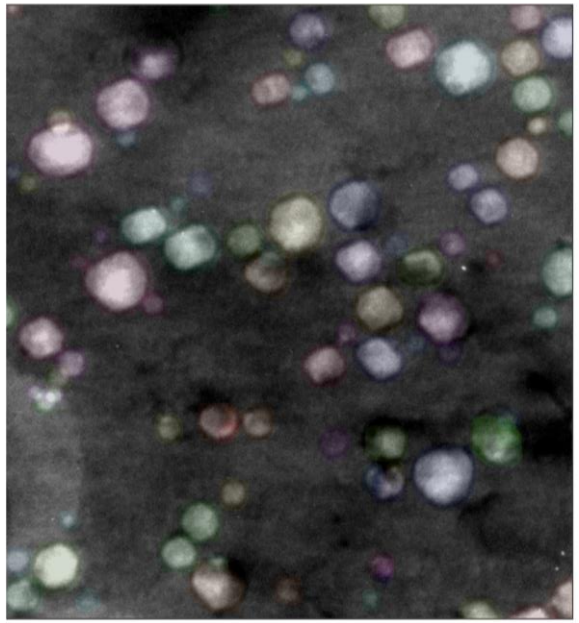
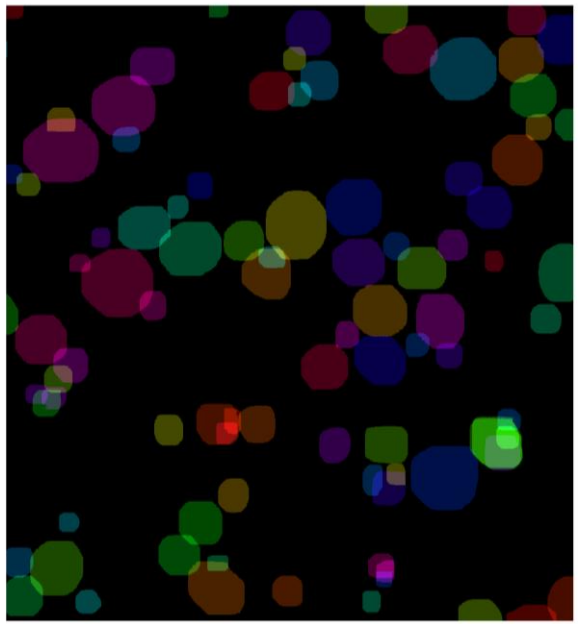
Virtual Summer School: Machine Learning in Electron Microscopy

Examples of DCNN in Electron Microscopy

Tommy Wong

Github:

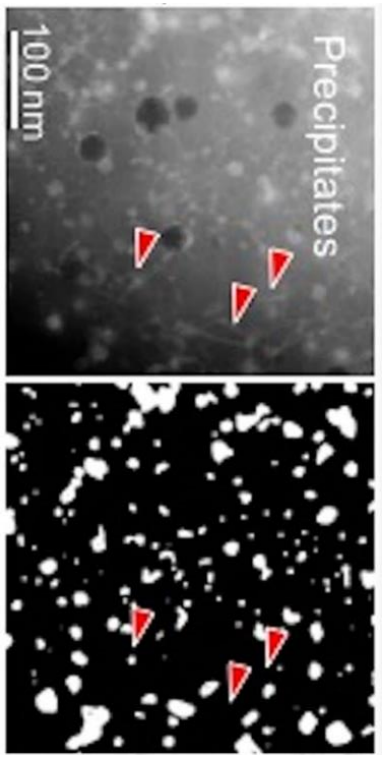




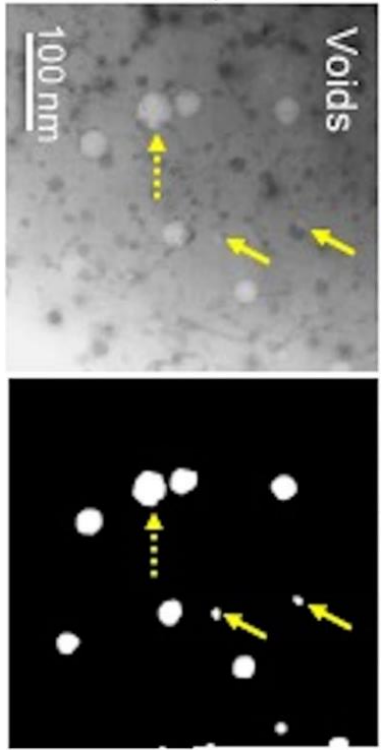
Labeling this image took ~ 1 hr, DL prediction took < 1 min.

Microstructural defects can be labeled for pixel-wise segmentation

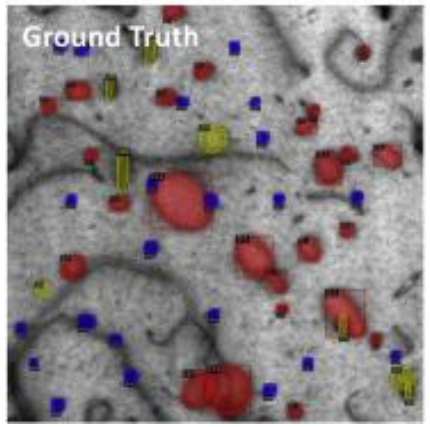
- Segmentation: associating each pixel in an image with a class



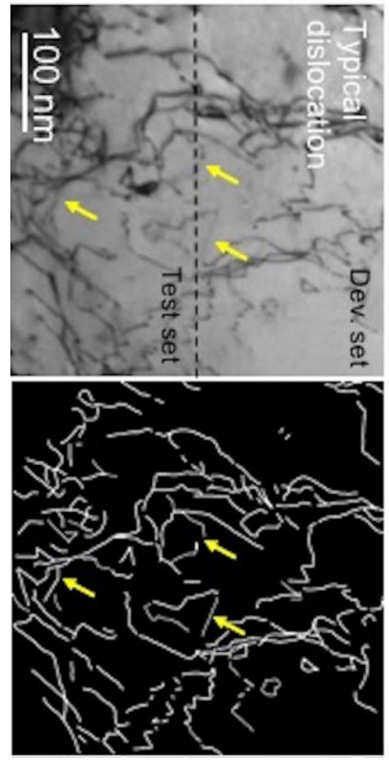
Cavities (bubbles & voids)



Precipitates



Dislocation loops

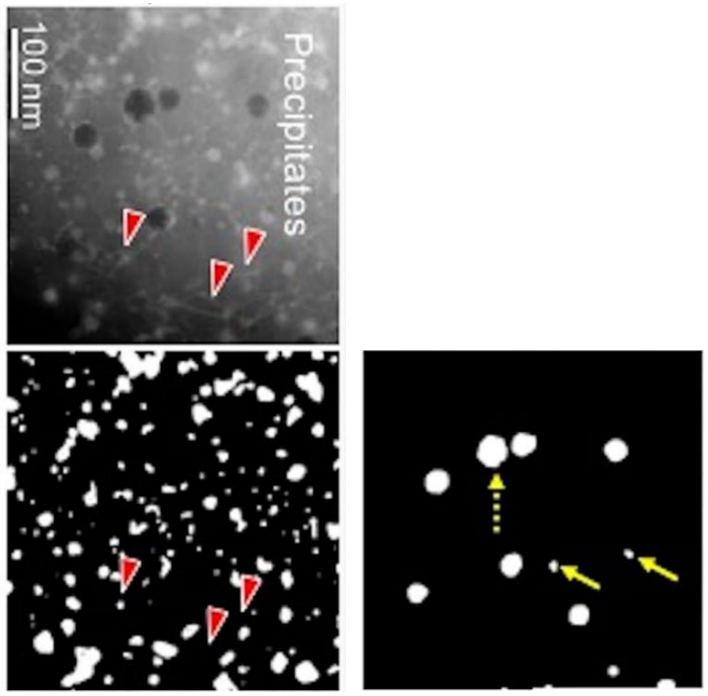


Dislocation lines

Adapted from Roberts et al. 2019 10.1038/s41598-019-49105-0, Jacobs et al. 2022 10.1016/j.xcrp.2022.100876

Different labeling systems are required for different segmentation algorithms

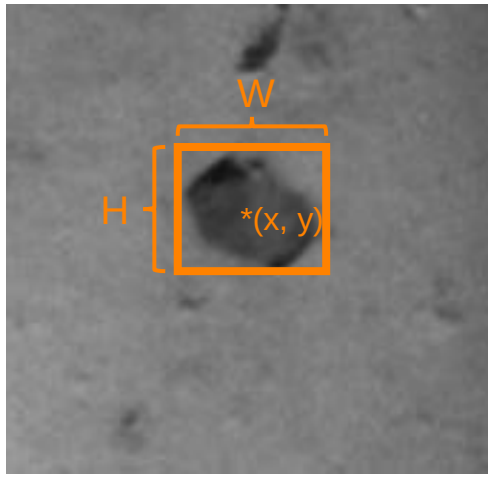
- **Important: labels are usually either 0 or 1**



Precipitates

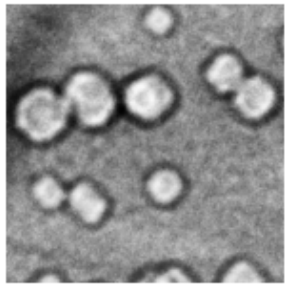
Voids

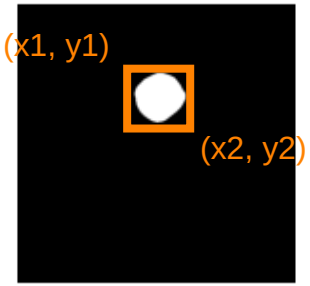
Semantic segmentation: one-hot encoding (U-Net)

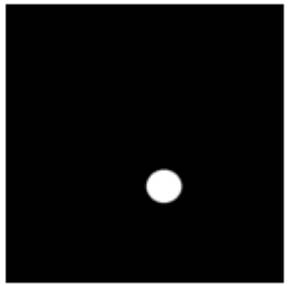


[Class x y W H]

Object identification: bounding box (YOLO)







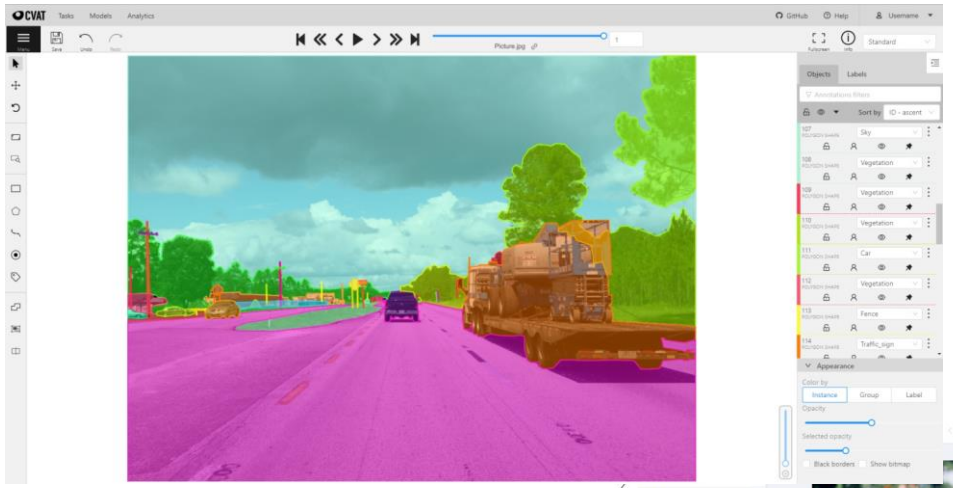
…

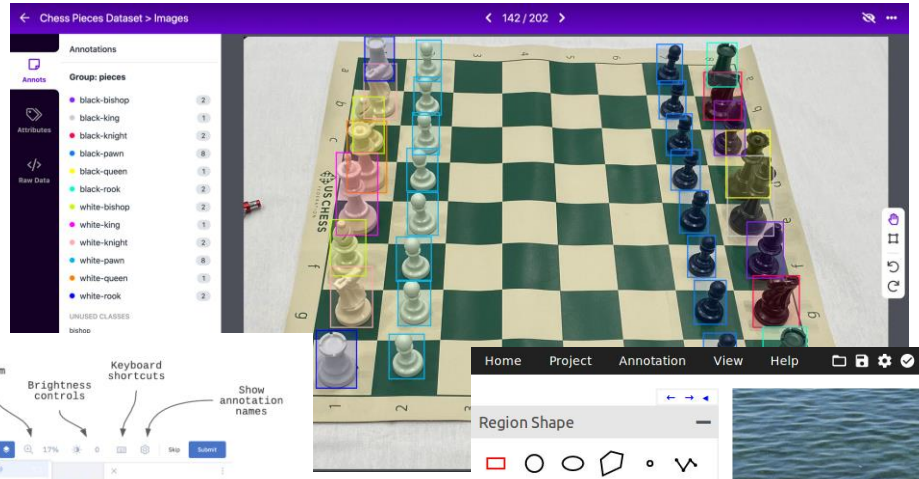
Dictionary{
'boxes': [x1 y1 x2 y2]
'labels': class
'masks': feature mask
}

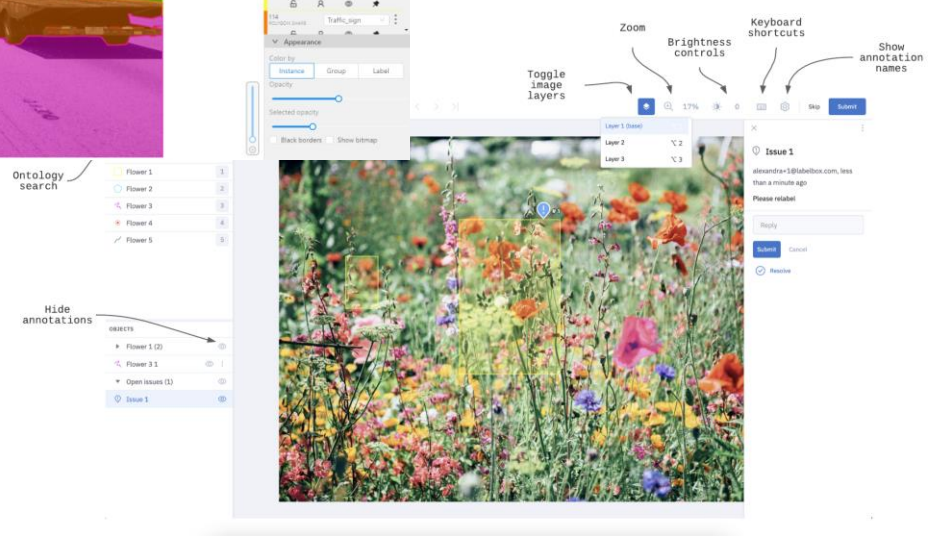
Instance segmentation: label encoding (Mask R-CNN)

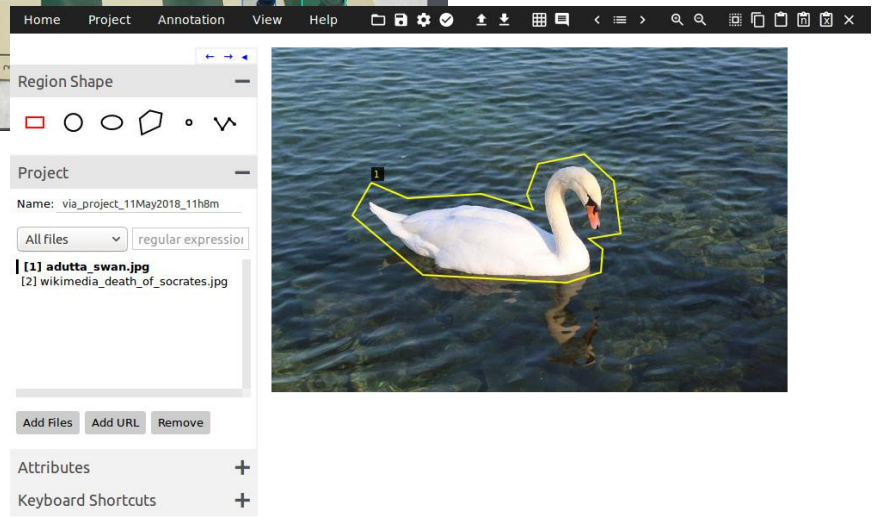
Adapted from Roberts et al. 2019 10.1038/s41598-019-49105-0, Yang et al. 2011 10.1038/srep00190

Web-based GUI tools are used for labeling









Labeling using Computer Vision Annotation Tool (CVAT)

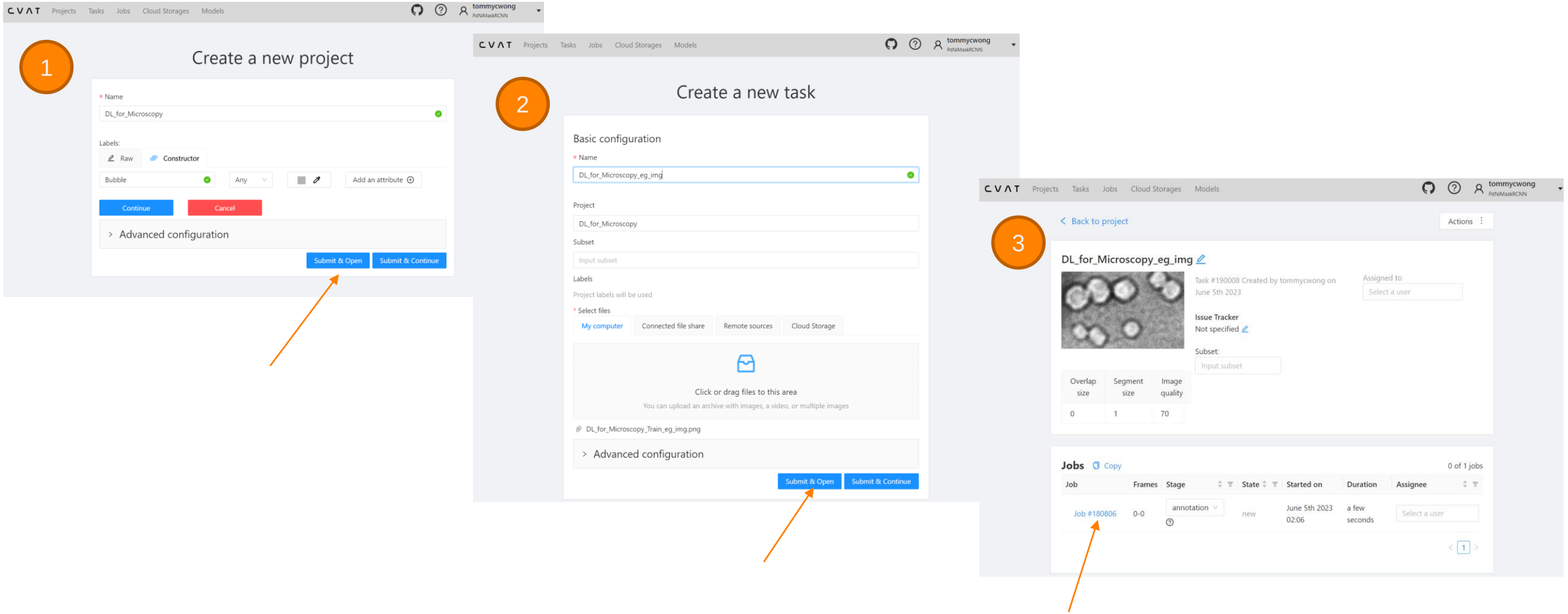
cvat.ai

Documentation:
github.com/TaSeeMba/cvat/blob/master/cvat/apps/documentation/user_guide.md

CVAT labeling workflow

Important: before labeling, ensure all image data have the same dimensions e.g. 1024x1024

Creating a project and labeling tasks

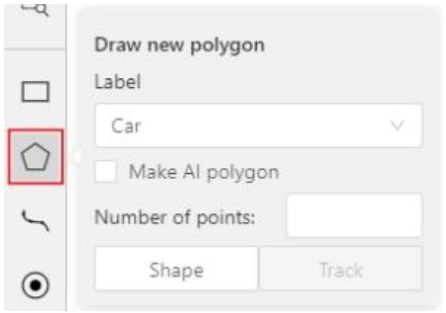


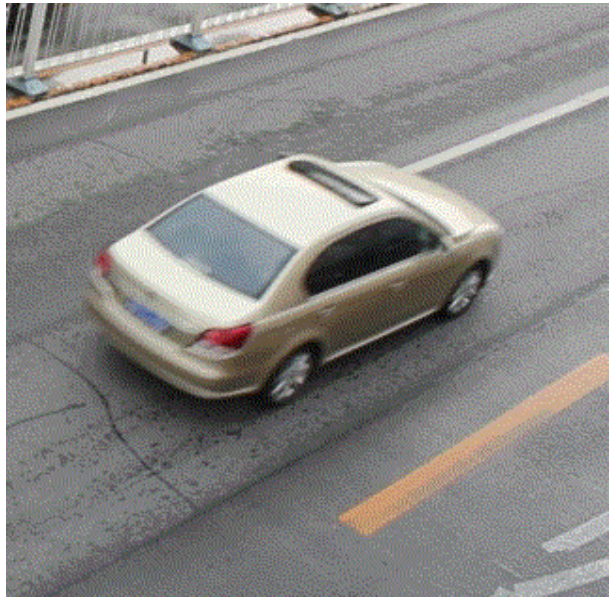
Labeling using polygon and ellipse tools

Remember to click **Save**

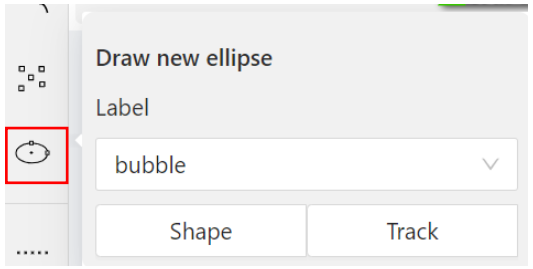
Polygon tool

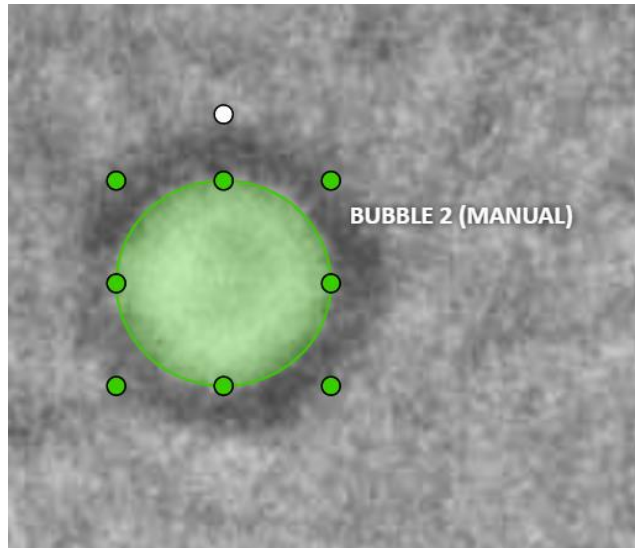
- Hold **Shift** to draw





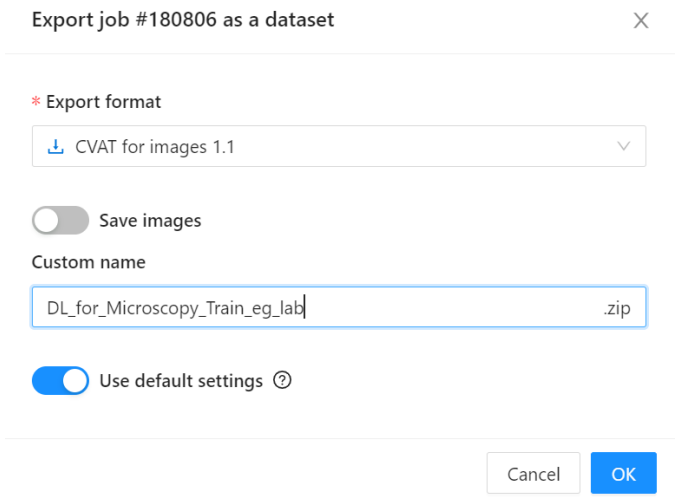
Ellipse tool





Exporting and parsing labels

Export as .xml



Parsing labels using Python

```
get_imgs(train_img_names)

parse_anno_file(xml, train_img_filename)

get_unet_mask(annos)

get_maskrcnn_mask(annos)

get_maskrcnn_dataset(images=train_imgs,
labels=maskRcnn_masks)
```

```
<image id="0" name="DL_for_Microscopy_Train_eg_img.png" width="512" height="512">
  <ellipse label="Bubble" source="manual" occluded="0" cx="291.95" cy="334.99" rx="32.35" ry="30.94" z_order="0">
  </ellipse>
  <polygon label="Bubble" source="manual" occluded="0" points="282.22,131.08;289.47,135.54;295.61,142.23;300.07,1
  </polygon>
```

Additional notes on labeling

- Features typically should have a convex mask
  - Concave masks are likely occluded convex masks
- Don’t leave holes between multiple overlapping masks
- Keep in mind output files: different parsing scripts needed for .xml, .json, etc.

Data Augmentation Techniques

Github:

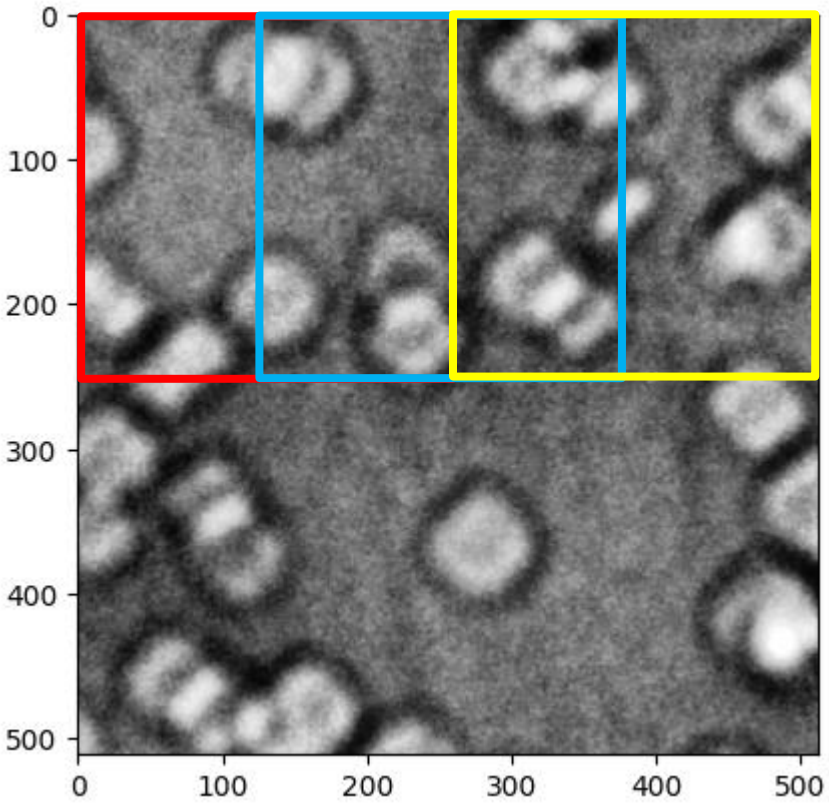
THE UNIVERSITY OF
TENNESSEE
KNOXVILLE

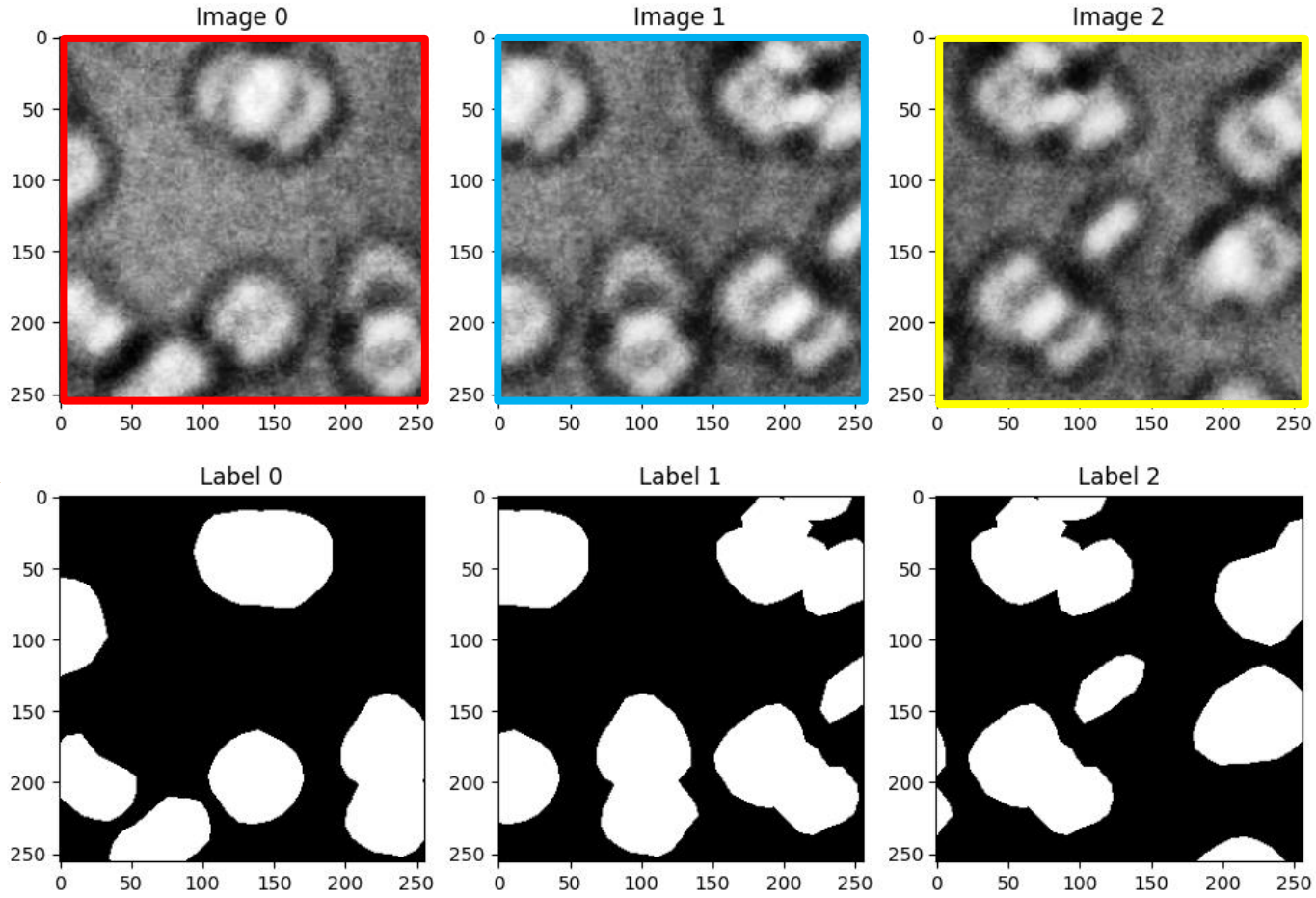
Rationale: more varied data → less overfitting

Electron microscopy experiments yield (relatively) small datasets

∴ requires augmentation

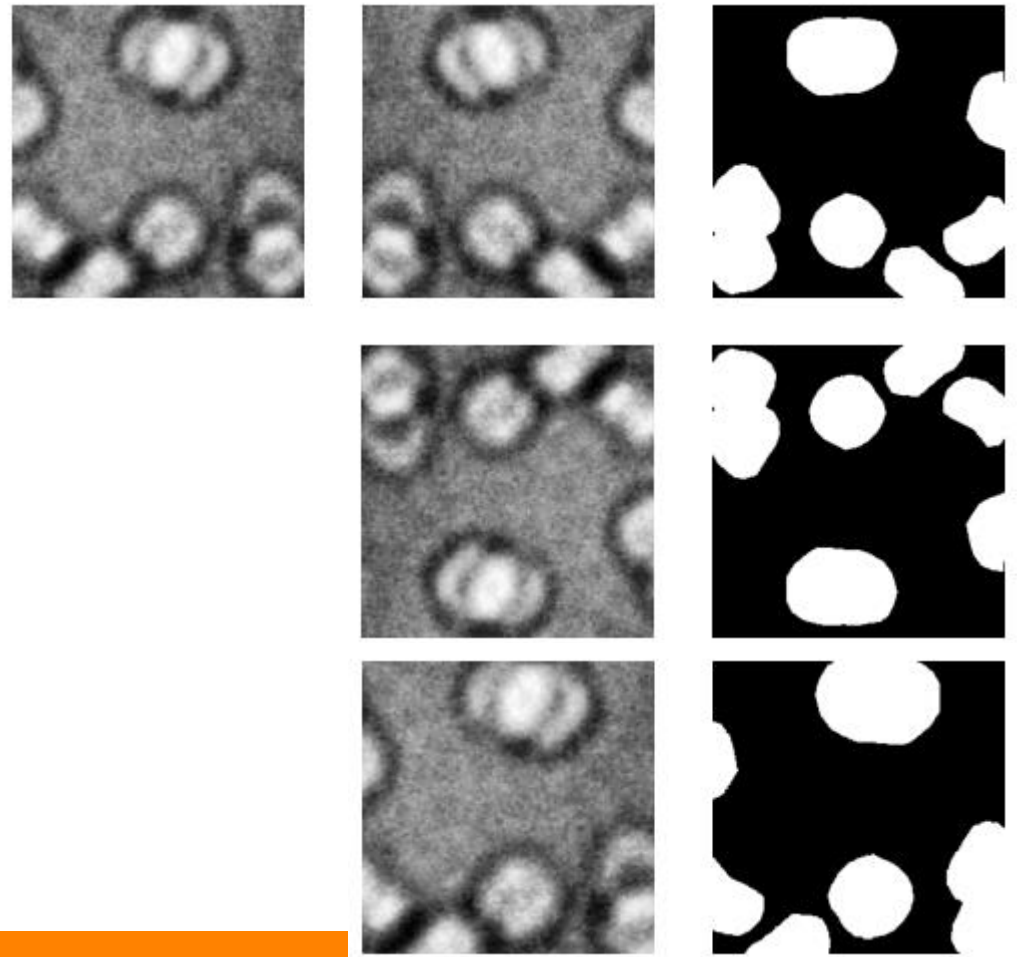
Use a sliding window cropper to enhance the dataset





Rotate, flip, resize the image to increase variability

Original



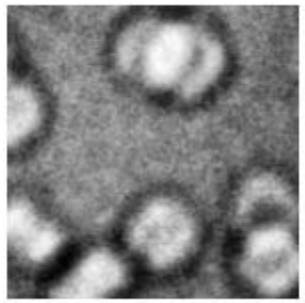
Flipped/Mirrored

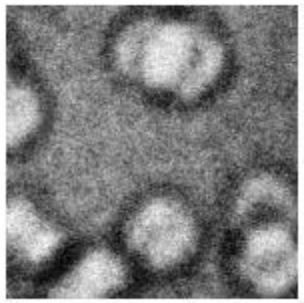
Rotated 180 degrees

Zoomed in

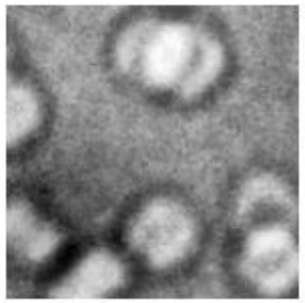
Adding noise to simulate noises during imaging

Original

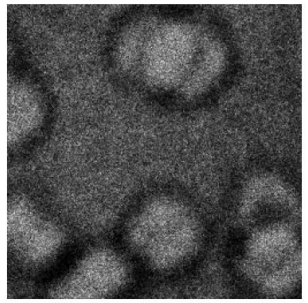




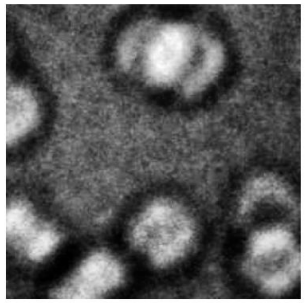
Gaussian



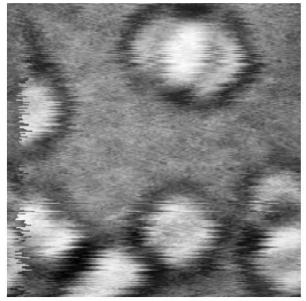
Background noise



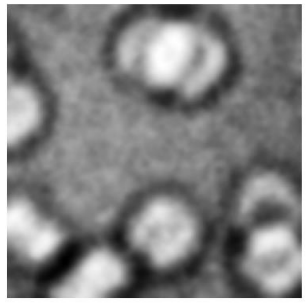
Poisson



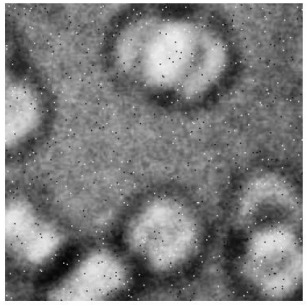
Contrast



Jitter



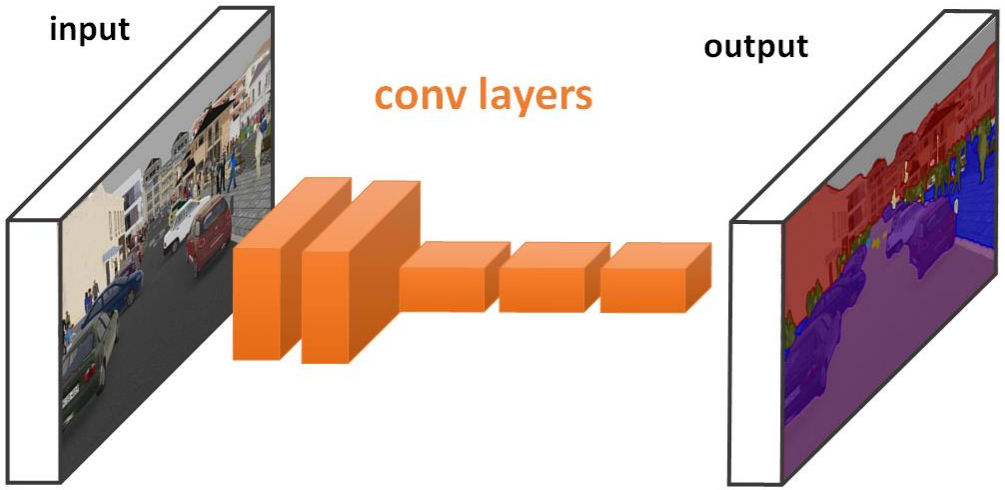
Blur



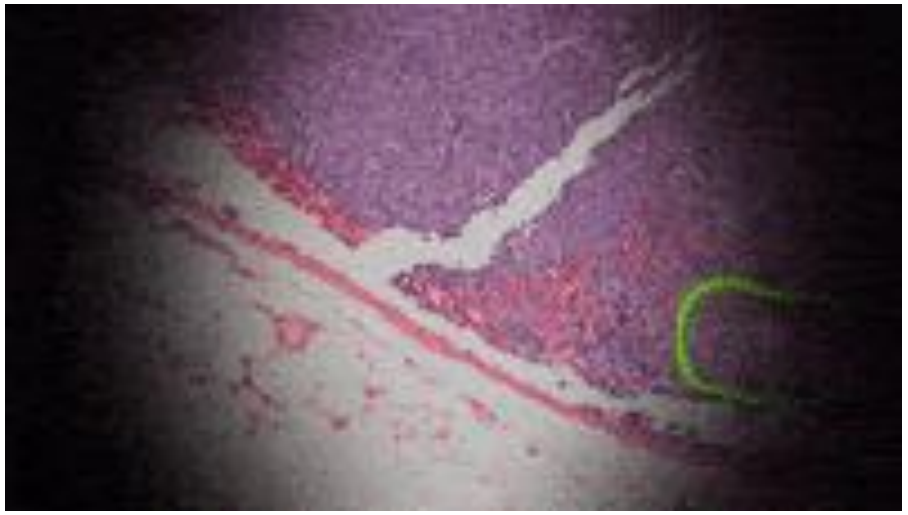
Salt & pepper

Convolutional Neural Networks (CNN)

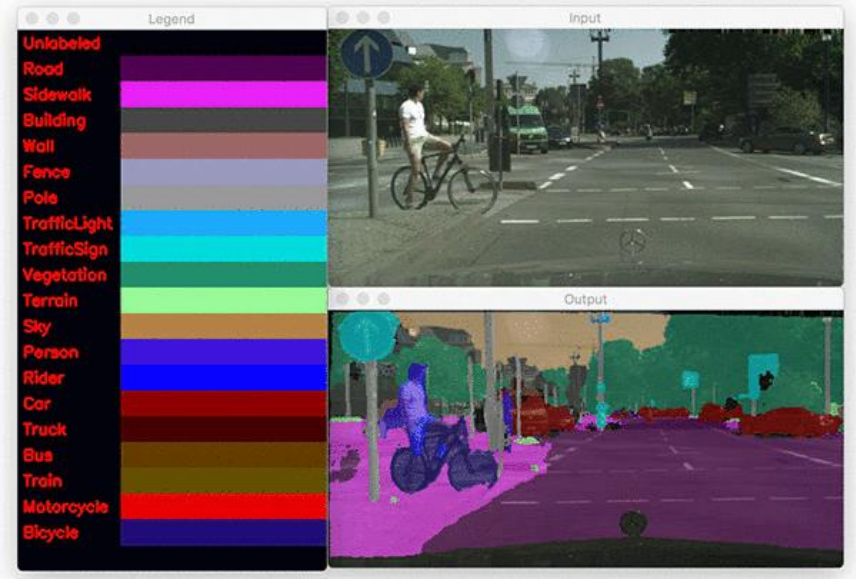
THE UNIVERSITY OF
TENNESSEE
KNOXVILLE



Semantic segmentation assigns class labels to each pixel

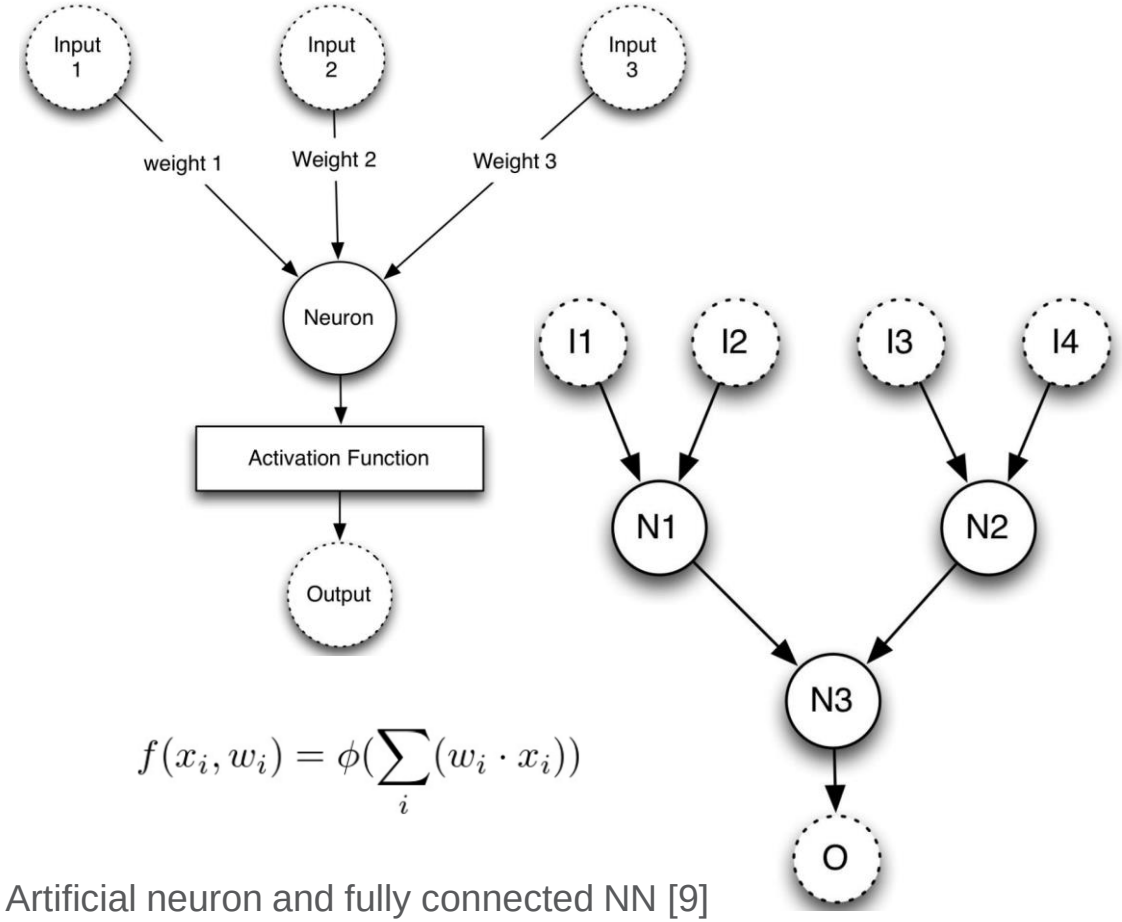


Augmented reality microscope *[4]*



Semantic segmentation of street views *[2], [3]*

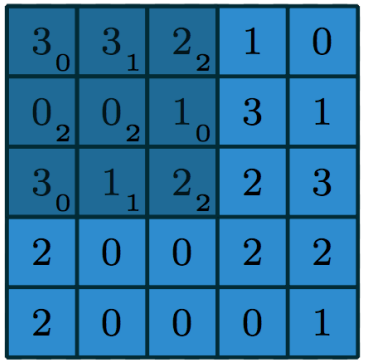
Convolutional Neural Network (CNN)

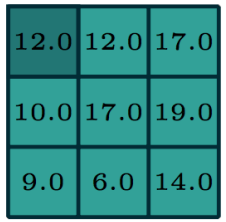


$$f(x_i, w_i) = \phi(\sum_i (w_i \cdot x_i))$$

Artificial neuron and fully connected NN [9]

Convolutional Neural Network (CNN)

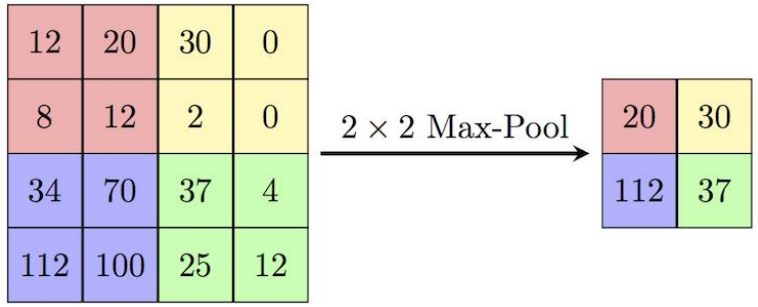


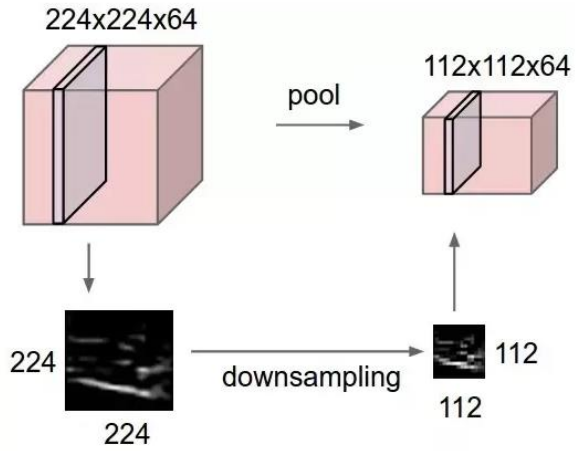


www.cs.toronto.edu/~lczhang/360/lec/w04/convnet.html

Normal convolution kernel

$$(3 * 0 + 3 * 1 + 2 * 2) + (0 * 2 + 0 * 2 + 1 * 0) + (3 * 0 + 1 * 1 + 2 * 2) = 12$$





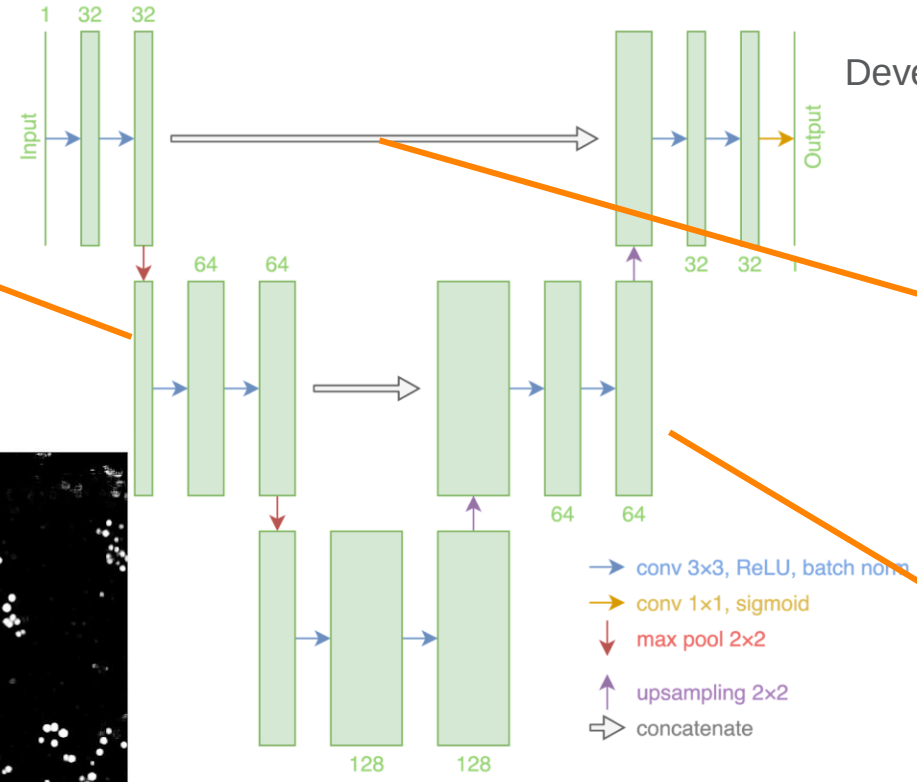
Maxpooling

https://computersciencewiki.org/index.php/Max-pooling_/_Pooling

Semantic segmentation w/ U-Net

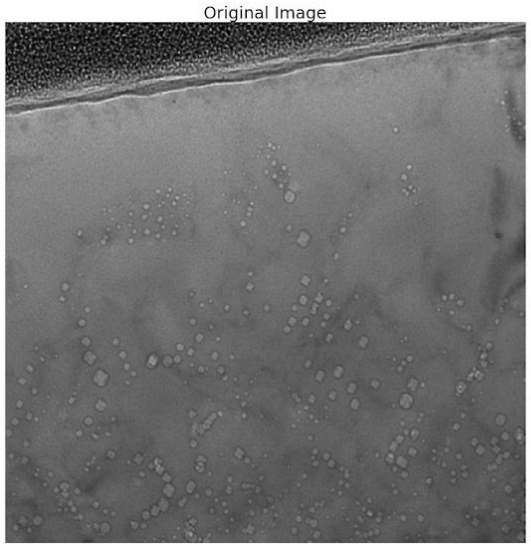
Look for salient features

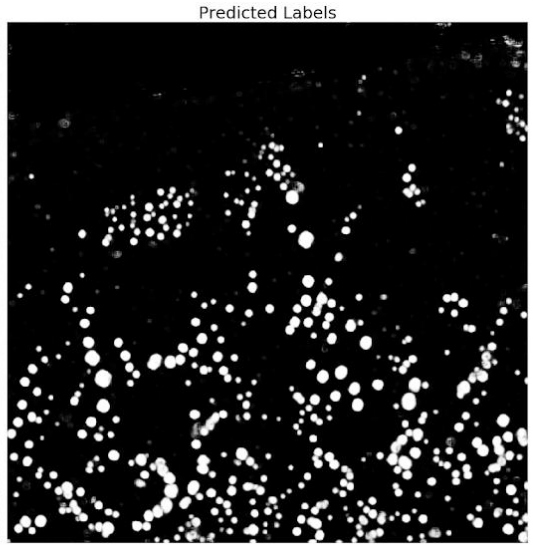
Developed by Ronneberger et al. (2015) [11]



Preserve “big picture”

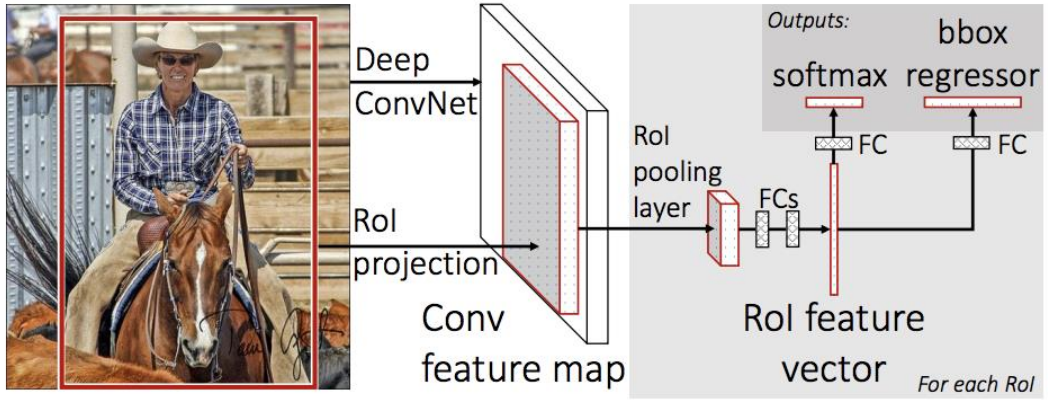
Expand back to original size

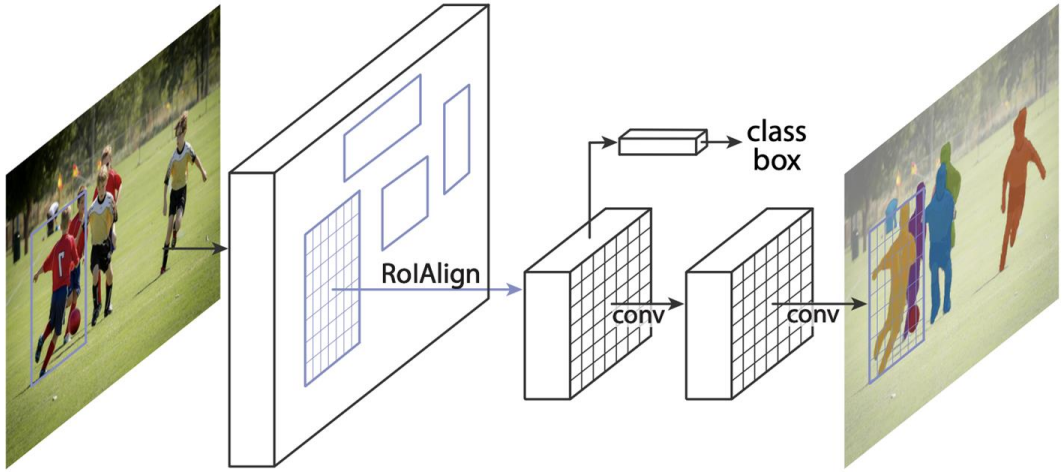


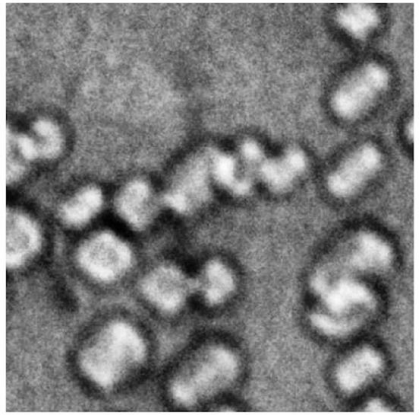


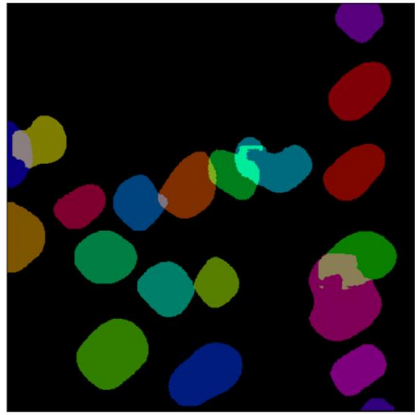
U-Net demonstration

Instance segmentation w/ Mask R-CNN









Mask R-CNN demonstration