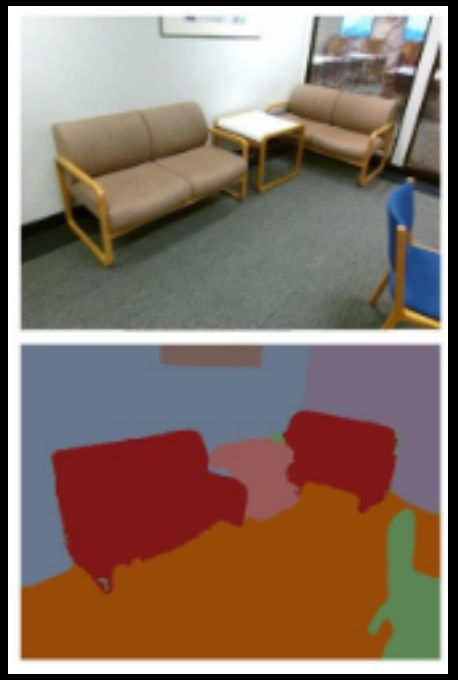
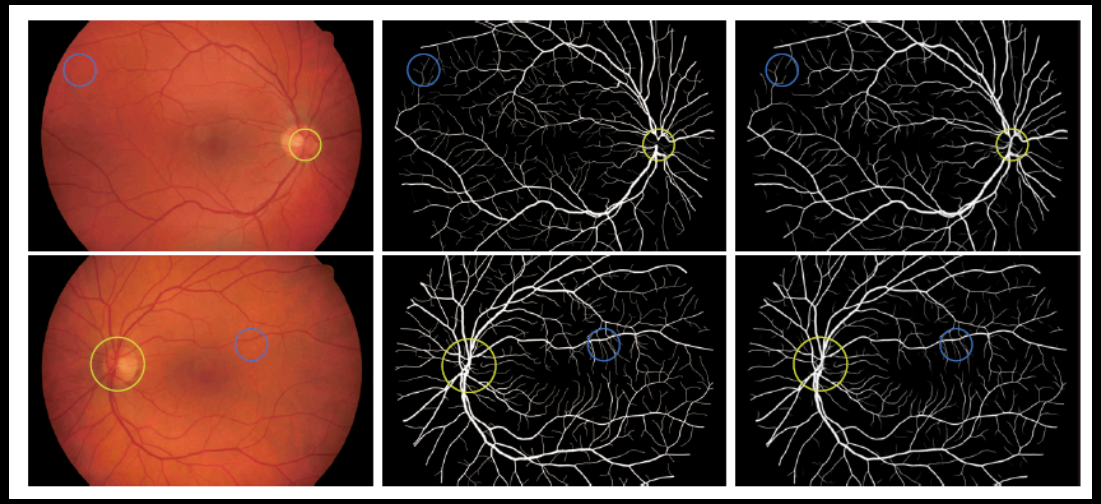
Image - Segmentation

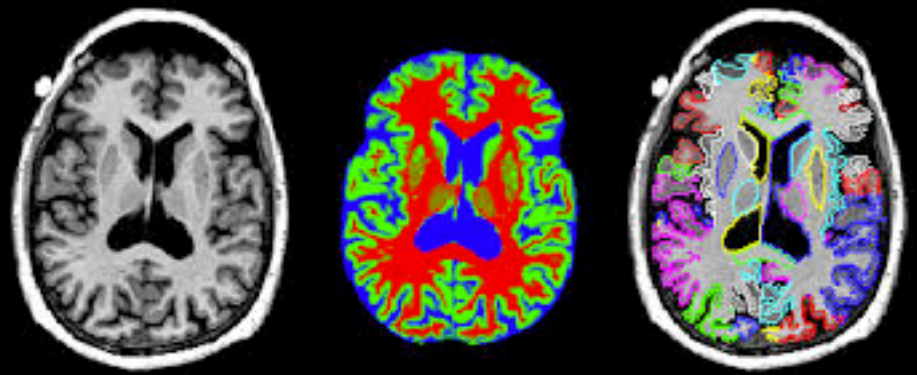
Problem definition :

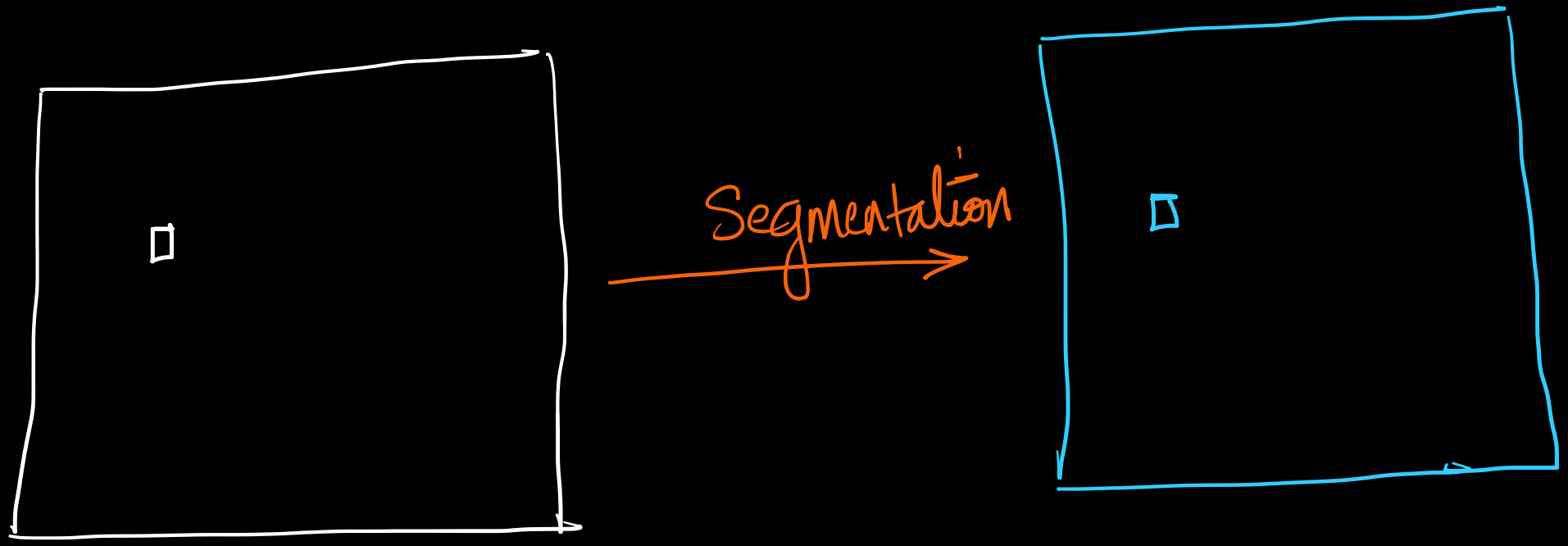


Google AI Blog

Problem definition:







Input - Image
(RGB / Grayscale)

Output - Image
(color - coded)

Datasets:

http://cocodataset.org/#home (like ImageNet)

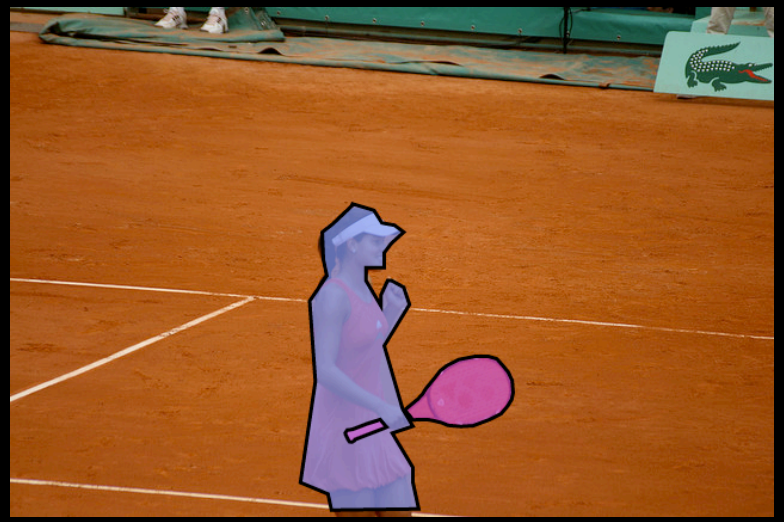
Many domain-specific datasets

"semantic segmentation traffic dataset" https://www.cityscapes-dataset.com/examples/

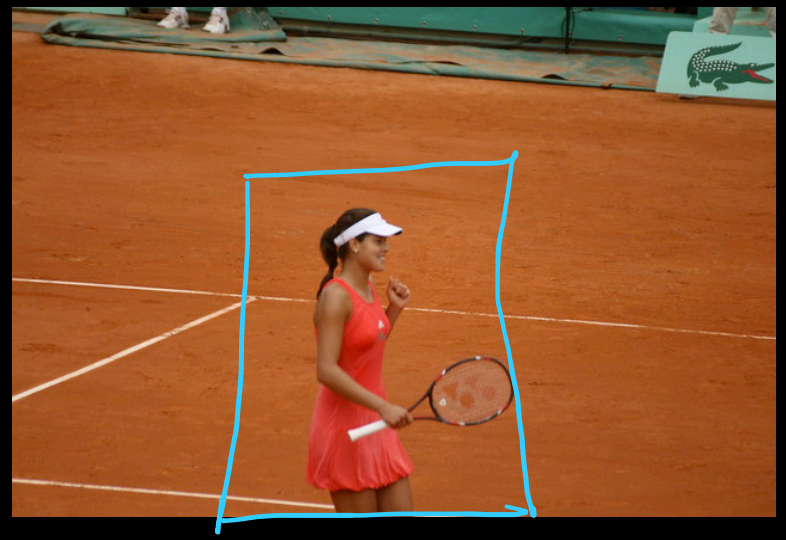
"Kaggle medical segmentation" https://www.kaggle.com/mateuszbuda/lgg-mri-segmentation/version/2

Segmentation vs object Detection:

Input -- Image



Pixel-level Segmentation

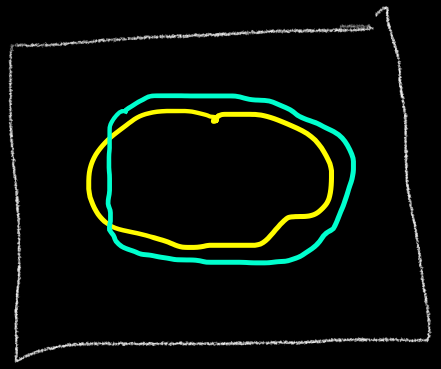


bounding box - OD

Performance Metric

① Average Precision & Recall for each class/color. [COCO]

② Jaccard Sim = $\frac{R_1 \cap \hat{R_1}}{R_1 \cup \hat{R_1}}$ (IOU)



[Cityscapes]

Classical Approaches

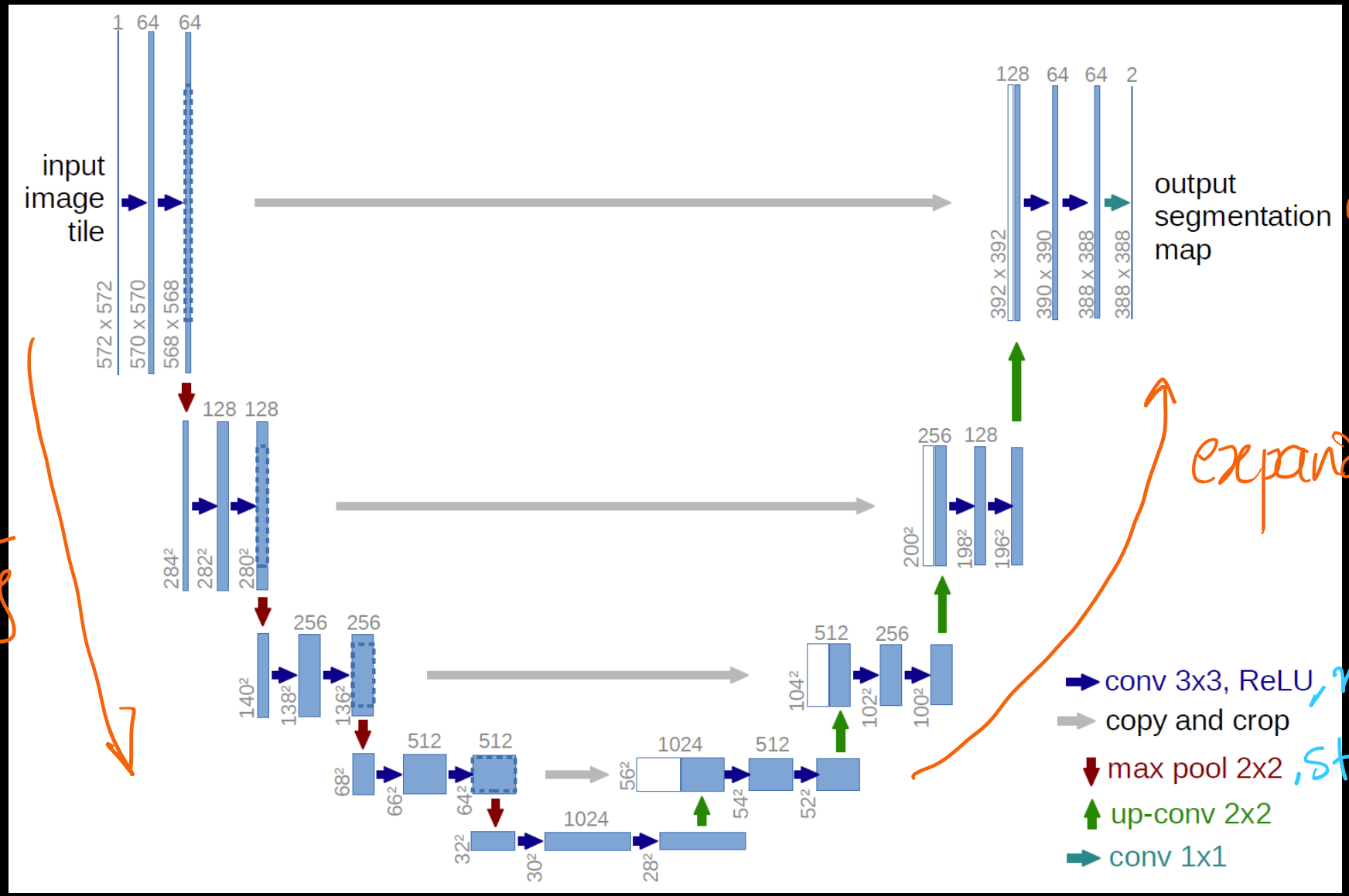
1. Clustering
2. Edge-detection
3. Graph-theoritic (Spectral clustering)

Deep-Learning based methods:

1. UNets
2. Fully convolutional Networks (FCN)
3. Mask R-CNN
4. SegNet

& many-more....

U-Nets: https://lmb.informatik.uni-freiburg.de/people/ronneber/u-net/



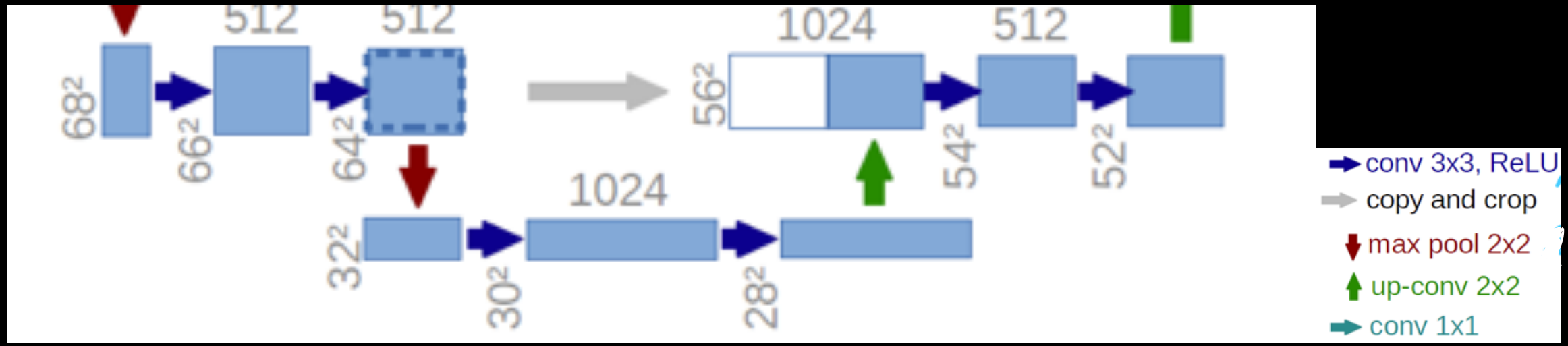
Loss: cross entropy

Contracting path / Encoder

expanding-path / decoder

no padding, stride = 2

APPLIED COURSE



28x28x1024 —upconv 2x2→ 56x56x[512+512]

64x64x512 —crop+copy→

Keras Model :

https://github.com/zhixuhao/unet/blob/master/model.py#L33

```
drop5 = Dropout(0.5)(conv5)

up6 = Conv2D(512, 2, activation = 'relu', padding = 'same', kernel_initializer = 'he_normal')(UpSampling2D(size = (2,2))(drop5))
merge6 = concatenate([drop4,up6], axis = 3)
```

Data Augmentation:

→ shift, Rotate, shear, zoom, ... (problem-specific)

→ grayscale adjustments

→ Keras ImageGenerator

Student Case-study:

https://towardsdatascience.com/understanding-semantic-segmentation-with-unet-6be4f42d4b47

Lots of Segmentation-models:

https://github.com/divamgupta/image-segmentation-keras

Mobile-Seg (Google):

https://ai.googleblog.com/2018/03/mobile-real-time-video-segmentation.html