A Taxonomy of Computer Vision Methods

(How is your problem called)

A. Nicolaou

Introduction

- Computer Vision has had many techniques
- Now it's mostly deep learning

Knowledge transfer

- Labeled data rare!
- Tuning training very sensitive
- The more common a problem is, the easier to find reusable work

Supervised learning workflow

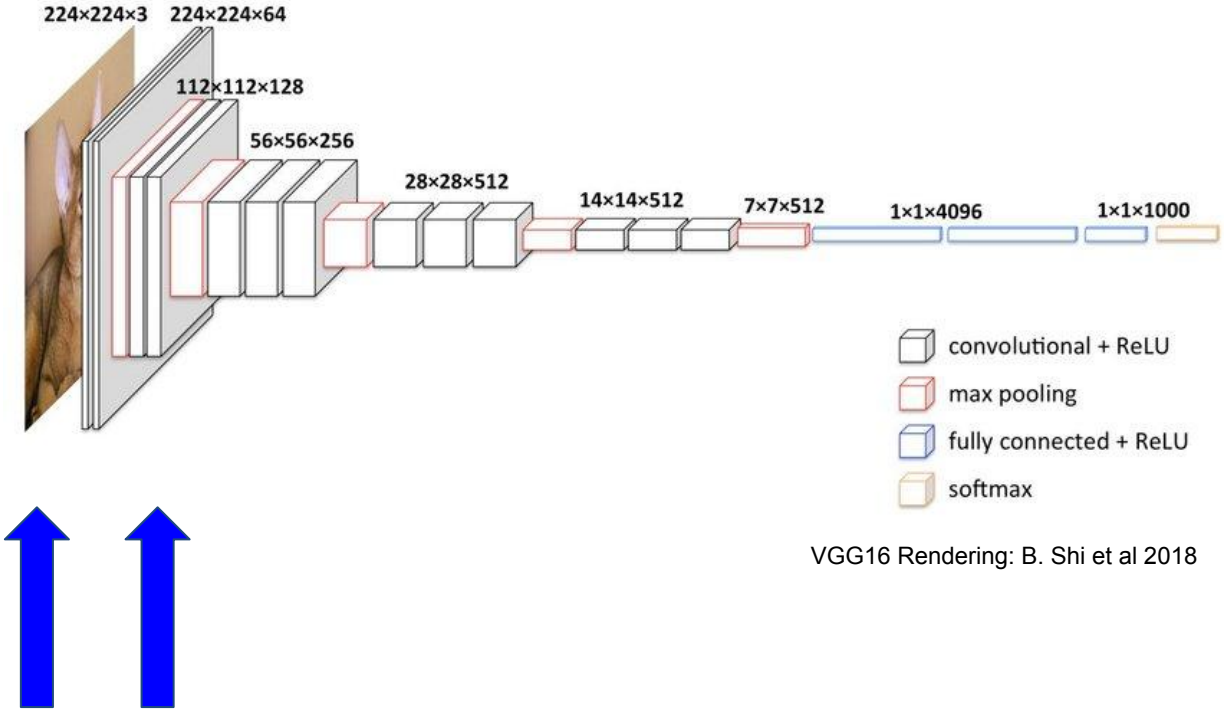
1. Split data into working and testing
2. ...
3. Profit

Supervised learning workflow

1. Split data into working and testing
2. ... Split working data into training and validation
3. ... Preprocess our data
4. ... Train a statistical model on the workdata we could find
5. ... Apply the model
6. Profit

Convolutional Neural Networks

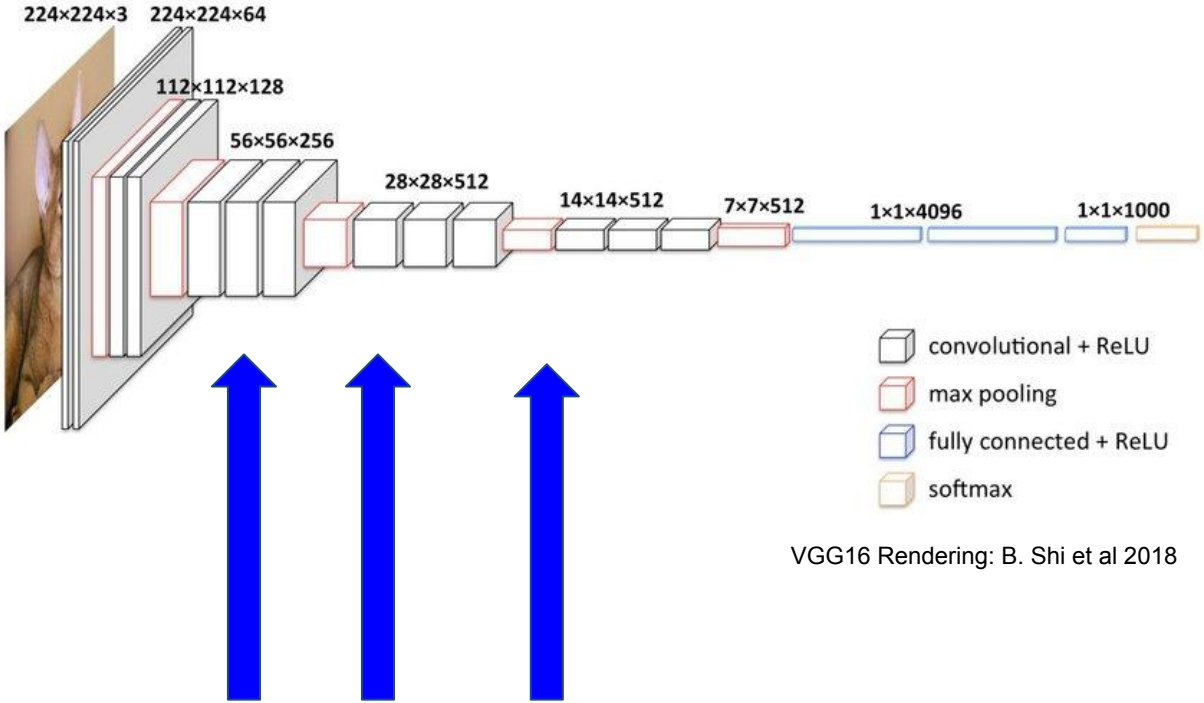
An image is propagated through a sequence of convolutional layers



VGG16 Rendering: B. Shi et al 2018

Convolutional Neural Networks

Deeper layers have more channels so we have to reduce the resolution with pooling

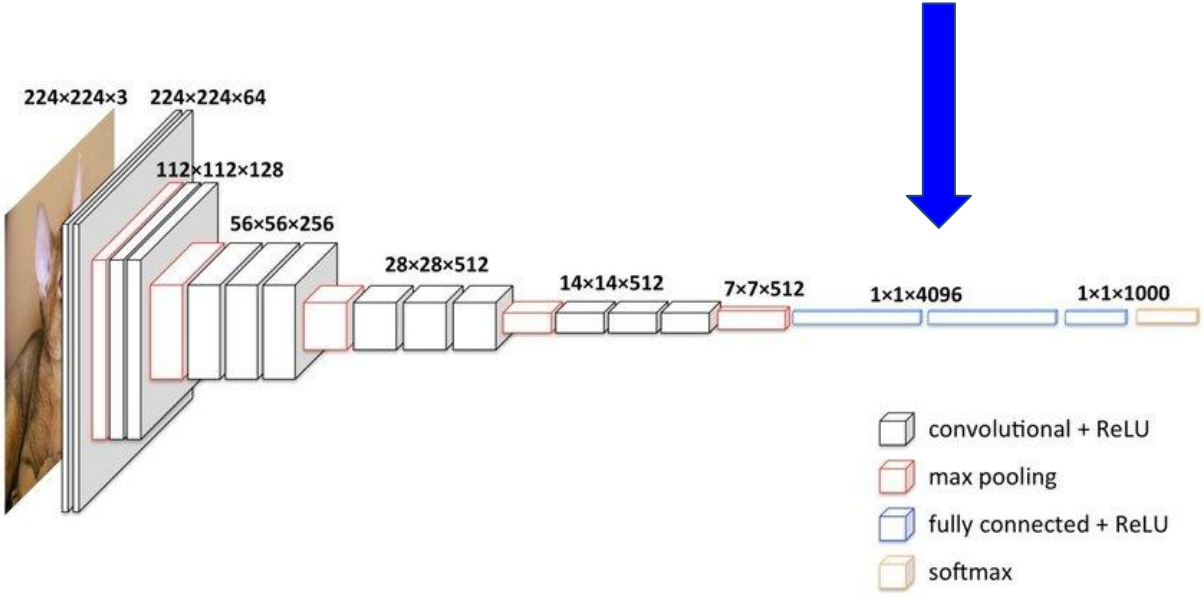


VGG16 Rendering: B. Shi et al 2018

Convolutional Neural Networks

Eventually the representation is 1 pixel with many Channels

In essence a vector



VGG16 Rendering: B. Shi et al 2018

Back Error propagation

Chain Rule of differentiation

$$h(x) = f(g(x))$$
$$h'(x) = f'(g(x))g'(x).$$

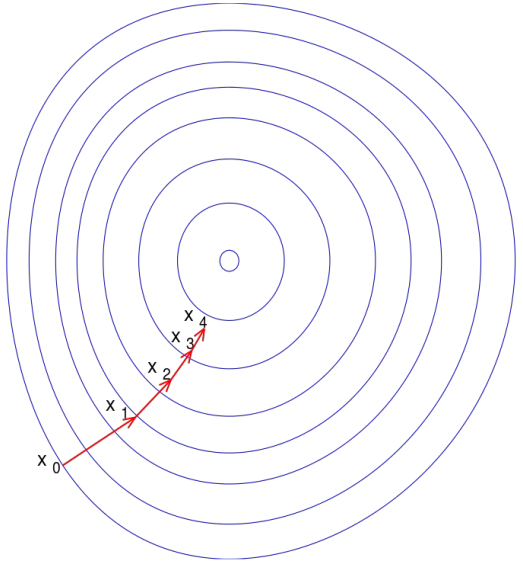
Applied to the errors computed on the output

x are the model parameters

Can we define the error gradient (slope) with respect to the model parameters?

Gradient Descent

- How should we change the parameters so that we reduce the error?
- How do we get to sea level from a mountain with fog?
- Stochastic: subset of the train set

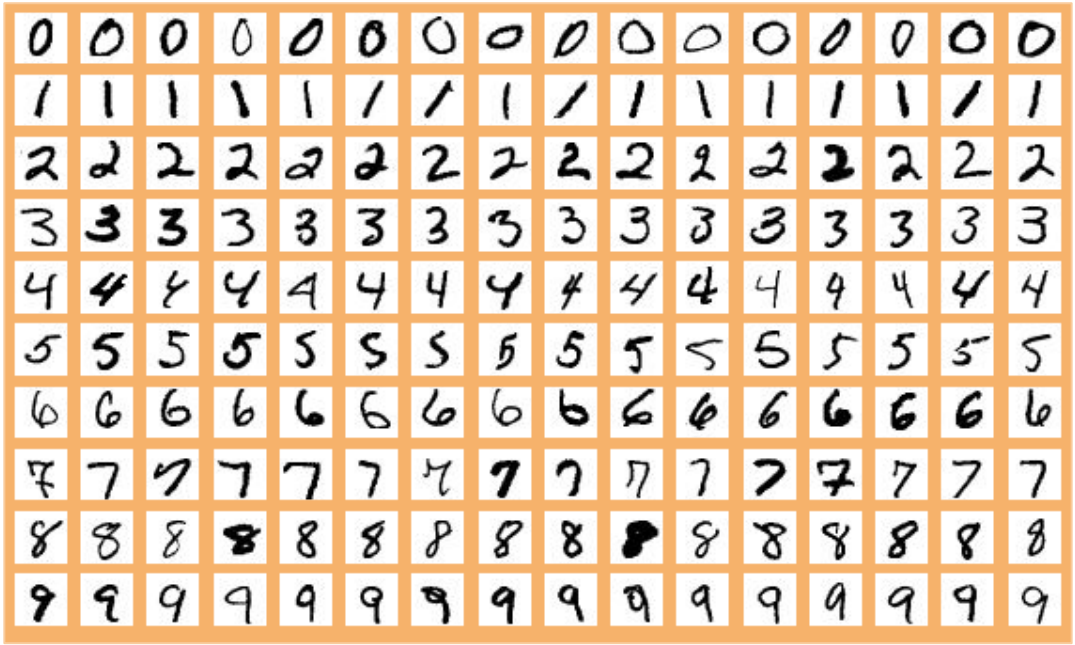


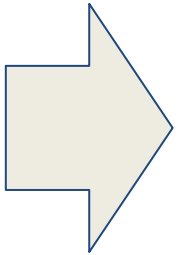
GD If we only had 2 parameters

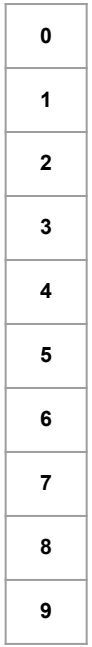
Hardware Considerations

- Linear Layers
  - |Parameters| > |Activations|
- Convolutional layers
  - |Activations| > |Parameters|
- Activations:
  - Cached in memory when training
  - Proportional to FLOPS => energy
- Weights:
  - Size of storing a trained model
  - How much can we learn?
- All must fit inside the GPU

Image Classification







Y. Lecun et al. 1998 (Gradient-based learning applied to document recognition)

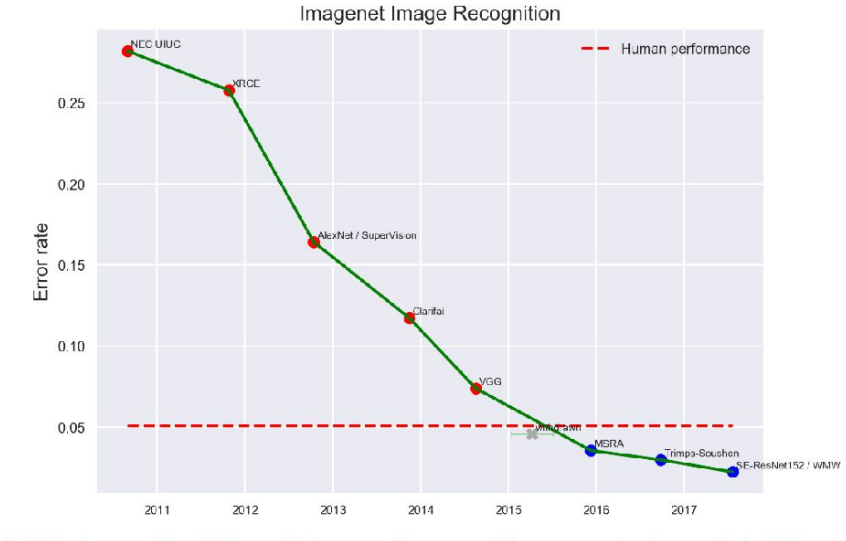
ImageNet

- 2009
- Based on wordnet schema
- 22K fine grained classes (wordnet)

https://production-media.paperswithcode.com/datasets/ImageNet-0000000008-f2e87edd_YofT5zg.jpg

ImageNet Challenge

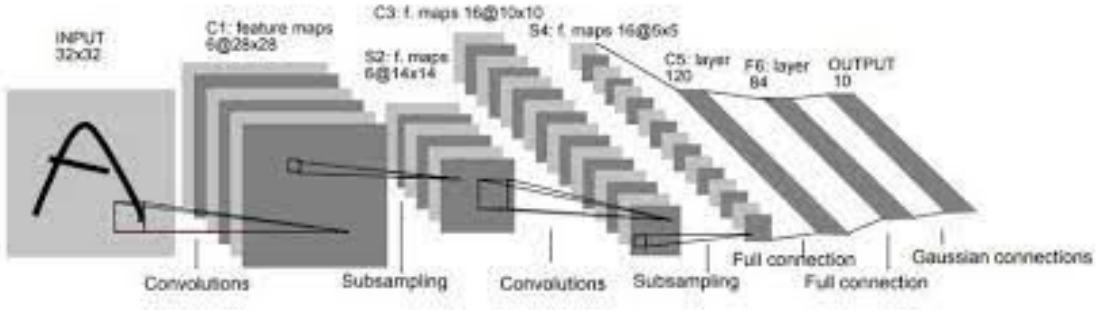
- 2011-2016
- Challenge 1K classes
  - mapped from 22K
- 1.2M Trainset Images
- Every year blind test
  - Sequestered data
  - 50K Validation set
  - 100K Testset
- In 2016 Baidu was caught cheating
- Standart CNN pre-trainer



A. Krizefski et al. : ImageNet classification with deep convolutional neural networks

CNN Evolution: Lenet-5

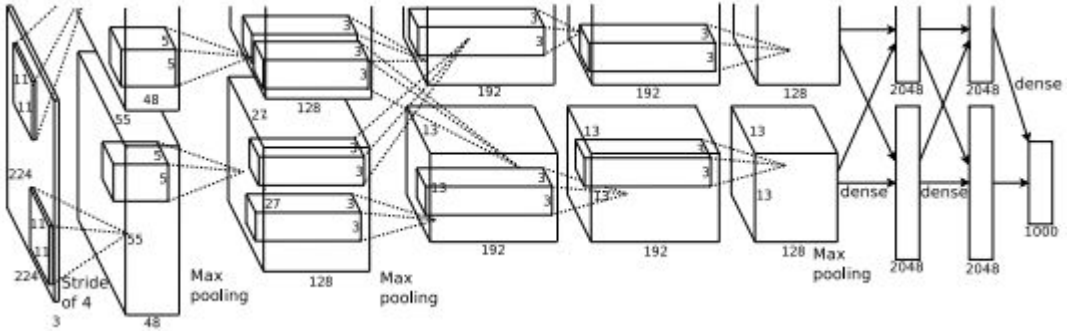
- 1998
- 5 learnable layers
- Sigmoid Activations
- 60K learnabel parameters



Y. Lecun et al. 1998 (Gradient-based learning applied to document recognition)

CNN Evolution: Alexnet

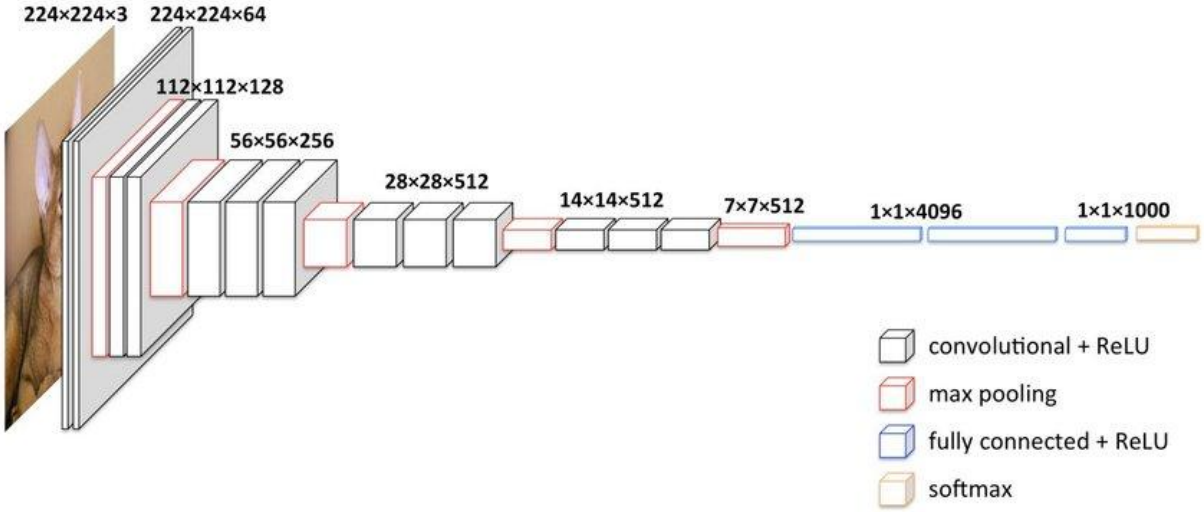
- 2012
- 9 Learnable Layers
- Many innovations:
  - GPU computation
  - Dropout
  - ReLU
- Imagenet SotA
- 60M parameters



Krizhevsky et al. 2012 (ImageNet Classification with Deep Convolutional Neural Networks)

CNN Evolution: VGG

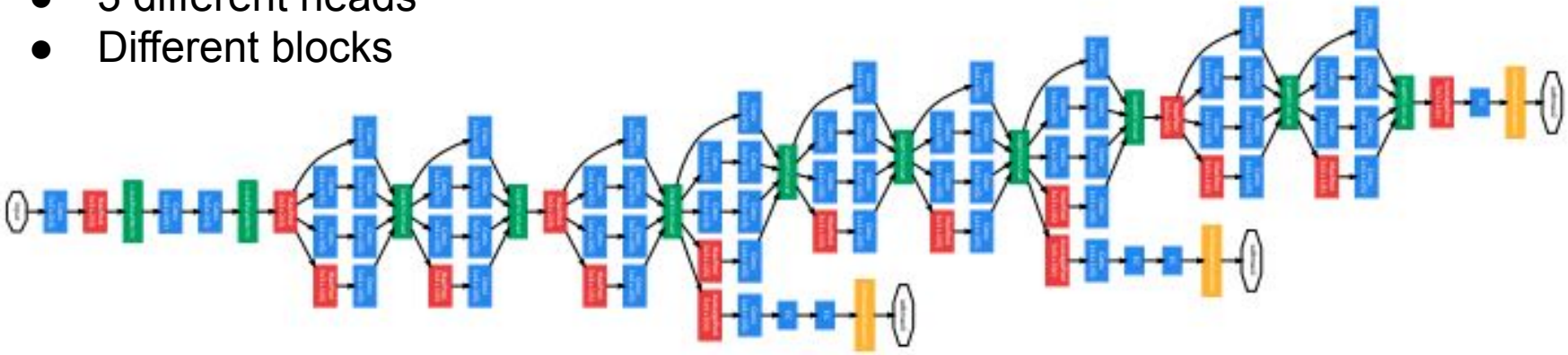
- 2014
- 19 Learnable Layers
- 3x3 kernels
- Different blocks
- Imagenet SotA*
- 138M parameters



VGG16 Rendering: B. Shi et al 2018

CNN Evolution: GoogLeNet

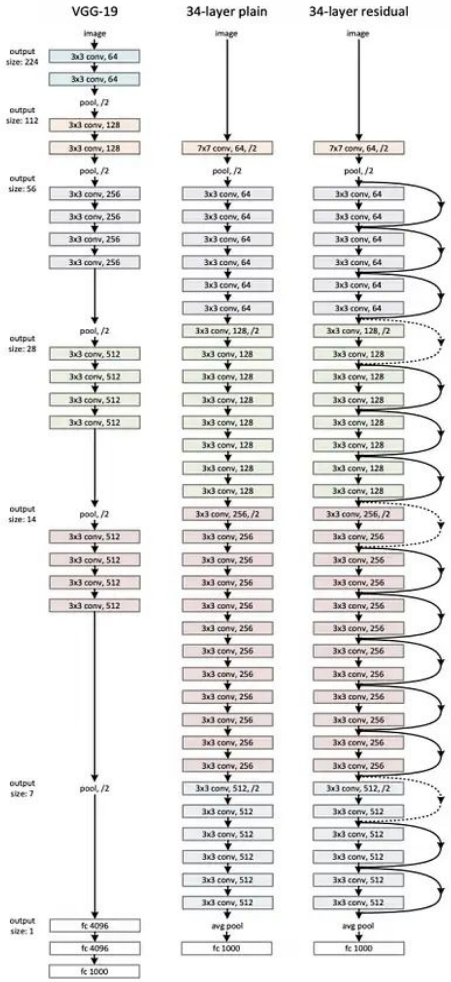
- AKA Inception
- 2014
- Very Deep
  - Longest path 21 learnable layers
- 3 different heads
- Different blocks



C. Szegedy et al. (Going Deeper with Convolutions)

CNN Evolution: ResNet

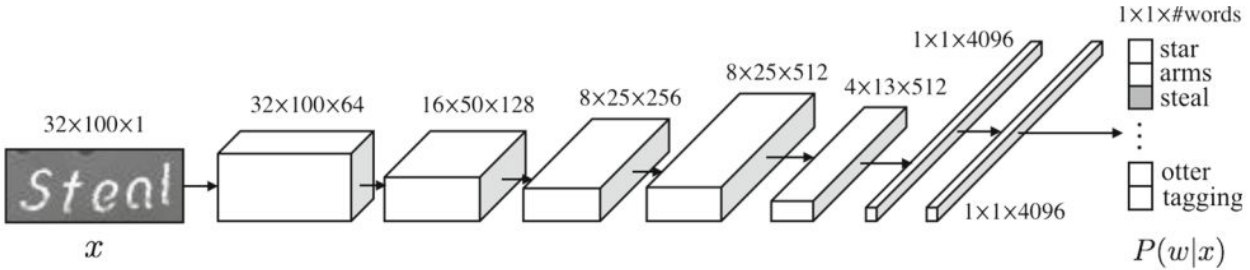
- Residual Connections
  - Same input and output size for a block
  - Output added to the input
  - Learn to forget instead of learn to remember
  - Highway for the gradient from output to input
- 2015
- Very Deep
  - ResNet50 48 learnabel layers, 23M parameters
  - Experiments > 1000 layers
-



K. He et al. 2015 (Deep Residual Learning for Image Recognition)

Classification: Word Spotting

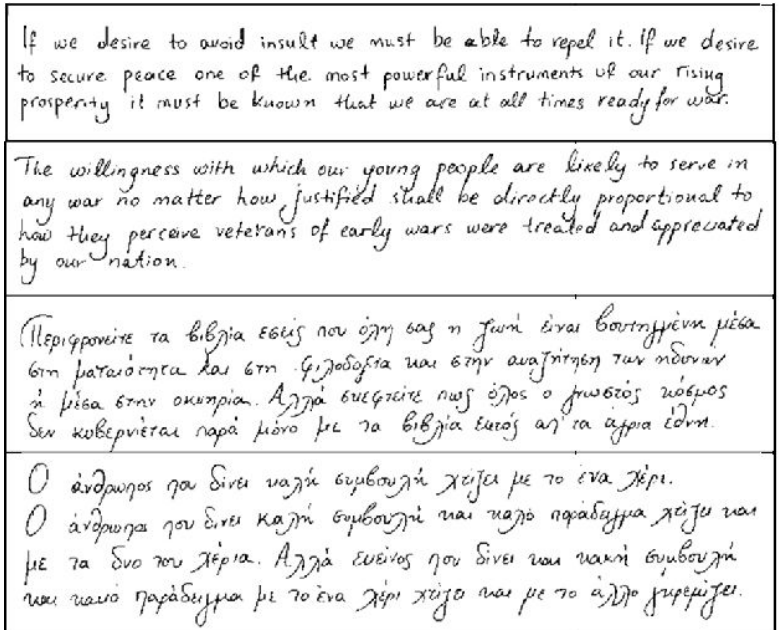
- When we can't read, we can choose
- 80K classes
- Aspect ratio not preserved



M. Jaderberg et al. 2013

Classification: Texture

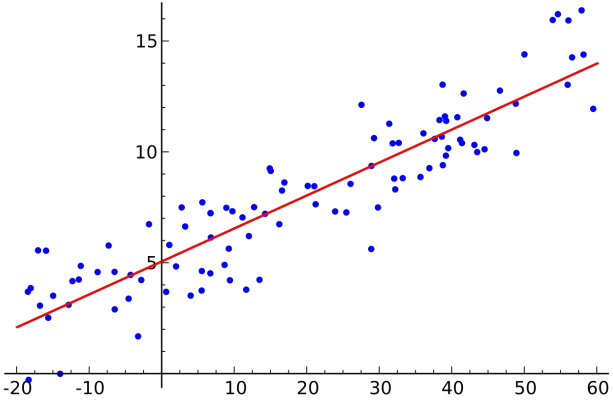
- Classify materials
- Script identification
- Writer identification
- Non learned features work quite well also



Ntirogiannis et al. 2013

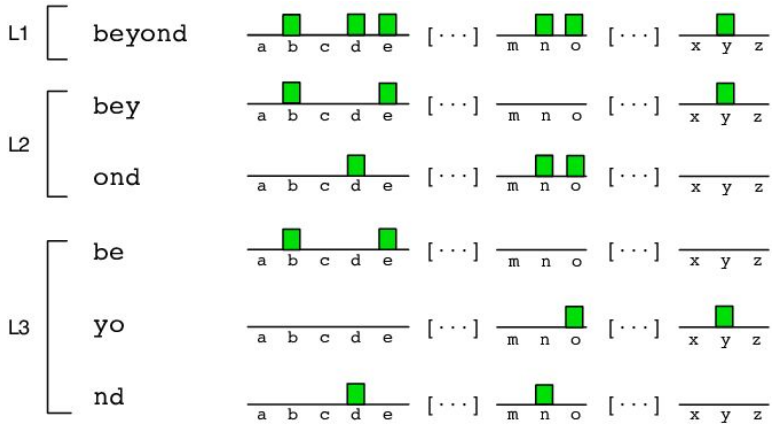
Regression

- Loss Mean Square Error
- Learning a continuous variable from an image
- Eg:
  - guessing age from portrait
  - guessing weather temperature from from a scene image
- Why not learn many outputs, a vector for an image?



Regression: PHOC

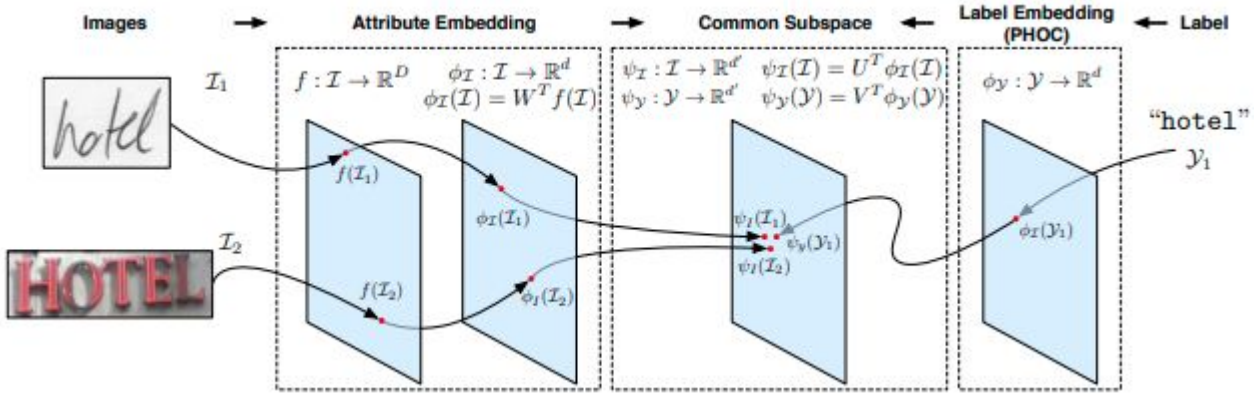
- Analyse word a bag of letters
- At different scales
- (1+2+3+5) * 24 histogram bins
- All strings are embedded in R^264



J. Almazan et al. 2014 (Word Spotting and Recognition with Embedded Attributes)

Regression: PHOC

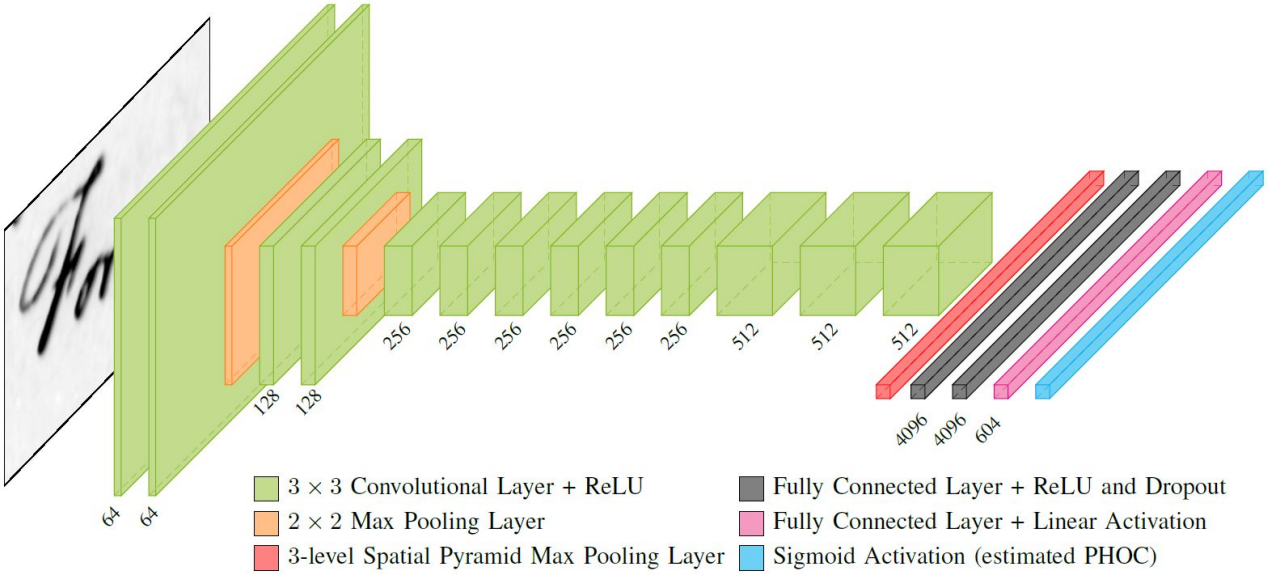
- We can put images of text in the same space as PHOC
- Any kind of text
- And then it all retrieval



J. Almazan et al. 2014 (Word Spotting and Recognition with Embedded Attributes)

Regression: PHOCNet

- PHOC Net
- Deep CNN
- Regression loss



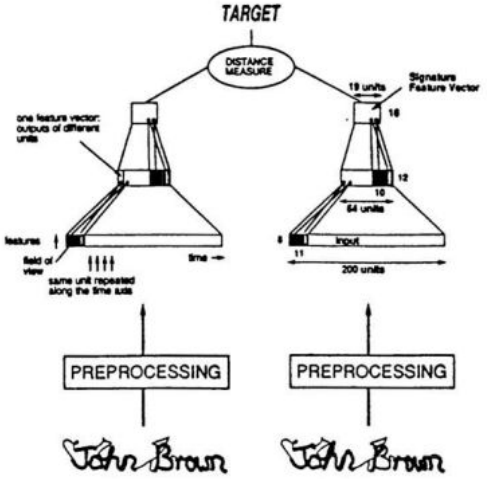
S. Sudhold et al. 2016 (PHOCNet)

Metric Learning

- Writer Identification
- Face identification
- Can turn to a classifier as a Nearest Neighbor

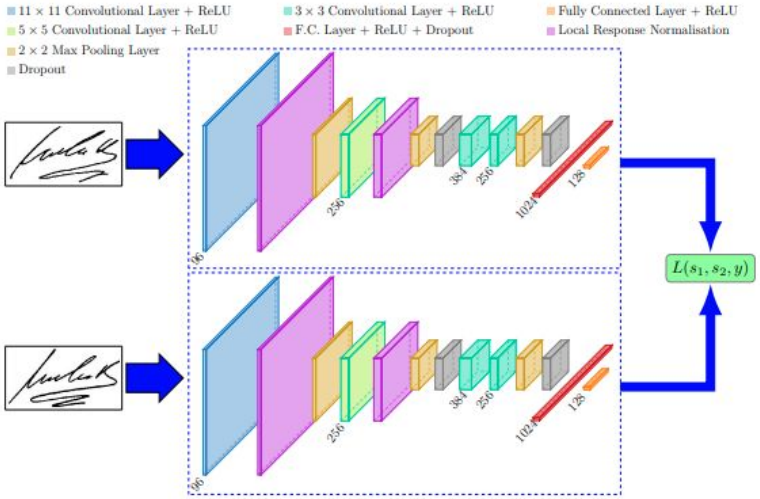
Metric Learning: Siamese Networks

- Let's minimise the distance for similar samples
- Introduced in 1994 by Jane Bromley et al.
- Siamese : symmetric
- Pseudo-siamese: asymmetric



Metric Learning: Siamese Networks

- Modern case
  - Triplet loss usually works better
  - Hard negative mining
- Other applications
  - learning point descriptors



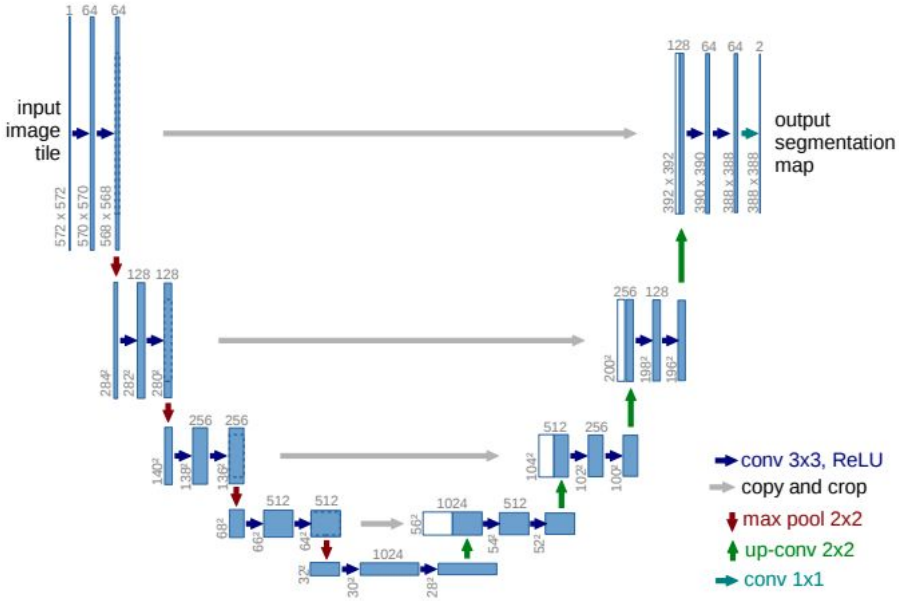
Dey et al. 2016

Object Detection

- **Y**ou **O**nly **L**ook **O**nce
- Algorithm:
  - Detecting lots of boxes
  - Regressing their edges
  - Post-processing: conflicting predictions fight over space
- Extremely fast

Segmentation: U-Net

- Autoencoder
- Skip-through connectio
- Fully convolutional
- Self supervision

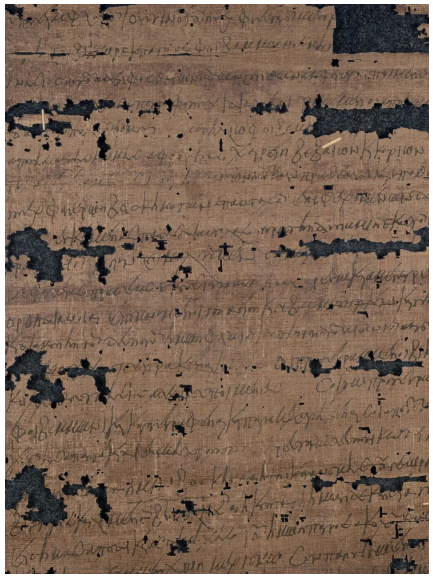


O. Ronneberger et al. 2016 (U-Net)

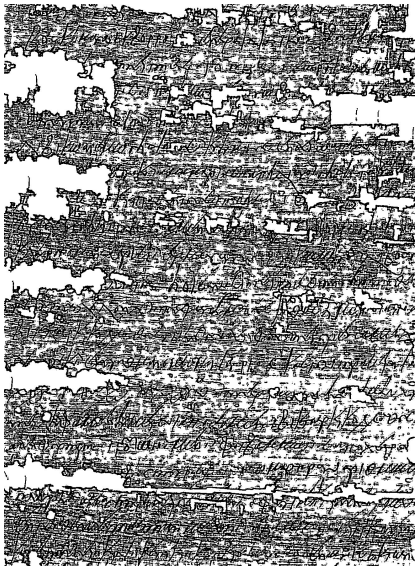
Segmentation: U-Net

- Trained on pure synthetic data
- Reversible U-Net allows full images

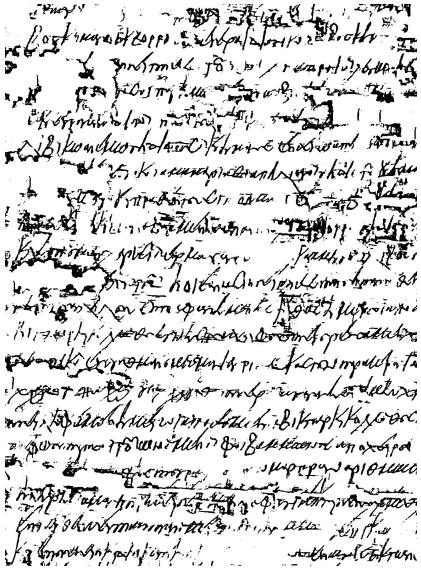
Input



Binarization Baseline



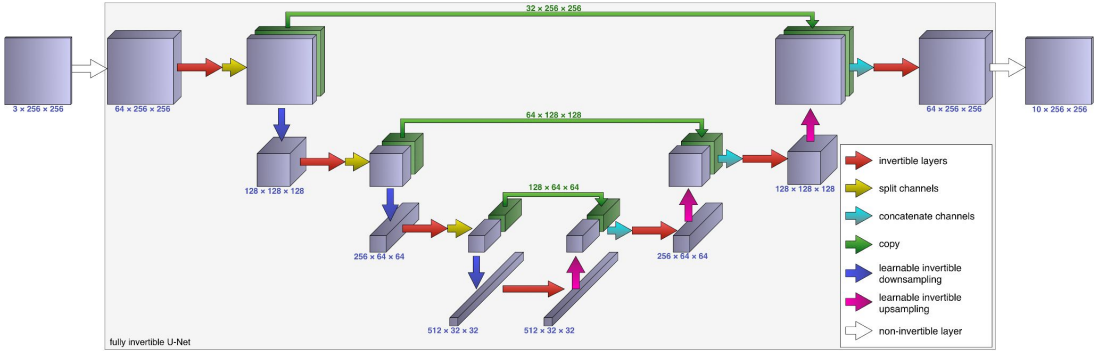
Synthetically trained U-Net



Christlein et al. 2022(Writer Retrieval and Writer Identification in Greek Papyri)

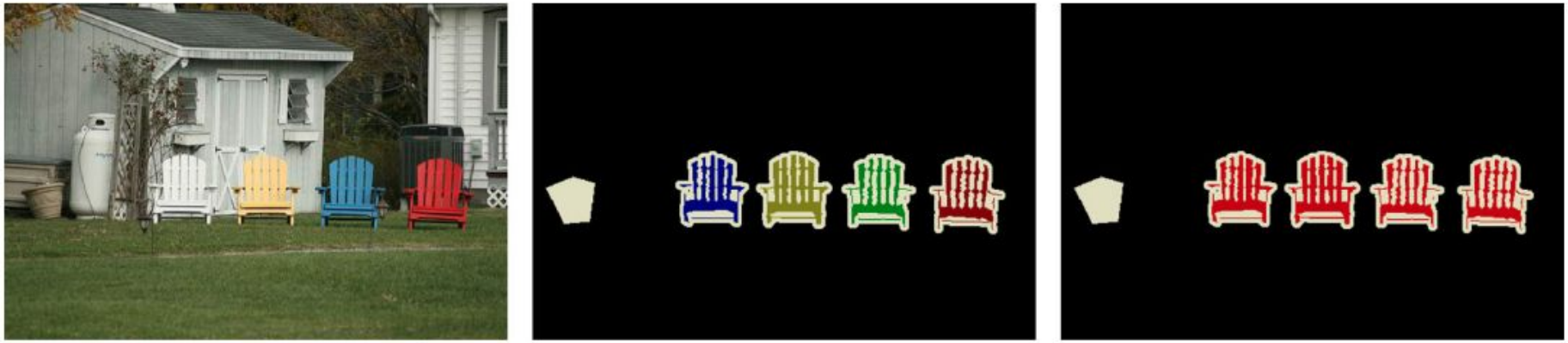
Segmentation: U-Net

- Trained on pure synthetic data
- Reversible U-Net allows full images



C. Etmann et al. 2020(iUNets: learnable invertible up-and downsampling for large-scale inverse problems)

Instance Segmentation

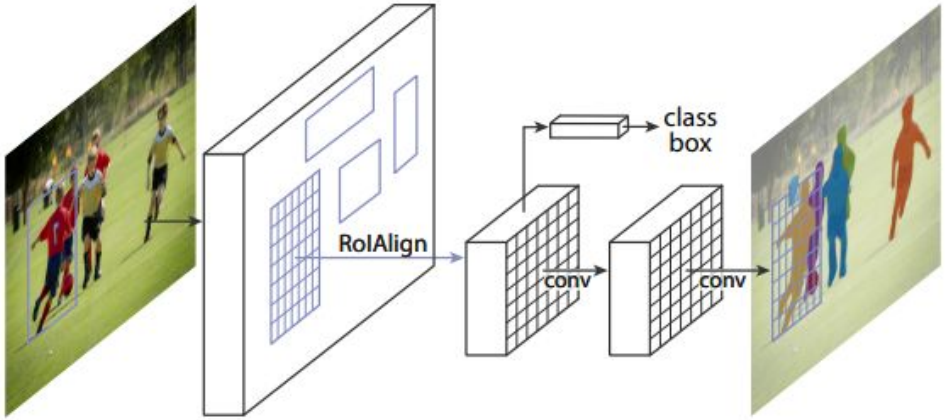


Pascal VOC Everingham et al. 2014

- When you need to count items
- Quite more challenging
- Big datasets: MS-COCO, Pascal-VOC

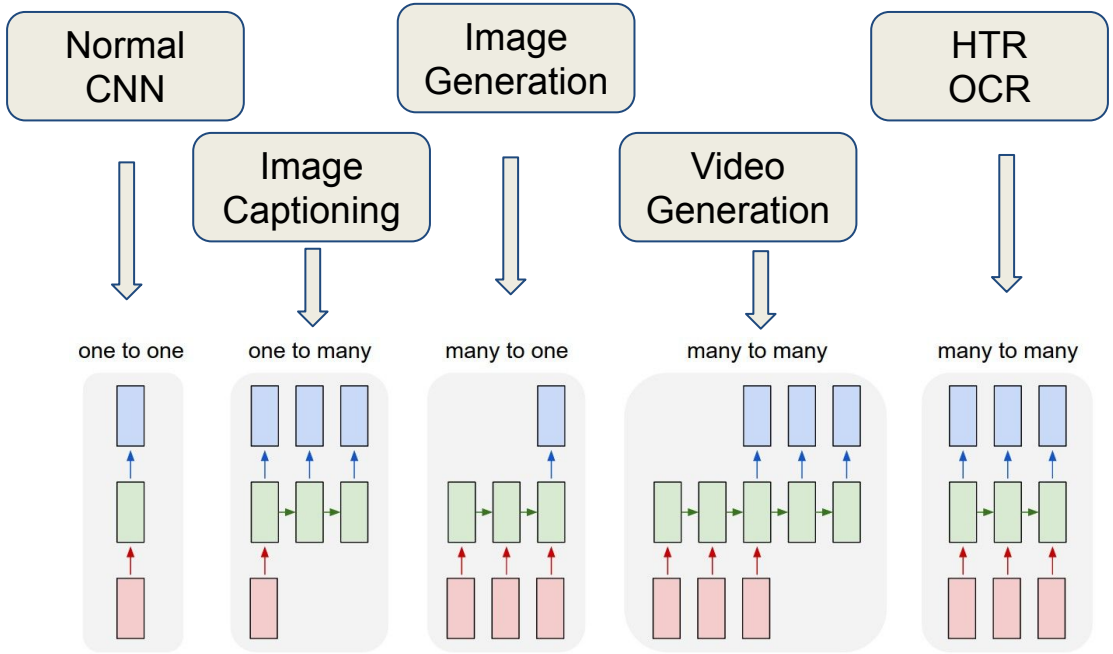
Instance Segmentation

- Masked RCNN
  - Faster RCNN
  - RCNN
- Learn to classify all possible regions
- Learn to propose all object regions
- Learn to binarize each proposed region



K. He et al. 2017 (Masked RCNN)

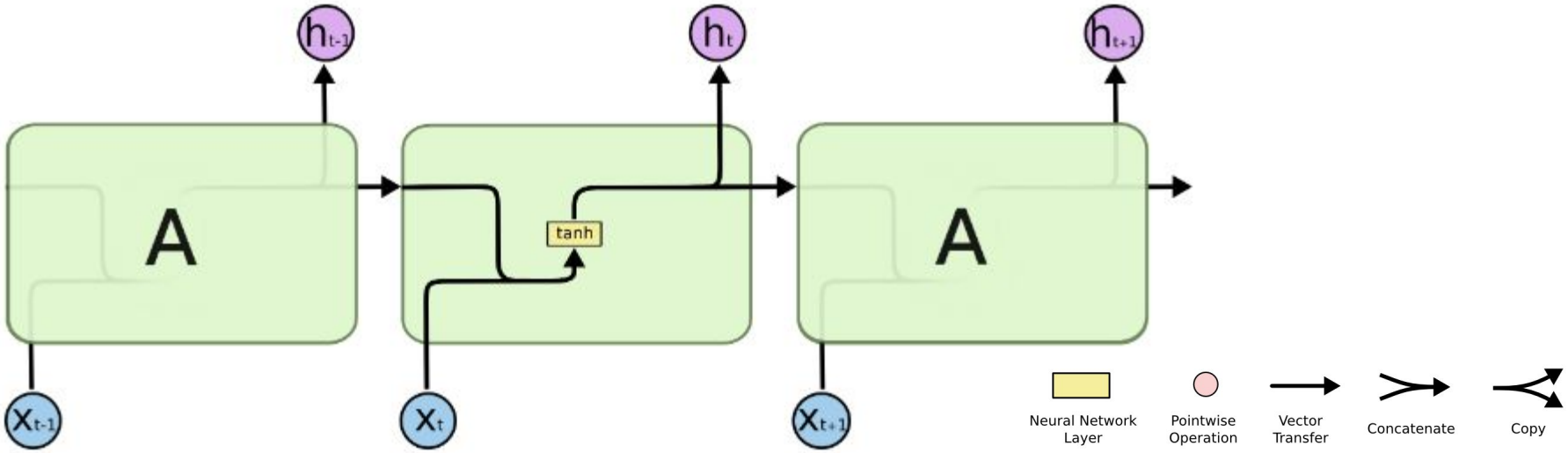
Sequences



http://karpathy.github.io/2015/05/21/rnn-effectiveness/

Sequences: LSTM

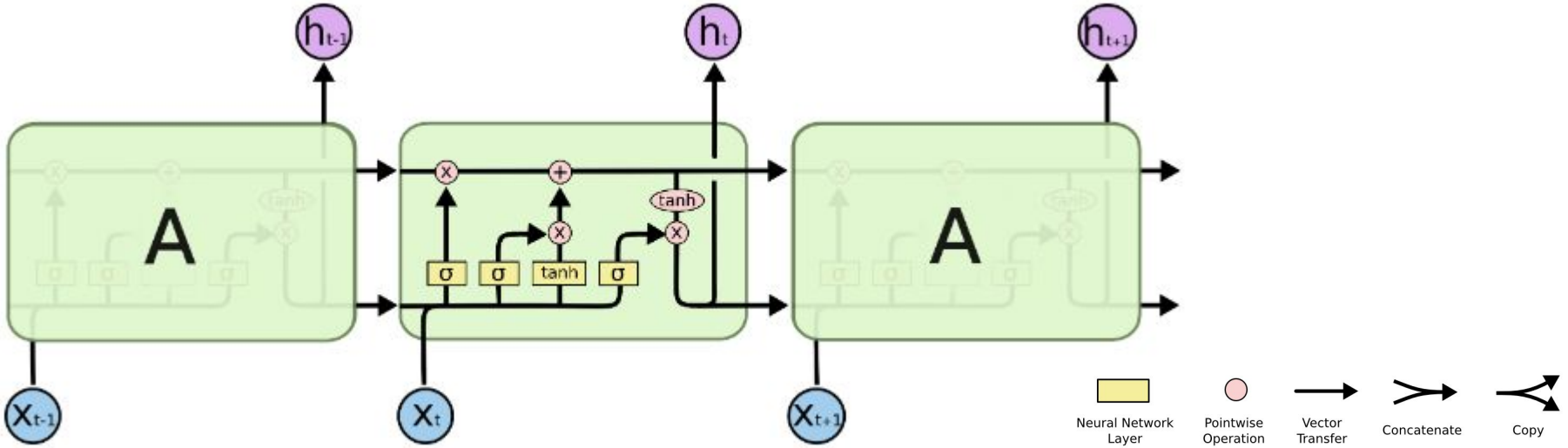
- We concatenate the new input to the previous output
- Pass them through a single linear layer with a sigmoid nonlinearity
- And on and on it goes



https://colah.github.io/posts/2015-08-Understanding-LSTMs/index.html

Sequences: LSTM

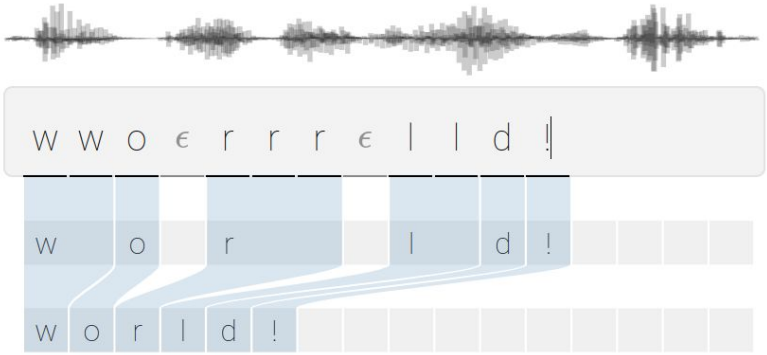
- But also we choose what to preserve from the input
- And what to forget
- We finally realise that what we know should not be what we say



https://colah.github.io/posts/2015-08-Understanding-LSTMs/index.html

Sequences: CTC Loss

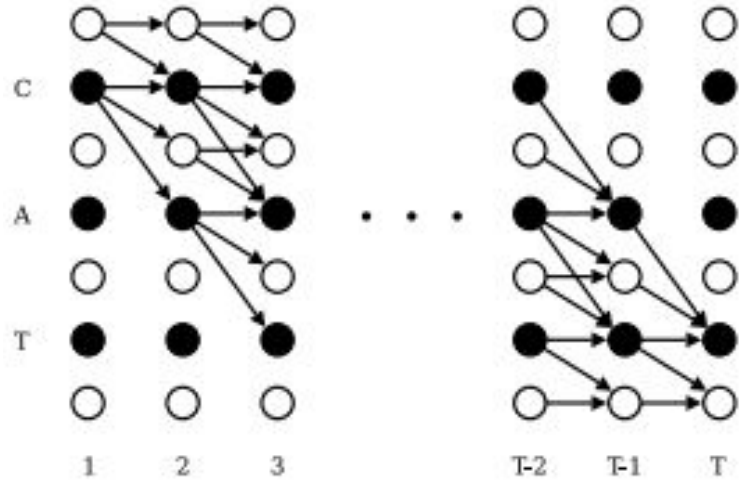
- Unaligned inputs and targets
- Applicable to class outputs (symbols)
- Allow I don't know
- Remove consecutive predictions of the same symbol
- Lots of work during inference (beam search)



https://distill.pub/2017/ctc/

Sequences: CTC Loss

- Dynamic programming
- Solve a problem by optimally caching partial solutions
- String edit distance



A. Graves et al. 2006 (Connectionist Temporal Classification)

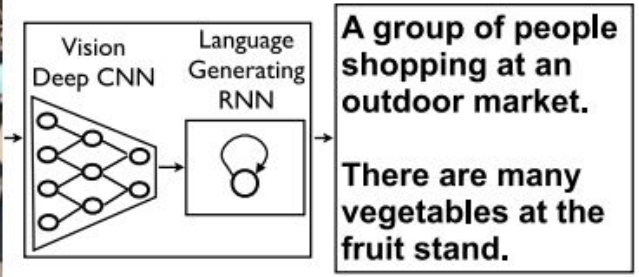
Sequence to Sequence: OCR

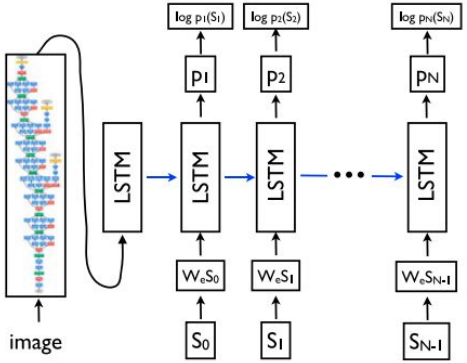
Ul-Hasan et. al 2013

- Initially RNN models
- Now transformers dominate
- Aligned sequence to sequence

Image to Sequence: Captioning

- GoogLeNet describes an image
- We teach an LSTM to identify "words" in the image
- We teach the LSTM to stitch that in good english



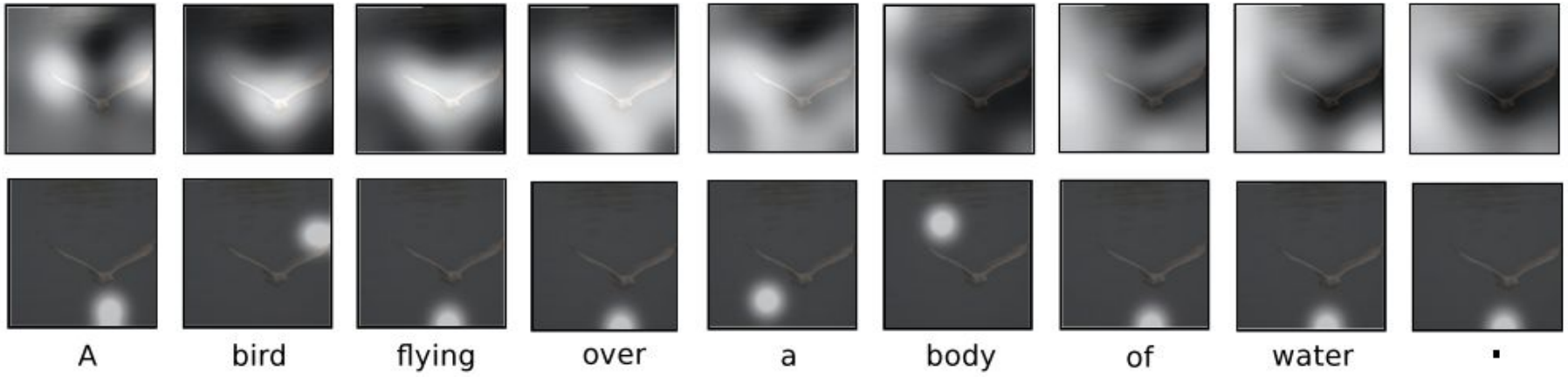


O. Vinyals et al. 2016 (Show and tell)

Image to Sequence: Captioning

- The importance of Attention
  - Soft
  - Hard

K. Xu et al. 2015 (Show, Attend and Tell)



K. Xu et al. 2015* (Show attend and tell)

Sequence to image:

- Extremely demanding

vibrant portrait painting of Salvador Dalí with a robotic half face

a shiba inu wearing a beret and black turtleneck

A. Ramesh et al. 2022 (Hierarchical Text-Conditional Image Generation with CLIP Latents)

Sequence to image:

- Extremely demanding in resources
- Non deterministic
- Could be used for deep fakes
  - Ethics
- Ethical considerations

vibrant portrait painting of Salvador Dalí with a robotic half face

a shiba inu wearing a beret and black turtleneck

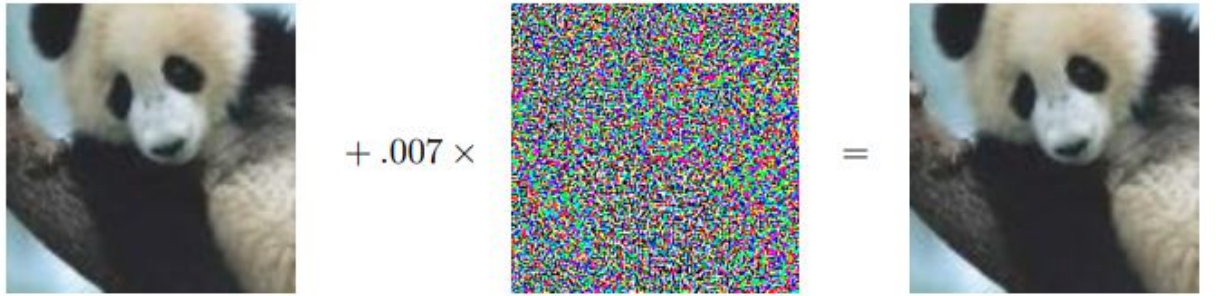
A. Ramesh et al. 2022 (Hierarchical Text-Conditional Image Generation with CLIP Latents)

Image Generation

- Adversarial Samples
- Style Transfer
- Generative Adversarial Networks
- Variational Autoencoders

Adversarial Samples

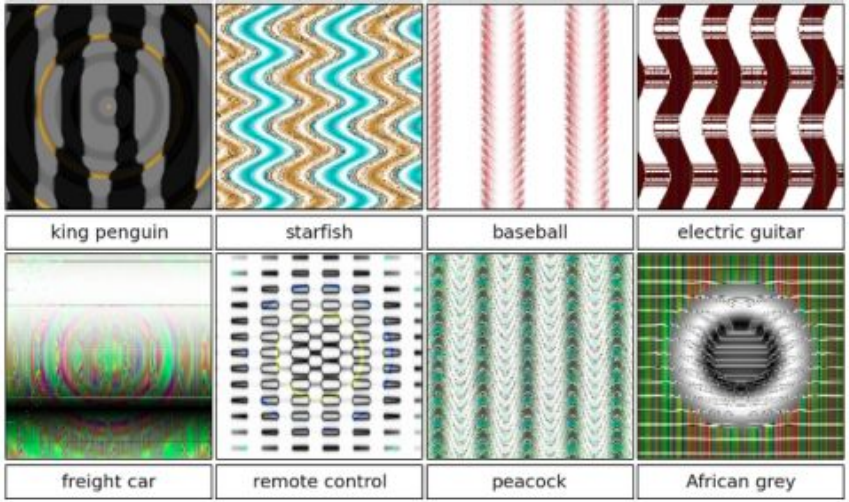
- Images that fool Neural Networks
- Computed with a trained network
- Let's find the minimum changes needed to fool the network



Goodfellow et al. 2014 (Explaining and harnessing adversarial examples)

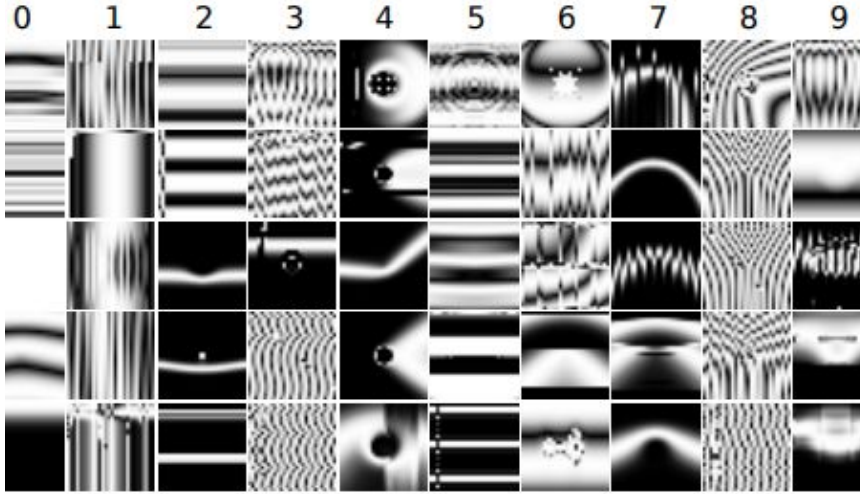
Adversarial Samples

- Computed with a trained network
- Network is a black-box
- Images are evolved with drawing primitives



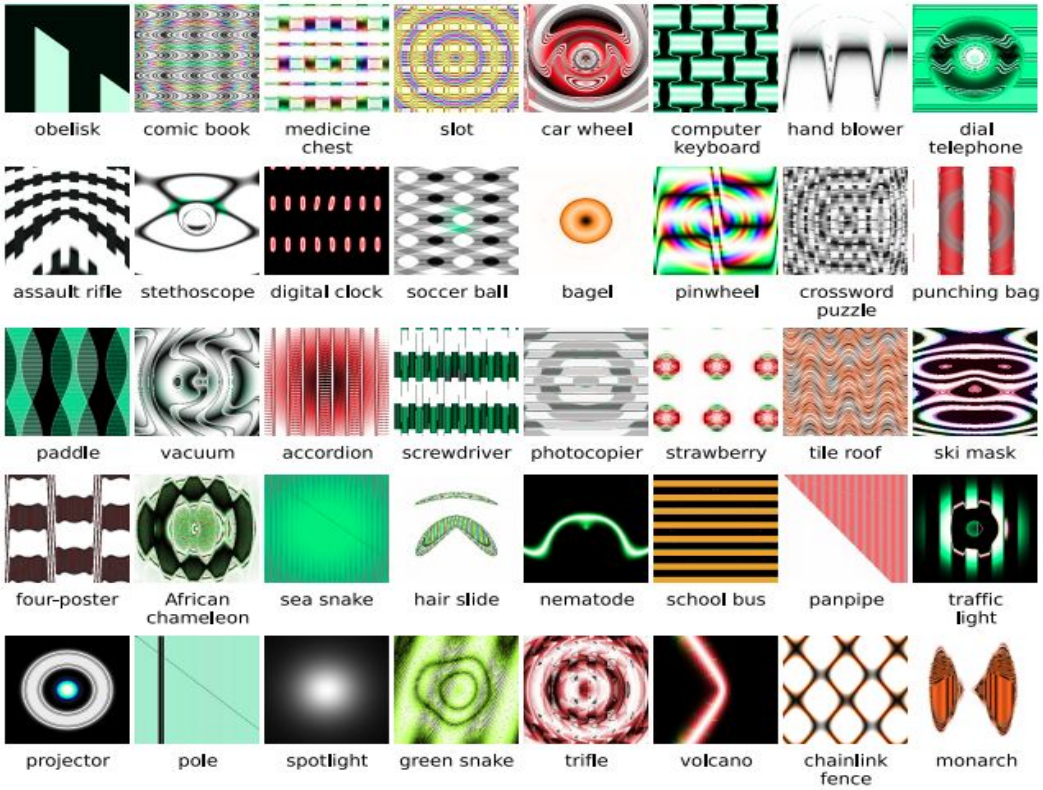
A. Nguyen et al. 2015 (Deep Neural Networks are Easily Fooled)

Adversarial Samples



A. Nguyen et al. 2015 (Deep Neural Networks are Easily Fooled)

Adversarial Samples



A. Nguyen et al. 2015 (Deep Neural Networks are Easily Fooled)

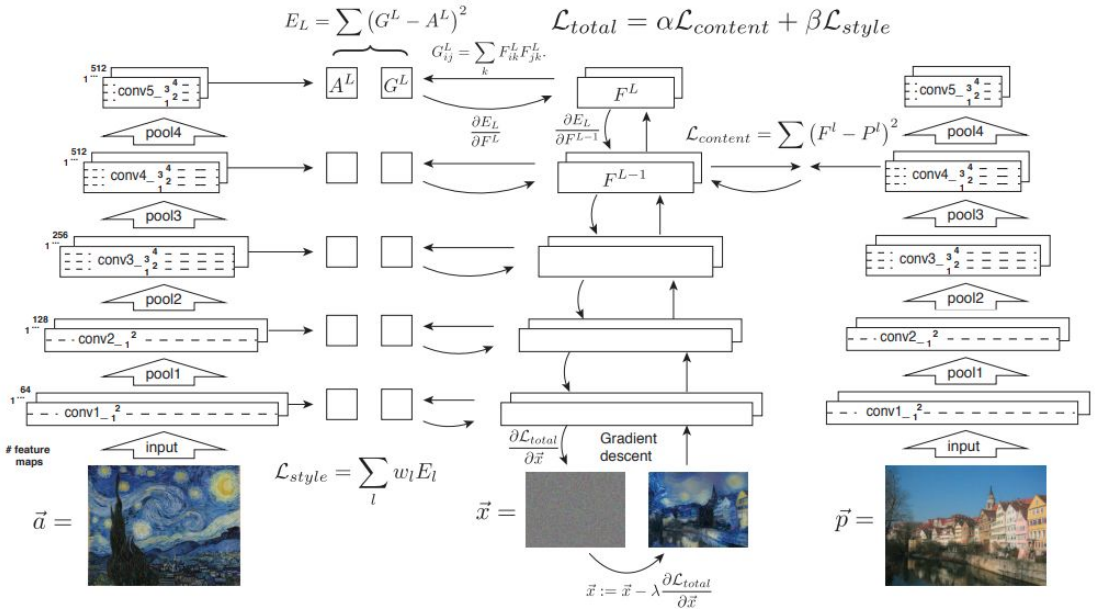
Style transfer

- 2016
- Generated directly by the optimizer
- No network training!
- Applicable on different scales (CNN layers)

L Gatys et al. 2016 (Image Style Transfer Using Convolutional Neural Networks)

Style transfer

- Make the deep layers resemble the content image
- Make early layers resemble the style image



L Gatys et al. 2016 (Image Style Transfer Using Convolutional Neural Networks)

Style transfer

- CNNs can't do physics

L Gatys et al. 2016 (Image Style Transfer Using Convolutional Neural Networks)

Generative Adversarial Networks

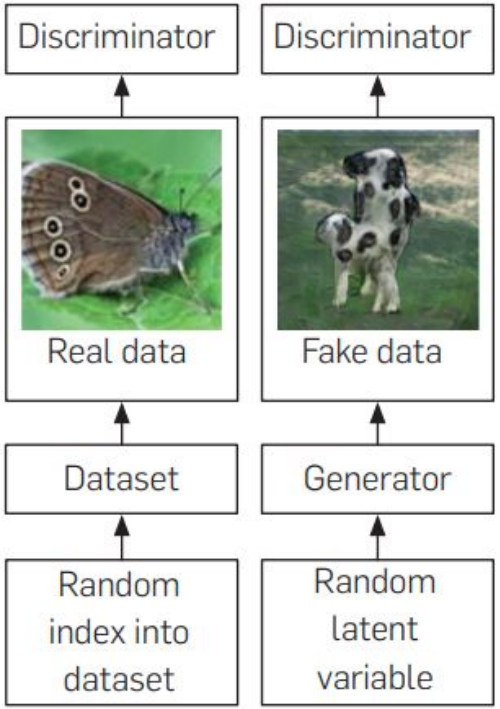
If we have no groundtruth what can we do?

Generative Adversarial Networks

Learn to mimic data!

Generative Adversarial Networks

- Invented in 2014
- Goodfellow et al.
- Lets have two networks fight!
  - One mimics the real data
  - The other tells them apart
- Find the Nash equilibrium



Goodfellow et al. 2020
(https://dl.acm.org/doi/pdf/10.1145/3422622)

Generative Adversarial Networks

- Notoriously hard to train
- Constant progress

Goodfellow et al. 2020
(https://dl.acm.org/doi/pdf/10.1145/3422622)

Generative Adversarial Networks

- Can invent things that never existed
- But a lot of work has gone into this

Goodfellow et al. 2020
(https://dl.acm.org/doi/pdf/10.1145/3422622)

StyleGAN

Learn high-level representations

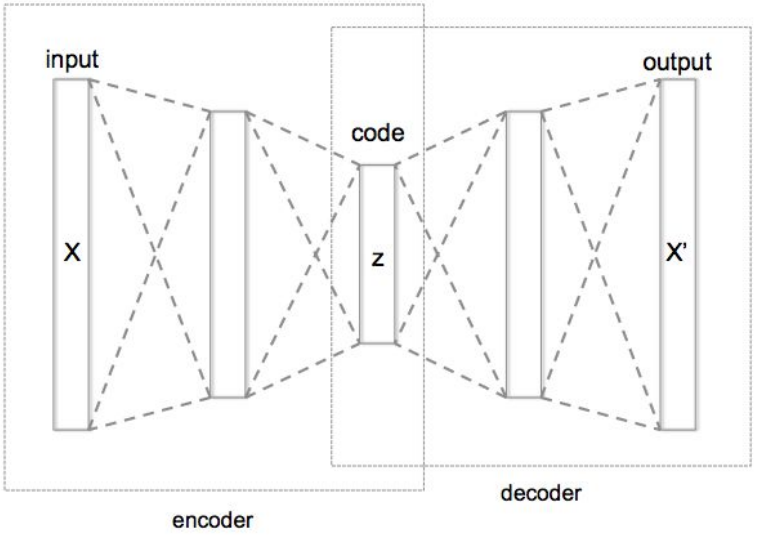
T. Karras et al. 2019 (A Style-Based Generator Architecture for Generative Adversarial Networks)

Generative Adversarial Networks

- Discriminative losses can be used in all sort of ways.
- They generate a pixel-level loss
- Without the effort of segmentation datasets

Autoencoders

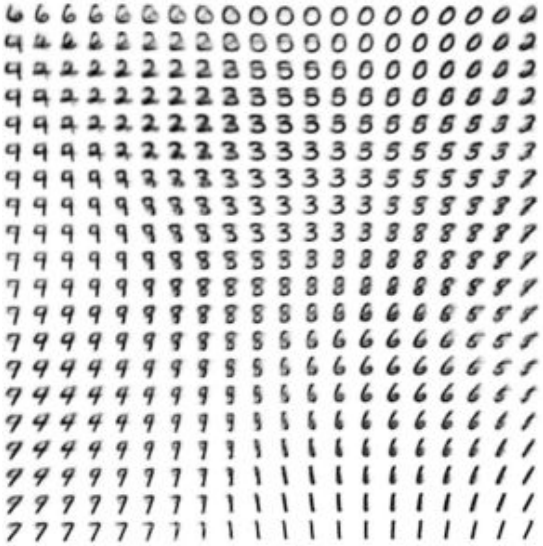
- Learn to reconstruct the input through a bottleneck layer
- Can be convolutional
- Self supervised compression



https://en.wikipedia.org/wiki/File:Autoencoder_structure.png

Variational Autoencoders

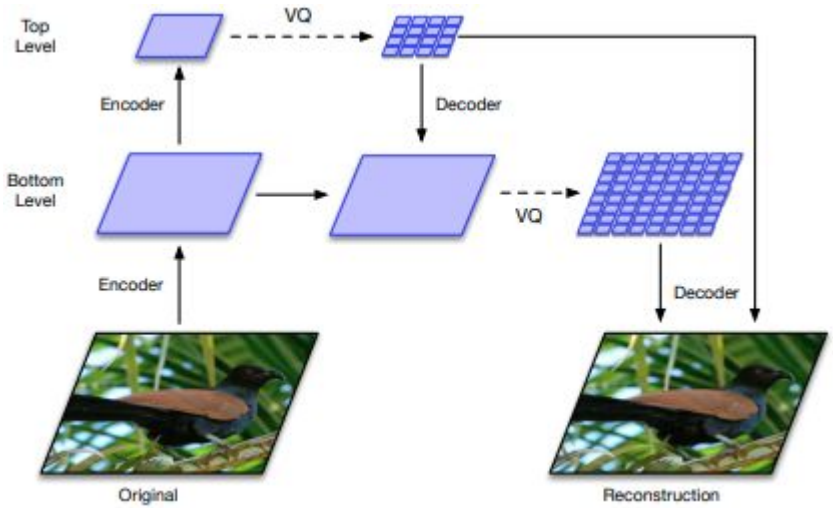
- Reparameterization trick:
  - Learn to predict the parameters of gaussian distributions sampled at the bottleneck
- This forces meaningful variables



P. Kingma et al. 2013 (Auto-Encoding Variational Bayes)

VQ Variational Autoencoders

- Vector Quantization
  - Learn to choose the most similar vector from a codebook



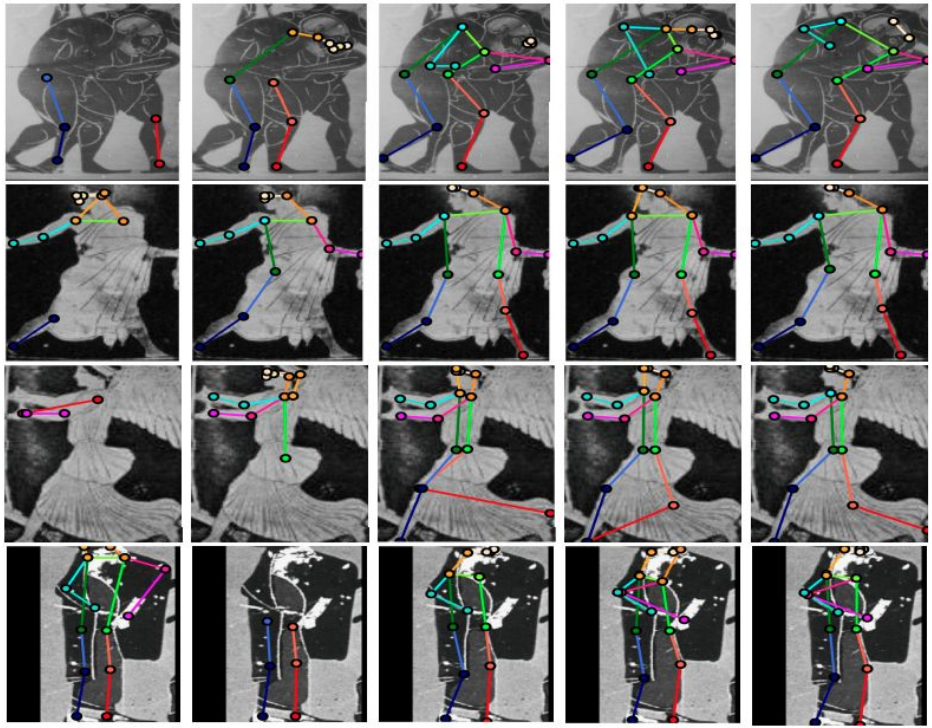
P. Kingma et al. 2013 (Auto-Encoding Variational Bayes)

Pose Estimation

P. Madhu et al. 2020

Pose Estimation

•



P. Madhu et al. 2020