

Deep Learning Segmentation

What is Segmentation?

- It is the process of **dividing an image** into **different regions** based on the **characteristics of pixels** to identify objects or boundaries to simplify an image and more efficiently analyse it.
- Think of it as predicting the class for each pixel in an image.

Is this a dog?



Image Classification

What is there in image and where?



Object Detection

Which pixels belong to which object?

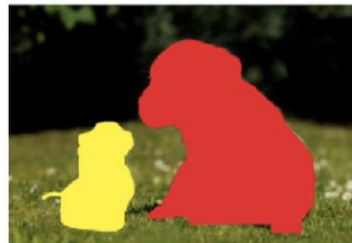


Image Segmentation

Segmentation Techniques

- Using K-means to extract/segment a cat

Picture and background:



Segmented: The pixels are partitioned into five groups, shown below. You can select the groups that form the kitten portion, discard the background (image 5, in this case), and move the segmented parts onto a separate background.



Final result: A kitty sleeping on a leaf!



- These methods don't work well for Multiple Classes and cluttered Scenes





Input

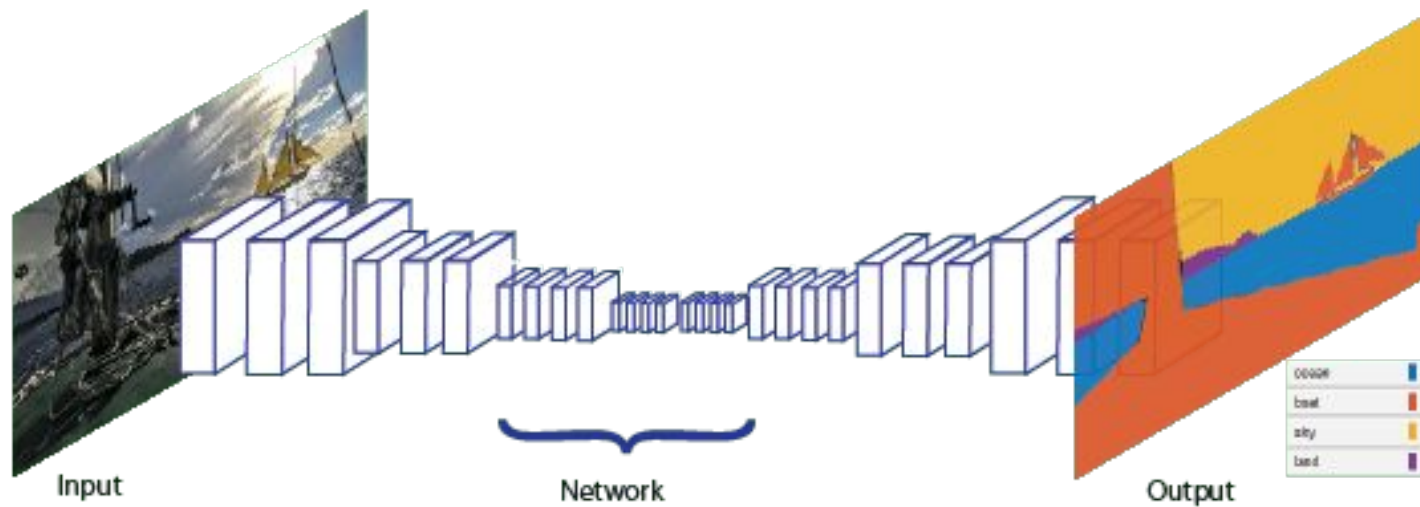


- 1: Person
- 2: Purse
- 3: Plants/Grass
- 4: Sidewalk
- 5: Building/Structures

3	3	3	3	3	3	3	3	3	3	3	3	3	5	5	5	5	5	5
3	3	3	3	3	3	3	3	3	3	3	3	3	5	5	5	5	5	5
3	3	3	3	3	3	1	1	3	3	3	3	3	5	5	5	5	5	5
3	3	3	3	3	1	1	1	1	3	3	3	3	5	5	5	5	5	5
3	3	3	3	3	3	1	1	3	3	3	3	5	5	5	5	5	5	5
5	5	3	3	3	3	1	1	3	3	5	5	5	5	5	5	5	5	5
4	4	3	4	1	1	1	1	1	1	4	4	4	5	5	5	5	5	5
4	4	3	4	1	1	1	1	1	1	4	4	4	4	4	5	5	5	5
4	4	4	1	1	1	1	1	1	1	1	4	4	4	4	4	4	4	4
3	3	3	1	1	1	1	1	1	1	1	4	4	4	4	4	4	4	4
3	3	3	1	2	2	1	1	1	1	1	4	4	4	4	4	4	4	4
3	3	3	1	2	2	1	1	1	1	1	4	4	4	4	4	4	4	4

Semantic Labels

Deep Learning for Semantic Segmentation



Encoder - Decoder Architecture

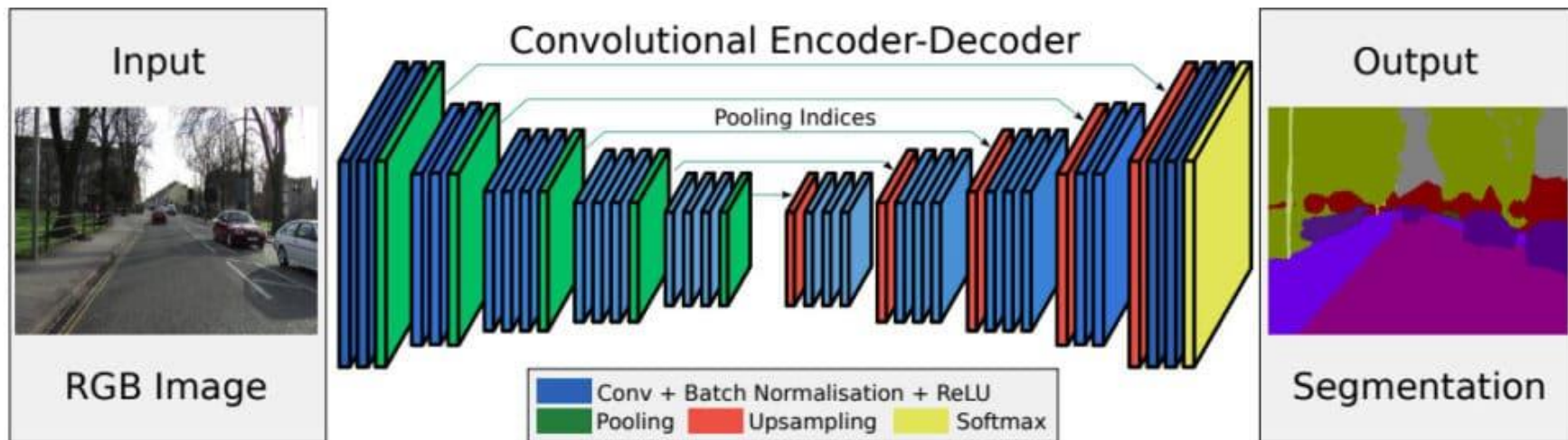
Encoder:

- **Purpose:** The encoder's primary role is to **extract features** from the input image. It gradually **reduces the spatial dimensions** (i.e., height and width) of the image while **increasing the depth** (the number of feature maps). This process helps in capturing the context in the image, which is essential for understanding the scene.
- **How it Works:** The encoder typically consists of a series of **convolutional layers**, often borrowed from established classification networks like **VGG, ResNet, or MobileNet**.
- **Output:** The output of the encoder is a high-level, semantically rich, but spatially smaller feature representation of the input image.

Encoder - Decoder Architecture

Decoder:

- **Purpose:** The decoder's job is to **map the high-level features (extracted by the encoder) back to the original spatial dimensions** to generate a **pixel-wise** classification map.
- **How it Works:** The decoder gradually **upsamples** the feature maps, either through upconvolution (transposed convolution) or simpler methods like bilinear upsampling. It might also **combine these upsampled features with lower-level features** from the encoder (via skip connections) to retain fine-grained details.
- **Output:** The final output of the decoder is a segmentation map of the **same size** as the input image, where each pixel is classified into one of the classes.



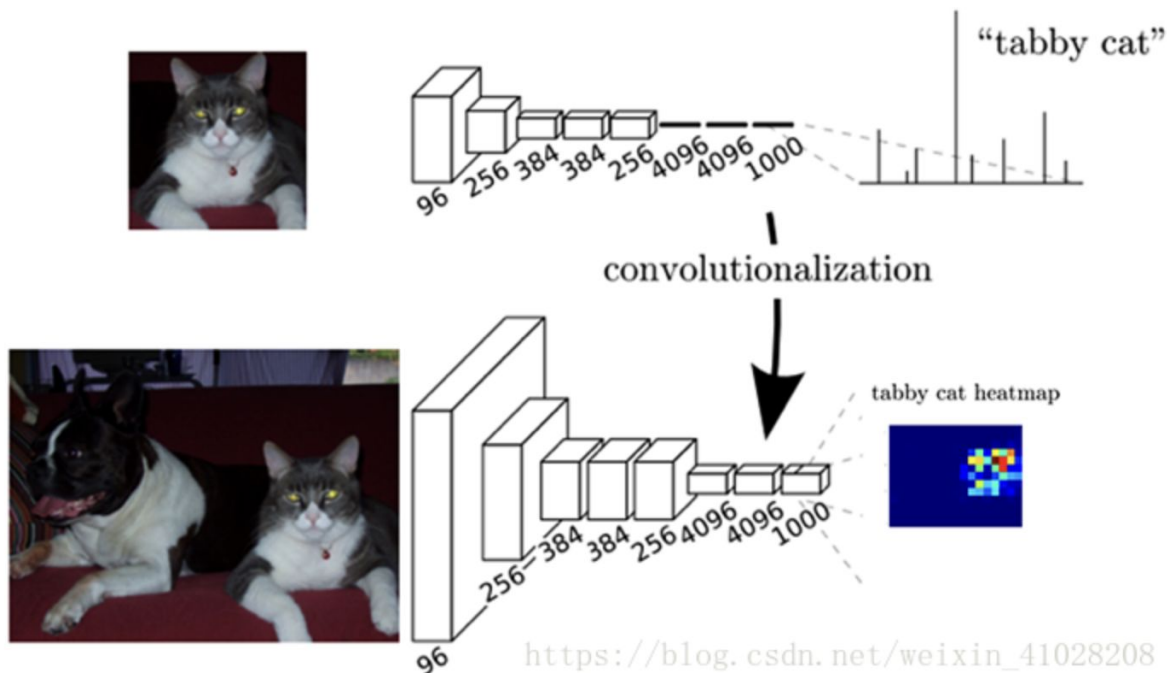
Segmentation Models

- Semantic segmentation:
 - FCN, U-Net, SegNet
- Instance segmentation:
 - Mask R-CNN

FCN (Fully Convolutional Networks)

- Replace the fully connected layers of CNN with **convolutional layers** to retain the information of **spatial features**.
- Use **upsampling** to return the convolutionalized feature map to its original size.

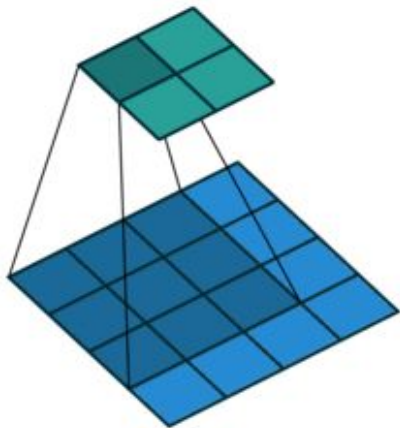
Replace Fully Connected Layers



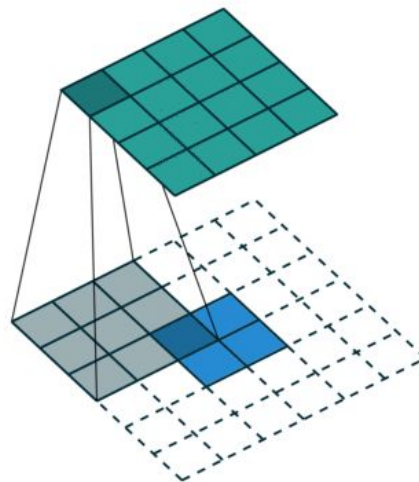
Upsampling

- Increase the size of feature map using **Deconvolution**, also called **Transposed Convolution**
- First padding, then convolution

Convolution

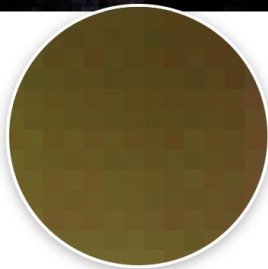


Deconvolution

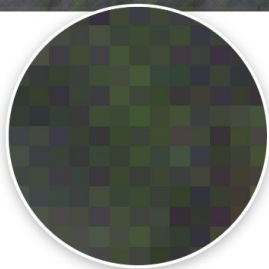
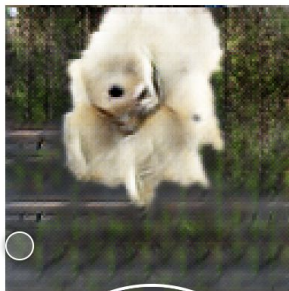


Checkerboard Artifacts

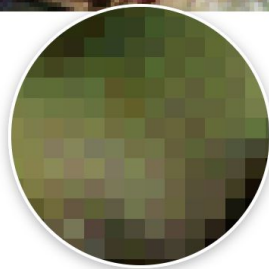
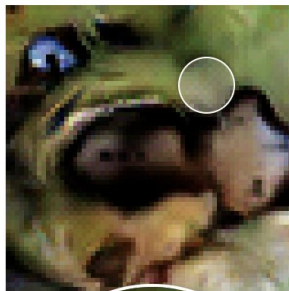
<https://distill.pub/2016/deconv-checkerboard/>



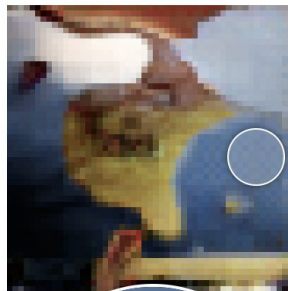
Radford, et al., 2015 [1]



Salimans et al., 2016 [2]



Donahue, et al., 2016 [3]

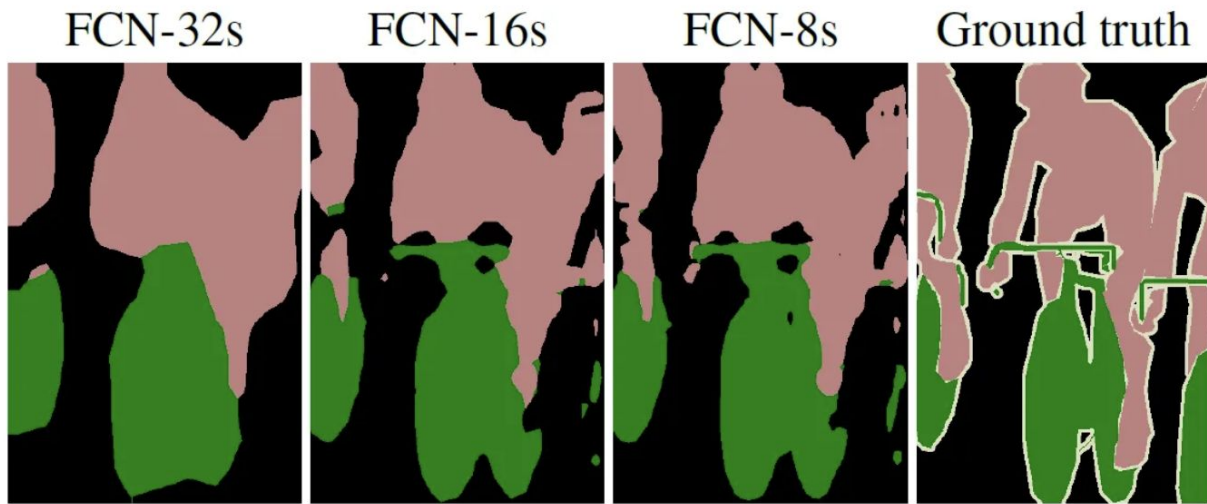


Dumoulin, et al., 2016 [4]



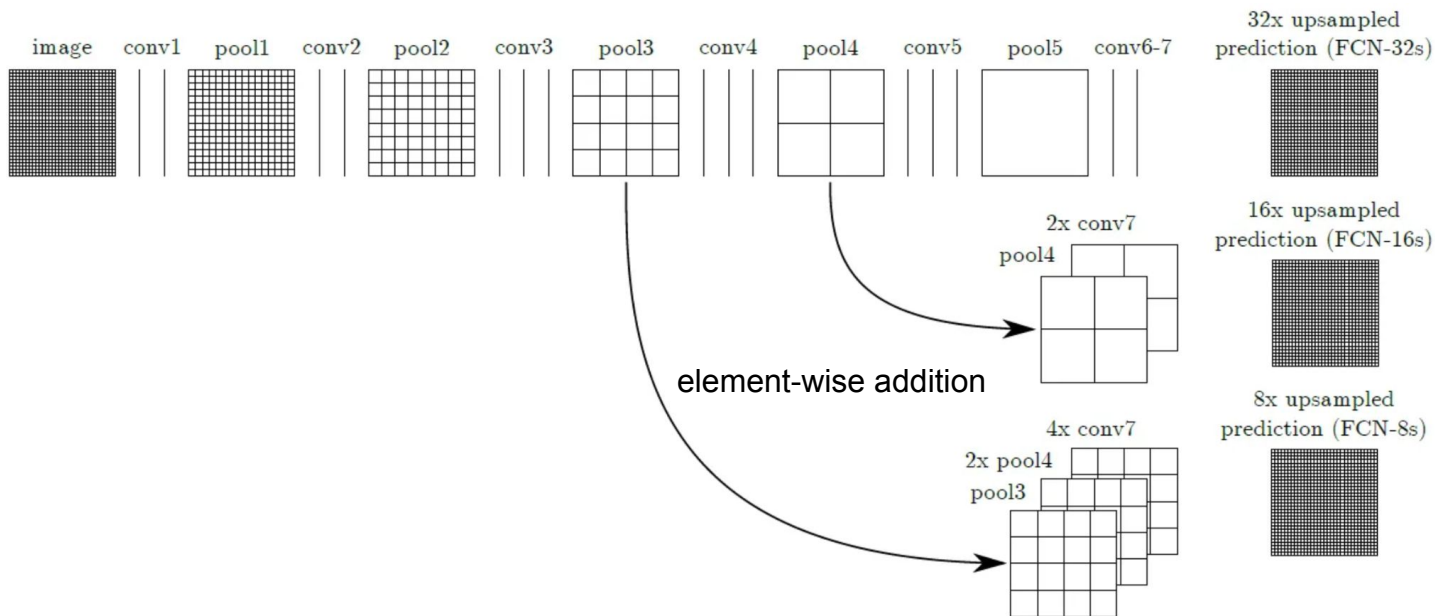
Upsampling

- Now FCN needs to enlarge the small size feature map back
- If you directly zoom back from the smallest feature map, the result will be very poor

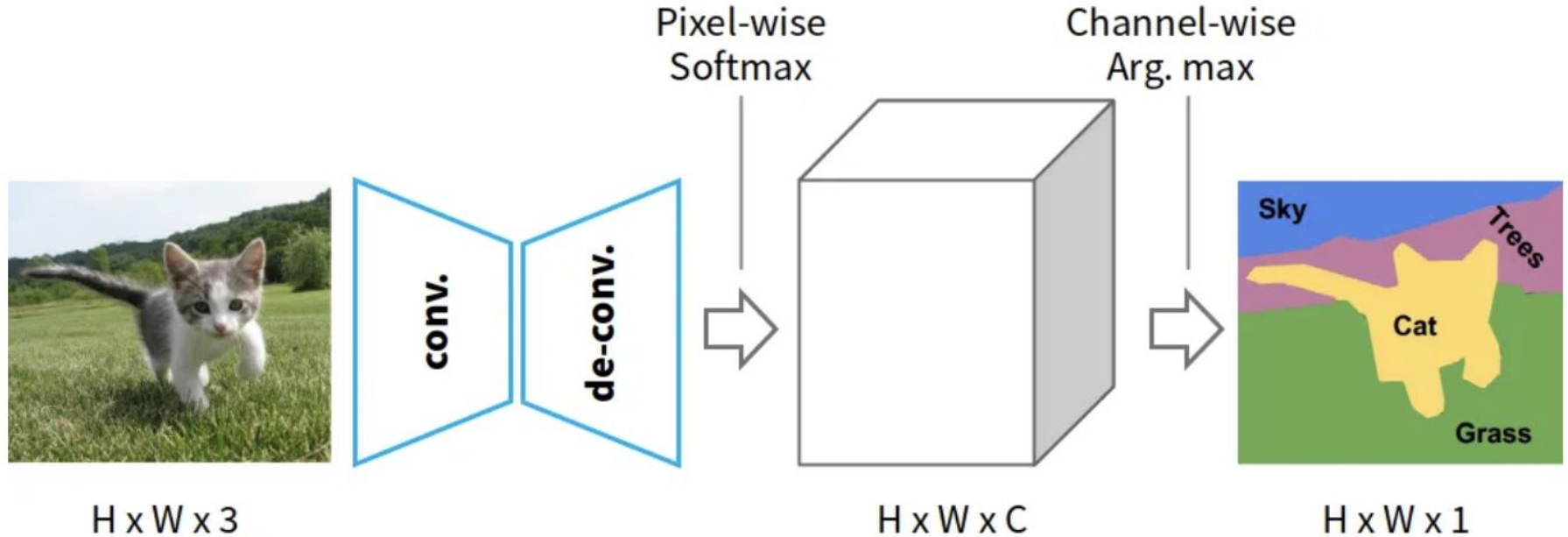


Upsampling

- Therefore, the feature map obtained during pooling needs to be taken into account when enlarging it, so that during the enlarging process, features at different scales can be integrated to obtain finer results.

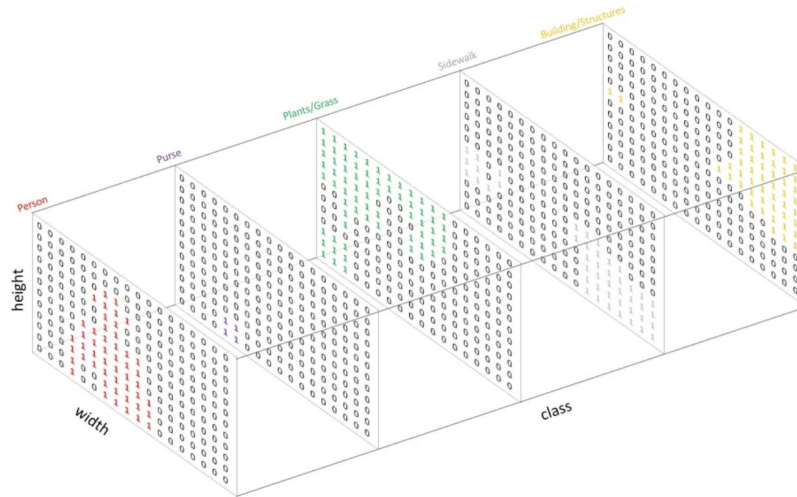


FCN (Fully Convolutional Networks)



Output of Semantic Segmentation

- If we take each channel apart and look at it, we will find that after FCN calculates the softmax probability of the category classification of each pixel.
- It will take the maximum probability channel-wise, and use the channel with the maximum probability as the category to which the pixel at that position belongs



Output of Semantic Segmentation



Input



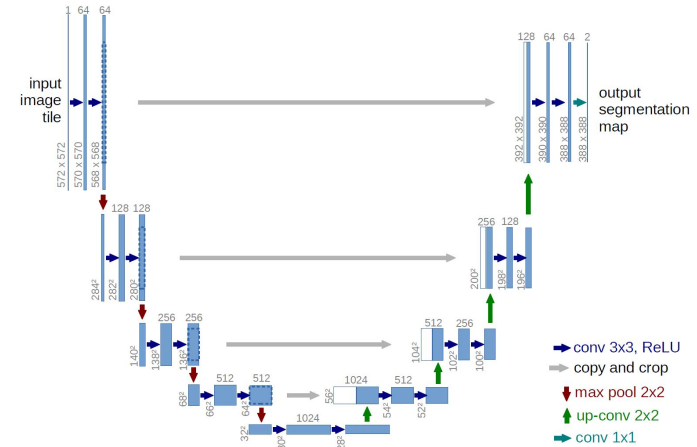
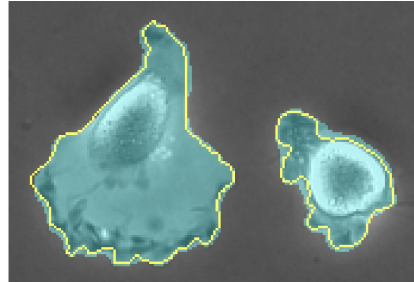
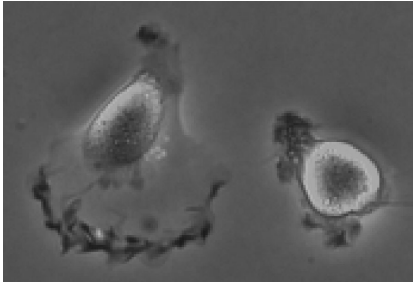
- 1: Person
- 2: Purse
- 3: Plants/Grass
- 4: Sidewalk
- 5: Building/Structures

3	3	3	3	3	3	3	3	3	3	3	3	3	5	5	5	5	5	5
3	3	3	3	3	3	3	3	3	3	3	3	3	5	5	5	5	5	5
3	3	3	3	3	3	1	1	3	3	3	3	3	5	5	5	5	5	5
3	3	3	3	3	1	1	1	1	3	3	3	3	5	5	5	5	5	5
3	3	3	3	3	3	1	1	3	3	3	5	5	5	5	5	5	5	5
5	5	3	3	3	3	1	1	3	3	5	5	5	5	5	5	5	5	5
4	4	3	4	1	1	1	1	1	1	4	4	4	5	5	5	5	5	5
4	4	3	4	1	1	1	1	1	1	4	4	4	4	4	5	5	5	5
4	4	4	1	1	1	1	1	1	1	1	4	4	4	4	4	4	4	4
3	3	3	1	1	1	1	1	1	1	1	4	4	4	4	4	4	4	4
3	3	3	1	2	2	1	1	1	1	1	4	4	4	4	4	4	4	4
3	3	3	1	2	2	1	1	1	1	1	4	4	4	4	4	4	4	4

Semantic Labels

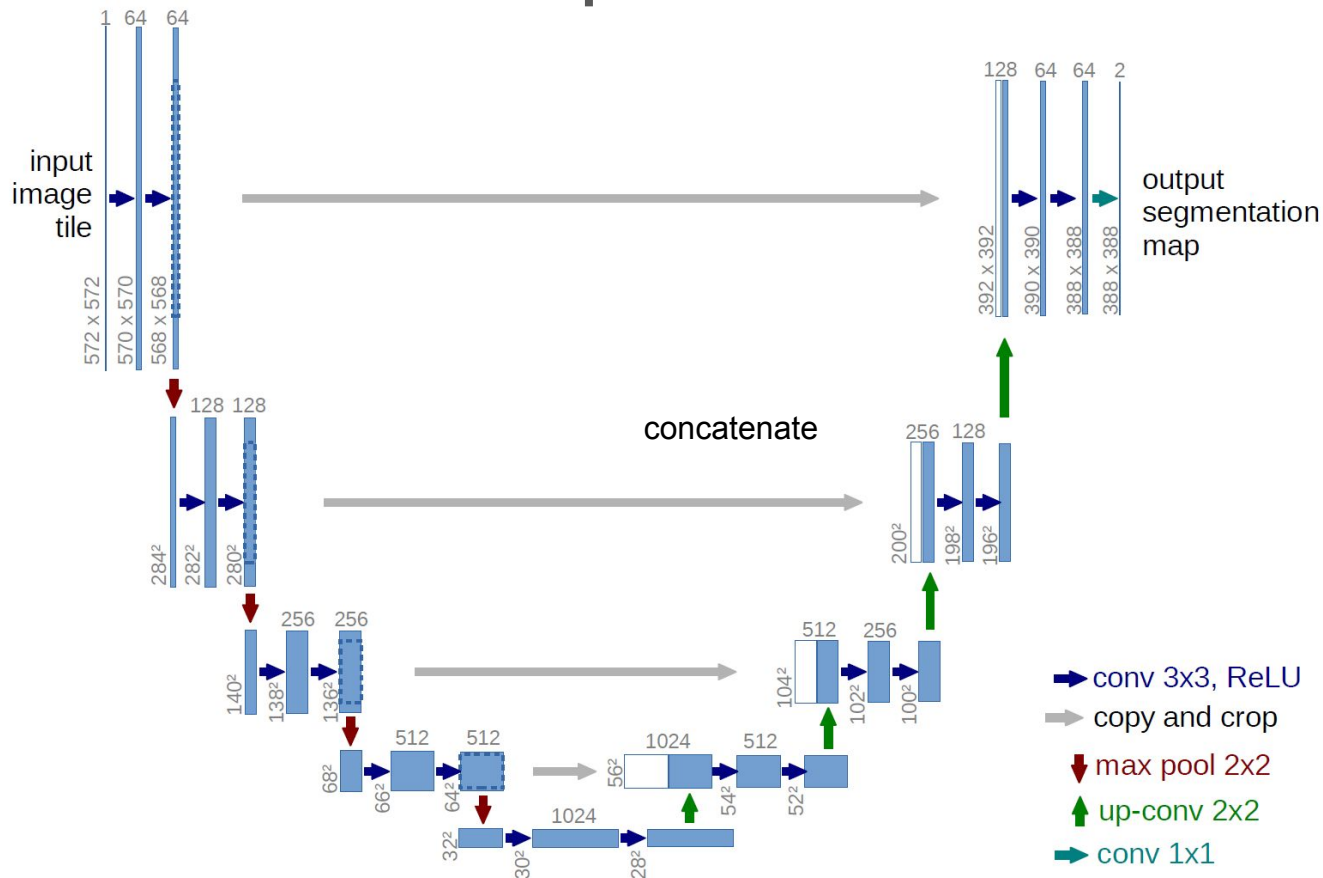
U-Net

- Created in 2015, U-Net was a unique CNN developed for Biomedical Image Segmentation.
- U-Net has now become a very popular end-to-end encoder-decoder network for semantic segmentation.
- It has a unique Up-Down architecture which has a Contracting path and an Expansive path.



Contraction path

Expansion path

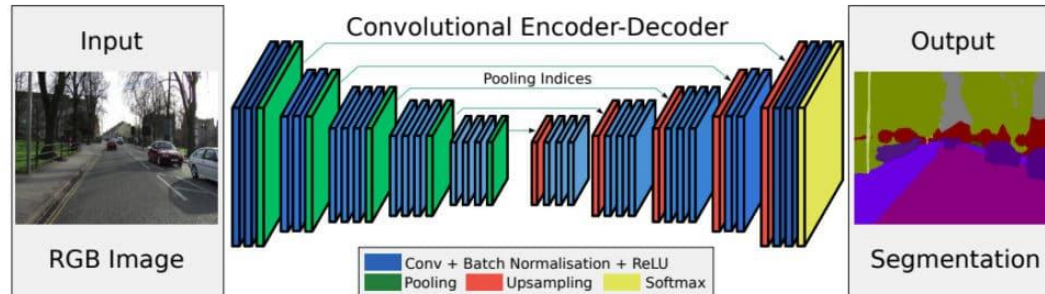


U-Net 優點

- 較精細的分割: skip connections 使U-Net能夠保留空間資訊和細節, 特別適合需要精細分割的任務。
- 適應小樣本: U-Net最初是為生物醫學影像設計的, 由於其架構設計可以充分利用數據中的訊息, 因此在數據量較小時可以表現良好。

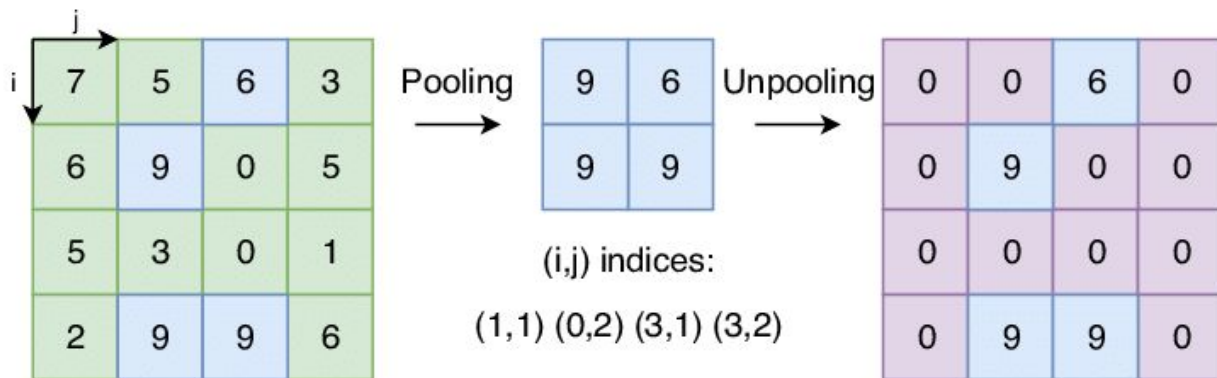
SegNet (Semantic Segmentation)

- Developed in 2015, SegNet is a semantic segmentation model.
- It consists of an encoder and decoder network followed by a pixel-wise classification layer.
- The architecture of the encoder network is topologically identical to the 13 convolutional layers in the VGG16 network.
- https://www.youtube.com/watch?v=CxanE_W46ts



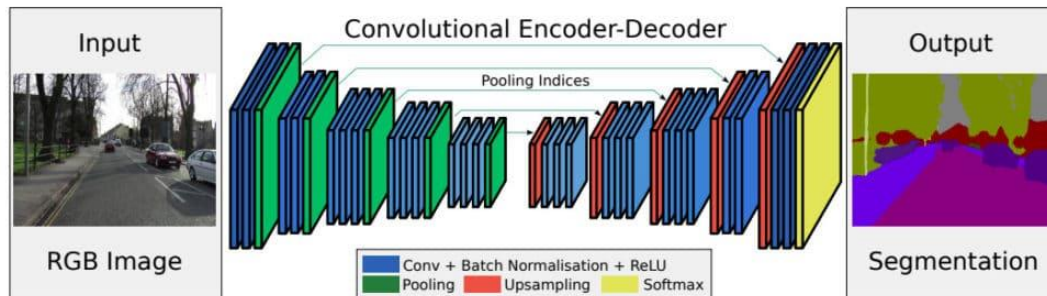
Encoder

- In the encoder, perform convolution and max pooling.
- VGG-16 has 13 convolutional layers. (The original fully connected layer will be discarded.)
- When executing 2×2 max pooling, the corresponding **max pooling index** (position) is stored.



Decoder

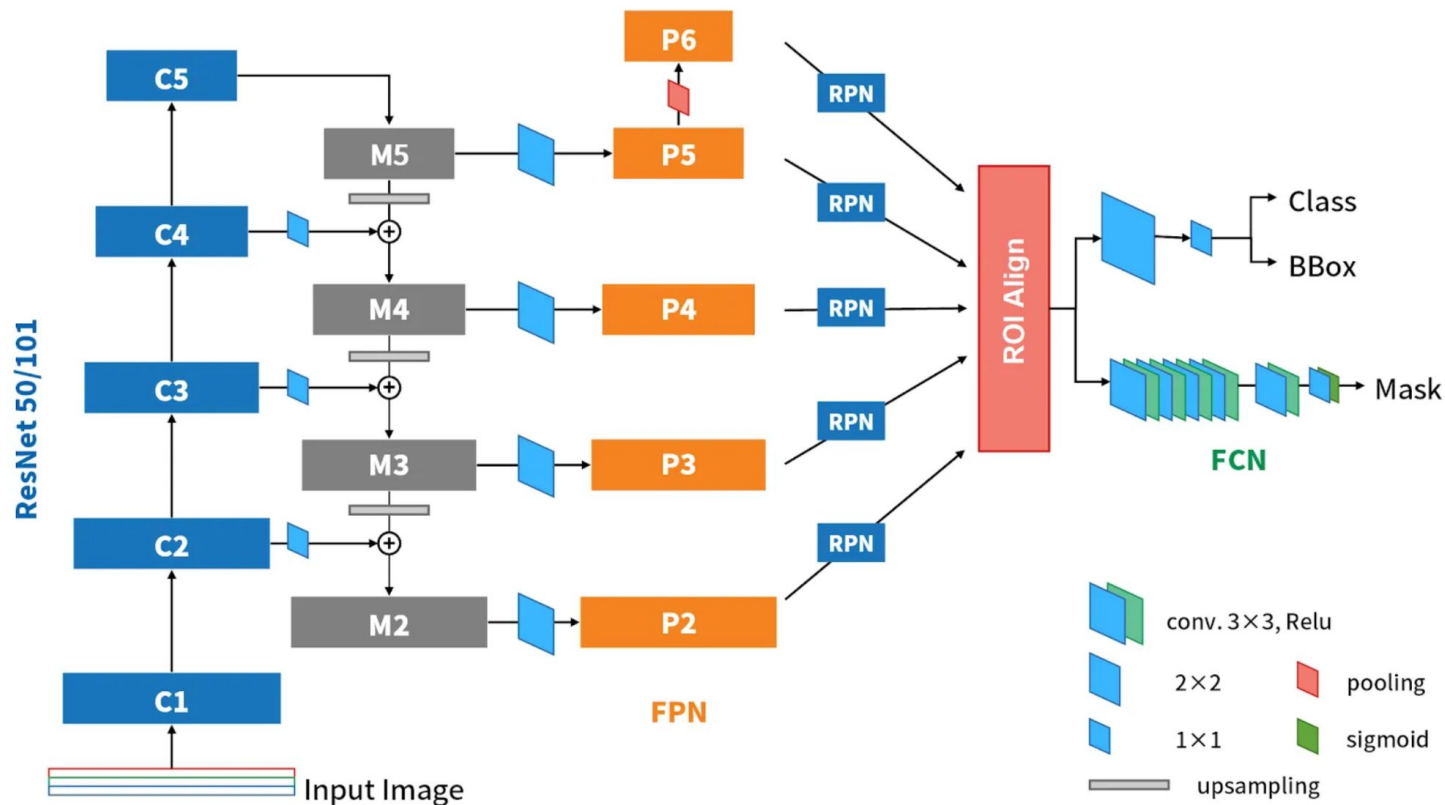
- In the decoder, perform **upsampling** and **convolution**.
- During upsampling, as shown above, the **max pooling index** of the corresponding encoder layer is called for upsampling.
- Finally, a K-class softmax classifier is used to predict the class of each pixel.



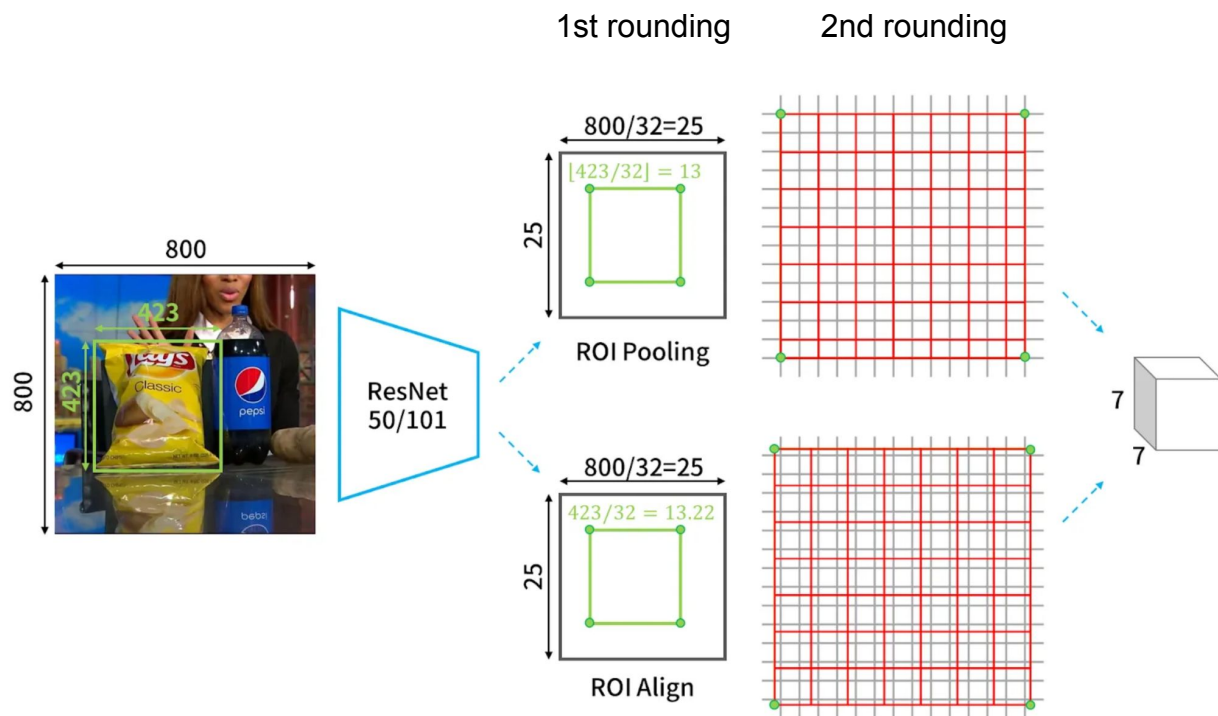
Mask R-CNN

- Can be imagined as Faster R-CNN that uses ResNet/ResNeXt and FPN to enhance feature extraction and feature integration capabilities
- Replace ROI Pooling with ROI Align which will not affect accuracy caused by rounding
- One more mask branch is used to perform instance segmentation tasks on the network
- Mask R-CNN is a quite large model, which also makes its inference speed not fast enough. If there are a large number of target categories, training is also very time-consuming, which makes it impractical for application.

Mask R-CNN

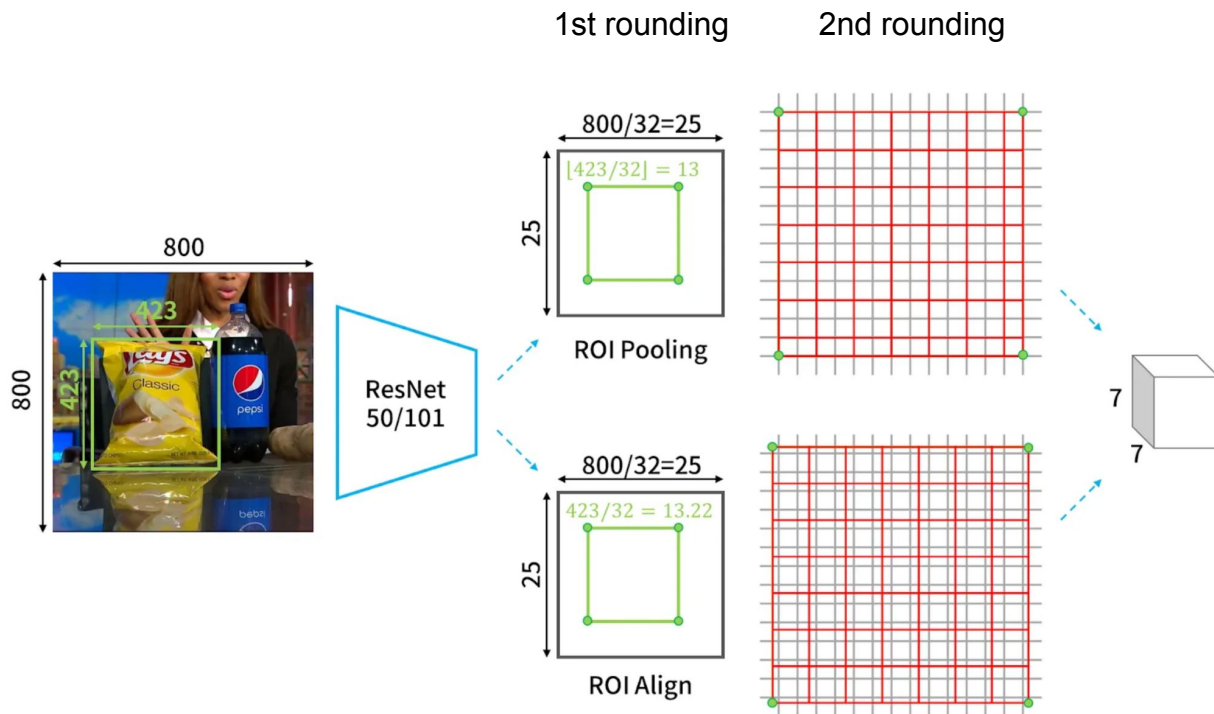


ROI Align



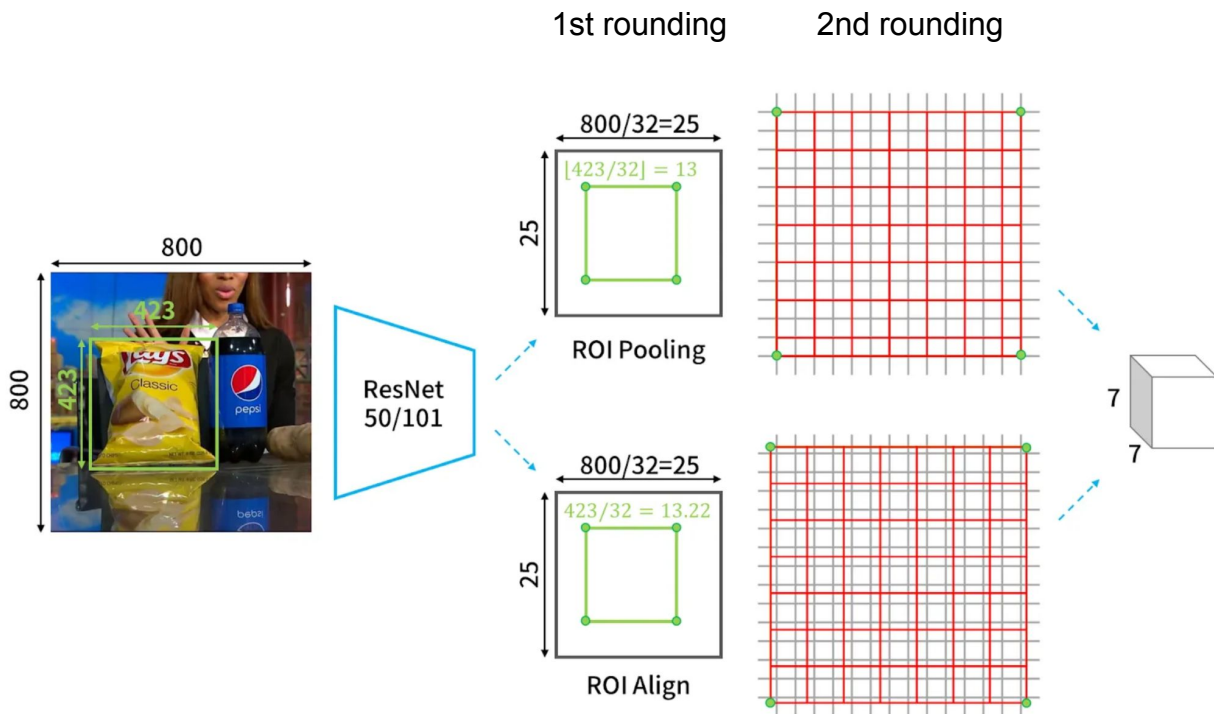
ROI Align

Instance segmentation requires fine pixel-level masks



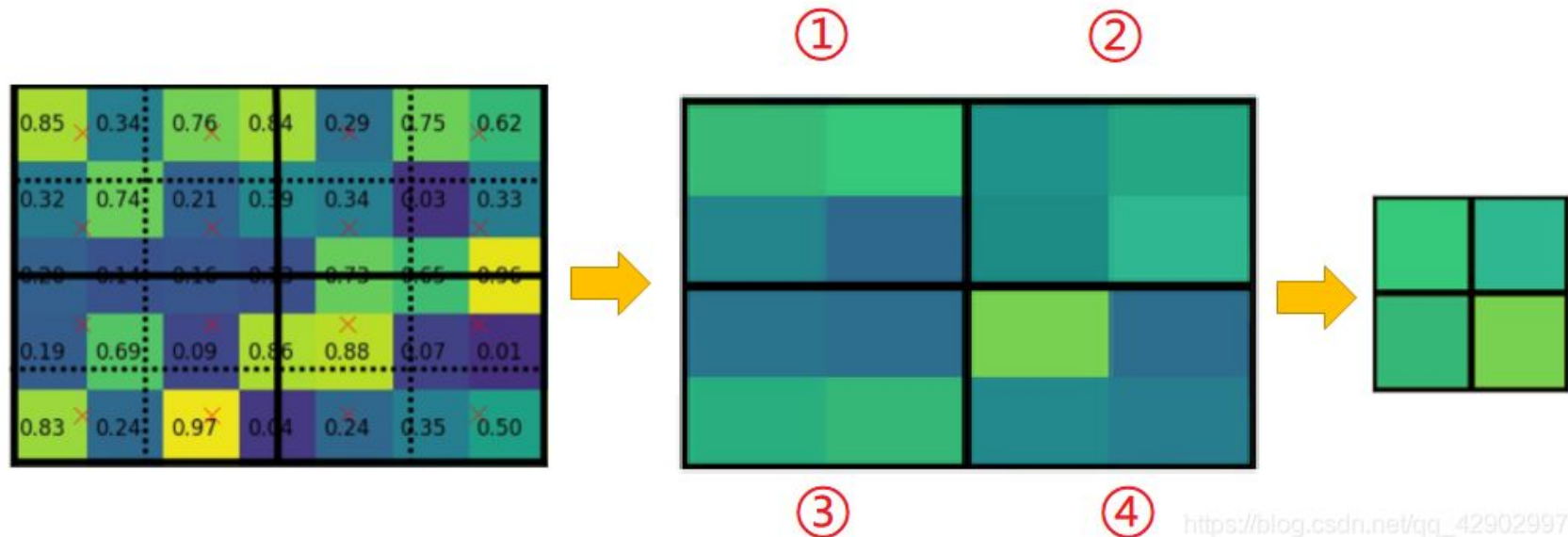
ROI Align

Instance segmentation requires fine pixel-level masks

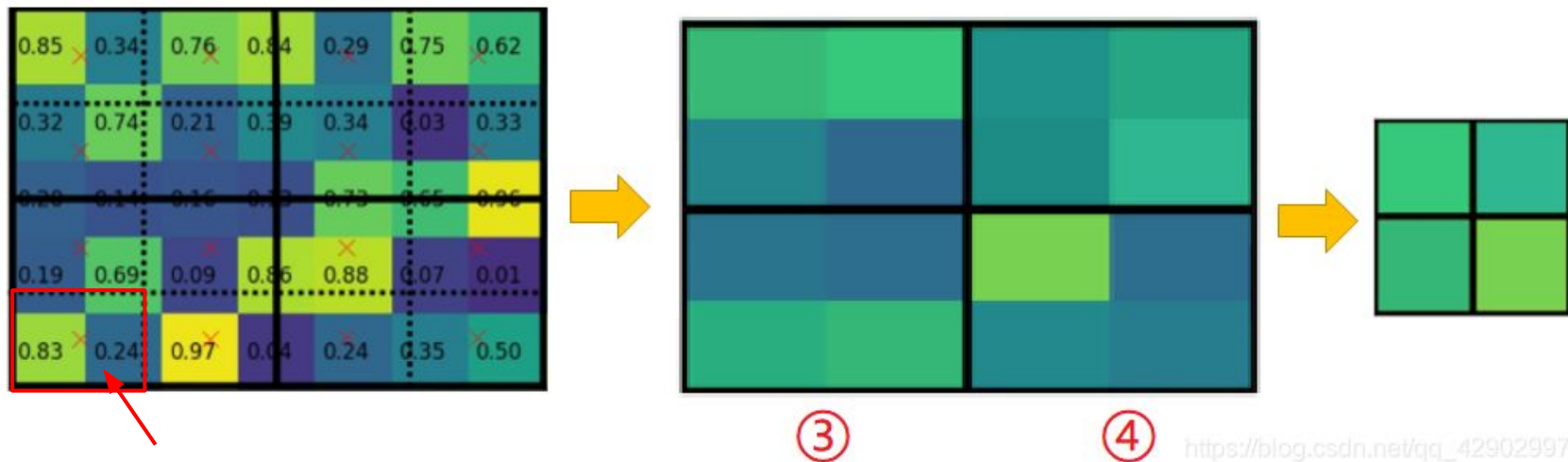


- The major improvement of ROI Align is the [description of position](#).
- ROI Pooling requires a [two rounding process](#).
- ROI Align uses **Bilinear Interpolation** to achieve good progress.

ROI Align Process



ROI Align Process



https://blog.csdn.net/qq_42902997

Bilinear Interpolation

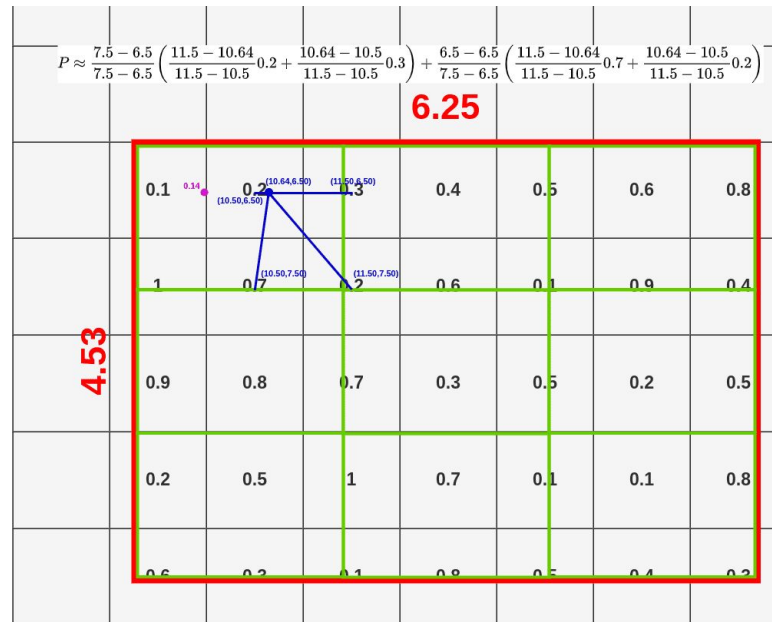
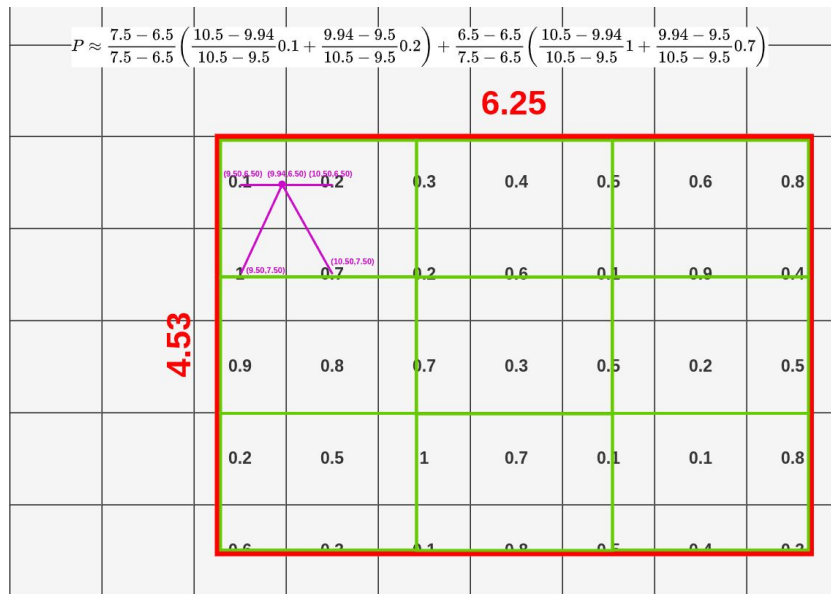


$$P \approx \frac{y_2 - y}{y_2 - y_1} \left(\frac{x_2 - x}{x_2 - x_1} Q_{11} + \frac{x - x_1}{x_2 - x_1} Q_{21} \right) + \frac{y - y_1}{y_2 - y_1} \left(\frac{x_2 - x}{x_2 - x_1} Q_{12} + \frac{x - x_1}{x_2 - x_1} Q_{22} \right)$$

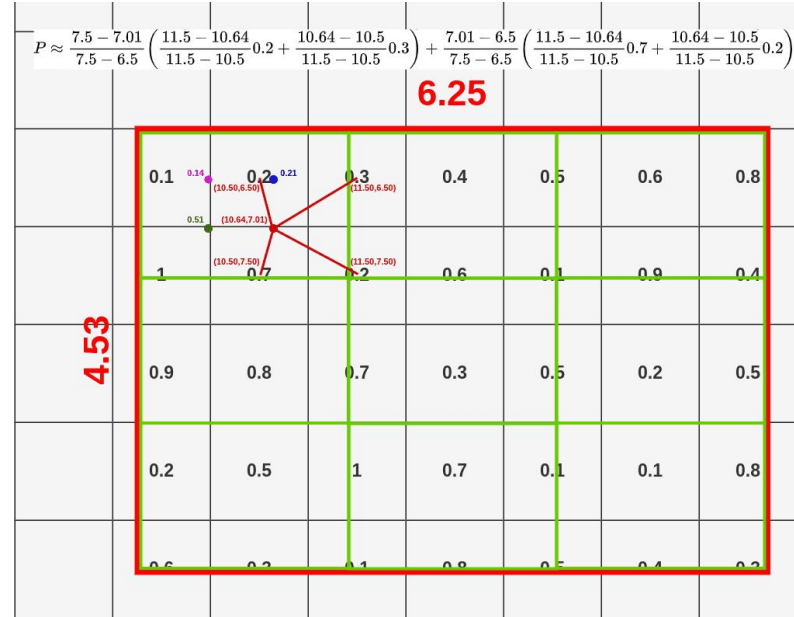
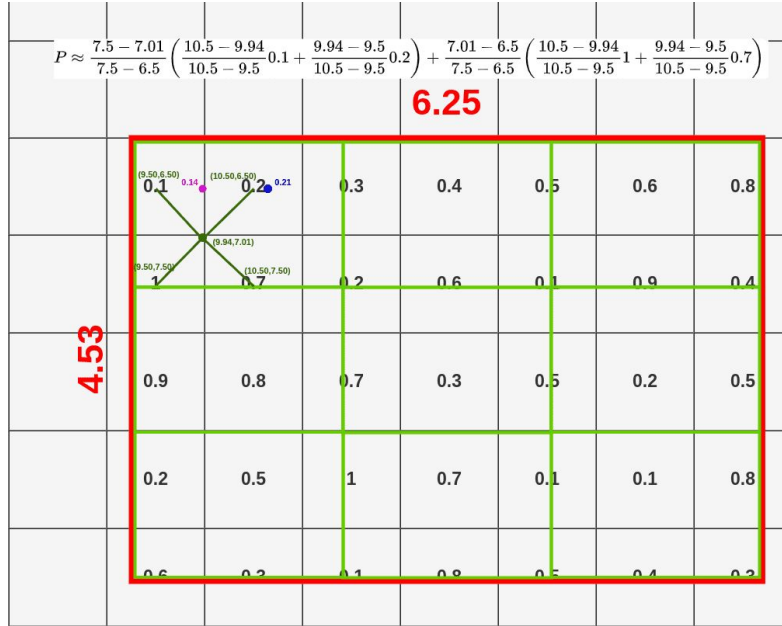
Bilinear Interpolation equation

Bilinear Interpolation

距離越近，權重越大



Bilinear Interpolation



Bilinear Interpolation



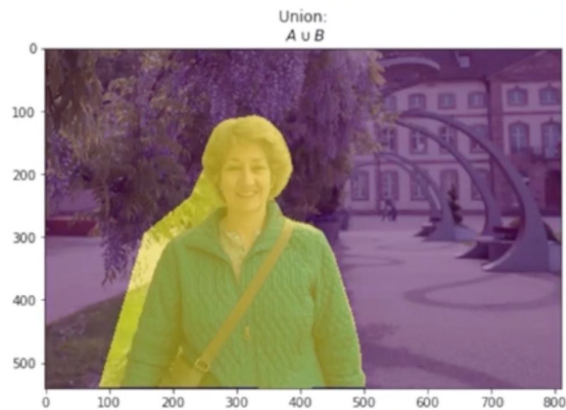
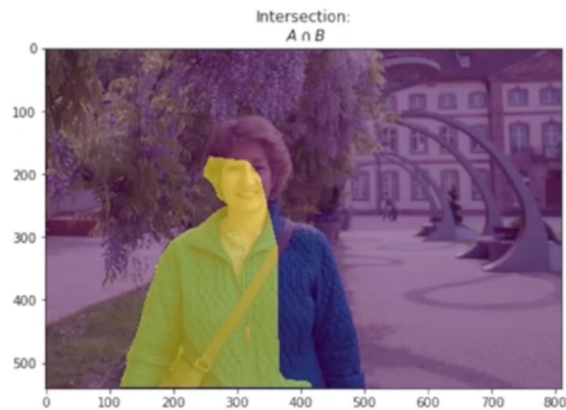
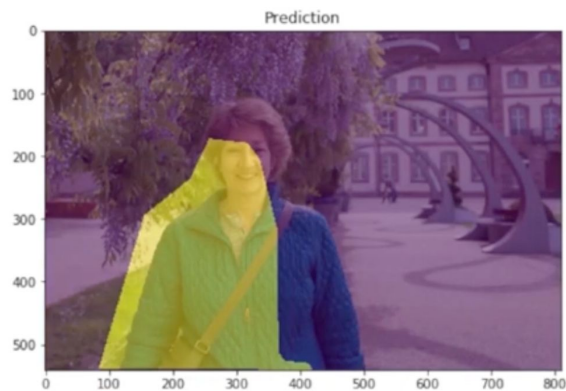
Comparison

	FCN	U-Net	SegNet	Mask R-CNN
架構	全卷積網路, 無全連接層	編碼器-解碼器 + 跳躍連接	編碼器-解碼器 + pooling indices	FPN + RPN + ROI Align + 分割分支
Upsampling Strategy	多層特徵融合(逐元素相加), 上採樣到原尺寸	跳躍連接 + 轉置卷積	Pooling index upsampling	轉置卷積 + ROIAlign
適用分割類型	Semantic	Semantic	Semantic	Instance
分割精度	較低	較高	中等	非常高
小物體表現	較差	較好	較差	較好
計算資源需求	較少	較多	較少	較高

Segmentation Metrics

- Intersection over Union (IOU)
- Mean Intersection over Union (mIOU)
- Pixel Accuracy
- Precision
- Recall
- F1-score

IoU



$$IoU = \frac{TP}{(TP + FP + FN)}$$

mIOU

- The average IOU for all classes:

$$mIOU = \frac{1}{n} \sum_{i=1}^n IOU_i$$

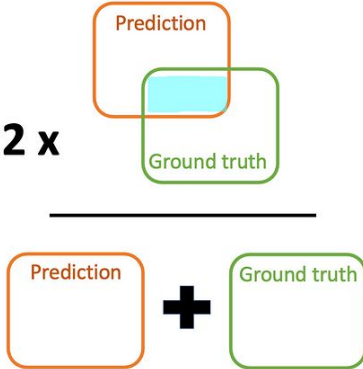
Pixel Accuracy (Accuracy)

- The overall accuracy of the segmentation by the percentage of pixels that are correctly classified
- Pixel Accuracy is not sensitive to **small objects**
- Suitable for overall evaluation of segmentation performance, but not suitable for data with severe class imbalance

$$\text{Pixel Accuracy} = \frac{\text{Number of correctly classified pixels}}{\text{Total number of pixels}}$$

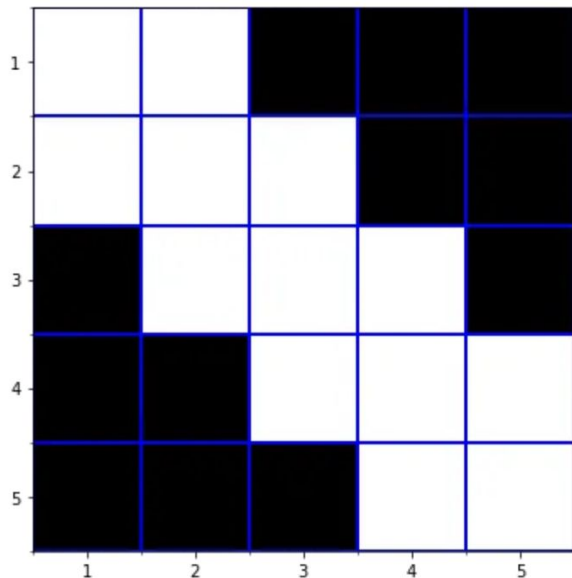
Dice Coefficient

- Dice coefficient ranges from 0 to 1. The higher the value, the more accurate the segmentation result is
- More sensitive to small targets and is suitable for processing unbalanced data

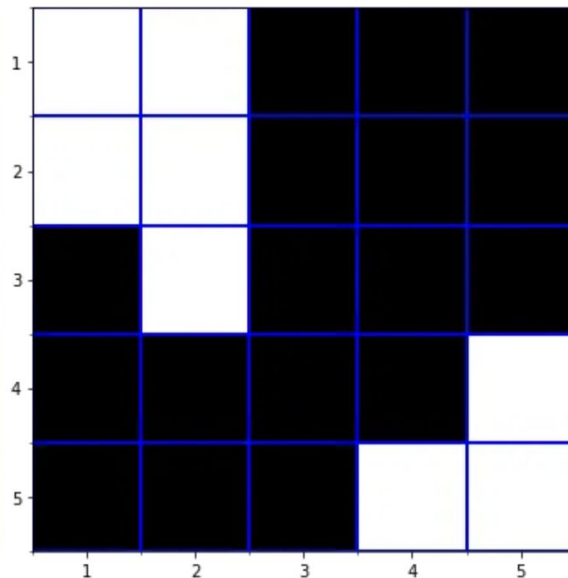
$$\text{Dice} = \frac{2 \times \text{Area of overlap}}{\text{Total area}} = \frac{2 \times \text{Area of overlap}}{\text{Area of Prediction} + \text{Area of Ground truth}}$$


Example

Ground Truth Mask



Prediction Mask



- Precision: 1
- Recall: 0.615
- Accuracy: 0.8
- F1: 0.762
- IOU: 0.615

Implementation

Download notebook at:

<https://github.com/albert831229/nchu-computer-vision/tree/main/113/night>