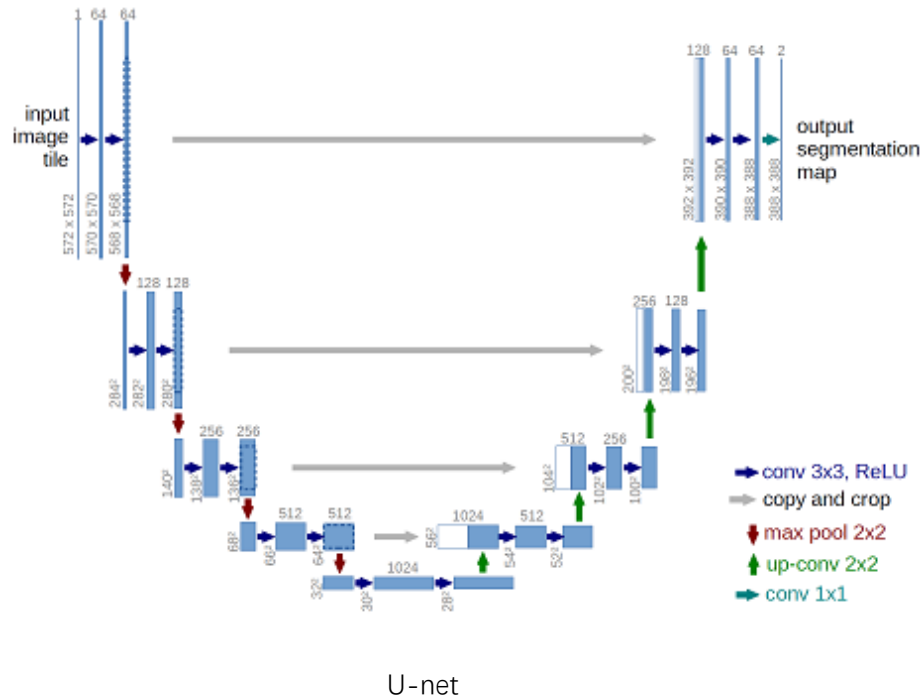


I use a traditional U-net with half kernels to train and predict based on keras-tensorflow.  
 U-net structure:



U-net

Pipelines:

- (1) Fill origin label images and make labels look like solid ellipses
- (2) Resize all images from (800,540) to (160,112)
- (3) Use the U-net to train and predict
- (4) Resize the prediction images from (160,112) to (800,540)
- (5) Use functions from opencv to predict Minimum external rectangle
- (6) Calculate the x, y, a, b and angle