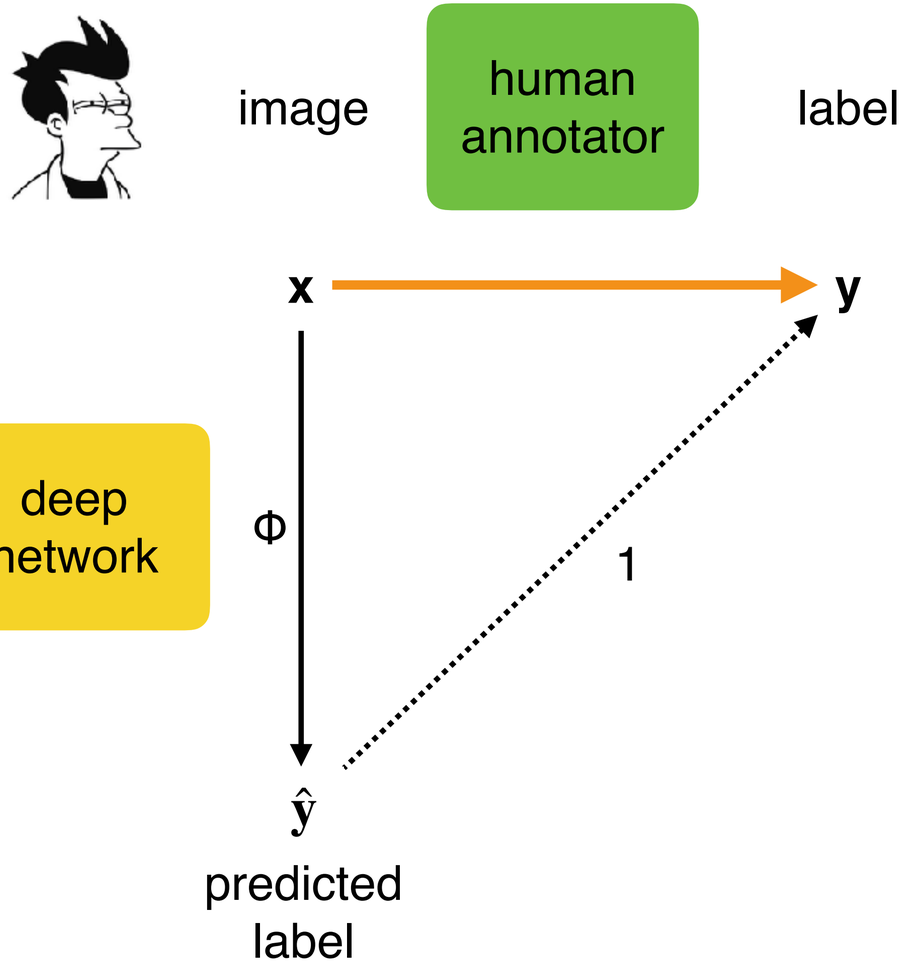
Perturbations

Motion

Modalities

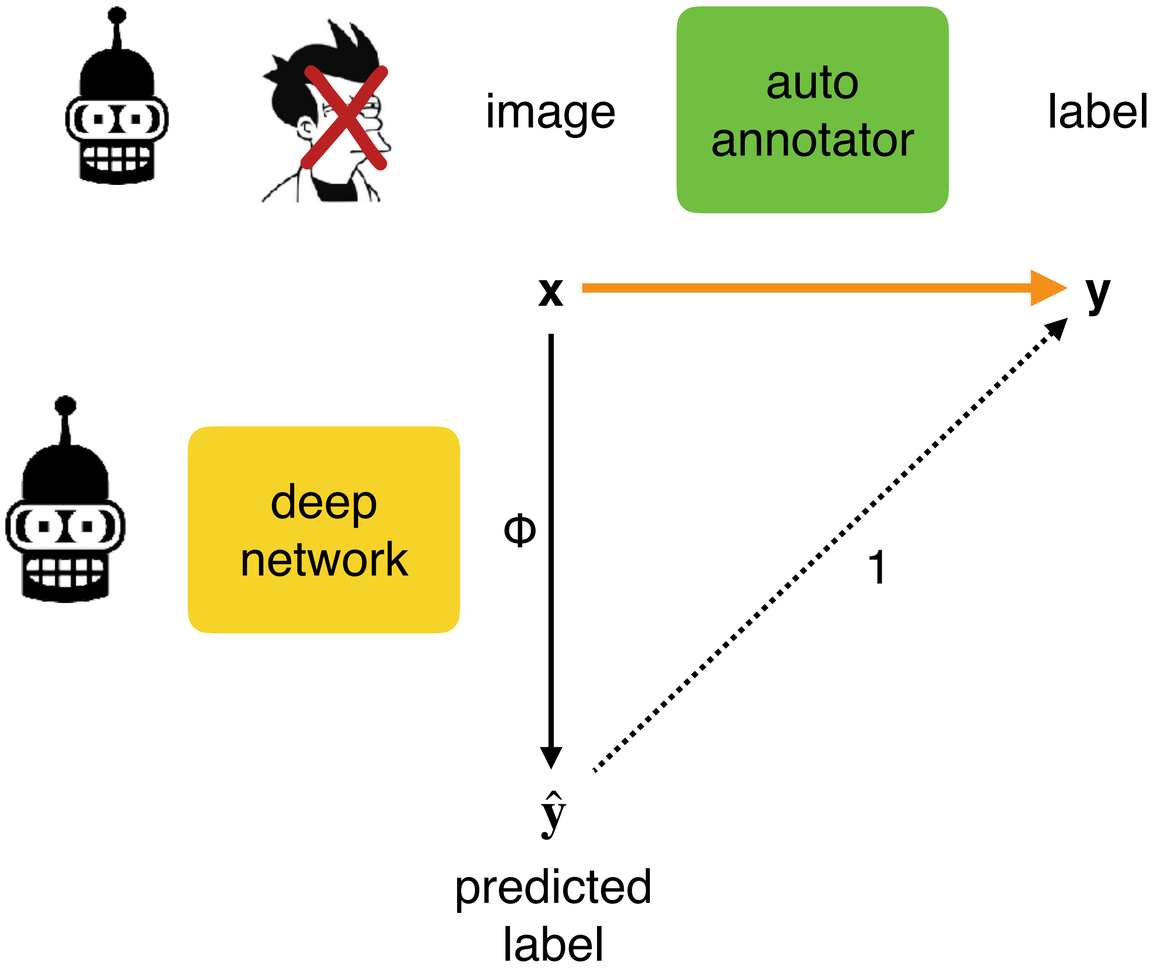
Standard supervision

Goal: learn a model Φ that reproduces a human annotator



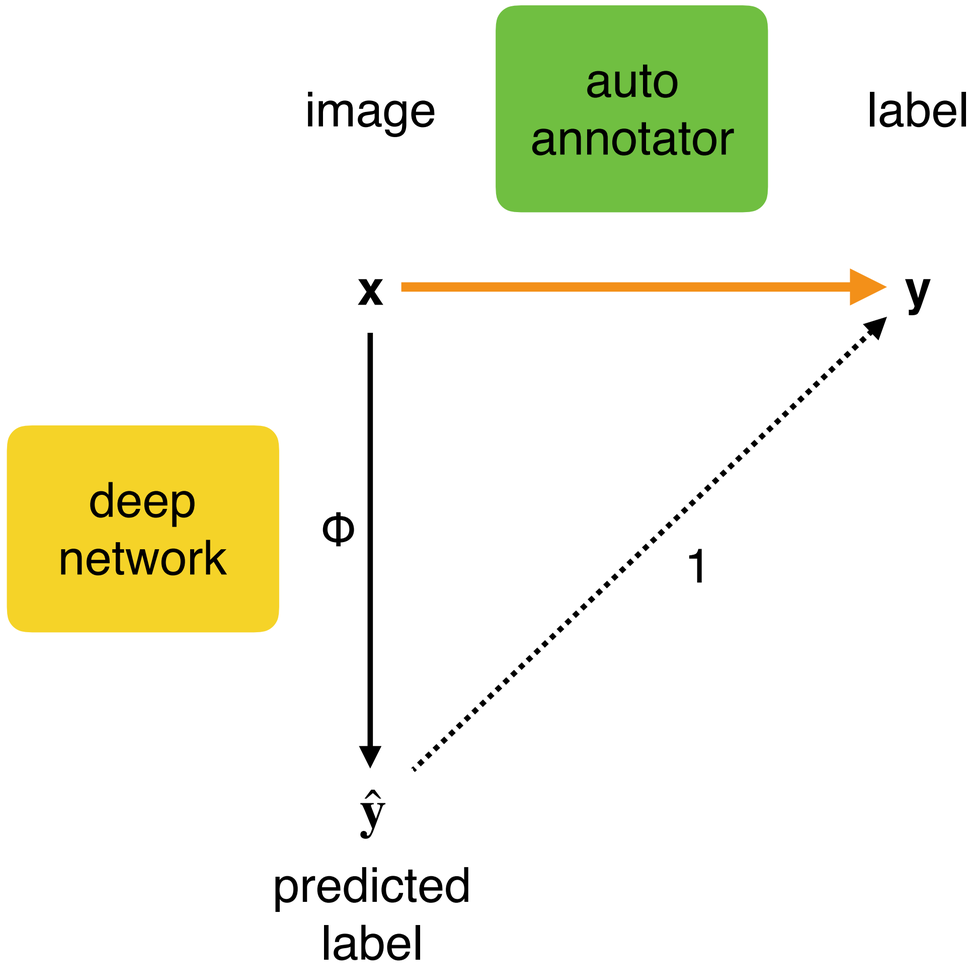
Automatic annotator?

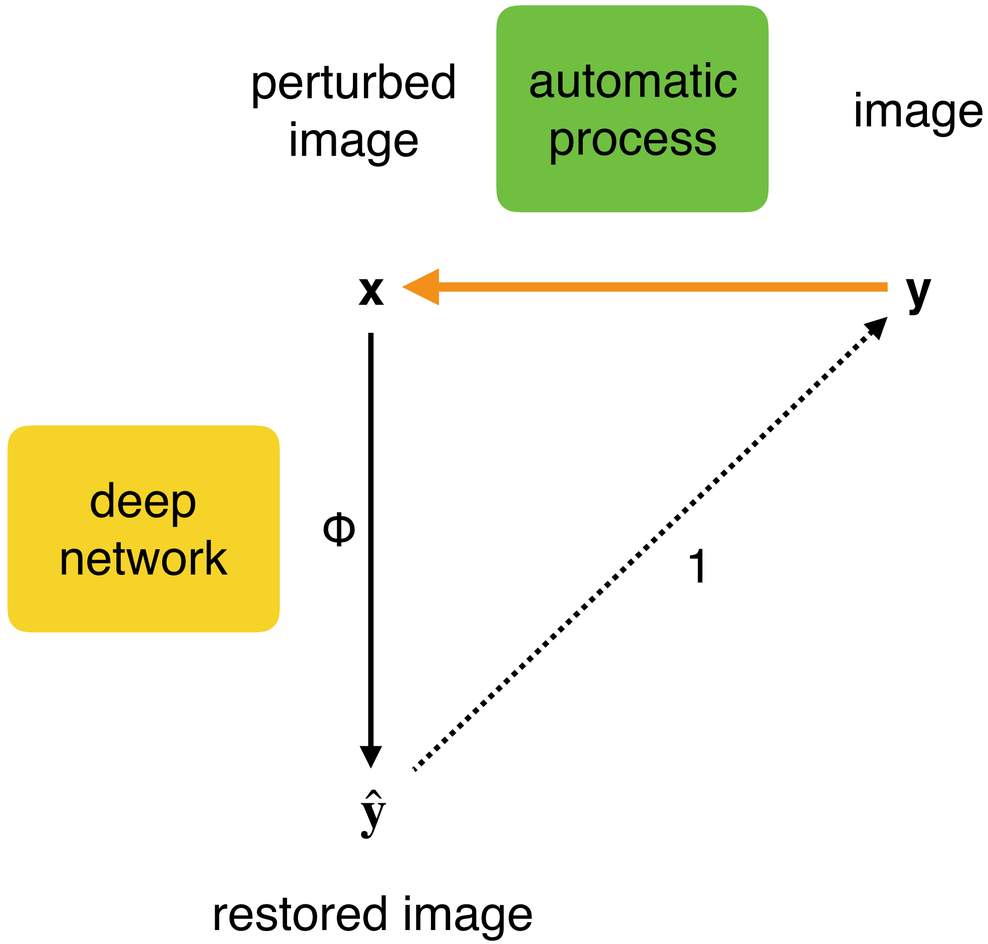
Nope: the "auto annotator" is just as complex as the model Φ



Self-supervision from perturbations

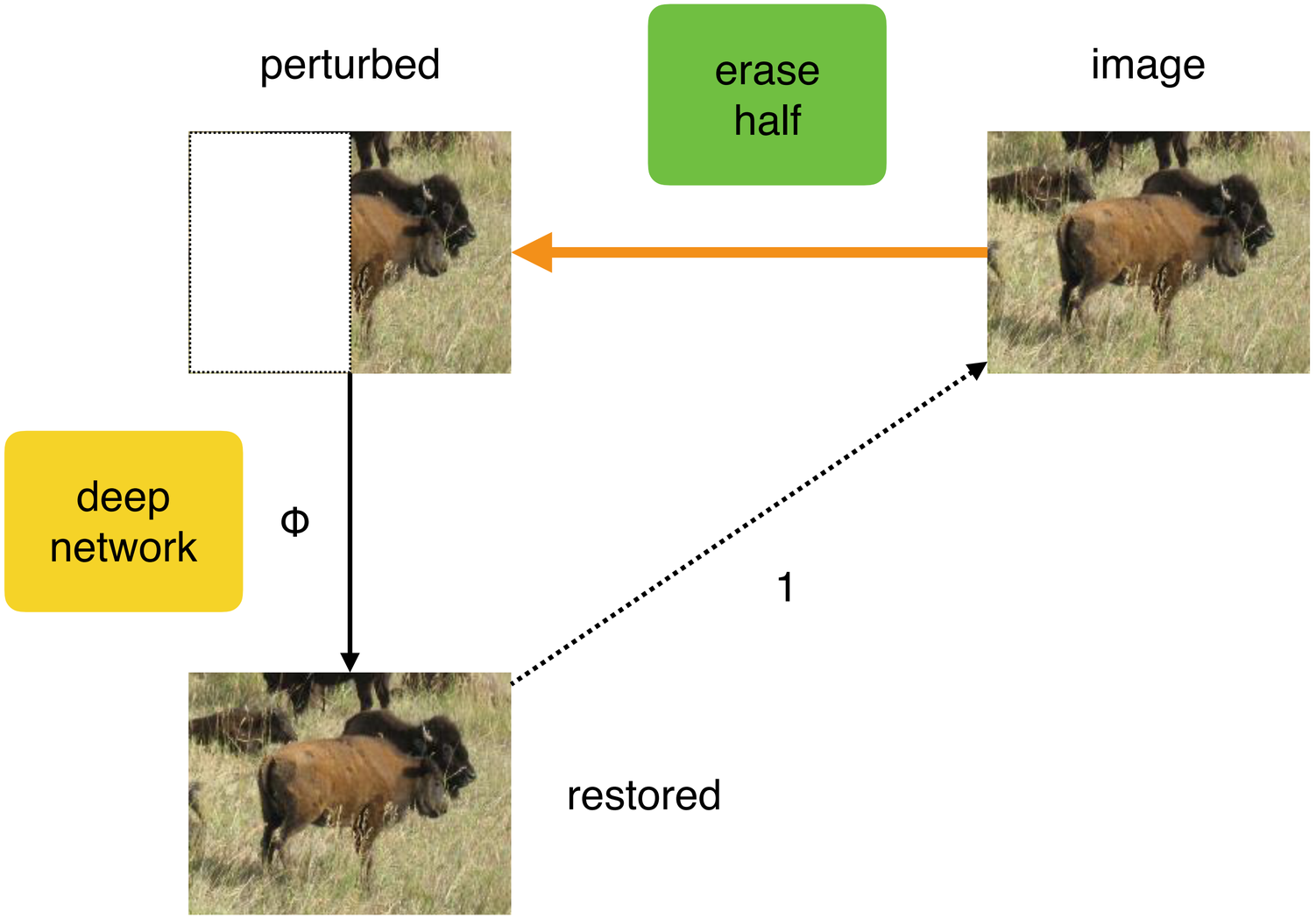
Use the image as label, their perturbations as input





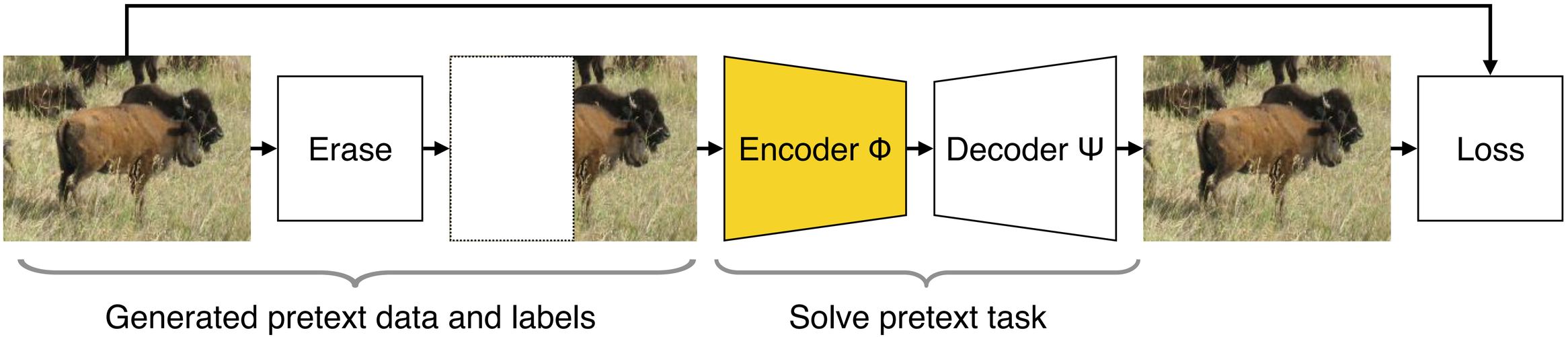
Self-supervision from perturbations

Example perturbation = delete half of the image



Self-supervision from perturbations

Concrete learning scheme



Intuition: completing an image may **require the network to learn about objects**

Perturbations useful for self-supervision

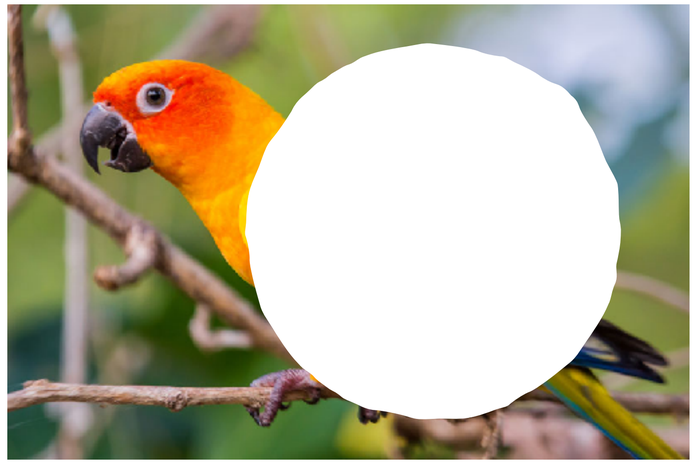
decolorize

Learning representations for automatic colorization. Larsson, Maire, Shakhnarovich. ECCV, 2016.

Colorful image colorization. Zhang, Isola, Efros. ECCV, 2016.

Colorization as a proxy task for visual understanding. Larsson, Maire, Shakhnarovich. CVPR, 2017.

erase



Context encoders: Feature learning by inpainting. Pathak, Krähenbühl, Donahue, Darrell, Efros. CVPR, 2016.

scramble

Unsupervised visual representation learning by context prediction. Doersch, Gupta, Efros. ICCV, 2015.

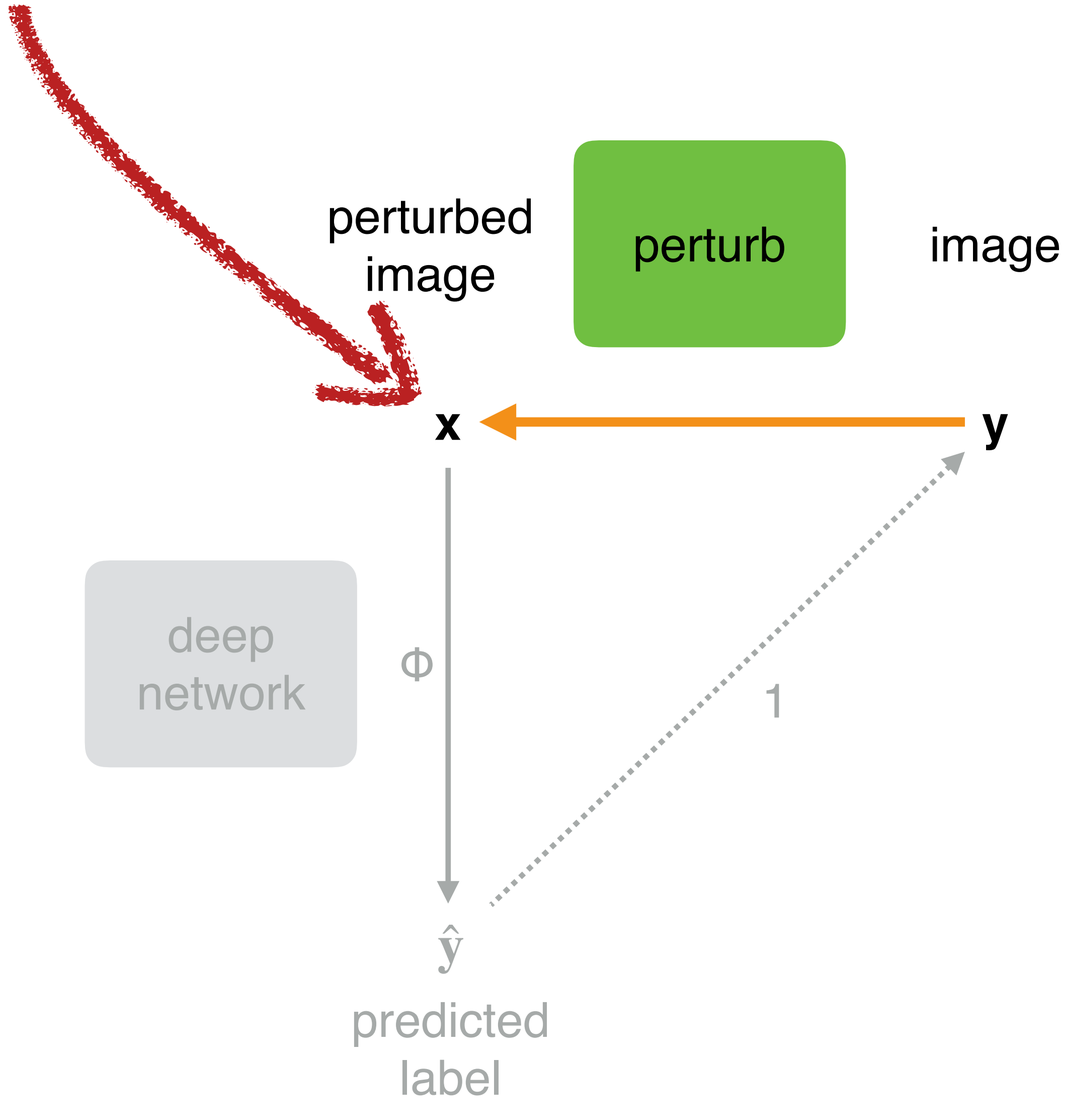
Unsupervised learning of visual representations by solving jigsaw puzzles. Noroozi Favaro. ECCV, 2016.

Boosting self-supervised learning via knowledge transfer. Noroozi, Vinjimor, Favaro, Pirsiavash. CVPR, 2018.

Improvements to context based self-supervised learning. Mundhenk, Ho, Chen. CVPR, 2018.

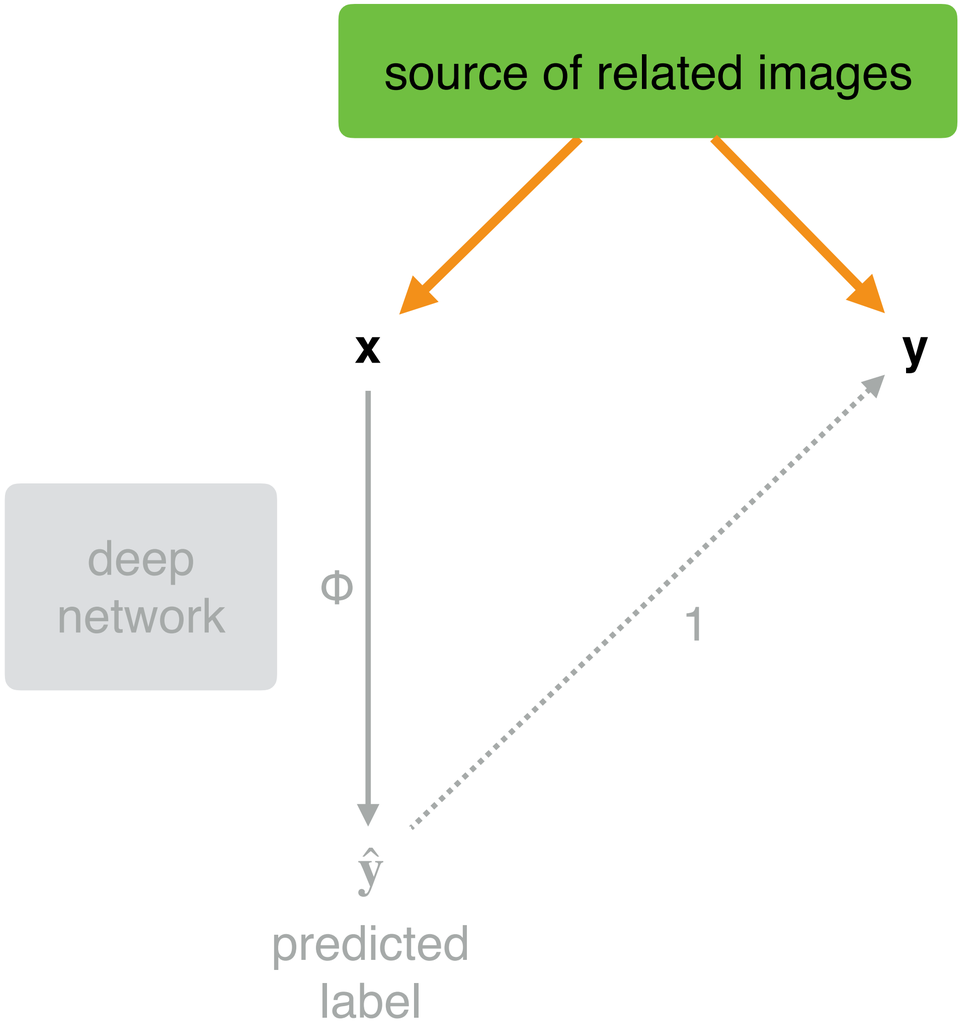
Self-supervision from perturbations

Disadvantage: the network learns to operate on perturbed data

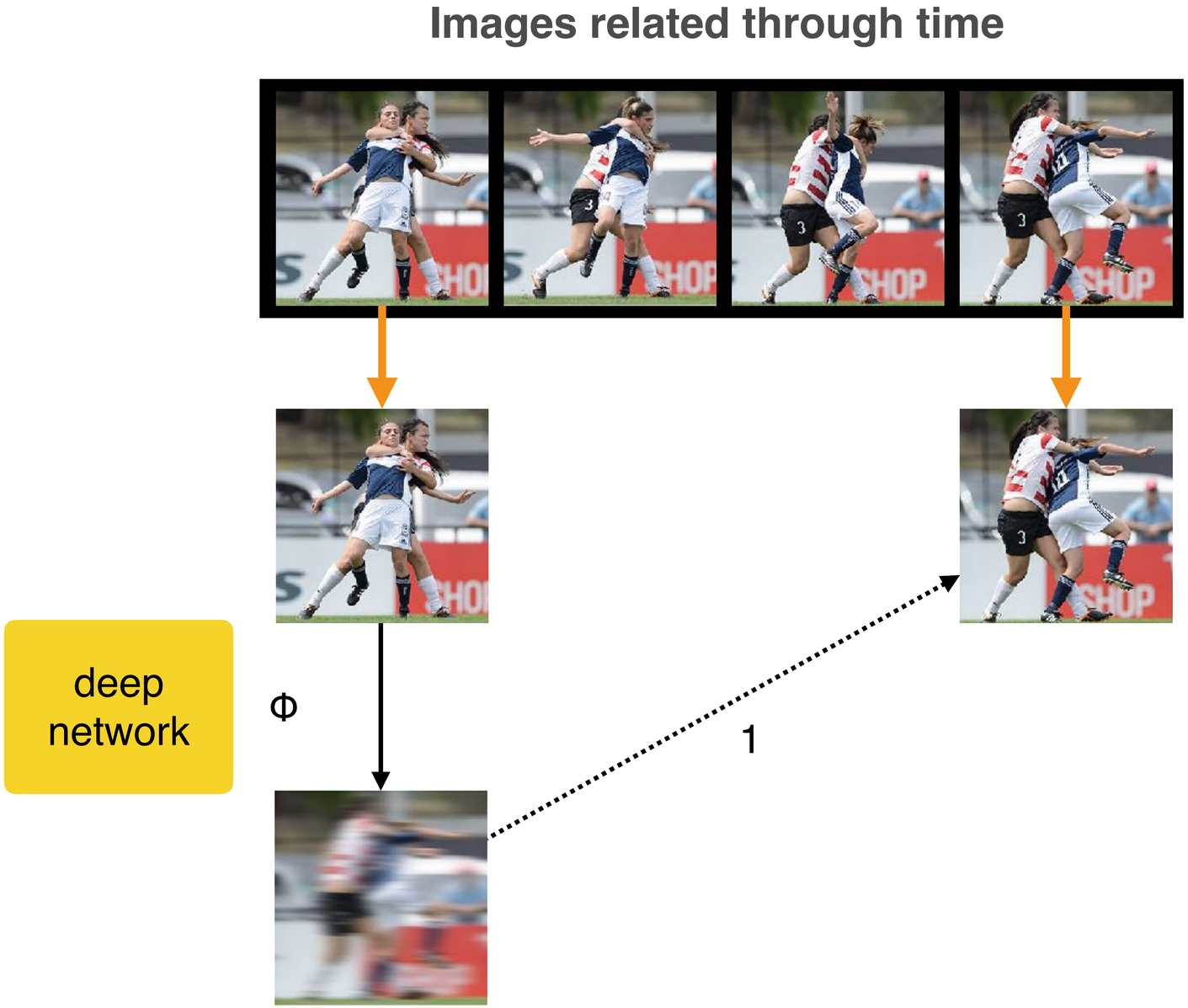


Self-supervision from related images

Look for image pairs that are correlated form the outset



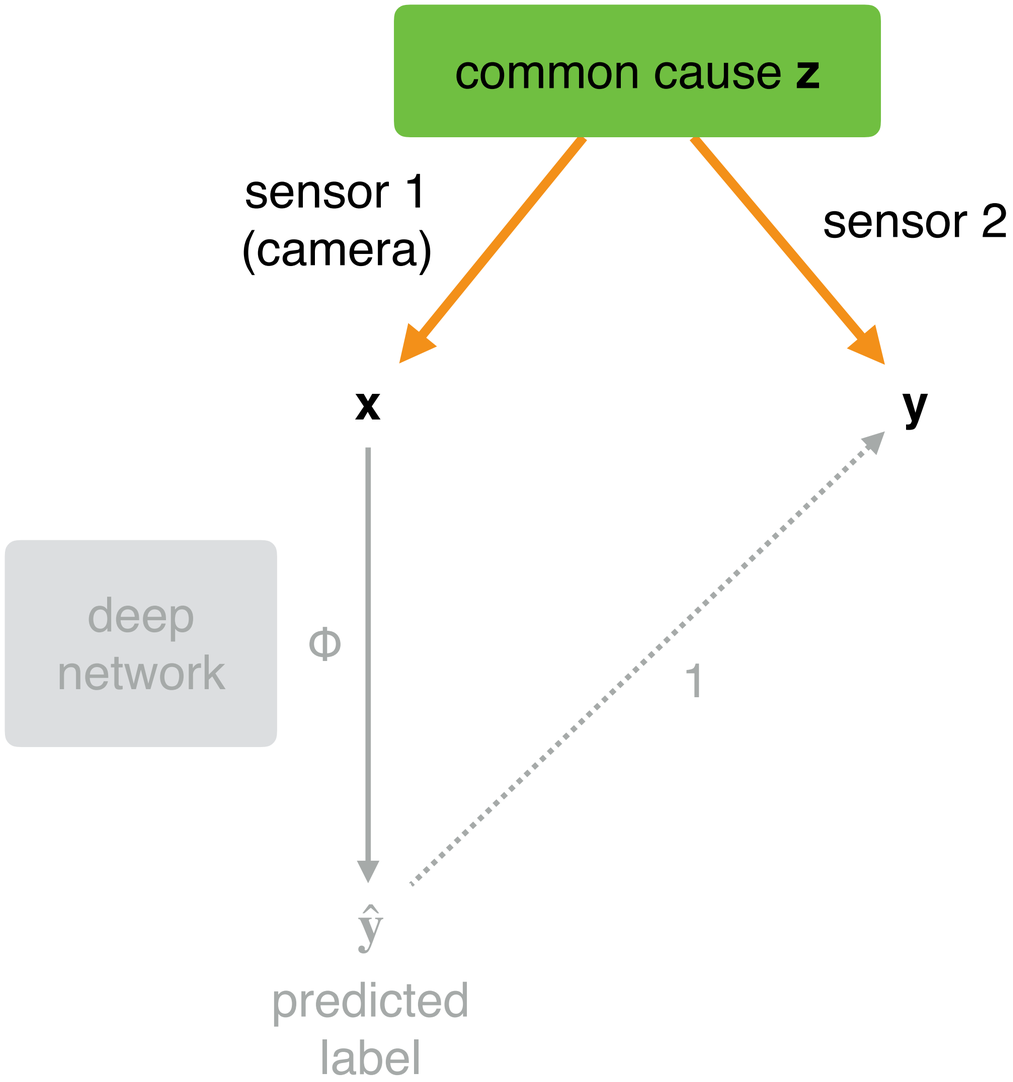
Self-supervision using video sequence



Unsupervised learning of video representations using LSTMs. Srivastava, Mansimov, Salakhutdinov. Proc. ICML, 2015.

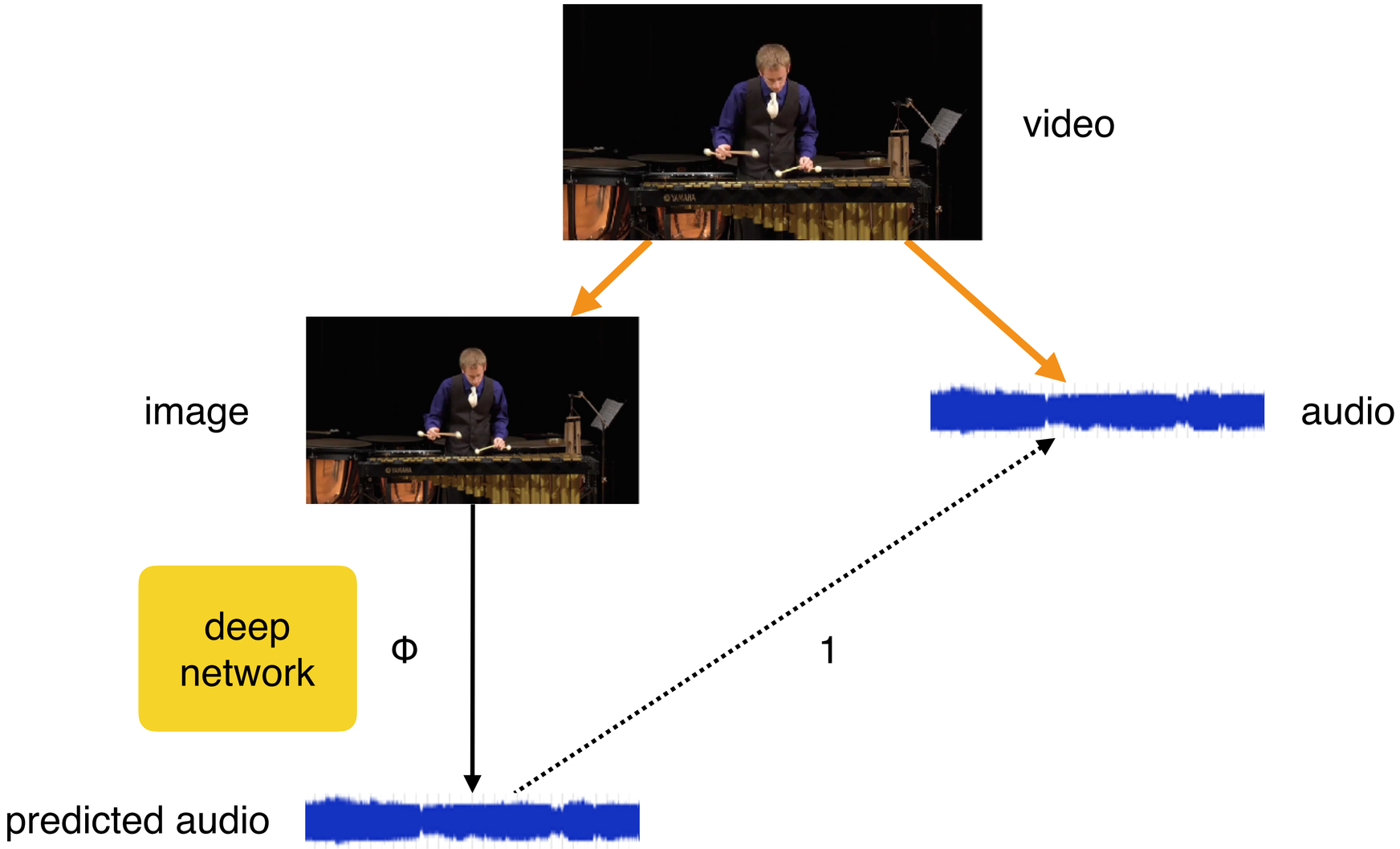
Self-supervision using multiple modalities

Data sharing the same root cause



Self-supervision using multiple modalities

Example: reconstruct sound from images



Video from: **Audio-visual scene analysis with self-supervised multisensory features.** Owens Efros. Proc. ECCV, 2018.

Accidental supervision

Example: photographer bias




Task: find out which picture is **upright**

The supervisory signal is in the photographer bias:
people take pictures with a specific orientation

Unsupervised representation learning by predicting image rotations. Girard, Singh, Komodakis. Proc. ICLR, 2018.

How to setup the learning problem

Self-supervised features

Perturbations

Motion

Modalities

Probability

Constraining

Marginalising

Correlating

How to setup the learning problem

Self-supervised features

Perturbations

Motion

Modalities

Probability

Constraining

Marginalising

Correlating

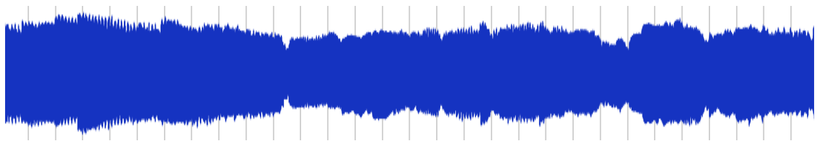
Addressing ambiguous pretext tasks

Many pretext tasks are ill-posed

colorization

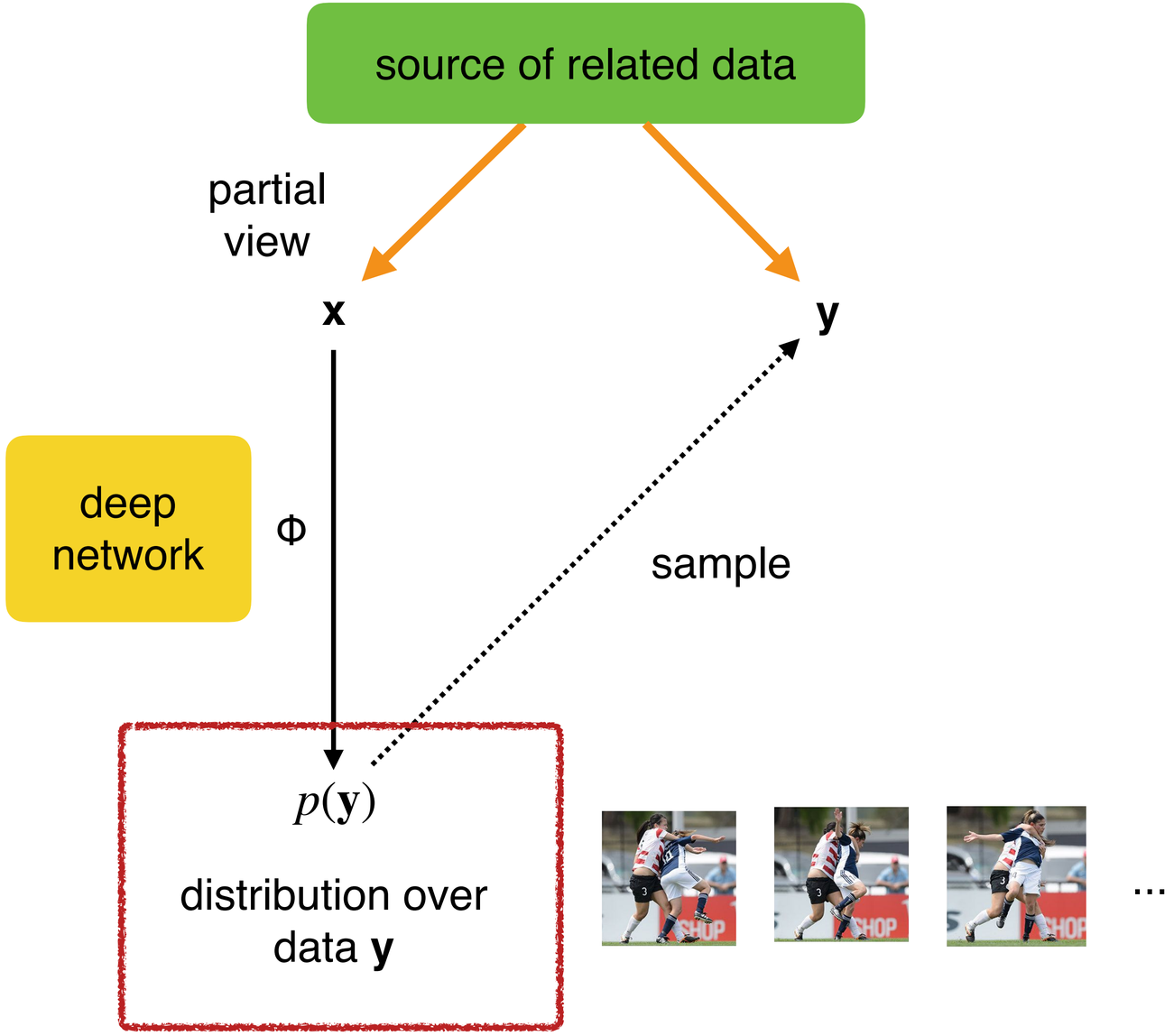
future frame prediction

image to sound



Addressing ambiguous pretext tasks

Approach 1: use a probability distribution



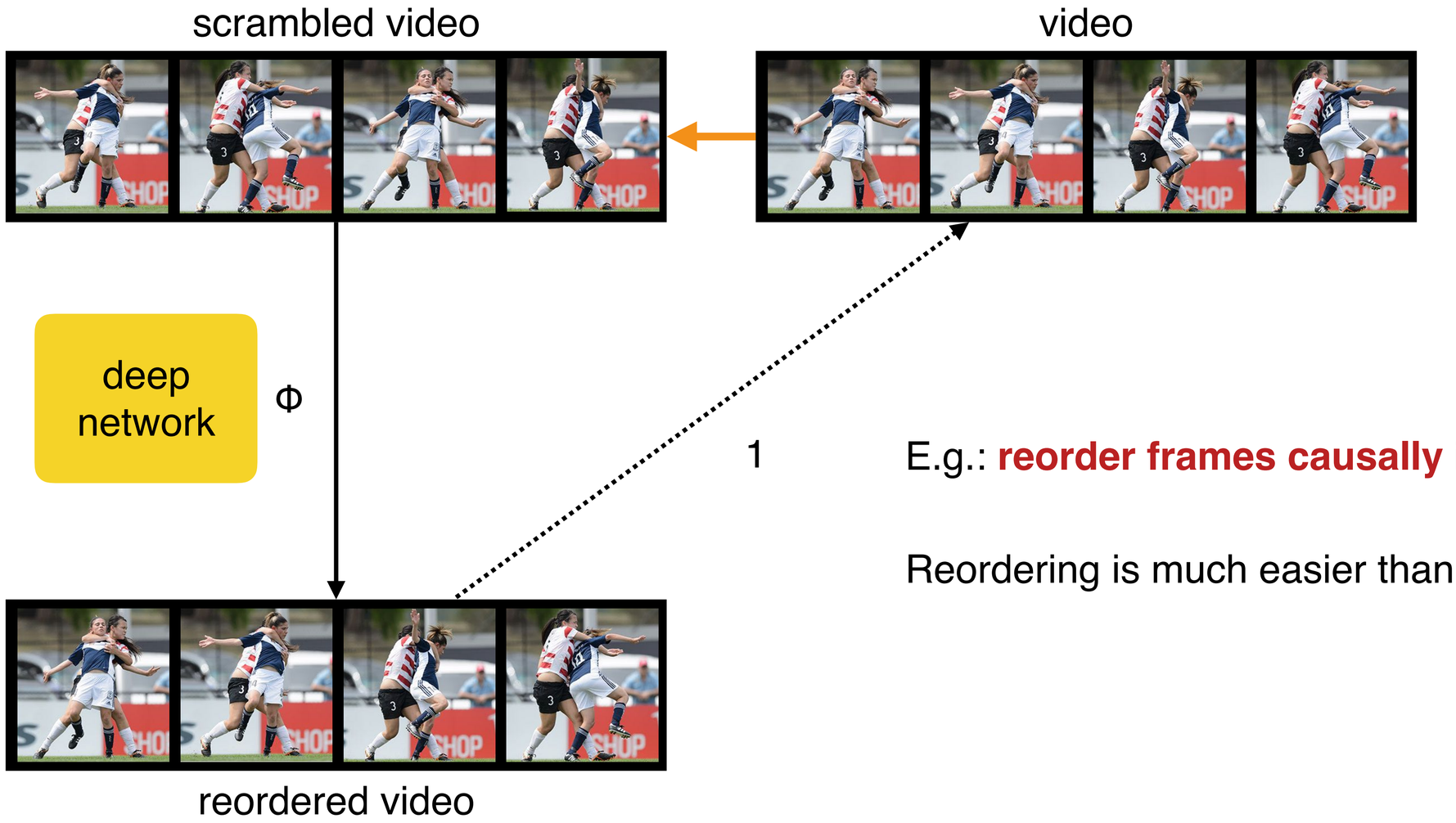
Explicitly model ambiguity by predicting a probability distribution

Disadvantage: probabilistic modeling can **challenging** or too **simplistic**

Colorful image colorization. Zhang, Isola, Efros. ECCV, 2016.

Addressing ambiguous pretext tasks

Approach 2: constrain the prediction task



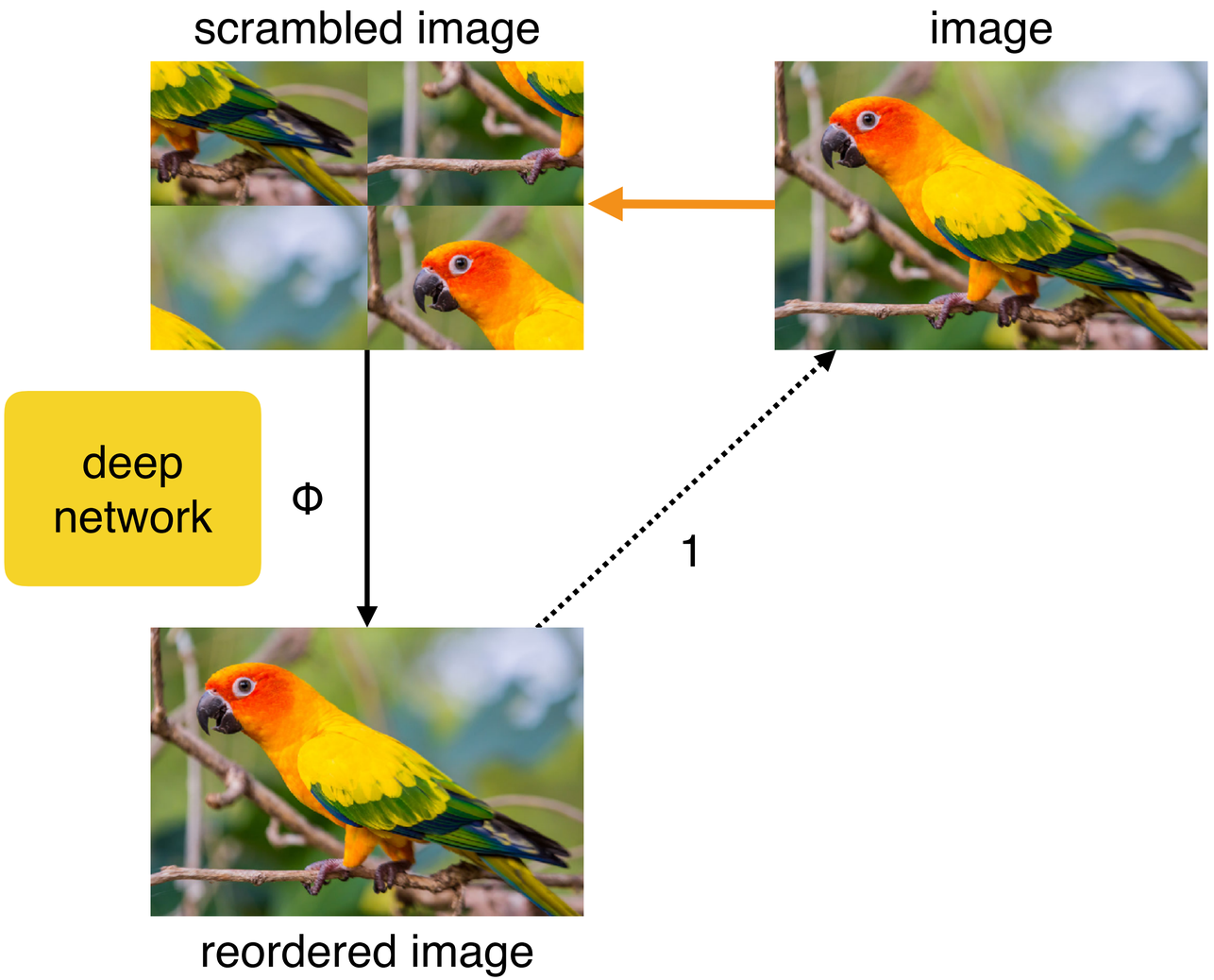
E.g.: **reorder frames causally** instead of generating them

Reordering is much easier than generation

Shuffle and learn: Unsupervised learning using temporal order verification. Misra, Zitnick, Hebert. ECCV, 2006.
Learning and using the arrow of time. Wei, Lim, Zisserman, Freeman. CVPR, 2018.
Unsupervised representation learning by sorting sequence. Lee, Huang, Singh, Yang. ICCV, 2017.

Addressing ambiguous pretext tasks

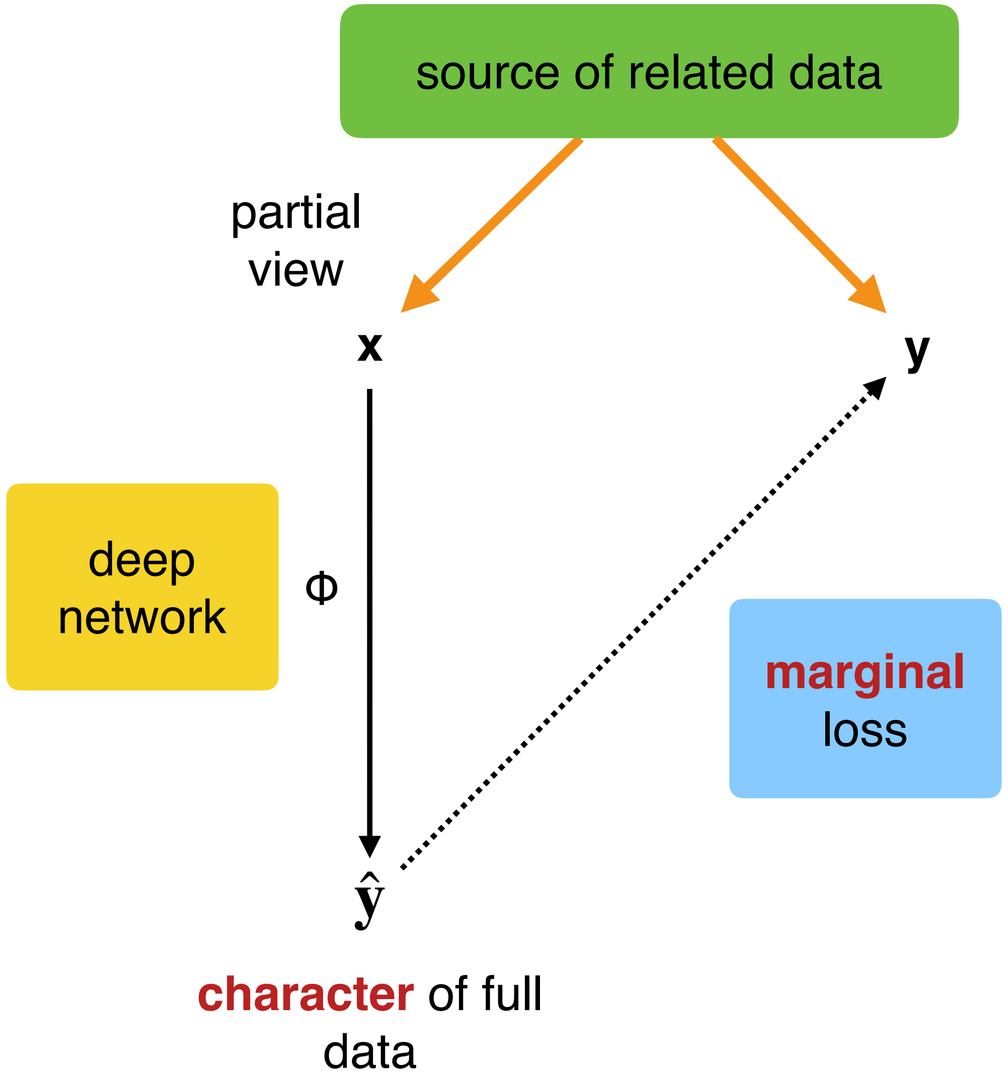
Approach 2: constrain the prediction task



Reordering for image patches

Addressing ambiguous pretext tasks

Approach 3: reduce the amount of predicted information



Predict a "**character**" of the full data

This is captured by a "**marginal loss**"

- Via projection

$$\|\hat{\mathbf{y}} - \pi(\mathbf{y})\|$$

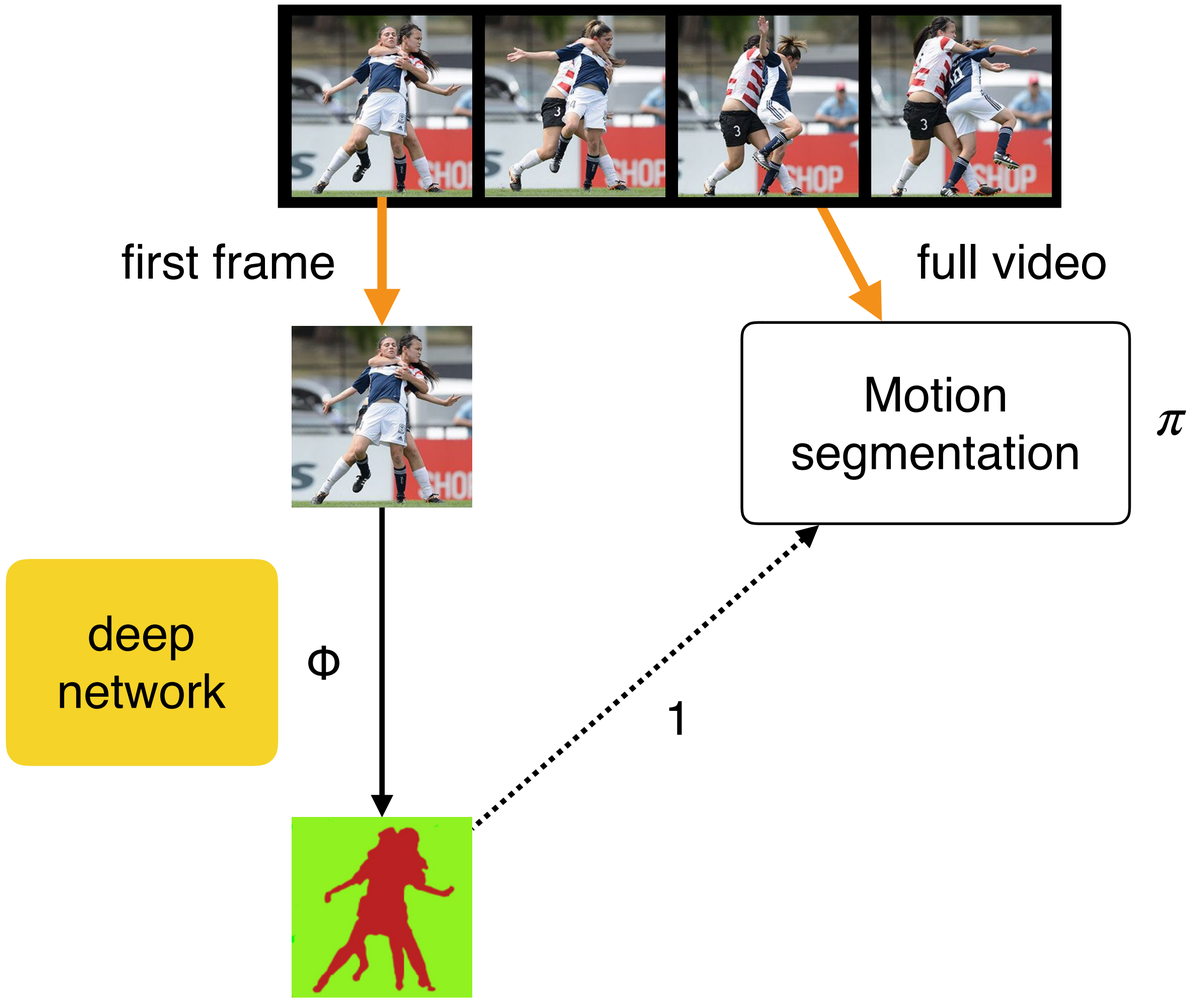
- Via marginalization

$$\min_{\eta} \|\mathbf{y} - f(\hat{\mathbf{y}}, \eta)\|$$

Disadvantage: ad-hoc design

Addressing ambiguous pretext tasks

Approach 3: reduce the amount of predicted information

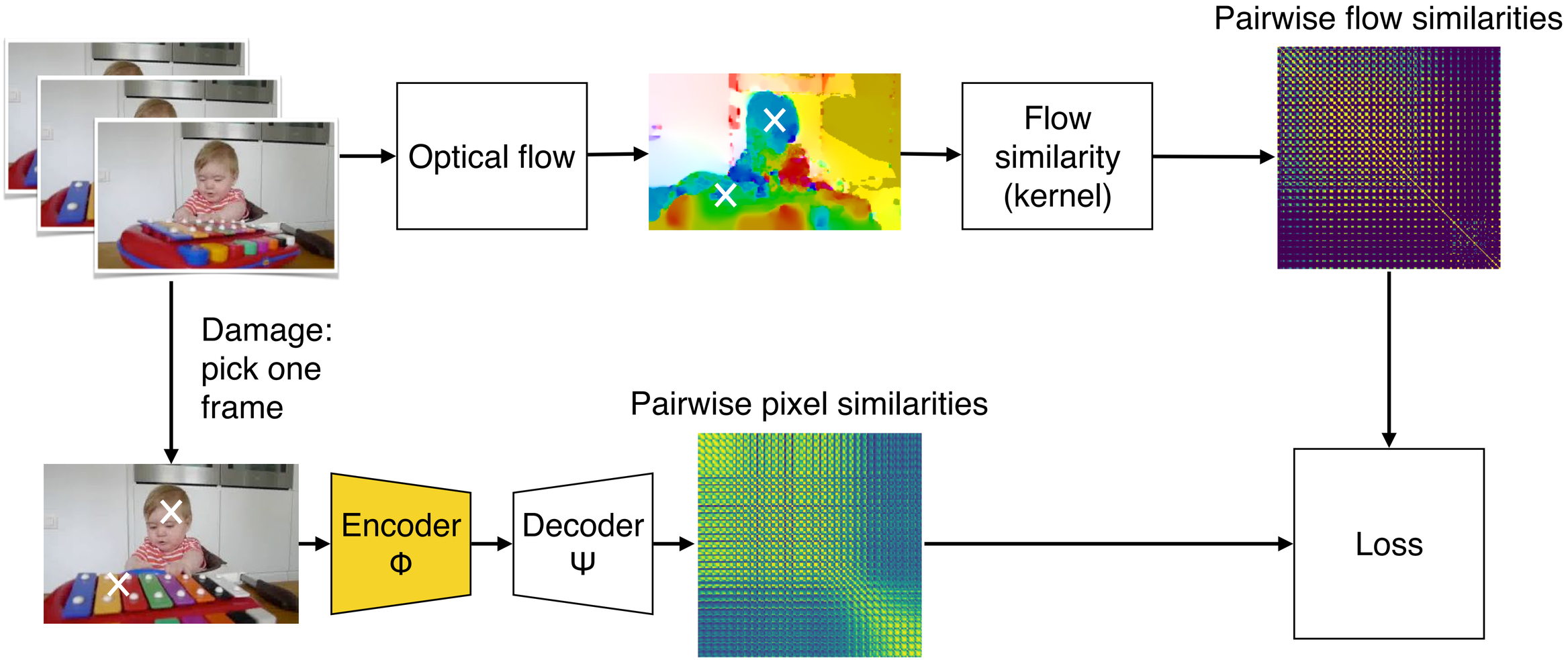


Example: only **predict what could move** in an image

Learning visual groups from co-occurrences in space and time. Isola, Zoran, Krishnan, Adelson. ICLR Workshop, 2015.
Learning features by watching objects move. Pathak, Girshick, Dollár, Darrell, Hariharan. CVPR, 2017.

Common fate principle

Group together what moves together



Pixel Optical Flow Similarity for Self-Supervised Learning. Mahendran, Thewlis, Vedaldi. arXiv, 2018

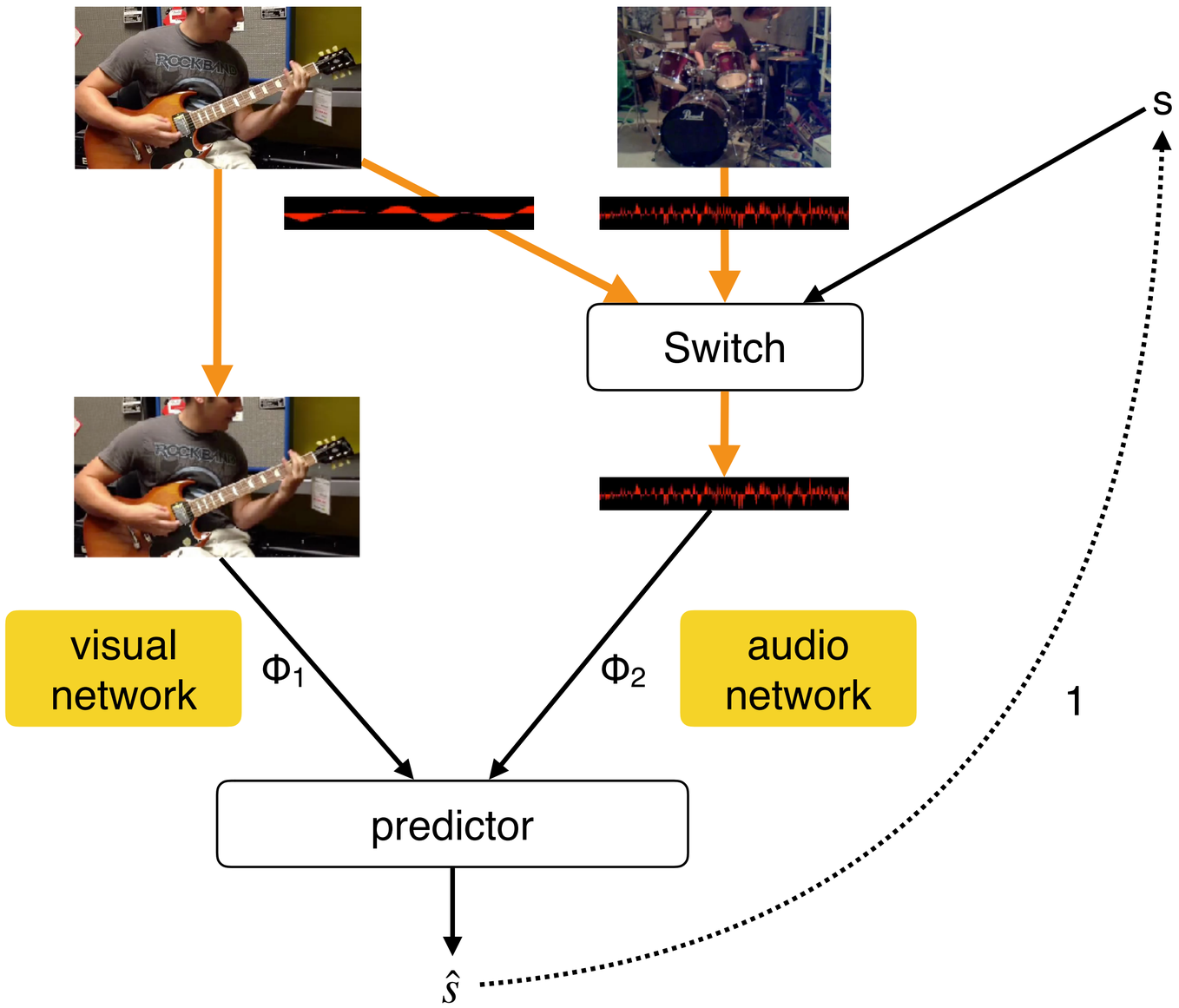
Addressing ambiguous pretext tasks

Approach 4: learning correspondences

Learning classification from unlabelled data. De Sa. Proc. NIPS, 1994.
Pixels that sound. Kidron, Schechner, Elad. Proc. CVPR, 2005.
Ambient sound provides supervision for visual learning. Owens, Jiajun, McDermott, Freeman, Torralba. Proc. ECCV, 2016.
Visually indicated sounds. Owens, McDermott, Torralba, Adelson, Freeman. Proc. CVPR, 2016.
Audio-visual scene analysis with self-supervised multisensory features. Owens Efros. Proc. ECCV, 2018.

Matching modalities

Audio-visual correspondences



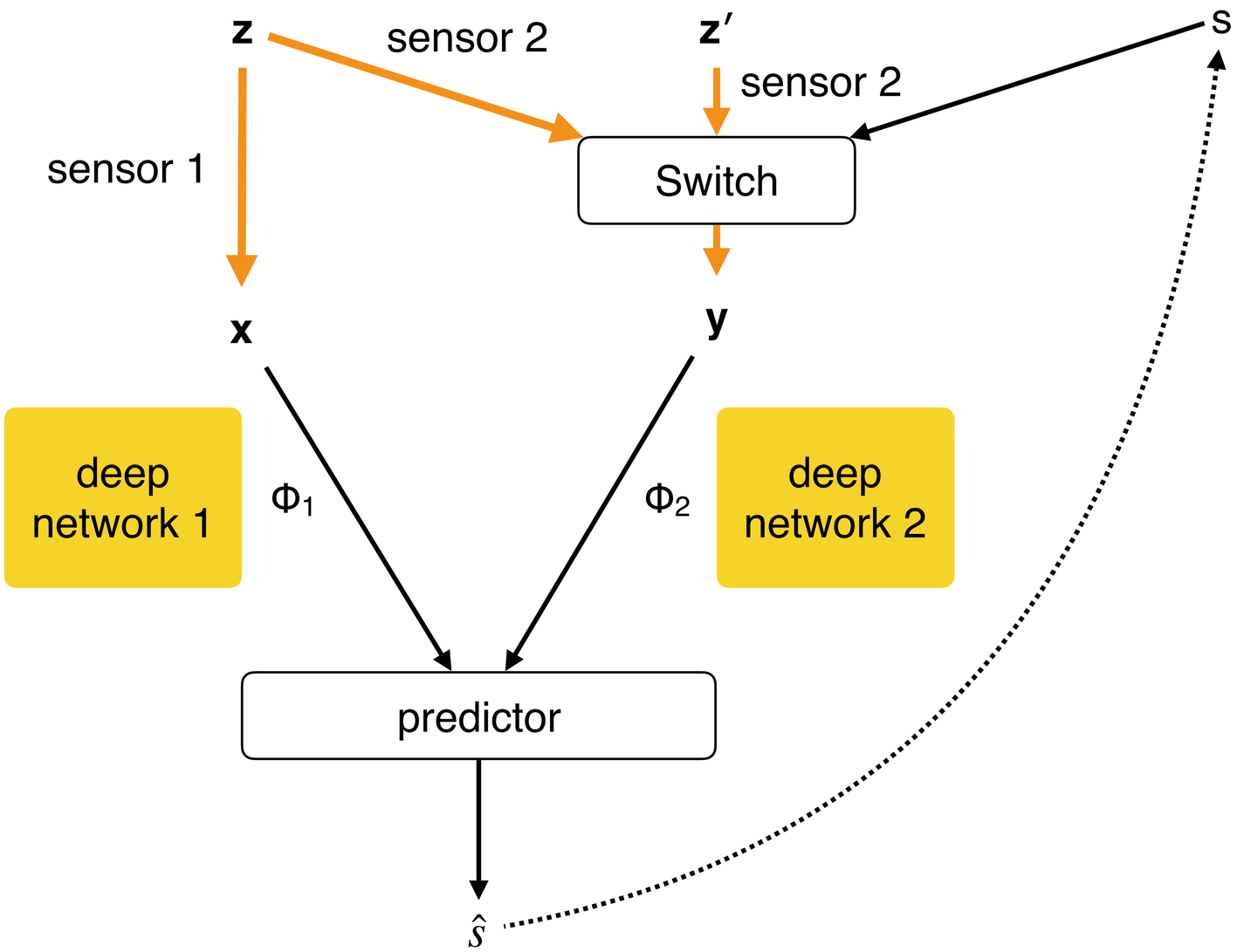
Randomly extract **x** and **y** from either the same or different data

Learn to tell which one is the case

Reduces to predicting a binary switch

Addressing ambiguous pretext tasks

Approach 4: learning correspondences



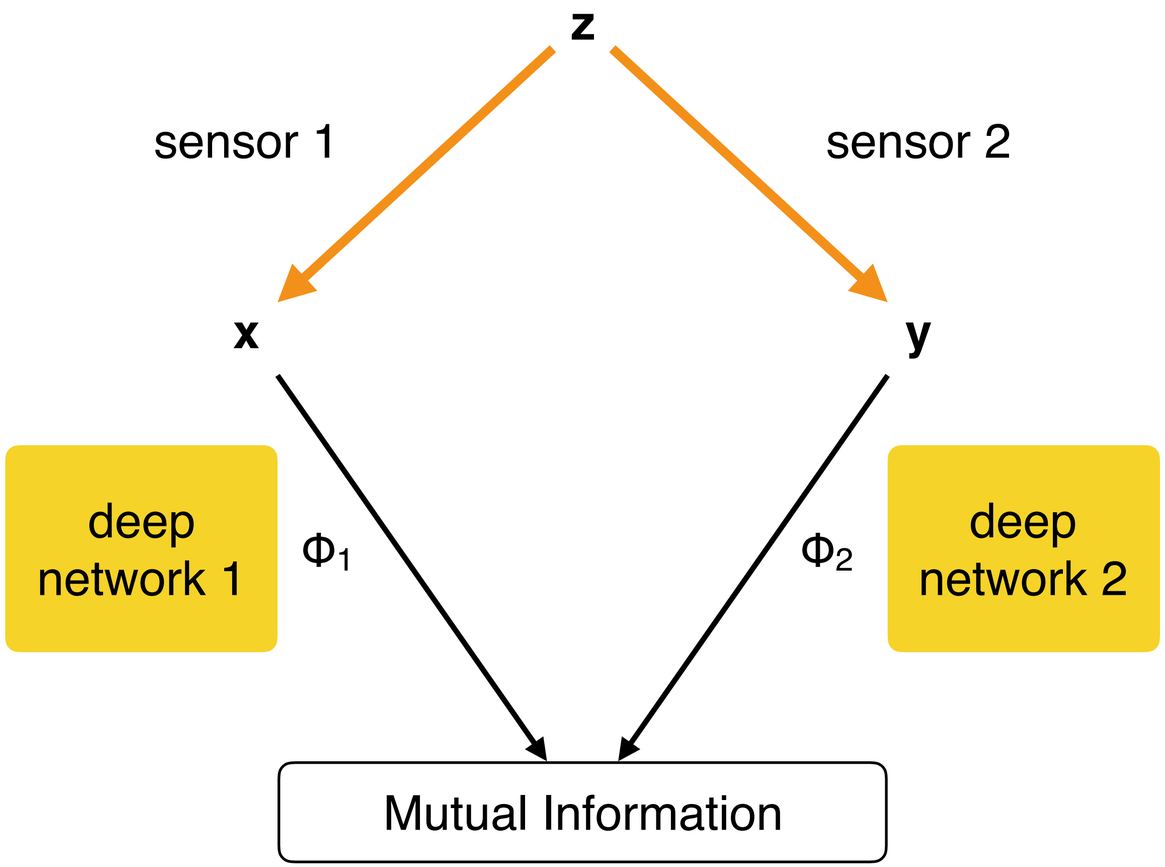
Randomly extract **x** and **y** from either the same or different data

Learn to tell which one is the case

Reduces to predicting a binary switch

A statistical “distillation” principle

Co-occurrences may be better captured by mutual information



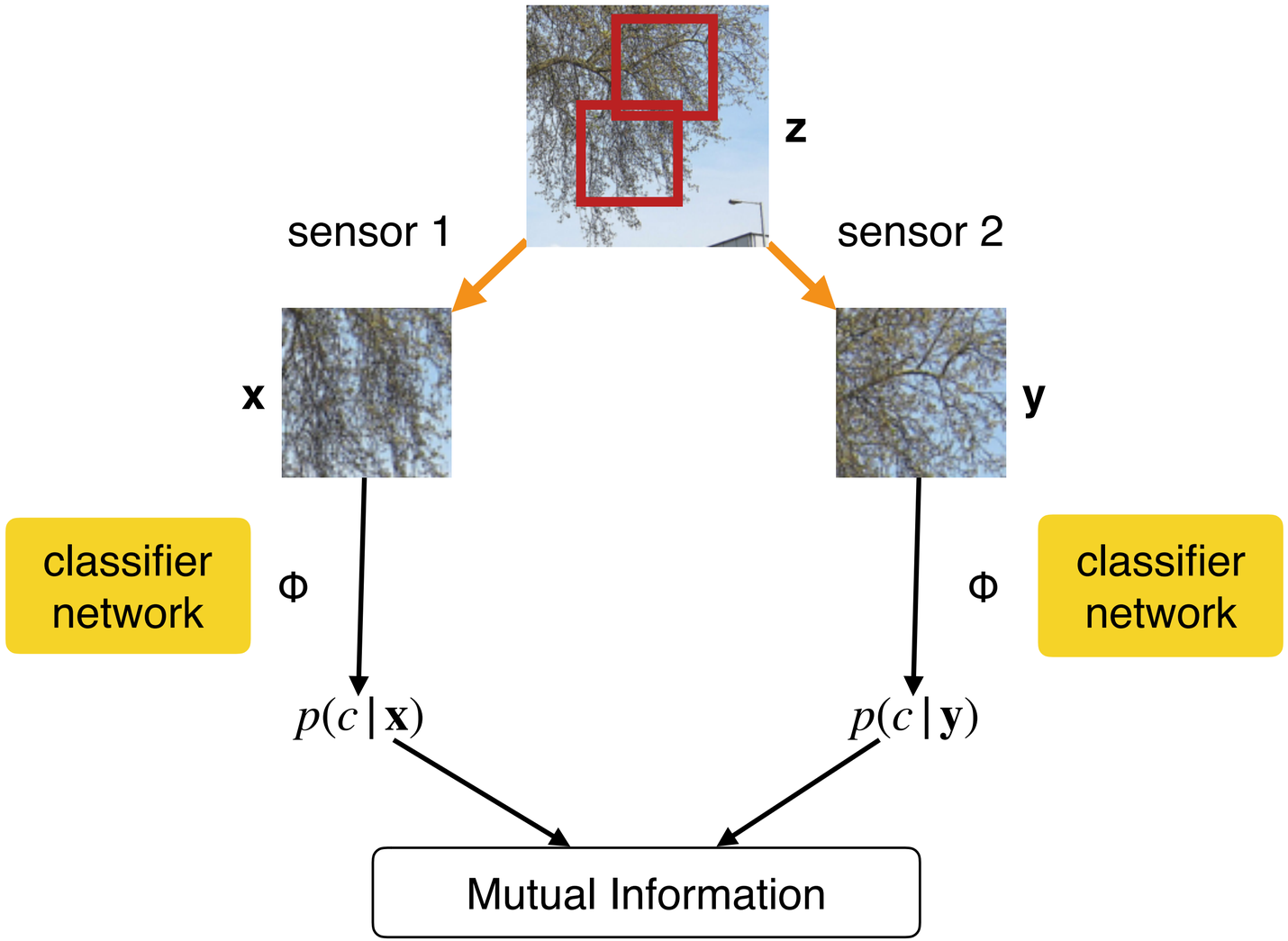
Learning objective

$$\max_{\Phi_1,\Phi_2} I(\Phi_1(f_2(\mathbf{z})), \Phi_2(f_2(\mathbf{z})))$$

Self-organizing neural network that discovers surfaces in random-dot stereograms. Becker Hinton. Nature, 355(6356):161, 1992
Invariant information distillation for unsupervised image segmentation and clustering. Ji, Henriques, Vedaldi. arXiv, 2018.

Image clustering and segmentation

Learn maximally mutually informative classes



Take image pairs related by **proximity** and/or **geometric transformation**

Classes are likely to be the same, or at least correlated

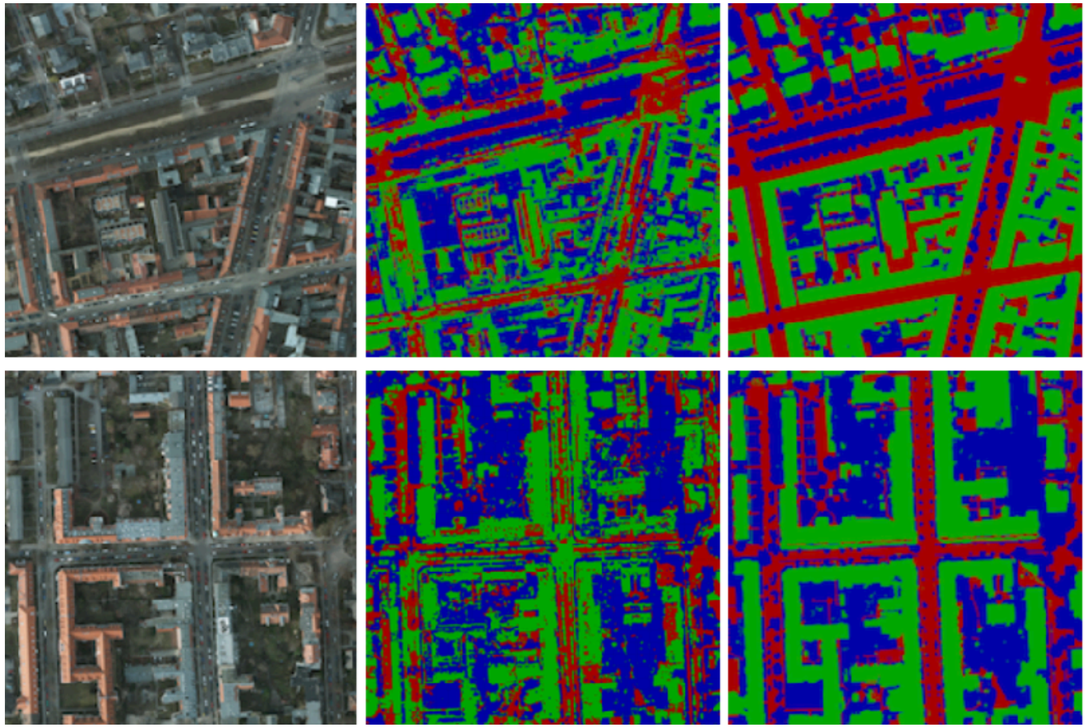
Learn a classification function to maximize the **mutual information between classes**

Self-organizing neural network that discovers surfaces in random-dot stereograms. Becker Hinton. Nature, 355(6356):161, 1992
Invariant information distillation for unsupervised image segmentation and clustering. Ji, Henriques, Vedaldi. arXiv, 2018.

Clustering results

ImageNet fruits clustering

Satellite image segmentation



unsup. seg. gt

Invariant information distillation for unsupervised image segmentation and clustering. Ji, Henriques, Vedaldi. arXiv, 2018.

Self-supervised features

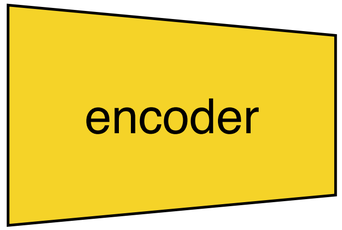
Self-supervised structure

Deep image prior

From features to structure

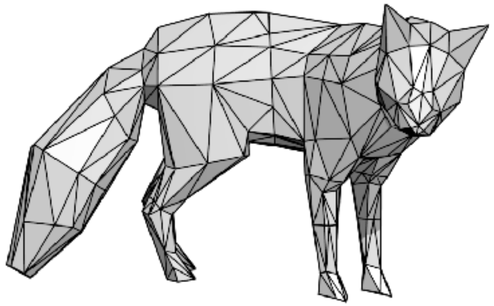
Can we learn the structure of visual objects explicitly?

Implicit features



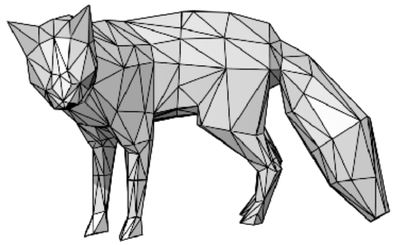
vs

Explicit concept



Object model

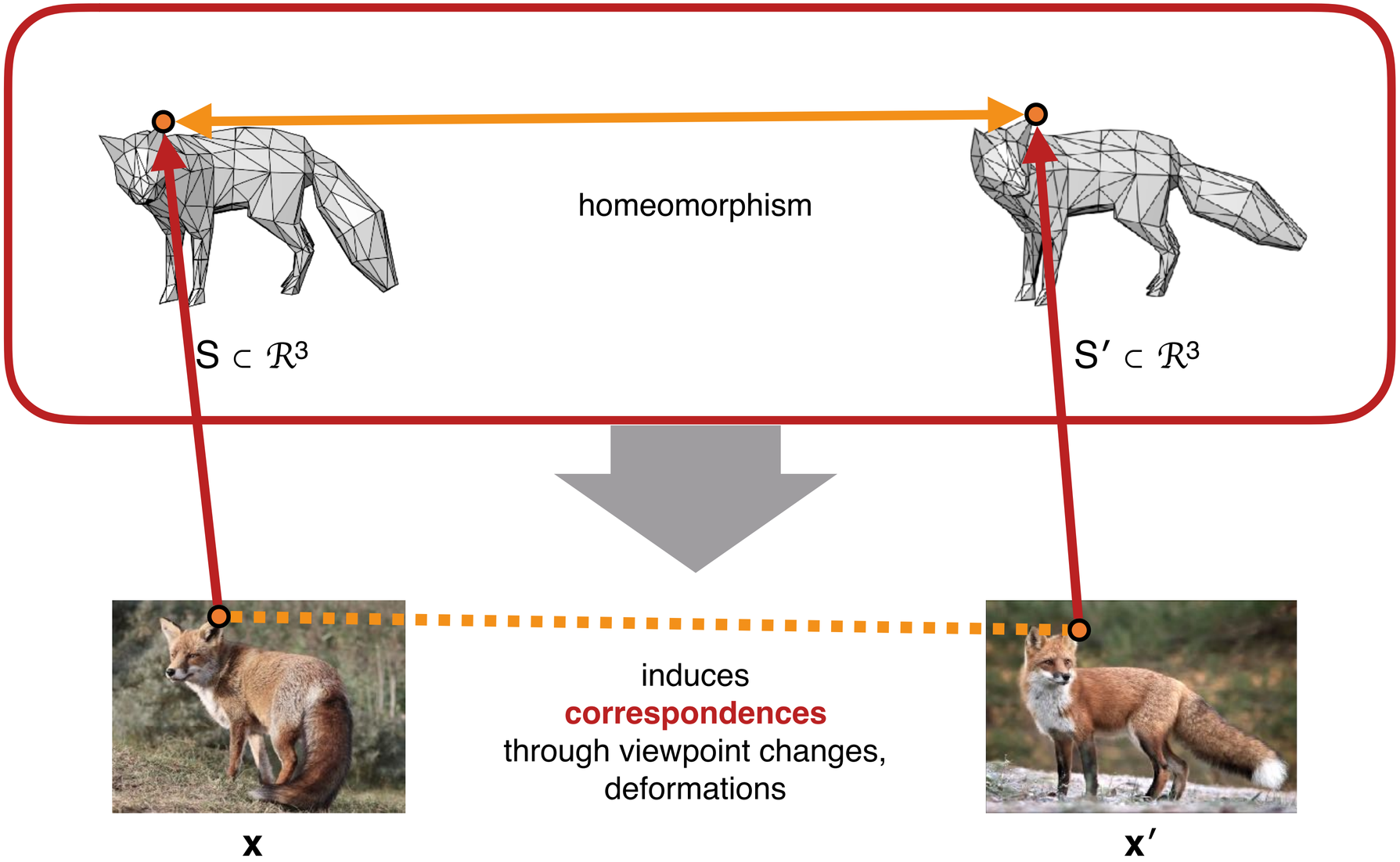
Take I: A 3D surface



$S \subset \mathcal{R}^3$

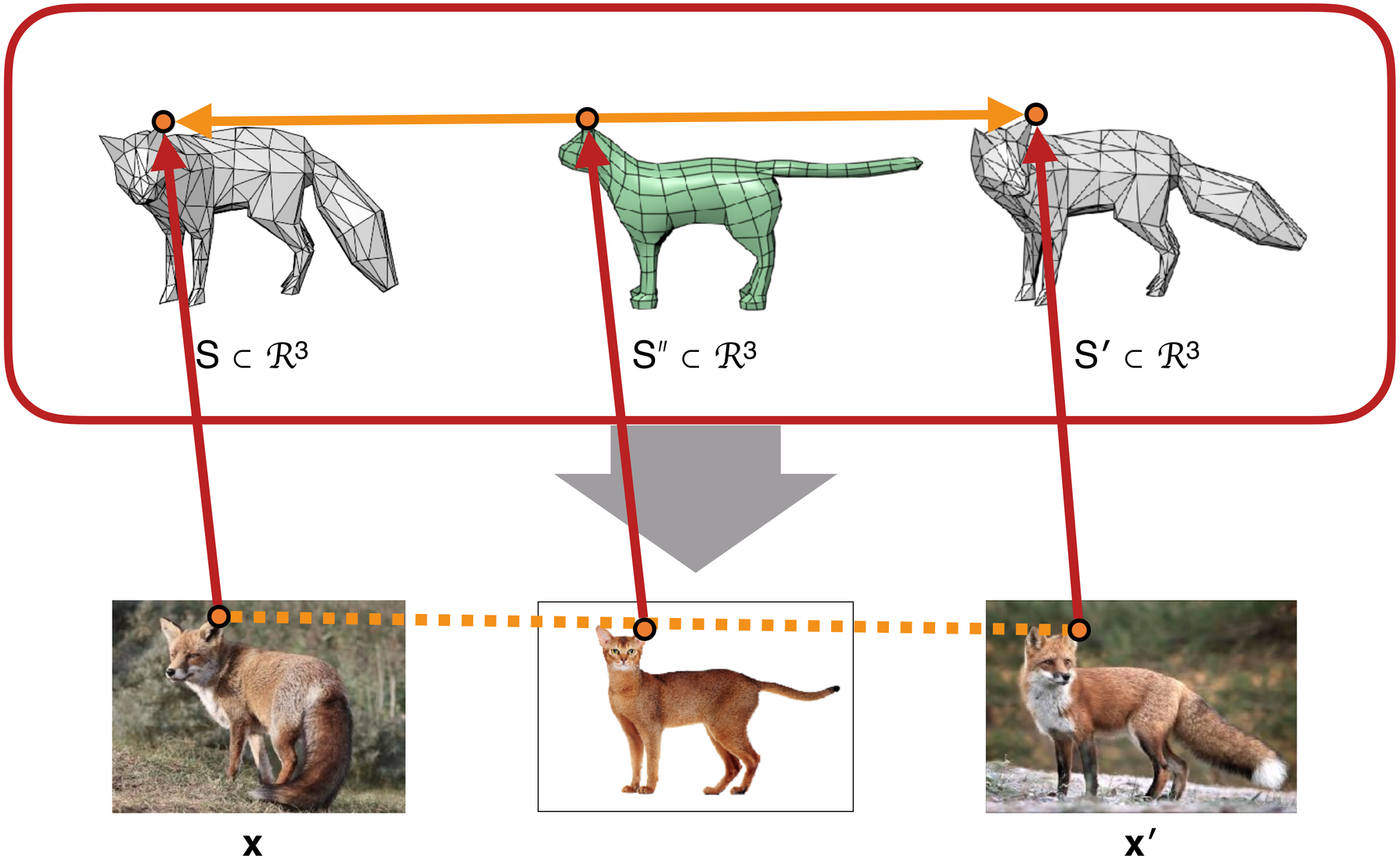
Object model

Take II: an equivalence class of deformable surfaces



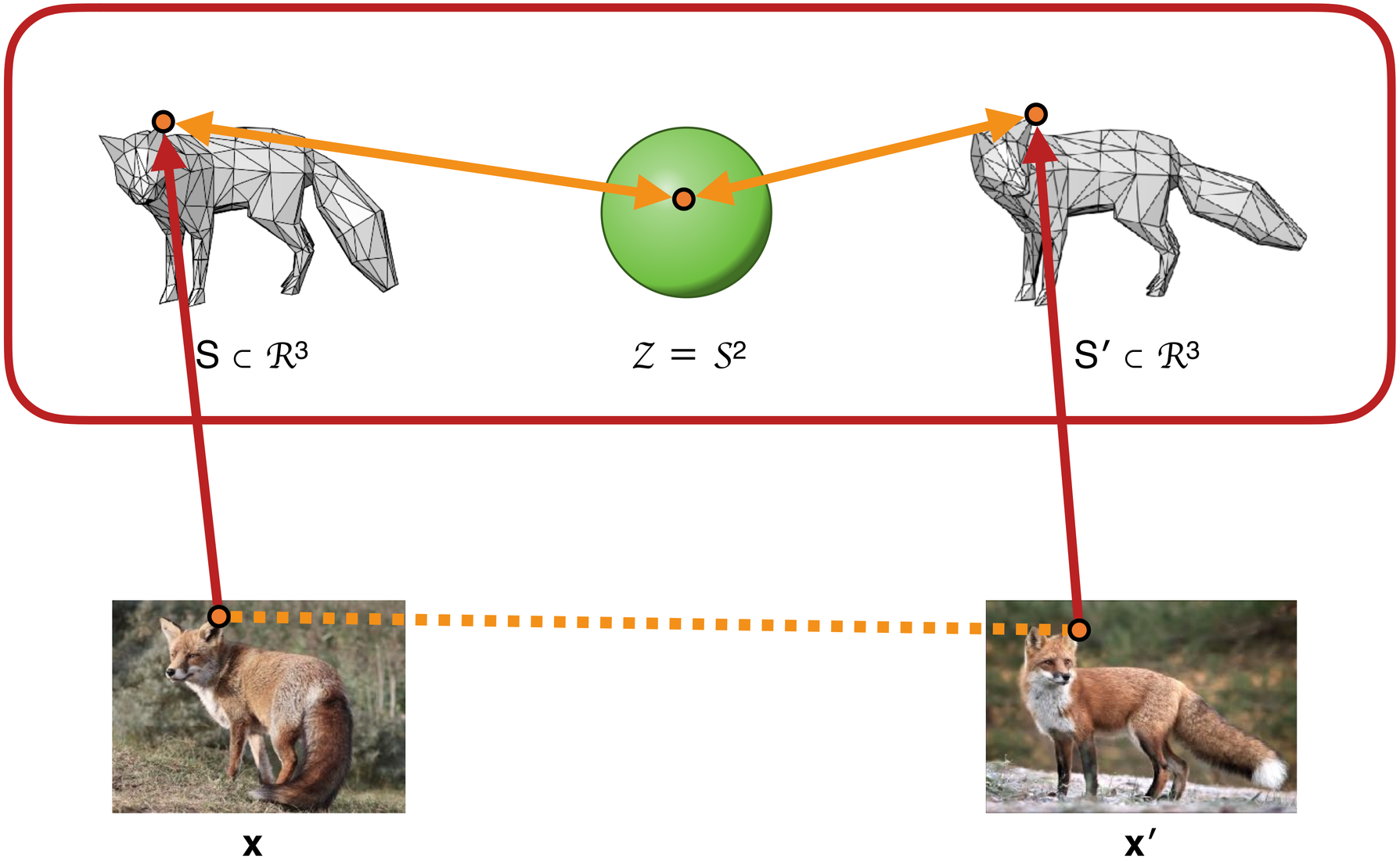
Object model

Can put in **correspondence** different instances, categories



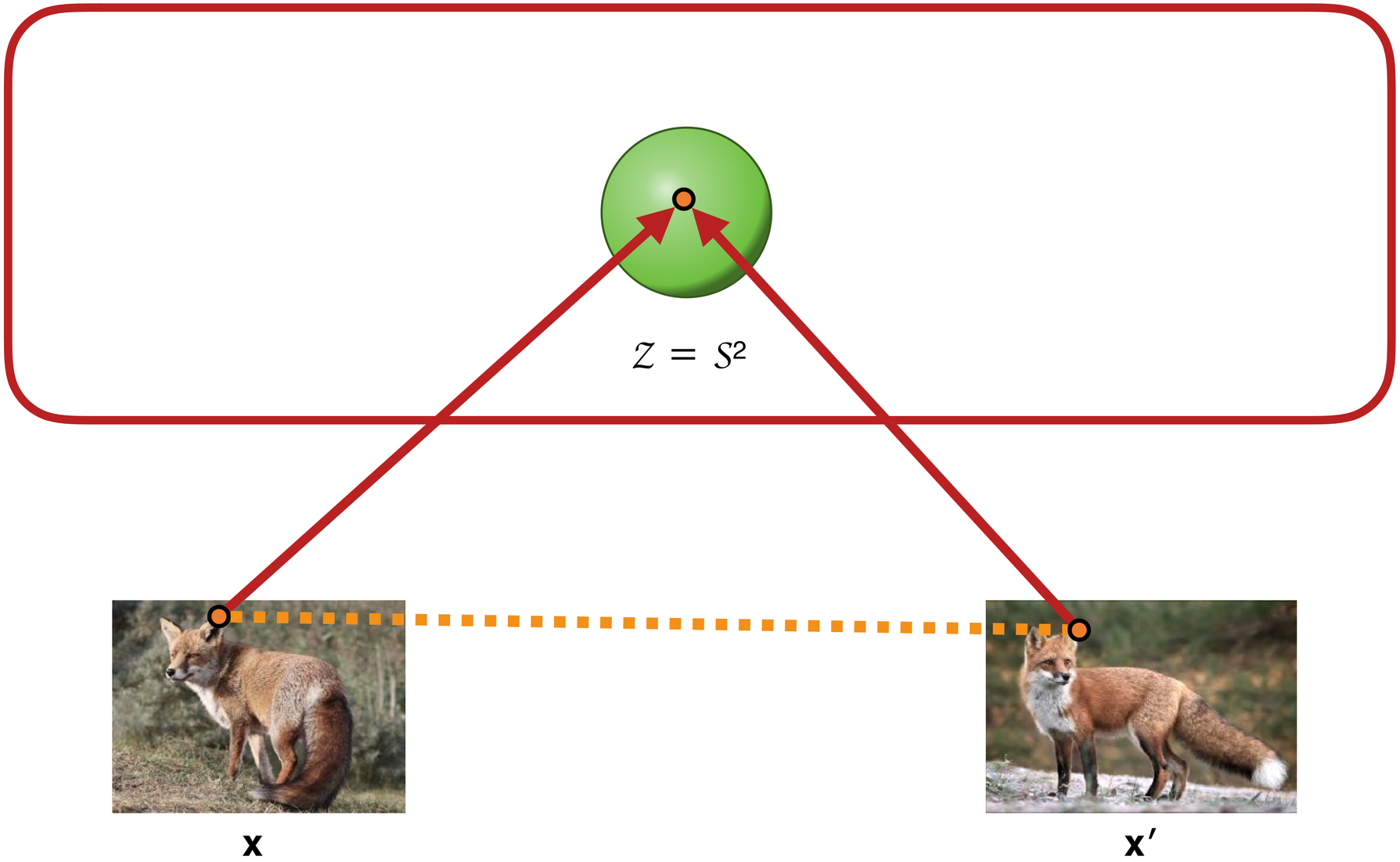
Object model

Homeomorphic to a sphere

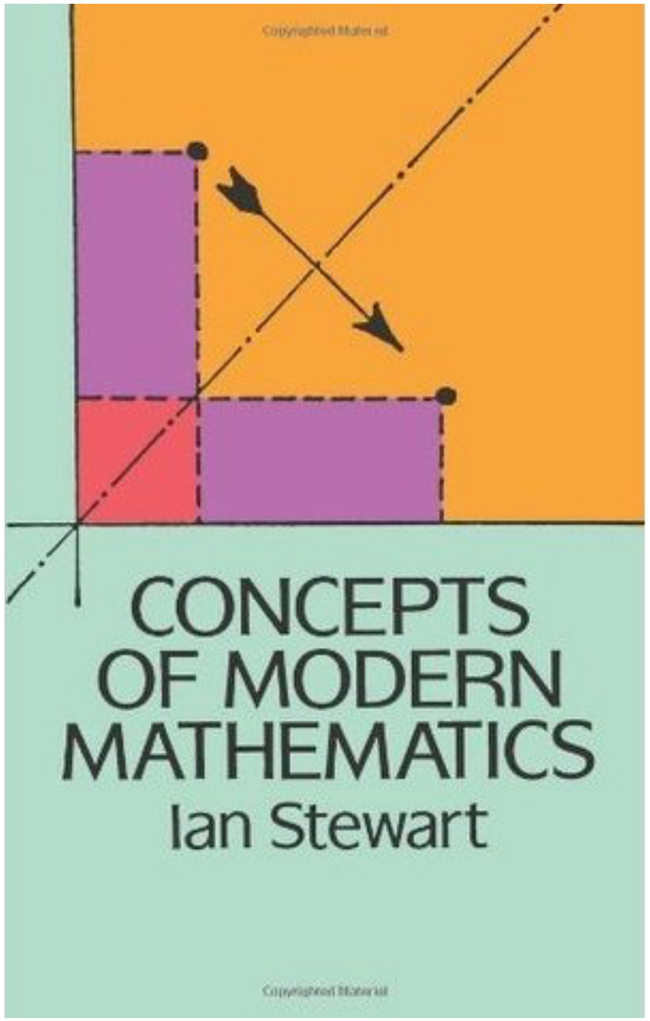


Object model

Take III: Sphere



Mathematical dogs



The Hairy-Ball Theorem

Those are a few of the concepts and objects studied by topology: now we'll look at a theorem.

If you look at the way the hairs lie on a dog, you will find that they have a 'parting' down the dog's back, and another along the stomach. Now topologically a dog is a sphere (assuming it keeps its mouth shut and neglecting internal organs) because all we have to do is shrink its legs and fatten it up a bit (Figure 90).

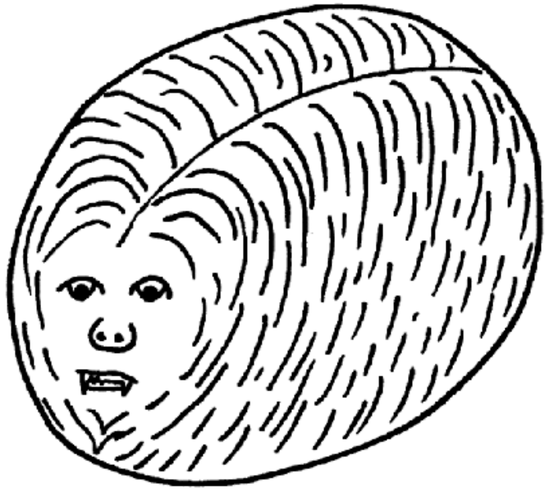


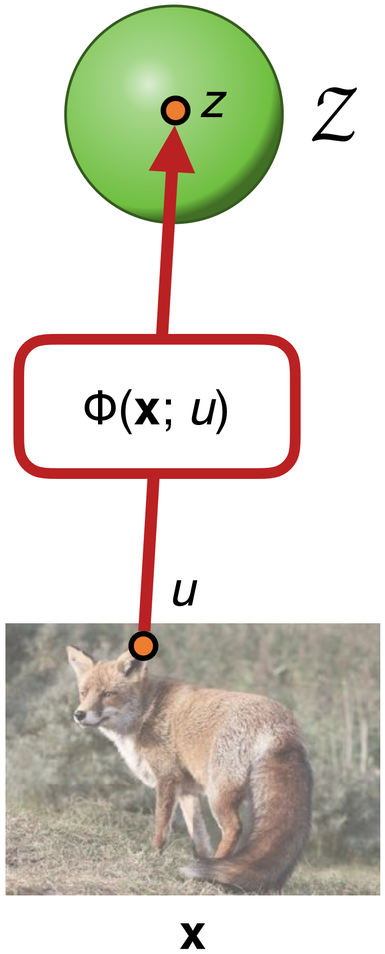
Figure 90

One might wonder whether it is possible to comb the hairs in such a way that all partings were eliminated. This would give a smooth

Object model

Labelling function Φ

Φ maps pixels *u* of image **x** to **object frame coordinates** *z*

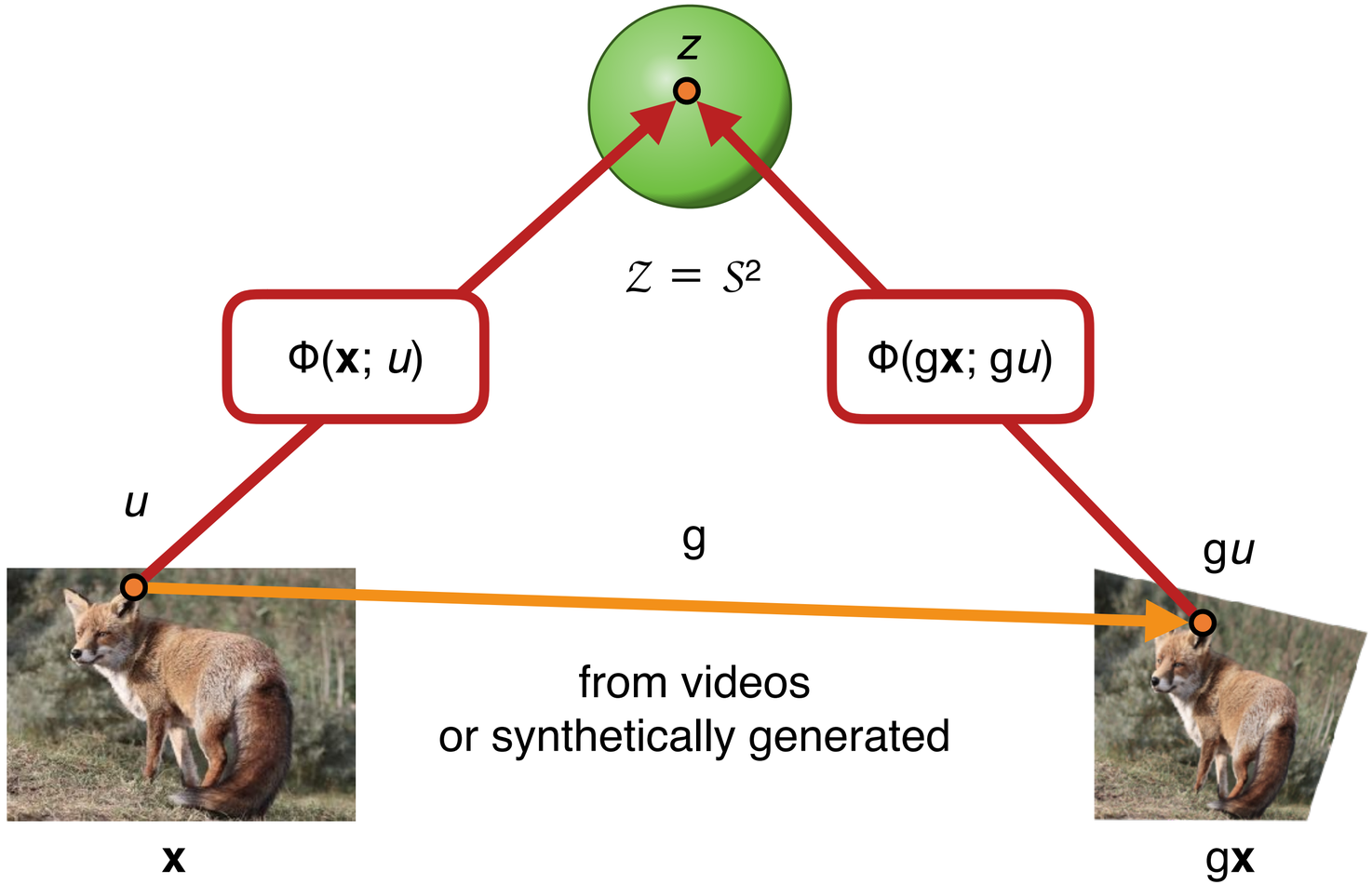


How can we learn this function
without manual supervision?

Invariance

Geometric consistency of the labelling function

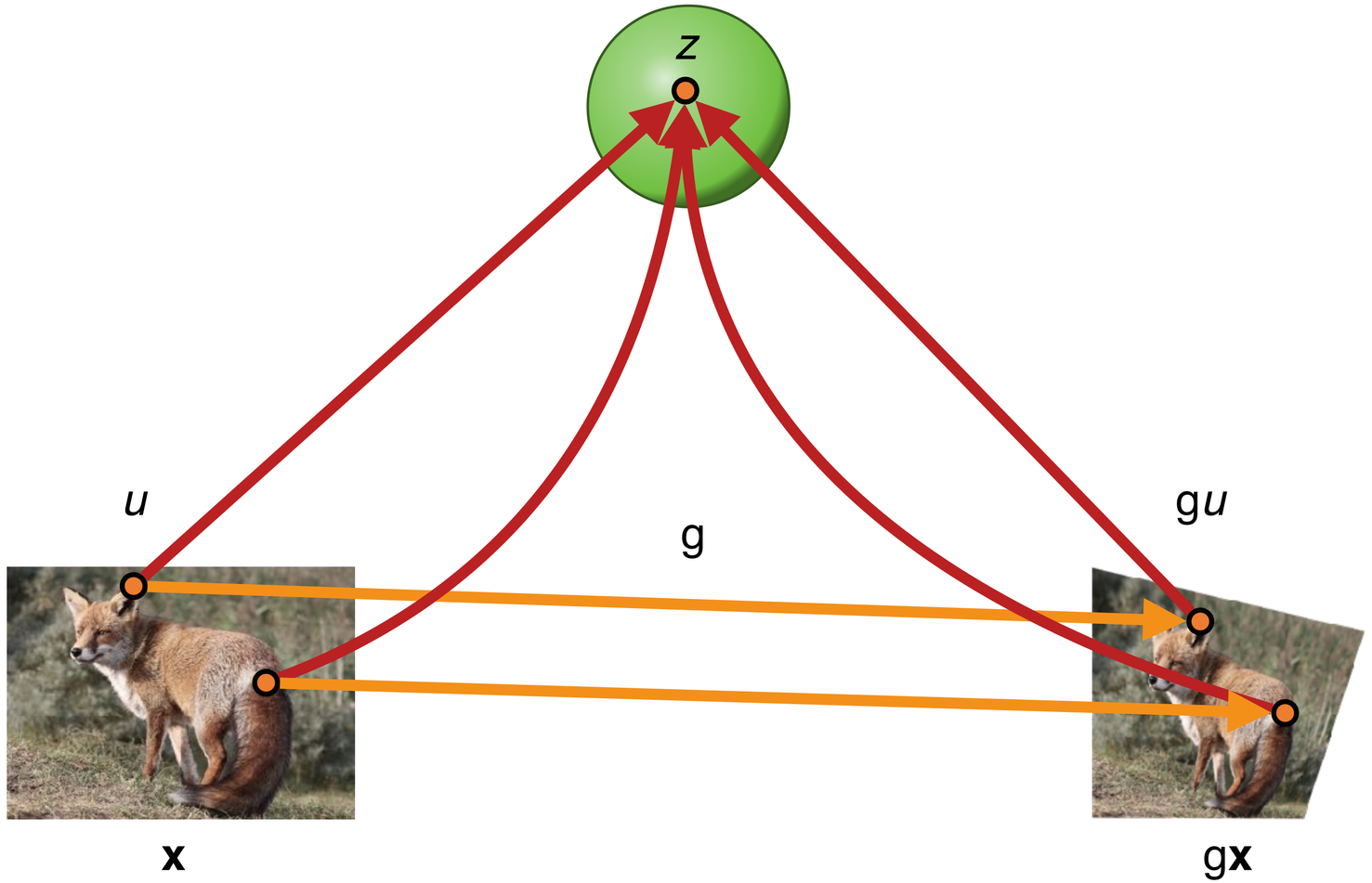
$$\forall u : \Phi(\mathbf{x}; u) = \Phi(g\mathbf{x}; gu)$$



Invariance

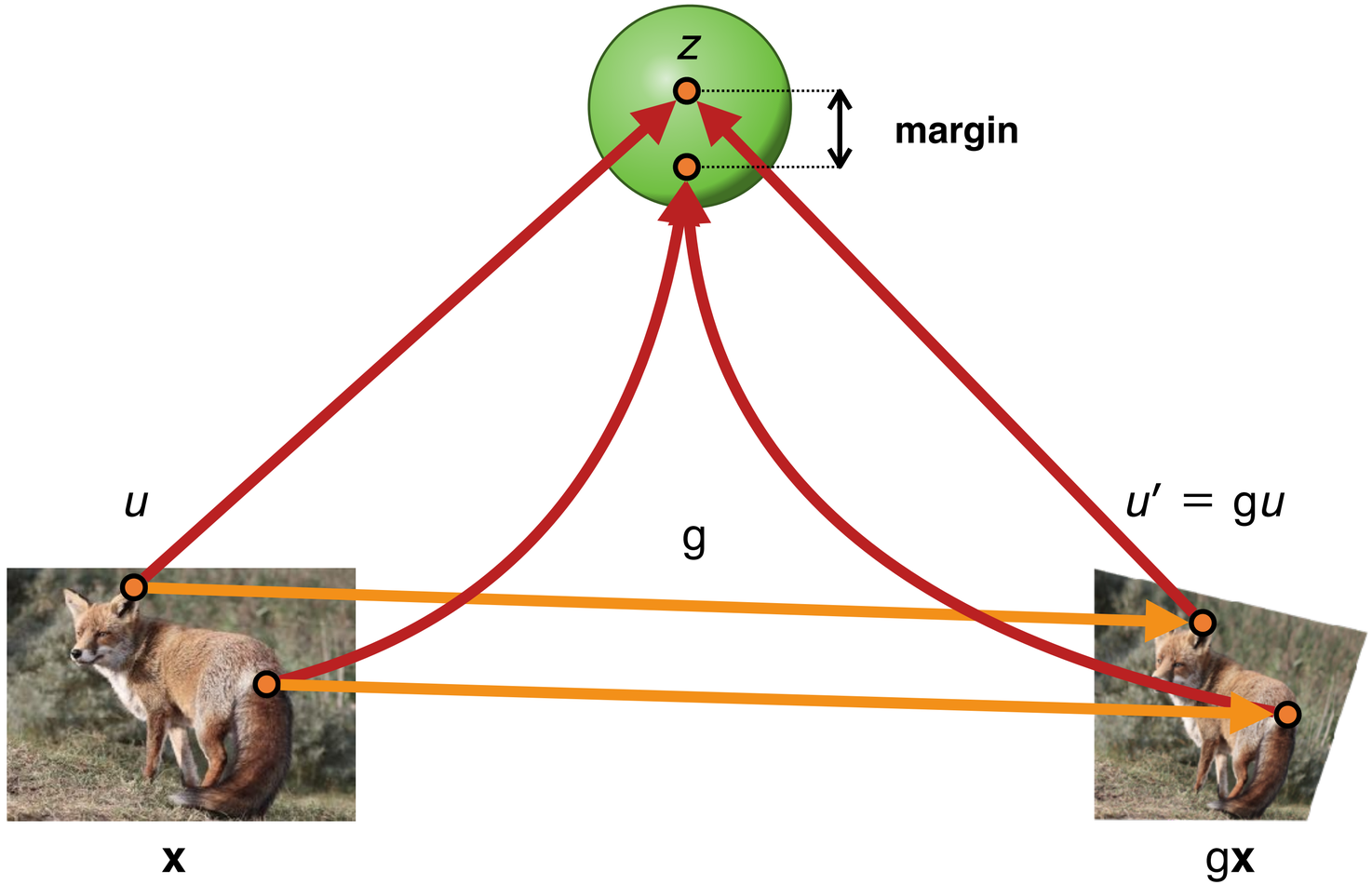
Degeneracy: invariance can be trivially satisfied

$$\forall u: \ \Phi(\mathbf{x}; u) = \Phi(g\mathbf{x}; gu)$$



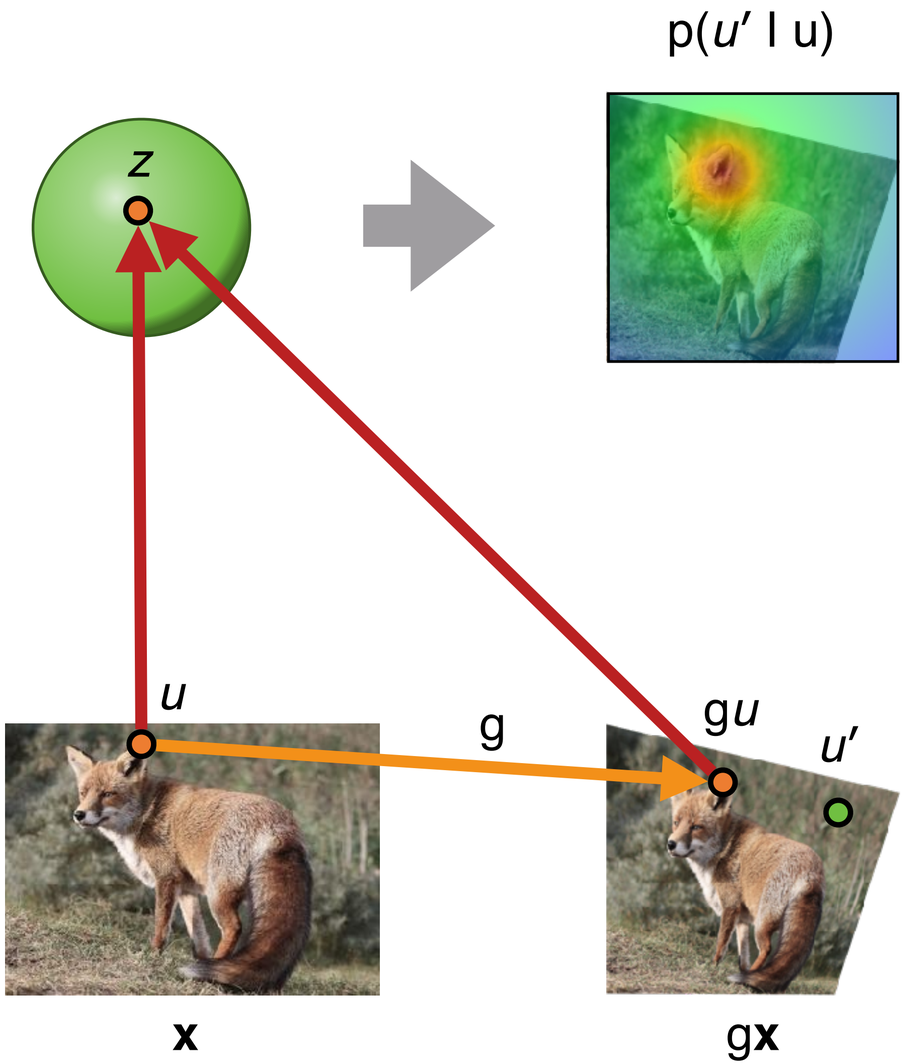
Addressing the degeneracy

$$\forall u, u': \quad [\Phi(\mathbf{x}; u) = \Phi(g\mathbf{x}; u') \Leftrightarrow u' = gu]$$



A probabilistic formulation

Induces invariance and distinctiveness



Map pixels to 3D vectors $\Phi(\mathbf{x};u) \in \mathcal{R}^3$

- Vector length codes **certainty**

Probabilistic vector matching

- Conditional **heat map**:

$$S(u' \mid u) = \langle \Phi(\mathbf{x};u), \Phi(g\mathbf{x};u') \rangle$$

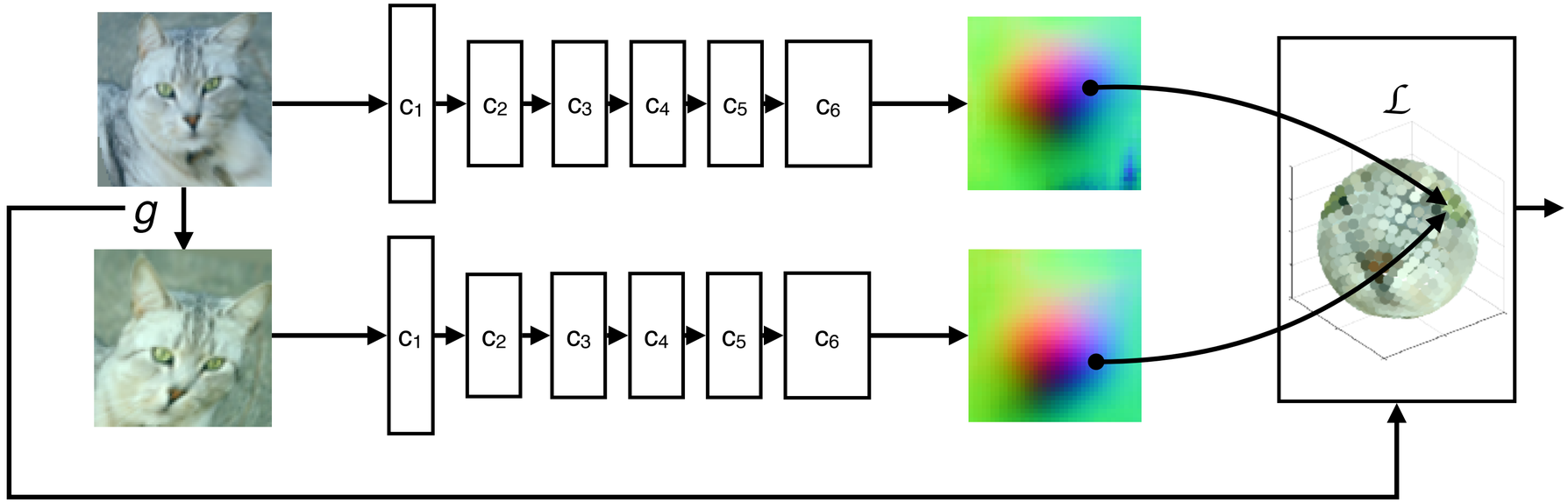
- Conditional correspondence **probability**:

$$p(u'|u) = \frac{e^{S(u'|u)}}{\int e^{S(u'|u)}\, du'}$$

Loss encourages both **precision** and **accuracy**:

$$\mathcal{L} = \int \|u' - gu\|^{\gamma} p(u'|u)\, du\, du'$$

Unsupervised learning of a dense equivariant labeller



$$\min_{\Psi} \frac{1}{n} \sum_{i=1}^{n} \int \|u' - g_i u\|^{\gamma} \frac{e^{\langle \Psi(\mathbf{x}_i;u), \Psi(g_i \mathbf{x}_i;u') \rangle}}{\int e^{\langle \Psi(\mathbf{x}_i;u), \Psi(g_i \mathbf{x}_i;v) \rangle} \, dv} \, du$$

Data: ($\mathbf{x}_i$, g_i) where $\mathbf{x}_i$ is a **random image** and g_i the estimated **flow** or a **random warp**

Examples & results

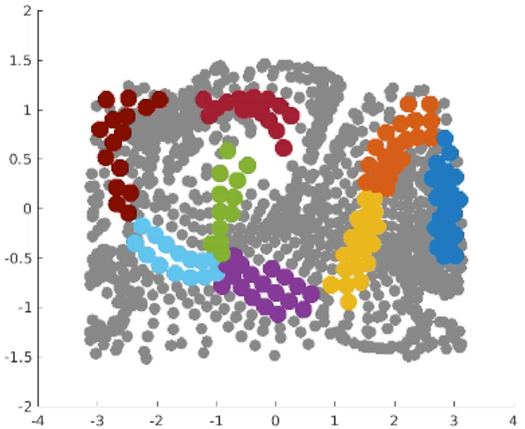
Learning an object from motion

Training data

- Pairs of **video frames** $\mathbf{x}_t$, and $\mathbf{x}_{t+\Delta}$
- g is the **optical flow**

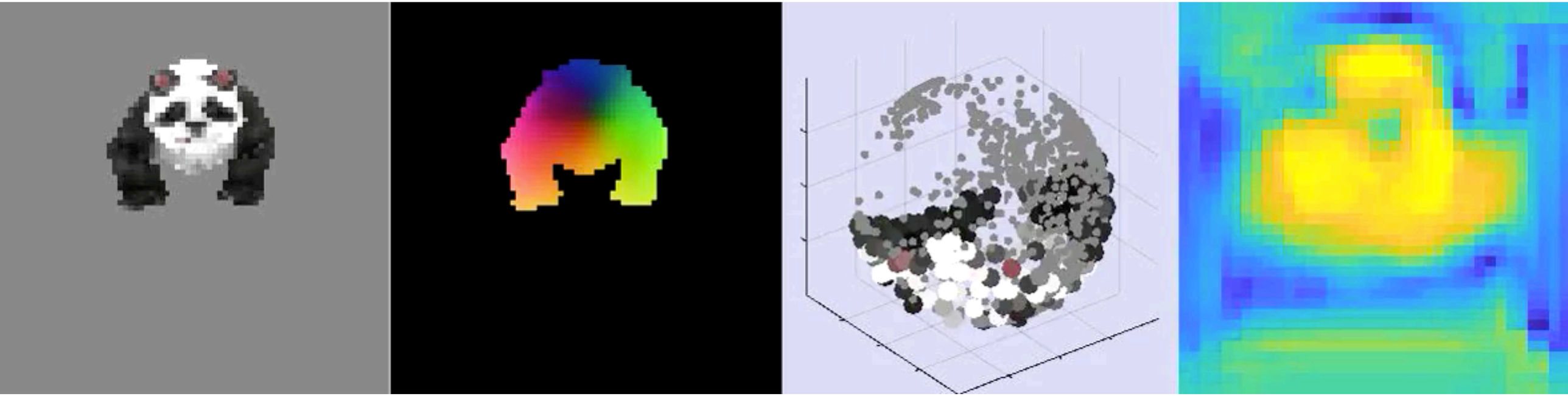
Learned embedding

- Map object points to a fixed reference coordinates
- The absolute reference is learned automatically



3D deformable object

Input Frames | Labelled Pixels | Projected to sphere | Vector Magnitude

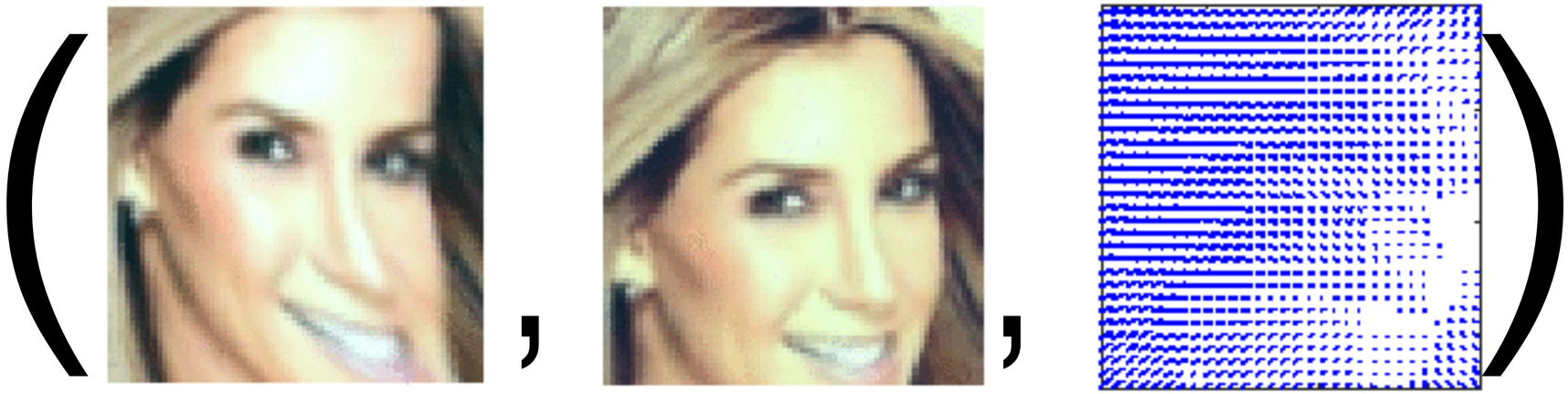


Faces

Training data

Pairs of **warped images** **x** and **x'** from a dataset of ~200K celebrity images

g are a **synthetic warps**

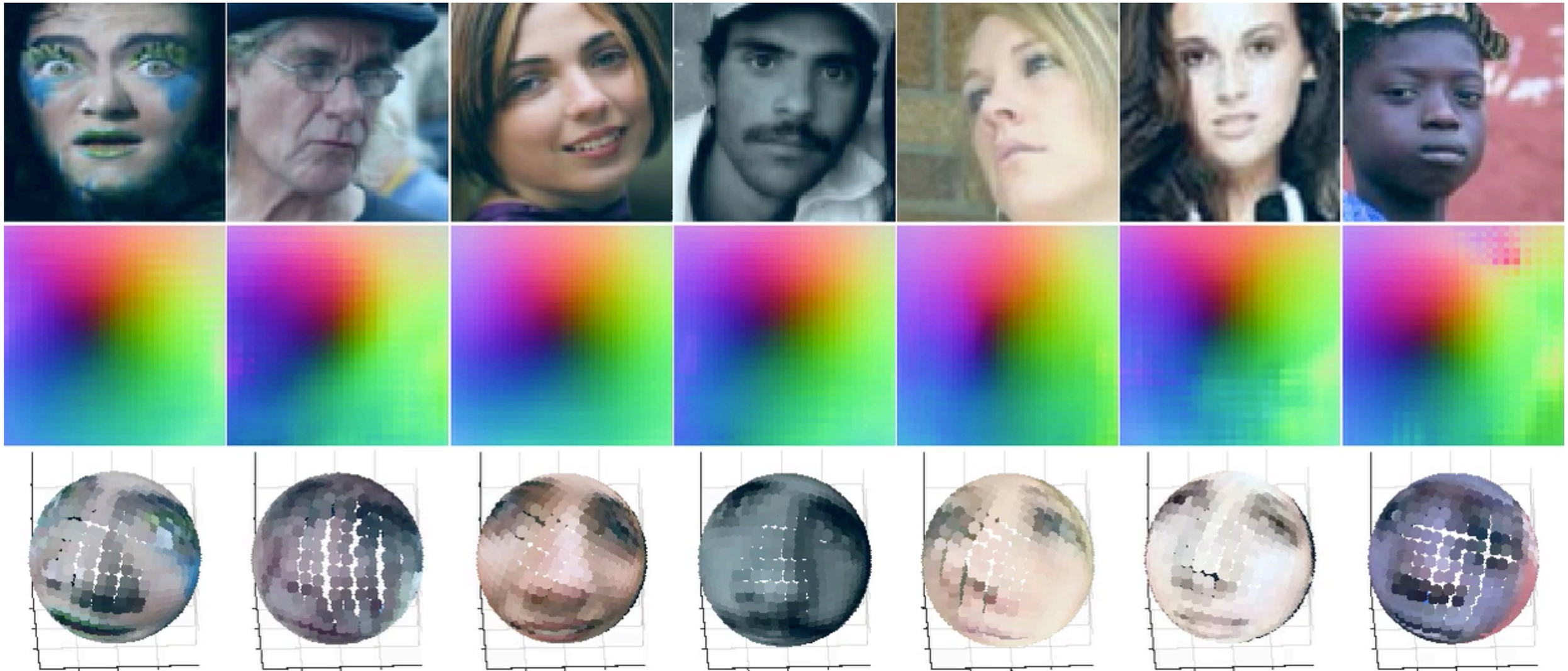


$\mathbf{x}_t$ $\mathbf{x}_{t+1}$ g

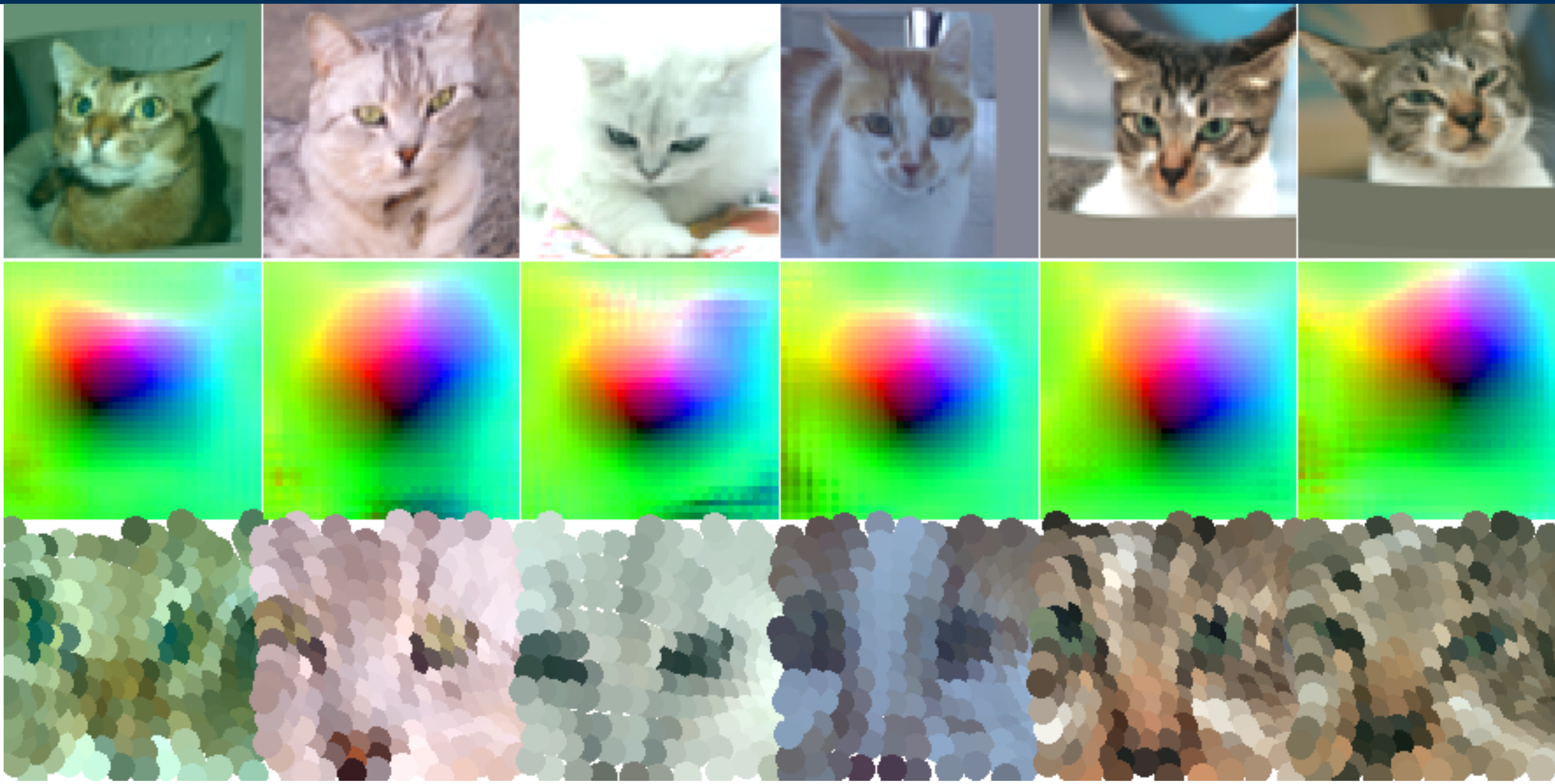
Faces

Learned embedding

Different face **instances** are **automatically aligned**

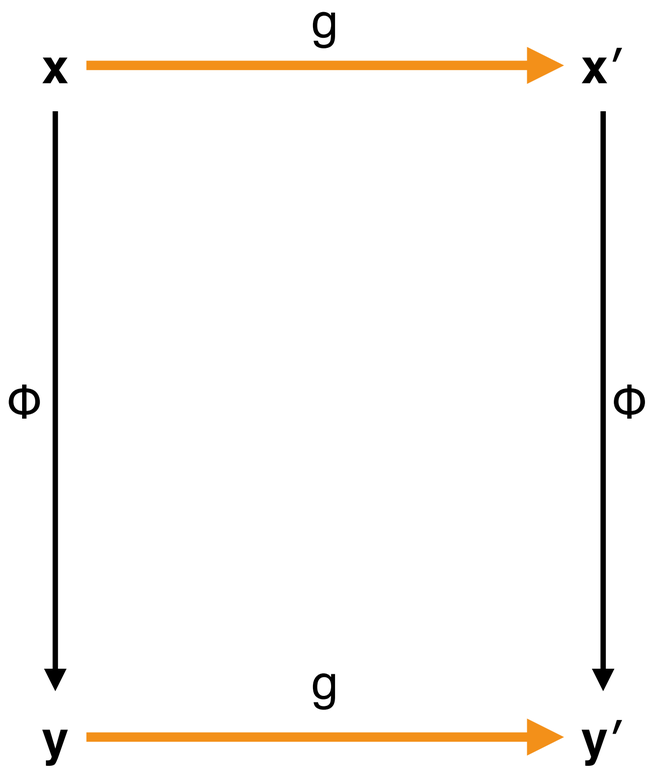


Cat faces



Equivariance revisited

$$\Phi(g\,\mathbf{x}) = g\,\Phi(\mathbf{x})$$

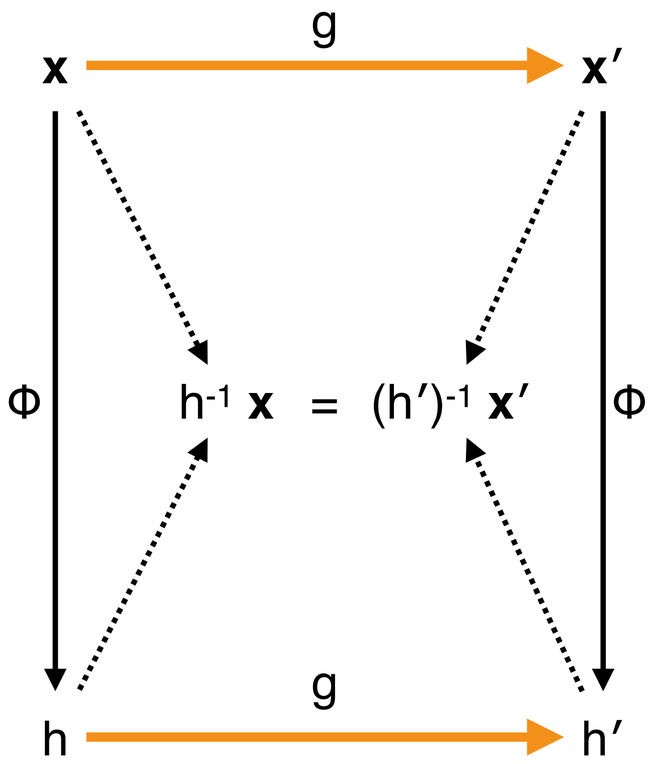


Key ideas

1. **Equivariance** learns a **data representation** from a **relationship** (**x**, **x′**, g) between data pairs
2. It works by trying to map the data into a **simpler representation** **y** where the relationship still holds true

Auto-equivariance, factorization, and normalization

$$\Phi(g\,\mathbf{x}) \circ \Phi(\mathbf{x})^{-1} = g$$



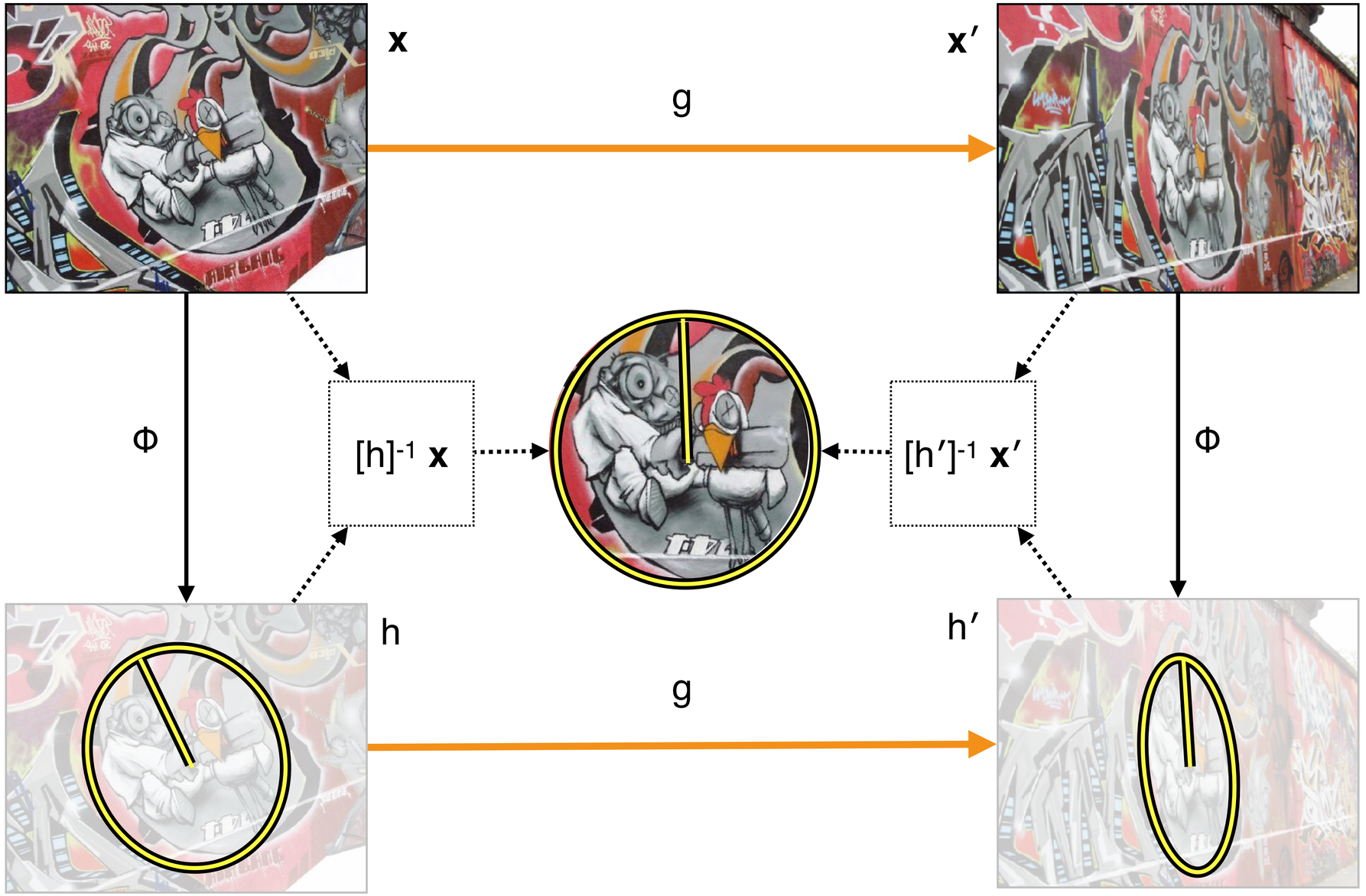
A new assumption

The representation space Y coincides with the transformation group G

Consequences

1. We can rewrite the constraint as a **factorization**
2. The factors provide an **absolute reference frame** for the data, *normalizing* it

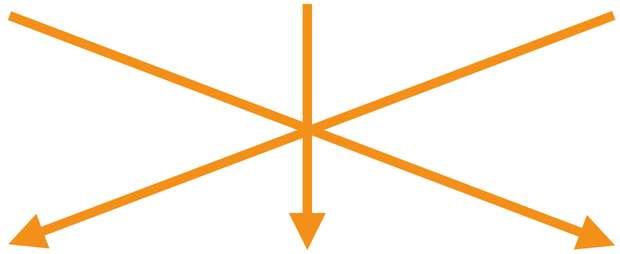
Learning co-variant feature detectors



Learning covariant feature detectors, Lenc Vedaldi, ECCW 2016

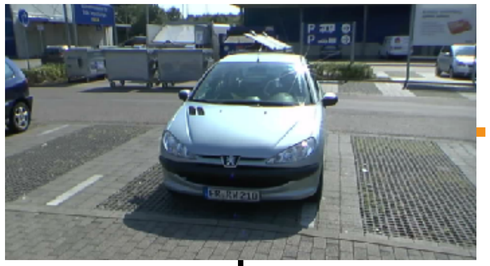
Aligning 3D object categories

Input video 1



Input video 2

Viewpoint factorization network

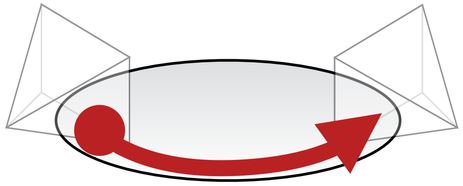


x

g

x′

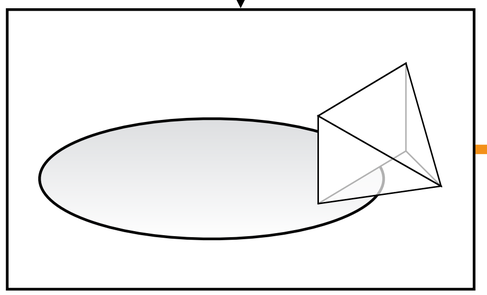
Relative viewpoint
from SFM



Absolute viewpoint
from CNN

Φ

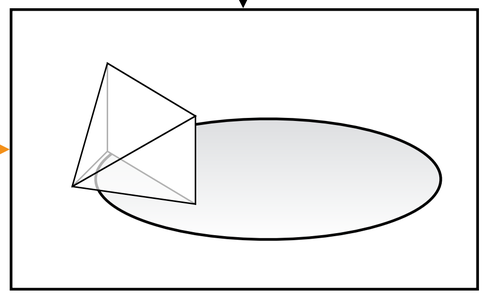
Φ



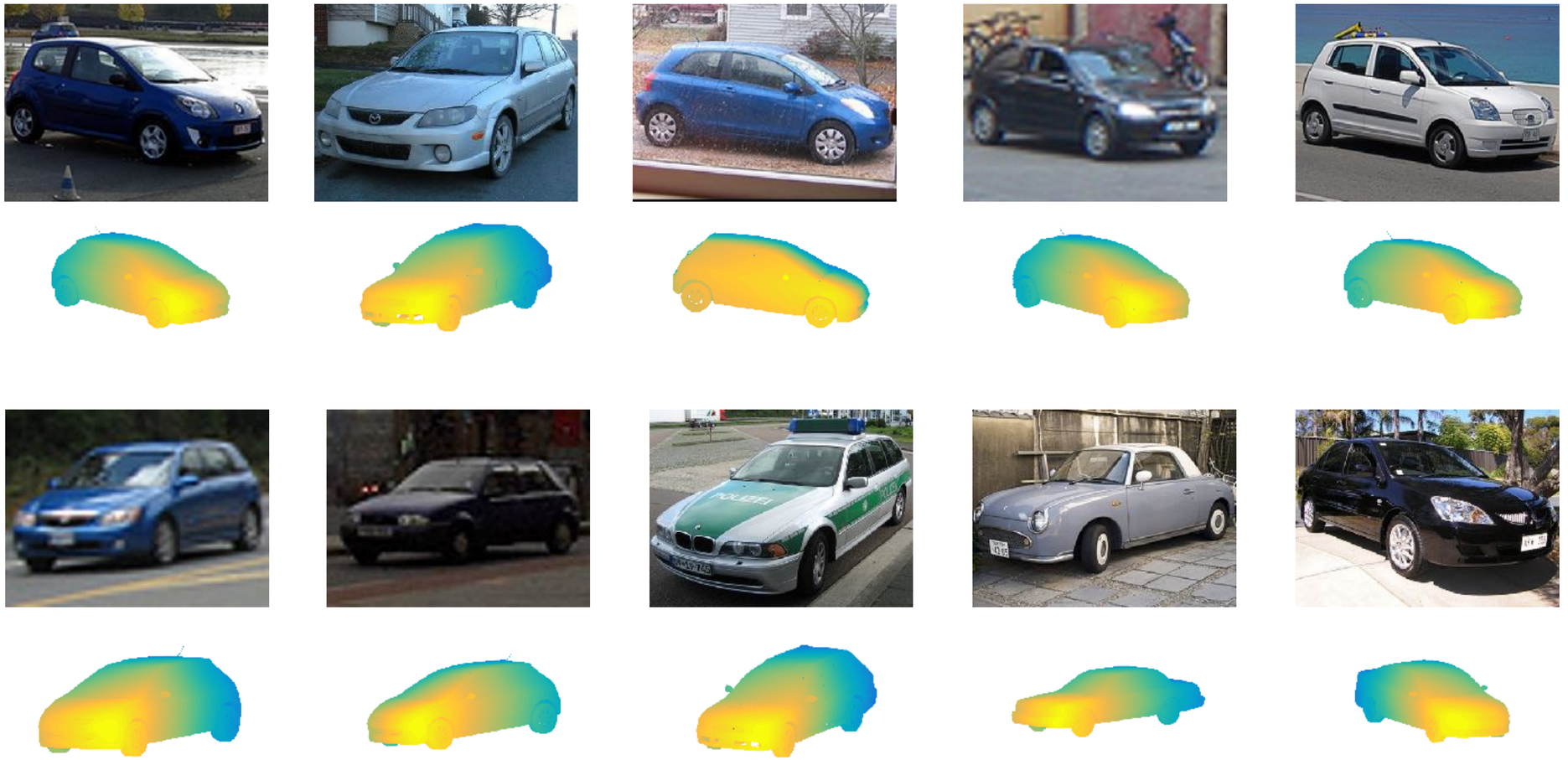
h

g

h′

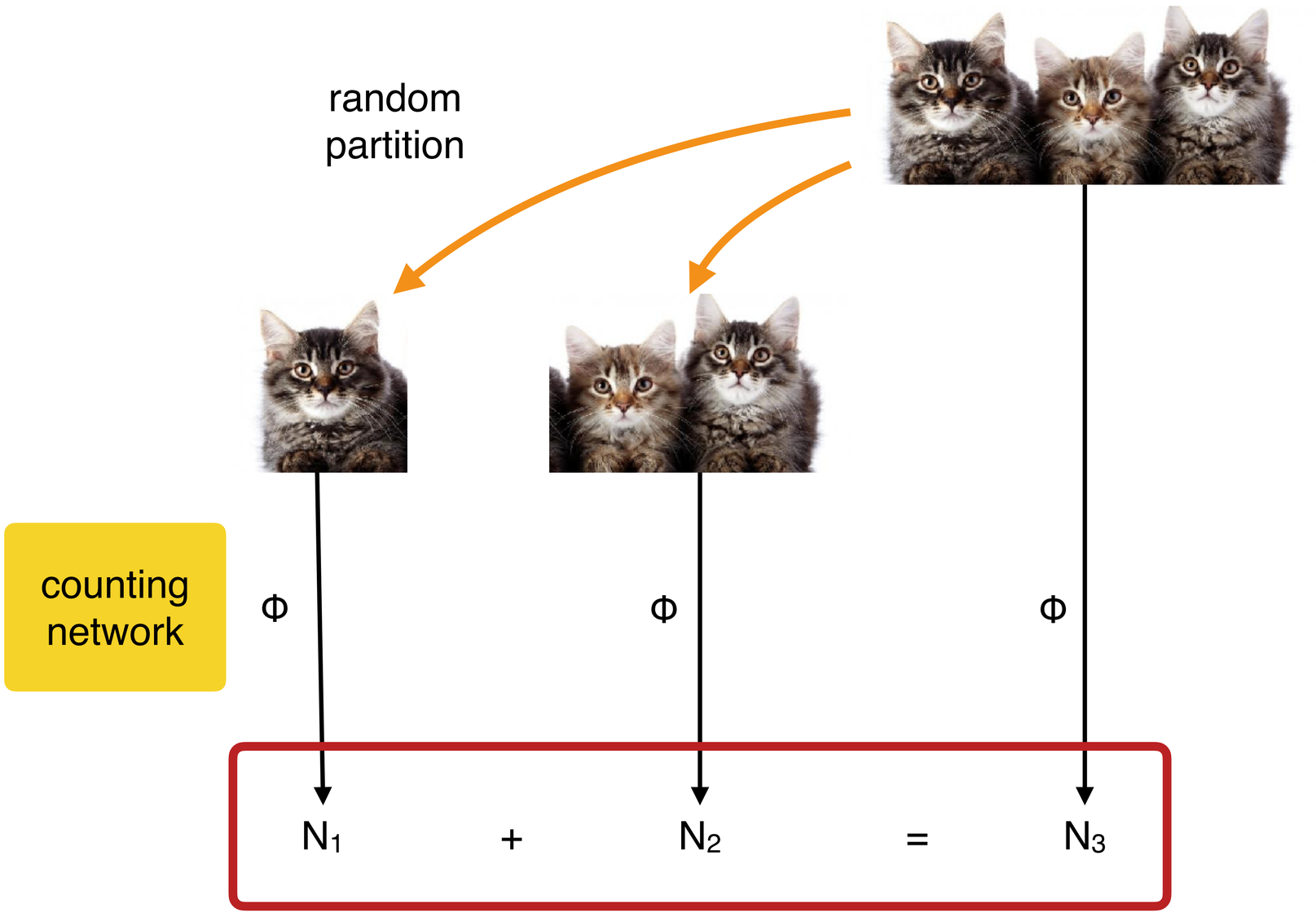


Learned absolute viewpoint



Learning 3D object categories by looking around them. Novotny, Larlus, Vedaldi, ICCV, 2017

Learning to count



Train a network to count the number of “objects” within an image

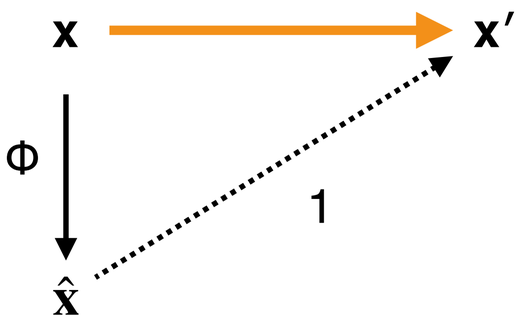
Key idea: the count of the full image must be the sum of the counts of the subparts

Three-ways factorization

Representation learning by learning to count. Noroozi, Piriavash, Favaro. Proc. ICCV, 2017.

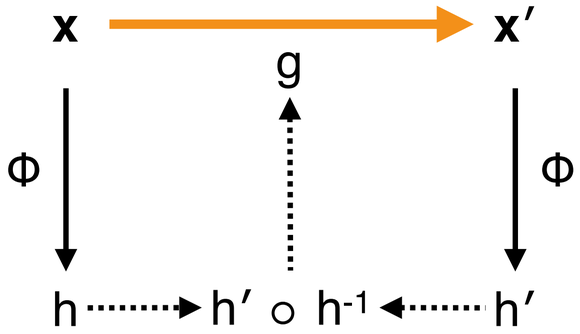
Unsupervised signals

Pretext tasks



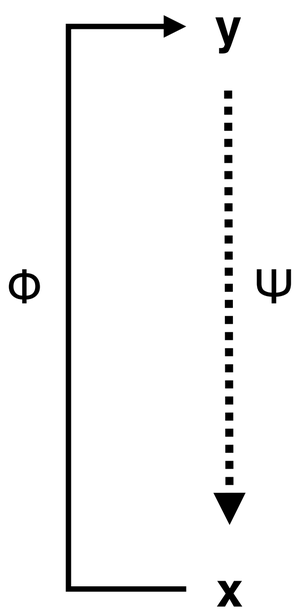
- Perturbations
- Motion
- Modalities

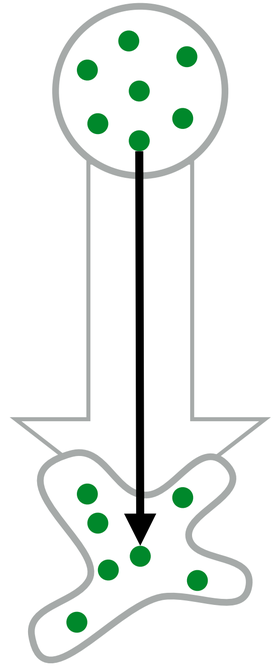
Equivariance



- Equivariance

Generative modeling





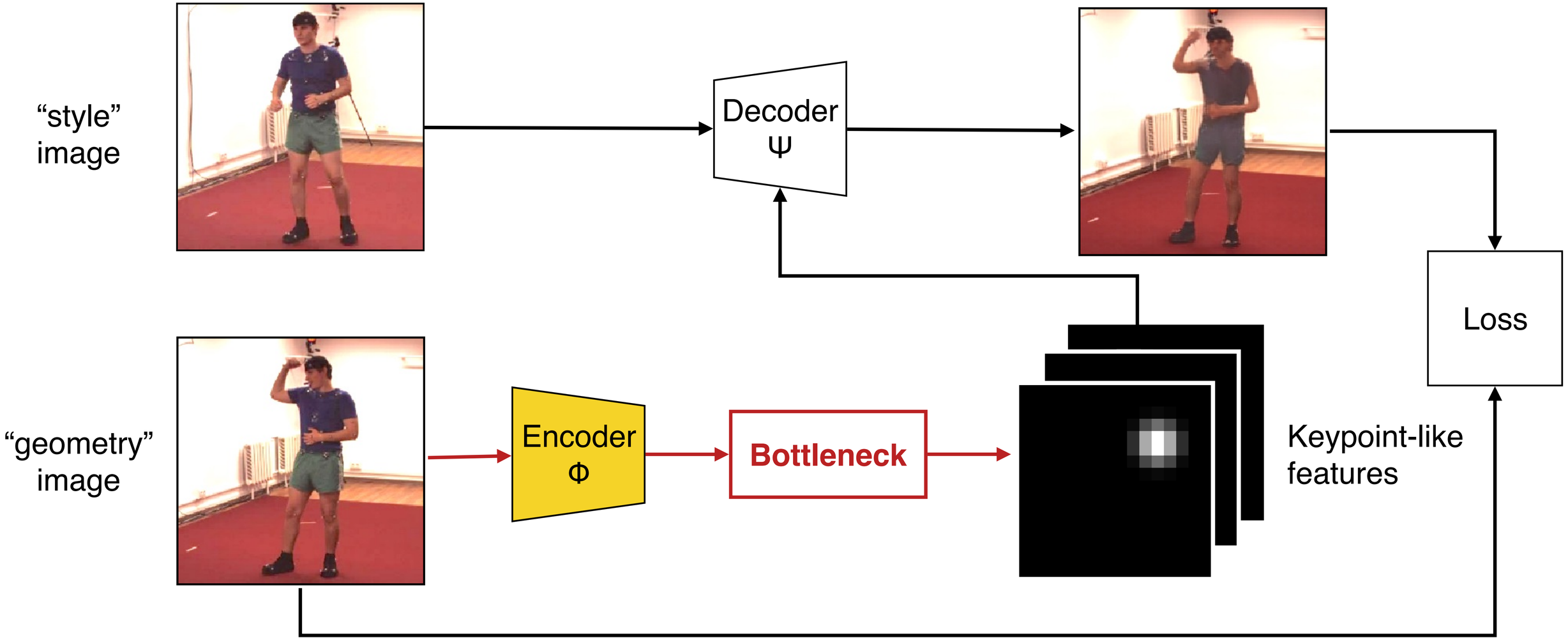
fixed, simple distribution

empirical distribution

- Autoencoders
- VAE
- GAN
- ...

Future frame prediction revisited

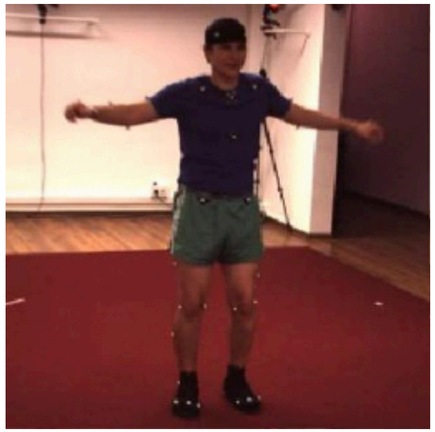
Conditional generation, using structure (keypoints) as bottleneck

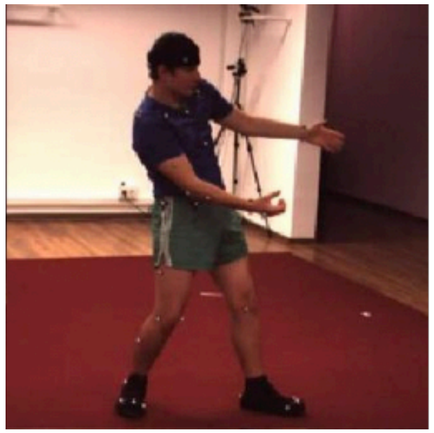


Unsupervised discovery of object landmarks as structural representations. Z., G., J., L., H., L. Proc. CVPR, 2018.
Conditional Image Generation for Learning the Structure of Visual Objects, Jakab, Gupta, Bilen, Vedaldi, arXiv, 2018.

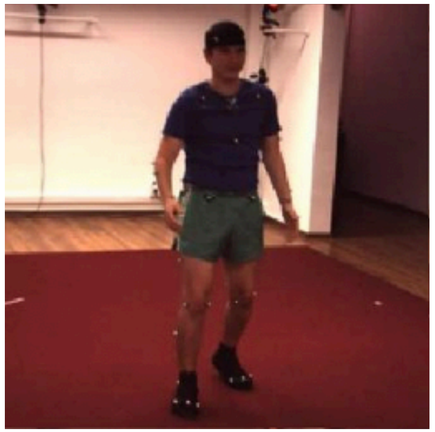
Self-supervised from video streams

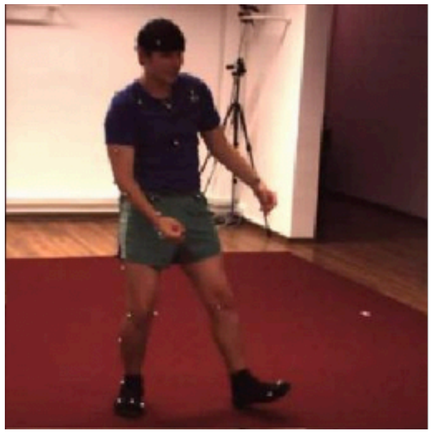
Desired style



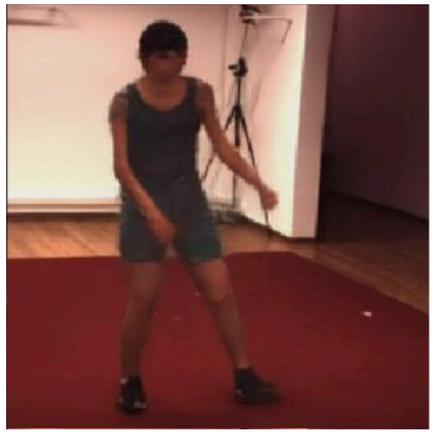


Desired geometry

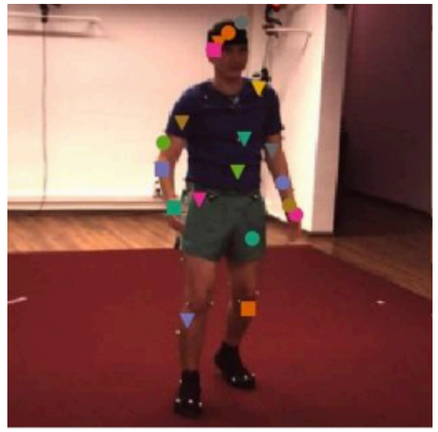


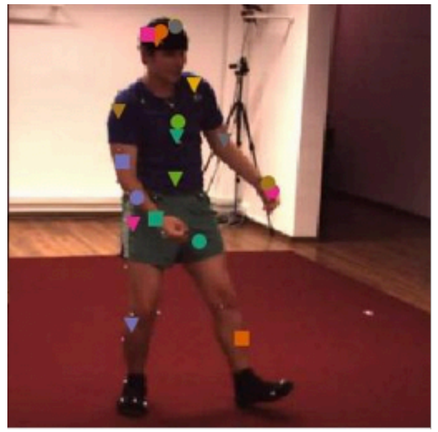


Generated



Keypoints

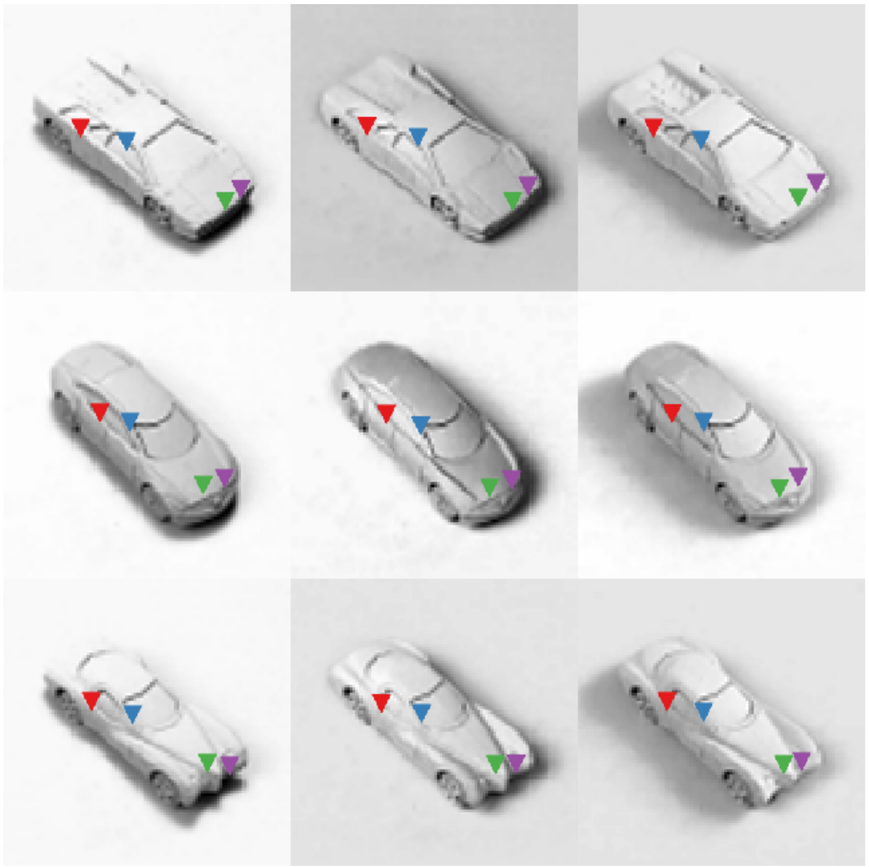




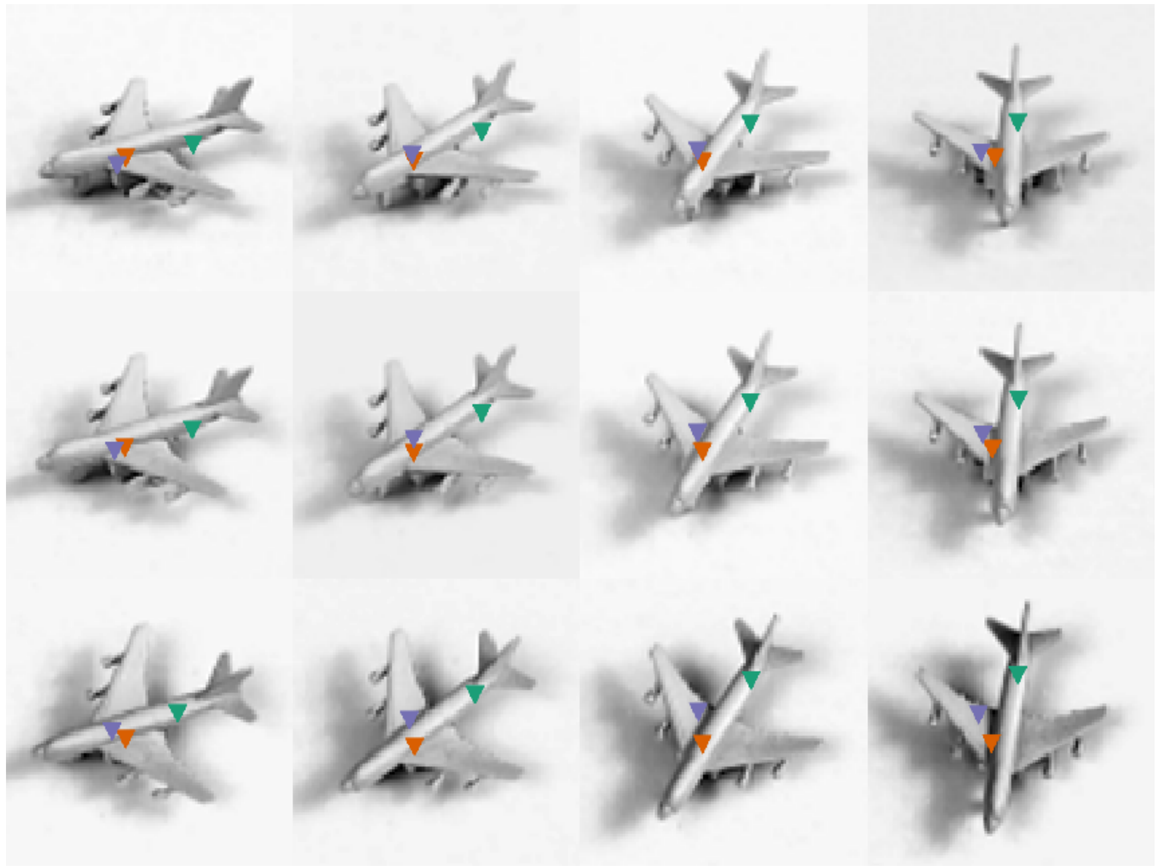
Conditional Image Generation for Learning the Structure of Visual Objects, Jakab, Gupta, Bilen, Vedaldi, arXiv, 2018.

Self-supervised from pairs of random viewpoints

Shape invariance



Viewpoint invariance



Conditional Image Generation for Learning the Structure of Visual Objects, Jakab, Gupta, Bilen, Vedaldi, arXiv, 2018.

Disentangling

The model clearly separates appearance from geometry

Geometry

Style

Generated

(Note: image generation is **not** the goal here!)

Conditional Image Generation for Learning the Structure of Visual Objects, Jakab, Gupta, Bilen, Vedaldi, arXiv, 2018.

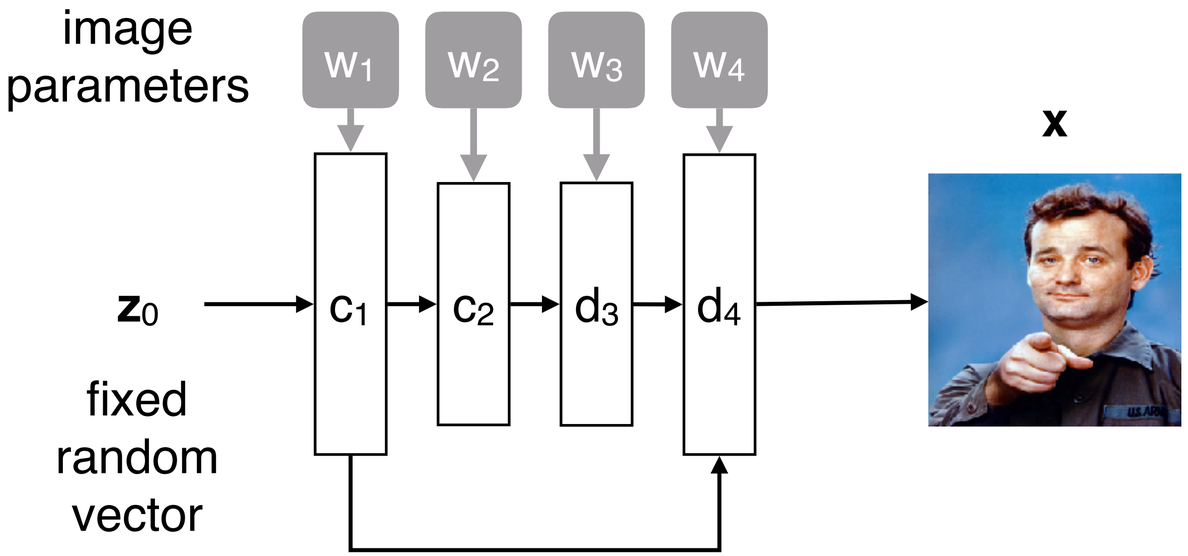
Self-supervised features

Self-supervised structure

Deep image prior

No supervision!

A priori information is contained in the structure of the CNN



The network provides a **parametrization** of **images**:

$$\mathbf{x} = \Psi(\mathbf{w}; \mathbf{z}_0)$$

$$\mathbf{w} \longrightarrow \mathbf{x}$$

w is **not learned**
but used as a **free image parameter**

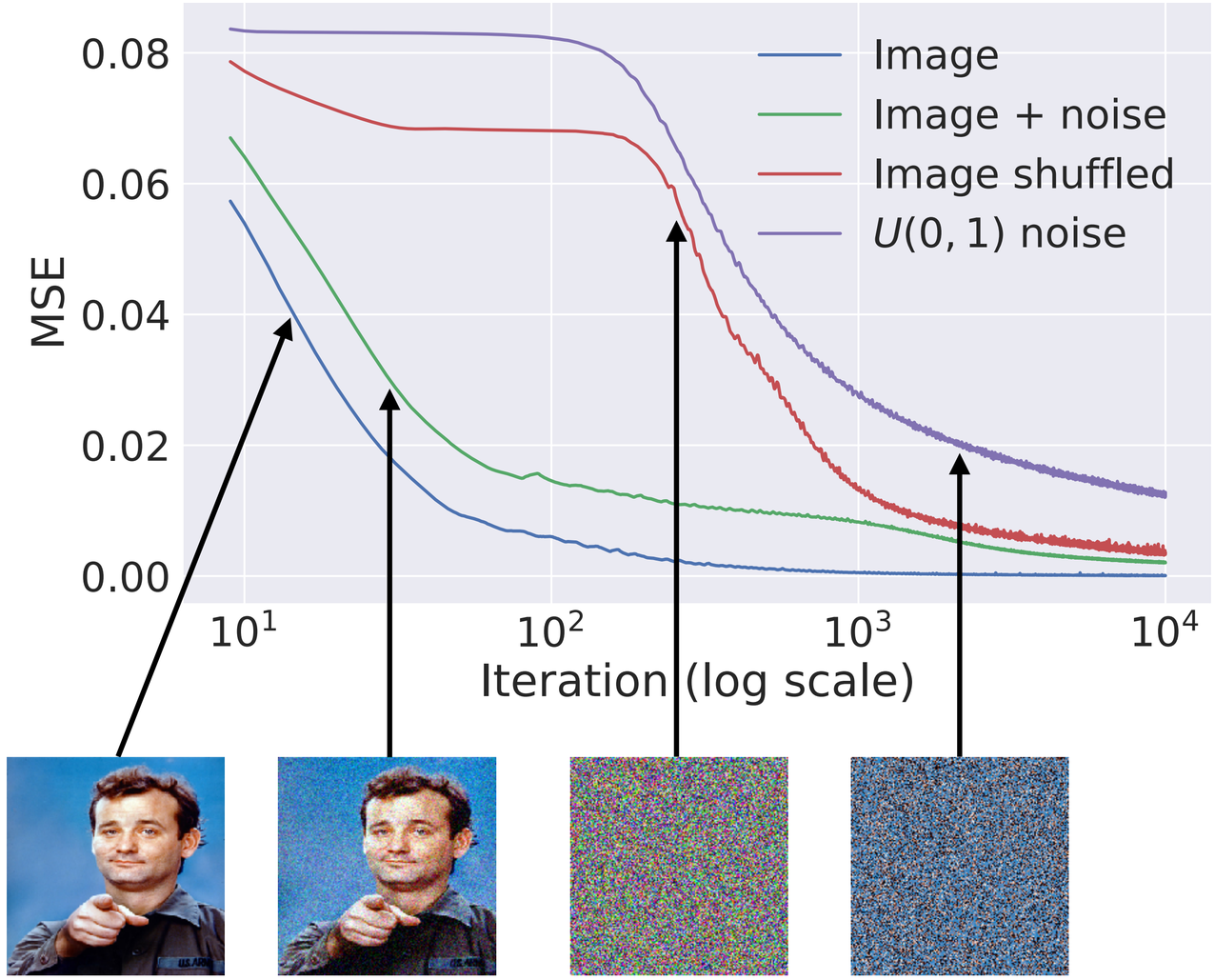
Deep image prior

A parameterization that offers high-impedance to noise

Find the **w** net parameters that reconstruct a target image **x:**

$$\min_{\mathbf{w}} \|\mathbf{x} - \Phi(\mathbf{w})\|^2$$

The **convergence speed** is proportional to how "**natural**" the image looks



Deep image prior, Ulyanov Lempitsky Vedaldi, CVPR 2018

Deep image prior: inpainting

For **inpainting** we only reconstruct the visible pixels, implicitly infer the others

$$\min_{\mathbf{w}} \|\mathbf{m} \odot (\mathbf{x} - \Phi(\mathbf{w}))\|^2$$

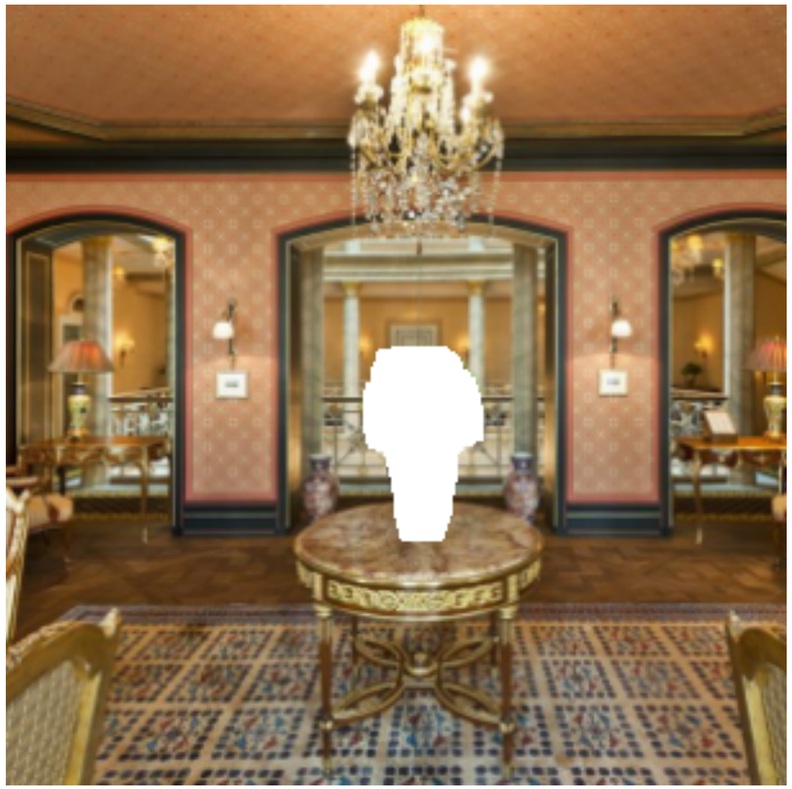
Conv. coding
Papyan et al. 2017

Deep Image Prior

Inpainting

Global/local completions
Izuka et al. 2017

Deep Image Prior



Deep image prior: inpainting

Masked image

Depth 2

Depth 4

ResNet-style network

UNet-style network

Deep image prior: inpainting

Deep-image prior completion

Three key challenges of deep learning

Universal Representations

Fewer models to train

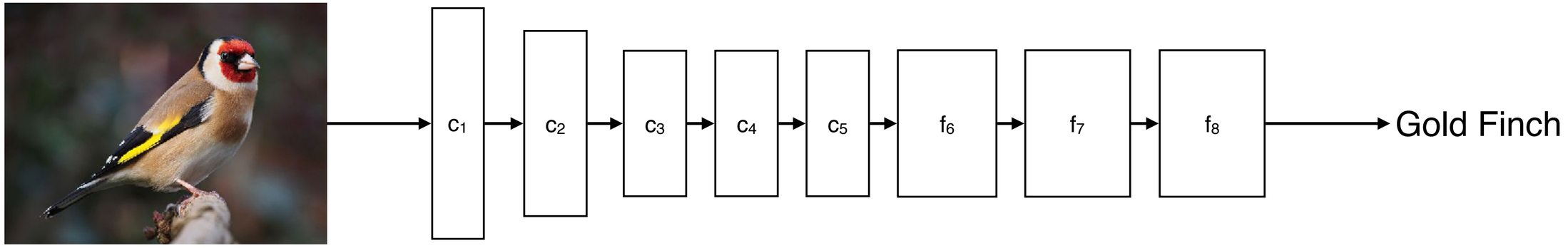
Unsupervised Representations

Less effort to train new models

Understandable Representations

Trust, safety, and usability

Peeking inside the black box



What does a net **do**?

- What concepts can it recognise?
- Spurious correlations?
- Limitations?

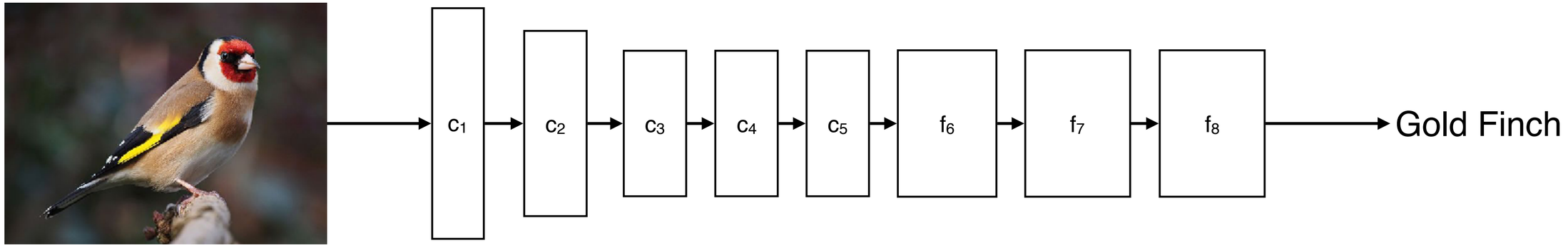
How does it **do** it?

- Template matching?
- Compositionality?
- Spatial reasoning?

How does it **learn** it?

- Generalization?
- Optimisation?

Peeking inside the black box

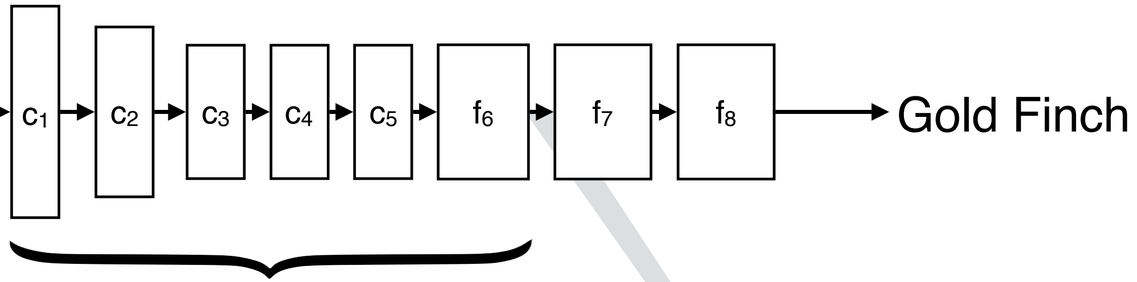


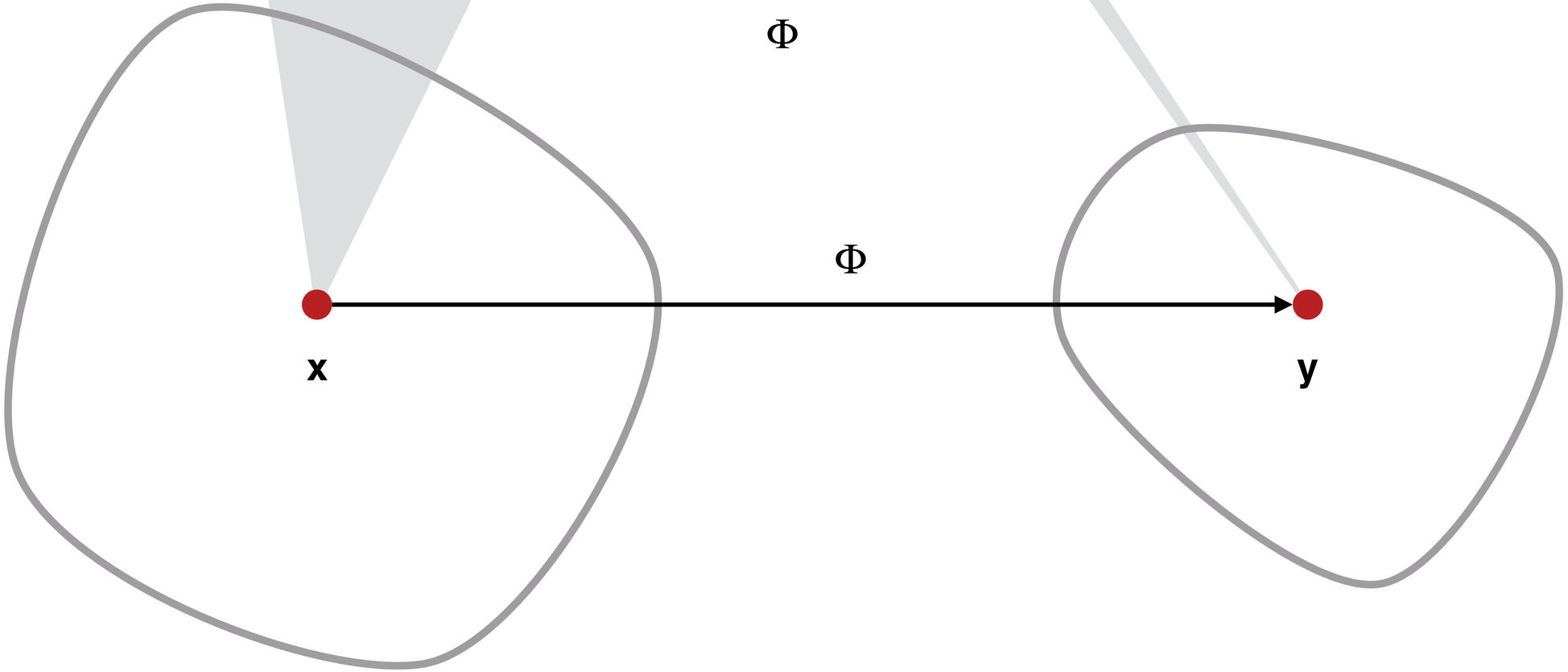
What does a net **do**?

- What concepts can it recognise?
- Spurious correlations?
- Limitations?

Deep networks as encoders

Each subnetwork Φ maps an image x to a code y





Deep networks as encoders

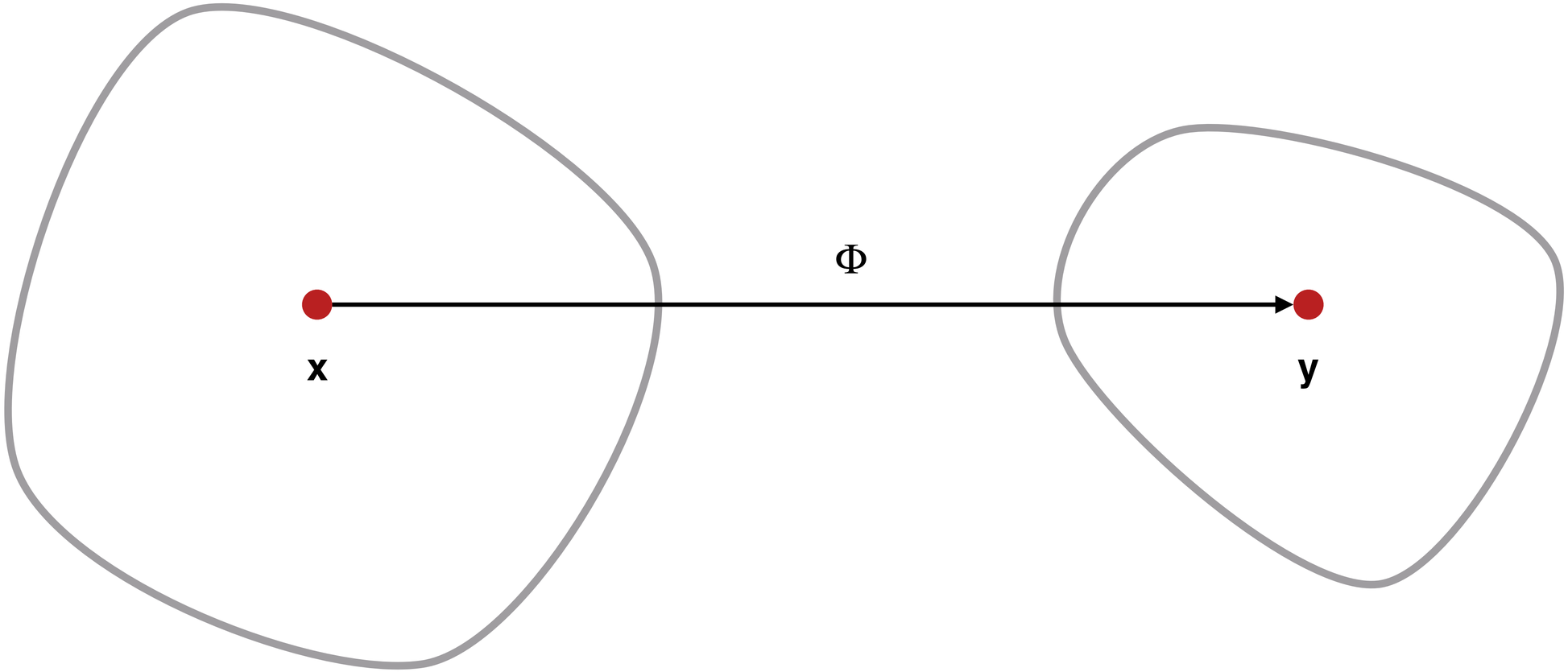
Each subnetwork Φ maps an image x to a code y

Images

$\mathcal{X} = \mathbb{R}^m$

Codes

$\mathcal{Y} = \mathbb{R}^n$



Generating iconic examples

Attribution

Semantic identification

Generating iconic examples

Attribution

Semantic identification

How much information about **x** does **y** contain?

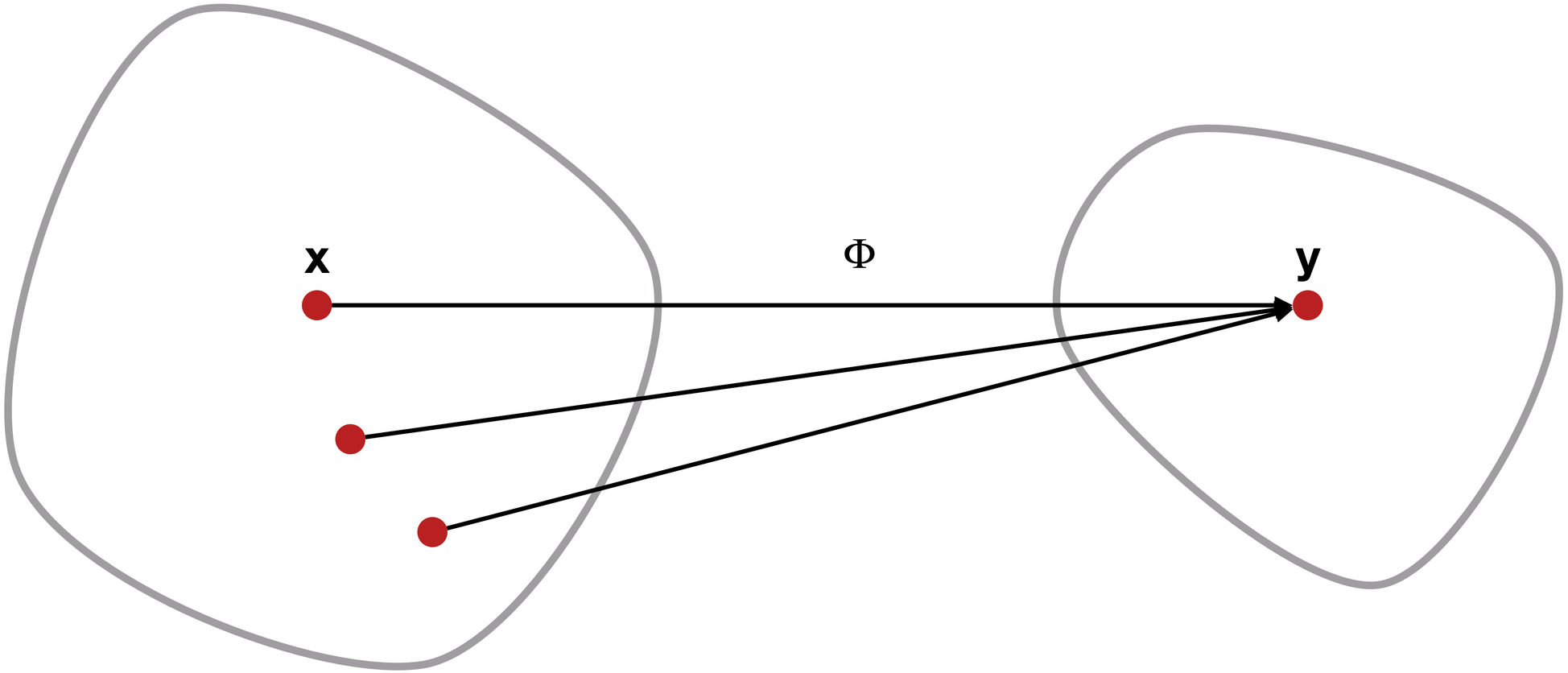
Find out "how much" of the image x can be reconstructed from the code y

Images

$\mathcal{X} = \mathbb{R}^m$

Codes

$\mathcal{Y} = \mathbb{R}^n$



How much information about **x** does **y** contain?

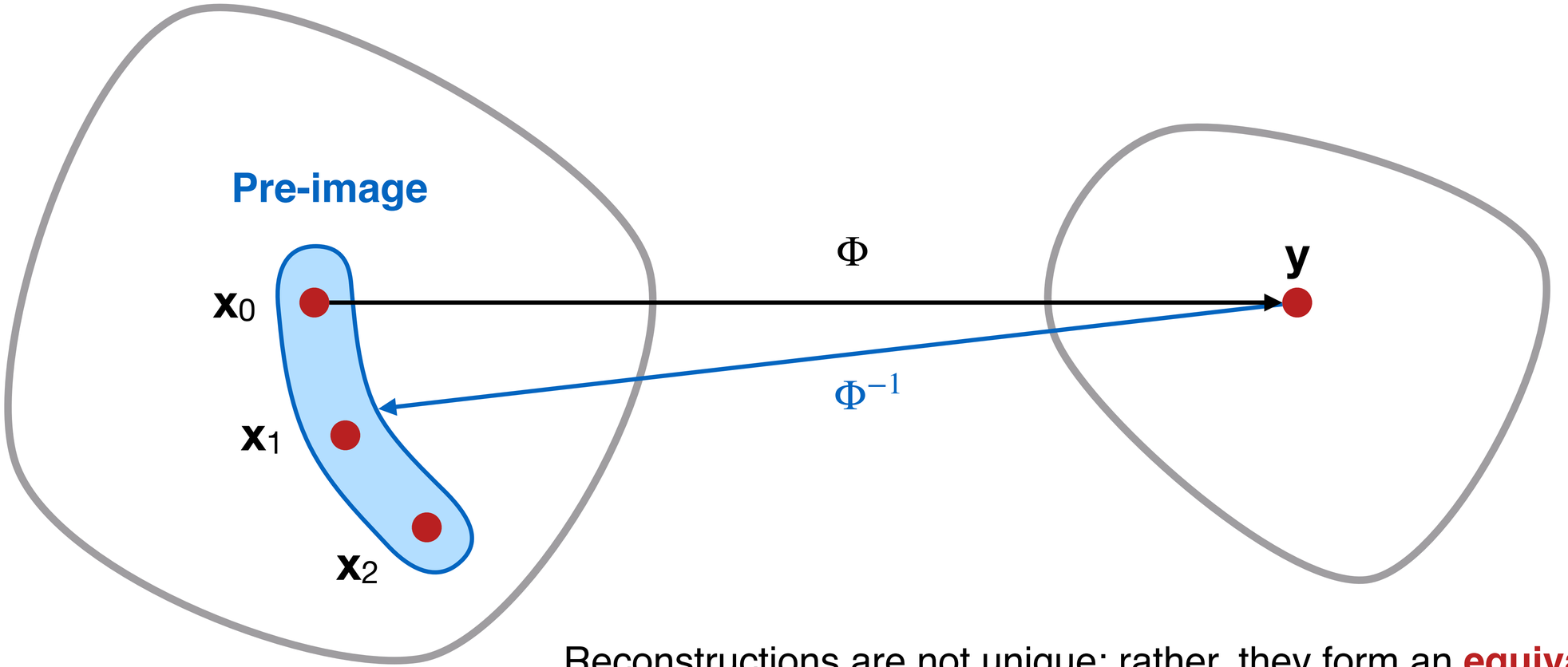
Find out "how much" of the image x can be reconstructed from the code y

Images

$\mathcal{X} = \mathbb{R}^m$

Codes

$\mathcal{Y} = \mathbb{R}^n$



Reconstructions are not unique; rather, they form an **equivalence class** of images that **are the same for the network**

Finding a pre-image in practice

Starting from random noise, "match" the code via direct optimization

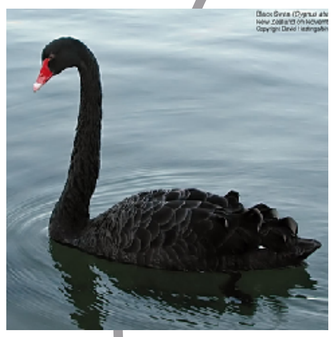
Images

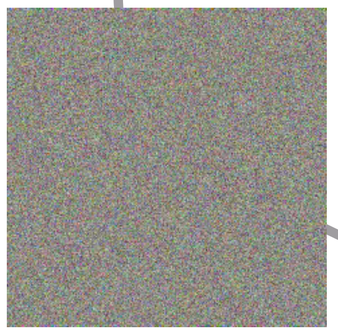
$\mathcal{X} = \mathbb{R}^m$

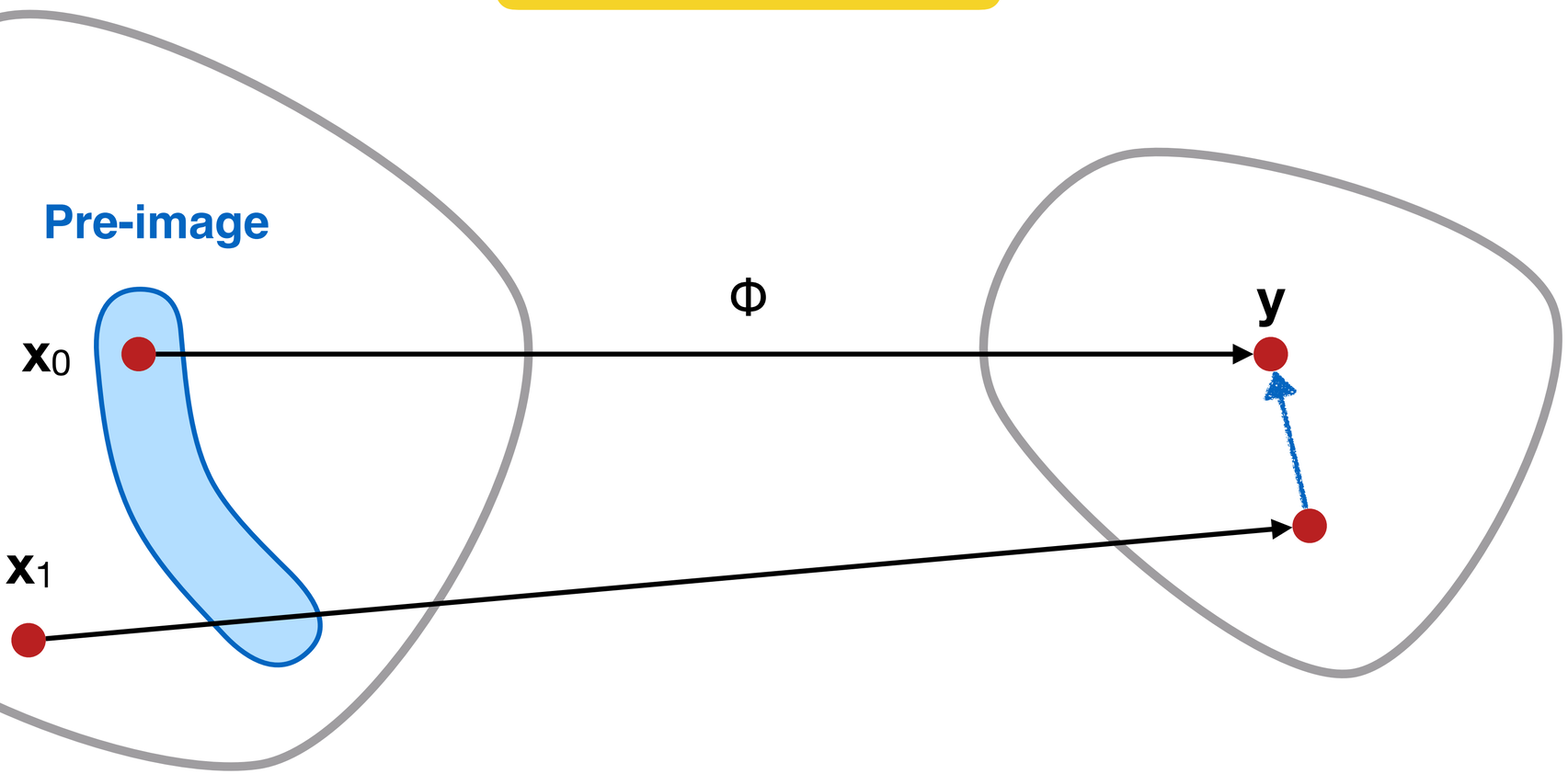
$$\min_{\mathbf{x}} \|\Phi(\mathbf{x}) - \Phi(\mathbf{x}_0)\|^2$$

Codes

$\mathcal{Y} = \mathbb{R}^n$

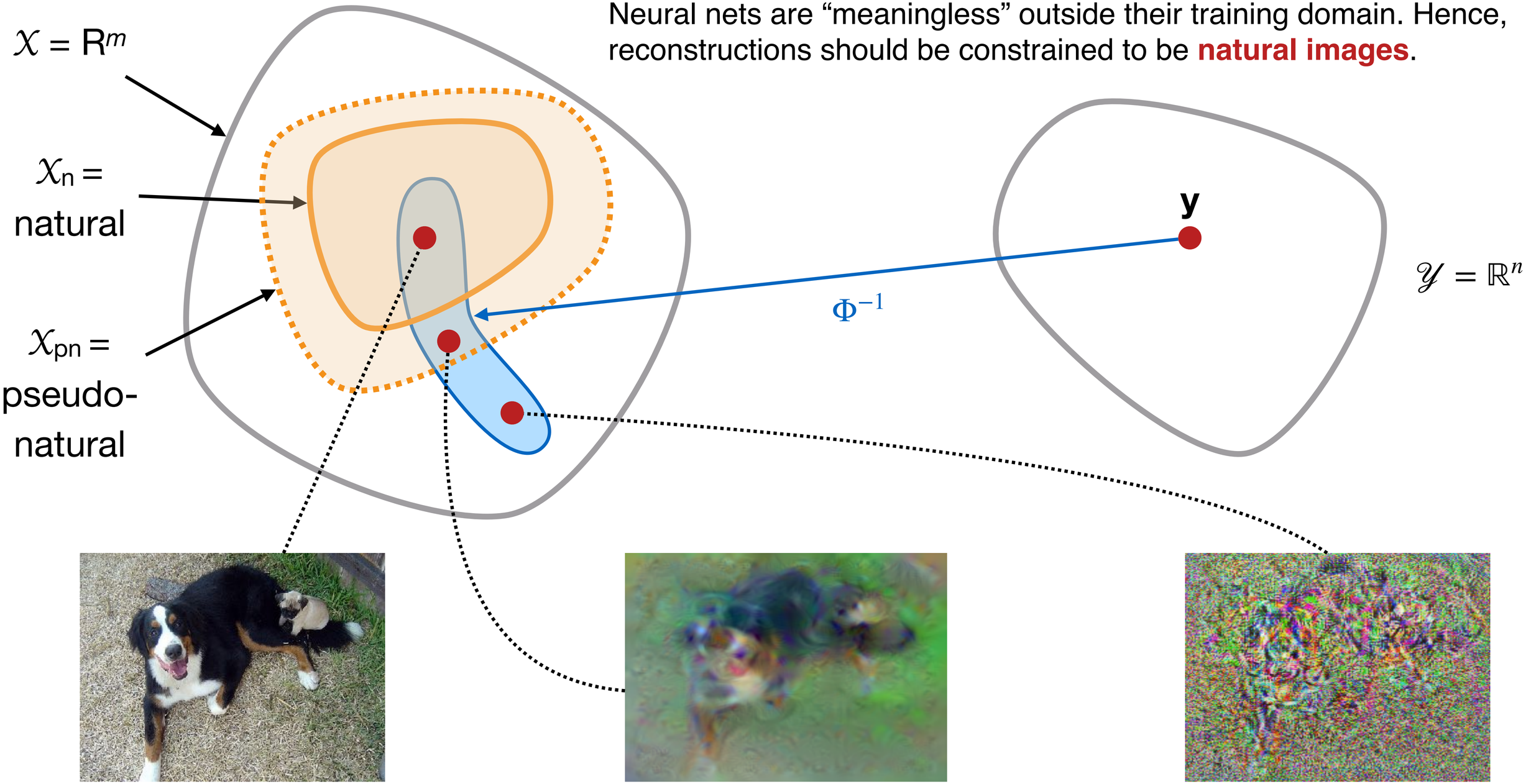






Natural pre-images

Neural nets are “meaningless” outside their training domain. Hence, reconstructions should be constrained to be **natural images**.



Natural pre-images

Several possible implementations

Regularized energy

$$\min_{\mathbf{x}} \|\Phi(\mathbf{x}) - \Phi(\mathbf{x}_0)\|^2 + \mathscr{R}(\mathbf{x})$$

For example TV-norm

Understanding deep image representations by inverting them
Mahendran Vedaldi, CVPR, 2015

Constrained optimization

$$\min_{\mathbf{x} \in \mathscr{X}_{pn}} \|\Phi(\mathbf{x}) - \Phi(\mathbf{x}_0)\|^2$$

For example Deep Image Prior

Deep image prior
Ulyanov Vedaldi Lempistky, CVPR, 2018

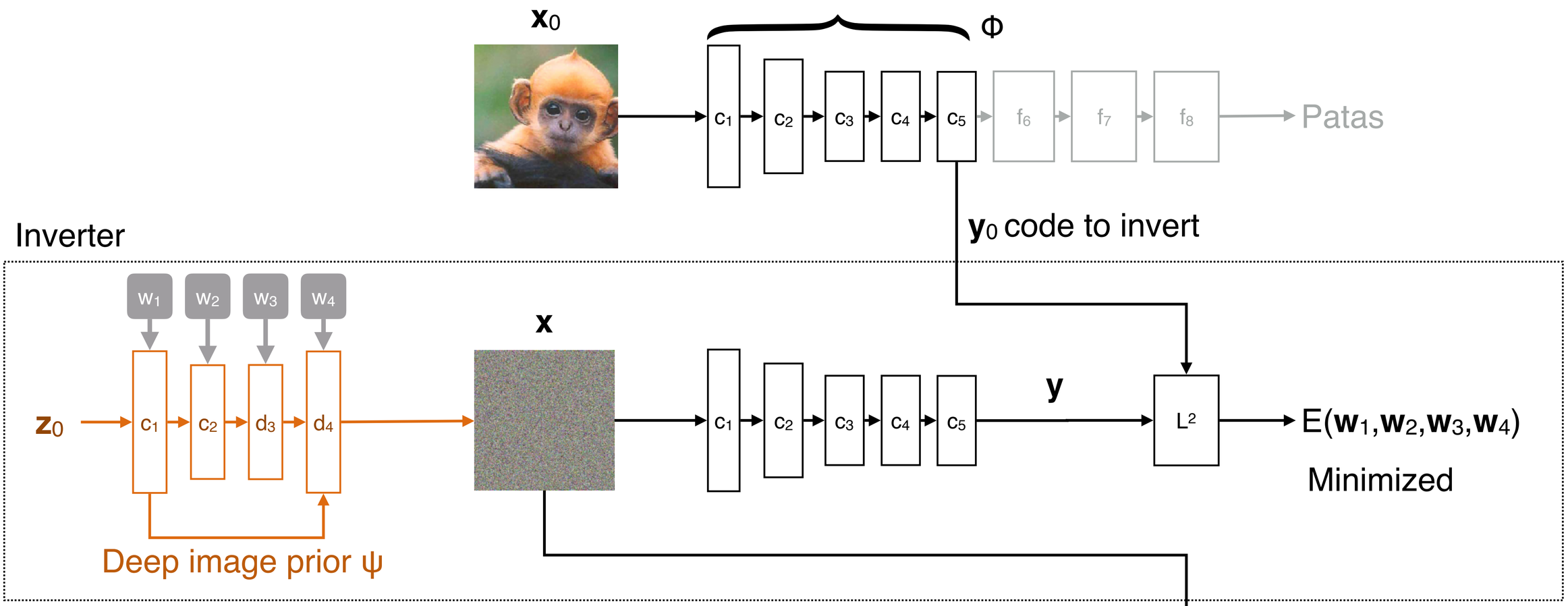
Posterior probability

$$p(\mathbf{x} \,|\, \mathbf{y}) \sim \delta(\Phi(\mathbf{x}) - \mathbf{y}) \cdot p(\mathbf{x})$$

For example Plug & Play gen. nets

Plug & play generative networks: Conditional iterative generation of images in latent space
Nguyen, Yosinksi, Bengio, Dosovitskiy, Clune, CVPR, 2017

Inverting codes *via* the deep image prior



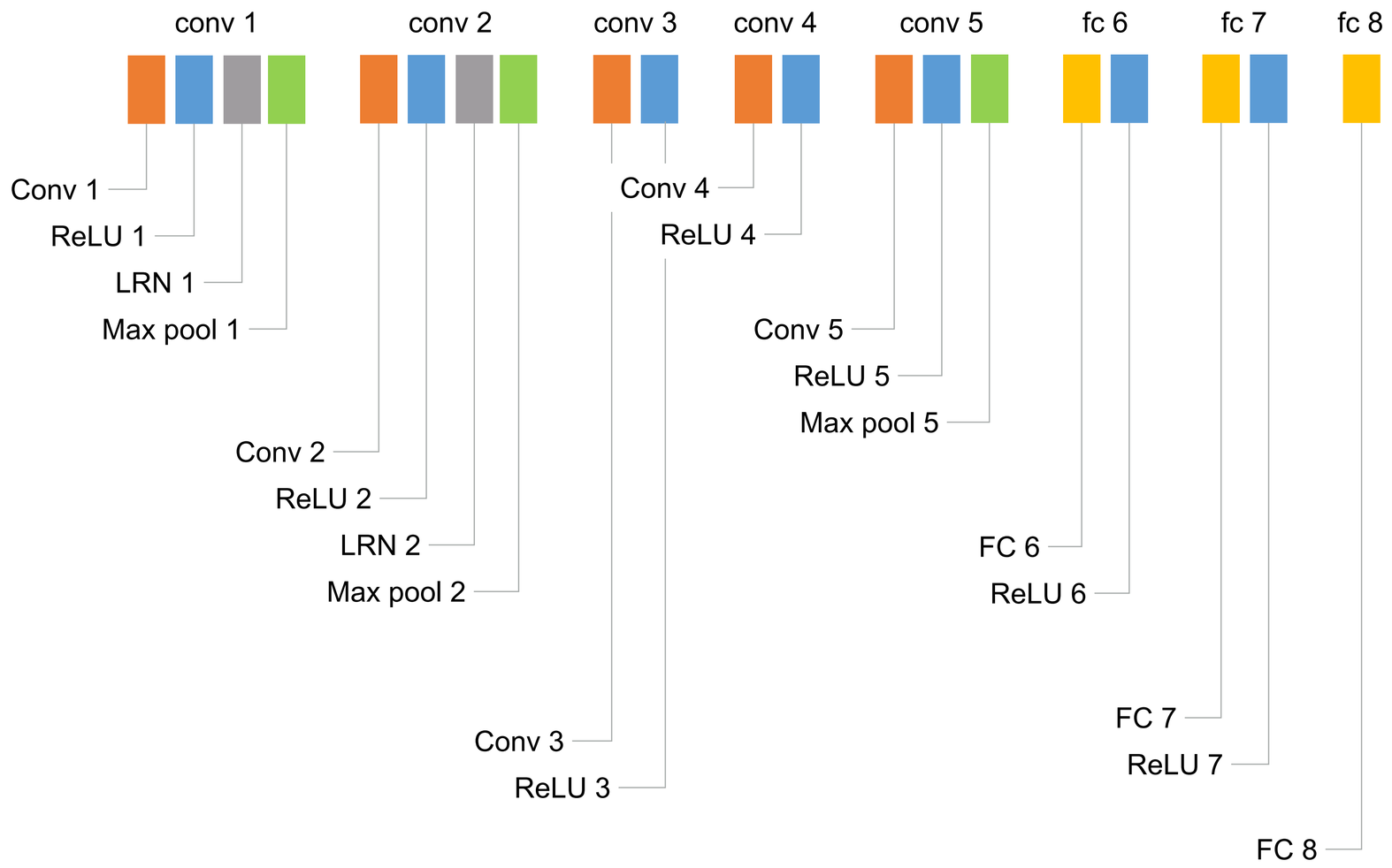
$$\min_{\mathbf{w}} \|\Phi(\Psi(\mathbf{w})) - \Phi(\mathbf{x}_0)\|^2$$

Inversion result

The inverter is only given

- The code $\mathbf{y}_0$
- The network to invert Φ
- The structure (not the parameters) of the generator ψ

Inverting a Deep CNN



AlexNet
[Krizhevsky et al. 2012]

Inverting a Deep CNN

Original image

Inverting a Deep CNN

Original image

Inverting a Deep CNN

Original image

Inverting a Deep CNN

Original image

Inverting a Deep CNN

Original image

Inverting a Deep CNN

Original image

Inverting a Deep CNN

Original image

Inverting a Deep CNN

Original image

Inverting a Deep CNN

Original image

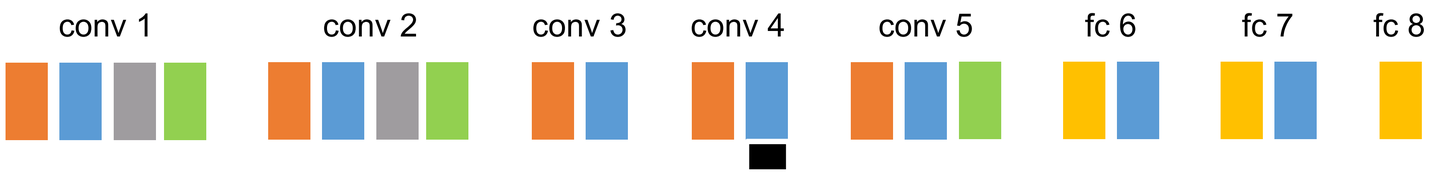
Inverting a Deep CNN

Original image

Inverting a Deep CNN

Original image

Inverting a Deep CNN



Original image

Inverting a Deep CNN

Original image

Inverting a Deep CNN

Original image

Inverting a Deep CNN

Original image

Inverting a Deep CNN

Original image

Inverting a Deep CNN

Original image

Inverting a Deep CNN

Original image

Inverting a Deep CNN

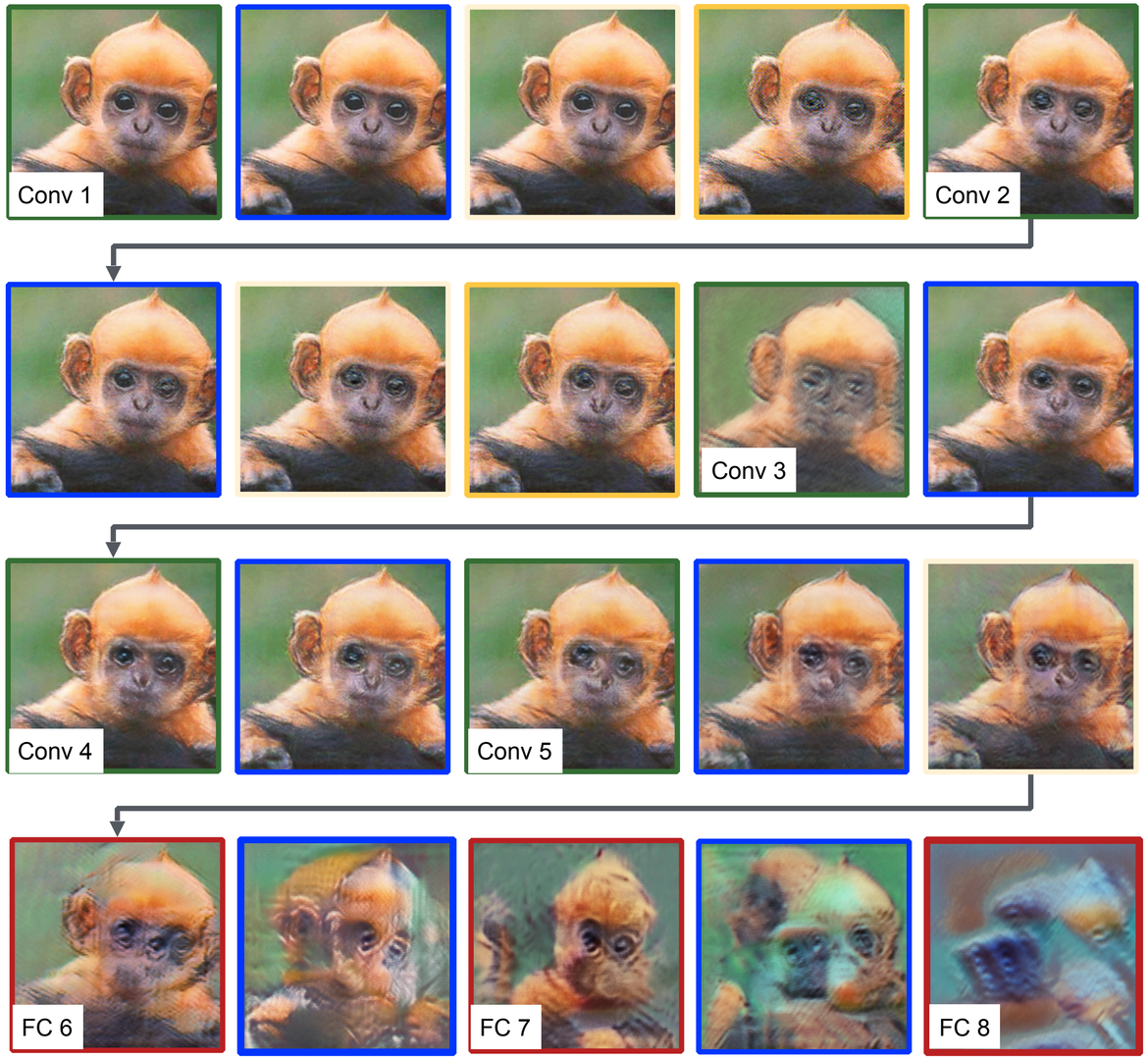
Original image

Inverting a Deep CNN

Original image

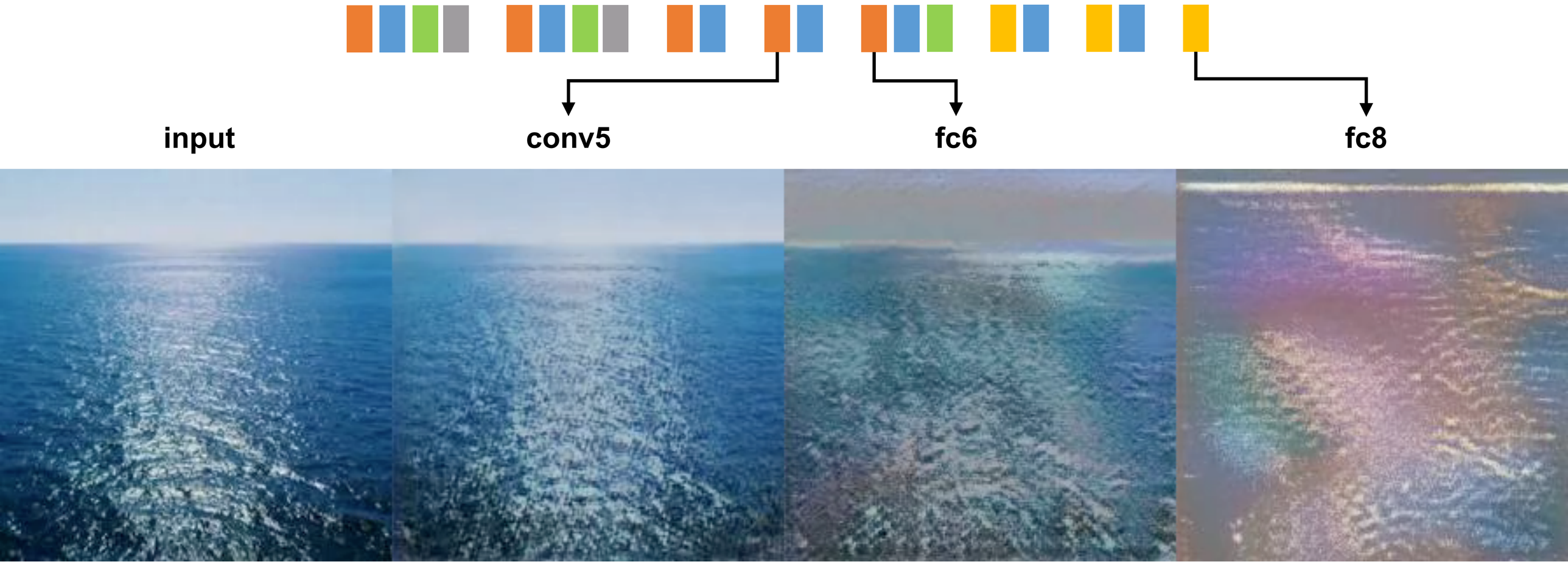
Inverting a Deep CNN

Original Image



Is the code semantic or visual?

Decoding AlexNet trained on ImageNet

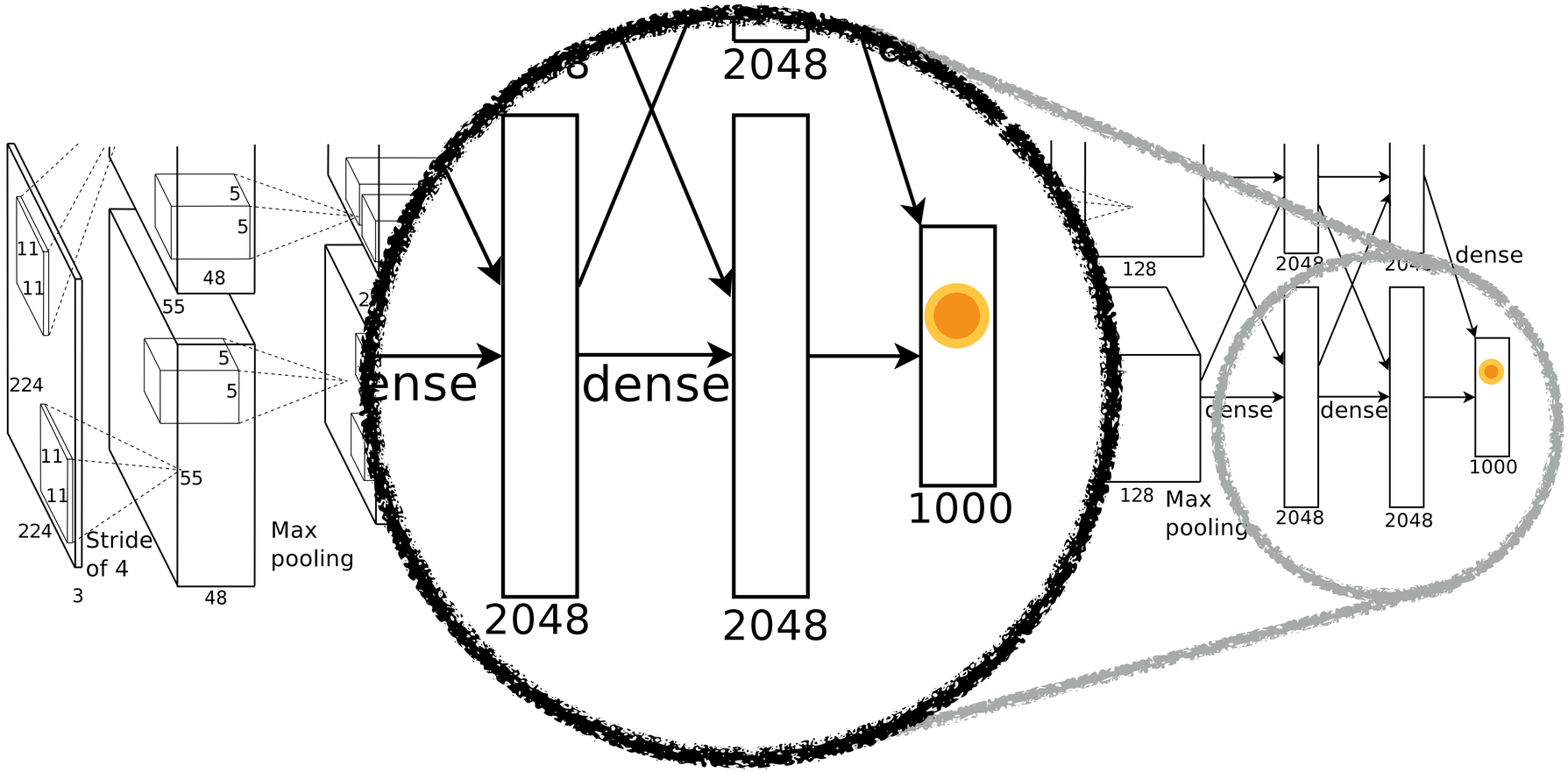


fc8 is a 1000-dimensional **class score vector**…
or is it?

Activation maximization

Look for an image that maximally activates a specific neuron activation

$$\min_{\mathbf{w}} -\langle \mathbf{e}_k, \Phi(\Psi(\mathbf{w})) \rangle$$



Deep Quiz

https://goo.gl/jURsCP


Black Swan (Cygnus atr
New Zealand on Novem
Copyright David Hastingsbr


Two's Company seen on thelovelyroom.com


Smartfood
Reese's
Kit Kat
M&M's
Skittles
Hot Cheetos
Fritos
Munchies

Visualization via direct optimization

Reading list

Visualizing higher-layer features of a deep network. Erhan, Bengio, Courville, U Montreal, 2009	**Activation maximisation** for class neurons
Visualizing and understanding convolutional networks Zeiler Fergus. ECCV, 2014.	Activation maximization using **empirical prior**, **deconvnet**
Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps Simonyan Zisserman Vedaldi, ICLR, 2104	Activation maximization and **saliency**
Understanding deep image representations by inverting them Mahendran Vedaldi, CVPR, 2015	**Inversion** at different depths, **natural image prior**
Google "inceptionsm" Mordvintsev et al. 2015	Activation maximisation for **intermediate neurons** Improved regularizers, artistic applications (deep dreams)
Understanding neural networks through deep visualisation Yosinksi et al. ICMLW, 2015	More regularizers, toolbox
Plug & play generative networks: Conditional iterative generation of images in latent space Nguyen, Yosinksi, Bengio, Dosovitskiy, Clune, CVPR, 2017	Strong learned regularizer, sample **diversity**
Deep image prior Ulyanov Vedaldi Lempistky, CVPR, 2018	Advanced "data agnostic" regularization

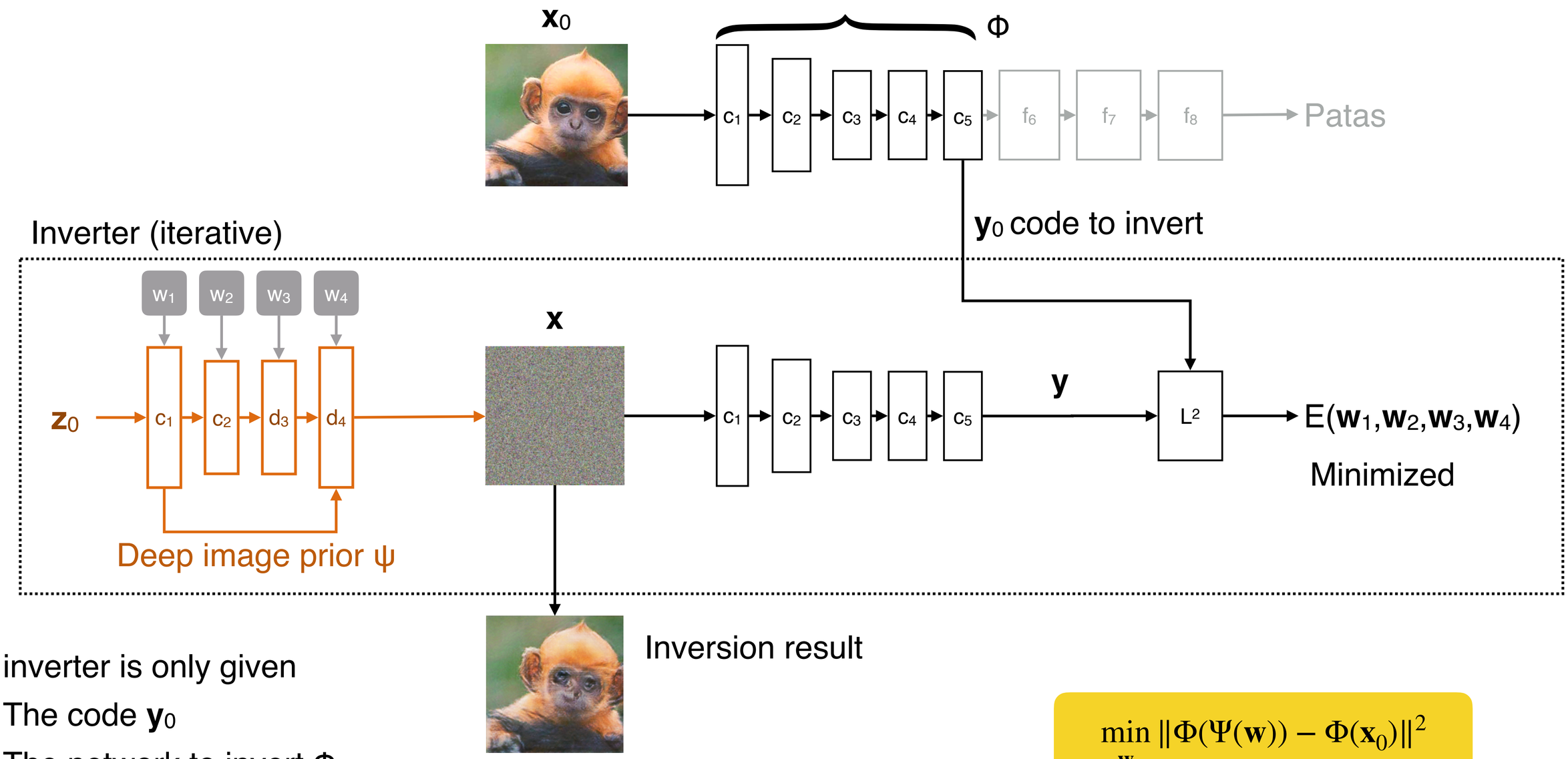
The choice of inversion prior has a strong effect

AlexNet Visualizations

Deep Image Prior

TV-Norm Prior

Using the deep image prior is just **one** option …

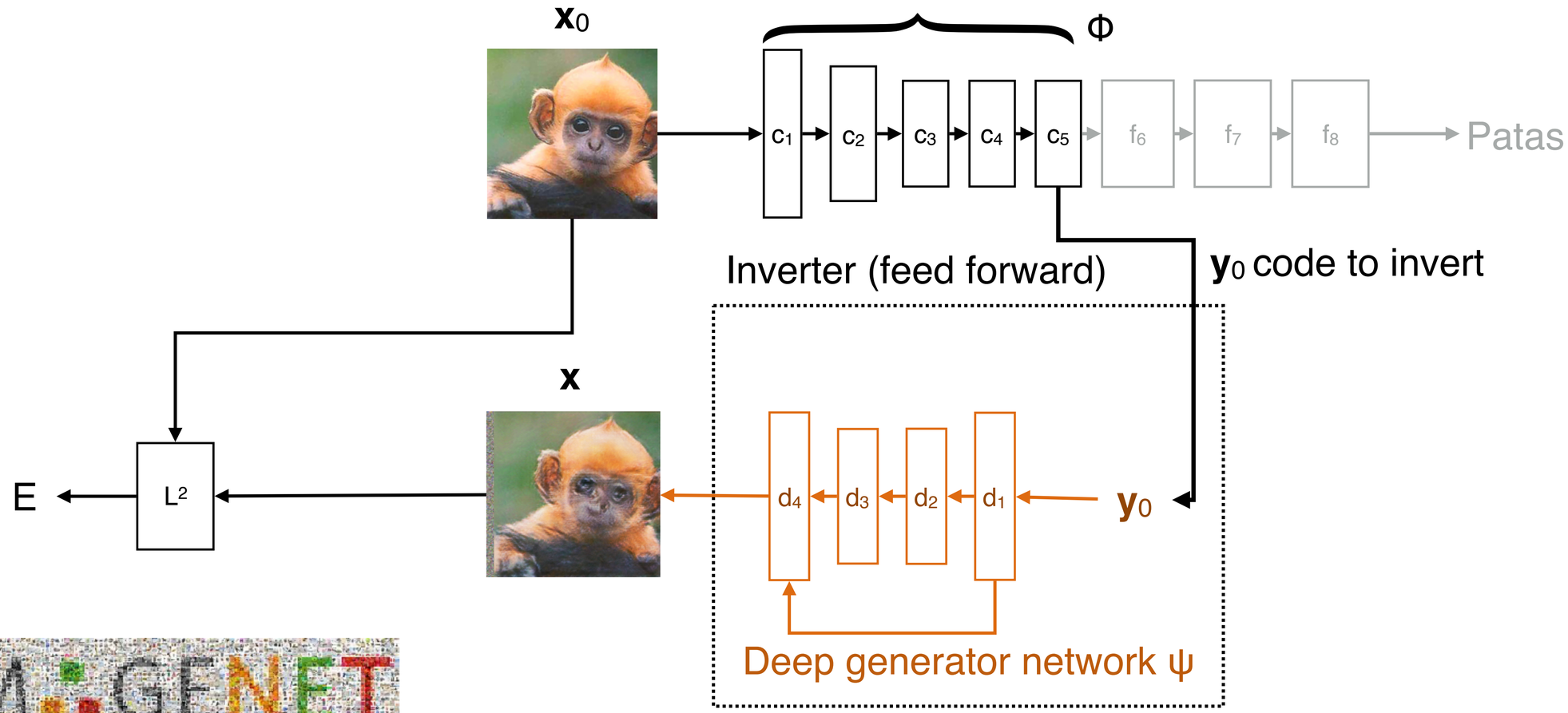


The inverter is only given

- The code $\mathbf{y}_0$
- The network to invert Φ
- The structure (not the parameters) of the generator ψ

$$\min_{\mathbf{w}} \|\Phi(\Psi(\mathbf{w})) - \Phi(\mathbf{x}_0)\|^2$$

… another is to learn an inverse generator network



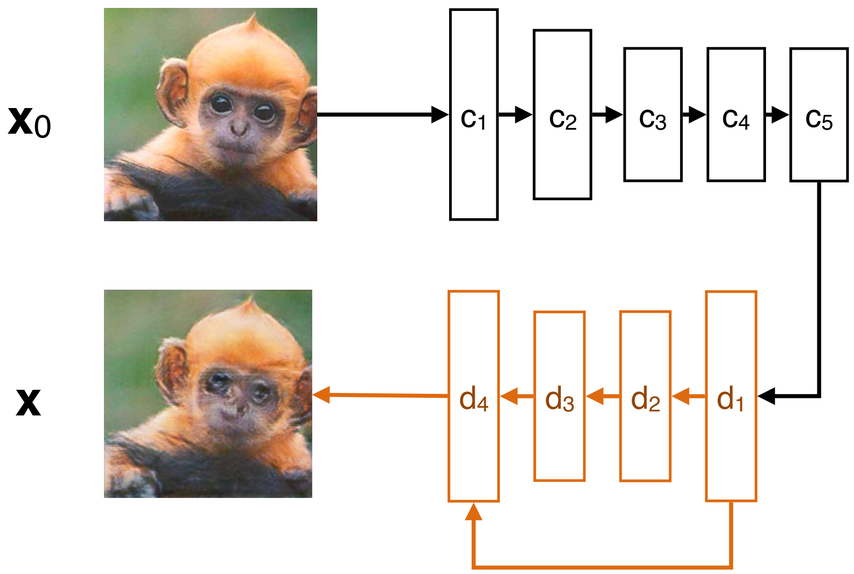
$$\min_{\Psi} \frac{1}{N} \sum_{i=1}^{N} \|\Psi(\Phi(\mathbf{x}_i)) - \mathbf{x}_i\|^2$$

The inverter is now given:

- The code $\mathbf{y}_0$
- A large **training set of images** to learn a generator from

Visualization via learning a generator

Train a strong prior from examples



$\mathbf{x}_0 \cong \mathbf{x}$ using a **perceptual loss**

$\Phi(\mathbf{x}_0) \cong \Phi(\mathbf{x})$ **inversion loss**

$p(\mathbf{x}_0) = p(\mathbf{x})$ **GAN loss**

acquire a very good prior $p(\mathbf{x})$

Inverting convolutional networks with convolutional networks
Dosovitskiy Brox, CVPR, 2016

Synthesizing the preferred inputs for neurons in neural networks via deep generator networks
Nguyen, Dosovitskiy, Yosinski, Brox, Clune, NIPS, 2016

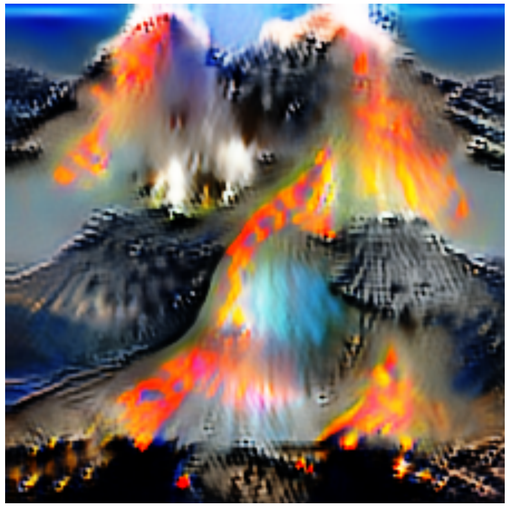
Generating images with perceptual similarity metrics based on deep networks
Dosovitskiy Brox, NIPS, 2016

Plug & play generative networks: Conditional iterative generation of images in latent space
Nguyen, Yosinksi, Bengio, Dosovitskiy, Clune, CVPR, 2017

Diagnostic *vs* aesthetic value

Our goal: diagnose a given **discriminator network Φ**

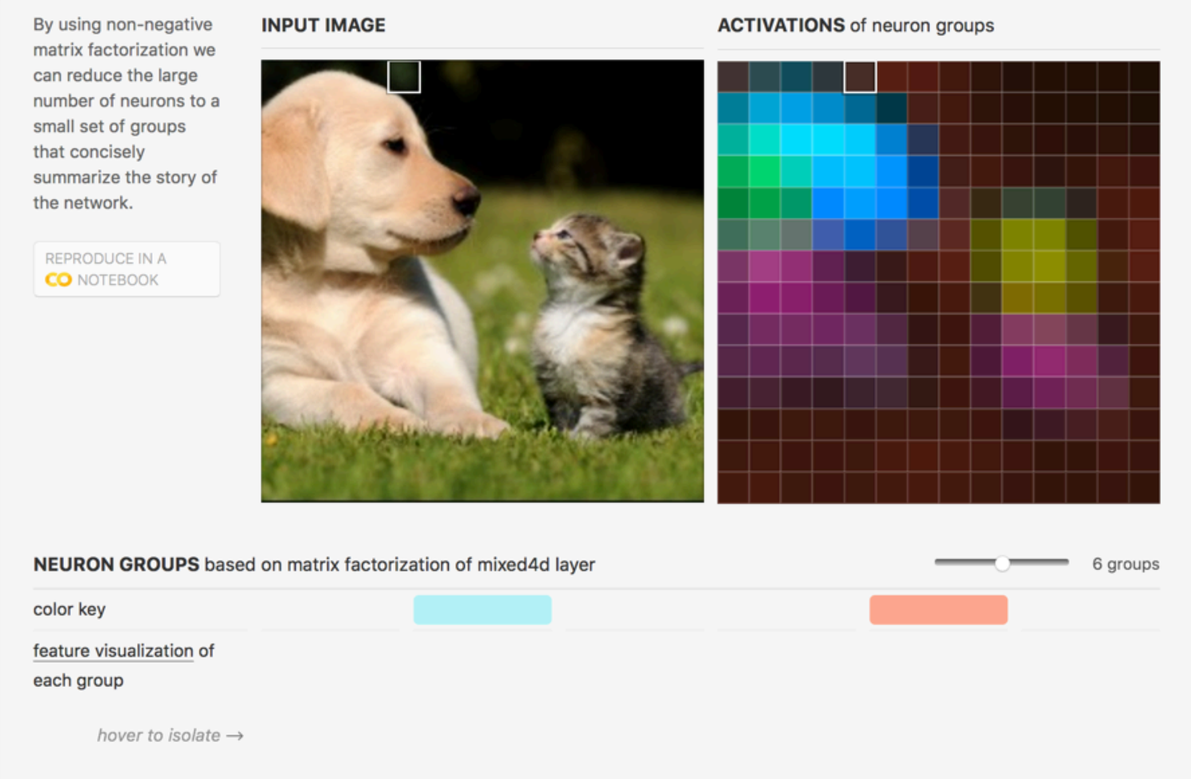
But inversions **also** reflect the chosen "natural image" **prior** $p(\mathbf{x})$

	Deep Image Prior	Plug & Play Gen. Net.	Empirical prior
			
$p(\mathbf{x})$ =	generator net **structure** only	generator net **trained GAN-like** from **ImageNet**	**ImageNet** validation set (**empirical distribution**)

Illustrates the **model Φ**

Illustrates the **prior** $p(\mathbf{x})$

If you want to dig further

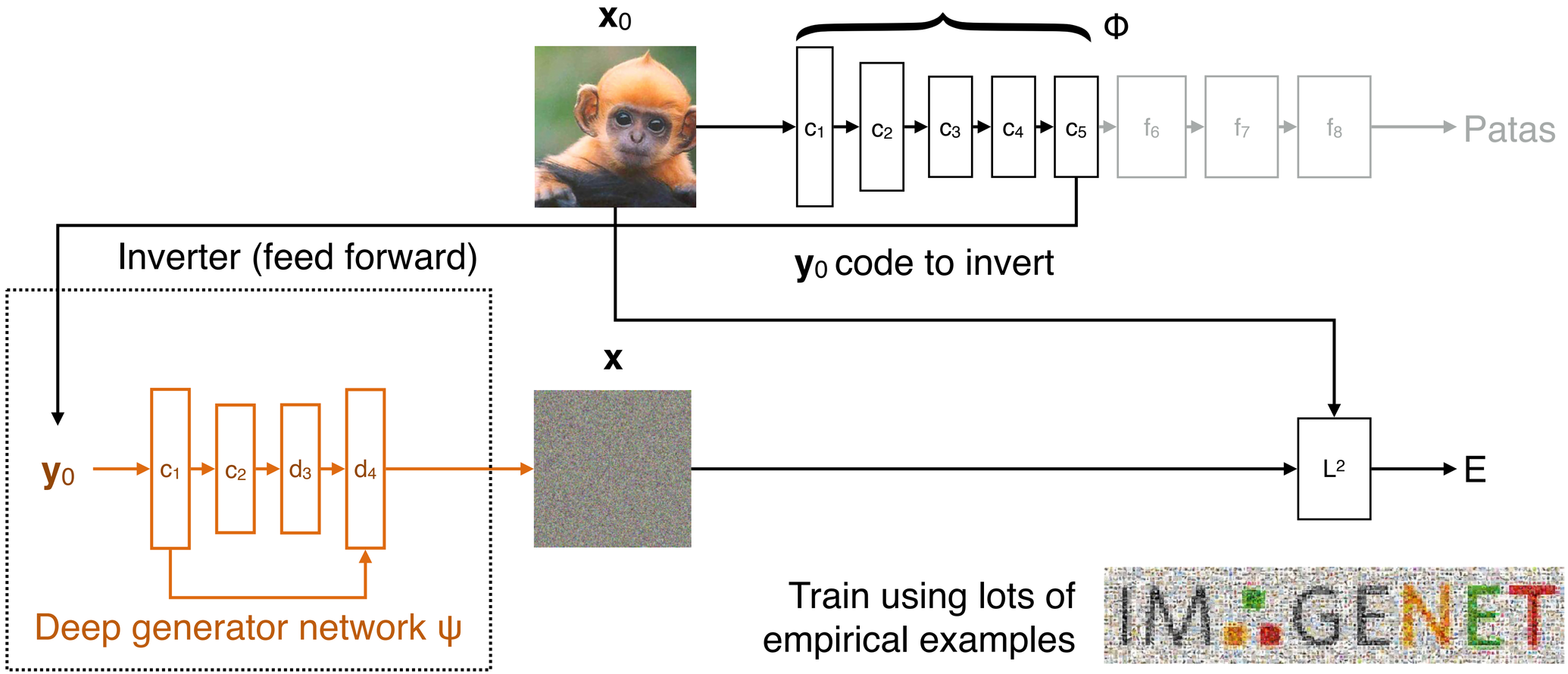


[Distill]

The building blocks of interpretability
Olah, Satyanarayan, Johnson, Carter, Schubert, Ye, Mordvintsev
Distill, 2018. https://distill.pub/2018/building-blocks

Understanding neural networks through deep visualisation
Yosinksi et al. ICMLW, 2015

.. another is to learn an inverse generator network



The inverter is now given:

- The code $\mathbf{y}_0$
- A large **training set of images** to learn a generator from

$$\min_{\Psi} \frac{1}{N} \sum_{i=1}^{N} \|\Psi(\Phi(\mathbf{x}_i)) - \mathbf{x}_i\|^2$$

Generating iconic examples

Attribution

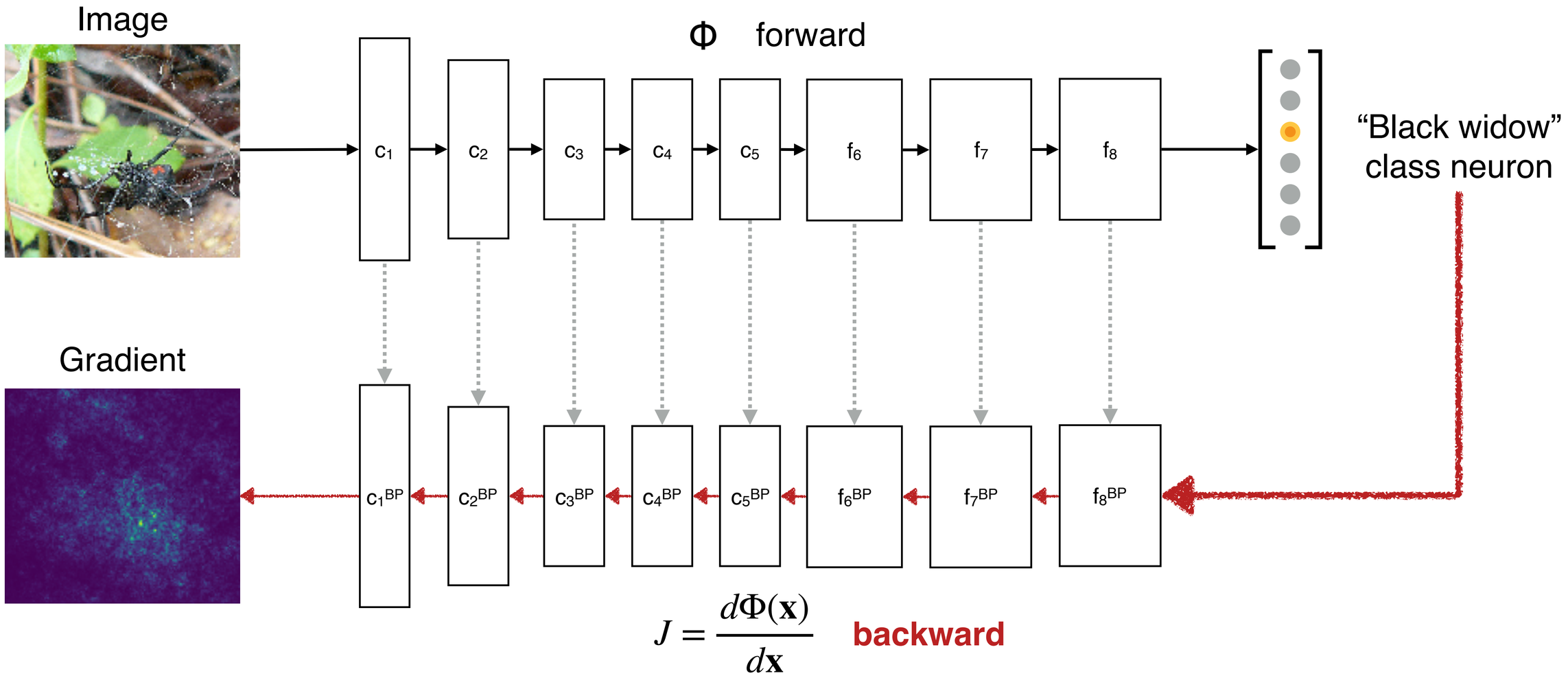
Semantic identification

Attribution

Find what parts of an image are salient for a deep network

Saliency: Backpropagation

Sensitivity analysis of target neuron w.r.t. input pixels



The "salient" pixels usually light up

Deep inside convolutional networks, Simonyan, Vedaldi, Zisserman, ICLR, 2014

Saliency: Gradient Backpropagation Variants

Three popular methods

Deconvolution

Visualizing and understanding convolutional networks
Zeiler Fergus, ECCV, 2014

Gradient (backpropagation)

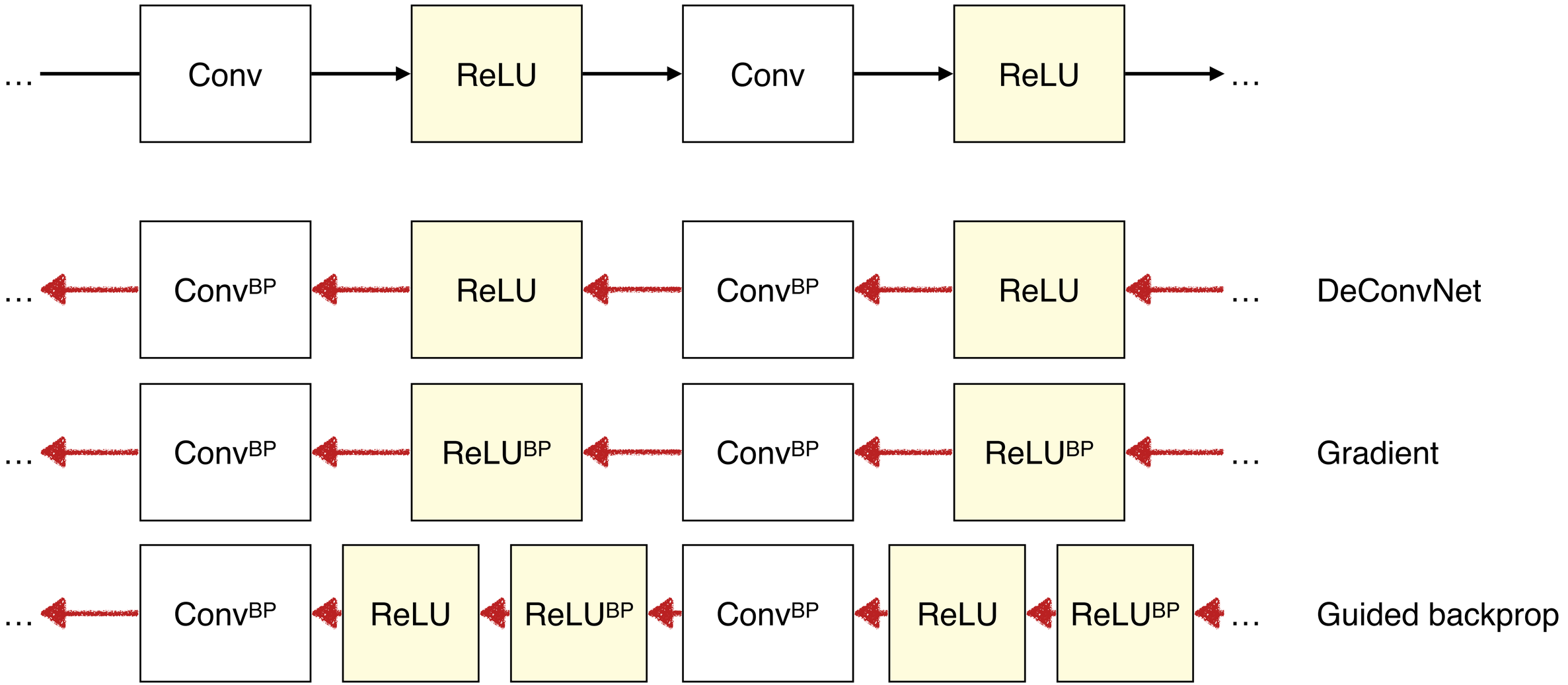
Deep inside convolutional networks: Visualising image classification models and saliency maps
Simonyan, Vedaldi, Zisserman, ICLR, 2014

Guided backpropagation

Striving for simplicity: The all convolutional net
Springenberg, Dosovitskiy, Brox, Riedmiller, ICLR, 2015

Saliency: Gradient Backpropagation Variants

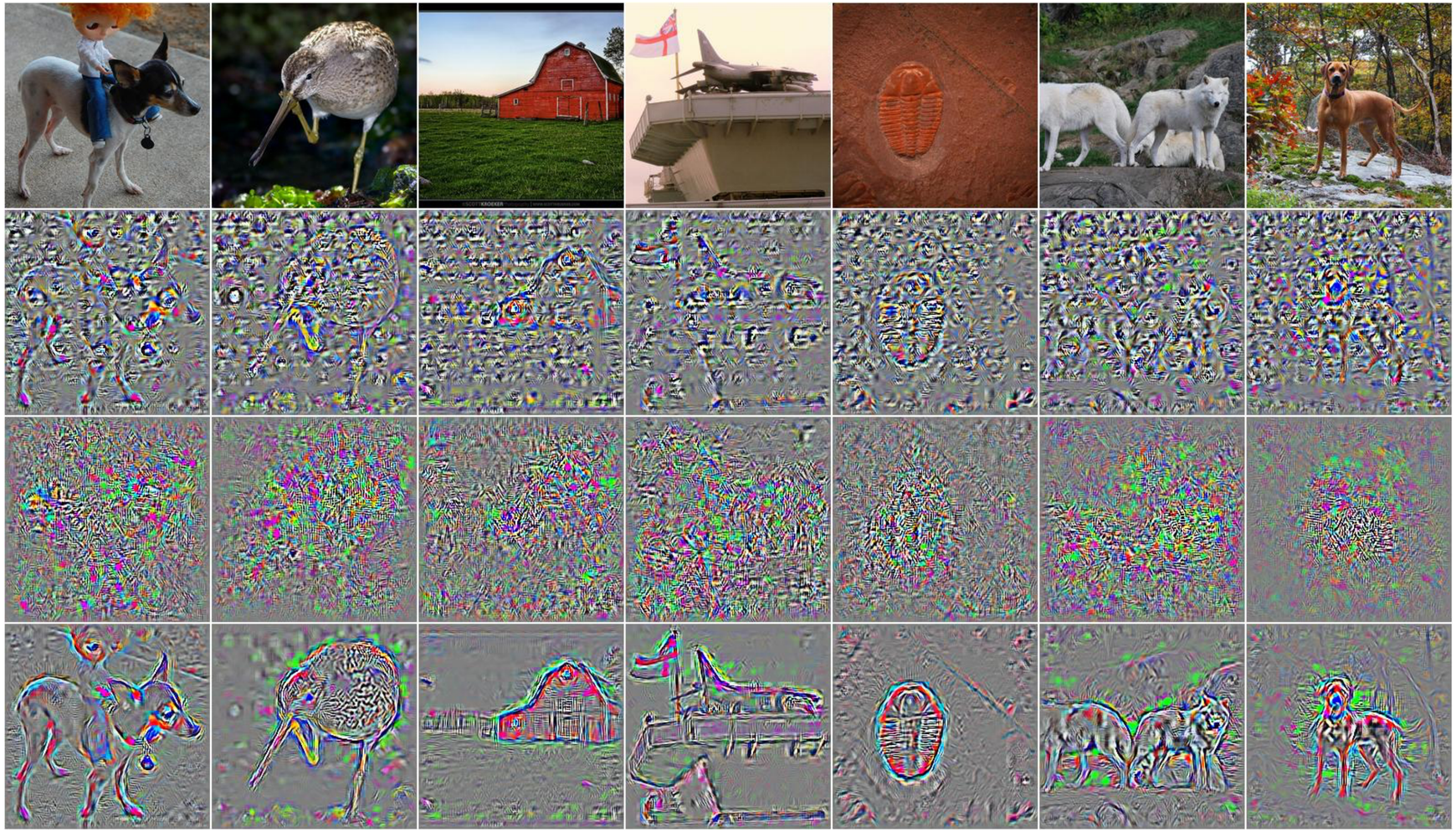
The only difference is in how ReLU is reversed!



Salient deconvolutional networks, Mahendran Vedaldi, ECCV, 2016

Saliency: Gradient Backpropagation Variants

… the only difference is in how ReLU is reversed



DeConvNet

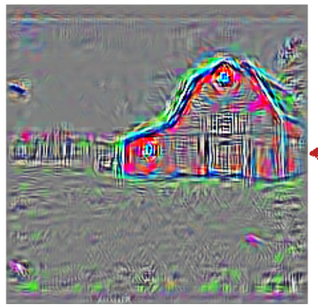
Gradient

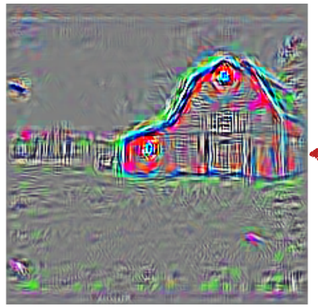
Guided backprop

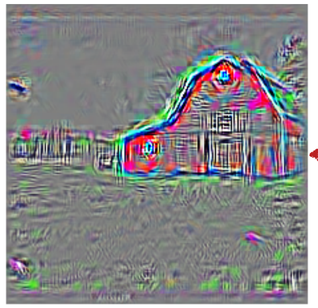
See comparison in [**Salient deconvolutional networks,** Mahendran Vedaldi, ECCV, 2016]

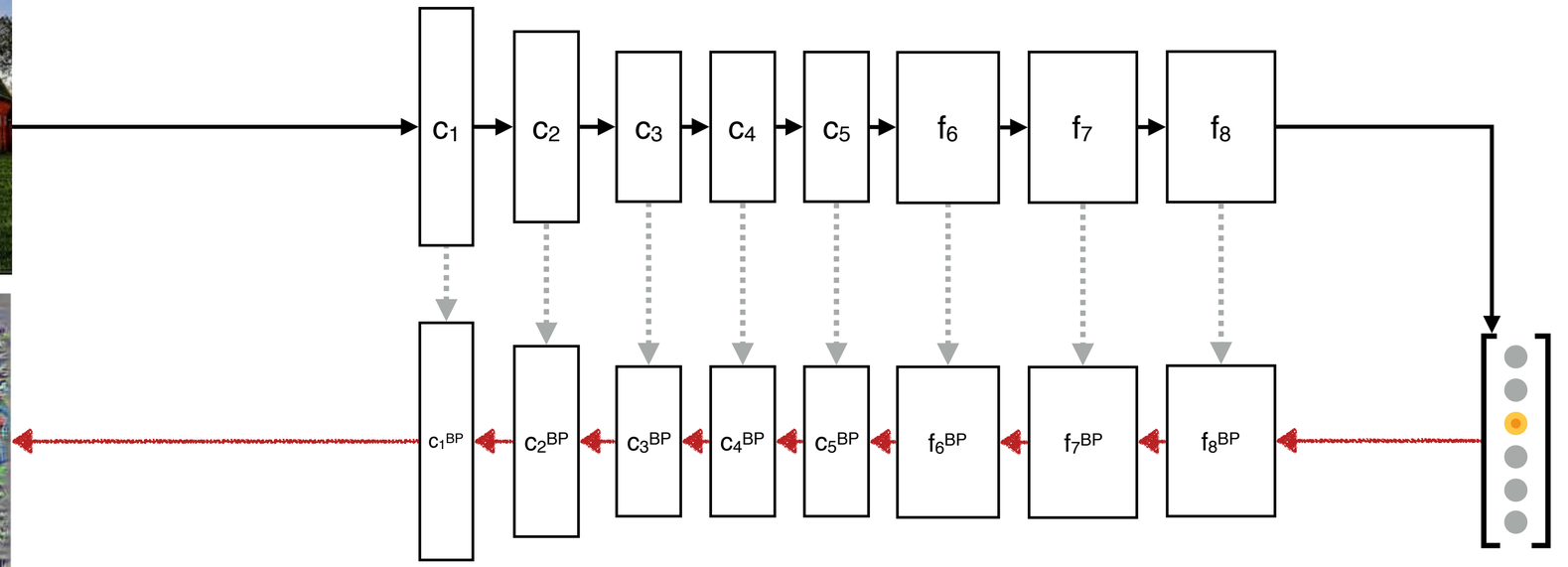
Lack of channel specificity

The saliency of neurons at the same location is about the same









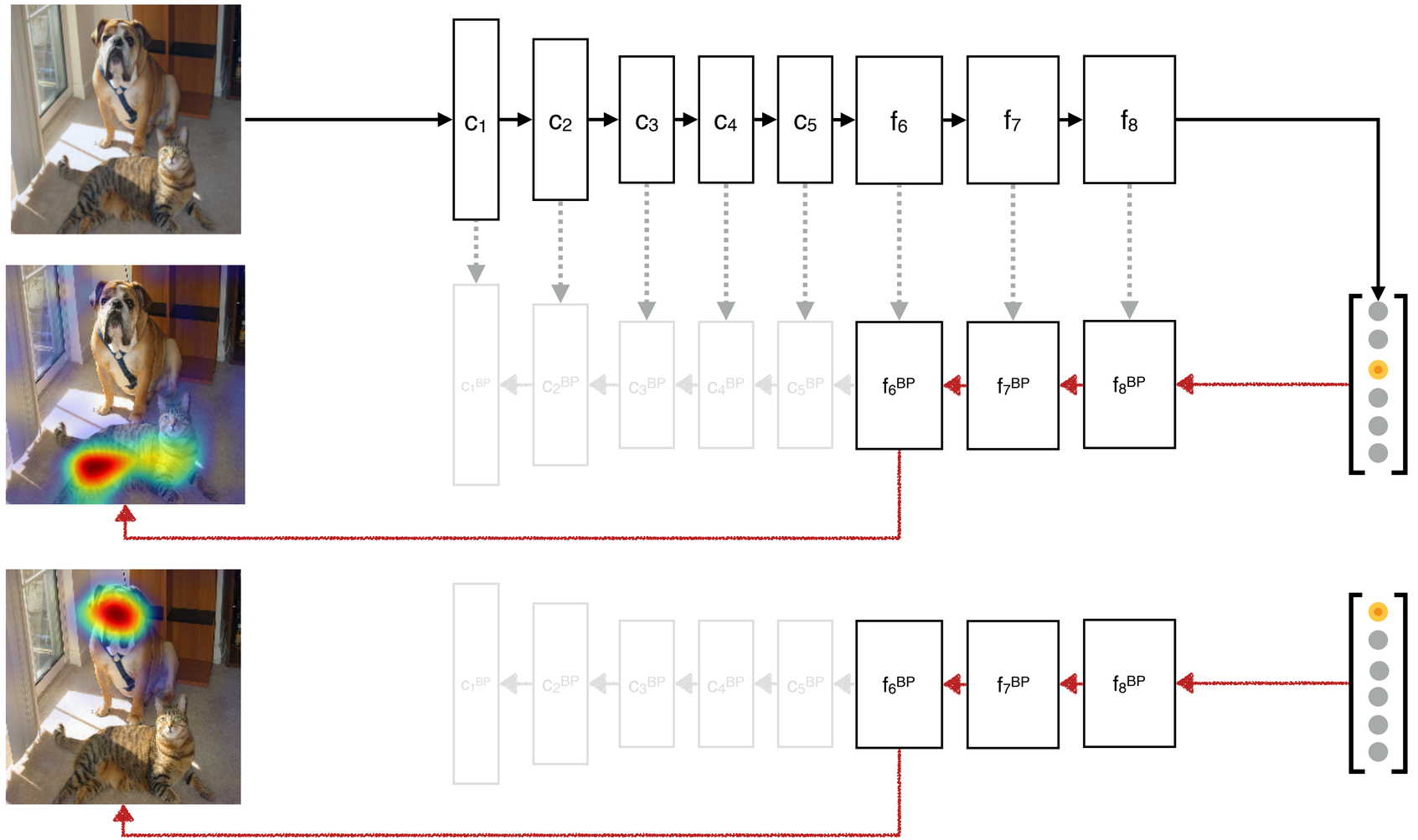
Saliency for:

maximally activated neuron

random neuron

minimally activated neuron

Better **channel specificity** can be achieved by backpropagating only a few layers

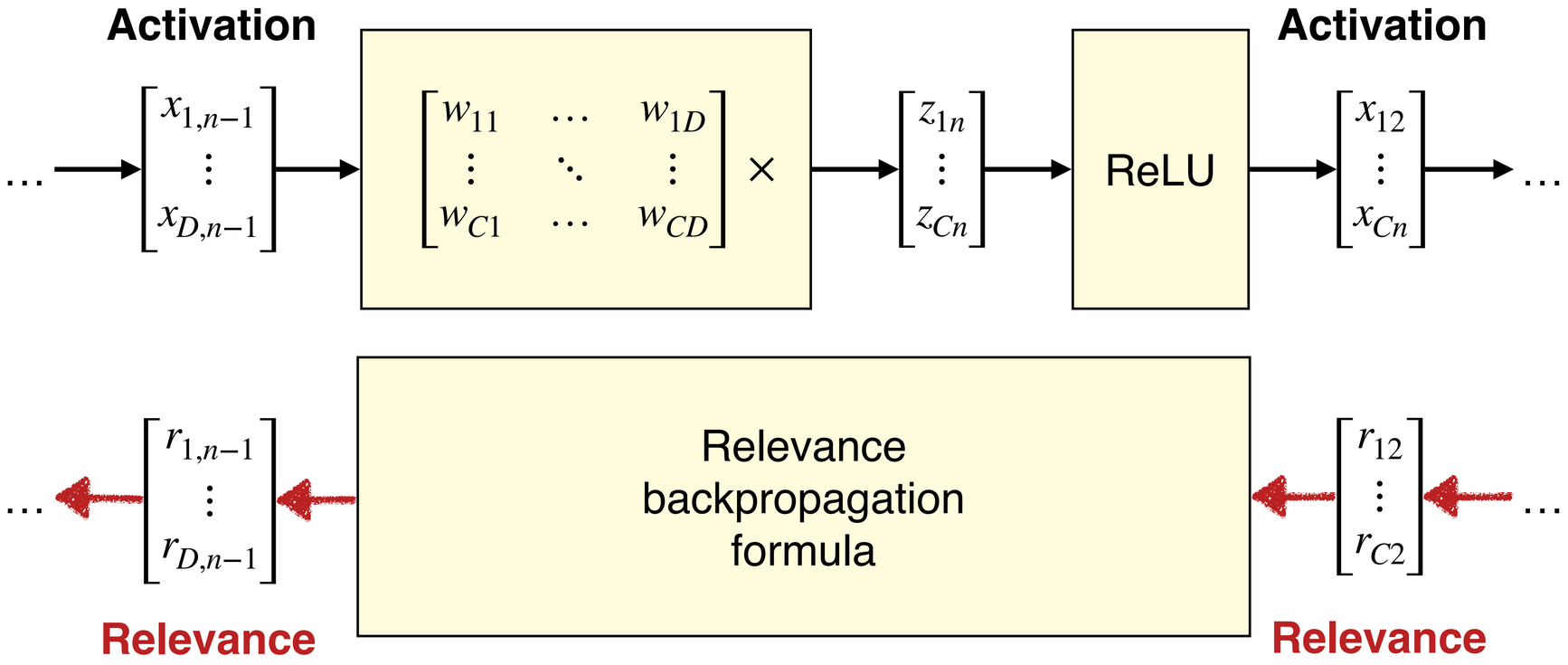


Learning deep features for discriminative localization
Zhou, Khosla, Lapedriza, Oliva, Torralba, CVPR, 2016

Grad-CAM: Visual explanations from deep networks via gradient-based localization
Selvaraju, Cogswell, Das, Vedantam, Parikh, Batra, ICCV, 2017

Saliency: Relevance and excitation backpropagation

Define some other rules to back-propagate the "relevance" of activations

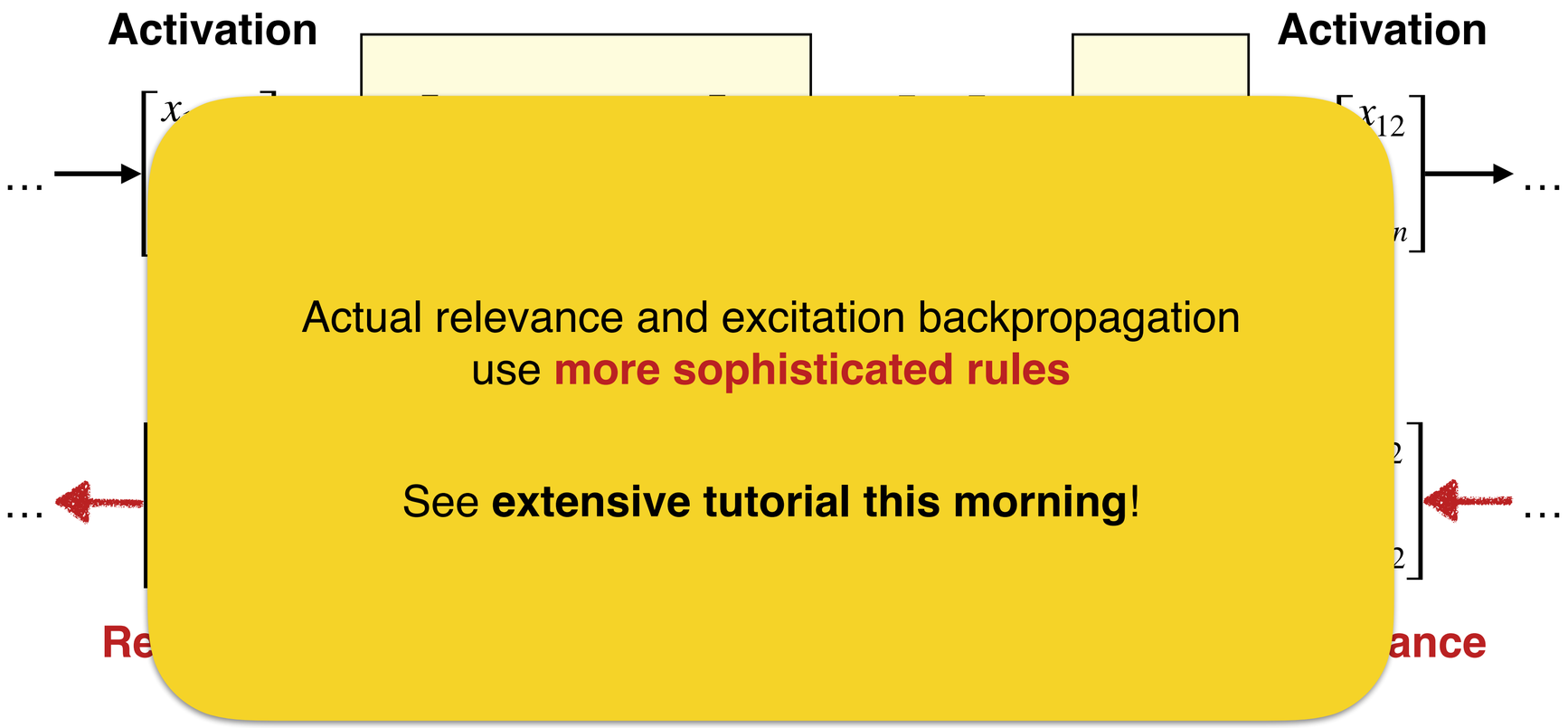


On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation
Bach, Binder, Montavon, Klauschen, Müller. PLOS one, 2015

Top-down neural attention by excitation backprop
Zhang, Lin, Brandt, Shen, Sclaroff, ECCV, 2016

Saliency: Relevance and excitation backpropagation

A **simple example**: gradients modulated by the forward activations



Actual relevance and excitation backpropagation
use **more sophisticated rules**

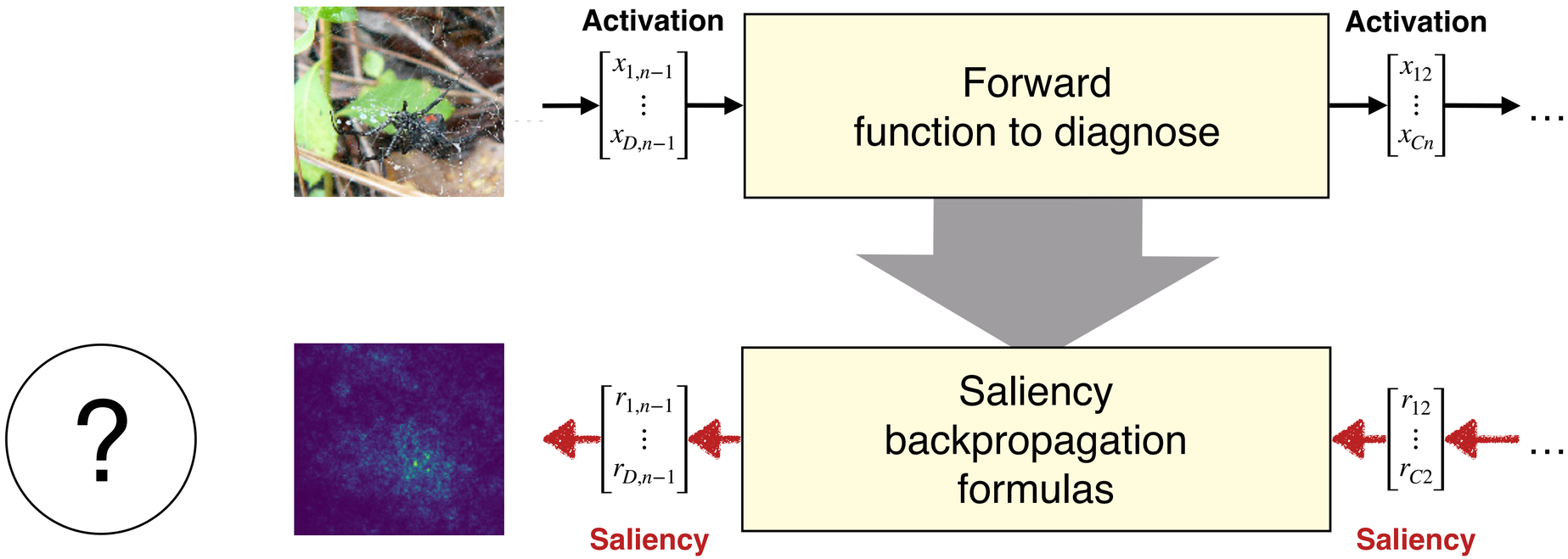
See **extensive tutorial this morning**!

Due to chaining and cancelations we get that at any level *m* relevance is the **modulated gradient**

$$\mathbf{r}_m^\top = \frac{dx_n}{d\mathbf{x}_m^\top} \cdot \mathrm{diag}(\mathbf{x}_m)$$

The meaning of saliency

Methods have been defined by specifying a “backpropagation formula”

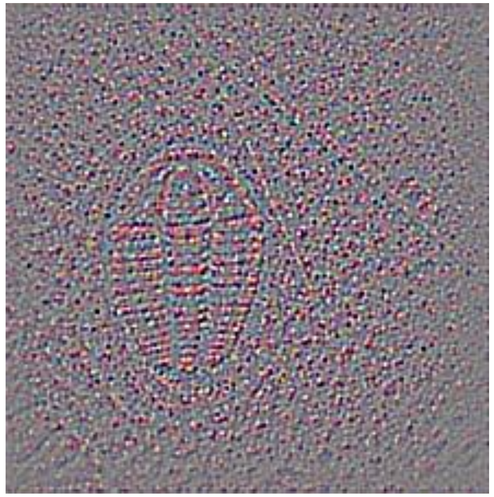


But what does the result of this computation actually mean?

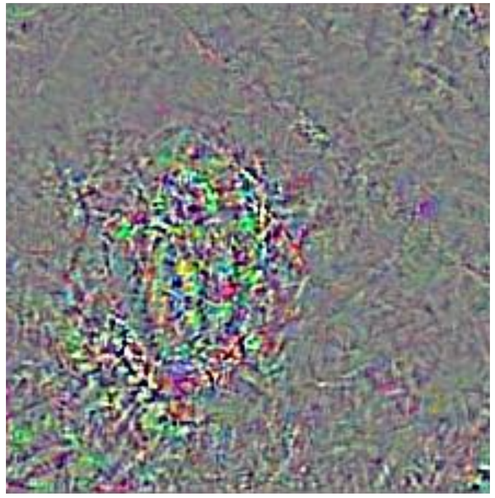
The meaning of saliency

It “looks good”

Deconvolution



Gradient



Guided Backprop

Deconvolution

- Sharp
- Poor spatial selectivity

Gradient

- Blurry
- Good spatial selectivity

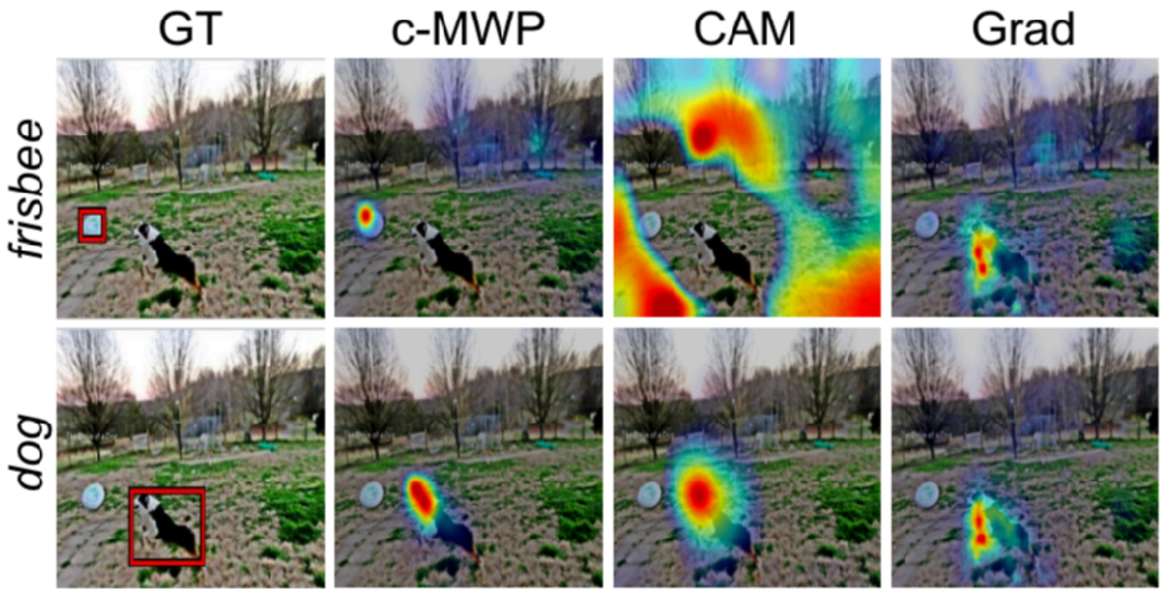
Guided Backprop

- Sharp
- Good spatial sensitivity

Reminder: they all still have poor channel selectivity

The meaning of saliency

It generates "semantic" heat maps



Pointing Game from Excitation Backprop

Measure: degree of correlation between the saliency results and **ground truth semantic labels** (e.g. objects).

A good correlation means that:

1) the **diagnosed model** "understand" the location of objects and

2) the **saliency method** can diagnose this fact

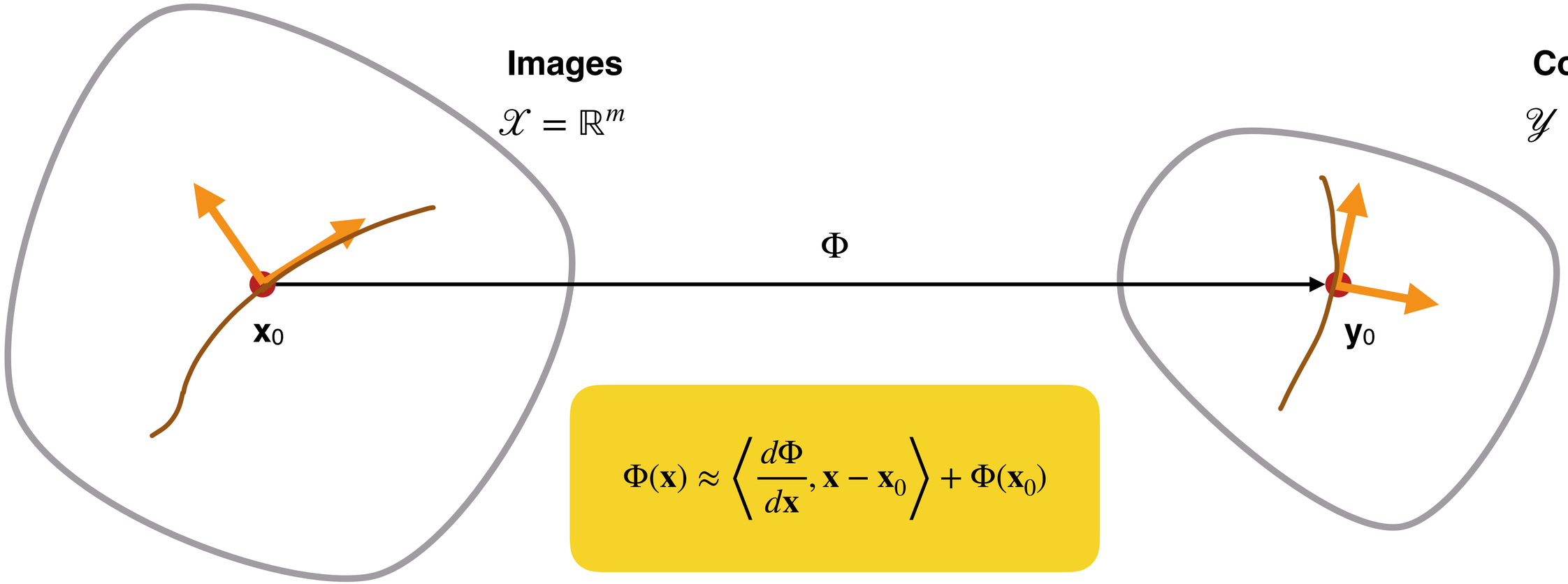
Drawbacks:

1) failure of localization **confound** limitations of the model and the saliency method

2) difficult to say which is which since the saliency formulas are largely **heuristics**

The meaning of saliency

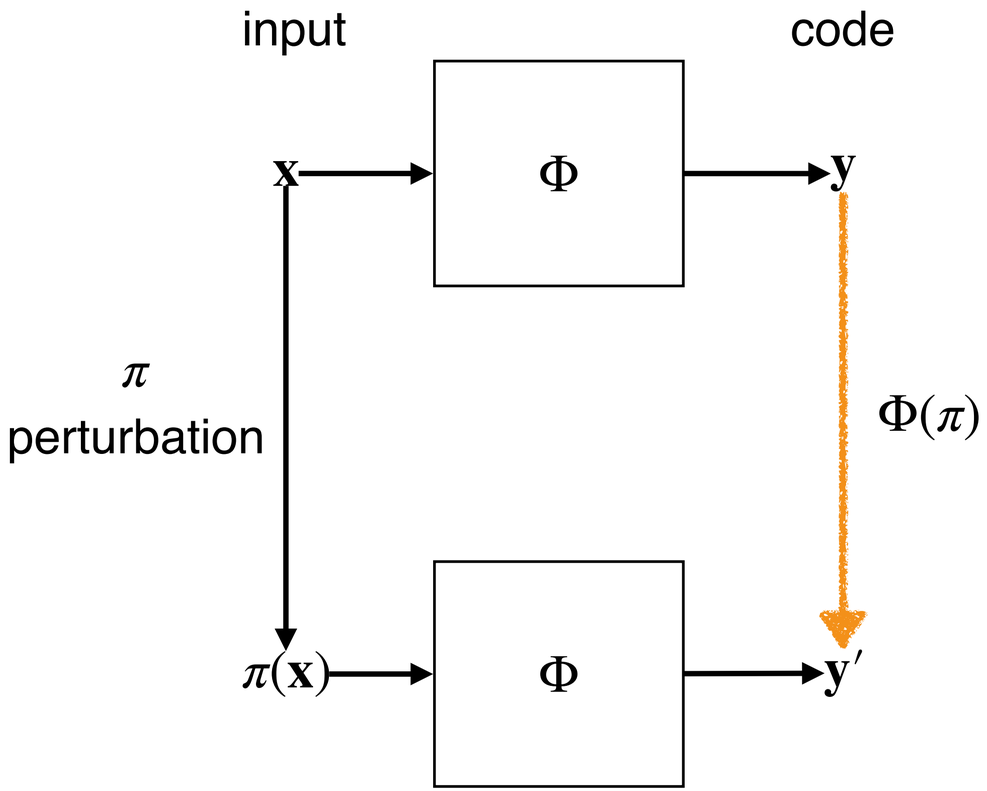
Gradients prove a local approximation of the model



The **gradient** can be directly interpreted as a **local linear approximation** of the model

However, all other saliency propagation rules do not have simple interpretations such as this

Towards a formal approach to explanations



Study how Φ(**x**) changes up to perturbations π(**x**) of the input **x**

Perturbation should be **meaningful** (interpretable). E.g:

- Injecting noise
- Rotating or translating the image
- Erasing parts of the image

The **representation** may

- Be **invariant** (stay the same)
- Be **equivariant** (respond predictably)

The **analysis** may be

- Local around **x** and π
- For a distribution p(**x**) and a fixed π
- For a distribution p(π) and a fixed **x**
- …

Meaningful perturbation analysis: saliency

Saliency via eliding objects

Original

"cat" probability
1.00

Redact-out

"cat" probability
0.5

(ineffective)

Blur-out

"cat" probability
0.01

(more meaningful)

We seek the "**smallest elision**" that maximally changes the neuron activation

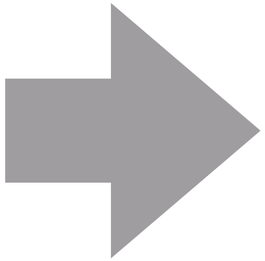
Meaningful perturbation analysis: saliency

Searching for the smallest elision mask via optimization

$\mathbf{x}$

$g_\sigma * \mathbf{x}$

m



$(1 - m) \odot \mathbf{x} + m \odot (g_\sigma * \mathbf{x})$

$$E(m) = \Phi_c((1 - m) \odot \mathbf{x} + m \odot (g_\sigma * \mathbf{x})) + \lambda \|m\|_1, \qquad 0 \leq m \leq 1$$

The energy rewards lowering the class score while using a small mask

Use SGD to optimize over the mask *m*

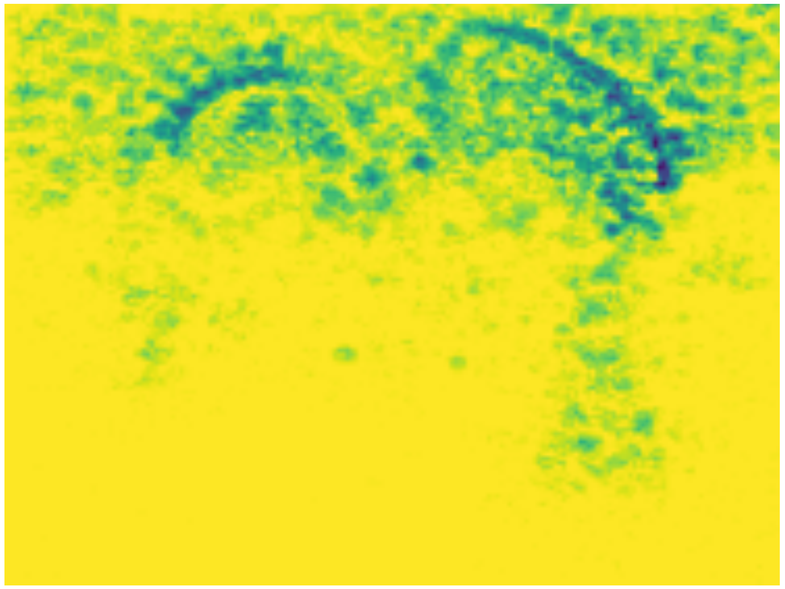
Adversarial *vs* meaningful perturbations

Looking beyond neural network artifacts

Original

Elision (to gray) result

Mask



Neural networks are **fragile** to **adversarial perturbations**

Adversarial perturbations attract gradient descent

Intriguing properties of neural networks. Szegedy, Zaremba, Sutskever, Bruna, Erhan, Goodfellow, Fergus. CoRR 2013

Adversarial *vs* meaningful perturbations

Looking beyond neural network artifacts

Adversarial elision

improbable in nature

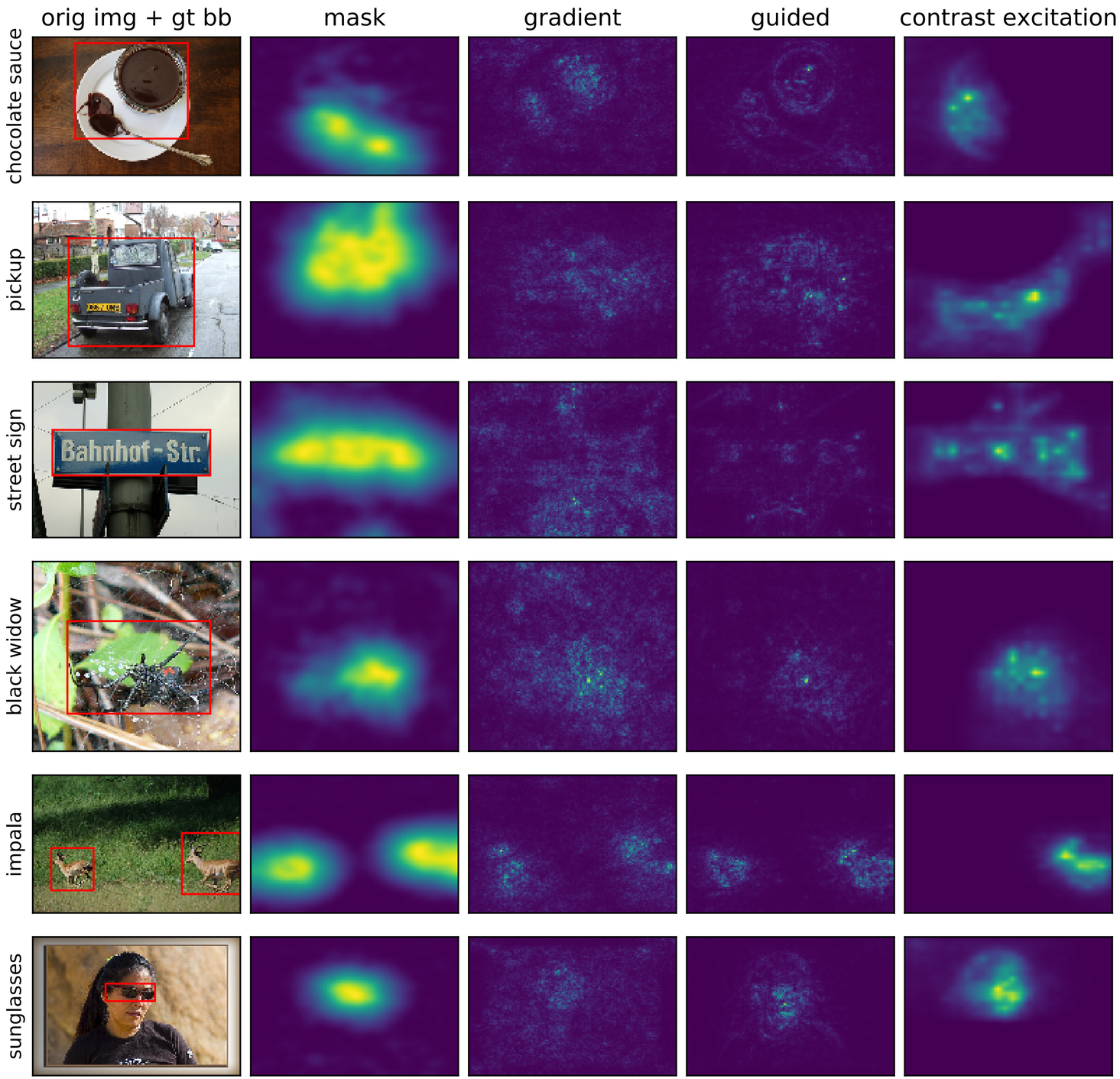
Meaningful elision

likely in nature

Regularization can help finding more meaningful perturbations

Examples: simplify the mask, look for the average effect of a pool of similar masks

Examples



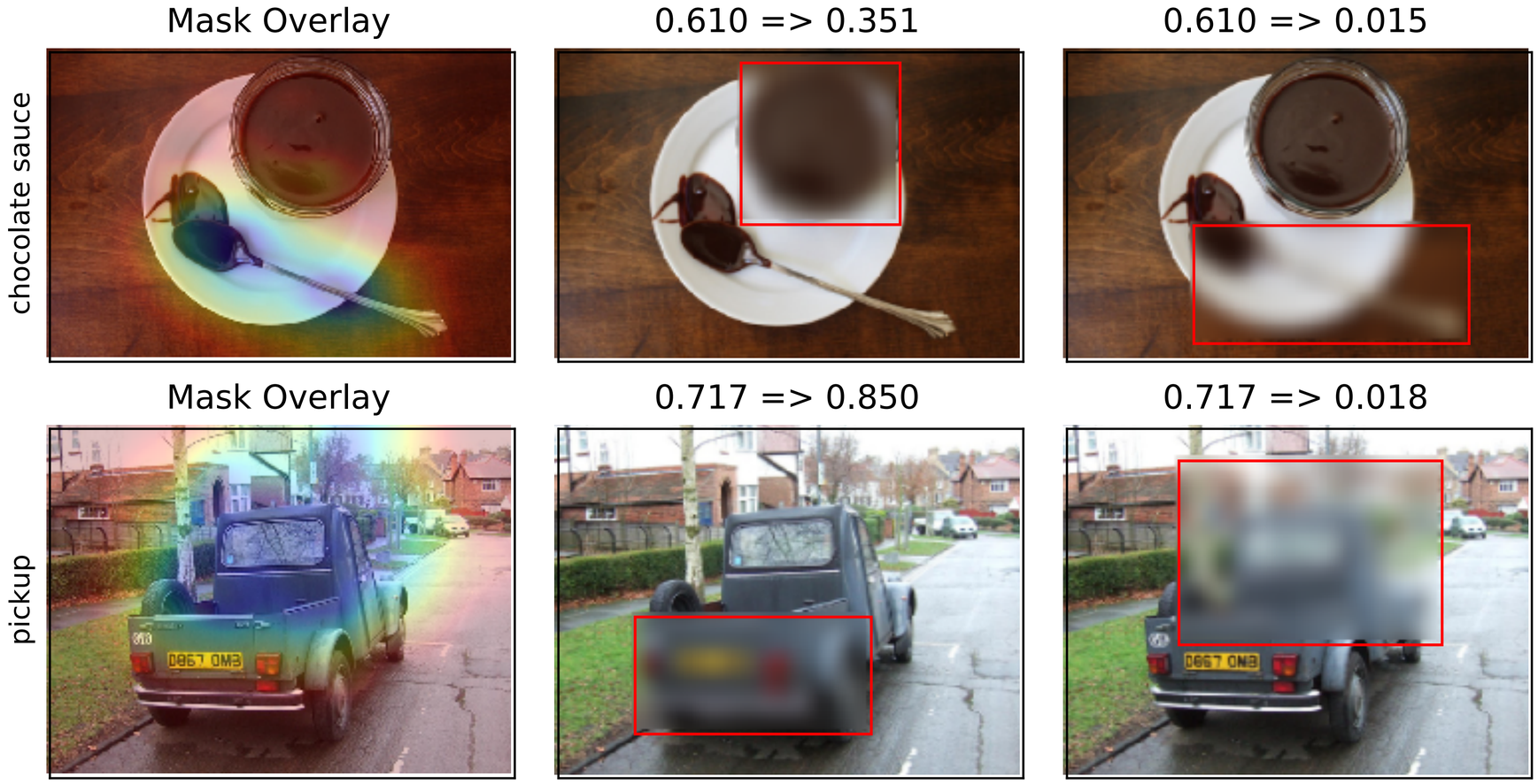
Meaningful minimal elisions

Crisp regions

Similar to gradient, meaning is “obvious” by definition

Interpretable explanations of black boxes by meaningful perturbation, Fong Vedaldi, CVPR, 2017

What is salient may not be meaningful



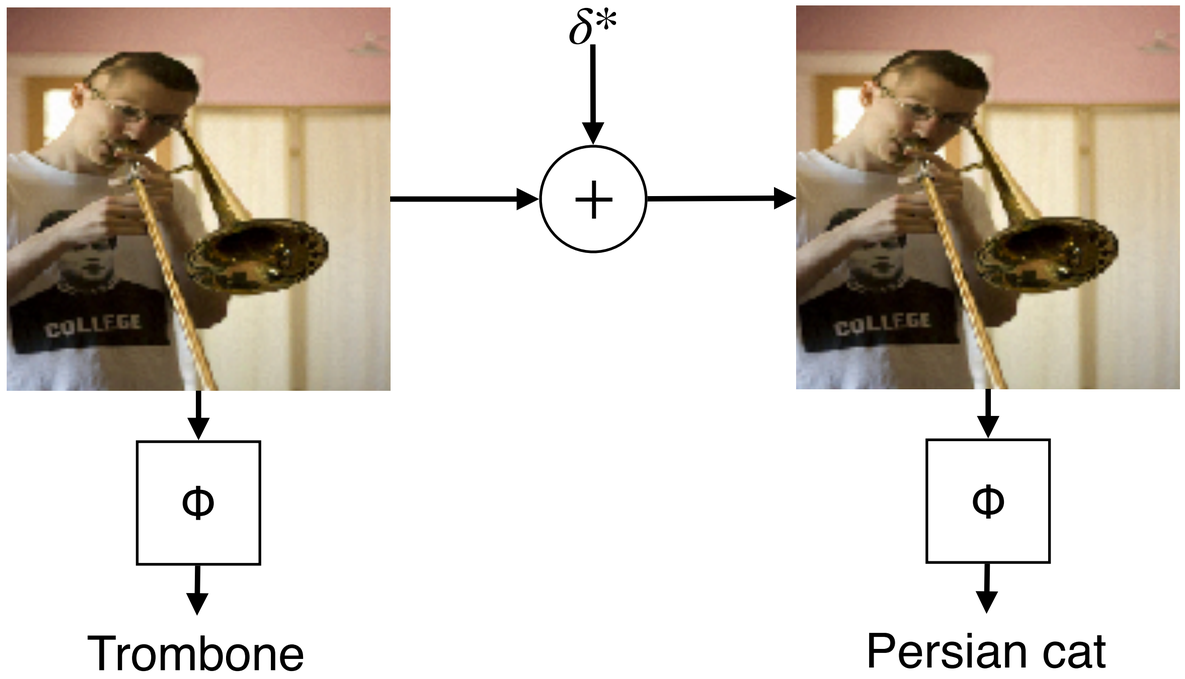
Example: the hot chocolate is recognized via the spoon and the truck vs the license plate

CNN fragility

Easily fooled by adversarial examples

Let **y** = Φ(**x**) be the label predicted for image **x** by the deep net

Empirically, we can find tiny perturbations **x** + δ that change **y** arbitrarily!



$$\delta^* = \underset{\|\delta\|<\epsilon}{\operatorname{argmin}} \|\mathbf{y}_{\text{arbitrary}} - \Phi(\mathbf{x}+\delta)\|$$

Intriguing properties of neural networks
Szegedy, Zaremba, Sutskever, Bruna, Erhan, Goodfellow, Fergus. CoRR, 2013

Dangerous adversaries

Advarsarial examples can be suffesfully “injected” in real life

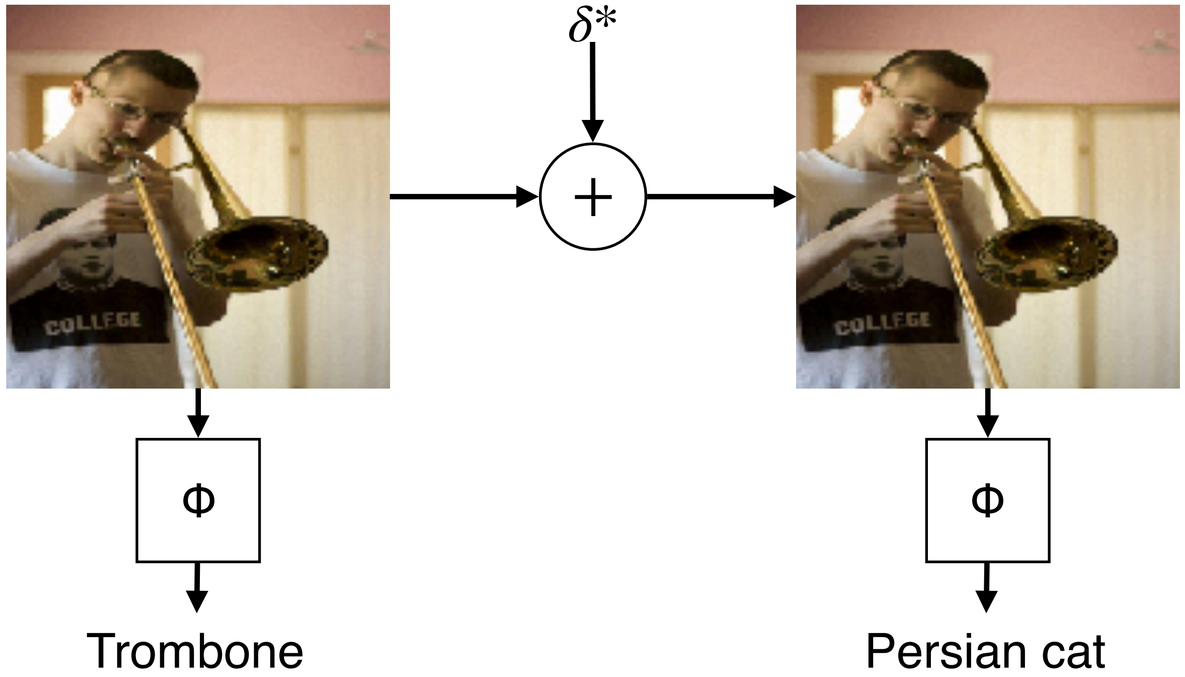
Adversarial glasses fooling face recognition

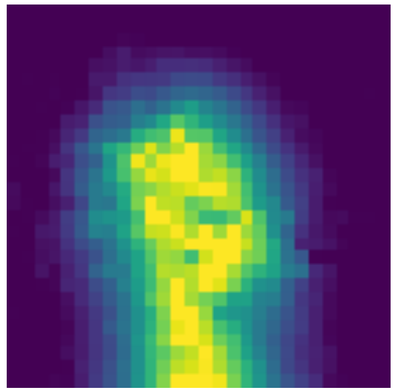
Adversarial stickers fooling sign recognition

Accessorize to a crime: Real and stealthy attacks on state-of-the-art face recognition. Sharif, Bhagavatula, Bauer, Reiter. CSS, 2016.

Robust physical-world attacks on machine learning models. Evtimov, Kevin Eykholt2, Li, Prakash, Rahmati, Song. arXiv, 2017.

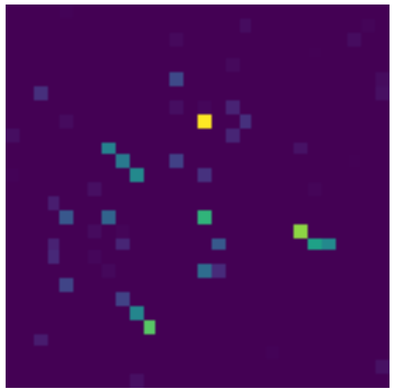
Adversarial defence



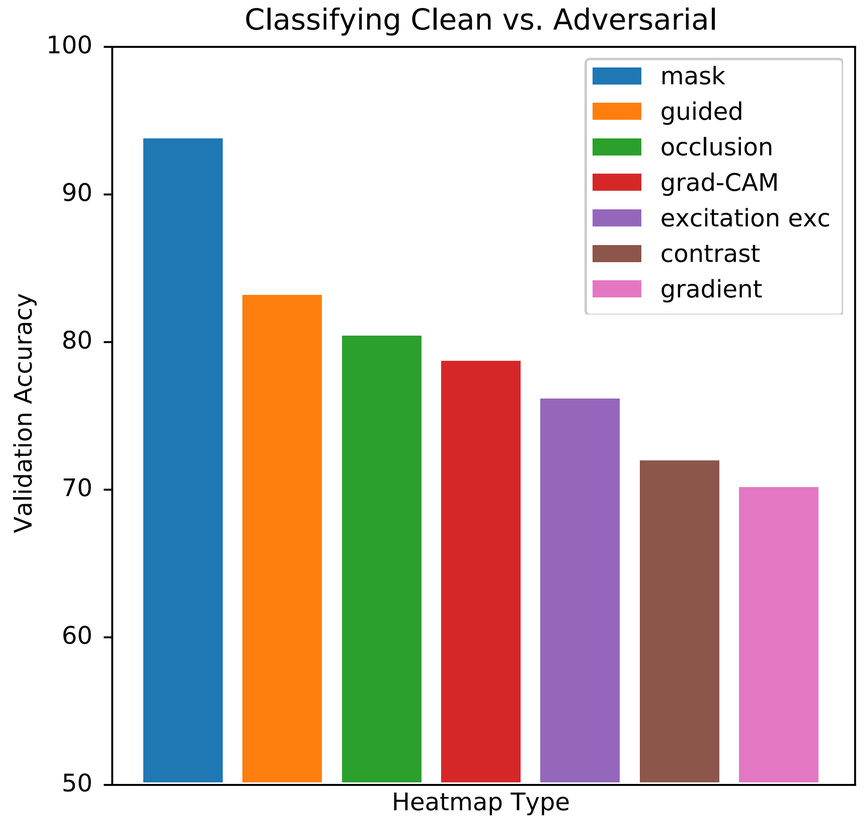


Trombone saliency

Perturbation analysis



Persian cat saliency

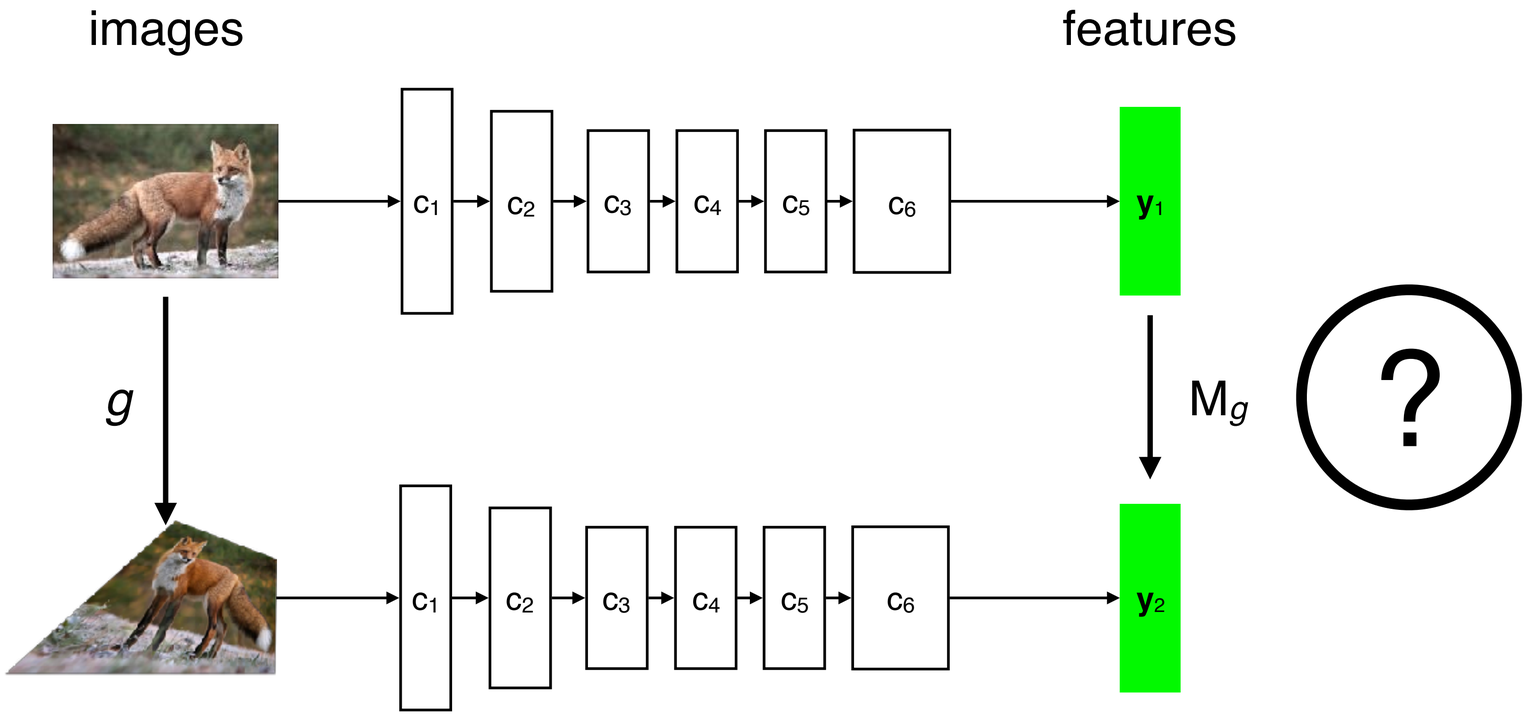


Method: recognize genunie vs adversarial images by learning a classifier on top of the saliency maps

(Illustrative of proerties of saliency, not really a recommended defense stragety!)

Equivariance

How is a representation affected by an image warp?

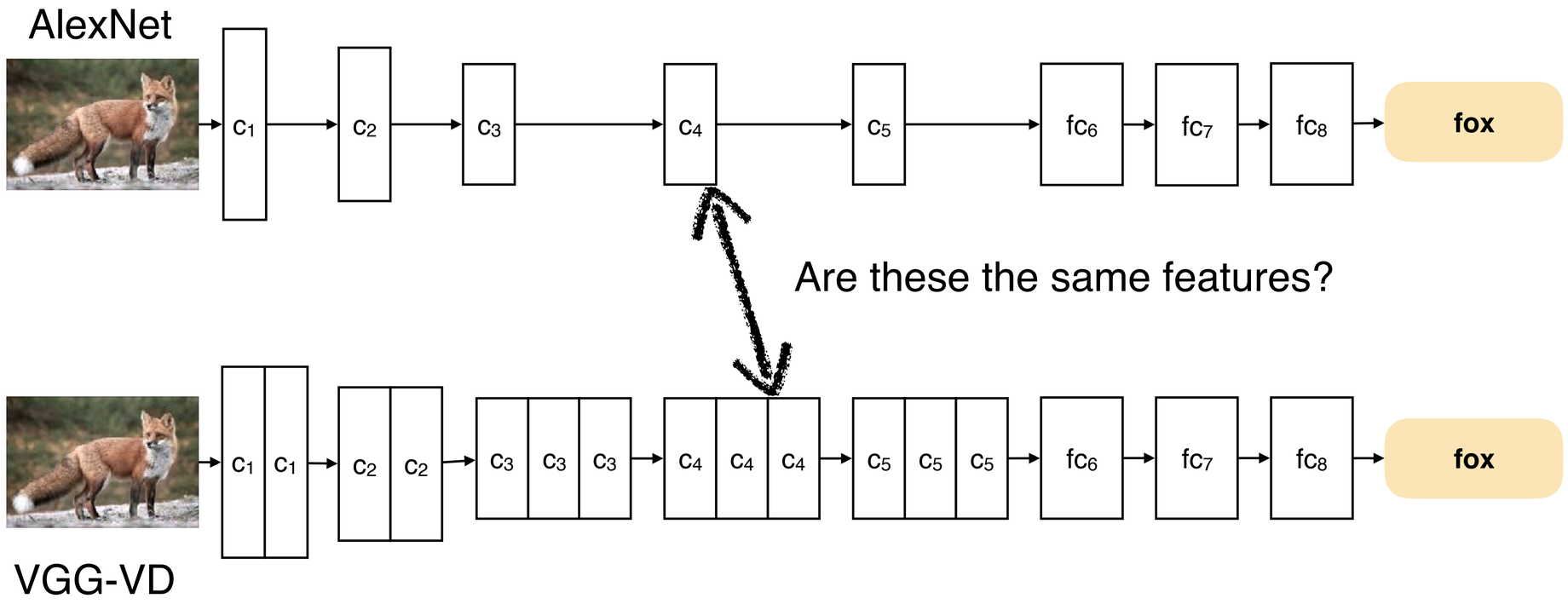


Short answer: warping image usually reduces to sparse linear tf in feature space.

Long answer: **Understanding image representations by measuring their equivariance and equivalence. Lenc Vedaldi**. CVPR 2015 & IJCV 2018

Equivalence

Are different neural networks “the same”?



Short answer: there generally are corresponding features in different networks (up to 1x1 linear tfs).

Long answer: **Understanding image representations by measuring their equivariance and equivalence. Lenc Vedaldi**. CVPR 2015 & IJCV 2018

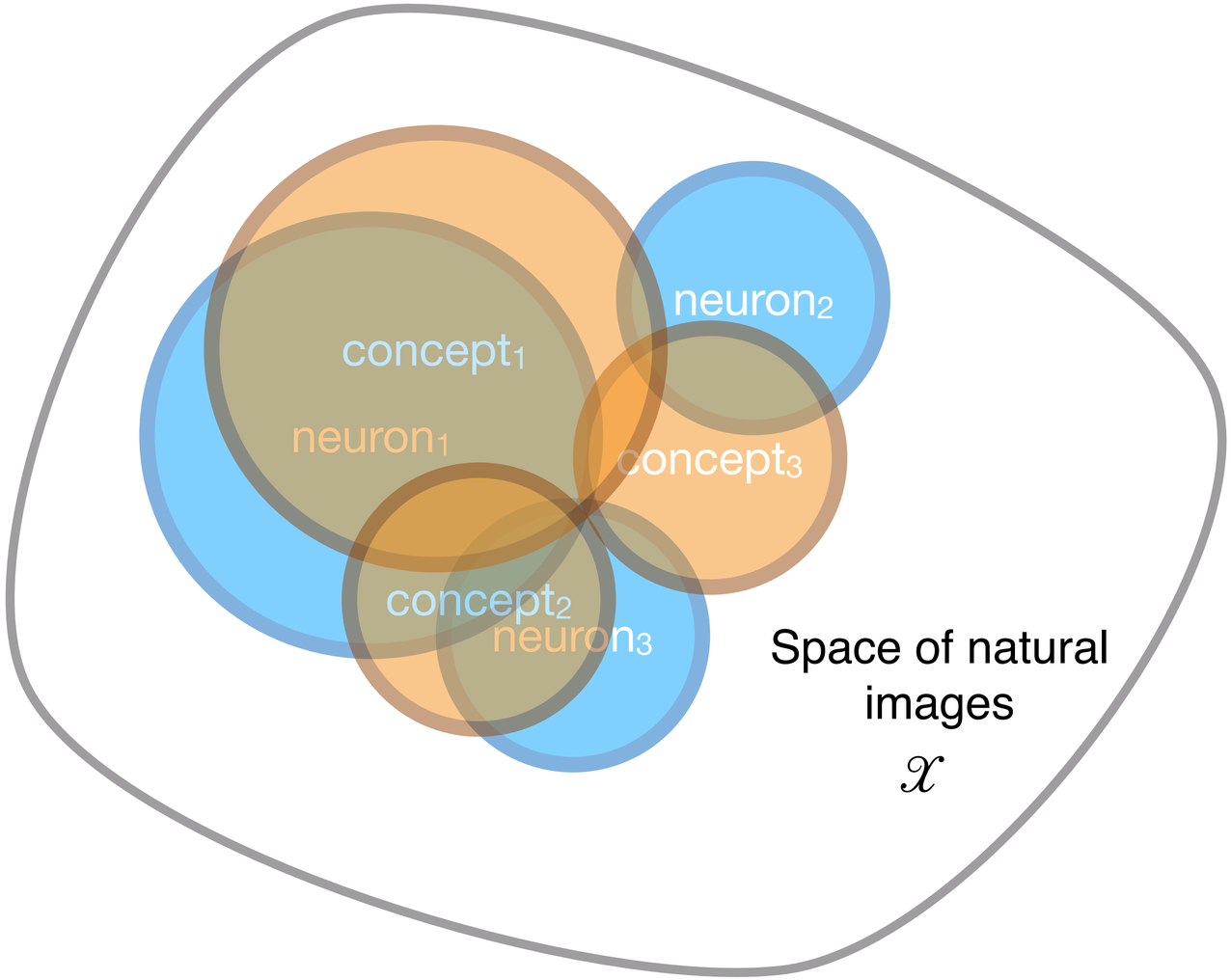
Generating iconic examples

Attribution

Semantic identification

Neurons and concepts: equivalence classes of images

Each neuron / concept "activates" for a subset of natural images (patches)



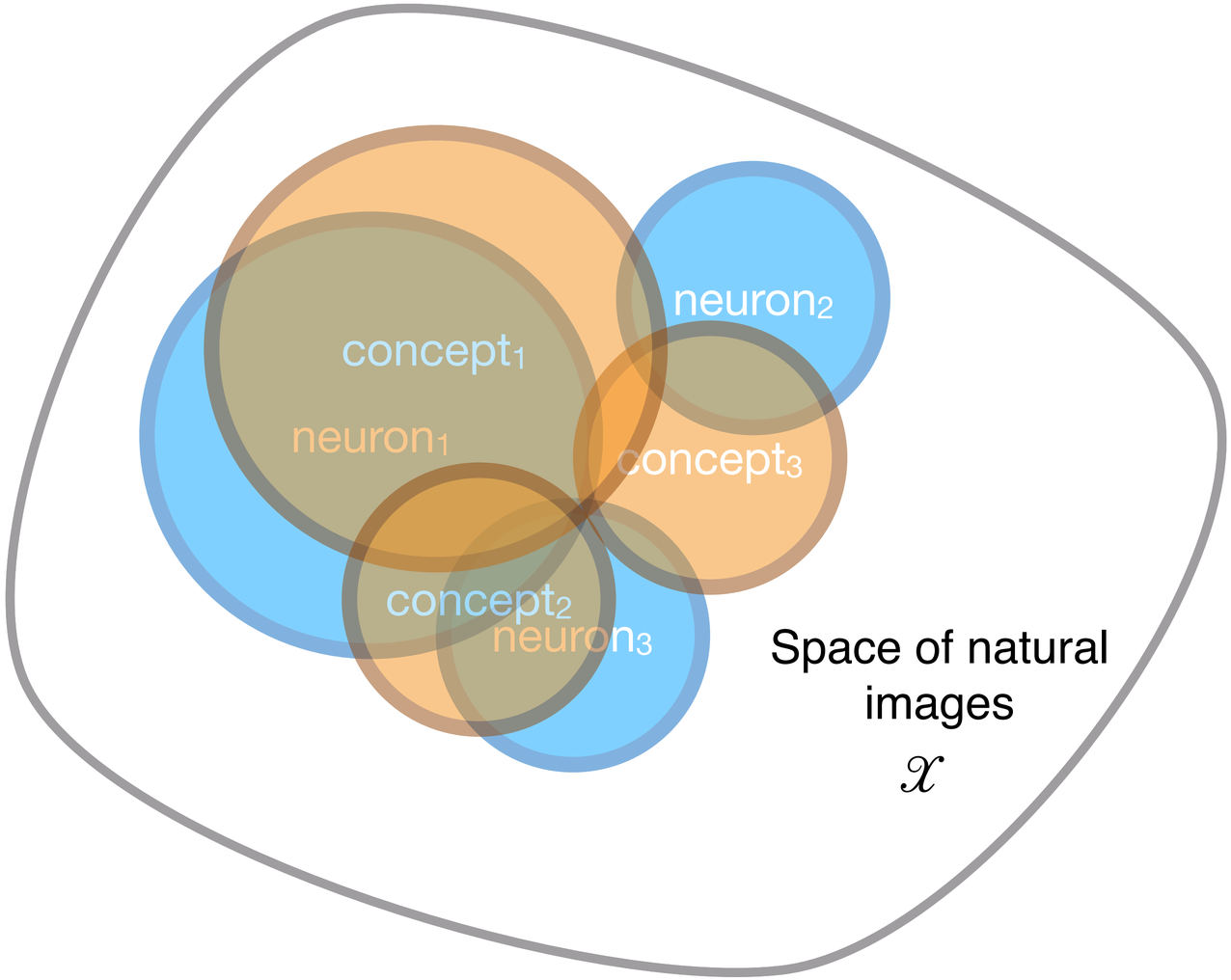
Assume that neurons have binary activation and that concepts apply deterministically

$$\text{concept}_i = \{\mathbf{x} \in \mathcal{X} : \text{concept}_i(\mathbf{x}) = \text{true}\}$$

$$\text{neuron}_k = \{\mathbf{x} \in \mathcal{X} : \Phi_k(\mathbf{x}) = 1\}$$

Neurons and concepts: equivalence classes of images

Each neuron / concept "activates" for a subset of natural images (patches)



Assume that neurons have binary activation and that concepts apply deterministically

$$\text{concept}_i = \{\mathbf{x} \in \mathcal{X} : \text{concept}_i(\mathbf{x}) = \text{true}\}$$

$$\text{neuron}_k = \{\mathbf{x} \in \mathcal{X} : \Phi_k(\mathbf{x}) = 1\}$$

Questions:

- Do neurons and concepts correspond one-to-one?
- How many neurons are required to express a concept?
- How many concepts are required to express a neuron?

Identifying the meaning of neuron responses

Reading list

Analyzing the performance of multilayer neural networks for object recognition
Agrawal, Girshick, Malik. ECCV, 2014

Correlates filters with ImageNet **patches**

Object-centric representation learning from unlabeled videos.
Gao, Jayaraman, Grauman, ACCV, 2016

Identifies the **semantics** of some convolutional filters

Places: An image database for deep scene understanding
Zhou, Khosla, Lapedriza, Torralba, Oliva. PAMI, 2016

BRODEN, fine-grained **semantic** of **individual filters**

Network dissection: Quantifying interpretability of deep visual representations
Bau, Zhou, Khosla, Oliva, Torralba. CVPR, 2017

Understand training task performance

Understanding intermediate layers using linear probes
Alain Bengio. ICLR Workshop, 2017

Learn **linear predictor** for **diagnostics**

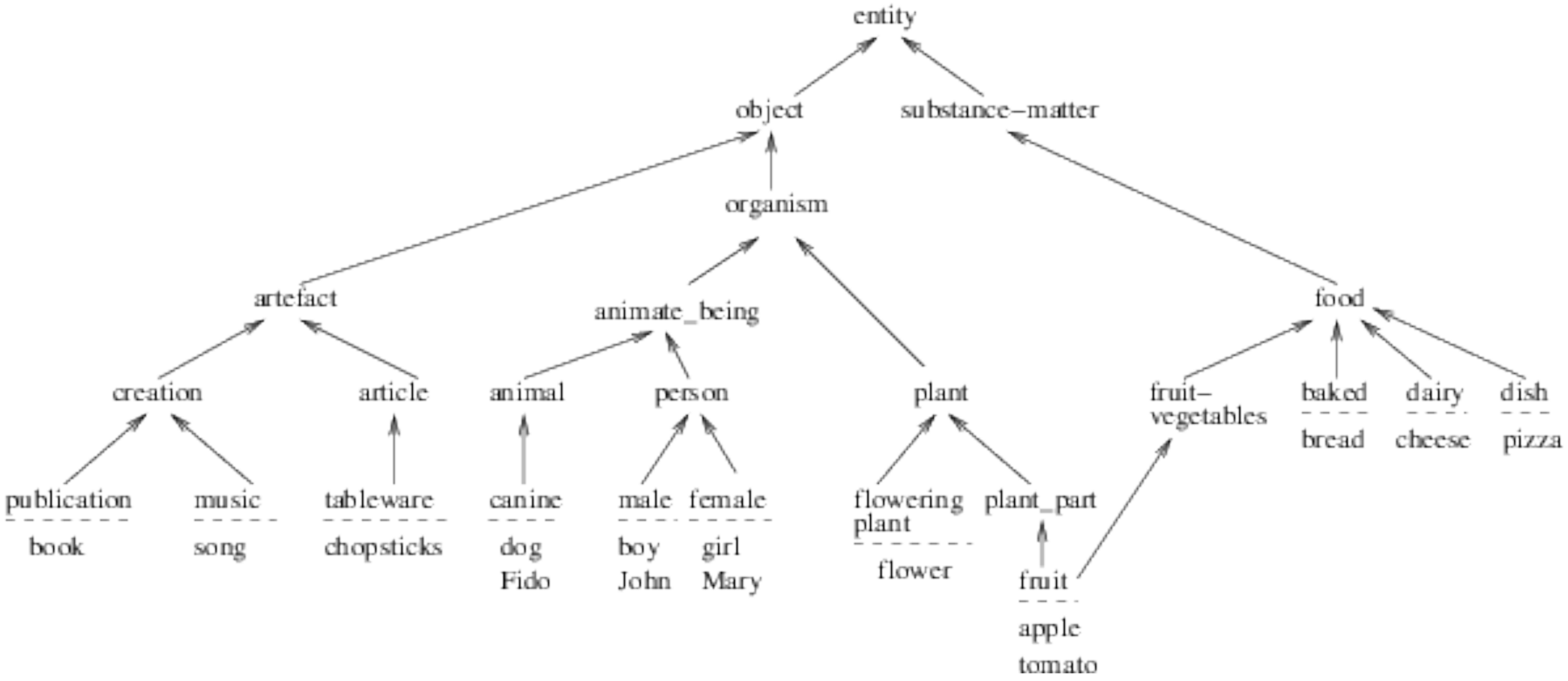
Revisiting the importance of individual units via ablation
Zhou He Bau Torraba, arXiv 2018

Relation between interpretability and classification performance

Concepts as a structured abstraction

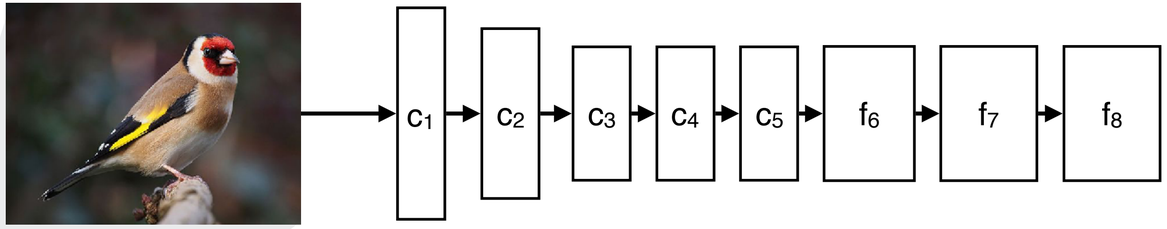
Beyond just responding to images

Concepts form a structured space. For example, WordNet induces an **is-a hierarchy**:



Concepts as a structured abstraction

Net2vec associates a linear concept space to a network



Φ

Space of natural images

wing

Train a classifier from a network slice Φ

$$\text{is-a-wing}(\mathbf{x}) \approx \langle \mathbf{w}_{\text{wing}}, \Phi(\mathbf{x}) \rangle$$

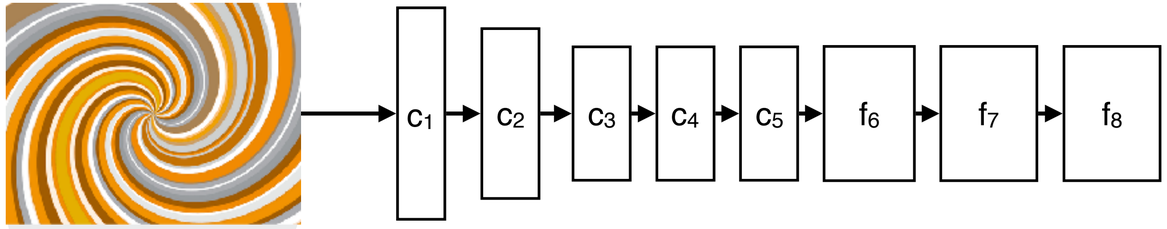
Space of classifier functions

$\mathbf{w}_{\text{wing}}$

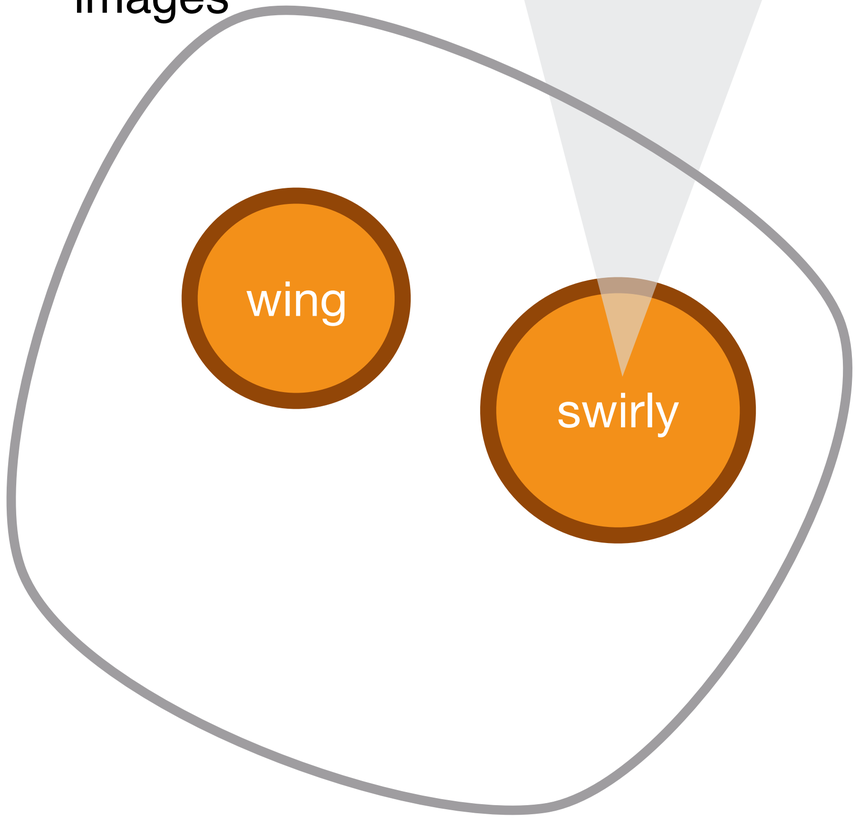
Net2Vec: Quantifying and explaining how concepts are encoded by filters in deep neural networks. Fong Vedaldi. CVPR, 2018

Concepts as a structured abstraction

Net2vec associates a linear concept space to a network



Space of natural images

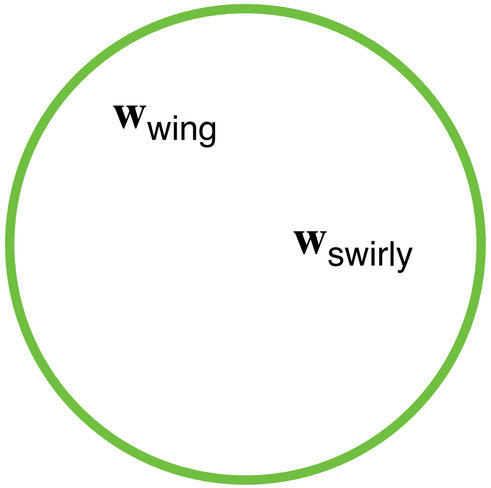


Train a classifier from a network slice Φ

$$\text{is-a-wing}(\mathbf{x}) \approx \langle \mathbf{w}_{\text{wing}}, \Phi(\mathbf{x}) \rangle$$

$$\text{is-swirly}(\mathbf{x}) \approx \langle \mathbf{w}_{\text{swirly}}, \Phi(\mathbf{x}) \rangle$$

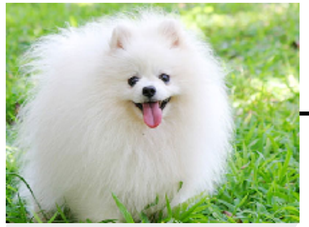
Space of classifier functions

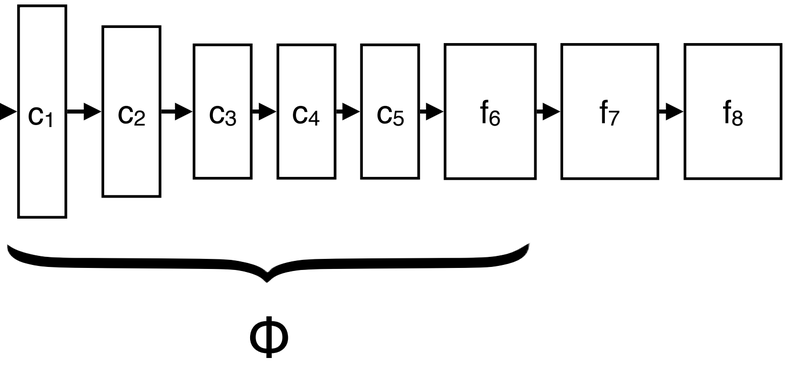


Net2Vec: Quantifying and explaining how concepts are encoded by filters in deep neural networks. Fong Vedaldi. CVPR, 2018

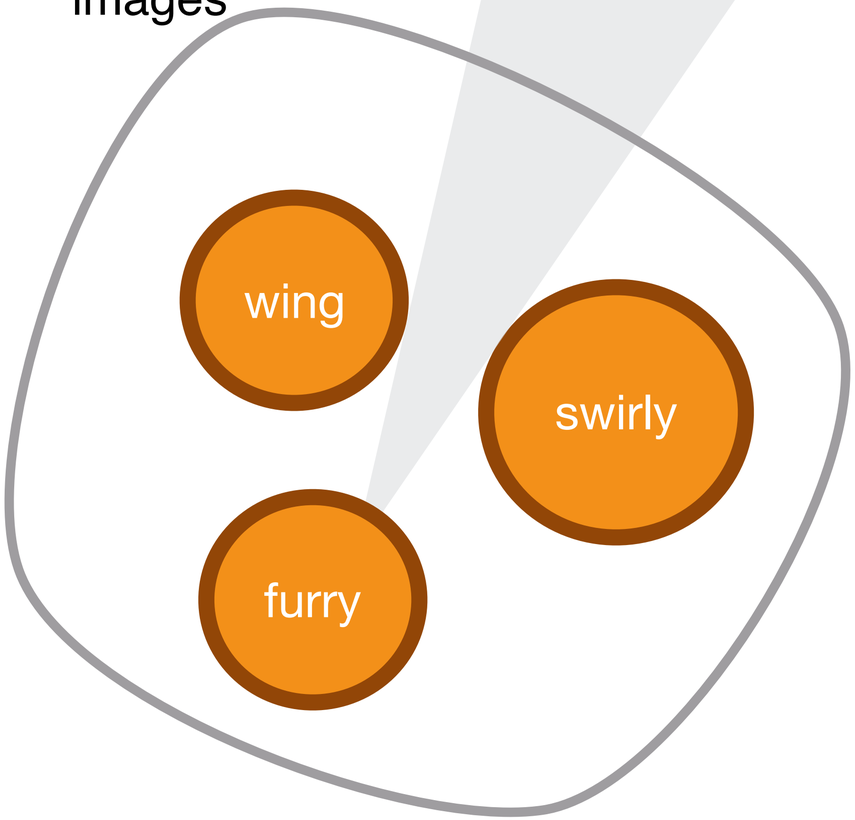
Concepts as a structured abstraction

Net2vec associates a linear concept space to a network





Space of natural images



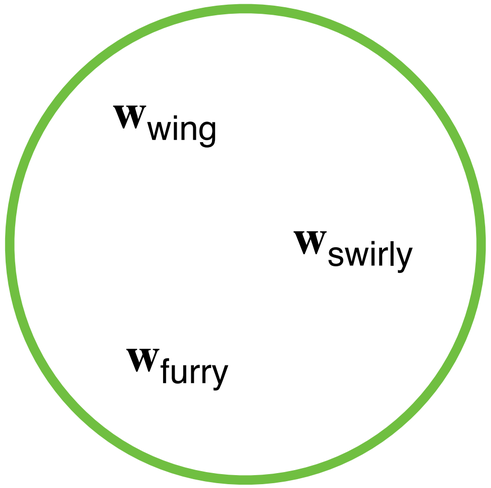
Train a classifier from a network slice Φ

$$\text{is-a-wing}(\mathbf{x}) \approx \langle \mathbf{w}_{\text{wing}}, \Phi(\mathbf{x}) \rangle$$

$$\text{is-swirly}(\mathbf{x}) \approx \langle \mathbf{w}_{\text{swirly}}, \Phi(\mathbf{x}) \rangle$$

$$\text{is-furry}(\mathbf{x}) \approx \langle \mathbf{w}_{\text{furry}}, \Phi(\mathbf{x}) \rangle$$

Space of classifier functions



Net2Vec: Quantifying and explaining how concepts are encoded by filters in deep neural networks. Fong Vedaldi. CVPR, 2018

BRODEN Dataset

Thousands of images annotated with hundreds of concepts, often densely

Image-level Annotations

street (scene)

swirly (texture)

Pixel-level Annotations

flower (object)

pink (color)

headboard (part)

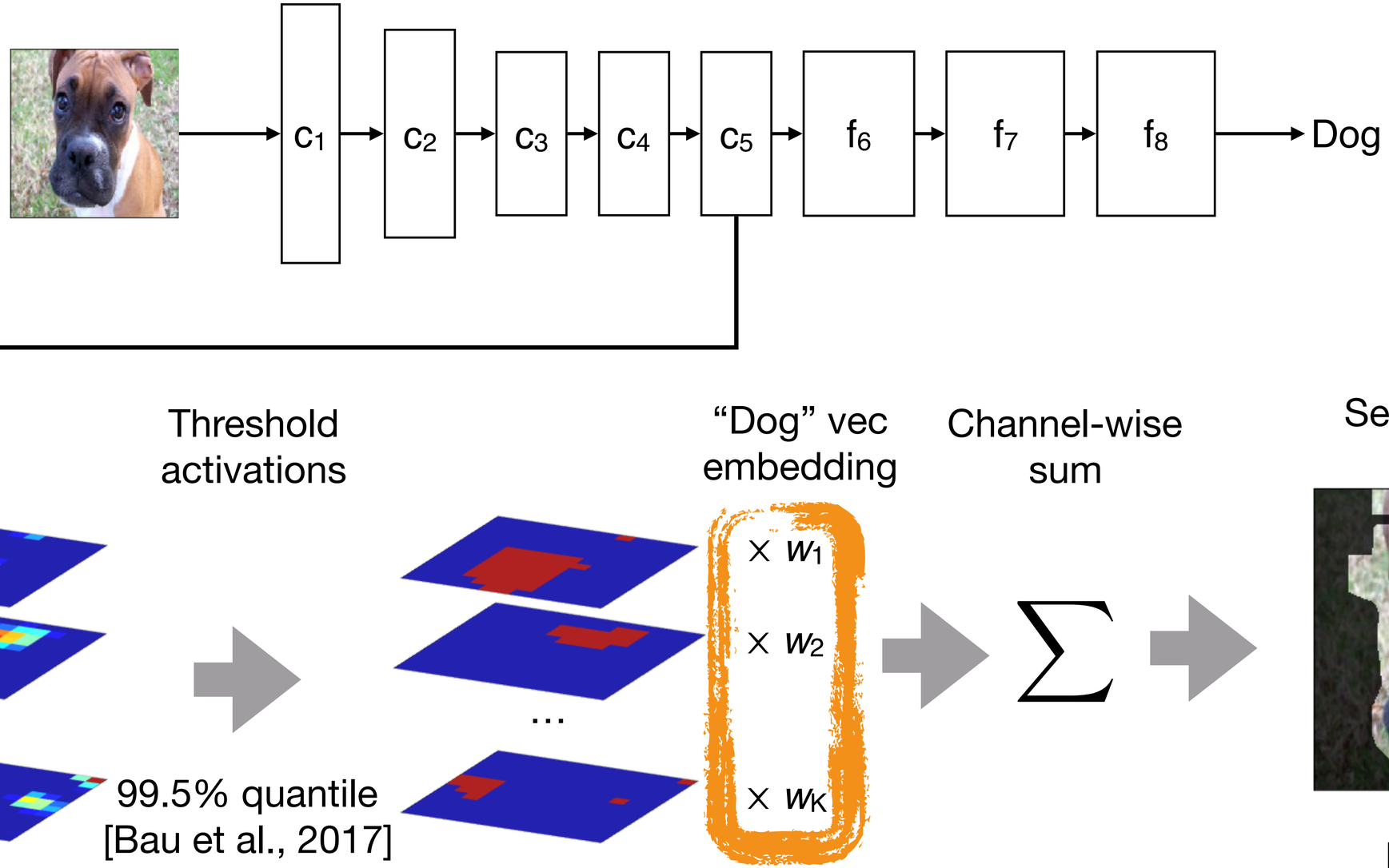
metal (material)

Network dissection: Quantifying interpretability of deep visual representations

Bau, Zhou, Khosla, Oliva, Torralba. CVPR, 2017

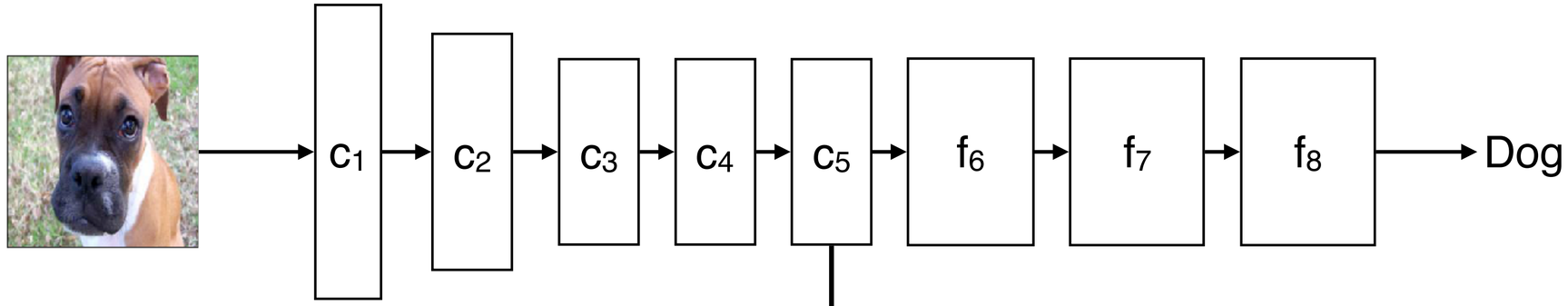
Net2vec: building concept embeddings

Vecs: pixel-wise linear predictor of a concept



Net2vec: building concept embeddings

Sparse vets (only a few neurons)

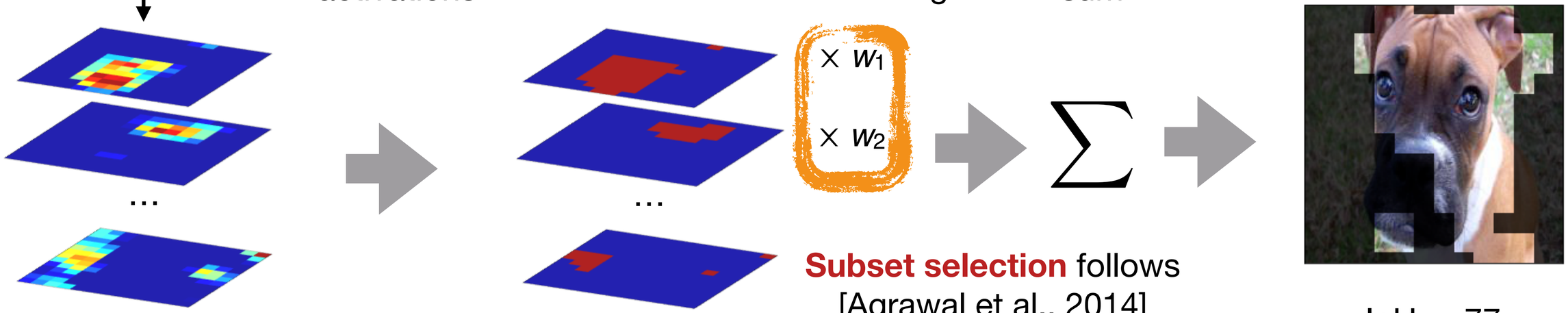


Threshold activations

“Dog” vec embedding

Channel-wise sum

Segmentation mask

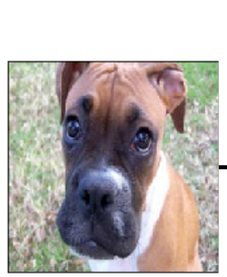


Subset selection follows [Agrawal et al., 2014]

IoU = .77

Net2vec: building concept embeddings

Singleton vests (only one neuron)



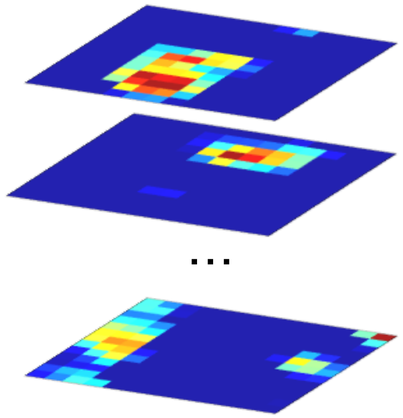
c_1 → c_2 → c_3 → c_4 → c_5 → f_6 → f_7 → f_8 → Dog

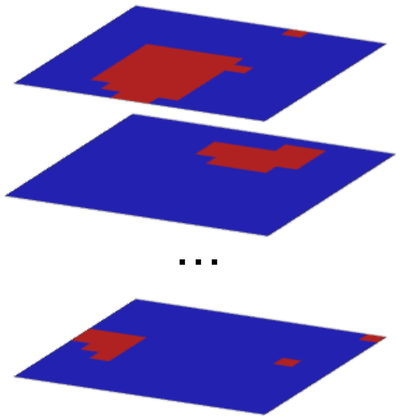
Threshold activations

“Dog” vec embedding

Channel-wise sum

Segmentation mask





× w_1

× w_2

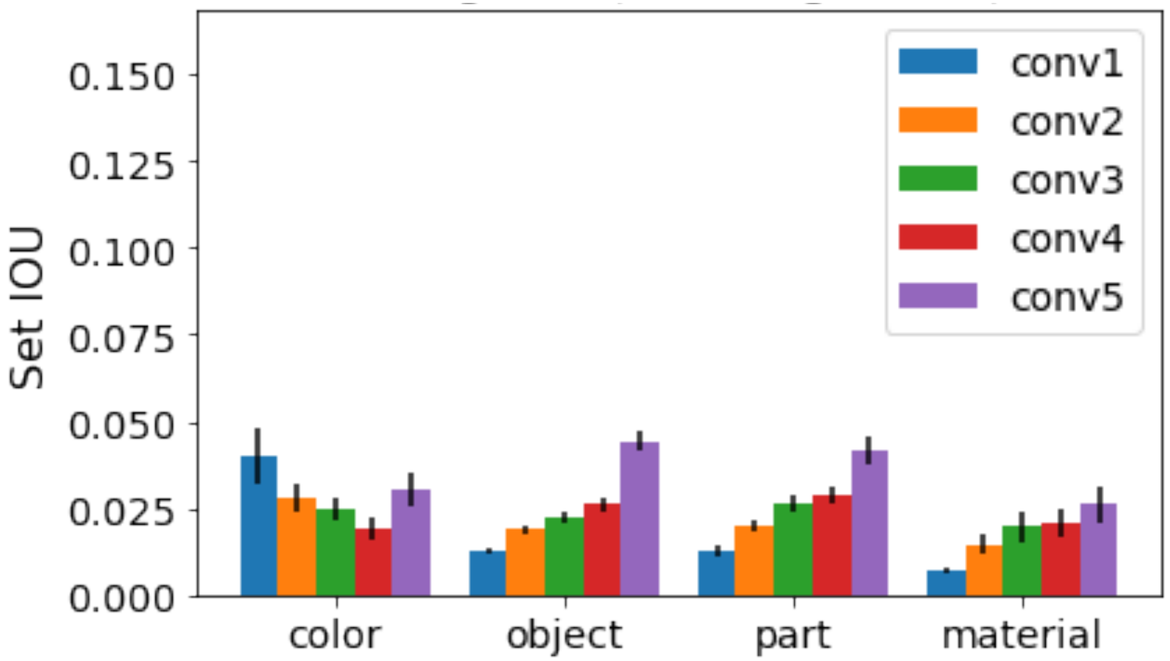
$\sum$

Best filter predictor
~ [Bau et al., 2017]

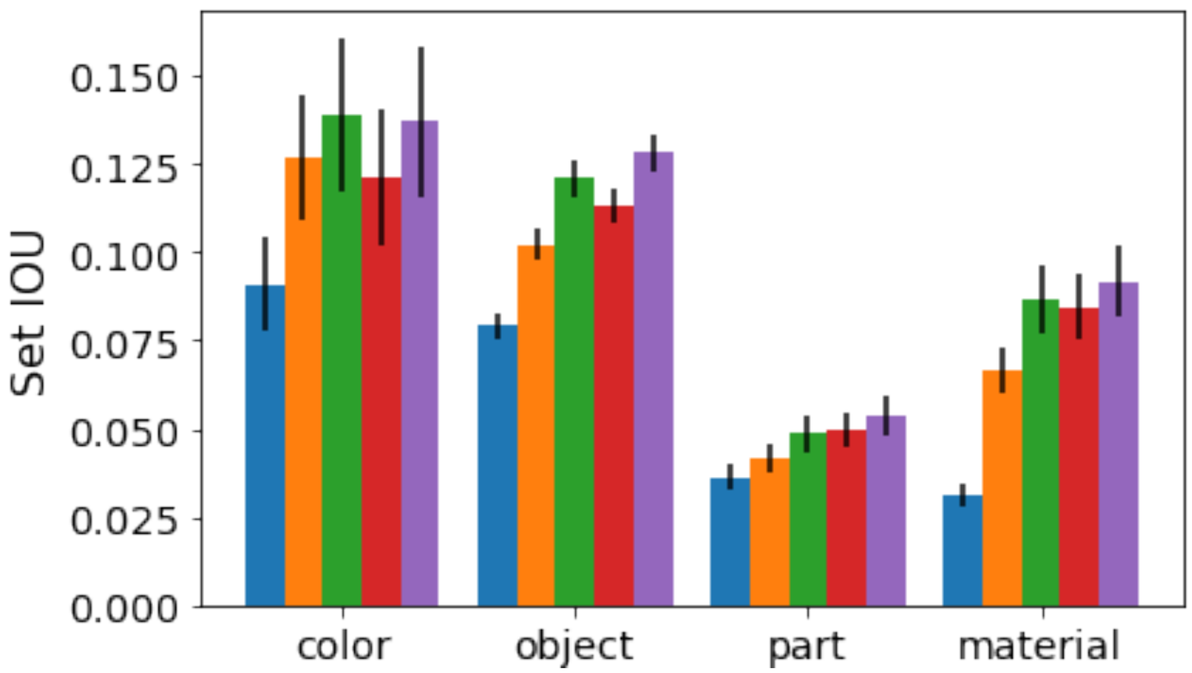
IoU = .77

Number of feature channels per concept

Single channel



All channels

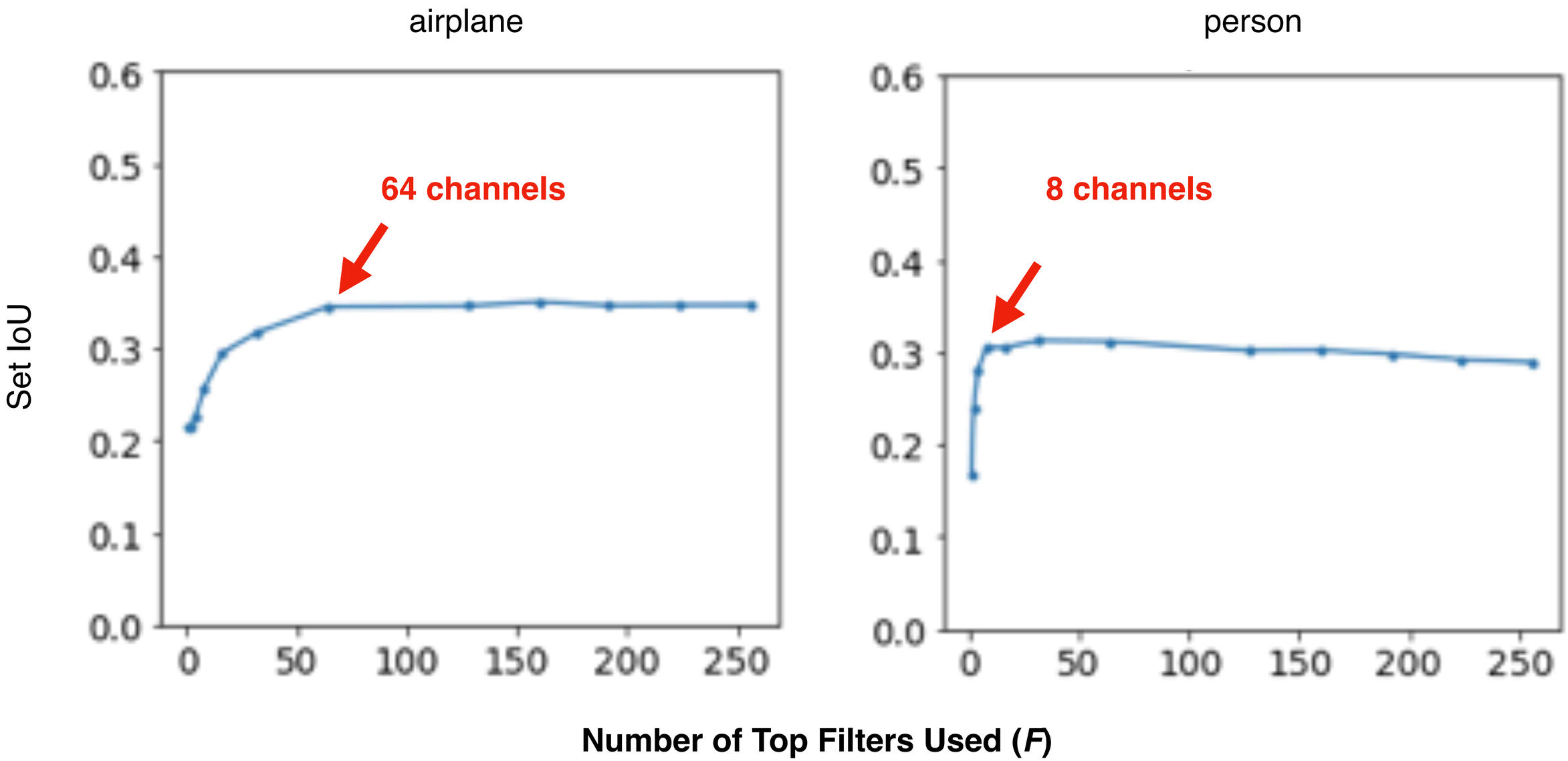


Observation: using more than one channel performs much better for most concepts

Individual neuron do not “isolate” concepts

Number of channels per concept

Depends strongly on the concept, even at similar level of abstractions



Concepts may need to be combined to explain neurons

Neuron may correspond to concept combinations, or to “unknown” concepts

Sheep
(IoU_{set} = .21)

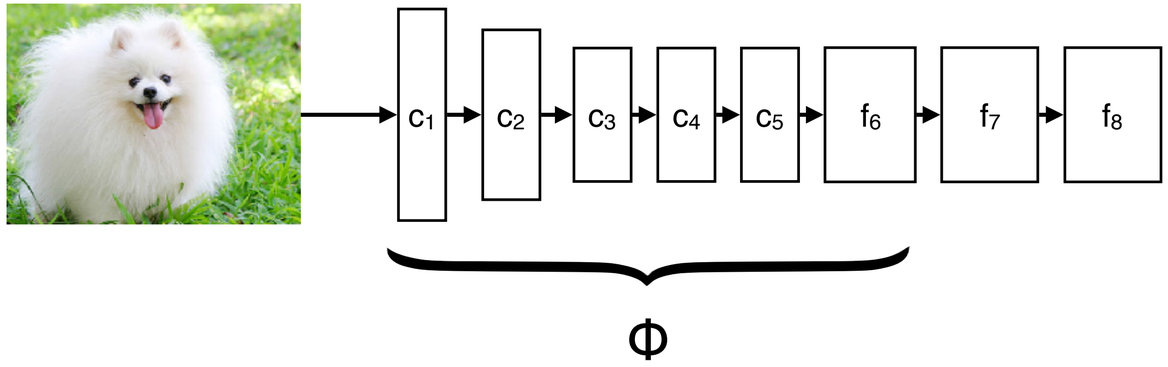
Horse
(IoU_{set} = .21)

Cow
(IoU_{set} = .20)

AlexNet conv5 66 is highly selective for multiple **farm animals**

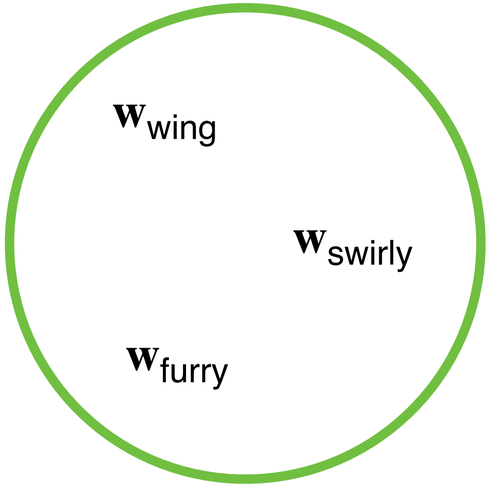
Exploring the structure of the learned concept space

Net2vec associates a linear concept space to a network



Net2vec

Space of classifier functions



The representation induces a similarity between concepts

$$[K_\Phi]_{ij} = \langle \mathbf{w}_{\text{concept}_i}, \mathbf{w}_{\text{concept}_j} \rangle$$

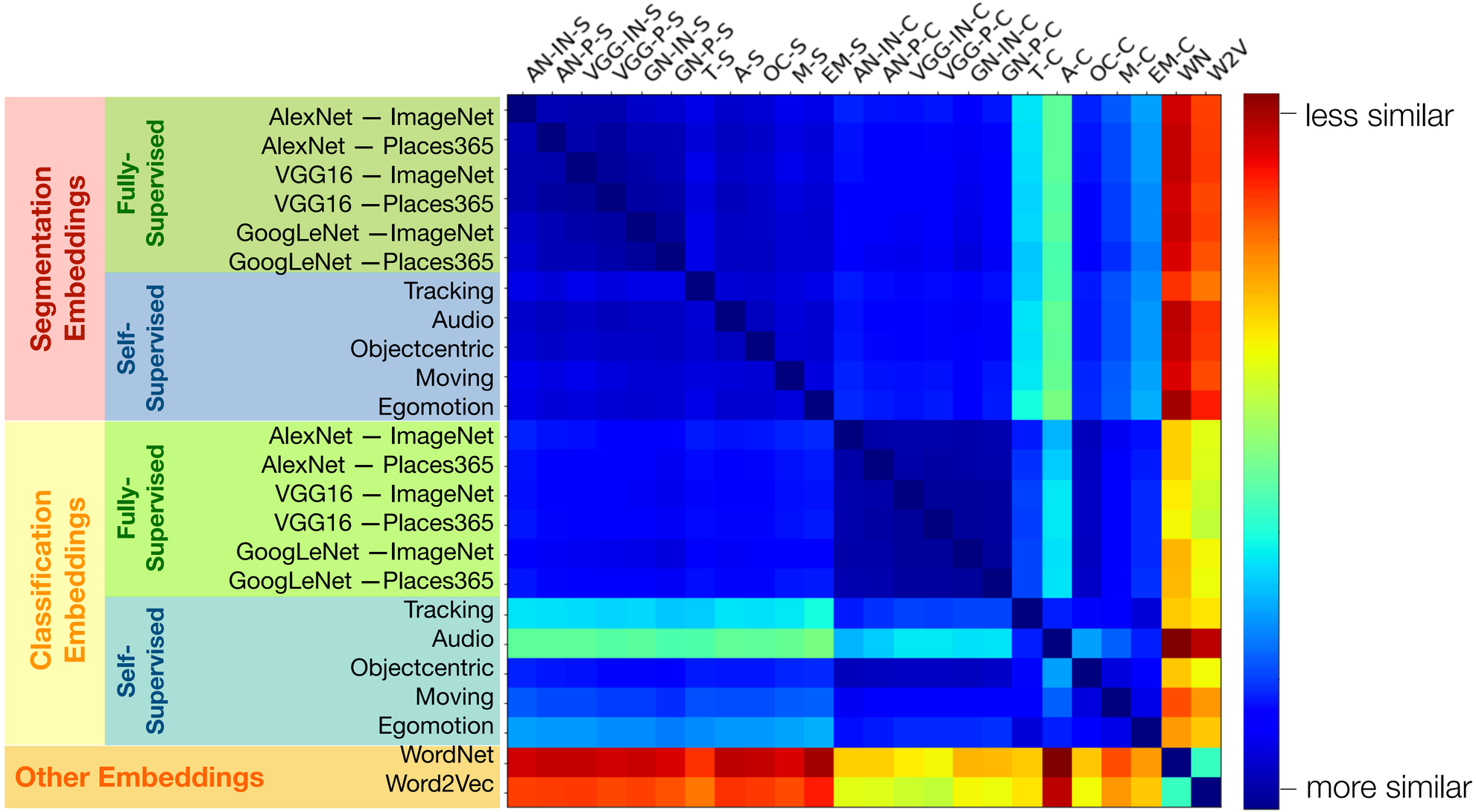
Structure = kernel in the space of classifier functions

We can now compare "conceptualizations"

$$\text{similarity}(\Phi, \Psi) = \frac{\sum_{ij} [K_\Psi]_{ij} [K_\Phi]_{ij}}{\sqrt{\sum_{ij} [K_\Psi]_{ij}^2} \cdot \sqrt{\sum_{ij} [K_\Phi]_{ij}^2}}$$

Comparing “conceptualizations”

AlexNet vs VGG16 vs GoogleNet vs Self-supervised vs Word Embeddings



Summary

Universal Representation

- Compact representation families

Unsupervised Representation

- Self-supervision for learning features
- Self-supervision for learning structure
- What’s in the prior

Understandable Representations

- Iconic visualizations
- Attribution
- Semantic identification

Acknowledgments


Karel Lenc


David Novotny


A. Mahendran

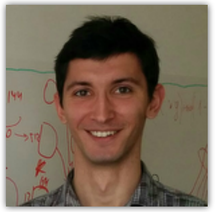

Hakan Bilen


Joao Henriques


James Thewlis


Ankush Gupta


Sam Albanie


Dmtry Ulyanov


Ruth Fong


S. Rebuffi


S. Ehrhardt


Lukas Neuman


Maria Klodt


Fatma Guney

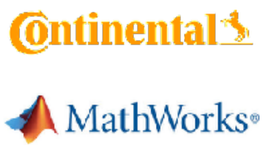

Oliver Groth


Xu Ji


Tomas Jakab


IDIU (Starting Grant)


SeeBiByte (Programme Grant)



Unsupervised, universal and understandable deep learning

While many would argue that recent progress in deep learning has allowed machines to approach and sometimes surpass human intelligence, this is true only in a very limited sense. In practice, machines are still far behind humans in many ways. For example, where humans learn universal models, applicable to a staggering variety of diverse problems, machines only learn narrow ones, with limited generalization capability. Furthermore, where humans can learn from raw data, machines still require explicit data annotations to perform well. In this talk, I will discuss our recent progress in universal and unsupervised learning. I will show how deep networks can generalize across apparently very diverse domains by means of residual adapters. I will also introduce the idea of factor learning and how this can be used to learn the geometry of objects by means of random data transformations.

I will also argue that, besides intrinsic limitations in the capabilities of machine learning algorithms and models as such, the area of machine learning is also limited in the sense that we still do not have a great deal of understanding of what machine learns to do. This is particularly important in safety-centric applications such as medical data processing, where the cost of machine errors to the public can be very high. I will summarize some of the current work in visualizing and understanding deep networks, focusing in particular on the techniques of natural pre-image and meaningful perturbations, and what we can (and cannot) learn from such an analysis.