Semantic Segmentation

Presented by :
Malav Bateriwala
Mohit Vaishnav

Supervisor :
Dr. Xavier Llado

Presentation Flow

- Introduction
- Semantic Segmentation

  What is semantic Segmentation

  Types of Semantic Segmentation
- Random Forest Approach

  Challenges of Traditional Semantic Segmentation
- Deep learning Approach
- Conclusions
- References

1. Introduction

- What is Segmentation?

  Segmentation is a partition of an image into several "coherent“(low level cues such as color or texture) parts, but *without* any attempt at understanding what these parts represent.

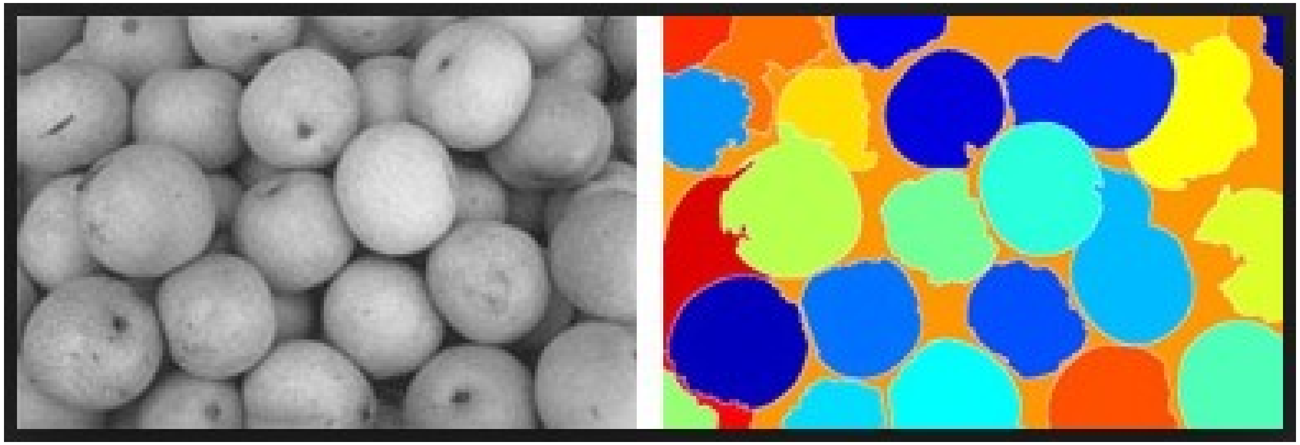


Image – Google Images

2. Semantic Segmentation

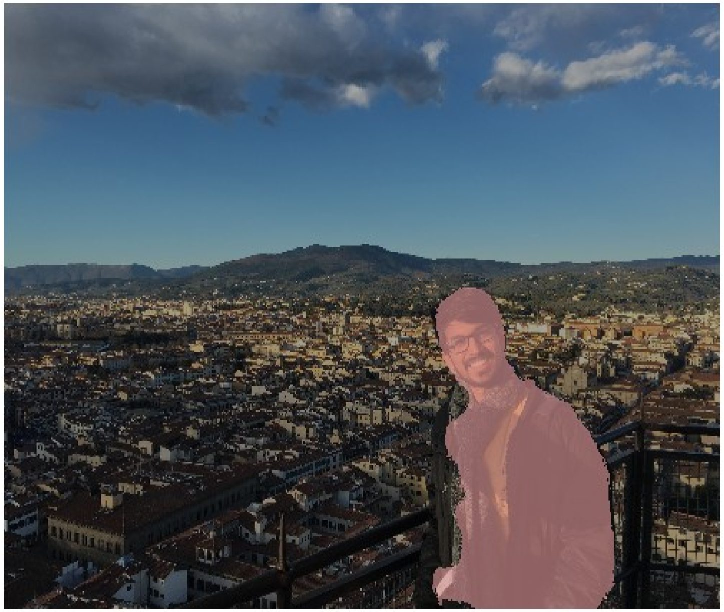
- What is Semantic Segmentation?

  When image is segmented into semantically meaningful parts, *and* each part is classified into pre-determined classes. In other words in Semantic Segmentation you will label each region of image.

Image – Oxford CS Blog

2. Semantic Segmentation

- Another examples of Semantic Segmentation



2. Semantic Segmentation

- Video of Semantic Segmentation of Car

Video – YouTube Video(Autonomous driving)

2. Semantic Segmentation - Types

- Traditional Approaches
  - Random Forest
  - Texton Forest

- Deep Learning Approaches (After 2012)
  - Patch Classification
  - FCN
  - SegNet
  - GoogLeNet

Random Forest Approach

3. Random Forest Approach

- The general method of random decision forests was first proposed by Ho in 1995.
- The introduction of random forests proper was first made in a paper by Leo Breiman in 2001.
- This paper describes a method of building a forest of uncorrelated trees procedure, combined with random nodes.

3. Random Forest Approach

- Decision Trees
- Trees answer sequential questions which send us down a certain route of the tree given the answer. The Model behaves with “if this than that” conditions ultimately yielding a specific result.
- In the Tree the nodes represent decisions and the edges or branches are binary (yes/no, true/false) representing possible paths from one node to another.

3. Random Forest Approach

- Flow of Random Forest Approach

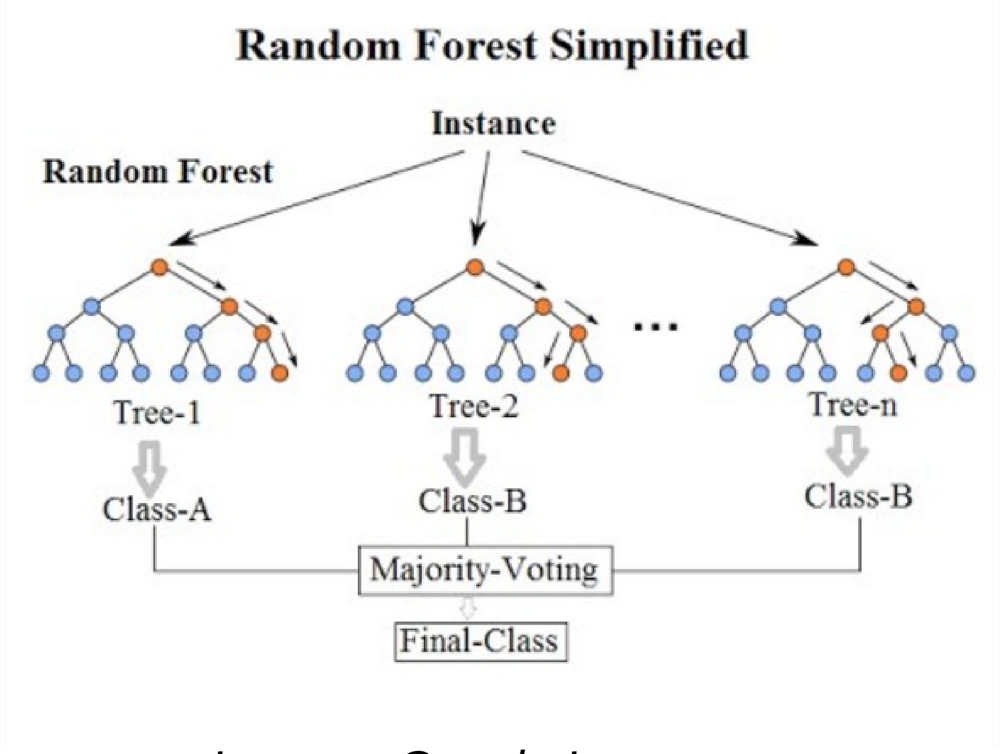


Image – Google Images

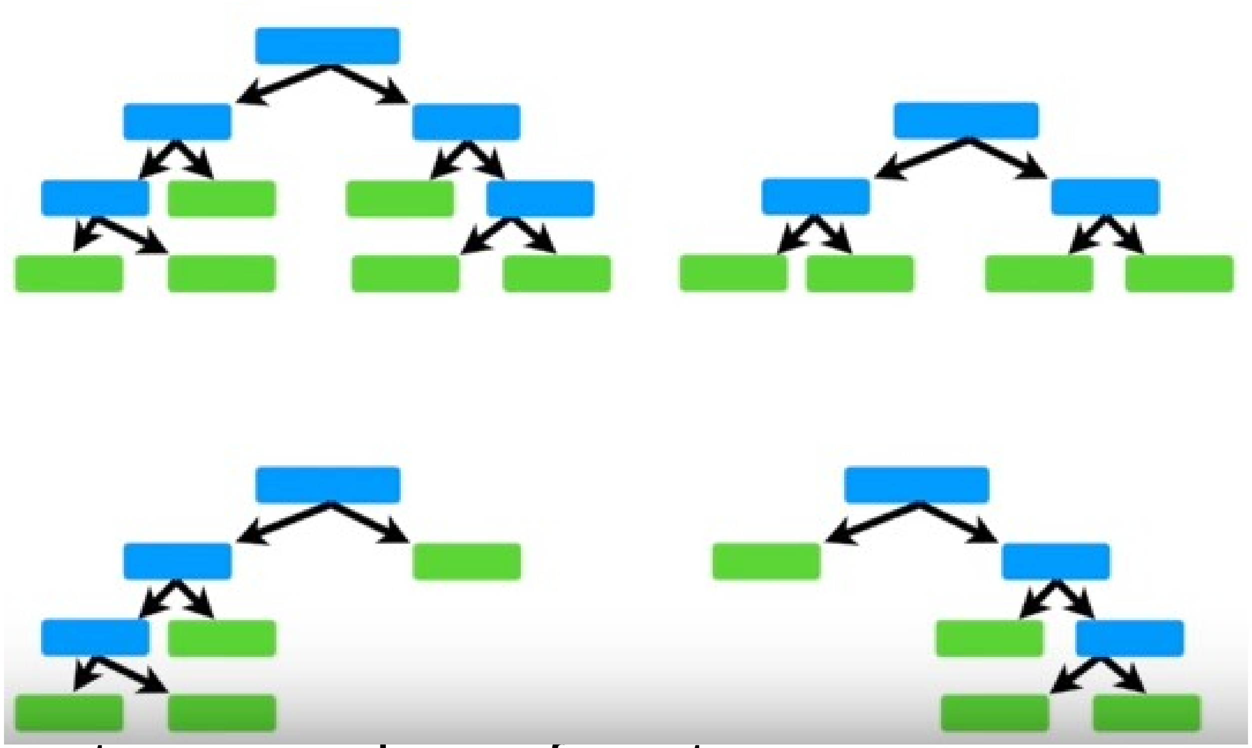
3. Random Forest Approach

- Typical Tree with multi Classes

Image – GitHub Kj13fH

3. Random Forest Approach

Defining a Decision Tree



So many trees , makes a forest.

3. Random Forest Approach

Output Classification

- Each tree will give us a output of a specific answer.
- So we iterate through all of these trees and get the weights of each answer.
- Finally we check if the weights of the answers and whichever has the majority, that will be the class of the region.

3. Random Forest Approach

Image – Microsoft Random Forest Example

3. Random Forest Approach

Appealing features of Random Forests.

1. Their computational efficiency in both training and classification compared to other traditional methods.
2. Their probabilistic output .
3. The seamless handling of a large variety of visual features (e.g. color, texture, shape, depth etc.)
4. They can be defined into many different classes.

3. Challenges – Traditional Approach

Why did we have deep learning approach?

1) We don’t require any tight assumptions about the data distributions.
2) More optimal options in techniques and approaches.
3) Deep learning promotes having more data, while traditional approaches tries to have a data reduction before modeling(Input, Sampling).

Deep Learning Approach

Fully Convolutional

- Fully connected layers can also be viewed as convolutions with kernels that cover their entire input regions

	AlexNet	FCN
Time	1.2ms	22ms
Image Size	227 x 227	500 x 500

Layers

- Here we list some of the most used layers
  - Convolution Layer
  - Max/Average Pooling Layer
  - Dropout Layer
  - Relu, Tanh, Sigmoid Layer (Non-Linearity Layers)

Architecture

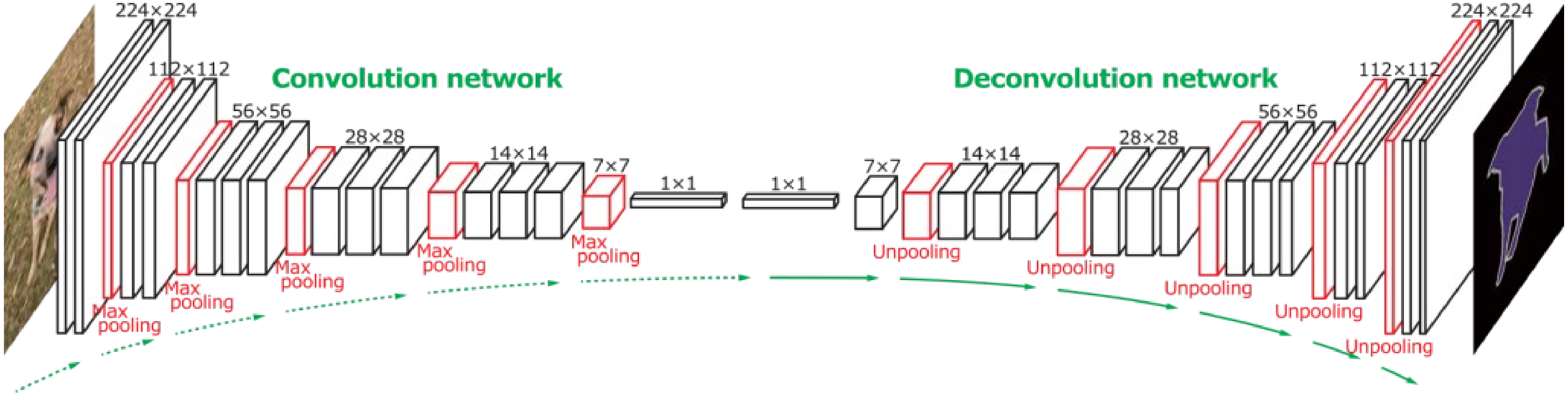


Image: https://medium.com/@wilburdes/semantic-segmentation-using-fully-convolutional-neural-networks-86e45336f99b

Convolutional Output

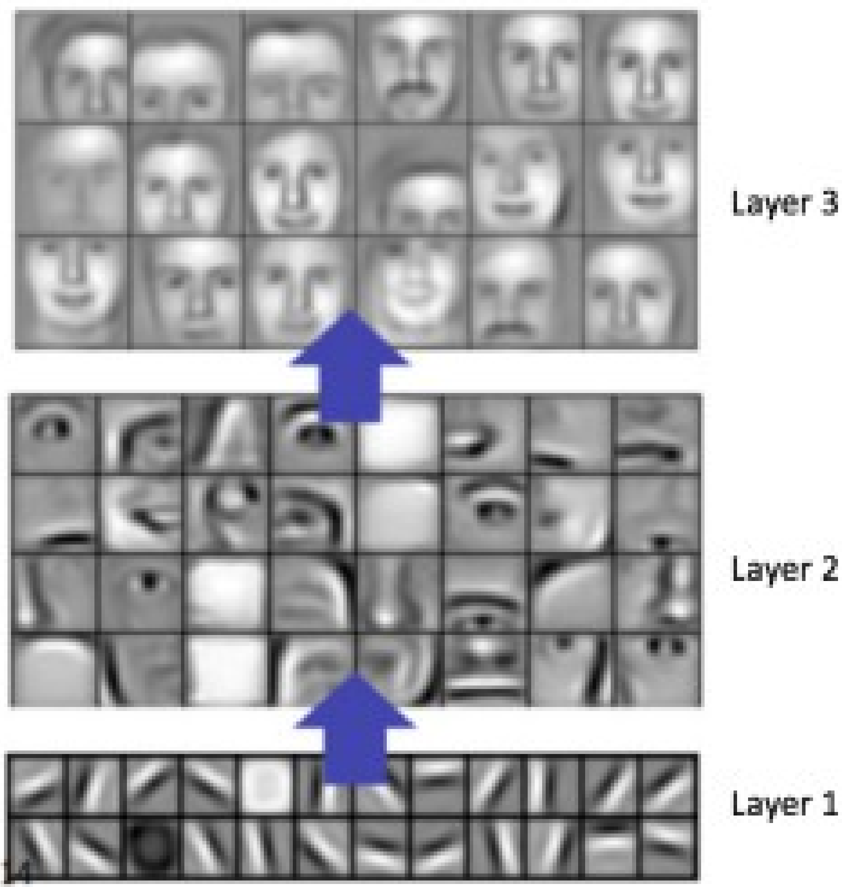
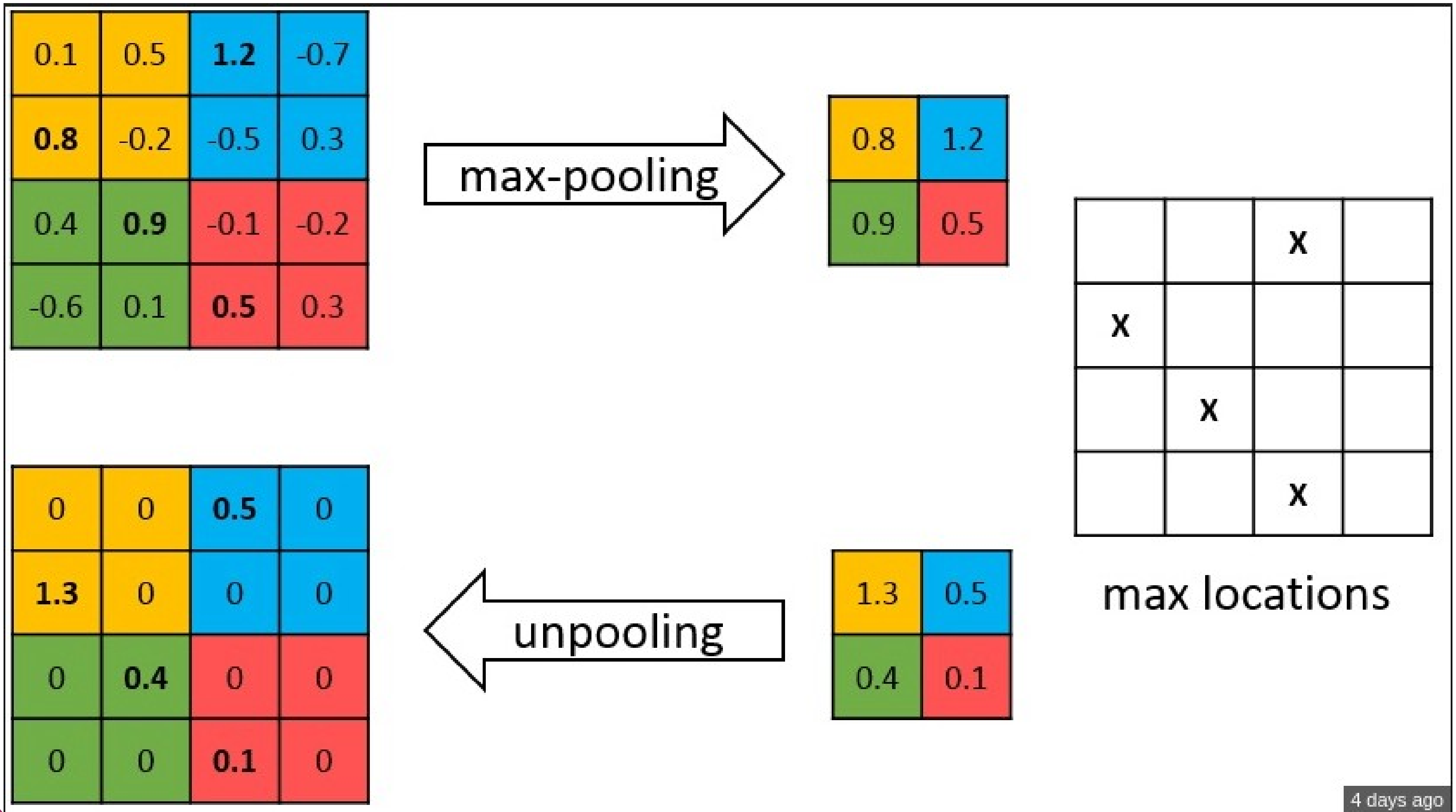


Image: https://ujjwalkarn.me/2016/08/11/intuitive-explanation-convnets/

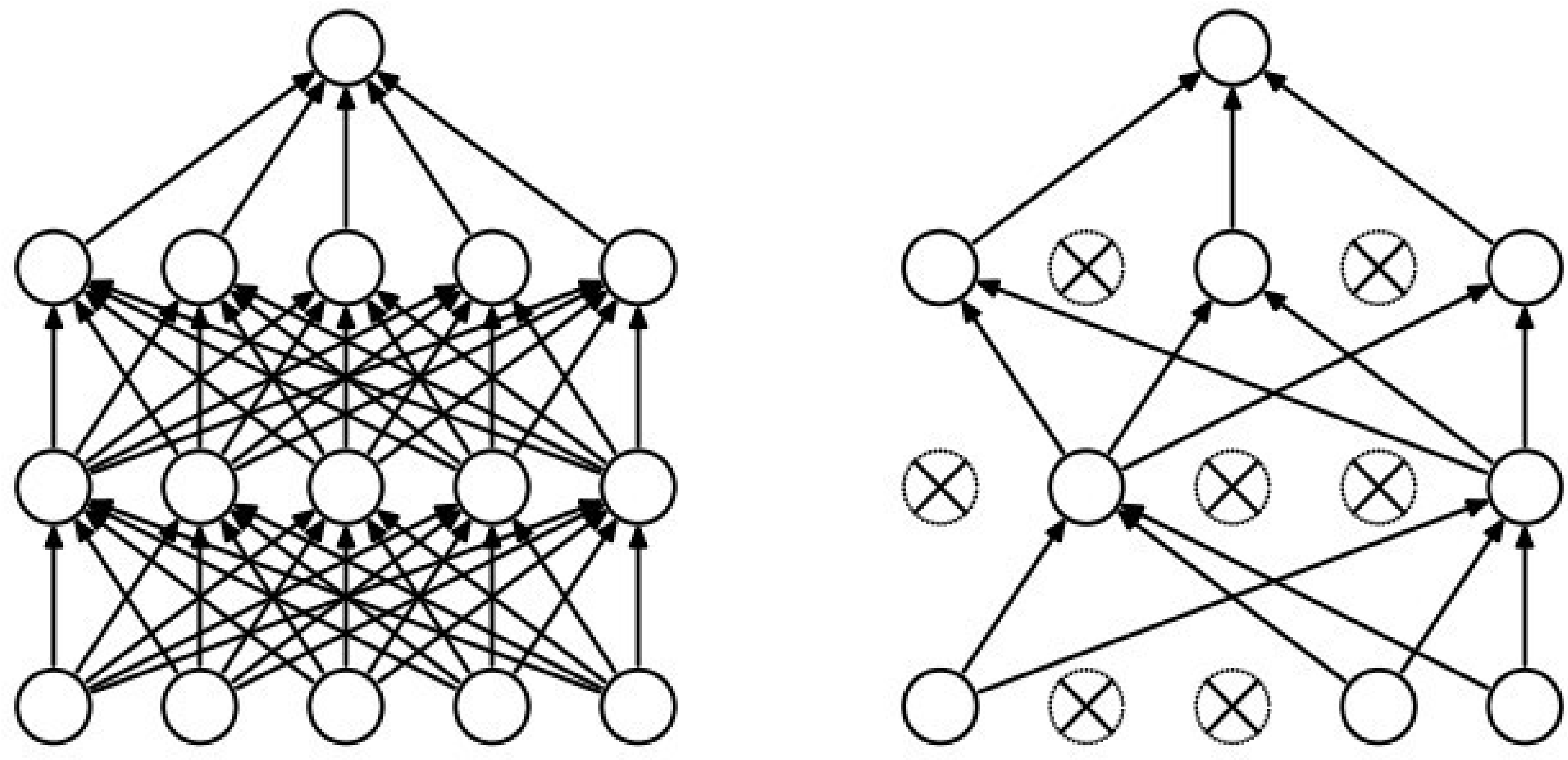
Pooling


0.1 0.5 1.2 -0.7
0.8 -0.2 -0.5 0.3
0.4 0.9 -0.1 -0.2
-0.6 0.1 0.5 0.3
max-pooling
0.8 1.2
0.9 0.5
x
x
x
x
max locations
1.3 0.5
0.4 0.1
unpooling
0 0 0.5 0
1.3 0 0 0
0 0.4 0 0
0 0 0.1 0
4 days ago

Pooling

- • Encodes a degree of invariance with respect to translations
- • Reduces the size of the layers

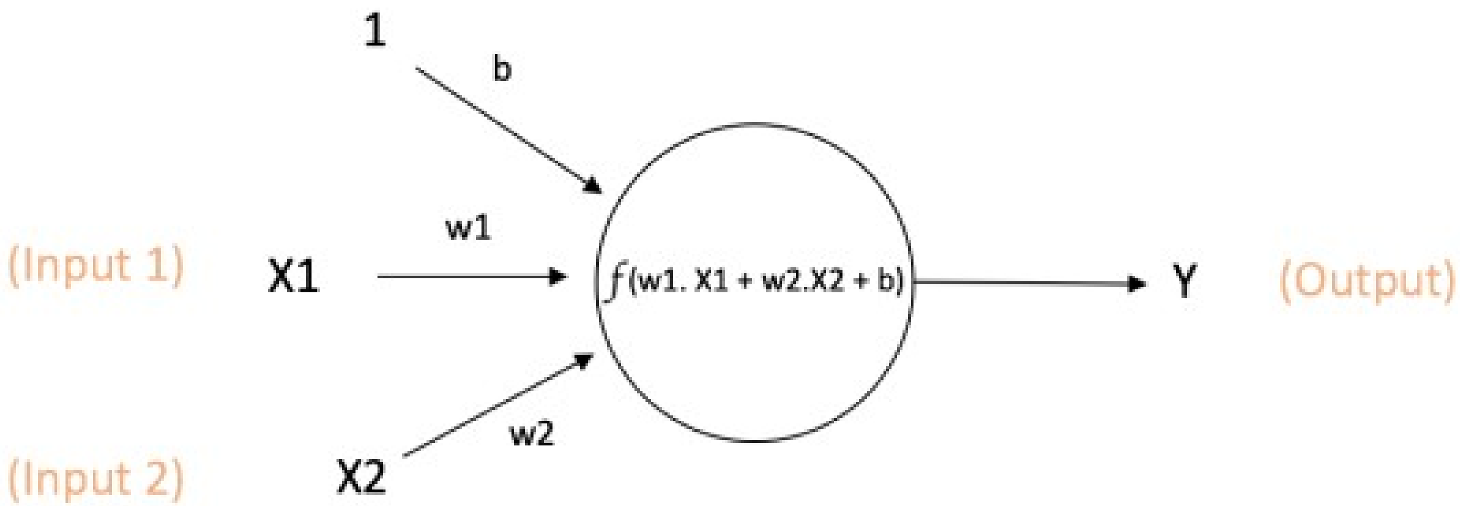
Dropout layer



(a) Standard Neural Net

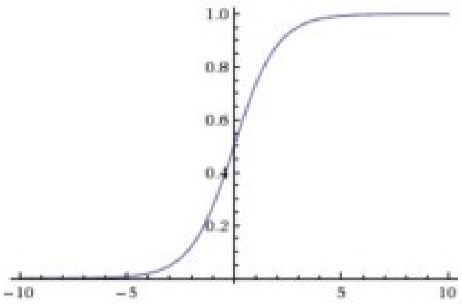
(b) After applying dropout.

Activation Function

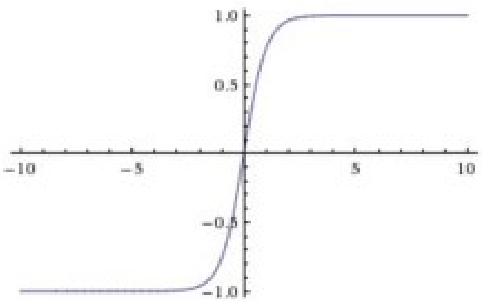


Output of neuron = Y= f(w1. X1 + w2.X2 + b)

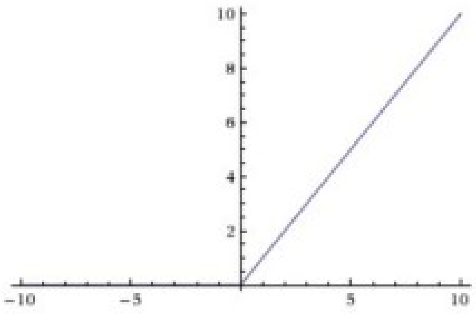
Types of Activation Function



Sigmoid



tanh

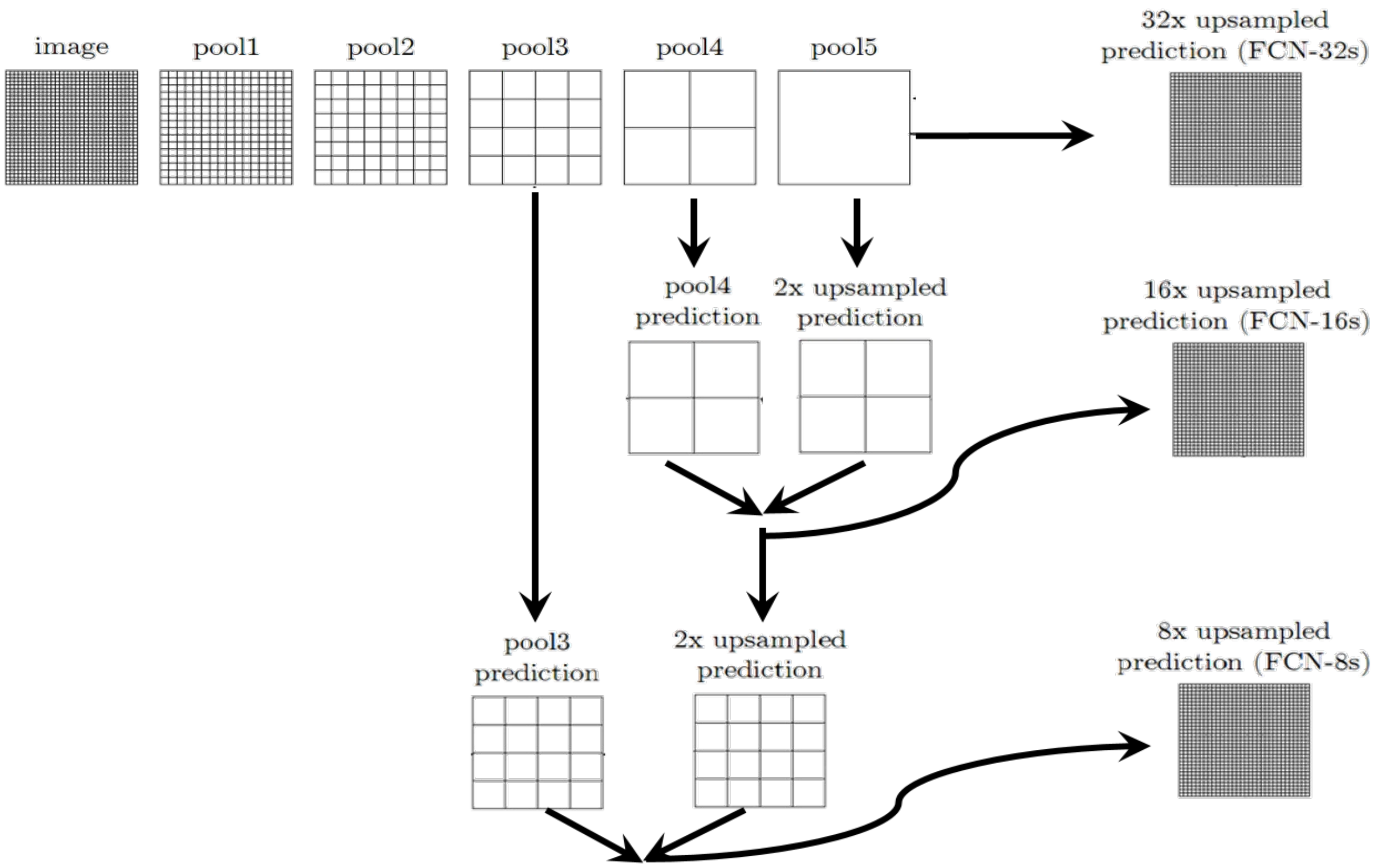


ReLU

Upsampling output

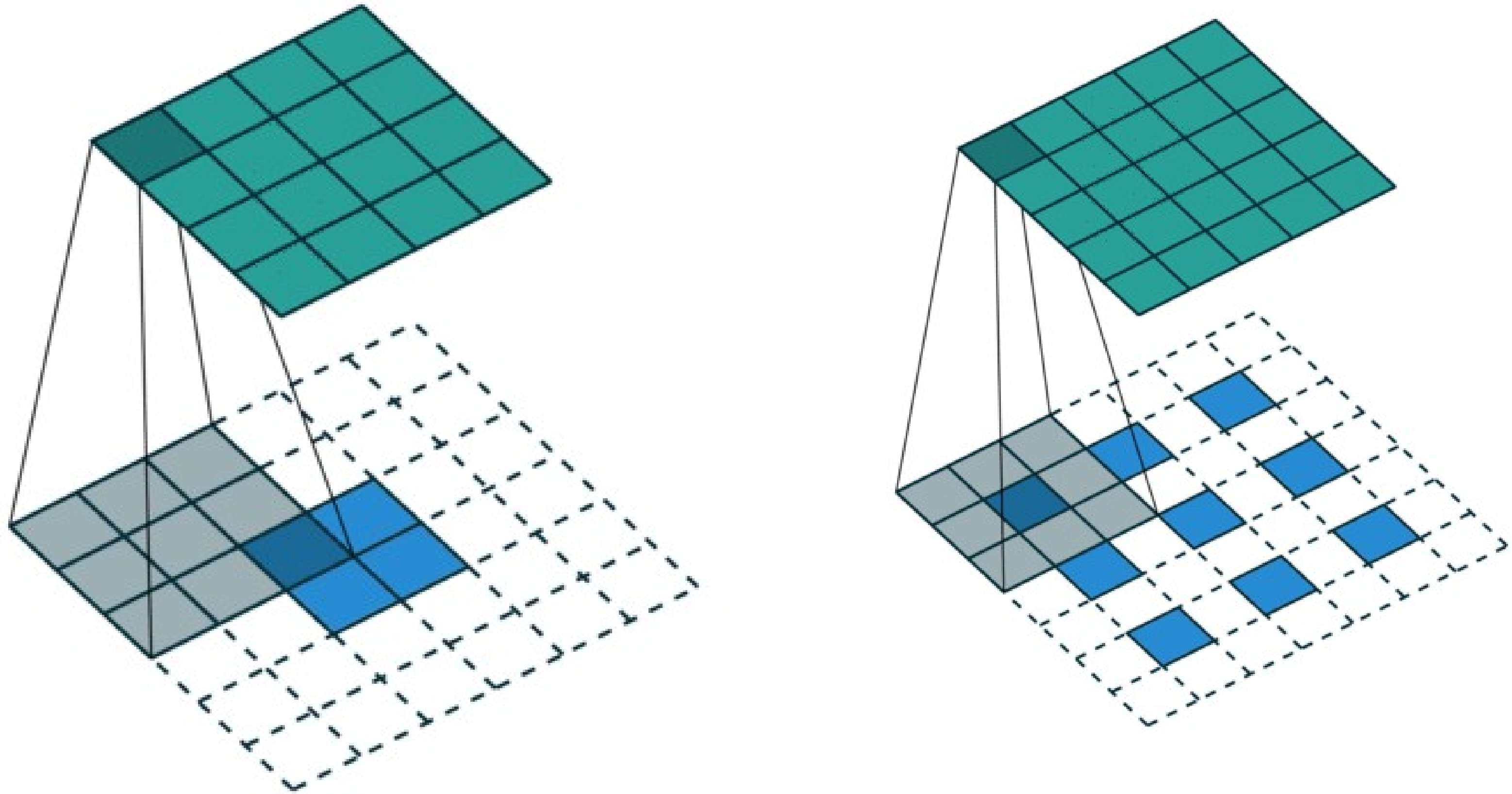
- Implements image enlargement.
- Method chosen: Deconvolution
- Up-sampling is performed in-network for end to end learning by back-propagation from the pixelwise loss

Upsample



https://meetshah1995.github.io/semantic-segmentation/deep-learning/pytorch/visdom/2017/06/01/semantic-segmentation-over-the-years.html

Deconvolution



Deconvolution

- Reversing the data-flow in convolution
- Deconvolutional layers associate a single input activation with multiple outputs

Results

- The FCN-8s architecture put forth achieved a 20% relative improvement to 62.2% mean IU on Pascal VOC 2012 dataset.

Results

	FCN AlexNet	FCN VGG16	FCN GoogleNet
Forward Time	50ms	210ms	59ms
Convolution Layers	8	16	22
No of Parameters	57M	134M	6M
Max Stride	32	32	32

Essential matrix

pixel accuracy: $\sum_i n_{ii} / \sum_i t_i$

mean accuraccy: $(1/n_{\text{cl}}) \sum_i n_{ii}/t_i$

mean IU: $(1/n_{\text{cl}}) \sum_i n_{ii} / \left(t_i + \sum_j n_{ji} - n_{ii}\right)$

frequency weighted IU:

$(\sum_k t_k)^{-1} \sum_i t_i n_{ii} / \left(t_i + \sum_j n_{ji} - n_{ii}\right)$

Comparison of Skip FCN

	Pixel accuracy	Mean IU
FCN 32s Fixed	83.0	45.4
FCN 32s	89.1	59.4
FCN 16s	90.0	62.4
FCN 8s	90.3	62.7

	Batch size	Momentum	Pixel accuracy	Mean accuracy
FCN Accumulation	20	.9	86	66.5
FCN Online	1	.9	89.3	76.2
FCN Heavy	1	.99	90.5	76.5

Conclusion

- (CNNs) methods have high accuracy.
- Stricter localization accuracy requirements
- Smaller objects ignored and classified as background

6. Reference

- https://research.googleblog.com/2018/03/semantic-image-segmentation-with.html
- https://wiki.tum.de/display/lfdv/Image+Semantic+Segmentation#ImageSemanticSegmentation-FullyConvolutionalNeuralNetworks
- http://cv-tricks.com/image-segmentation/transpose-convolution-in-tensorflow/
- https://arxiv.org/pdf/1609.07009.pdf

MOLTES GRÀCIES!!