Machine Learning for Design

Lecture 4

Machine Learning for Images. *Part 2*

How do humans see?

Hubel and Wiesel, 1959

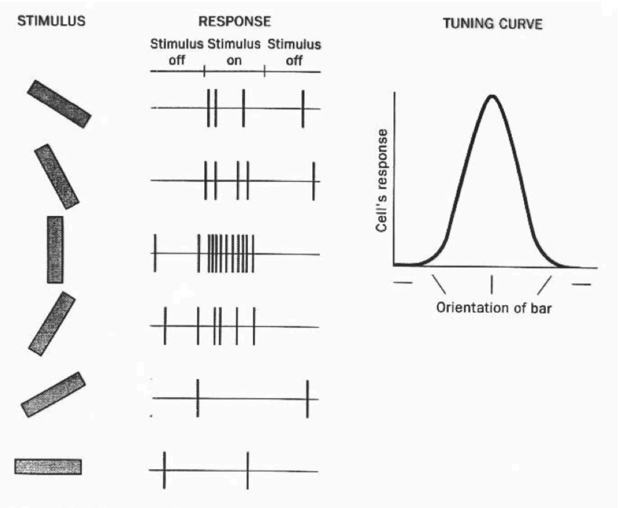
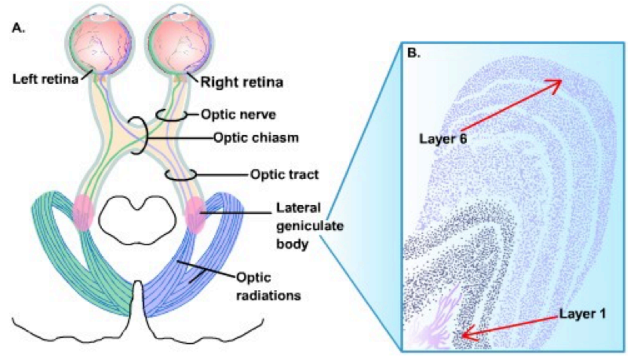


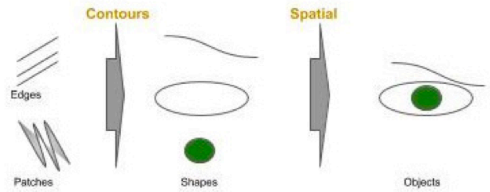
FIGURE 4.8 Response of a single cortical cell to bars presented at various orientations.

Neural Pathways



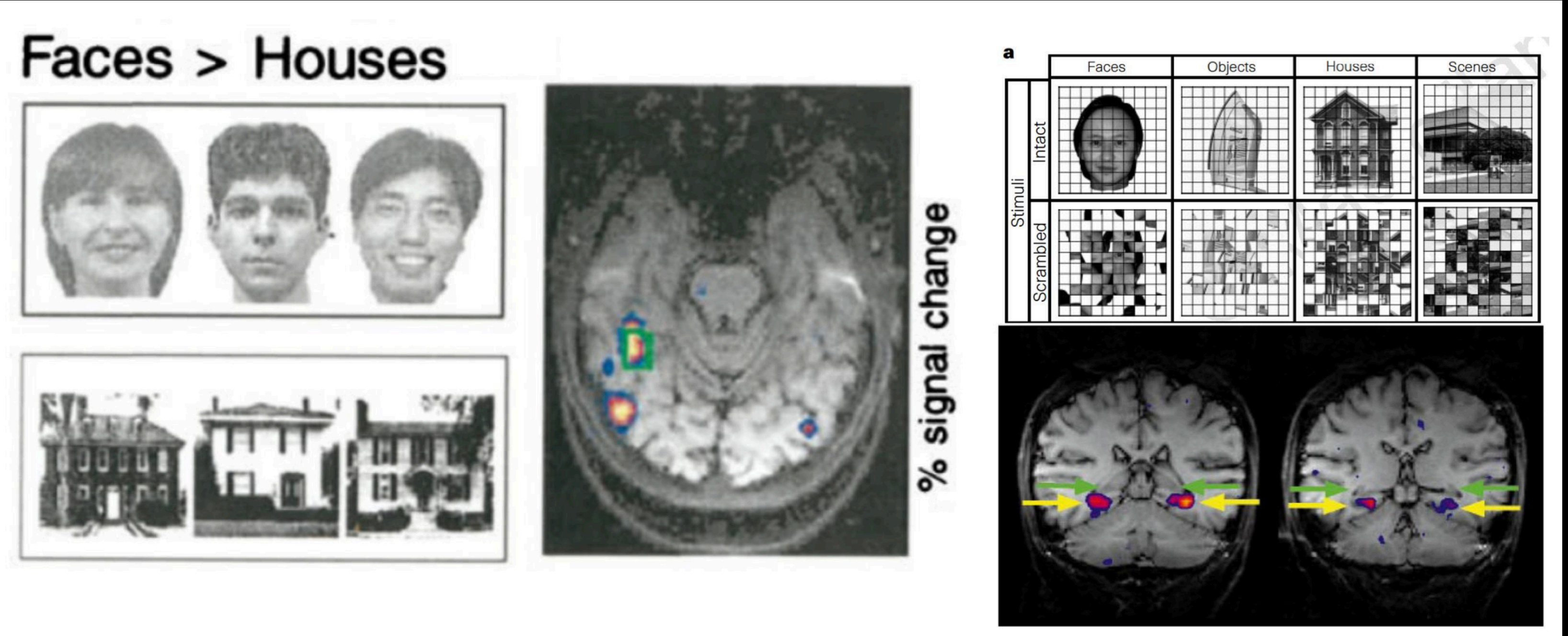
Edges → **Simple Shapes** → **Complex Shapes** → **Faces and Objects**

← **Lower layers** **Upper layers** →



Neural Correlation

of Objects & Scene Recognition



Kanwisher et al. J. Neuro. 1997

Epstein & Kanwisher, Nature, 1998

Why is machine vision hard?

The deformable and truncated cat

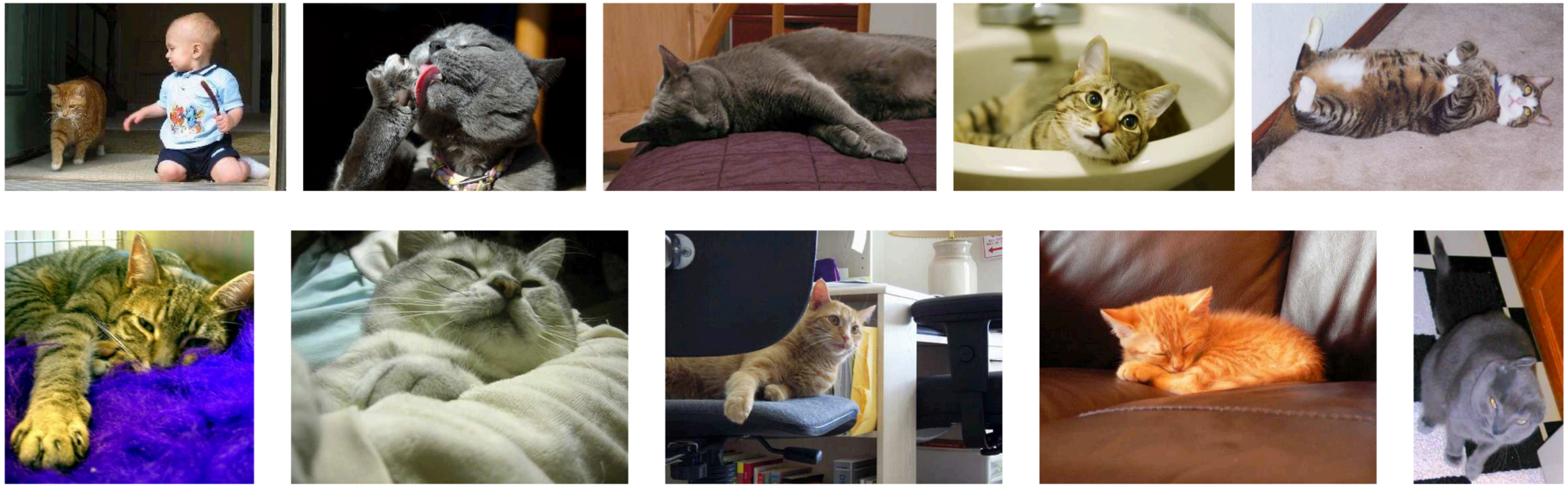
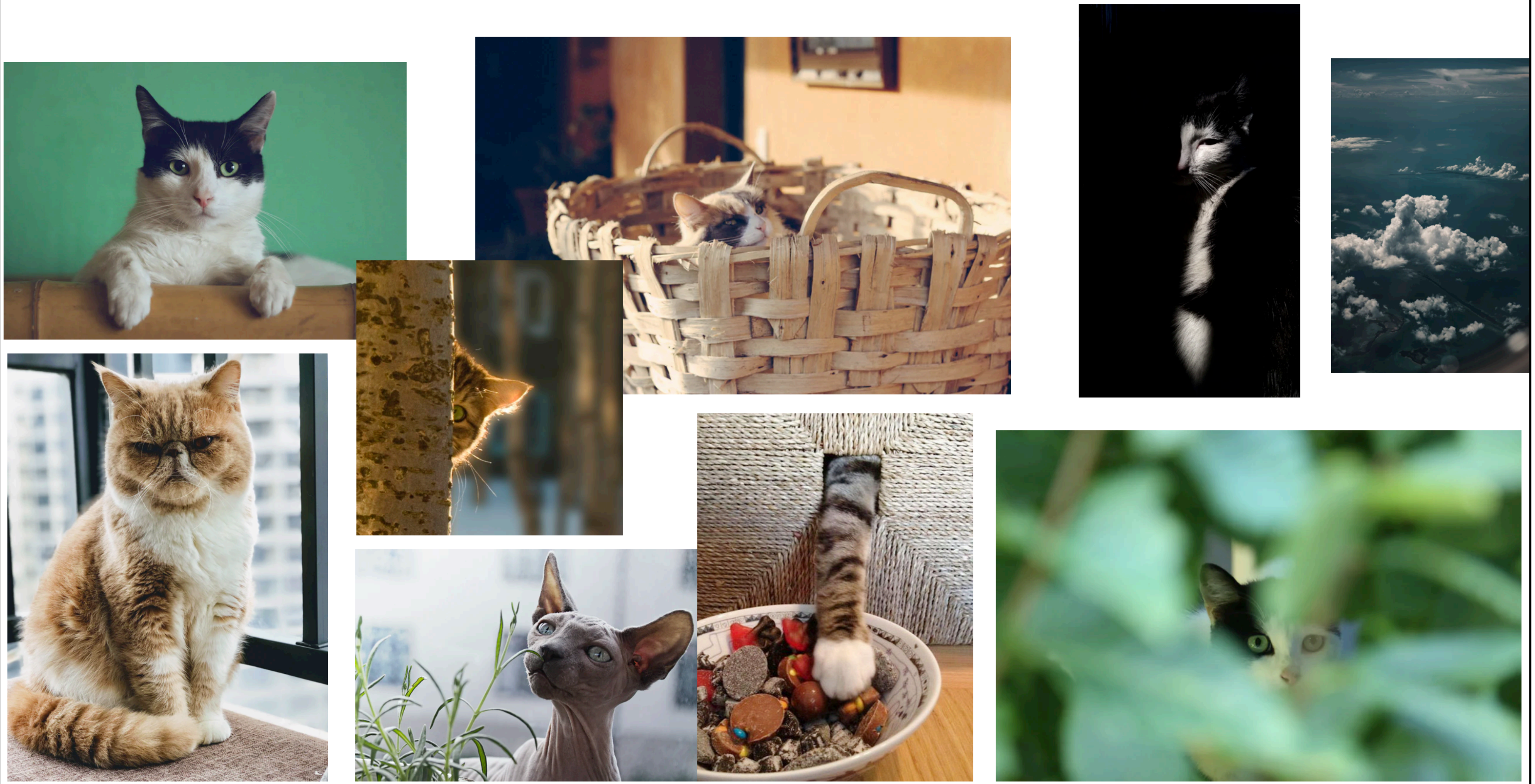
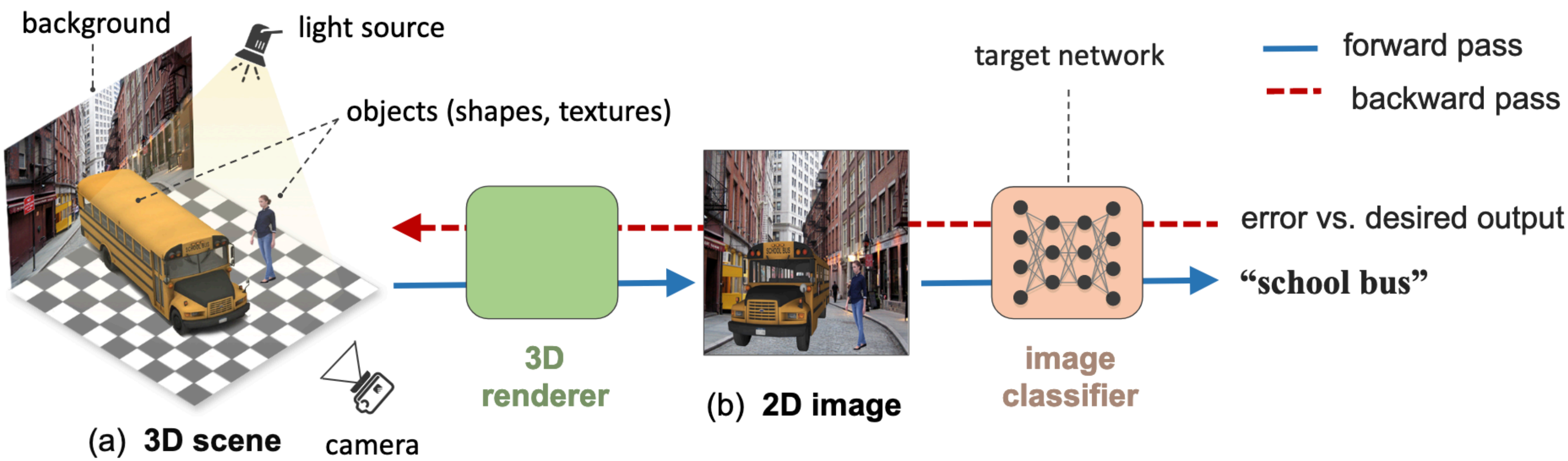


Figure 1. **The deformable and truncated cat.** Cats exhibit (almost) unconstrained variations in shape and layout.

Parkhi et al. *The truth about cats and dogs*. 2011

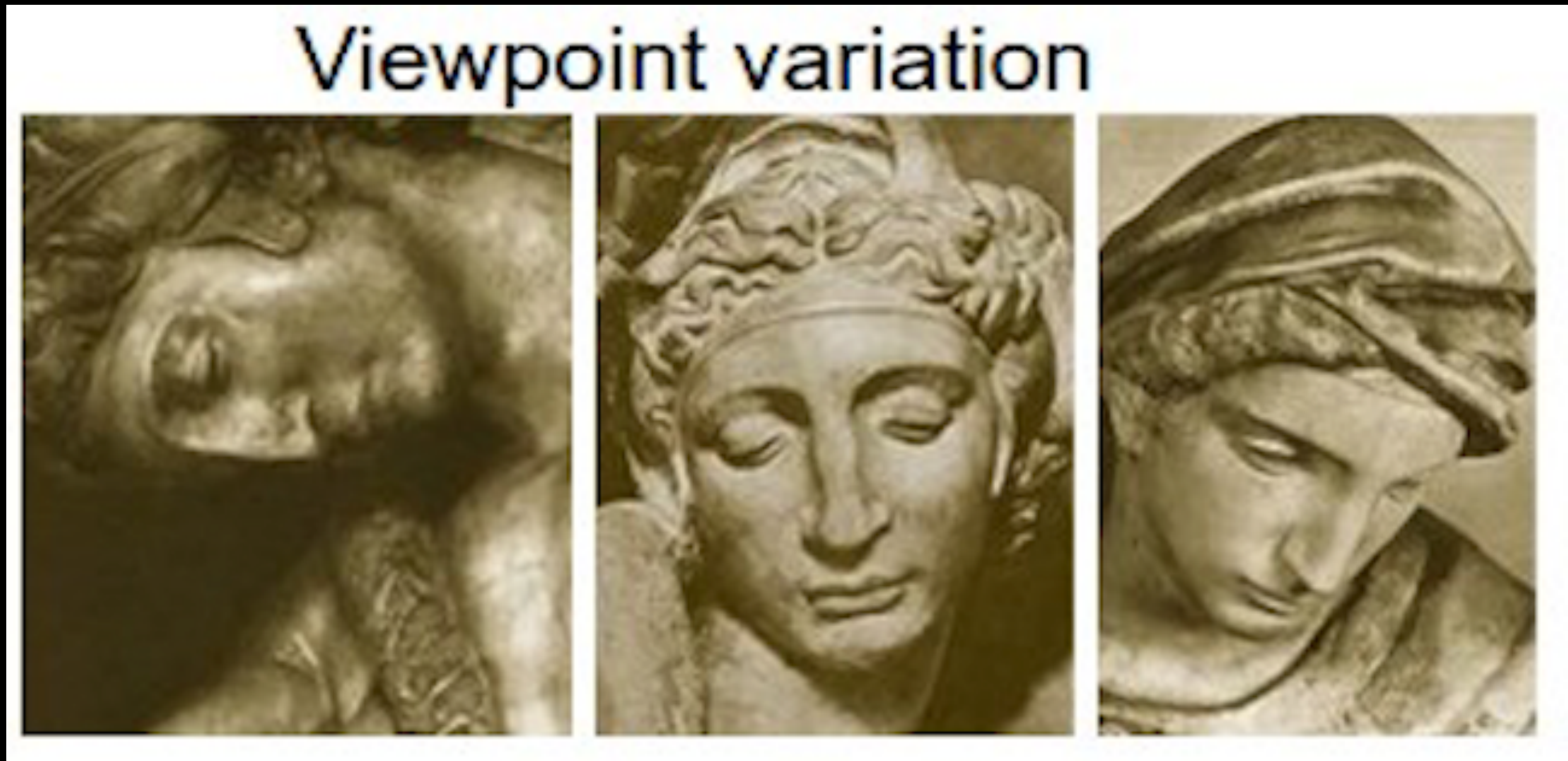



background
light source
objects (shapes, textures)
target network
forward pass
backward pass
error vs. desired output
"school bus"
3D renderer
(b) 2D image
image classifier
(a) 3D scene
camera

Computer Vision Challenges

Viewpoint Variation

A single instance of an object can be oriented in many ways to the camera.



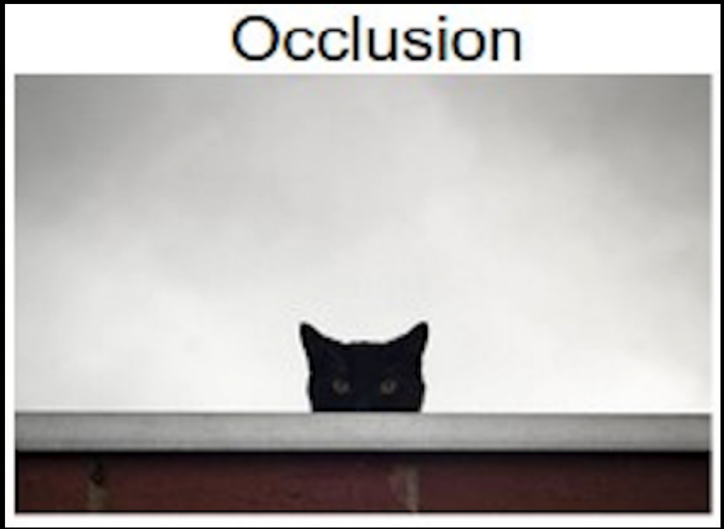
Scale variation

- Visual classes often exhibit variation in their size
  - Size in the real world
  - Size in the image

Deformation

Many objects of interest are not rigid bodies and can be deformed in extreme ways.

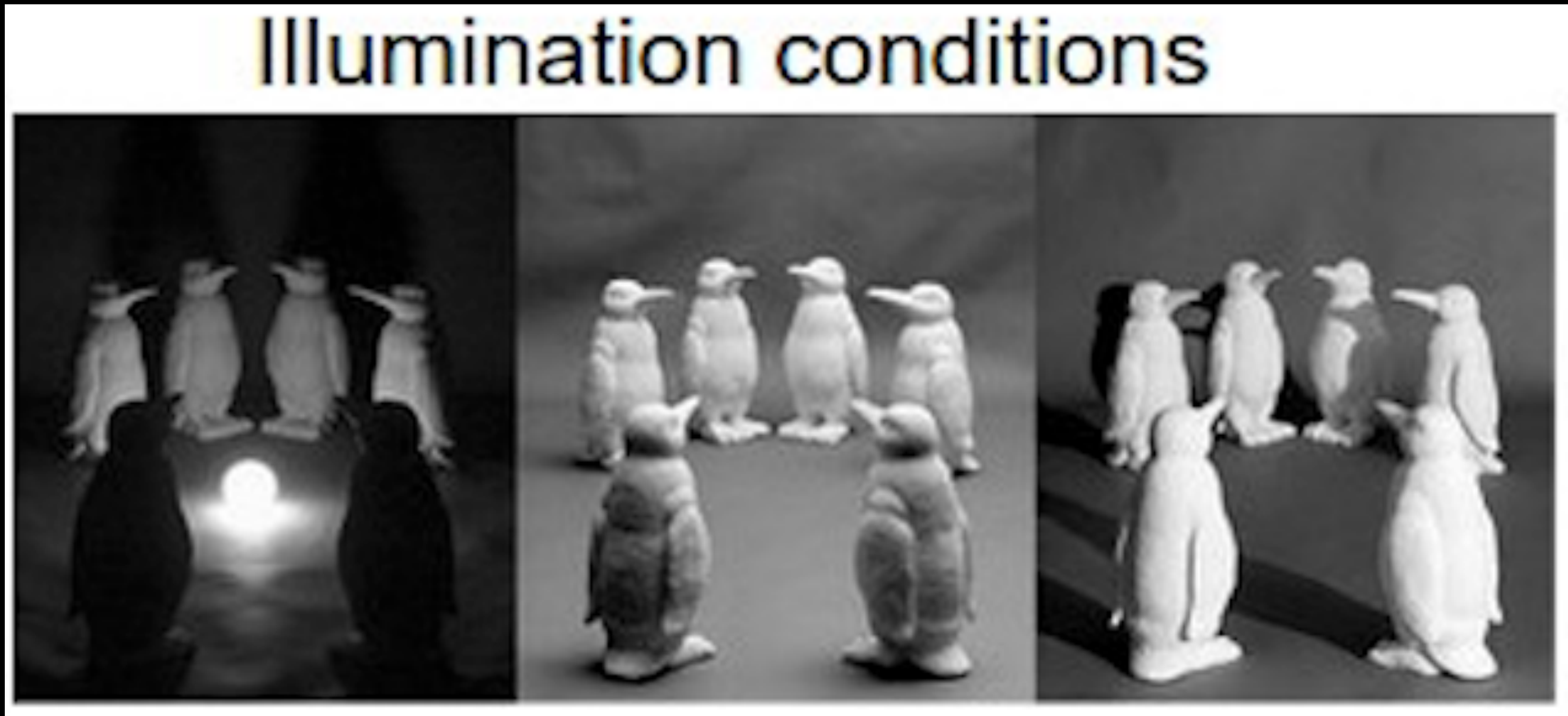
Occlusion



The objects of interest can be occluded. Sometimes only a small portion of an object (as little as few pixels) could be visible

Illumination Condition

The effects of illumination are drastic on the pixel level.

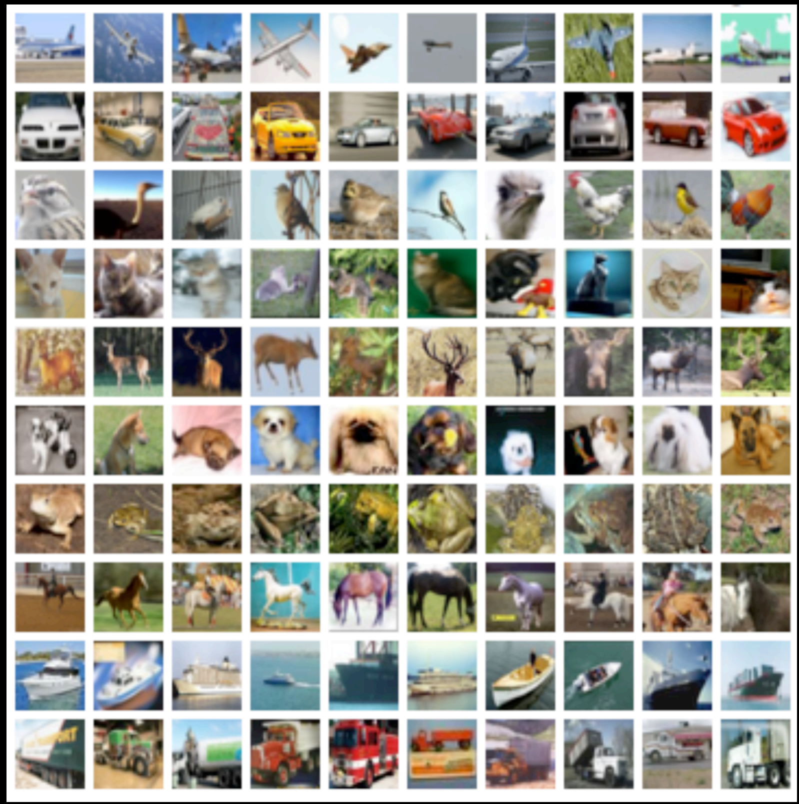


Background clutter

The objects of interest may blend into their environment, making them hard to identify.

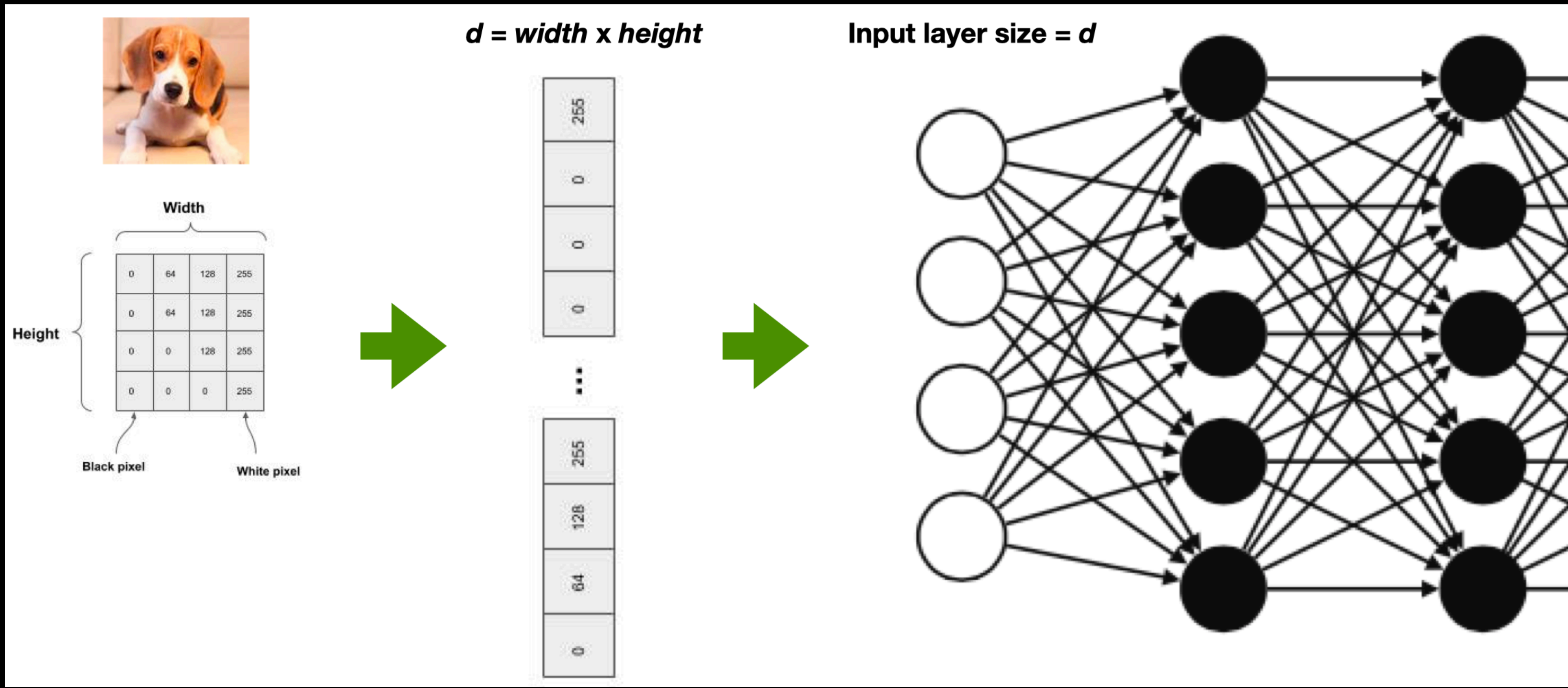
Intra-class variation

- The classes of interest can often be relatively broad, such as chairs.
- There are many different types of these objects, each with their appearance.



How CV models work?

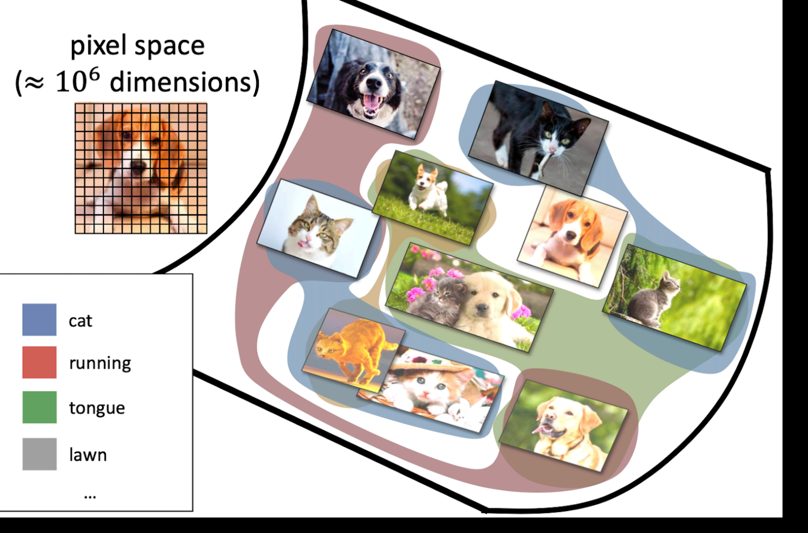
Flattening



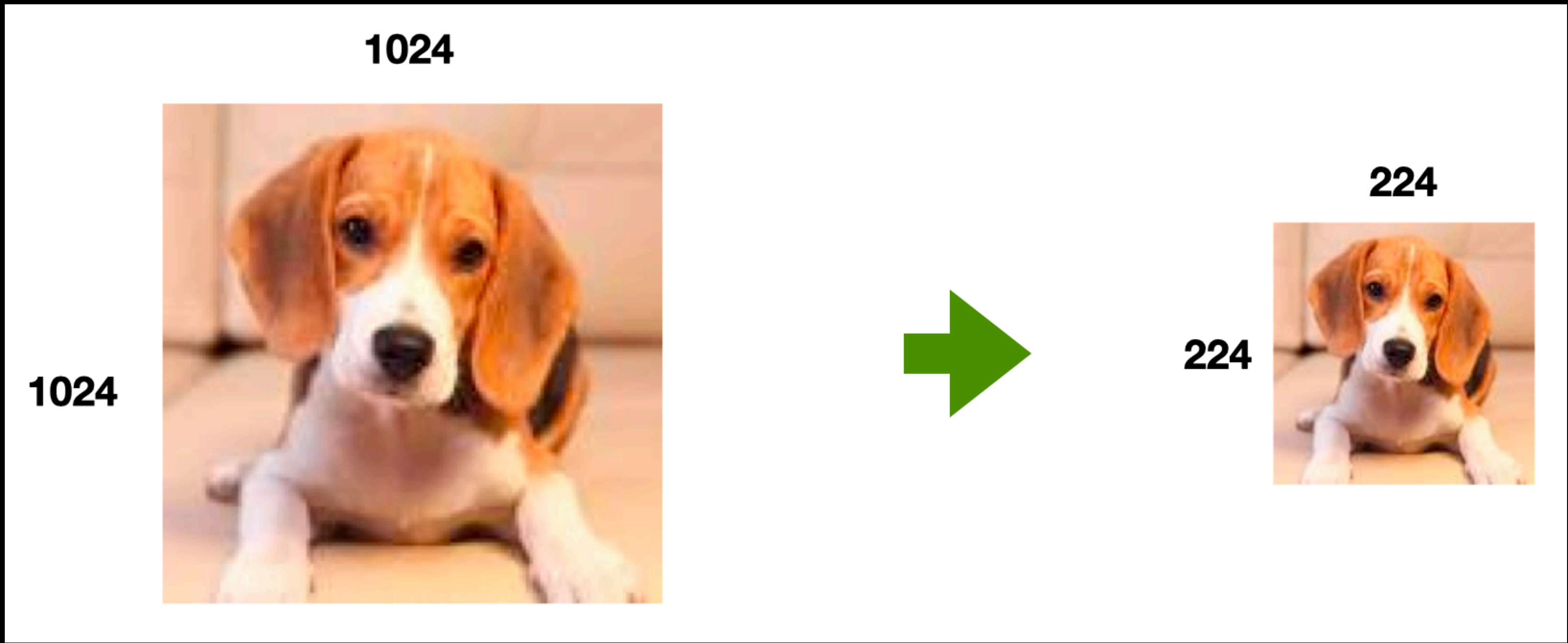
d = width x height
Input layer size = d
Width
Height
0
64
128
255
0
64
128
255
0
0
128
255
0
0
0
255
Black pixel
White pixel
255
0
0
0
...
255
128
64
0

Course of dimensionality

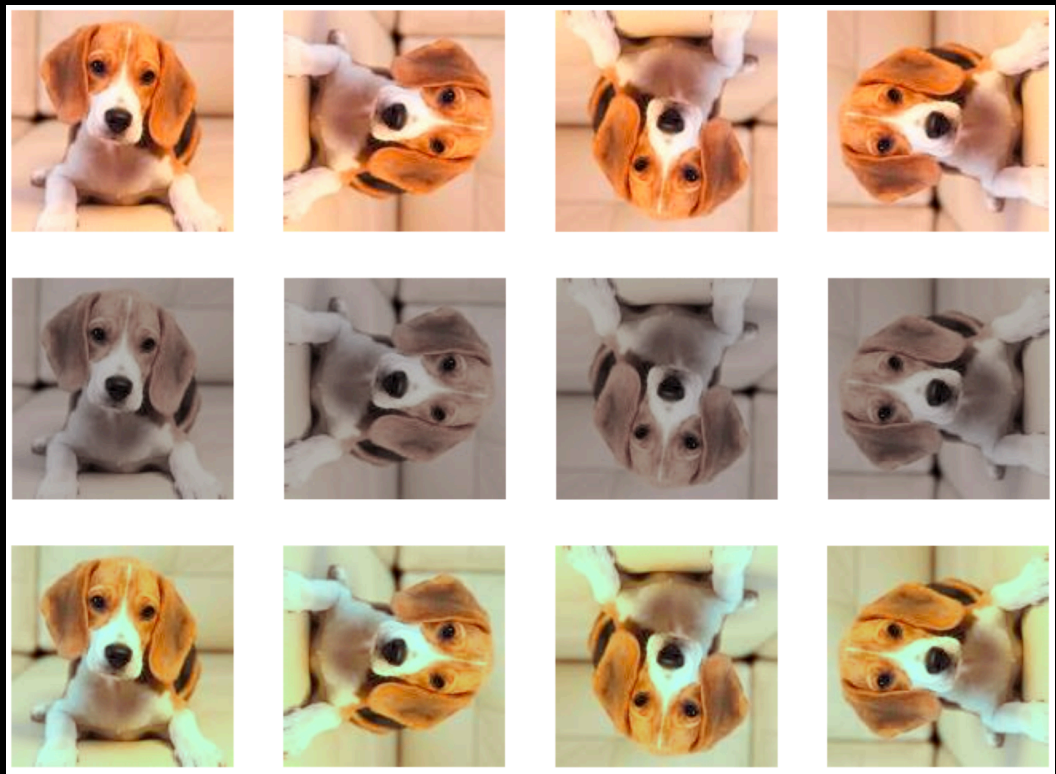
- High dimensionality
  - A 1024×768 image has d = 786432!
  - A tiny 32×32 image has d = 1024
- Decision boundaries in pixel space are extremely complex
- We will need “big” ML models with lots of parameters
  - For example, linear regressors need d parameters



Downsampling



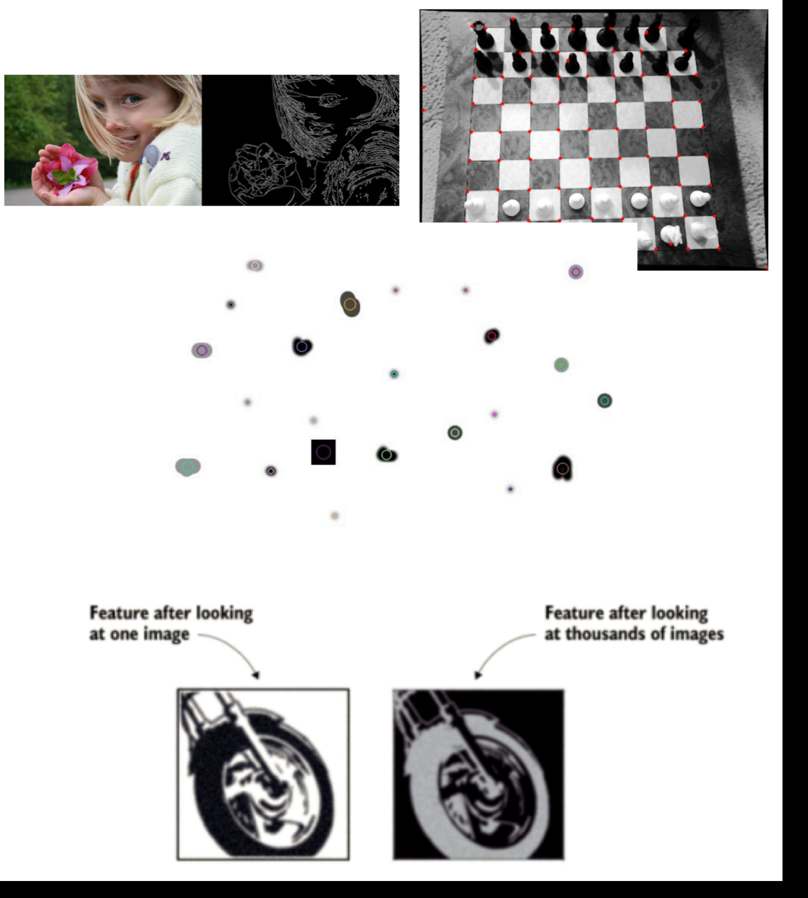
What about generalisation?



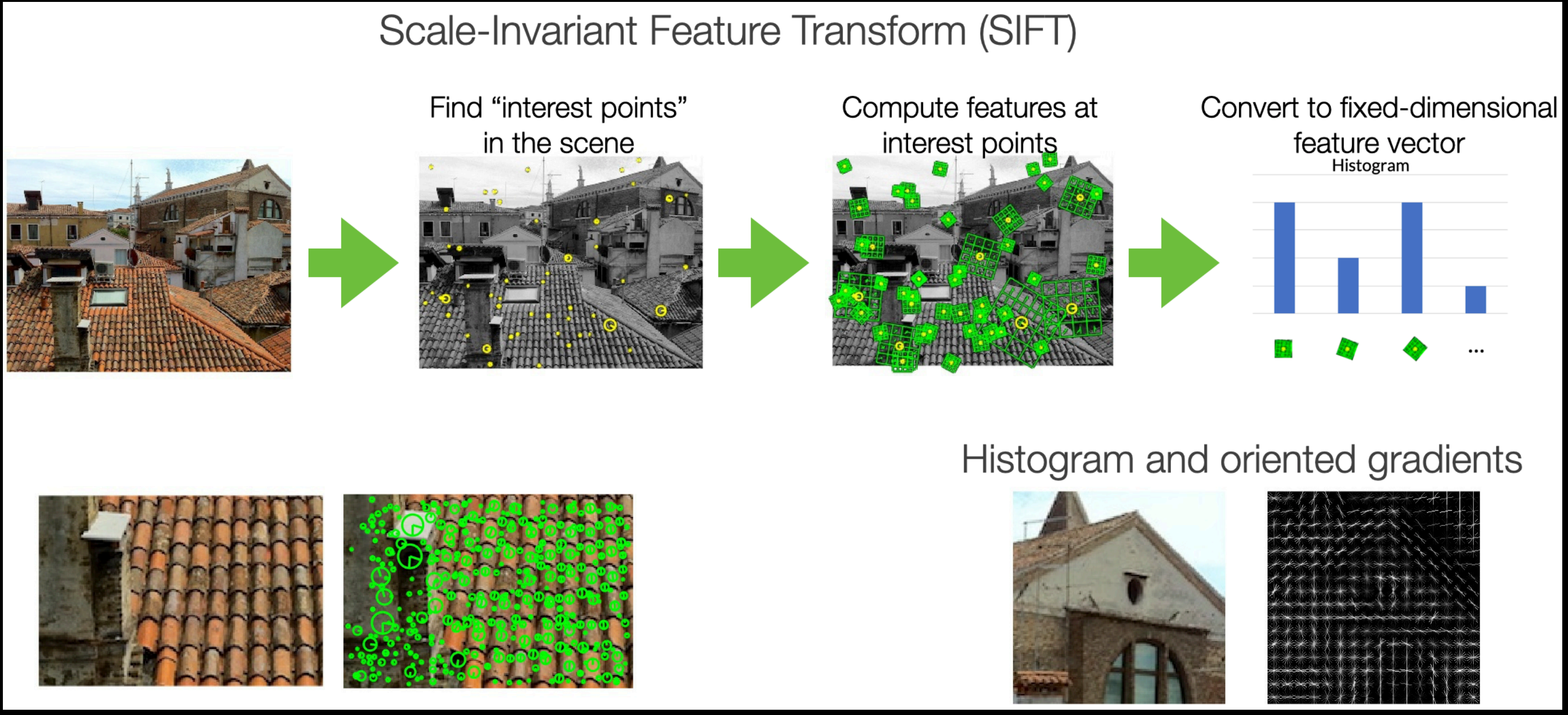
The "old days": Feature Extraction

- **Feature**
  - A relevant piece of information about the content of an image

    -e.g. edges, corners, blobs (regions), ridges
- **A good feature**
  - Repeatable
  - Identifiable
  - Can be easily tracked and compared
  - Consistent across different scales, lighting conditions, and viewing angles
  - Visible in noisy images or when only part of an object is visible
  - Can distinguish objects from one another

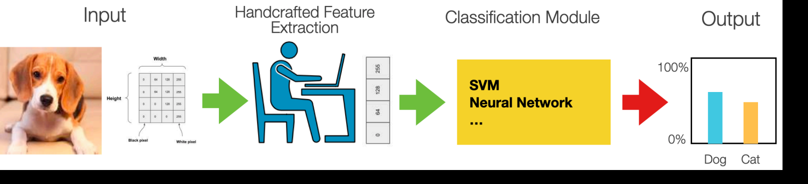


Feature Extraction Techniques



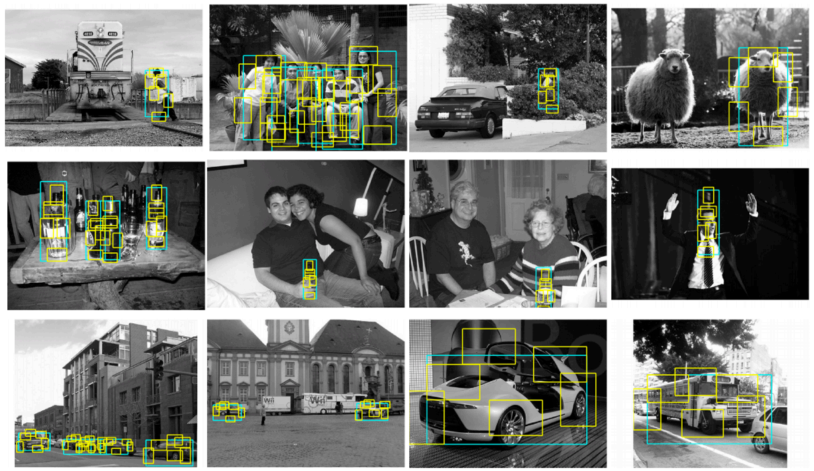
The "old days": Feature Engineering

- Machine learning models are only as good as the features you provide
- To figure out which features you should use for a specific problem
- Rely on domain knowledge (or partner with domain experts)
- Experiment to create features that make machine learning algorithms work better



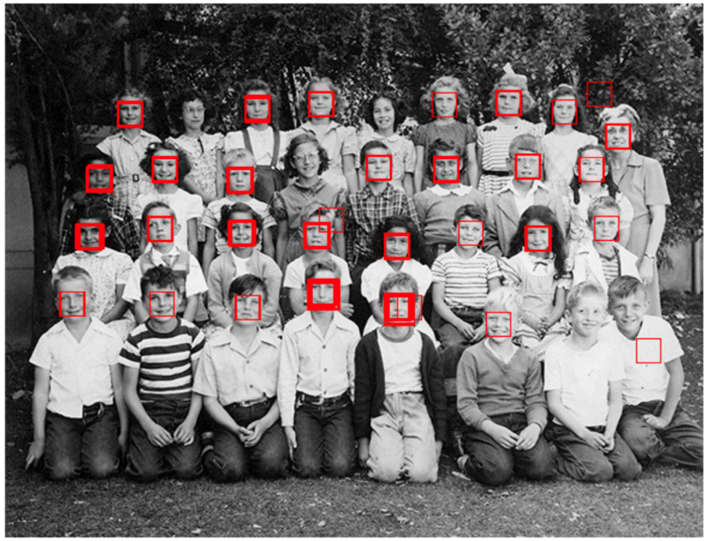
Performance

Object Detection (~2007)



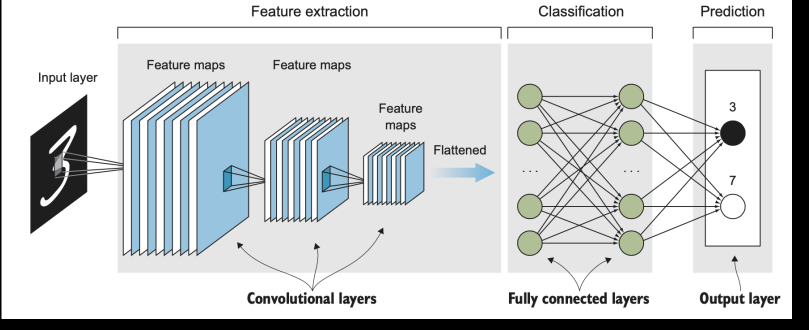
Felzenszwalb, Ramanan, McAllester. A Discriminatively Trained, Multiscale, Deformable Part Model. CVPR 2008 (DPM v1)

Face Detection (~2013)



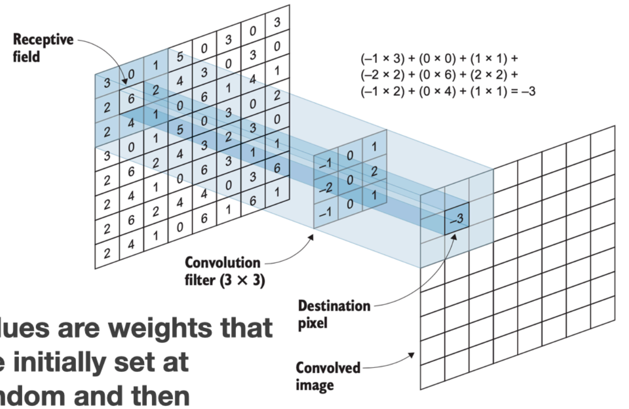
https://github.com/alexdemartos/ViolaAndJones

Convolutional Neural Networks

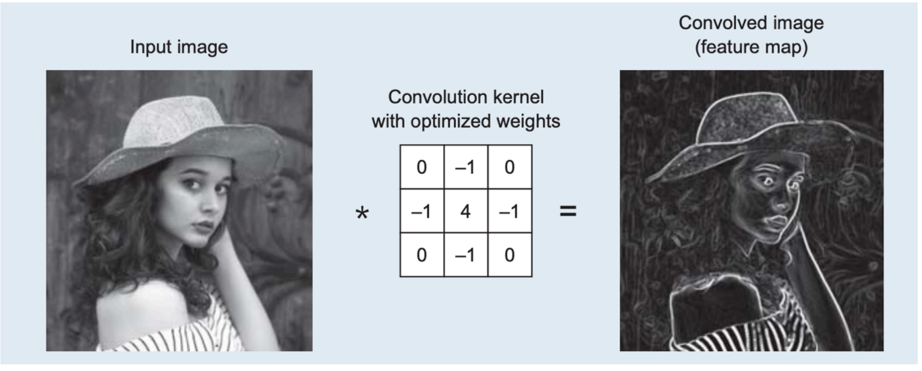


- CNNs exploit image properties to reduce the number of model parameters drastically
- Feature maps
  - Automatically extracted hierarchical
  - Retain spatial association between pixels
- Translation invariance
  - a dog is a dog even if its image is shifted by a few pixels
- Local interactions
  - all processing happens within very small image windows
  - within each layer, far-away pixels cannot influence nearby pixels

Convolution & Feature Maps

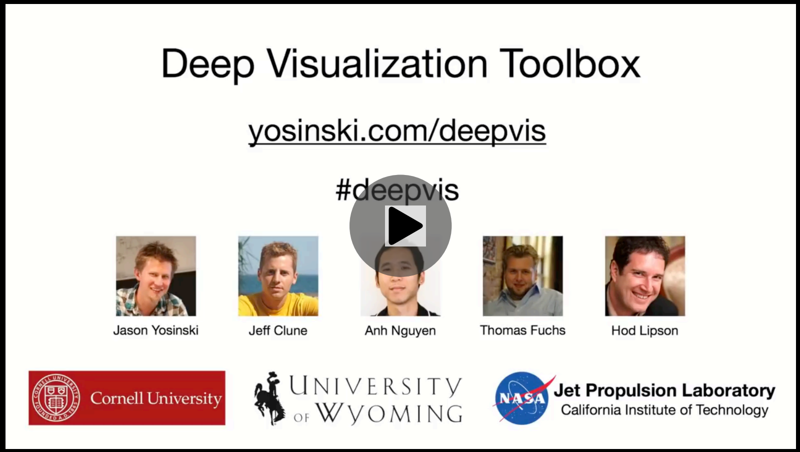


Values are weights that are initially set at random and then learned

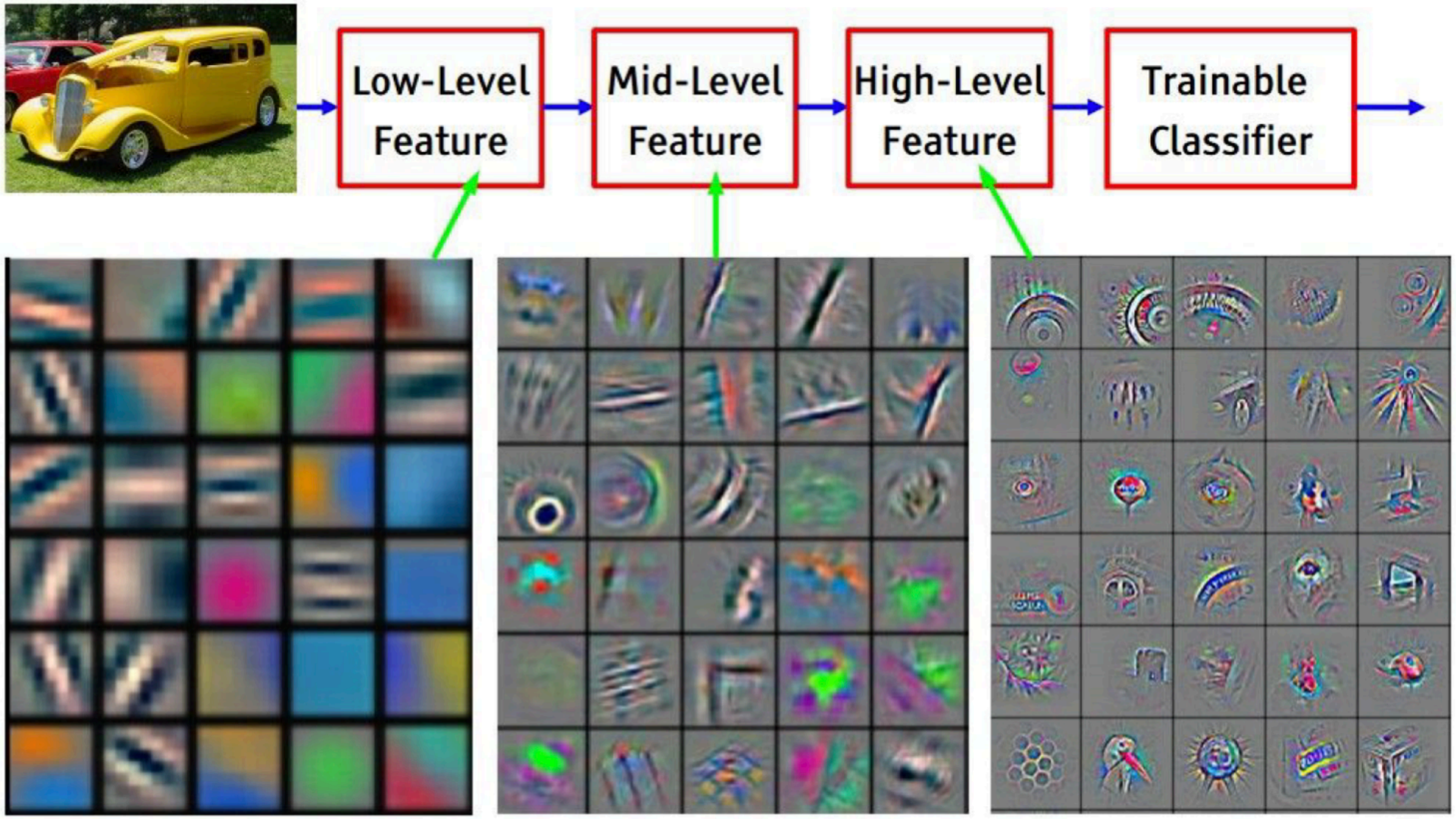


What CNNs learn?

[Deep Visualization Toolbox]

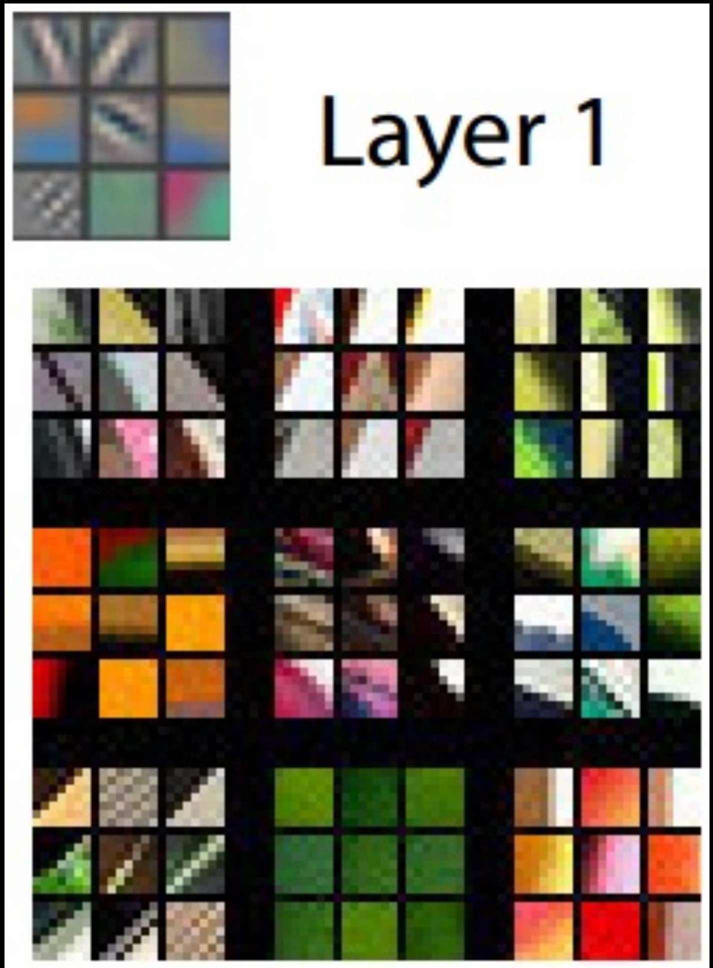


Feature Visualisation

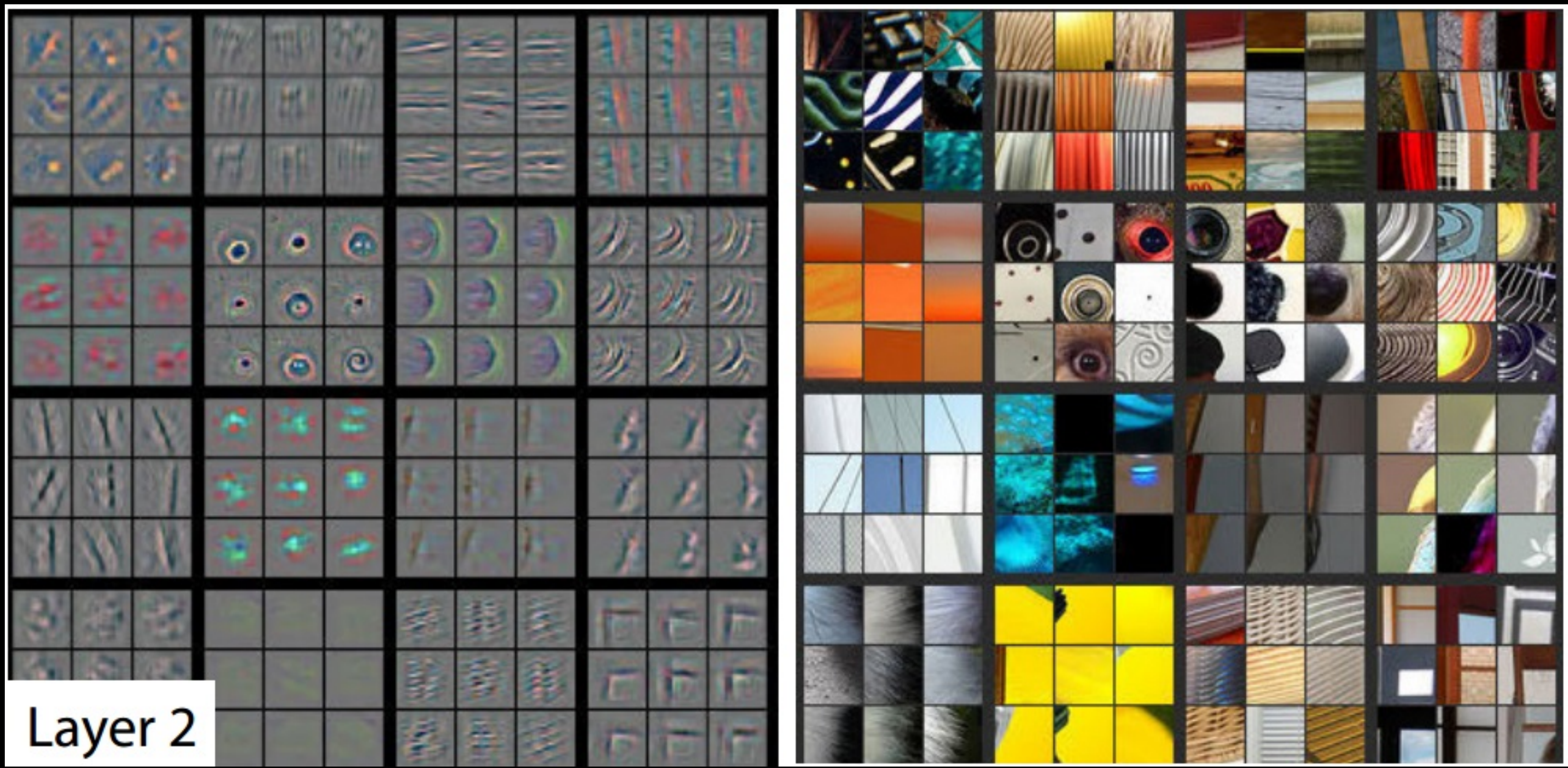


Feature visualization of convolutional net trained on ImageNet from [Zeiler & Fergus 2013]

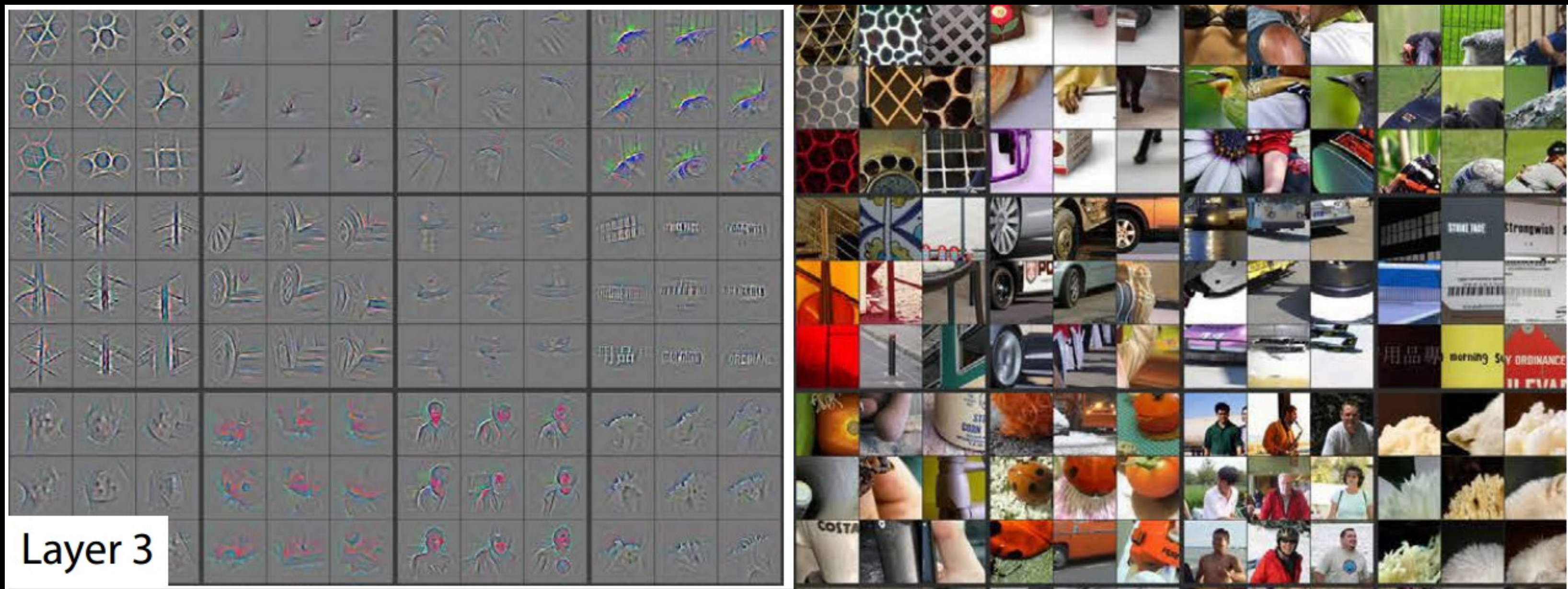
Layer 1



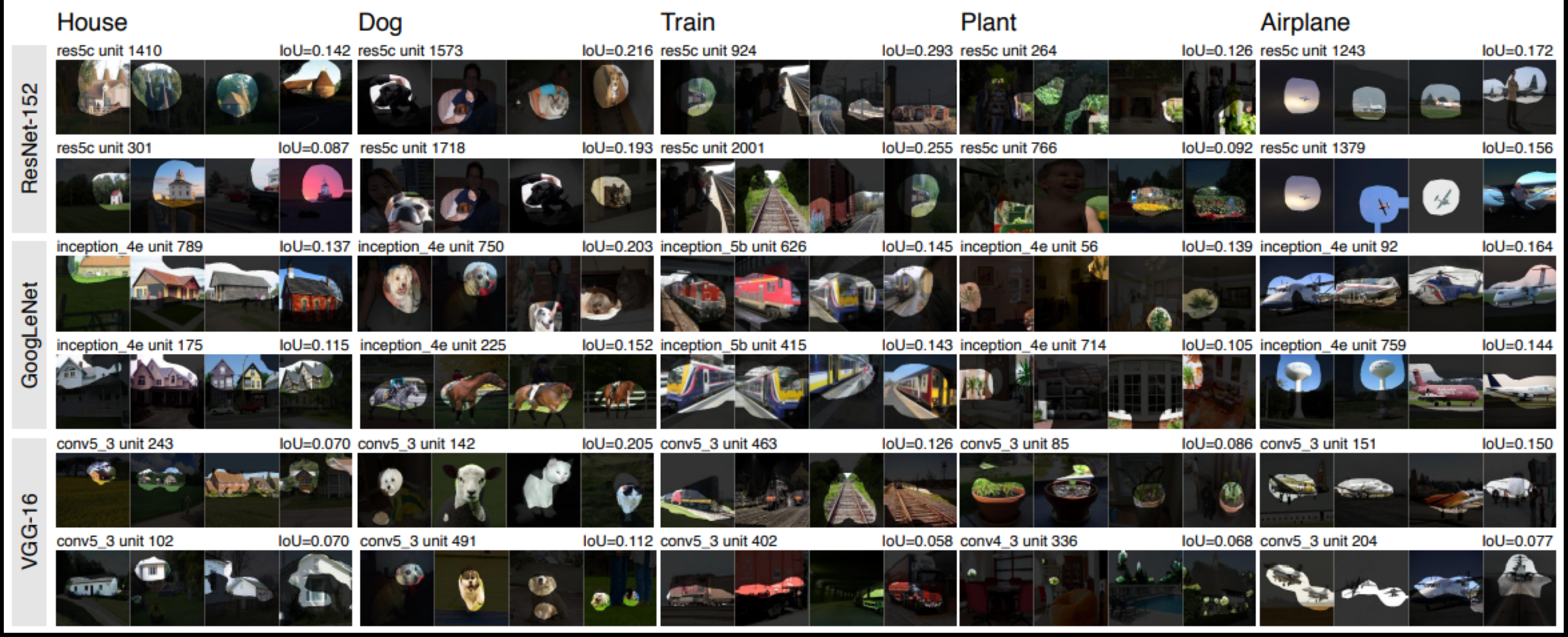
Layer 2



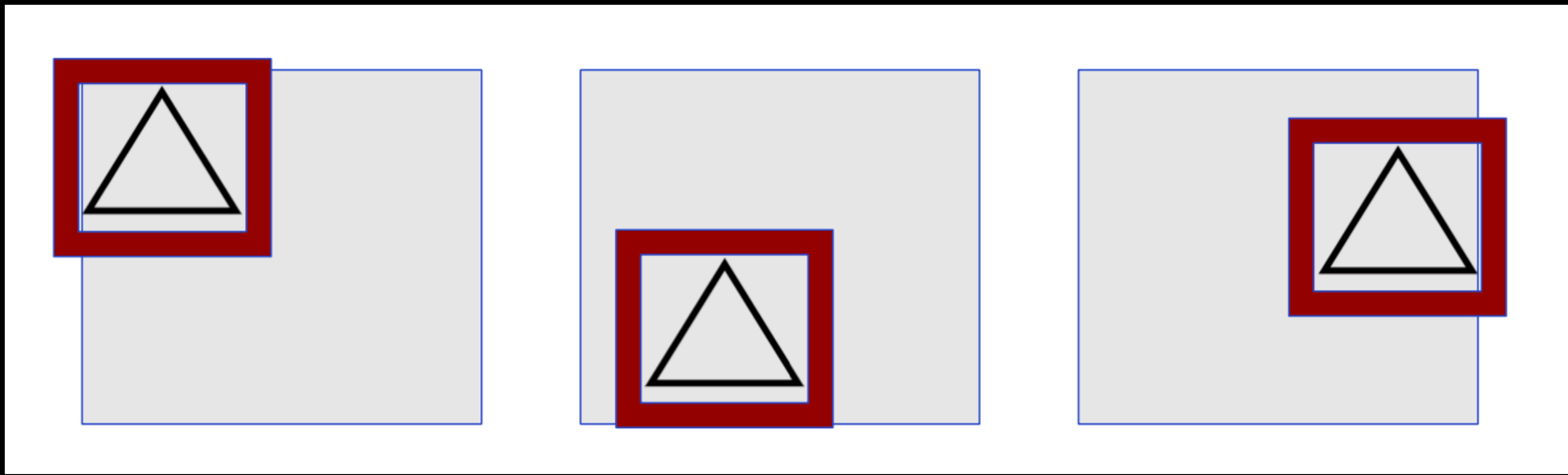
Layer 3



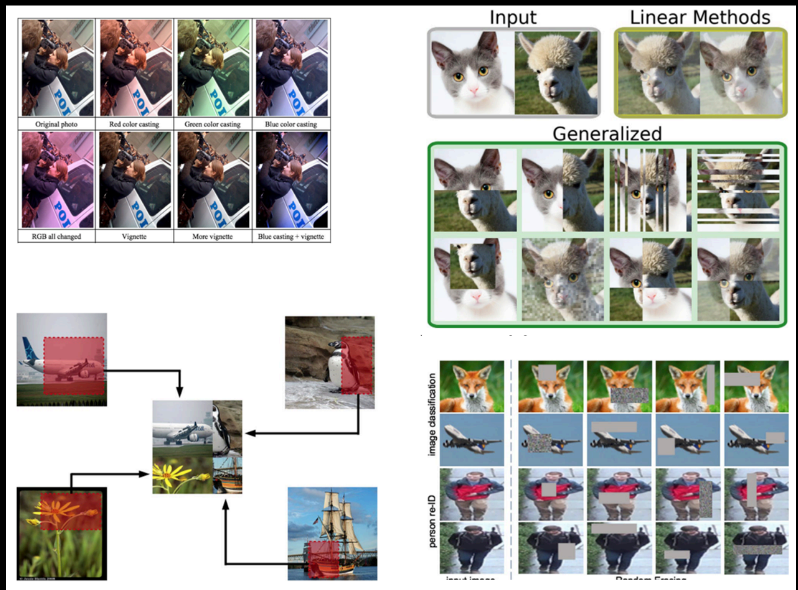
Network Dissection



Translation Invariance

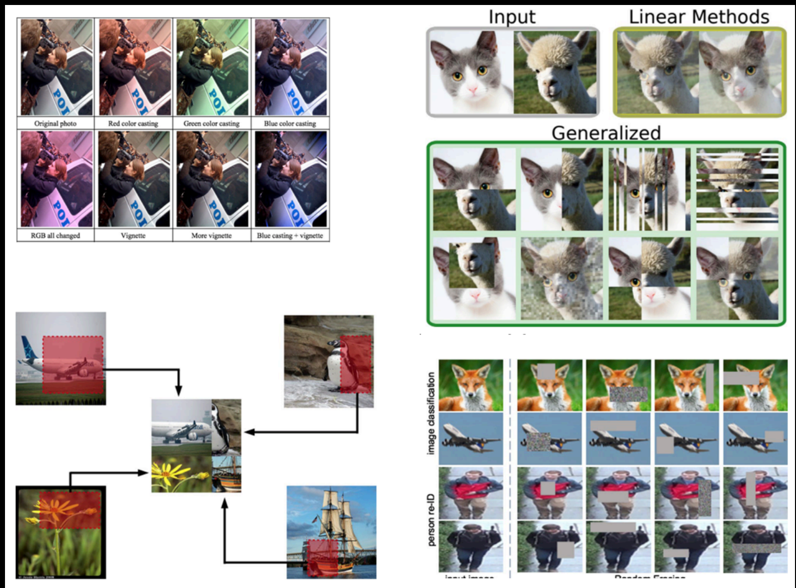


Data Augmentation



- Generate variations of the input data
  - To improve generalisability (out of distribution inputs)
  - Improve invariance (rotation, scaling, distortion)

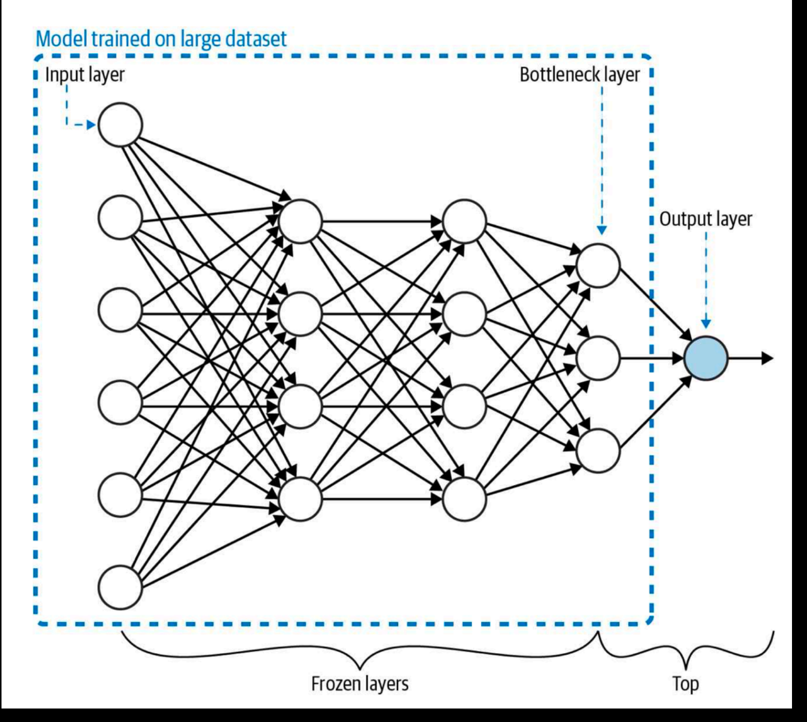
Data Augmentation



- Geometric
  - Flipping, Cropping, Rotation, Translation,
- Noise Injection
- Color space transformation
- Mixing Images
- Random erasing
- Adversarial training
- GAN-based image generation

Robustness to input variation

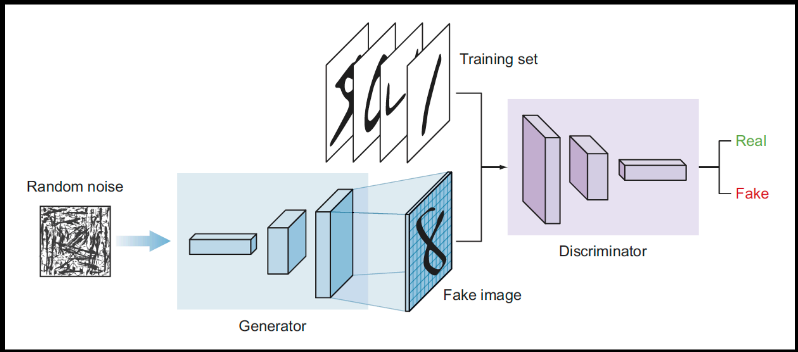
Transfer Learning



- **Problem**: training custom ML models requires extremely large datasets
- **Transfer learning** : take a model that has been trained on the same type of data for a similar task and apply it to a specialised task using our own custom data.
  - **Same data**: same data modality. same types of images (e.g. professional pictures vs. Social media pictures)
  - **Similar tasks** : if you need a new object classification model, use a model pre-trained for object classification

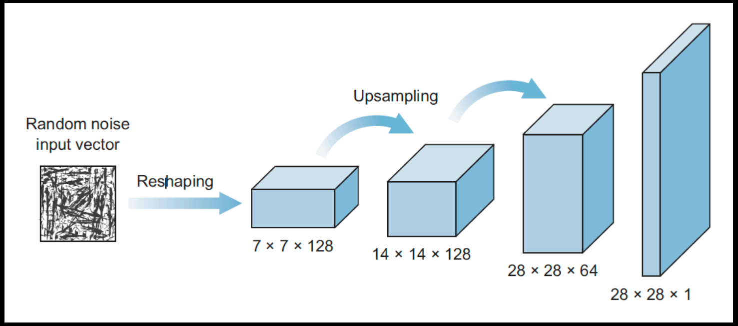
Advanced Computer Vision Techniques

Generative Adversarial Networks



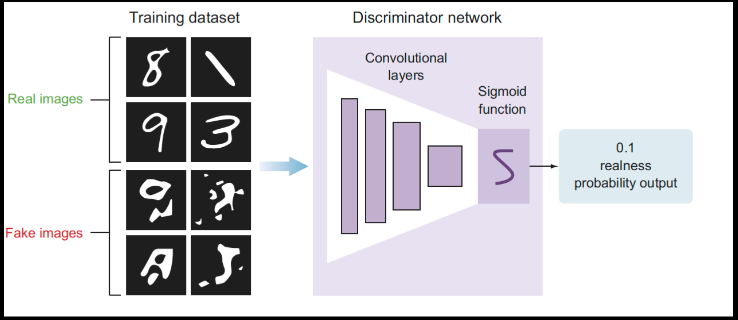
- Learn patterns from the training dataset and create new images that have a similar distribution of the training set
- Two deep neural networks that compete with each other
  - The **generator** tries to convert random noise into observations that look as if they have been sampled from the original dataset
  - The **discriminator** tries to predict whether an observation comes from the original dataset or is one of the generator's forgeries

- The **generator**'s architecture looks like an inverted CNN that starts with a narrow input and is upsampled a few times until it reaches the desired size



- The **discriminator**

's model is a typical classification neural network that aims to classify images generated by the generator as real or fake



Which face is real?

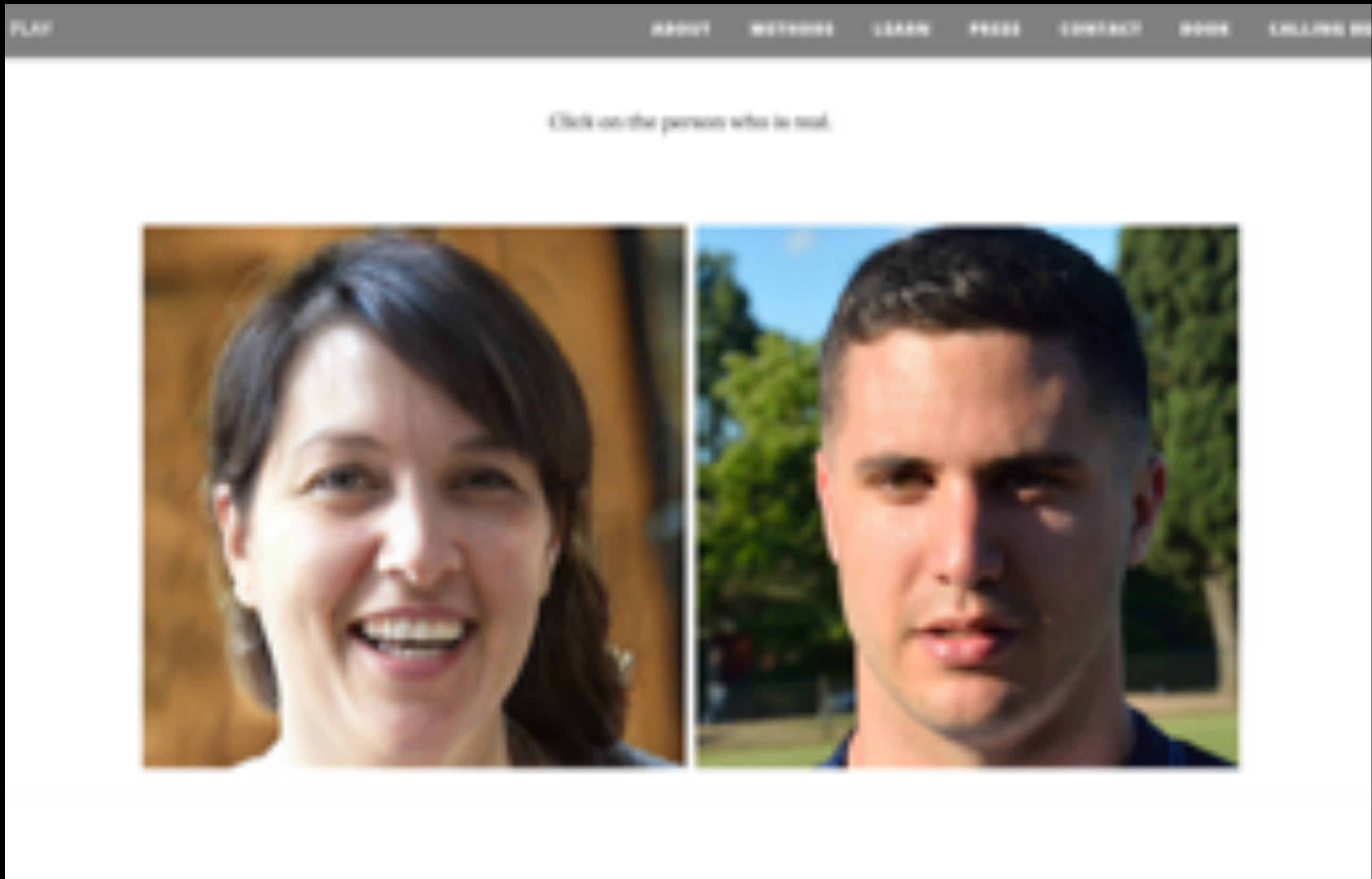
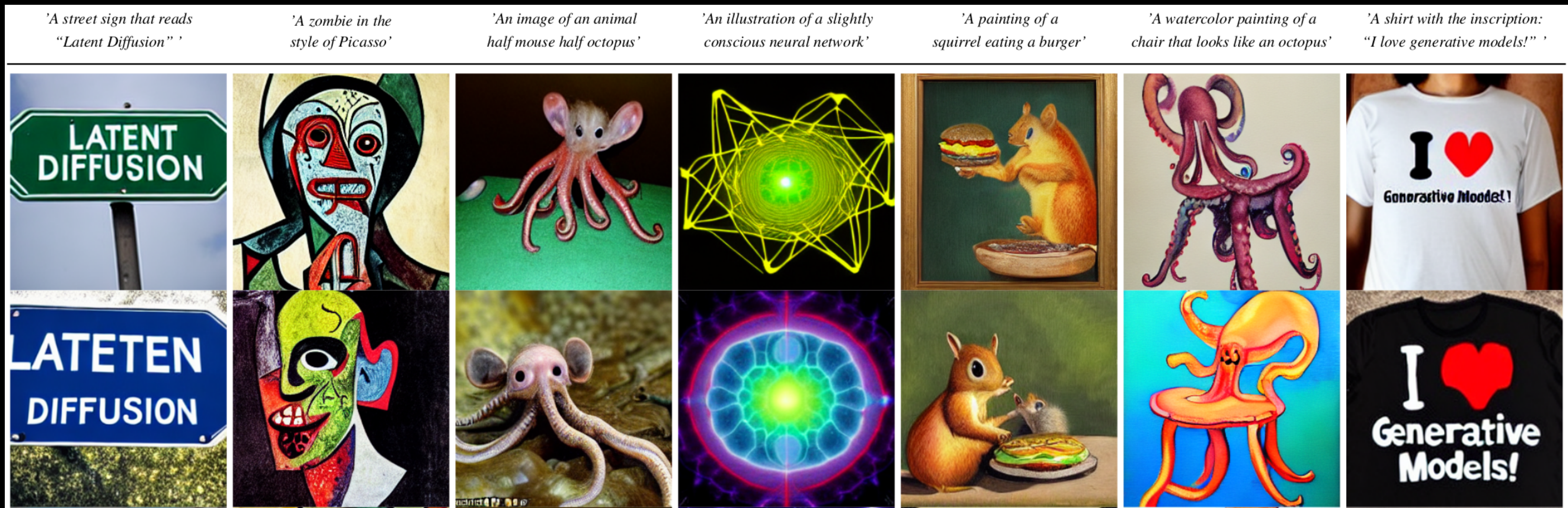


Image super-resolution GAN

– A good technical summary

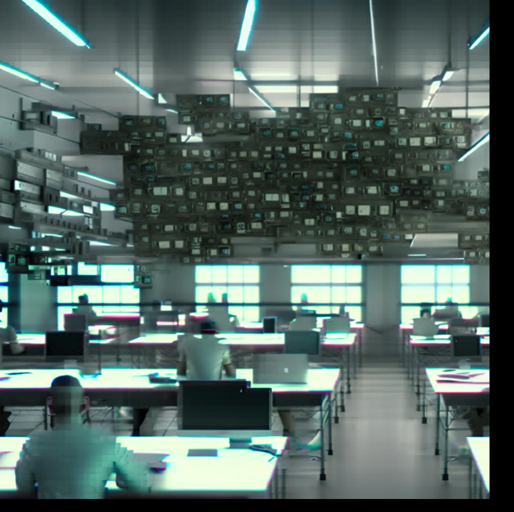
Text-To-Image Generation



Prompt: "*A dream of a classroom full of interested XXX students*"

Design

CS



- ML-generated painting sold for $432,500
- The network trained on a dataset of 15,000 portraits painted between the fourteenth and twentieth centuries
- Network “learned” the style, and generated a new painting

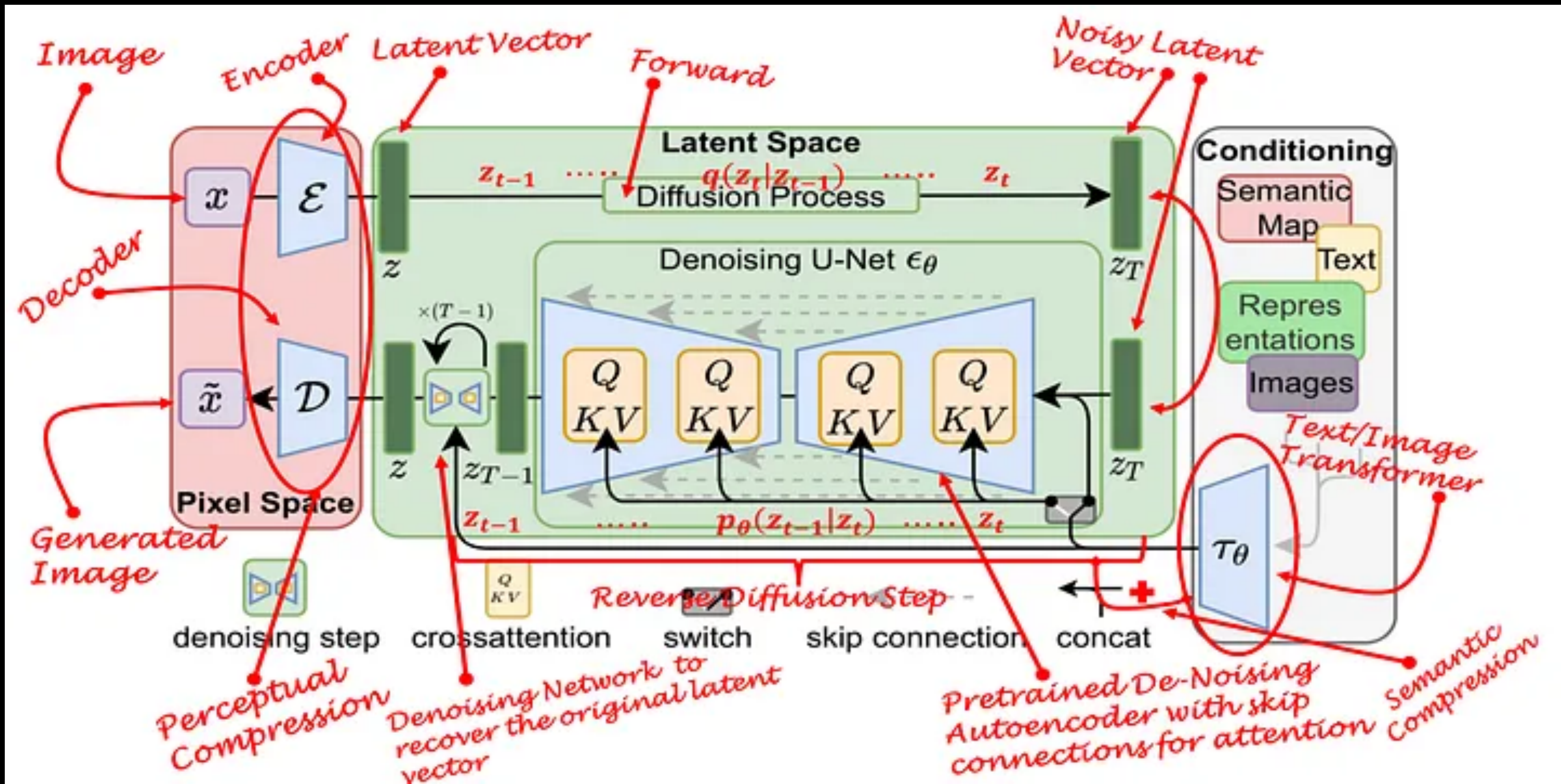
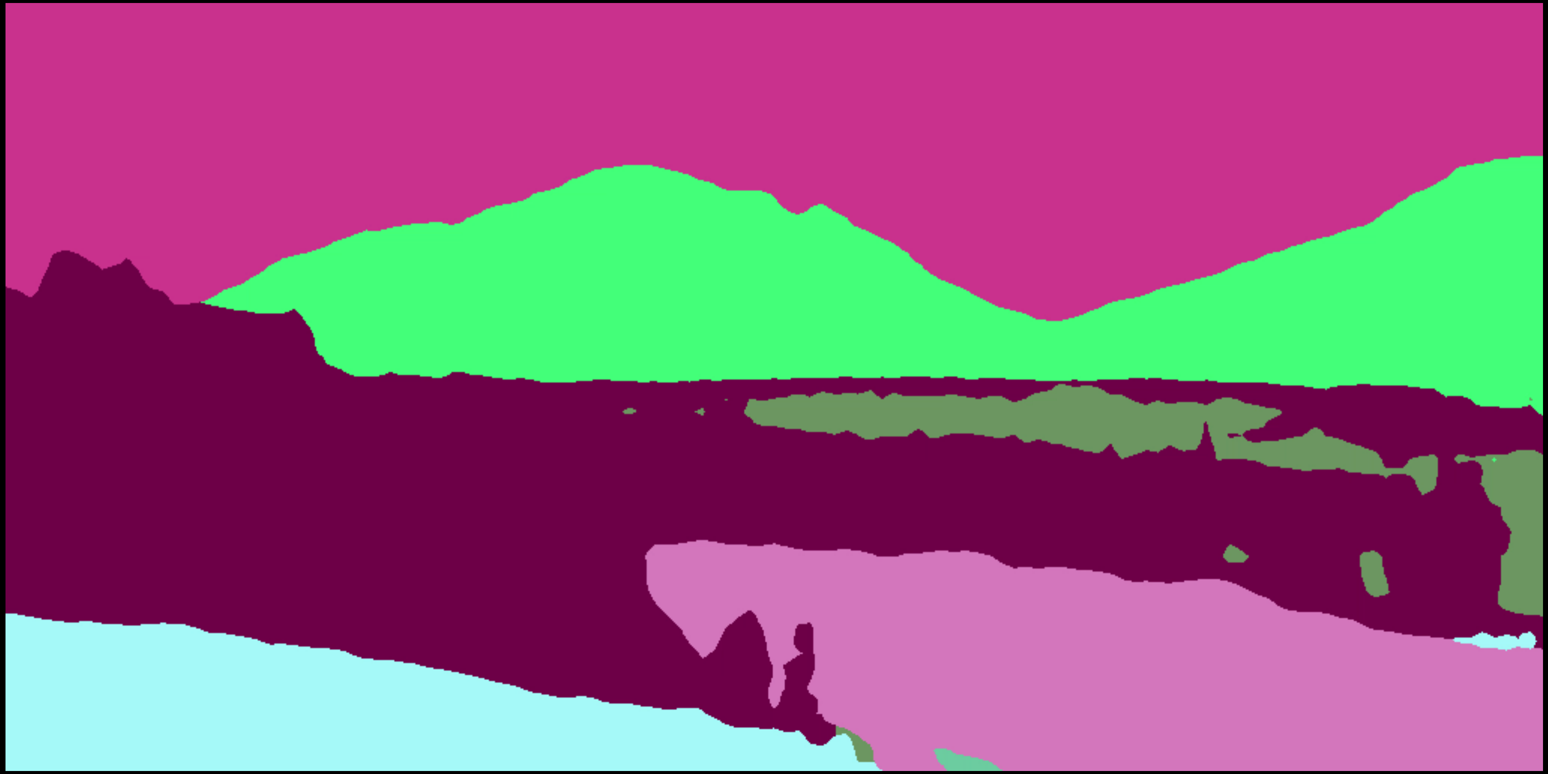
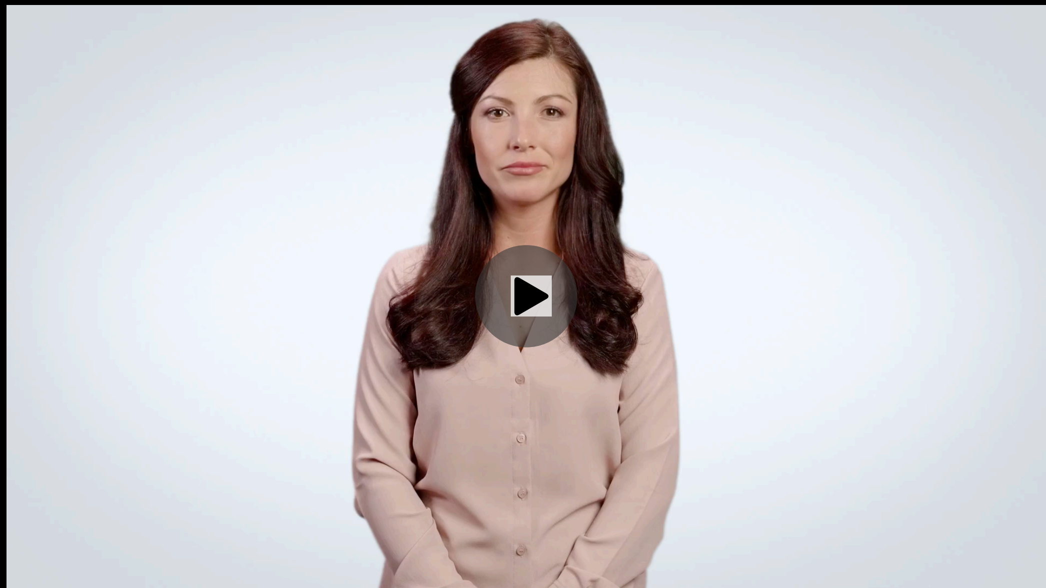


Image
Encoder
Latent Vector
Forward
Noisy Latent Vector
Latent Space
Diffusion Process
Conditioning
Semantic Map
Text
Repres entations
Images
Decoder
Denoising U-Net ϵ_θ
$\times(T-1)$
Pixel Space
Text/Image Transformer
Generated Image
Reverse Diffusion Step
denoising step
crossattention
switch
skip connection
concat
Perceptual Compression
Denoising Network to recover the original latent vector
Pretrained De-Noising Autoencoder with skip connections for attention
Semantic Compression

Image-to-Image Generation

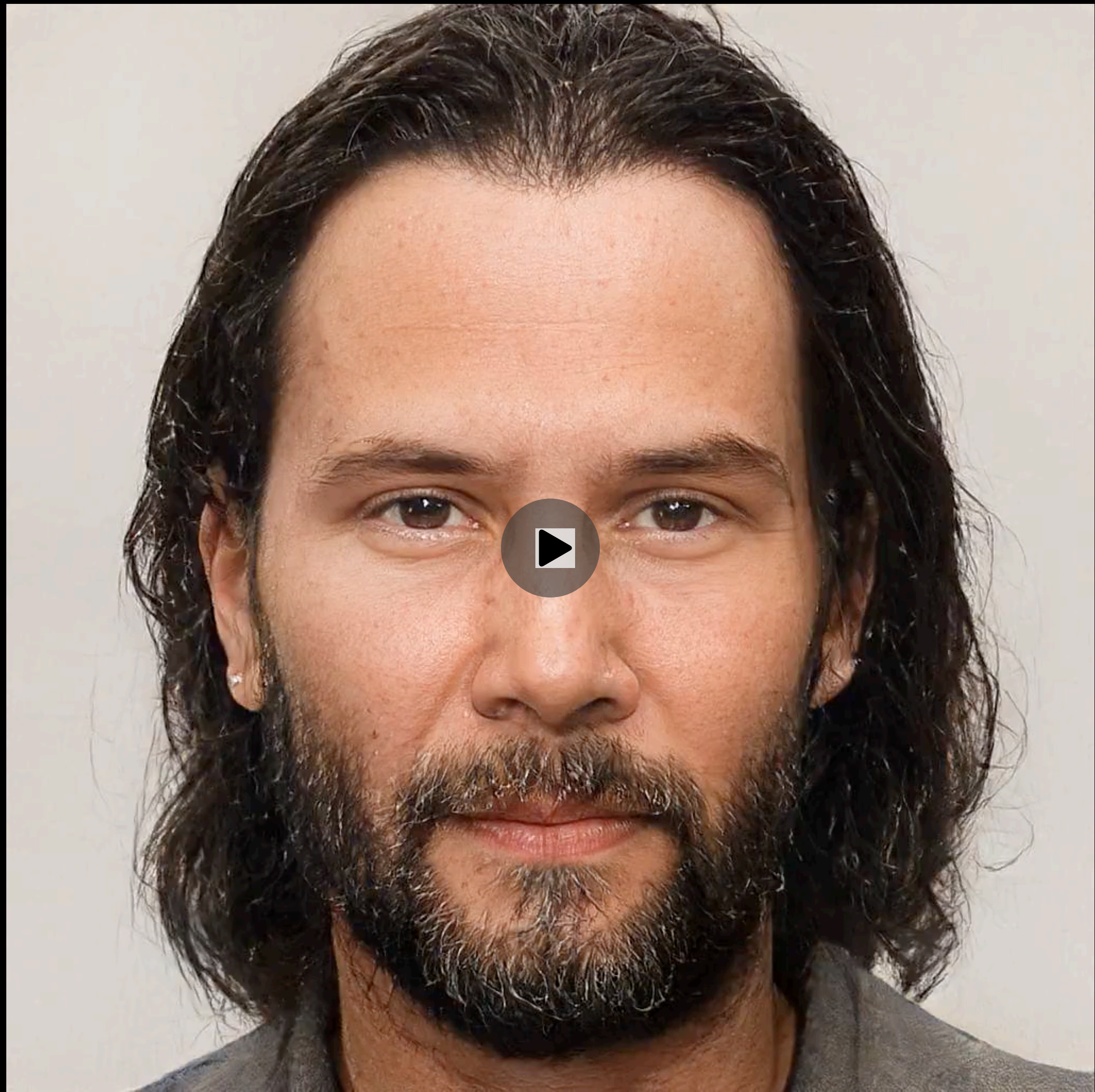


Synthetic Video Generation



Generated from Synthesia.io

Neural Style Transfer



Deep Fakes

[Very realistic Tom Cruise Deepfake]

Machine Learning for Design

Lecture 4

Machine Learning for Images. *Part 2*

Credits

CMU Computer Vision course - Matthew O'Toole.

Grokking Machine Learning. Luis G. Serrano. Manning, 2021

[CIS 419/519 Applied Machine Learning]. Eric Eaton, Dinesh Jayaraman.

Deep Learning Patterns and Practices - Andrew Ferlitsch, Maanning, 2021

Machine Learning Design Patterns - Lakshmanan, Robinson, Munn, 2020

Deep Learning for Vision Systems. Mohamed Elgendy. Manning, 2020