Deep Semantic Segmentation

Mennatullah Siam
PhD Student at University of Alberta - Canada

Tutors:
Hager Radi
Mai
Mohammed Zahran
Omar Abdeltawab

EGYPT IndabaX

Agenda

- Introduction
- Fully Convolutional Networks
- FCN8s Architecture
- DeepLab Architecture
- Instance Segmentation [Mask R-CNN]
- Few-shot Segmentation
- Video Object Segmentation

Agenda

- Introduction
- Fully Convolutional Networks
- FCN8s Architecture
- DeepLab Architecture
- Instance Segmentation [Mask R-CNN]
- Few-shot Segmentation
- Video Object Segmentation

What is Classification?

CNN

Class Scores:
Cat: 0.9
Dog:0.05
Car:0.01

What is Classification?

CNN

Class Scores:
Cat: 0.9
Dog:0.05
Car:0.01

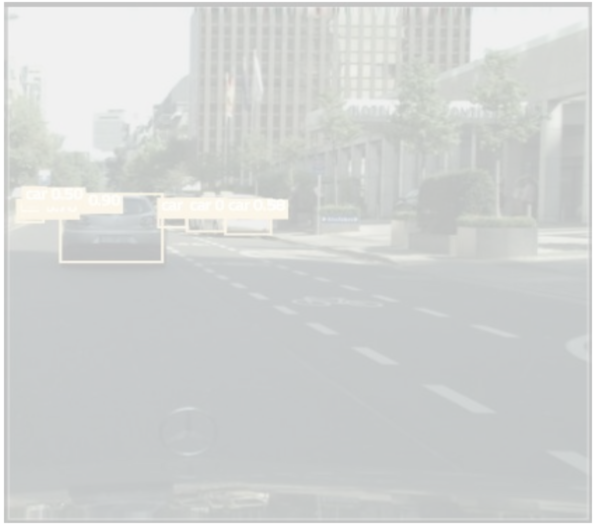
What Other Computer Vision tasks?

Semantic Segmentation

Road, Cars,
Trees, Sky

No objects, just pixels

Object Detection



Instance Segmentation

Multiple Object

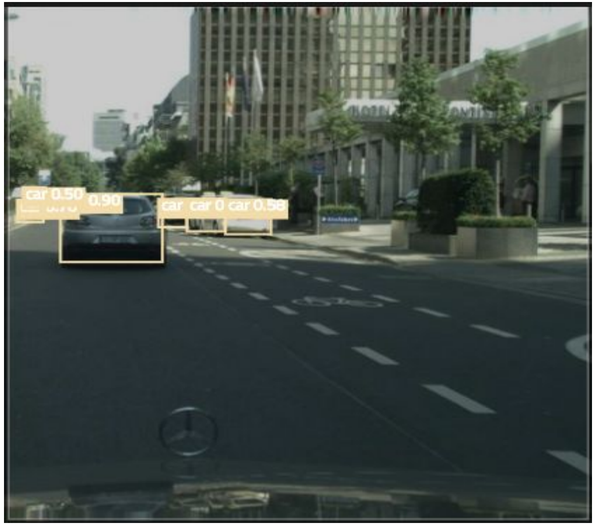
What Other Computer Vision tasks?

Semantic Segmentation

Road, Cars,
Trees, Sky

Pixel-wise classification

Object Detection



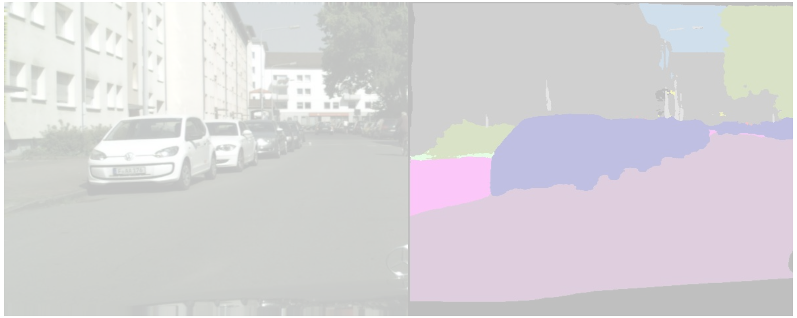
Instance Segmentation

Multiple Objects

What is semantic segmentation?

- One way to classify every pixel is to have a patchwise classification network.

- Another idea would be to get rid of the fully connected layers and instead use a **fully convolutional network** [1].

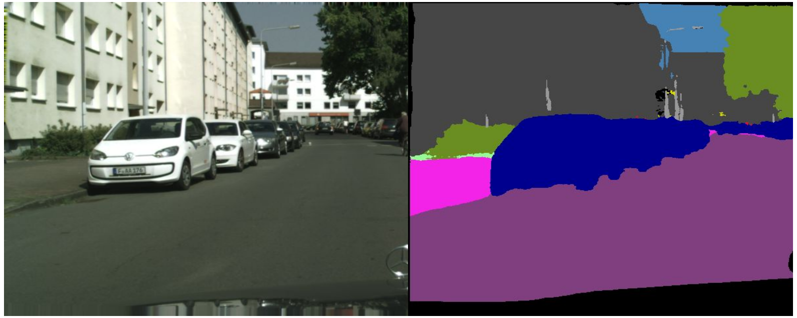


[1] Long, Jonathan, Evan Shelhamer, and Trevor Darrell. "Fully convolutional networks for semantic segmentation." Proceedings of the IEEE conference on computer vision and pattern recognition. 2015.

What is semantic segmentation?

- One way to classify every pixel is to have a patchwise classification network.

- Another idea would be to get rid of the fully connected layers and instead use a **fully convolutional network** [1].

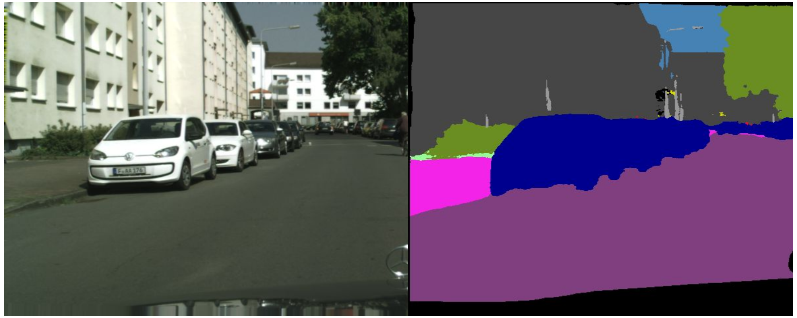


[1] Long, Jonathan, Evan Shelhamer, and Trevor Darrell. "Fully convolutional networks for semantic segmentation." Proceedings of the IEEE conference on computer vision and pattern recognition. 2015.

What is semantic segmentation?

- One way to classify every pixel is to have a patchwise classification network.

- Another idea would be to get rid of the fully connected layers and instead use a **fully convolutional network** [1].

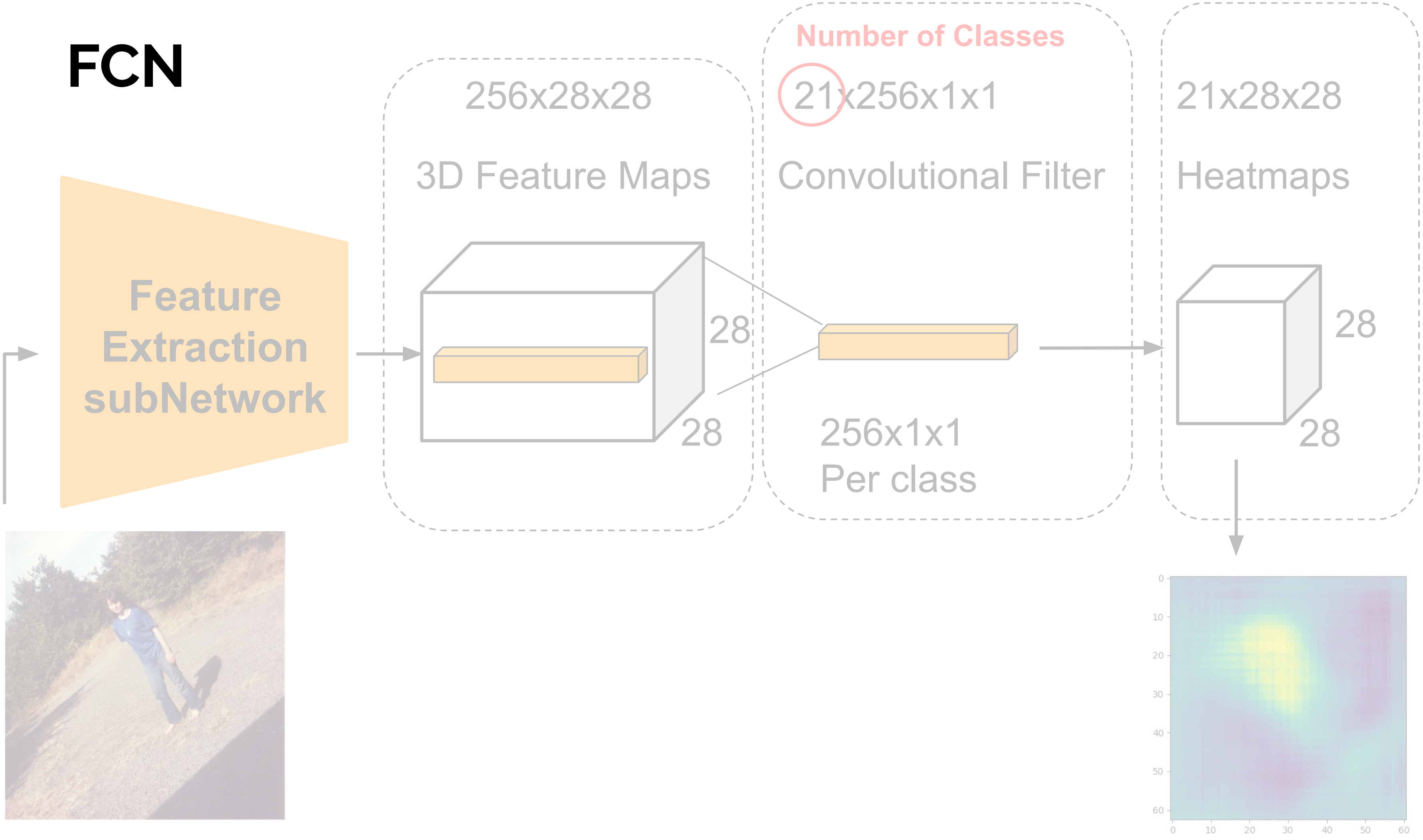


[1] Long, Jonathan, Evan Shelhamer, and Trevor Darrell. "Fully convolutional networks for semantic segmentation." Proceedings of the IEEE conference on computer vision and pattern recognition. 2015.

Agenda

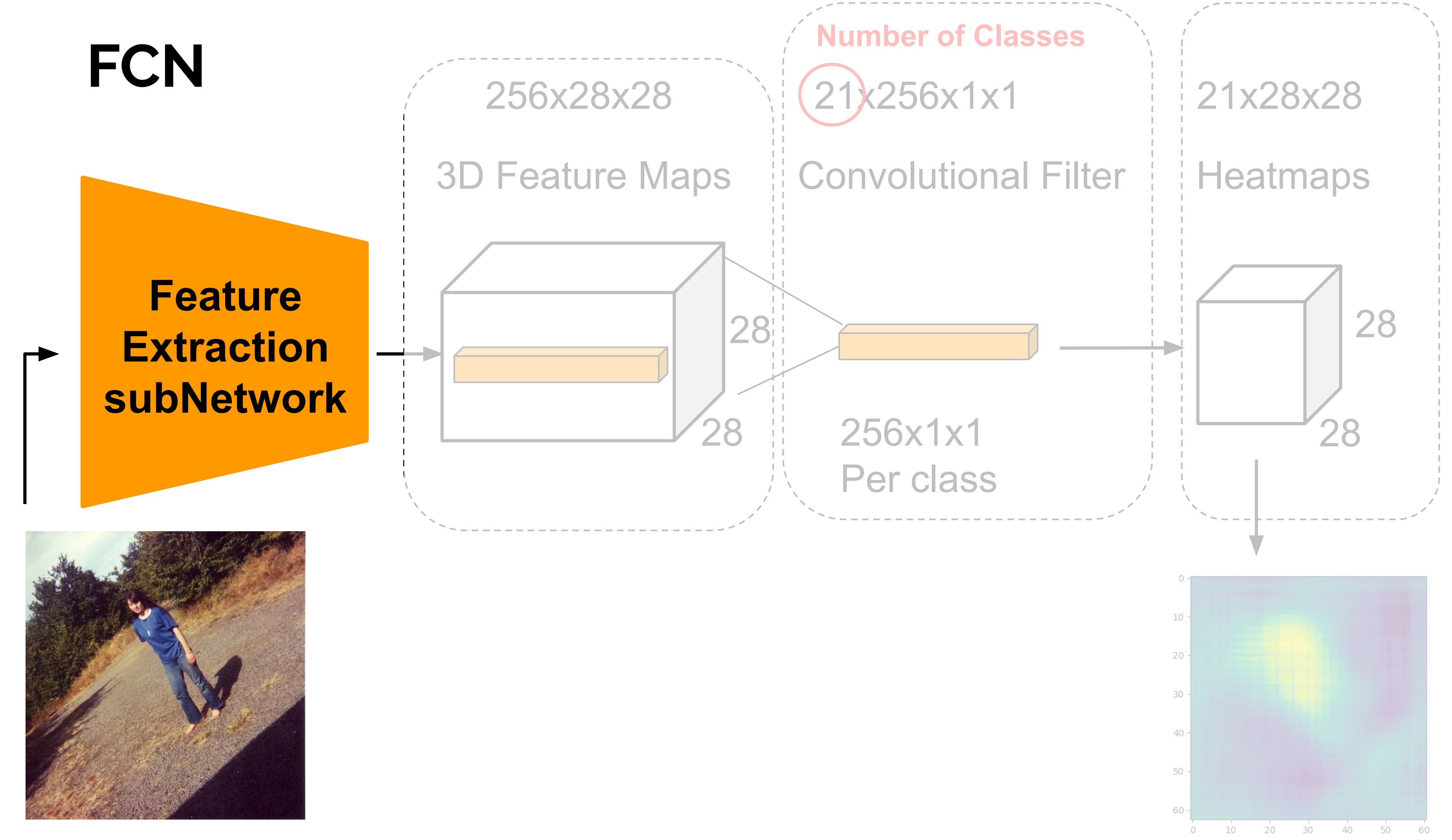
- Introduction
- Fully Convolutional Networks
- FCN8s Architecture
- DeepLab Architecture
- Instance Segmentation [Mask R-CNN]
- Few-shot Segmentation
- Video Object Segmentation

FCN



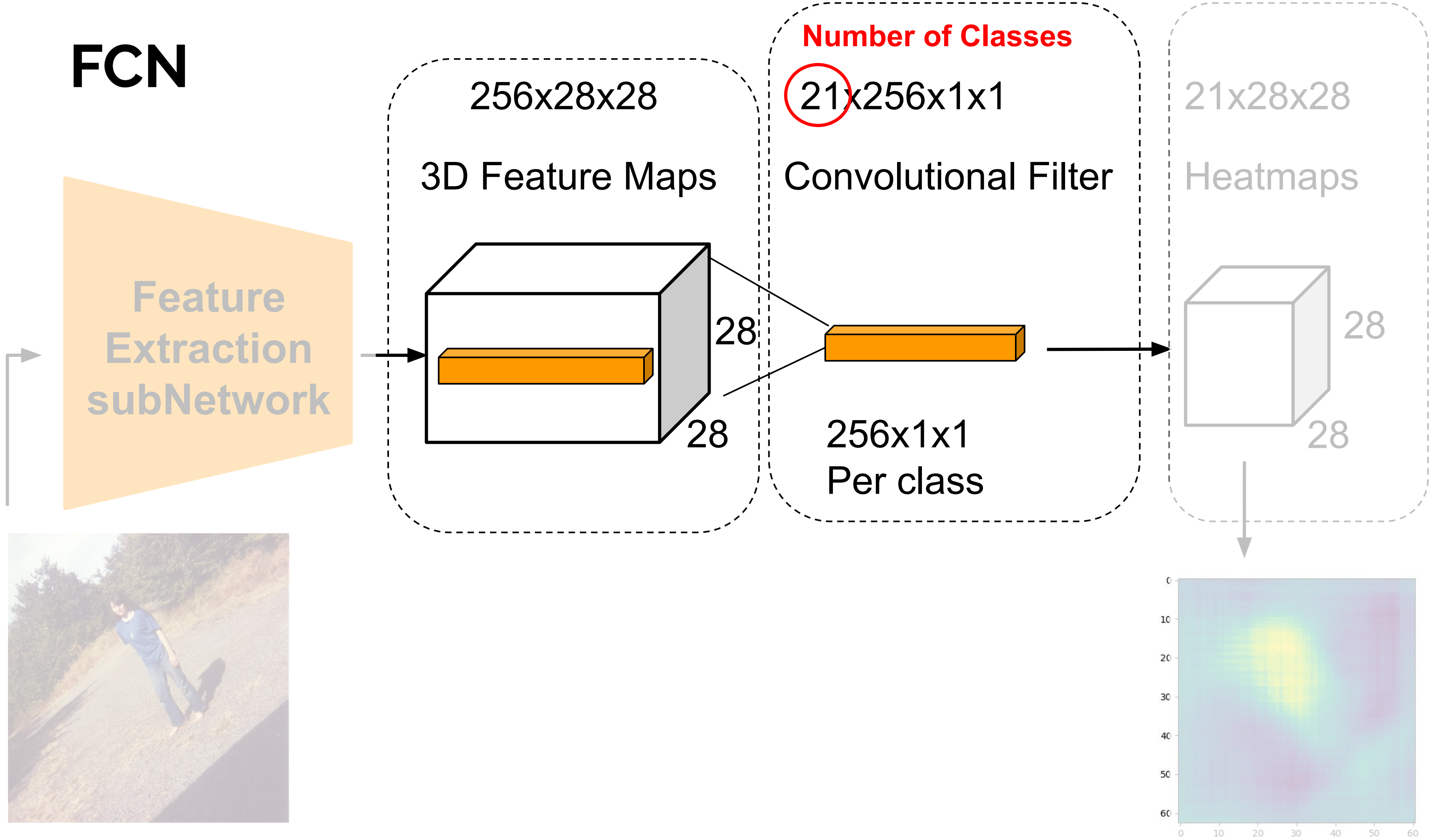
Number of Classes
256x28x28
3D Feature Maps
21x256x1x1
Convolutional Filter
21x28x28
Heatmaps
Feature Extraction subNetwork
28
28
256x1x1
Per class
28
28
0
10
20
30
40
50
60
0
10
20
30
40
50
60

FCN



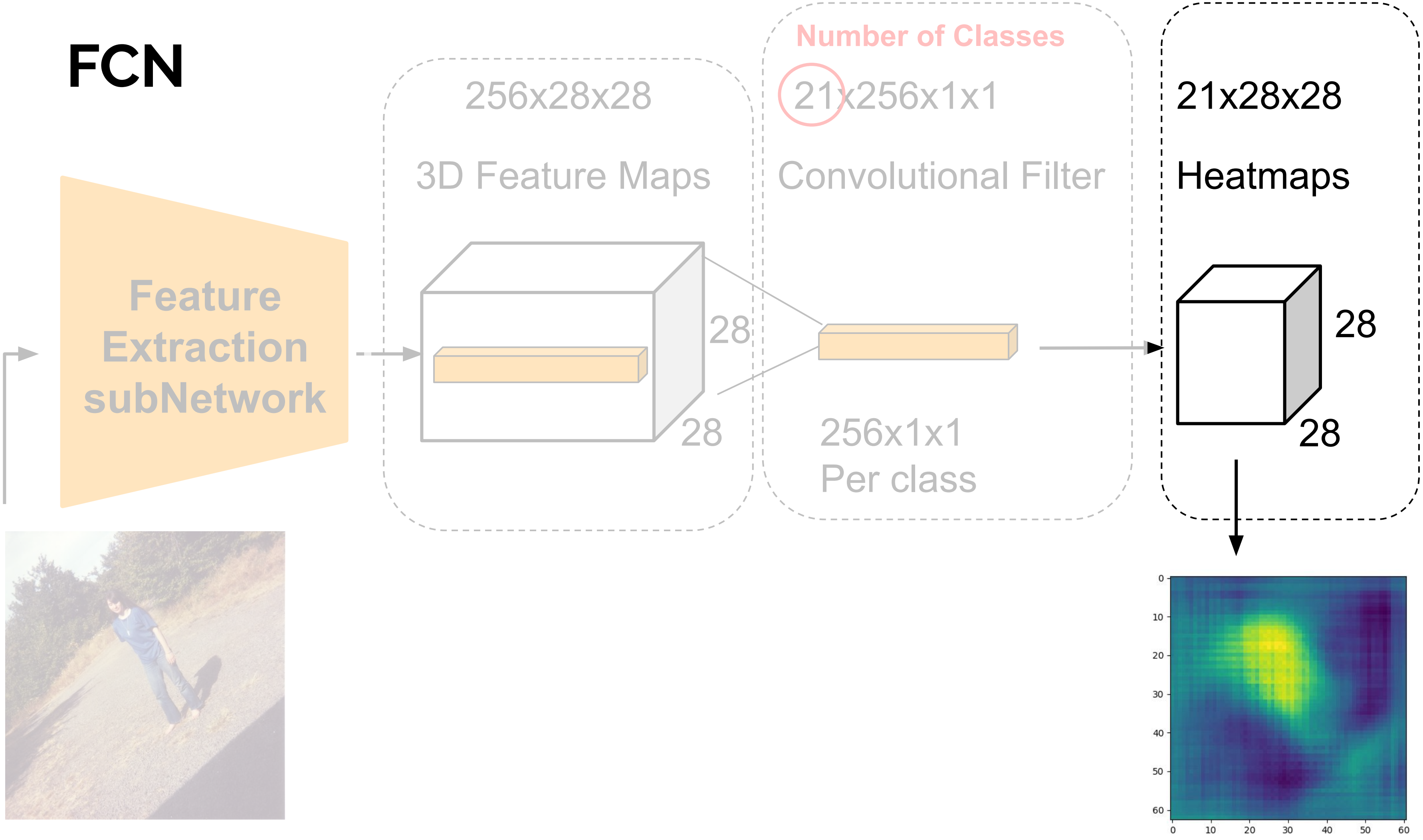
Feature Extraction subNetwork
256x28x28
3D Feature Maps
28
28
Number of Classes
21x256x1x1
Convolutional Filter
256x1x1
Per class
21x28x28
Heatmaps
28
28
0
10
20
30
40
50
60
0
10
20
30
40
50
60

FCN

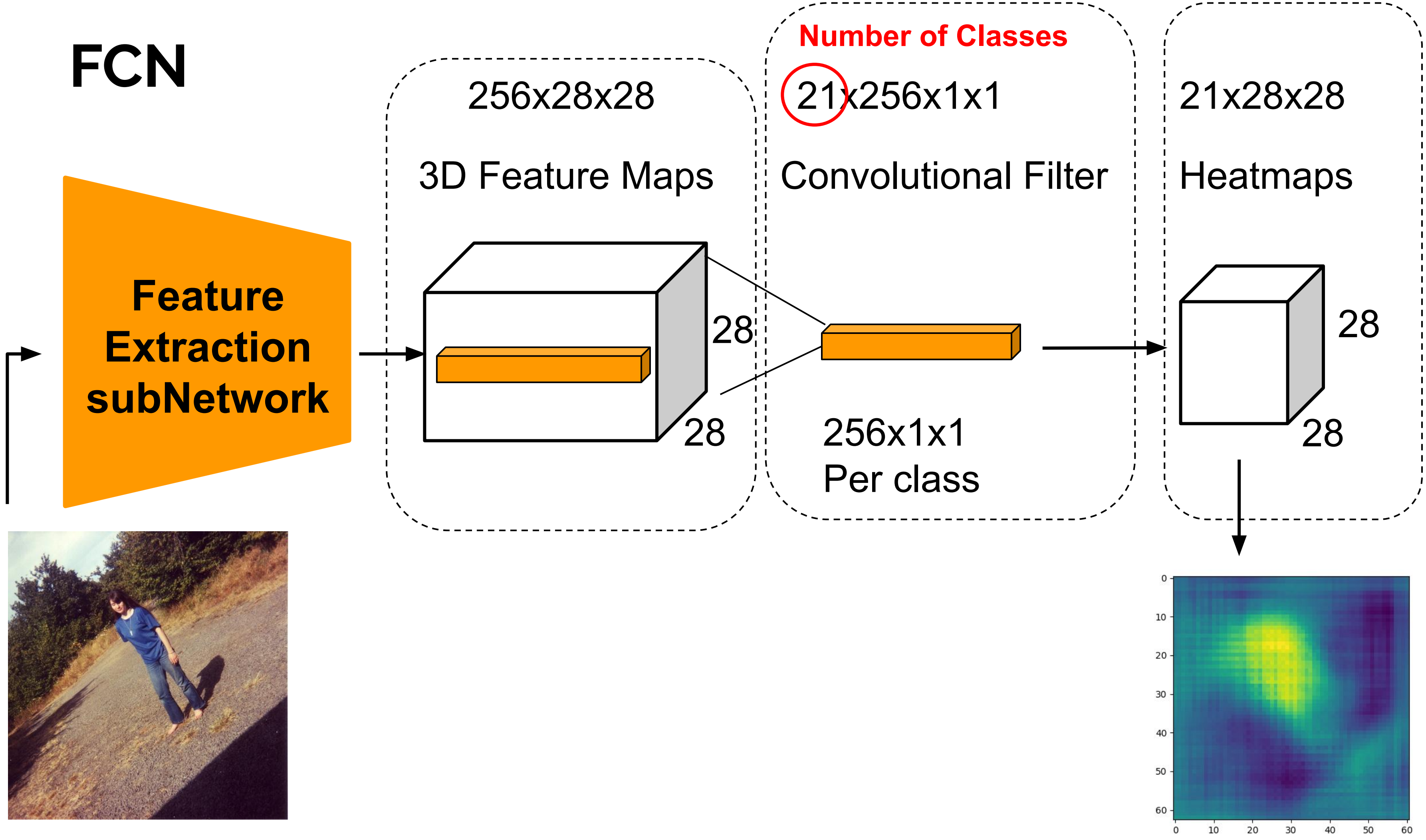


256x28x28
3D Feature Maps
Number of Classes
21x256x1x1
Convolutional Filter
21x28x28
Heatmaps
Feature Extraction subNetwork
28
28
256x1x1
Per class
28
28

FCN

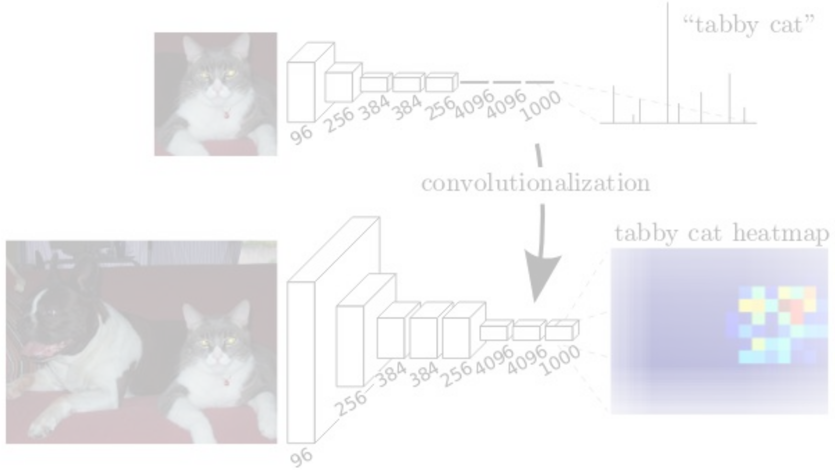

Number of Classes
256x28x28
3D Feature Maps
21x256x1x1
Convolutional Filter
21x28x28
Heatmaps
Feature Extraction subNetwork
28
28
256x1x1
Per class
28
28

FCN


Number of Classes
256x28x28
3D Feature Maps
21x256x1x1
Convolutional Filter
21x28x28
Heatmaps
Feature Extraction subNetwork
28
28
256x1x1
Per class
28
28
0
10
20
30
40
50
60
0
10
20
30
40
50
60

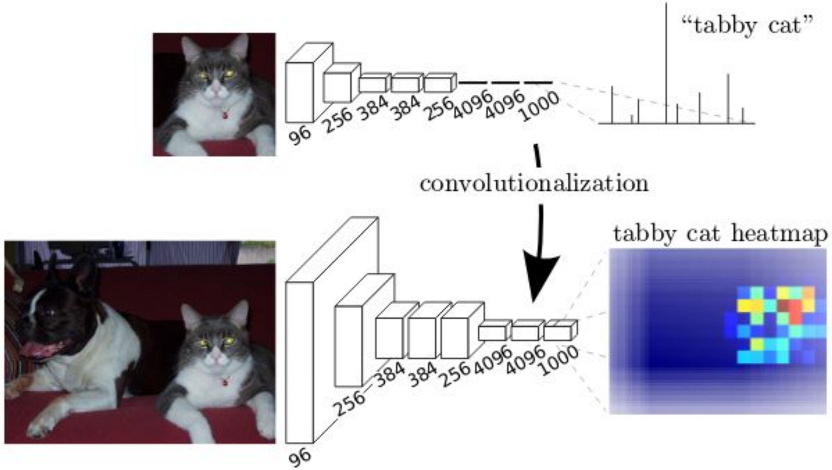
Fully Convolutional Networks

- **Fully connected** layers are equivalent to **1x1 convolutions**.
- FC W: **256x21.**
- 1x1 Conv W: **21x256x1x1**.
- Output is heatmap from your network.
- What other uses for 1x1 convolution?



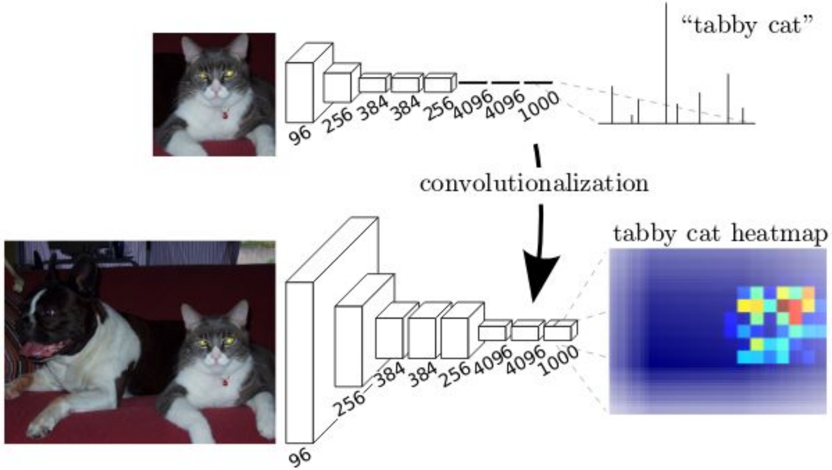
Fully Convolutional Networks

- **Fully connected** layers are equivalent to **1x1 convolutions.**
- FC W: **256x21.**
- 1x1 Conv W: **21x256x1x1**.
- Output is heatmap from your network.
- What other uses for 1x1 convolution?



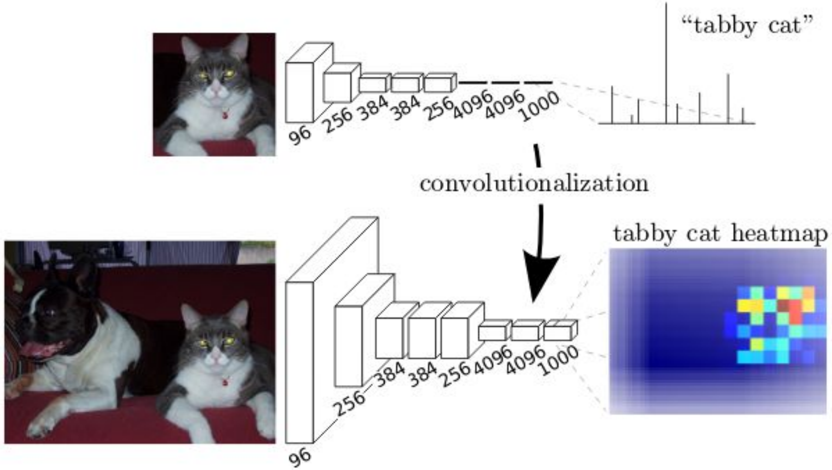
Fully Convolutional Networks

- **Fully connected** layers are equivalent to **1x1 convolutions**.
- FC W: **256x21.**
- 1x1 Conv W: **21x256x1x1**.
- Output is heatmap from your network.
- What other uses for 1x1 convolution?



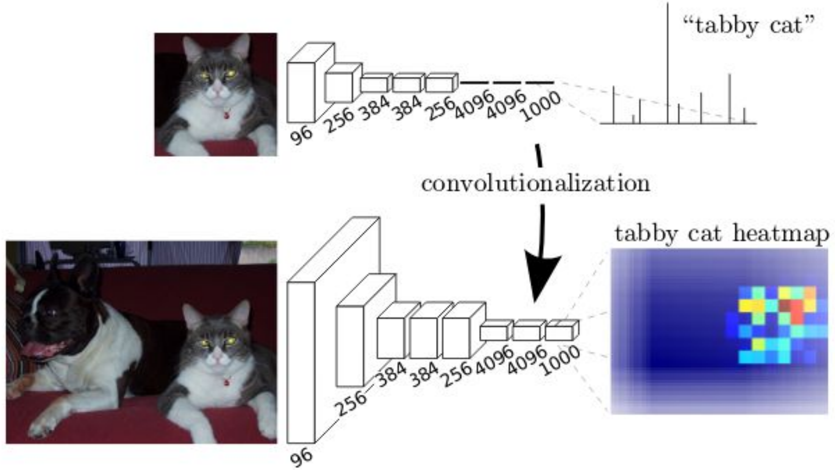
Fully Convolutional Networks

- **Fully connected** layers are equivalent to **1x1 convolutions**.
- FC W: **256x21.**
- 1x1 Conv W: **21x256x1x1**.
- Output is heatmap from your network.
- What other uses for 1x1 convolution?



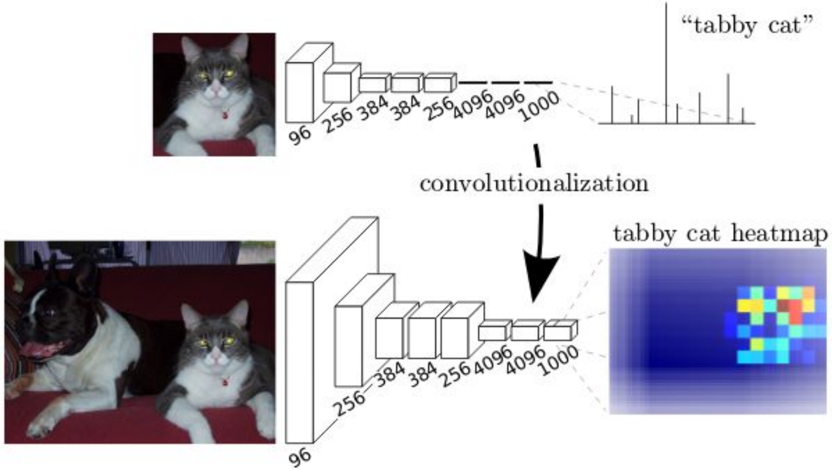
Fully Convolutional Networks

- **Fully connected** layers are equivalent to **1x1 convolutions**.
- FC W: **256x21.**
- 1x1 Conv W: **21x256x1x1**.
- Output is heatmap from your network.
- What other uses for 1x1 convolution?



Fully Convolutional Networks

- **Fully connected** layers are equivalent to **1x1 convolutions**.
- FC W: **256x21**.
- 1x1 Conv W: **21x256x1x1**.
- Output is heatmap from your network.
- What other uses for 1x1 convolution?



Loss Function

- Pixel-wise Cross Entropy

$$L = -\frac{1}{N}\sum_i y_i \log p_i$$

Label	Higher Loss Output 1	Lower Loss Output 2
y(dog) = 1	p(dog) = 0.4	p(dog) = 0.98
y(fox) = 0	p(fox) = 0.3	p(fox) = 0.01
y(horse) = 0	p(horse) = 0.05	p(horse) = 0
y(eagle) = 0	p(eagle) = 0.05	p(eagle) = 0
y(squirrel) = 0	p(squirrel) = 0.2	p(squirrel) = 0.01

- Weighted Cross Entropy [1] (Higher weight to less occurring classes)

[1] Paszke, Adam, et al. "Enet: A deep neural network architecture for real-time semantic segmentation." *arXiv preprint arXiv:1606.02147* (2016).

Loss Function

- Pixel-wise Cross Entropy

$$L = -\frac{1}{N}\sum_i y_i \log p_i$$

Label	Higher Loss Output 1	Lower Loss Output 2
y(dog) = 1	p(dog) = 0.4	p(dog) = 0.98
y(fox) = 0	p(fox) = 0.3	p(fox) = 0.01
y(horse) = 0	p(horse) = 0.05	p(horse) = 0
y(eagle) = 0	p(eagle) = 0.05	p(eagle) = 0
y(squirrel) = 0	p(squirrel) = 0.2	p(squirrel) = 0.01

- Weighted Cross Entropy [1] (Higher weight to less occurring classes)

[1] Paszke, Adam, et al. "Enet: A deep neural network architecture for real-time semantic segmentation." *arXiv preprint arXiv:1606.02147* (2016).

Loss Function

- Pixel-wise Cross Entropy

$$L = -\frac{1}{N} \sum_i y_i \log p_i$$

Label	Higher Loss Output 1	Lower Loss Output 2
y(dog) = 1	p(dog) = 0.4	p(dog) = 0.98
y(fox) = 0	p(fox) = 0.3	p(fox) = 0.01
y(horse) = 0	p(horse) = 0.05	p(horse) = 0
y(eagle) = 0	p(eagle) = 0.05	p(eagle) = 0
y(squirrel) = 0	p(squirrel) = 0.2	p(squirrel) = 0.01

- Weighted Cross Entropy [1] (Higher weight to less occurring classes)

[1] Paszke, Adam, et al. "Enet: A deep neural network architecture for real-time semantic segmentation." *arXiv preprint arXiv:1606.02147* (2016).

Loss Function

- Pixel-wise Cross Entropy
- Weighted Cross Entropy (Higher weight to less occurring classes)

- Boot-strapped Cross Entropy [1]

Hardest Pixels

$$L = -\frac{\sum_i^N \sum_j^K 1\{y_i = j \text{ and } p_{ij} < t\} \log p_{ij}}{\sum_i^N \sum_j^K 1\{y_i = j \text{ and } p_{ij} < t\}}$$

[1] Wu, Zifeng, Chunhua Shen, and Anton van den Hengel. "Bridging category-level and instance-level semantic image segmentation." *arXiv preprint arXiv:1605.06885* (2016).

Practical 1.1: Build your first FCN

https://bit.ly/2DNmoYm

Agenda

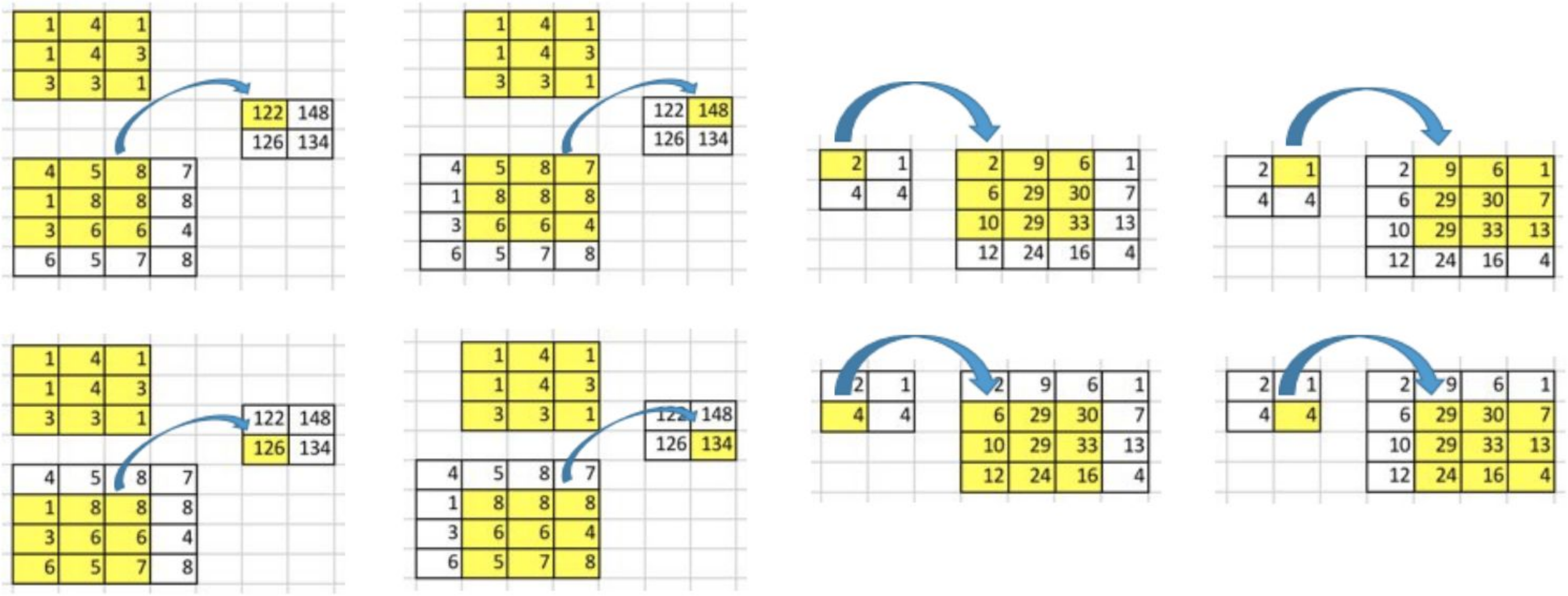
- Introduction
- Fully Convolutional Networks
- FCN8s Architecture
- DeepLab Architecture
- Instance Segmentation [Mask R-CNN]
- Few-shot Segmentation
- Video Object Segmentation

Upsampling Within Network

- The output heatmaps as you saw is a downsampled version due to multiple pooling layers (5 pooling layers in VGG-16).
- Upsample using bilinear interpolation
- A better way is to learn the upsampling within the network using a layer called **Transposed Convolution[1]** (deconvolution or back-strided convolution)

[1] Dumoulin, Vincent, and Francesco Visin. "A guide to convolution arithmetic for deep learning." arXiv preprint arXiv:1603.07285 (2016).

Transposed Convolution



Convolution

Transposed Convolution

Figure from : https://towardsdatascience.com/up-sampling-with-transposed-convolution-9ae4f2df52d0

Transposed Convolution

- Convolution → Matrix Multiplication

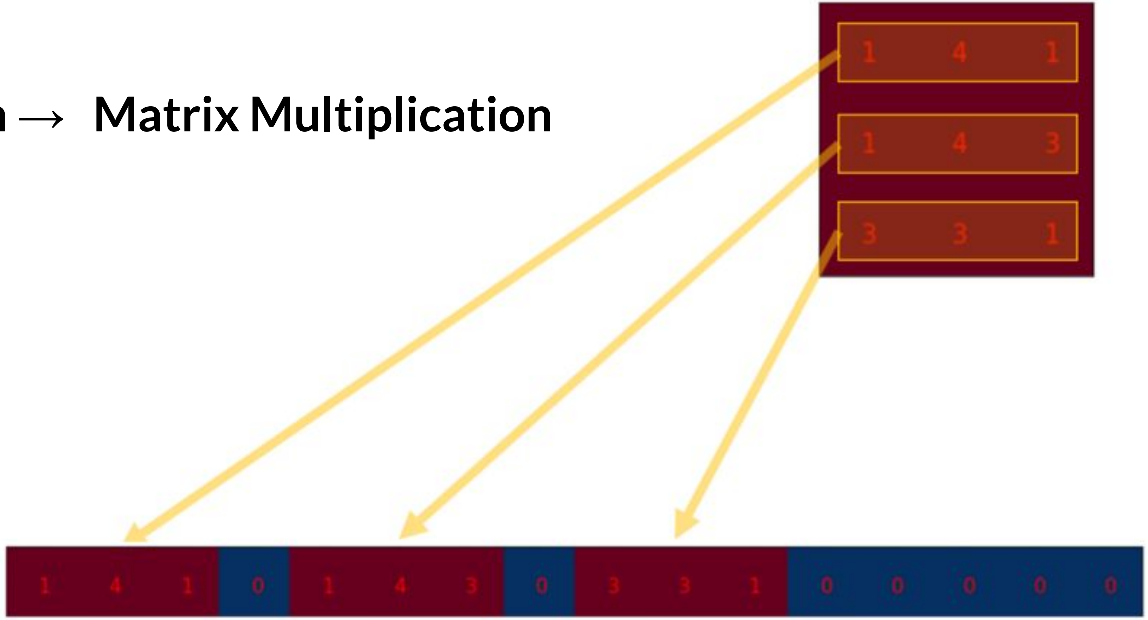
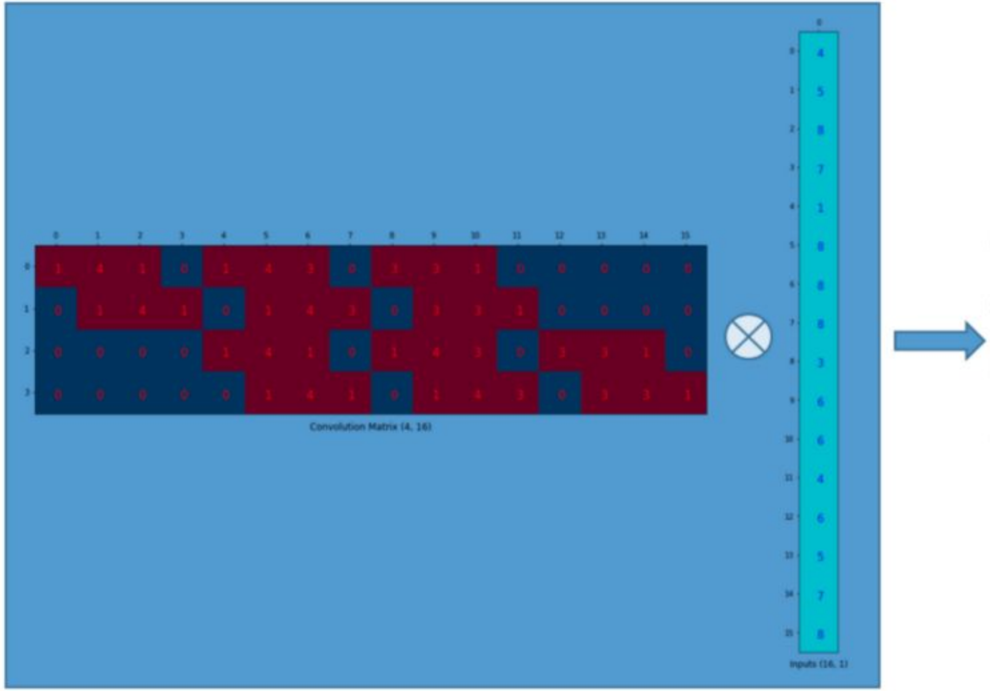
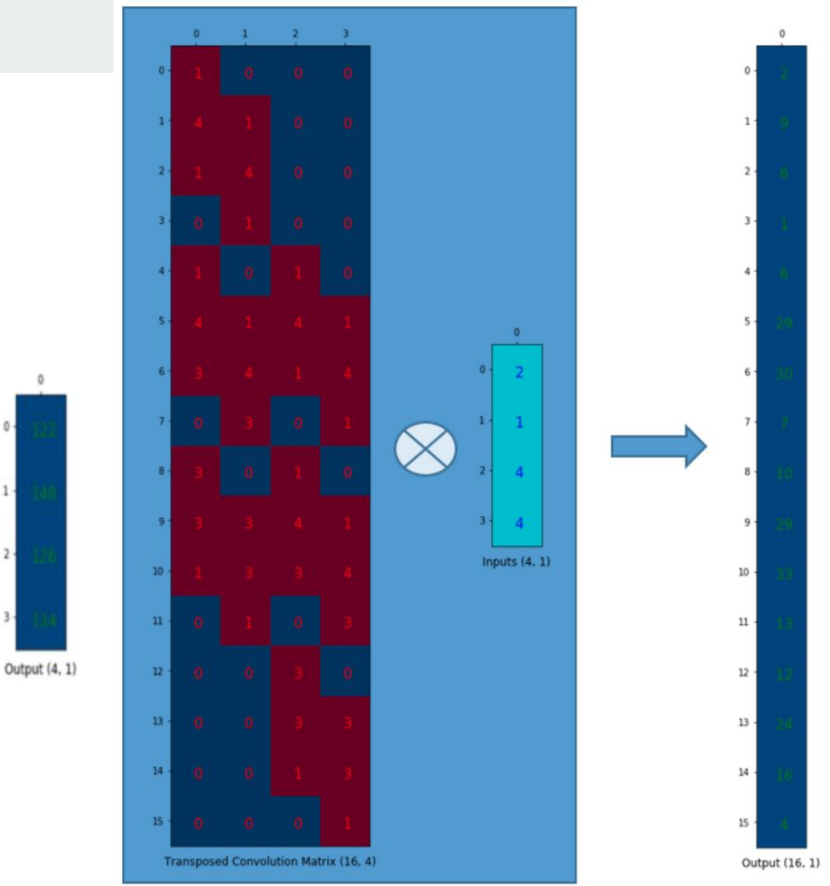


Figure from :
https://towardsdatascience.com/up-sampling-with-transposed-convolution-9ae4f2df52d0

Transposed Convolution





Convolution → Matrix Multiplication 4x16, and flatten the input 4x4 into 16x1

Figure from :

https://towardsdatascience.com/up-sampling-with-transposed-convolution-9ae4f2df52d0

Transposed Convolution

- Why is it considered the backpropagation of convolution?
- Let's first visualize Conv.

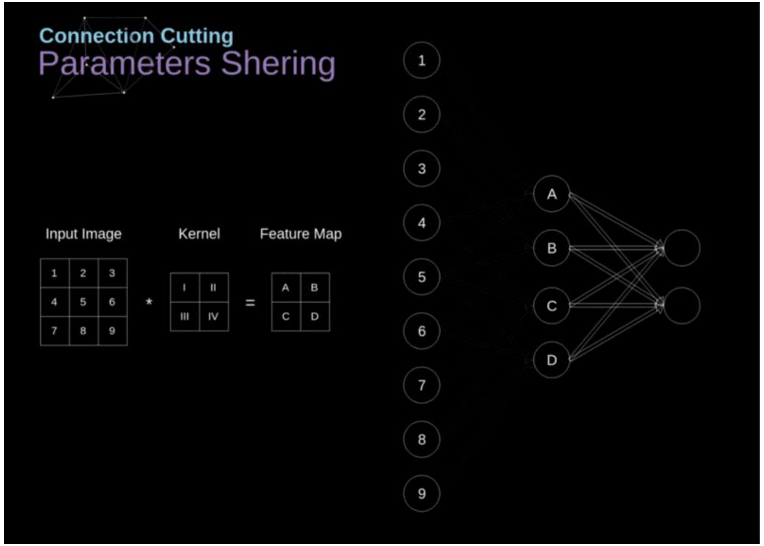


Figure from https://towardsdatascience.com/gentle-dive-into-math-behind-convolutional-neural-networks-79a07dd44cf9

Transposed Convolution

- Why is it considered the backpropagation of convolution?
- Backward Pass is still performing Convolution.

$$\frac{\partial J}{\partial w_i} = \frac{\partial J}{\partial y}\frac{\partial y}{\partial w_i}$$

$$\sum_{n=1}^{x-|w|+1} \frac{\partial J}{\partial y_n}\frac{\partial y_n}{\partial w_i}$$

$$\delta^y * x$$

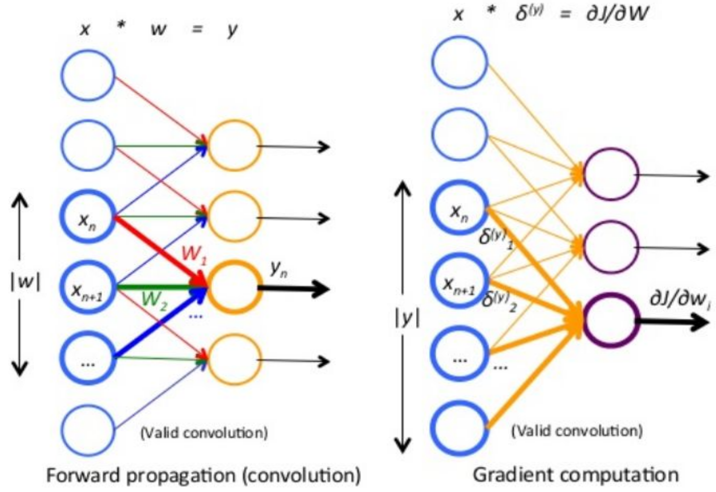
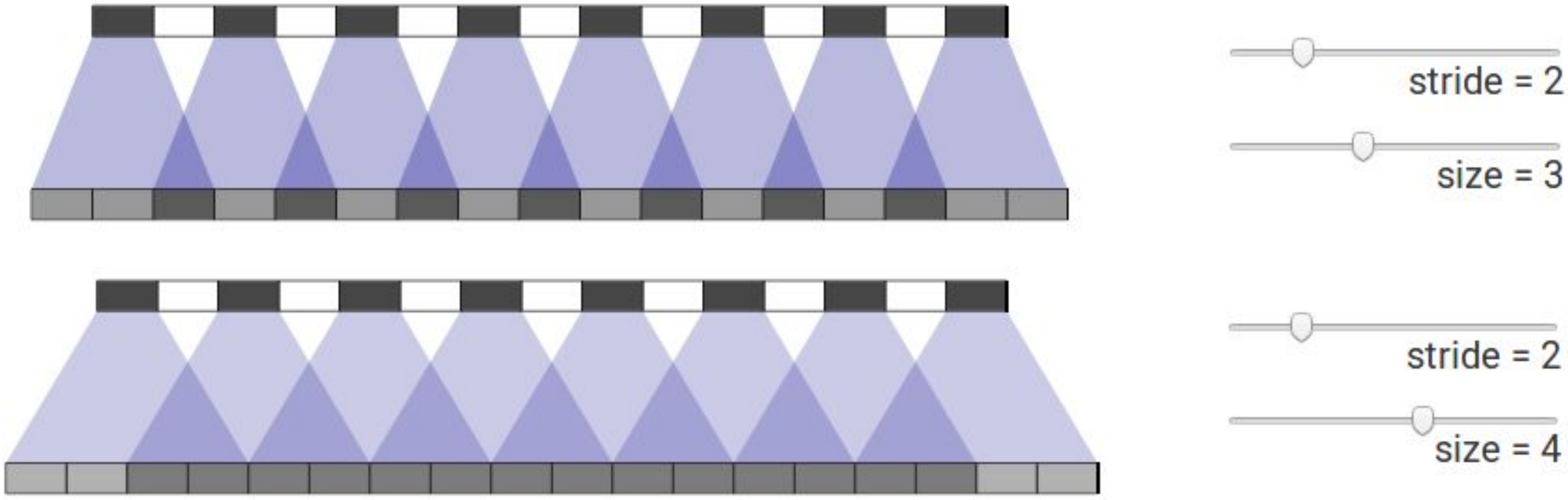


Figure from: https://www.slideshare.net/kuwajima/cnnbp

Checkerboard Effect

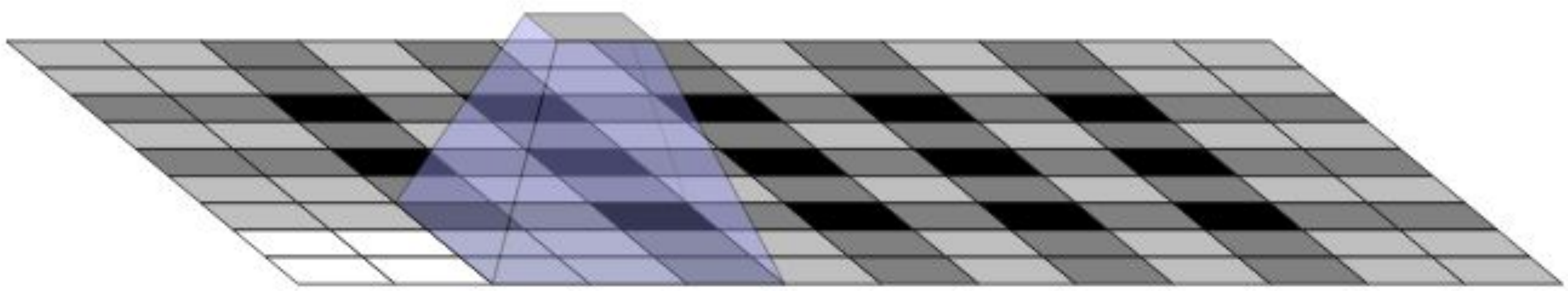
- Uneven Overlap (Kernel size ! divisible by stride) - 1D Case



Figures from : https://distill.pub/2016/deconv-checkerboard/

Checkerboard Effect

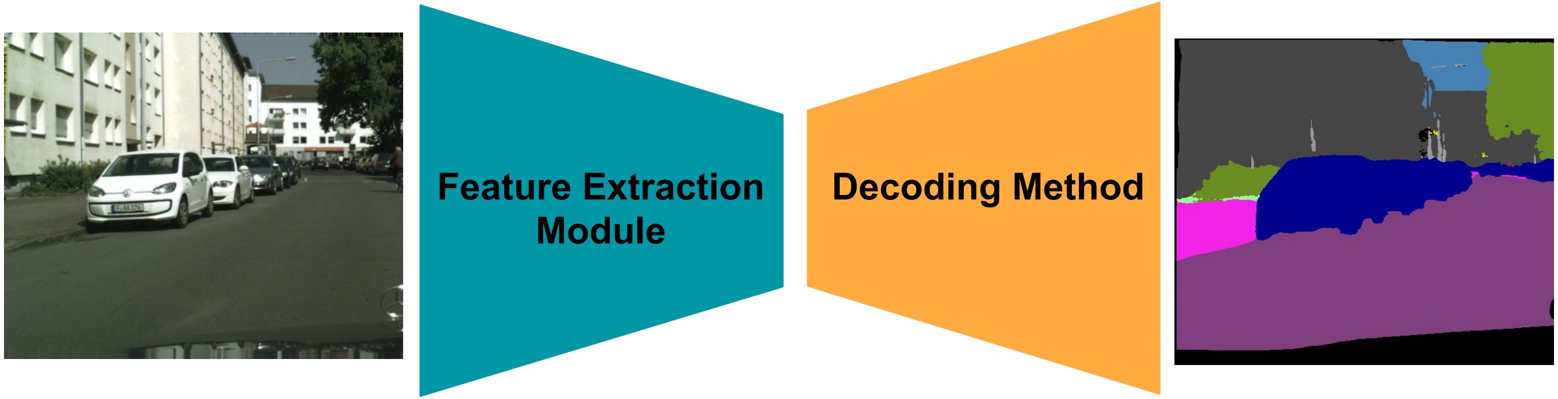
- Uneven Overlap (Kernel size ! divisible by stride) - 2D Case



Figures from : https://distill.pub/2016/deconv-checkerboard/

Encoder-Decoder

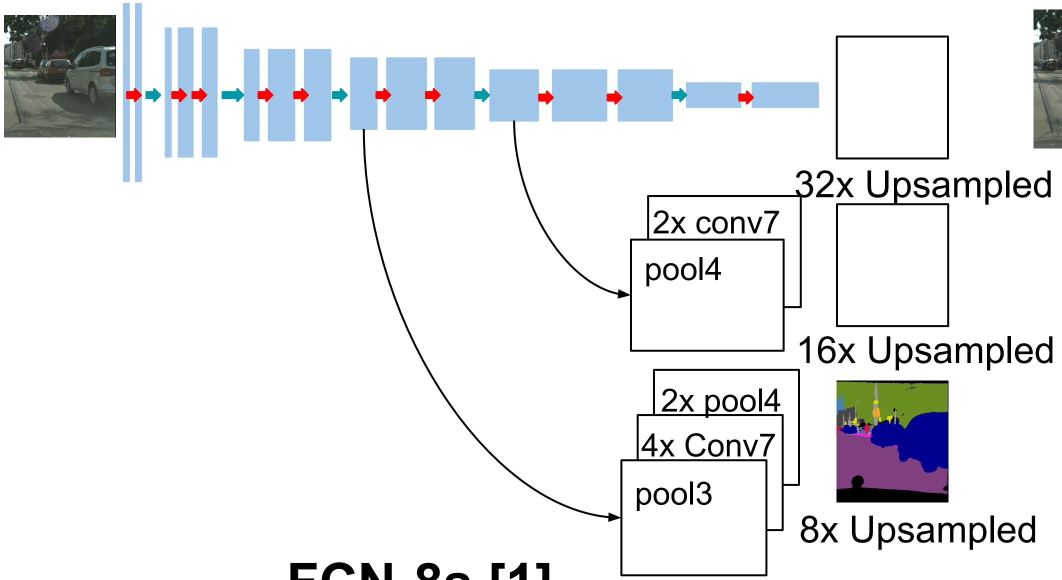
- Separating **Encoding** (Feature Extraction) from **Decoding** (Upsampling-Projecting to labels) method help analyze effect of different design choices



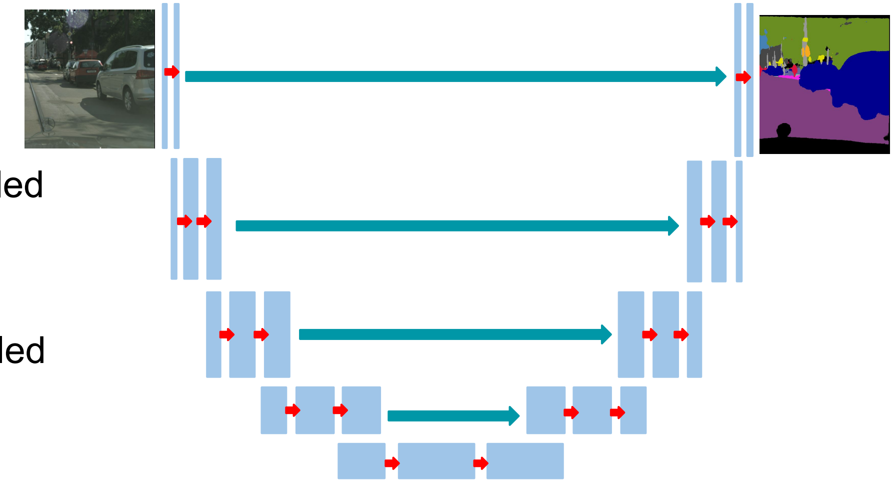
Skip Connections

- Pooling layers:
  - Increases the receptive field which is important for better segmentation.
  - It hurts the resolution which can degrade the accuracy.
- One way is to use skip Connections either in the:
  - Label Space **(FCN8s) -** Computationally efficient
  - Feature Space **(UNet) -** Better accuracy.

Fully Convolutional Networks



FCN-8s [1]



U-Net [2]

[1] Long, Jonathan, Evan Shelhamer, and Trevor Darrell. "Fully convolutional networks for semantic segmentation." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015.
[2] Ronneberger, Olaf, Philipp Fischer, and Thomas Brox. "U-net: Convolutional networks for biomedical image segmentation." *International Conference on Medical image computing and computer-assisted intervention*. Springer, Cham, 2015.

Practical 1.2: Skip Connections and Transposed Convolution

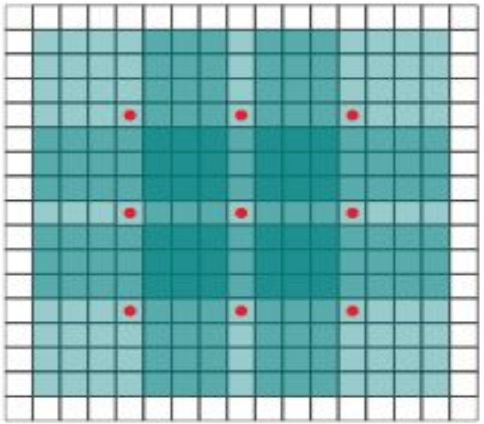
https://bit.ly/2DNmoYm

Agenda

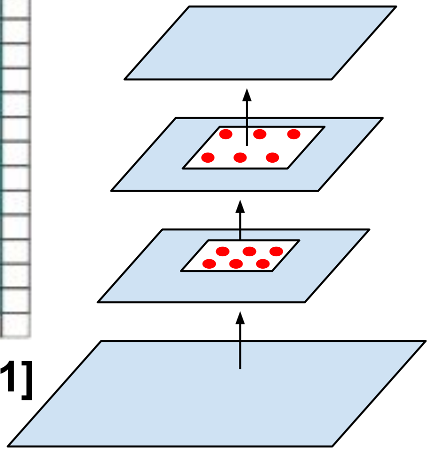
- Introduction
- Fully Convolutional Networks
- FCN8s Architecture
- DeepLab Architecture
- Instance Segmentation [Mask R-CNN]
- Few-shot Segmentation
- Video Object Segmentation

Dilated Convolution

- How about Increasing receptive field without pooling.
- Perform Convolution with holes (Atrous Convolution - Dilated Convolution) [1]



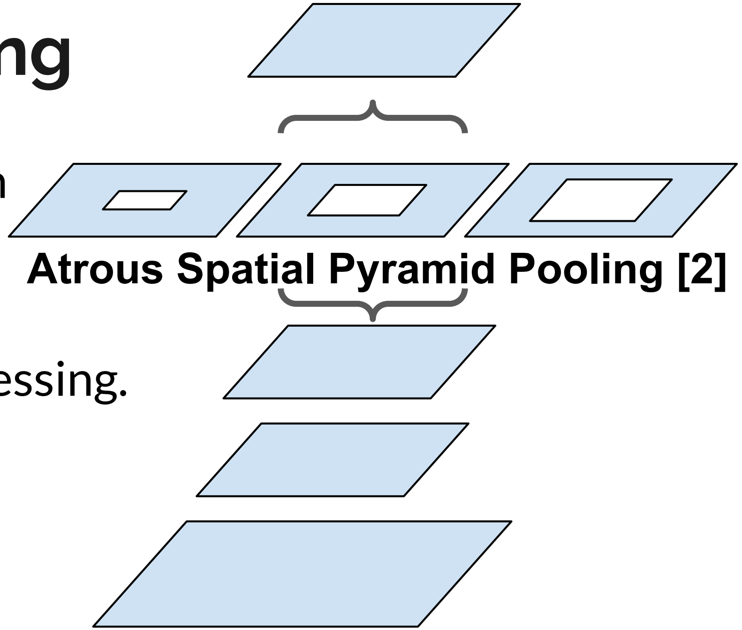
Atrous Convolution [1]



[1] Yu, Fisher, and Vladlen Koltun. "Multi-scale context aggregation by dilated convolutions." *arXiv preprint arXiv:1511.07122* (2015).

Atrous Spatial Pyramid Pooling

- Multiple Dilated Convolution in parallel with different dilation factor.
- DeepLab architecture.
- Used conditional random fields as post processing.



[1] Chen, Liang-Chieh, et al. "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs." *IEEE transactions on pattern analysis and machine intelligence* 40.4 (2018): 834-848.

Practical 1.3: Deeplab

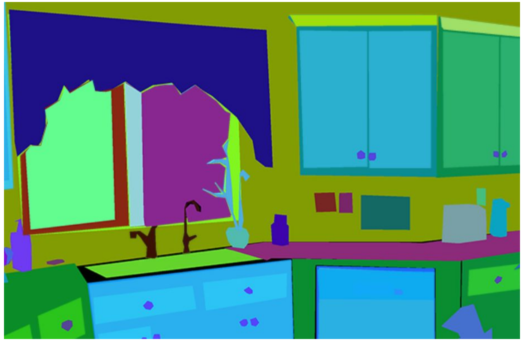
https://bit.ly/2DNmoYm

Guide for Training Neural Networks

- The invisible sword ^_^ : https://karpathy.github.io/2019/04/25/recipe/

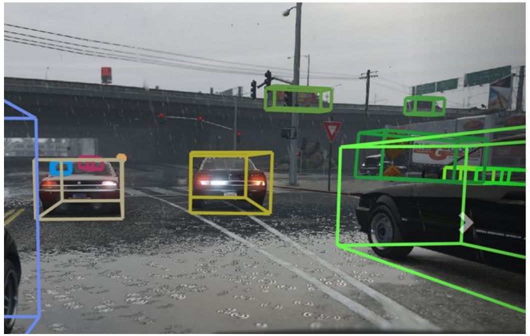
Semantic Segmentation Datasets

PASCAL VOC - PASCAL parts - MS COCO



ADE20K - NYU RGBD

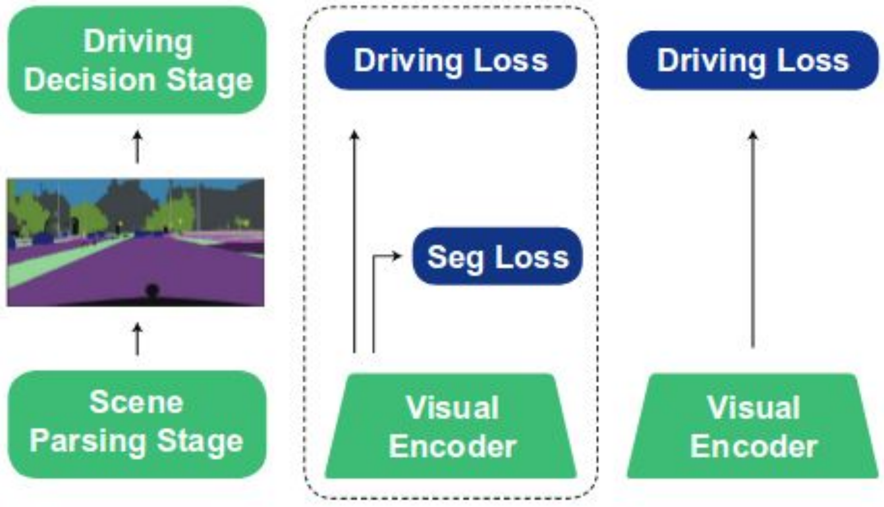
Cityscapes - BDD - Mapillary



Synthia - GTA - Virtual KITTI

Semantic Segmentation for Robotics

- Why do we need semantic segmentation? Why not end-to-end methods?
- Semantic segmentation can act as an auxiliary Loss. [1]



(a) Mediated Perception (b) Privileged Training (c) Motion Reflex

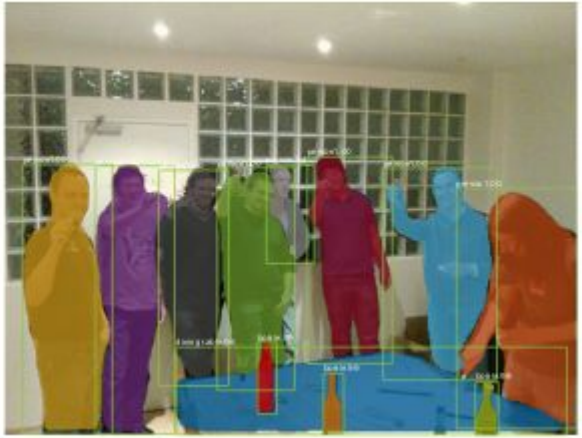
[1] Xu, Huazhe, et al. "End-to-end learning of driving models from large-scale video datasets." *arXiv preprint* (2017).

Agenda

- Introduction
- Fully Convolutional Networks
- FCN8s Architecture
- DeepLab Architecture
- Instance Segmentation [Mask R-CNN]
- Few-shot Segmentation
- Video Object Segmentation

Instance Segmentation

- Interested in segmenting each instance of a car on its own, not in just segmenting all cars



Faster R-CNN

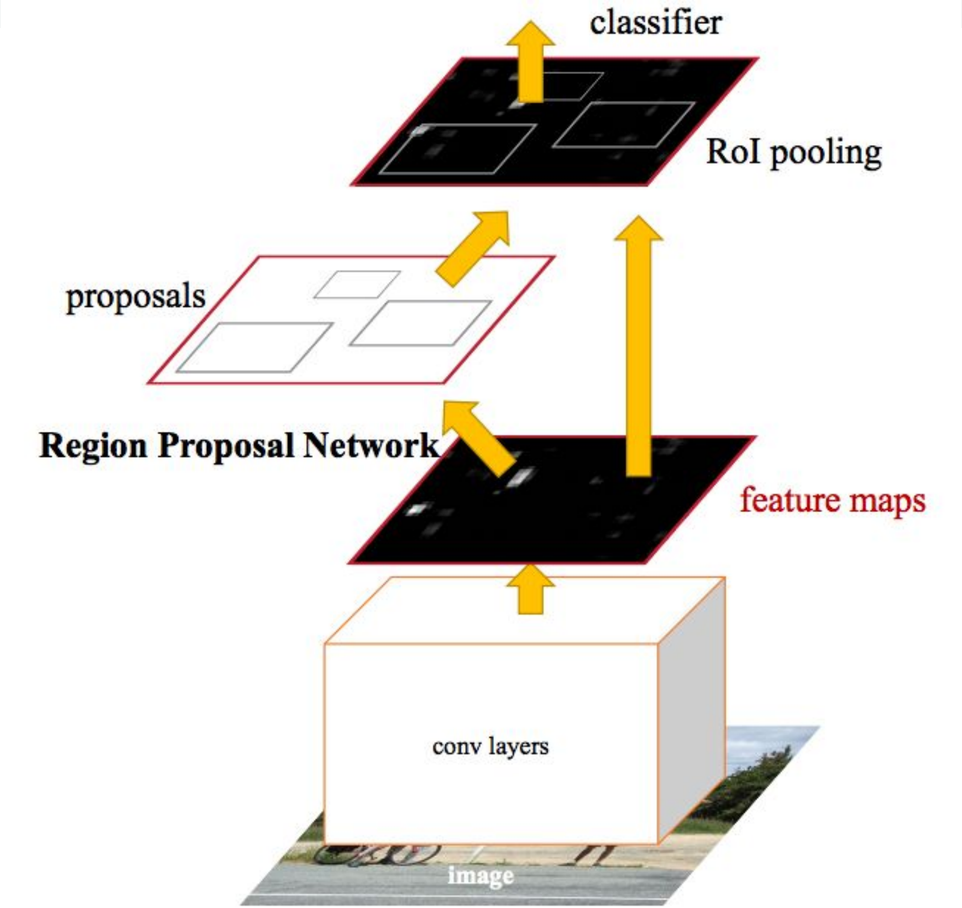
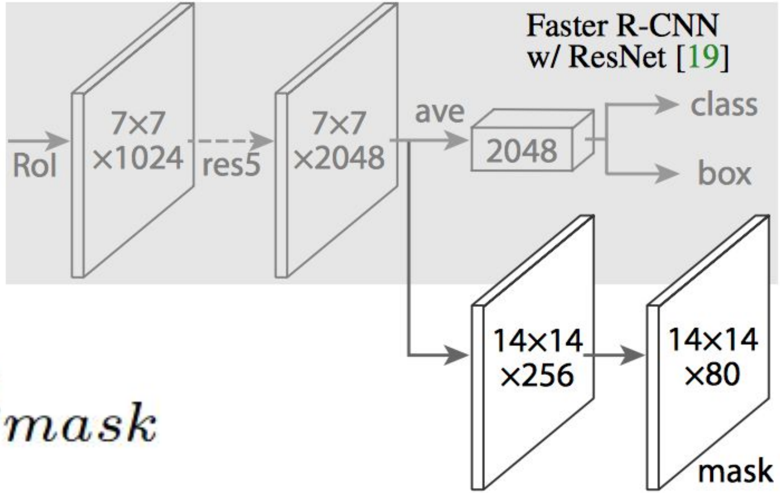


Figure:https://blog.athelas.com/a-brief-history-of-cnns-in-image-segmentation-from-r-cnn-to-mask-r-cnn-34ea83205de4

Mask R-CNN - MultiTask Loss

- Detection Head: Regression to refine bounding boxes
- Classification Head: Cross Entropy.
- Seg. Head: Pixel-wise Binary CE.

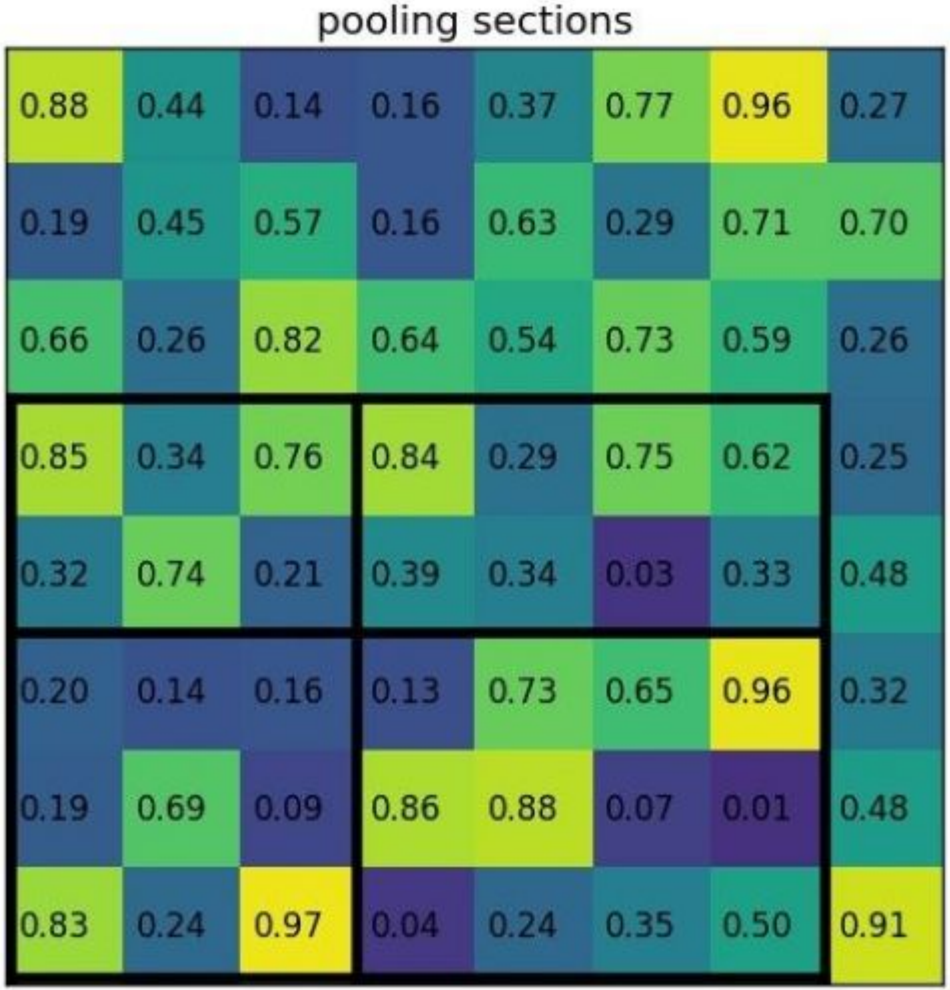


$$L = L_{cls} + L_{box} + L_{mask}$$

[1] He, Kaiming, et al. "Mask r-cnn." *Proceedings of the IEEE international conference on computer vision*. 2017.

ROI Pooling

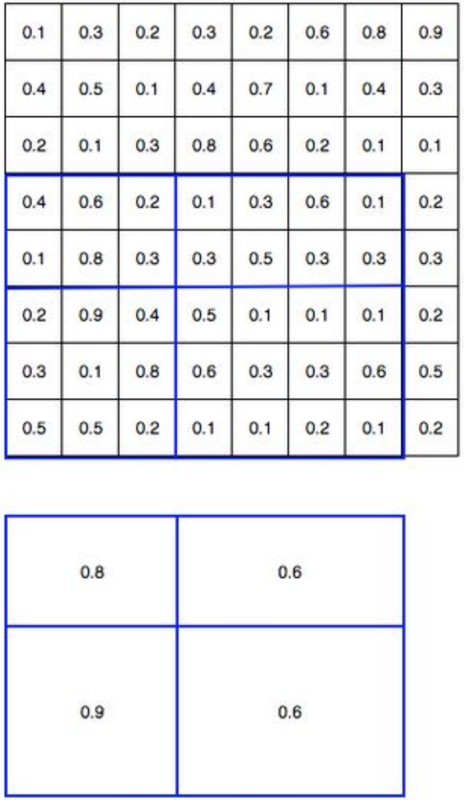
- Region Size: 7x5
- Output: 2x2

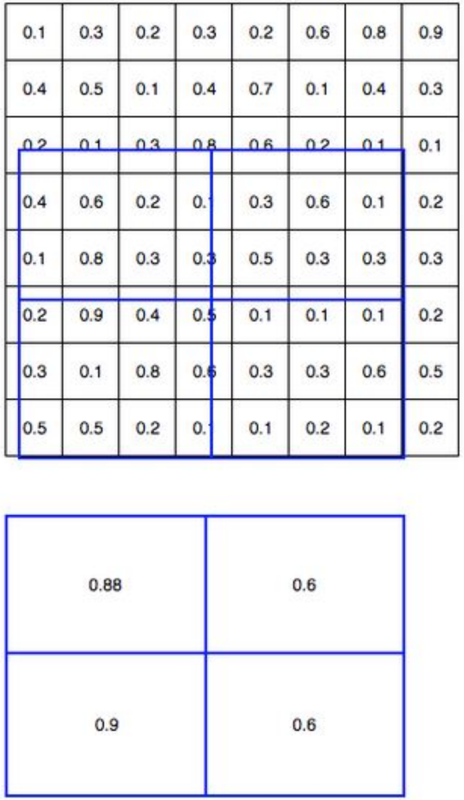


https://deepsense.ai/region-of-interest-pooling-explained/

ROI Align

Fix misalignment from ROI Pool
Bilinearly Interpolate features





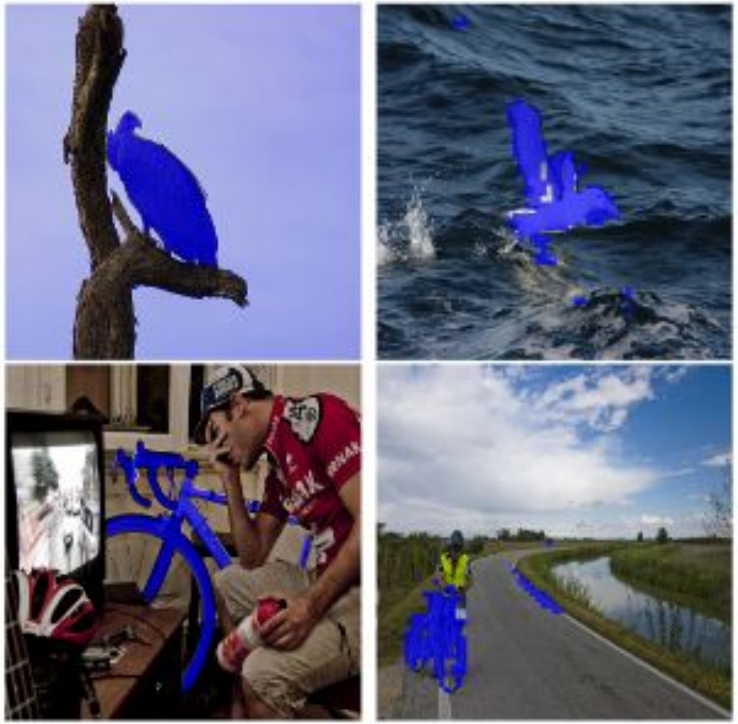
https://medium.com/@jonathan_hui/image-segmentation-with-mask-r-cnn-ebe6d793272

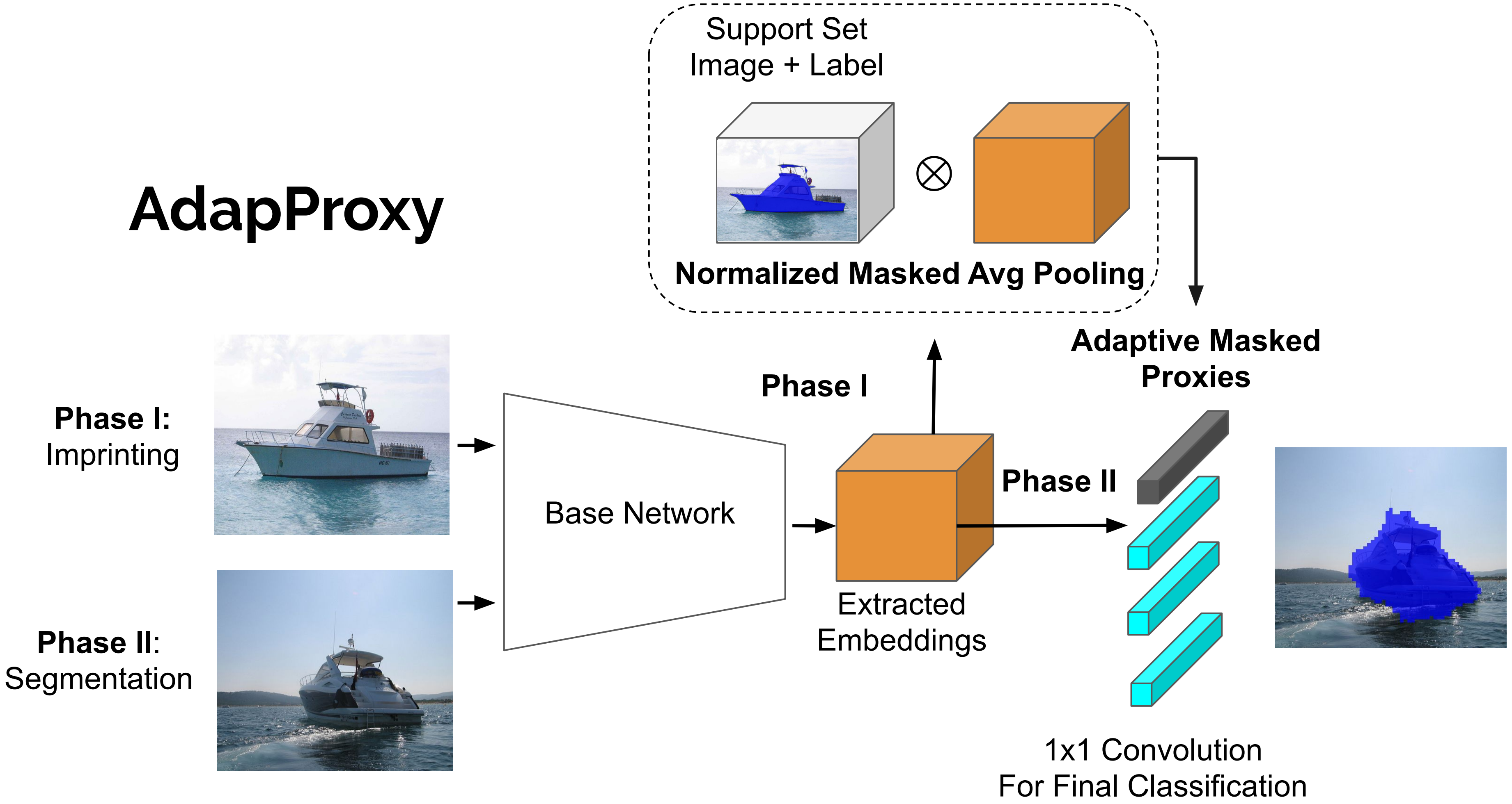
Agenda

- Introduction
- Fully Convolutional Networks
- FCN8s Architecture
- DeepLab Architecture
- Instance Segmentation [Mask R-CNN]
- Few-shot Segmentation
- Video Object Segmentation

Few-Shot Segmentation

- K-shot N-way formulation
- Support set, Query Image




AdapProxy
Support Set
Image + Label
Normalized Masked Avg Pooling
Adaptive Masked
Proxies
Phase I
Phase I:
Imprinting
Base Network
Phase II
Extracted
Embeddings
Phase II:
Segmentation
1x1 Convolution
For Final Classification

Metric Learning Relation to Softmax

- NCA (Neighbourhood Component Analysis) [1] : learns distance metric with a softmax-like loss.

$$L_{NCA}(x, y, Z) = -\log \frac{\exp(-d(x,y))}{\sum_{z \in Z} \exp\left(-d(x,z)\right)}$$

- NCA with Proxies [2]:

$$L_{proxy} = -\log \frac{\exp -d(x,p(x))}{\sum_{p(z) \in p(Z)} \exp(-d(x,p(z))}$$

[1] J. Goldberger, G. E. Hinton, S. T. Roweis, and R. R.Salakhutdinov. Neighbourhood components analysis. InAdvances in Neural Information Processing Systems, pages513–520, 2005.
[2] Y. Movshovitz-Attias, A. Toshev, T. K. Leung, S. Ioffe, andS. Singh. No fuss distance metric learning using proxies.InProceedings of the IEEE Conference on Computer Visionand Pattern Recognition, pages 360–368, 2017

Metric Learning Relation to Softmax

- Normalized Vectors $\min d(x, p(x)) = \max x^T p(x)$

$$L_{proxy} = -\log \frac{\exp - d(x, p(x))}{\sum_{p(z) \in p(Z)} \exp(-d(x, p(z)))}$$

Rethink of the
Weights as **Proxies**.

$$\downarrow$$

$$L_{softmax} = -\log \frac{\exp\left(x^T W_{q(x)}\right)}{\sum_{c \in C} \exp\left(x^T W_c\right)}$$

Adaptive Masked Proxies

- Normalized Masked Average Pooling Layer

$$P_l^r = \frac{1}{k} \sum_{i=1}^{k} \frac{1}{N} \sum_{x \in X} F^{ri}(x) Y_l^i(x)$$

$$\hat{P}_l^r = \frac{P_l^r}{||P_l^r||_2}$$

- Adaptation of proxies based on update rate α

$$\hat{W}_l^r = \alpha \hat{P}_l^r + (1 - \alpha) W_l^r$$

[1] Siam, Mennatullah, and Boris Oreshkin. "Adaptive Masked Weight Imprinting for Few-Shot Segmentation." *arXiv preprint arXiv:1902.11123* (2019).

Practical 1.4: AdapProxy Few-shot Segmentation

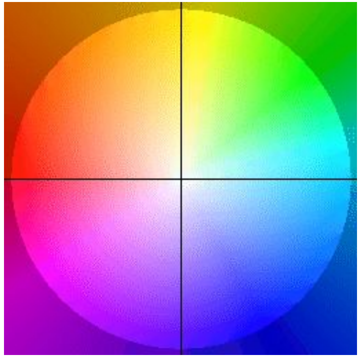
https://bit.ly/2DNmoYm

Agenda

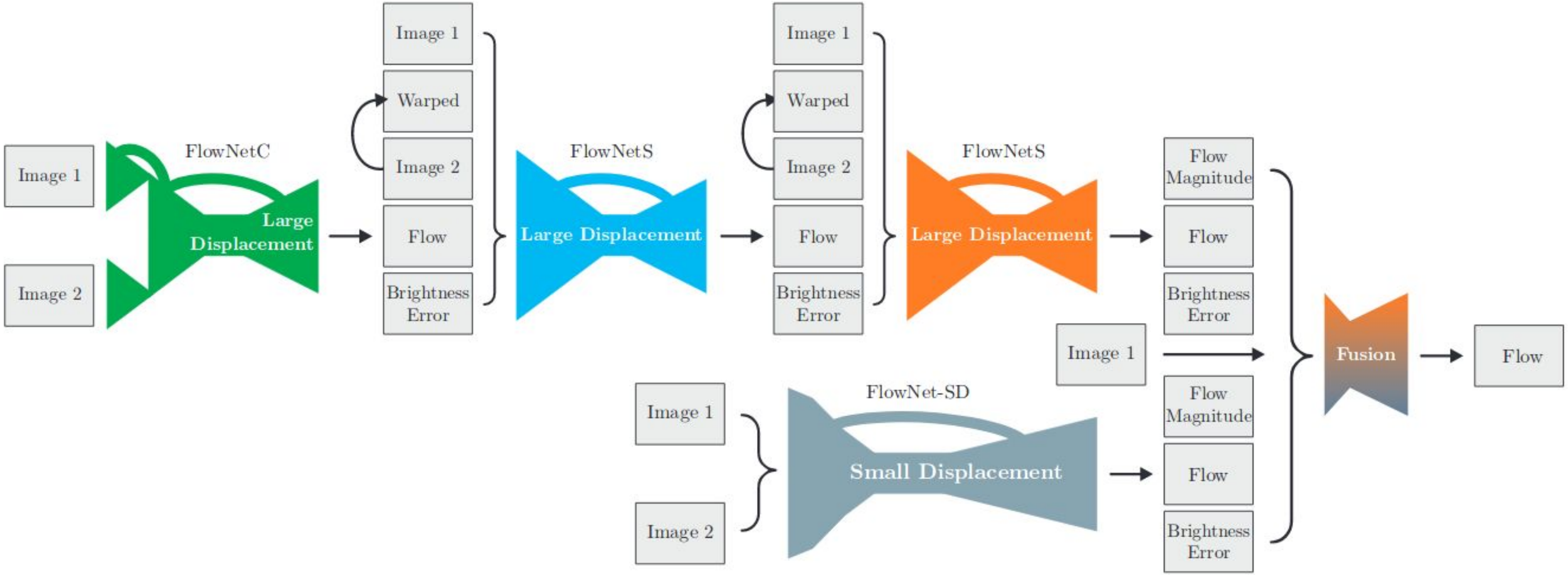
- Introduction
- Fully Convolutional Networks
- FCN8s Architecture
- DeepLab Architecture
- Instance Segmentation [Mask R-CNN]
- Few-shot Segmentation
- Video Object Segmentation

Video Object Segmentation

- Recurrent Networks
- Optical Flow



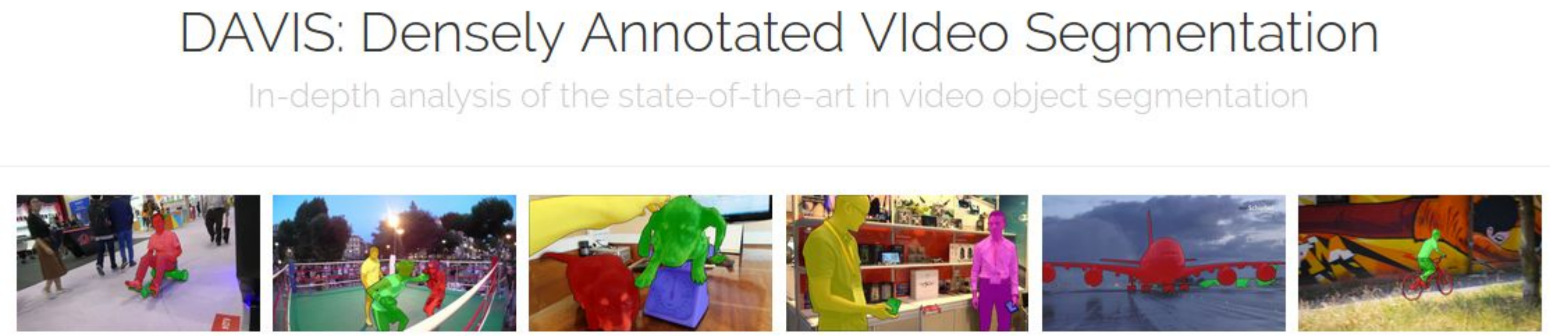
FlowNet 2.0



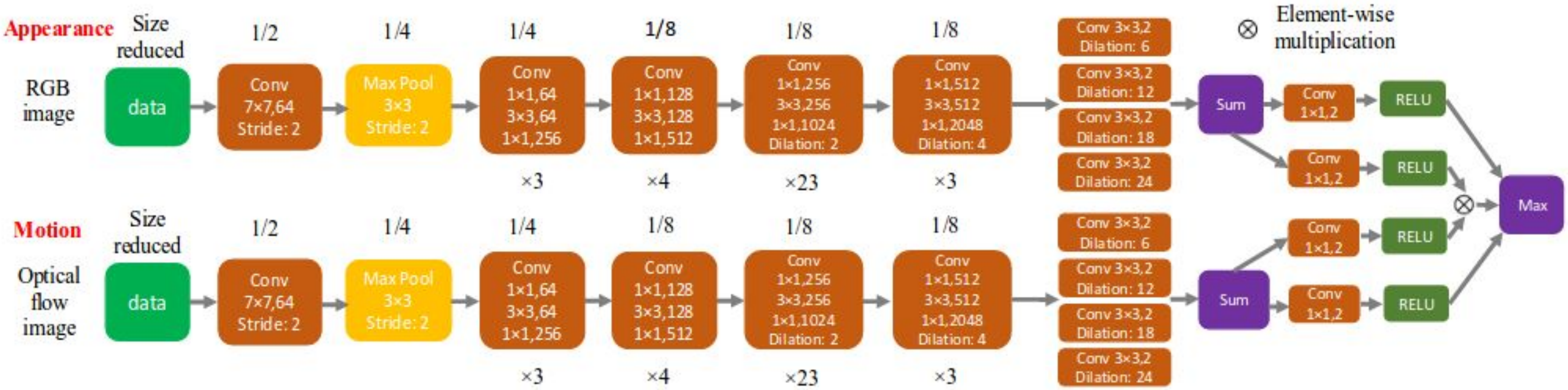
FlowNetC
Image 1
Image 2
Large Displacement
Image 1
Warped
Image 2
Flow
Brightness Error
FlowNetS
Large Displacement
Image 1
Warped
Image 2
Flow
Brightness Error
FlowNetS
Large Displacement
Flow Magnitude
Flow
Brightness Error
Image 1
FlowNet-SD
Image 1
Image 2
Small Displacement
Flow Magnitude
Flow
Brightness Error
Fusion
Flow

Video Object Segmentation

- **Semi-supervised:** Initialize with first frame mask.
- **Unsupervised:** No first frame initialization.
- **Interactive:** Scribbles from user.



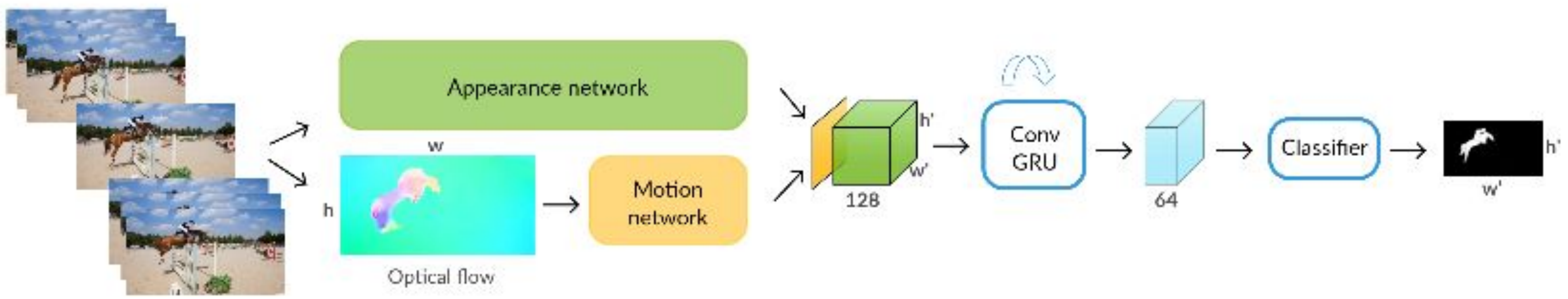
Unsupervised Video Object Segmentation



FusionSeg CVPR’17[1]

[1] Jain, Suyog Dutt, Bo Xiong, and Kristen Grauman. "Fusionseg: Learning to combine motion and appearance for fully automatic segmentation of generic objects in videos." *Proc. CVPR*. Vol. 1. No. 2. 2017.

Unsupervised Video Object Segmentation



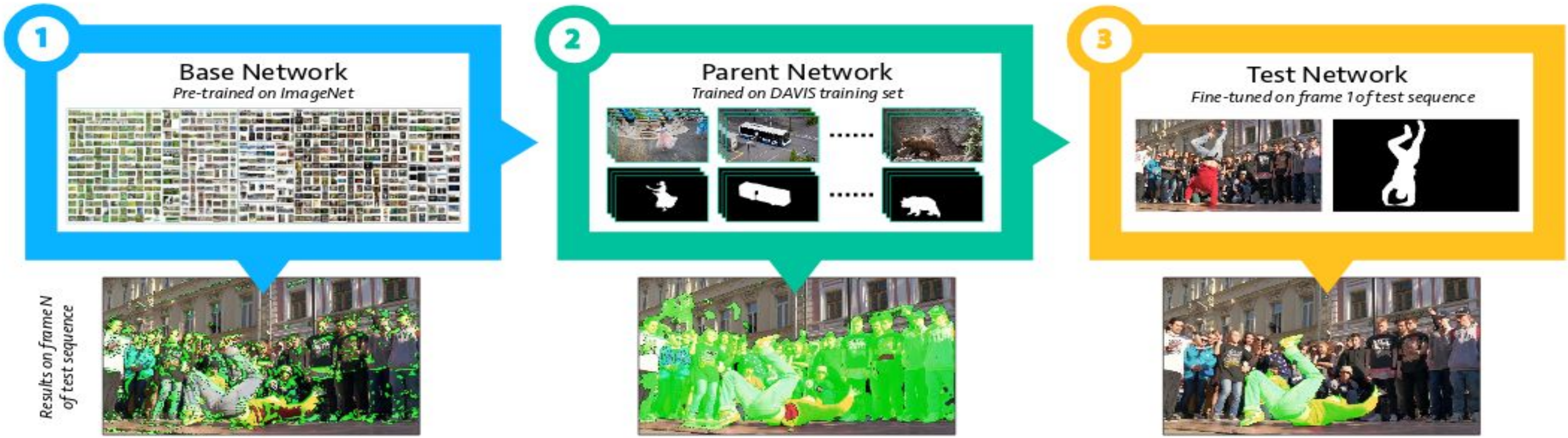
LVO ICCV'17 [2]

[2]Tokmakov, Pavel, Karteek Alahari, and Cordelia Schmid. "Learning video object segmentation with visual memory." *arXiv preprint arXiv:1704.05737* 3 (2017).

Practical 1.5: Two-stream FCN

https://bit.ly/2DNmoYm

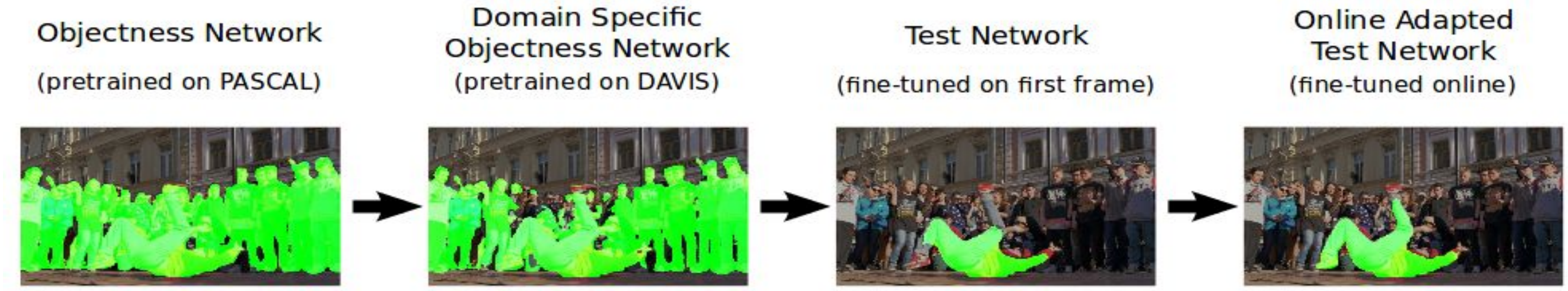
Semi-supervised Video Object Segmentation



OSVOS CVPR’17

[1] Caelles, Sergi, et al. "One-shot video object segmentation." *CVPR 2017*. IEEE, 2017.

Semi-supervised Video Object Segmentation



OnaVOS BMVC’17

[1] Voigtlaender, Paul, and Bastian Leibe. "Online adaptation of convolutional neural networks for video object segmentation." *arXiv preprint arXiv:1706.09364* (2017).

Thanks