**Assignment 1**

Wanyi Su

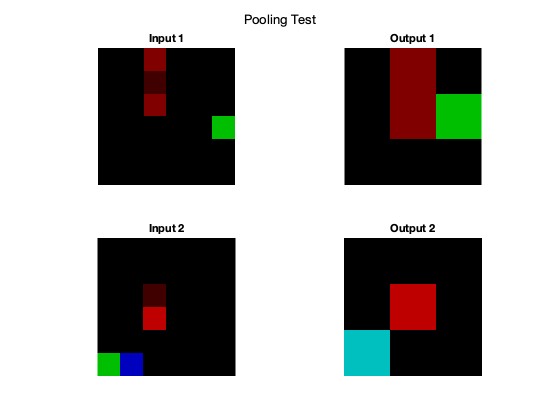
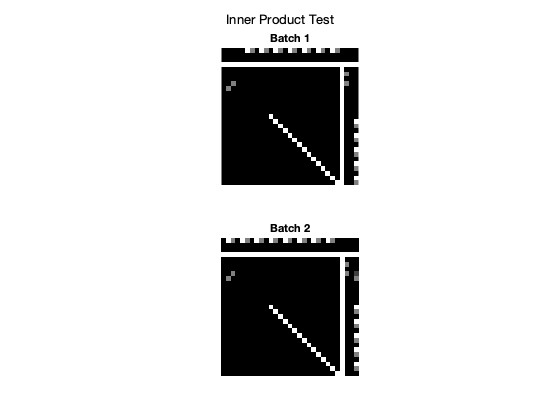
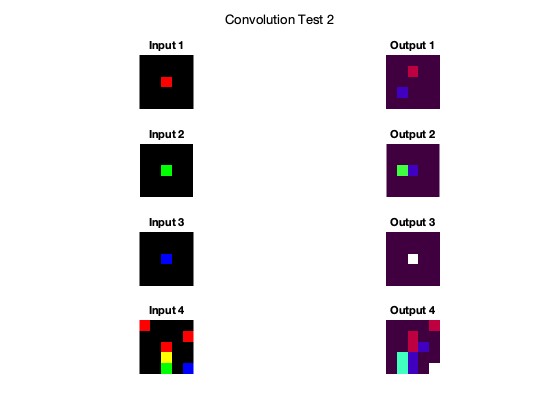
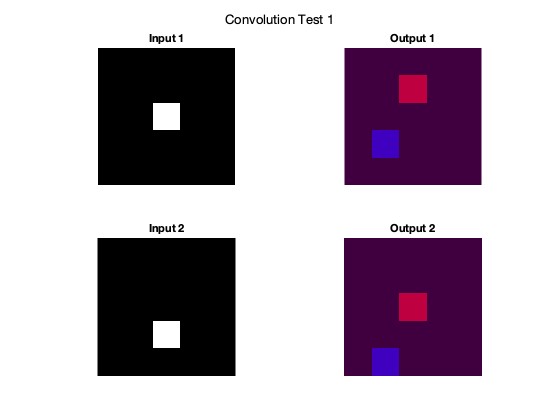
Student #: 301445656

***Use the extra 3 free late days.***

**Part 1**

Codes are in corresponding MATLAB files.

Figures:



**Part 2**

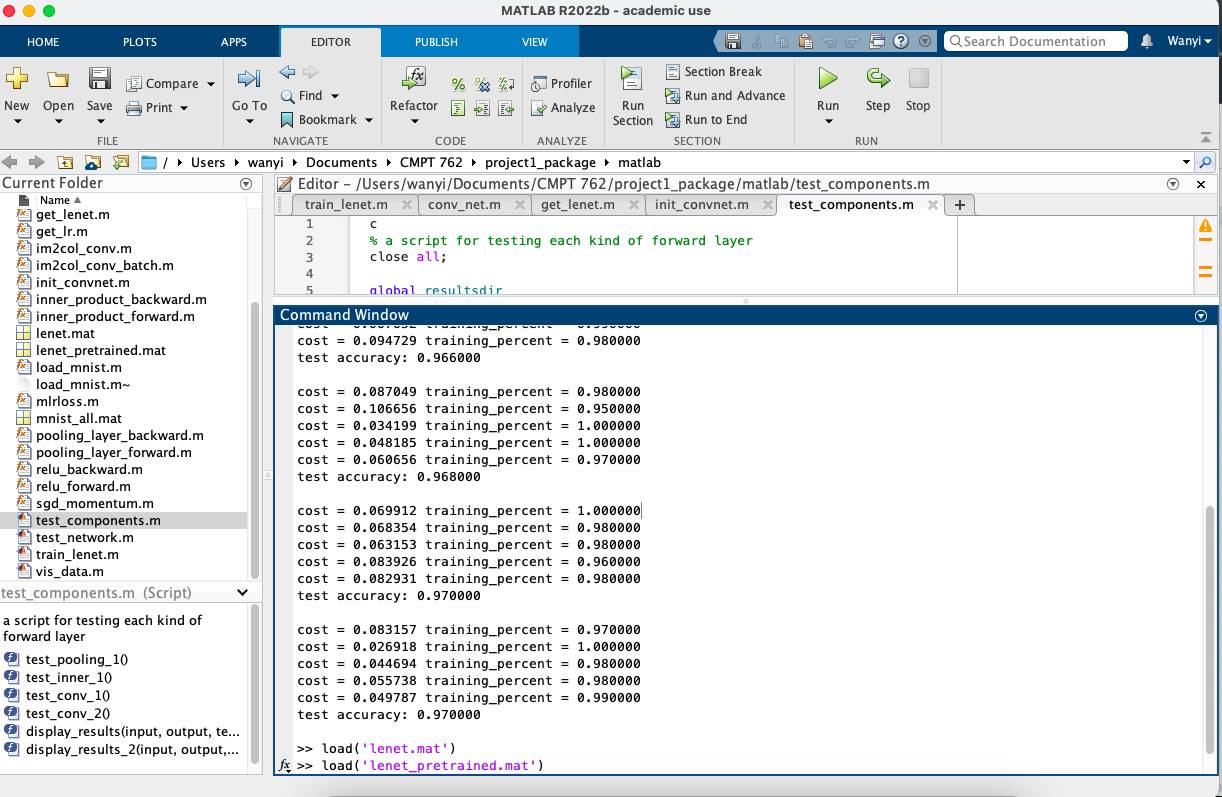
Codes are in corresponding MATLAB files.

**Part 3**

**Q3.1**

Codes are in corresponding MATLAB files.

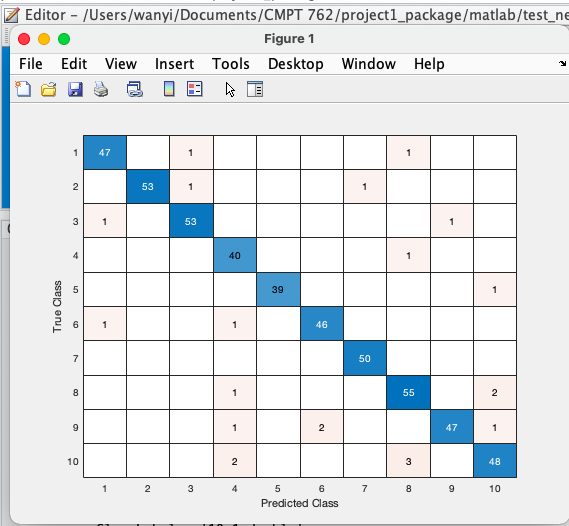
Test accuracy is 0.97 as below.



**Q3.2**

We have ten classes here (“0” to “9”). The top two confusing pairs of classes are pair of class 8 (number is “7”) with class 10 (number is “9”), which has three “7” detected as “9” and two “9” detected as “7”. Another confusing pair is the pair of class 6 (number is “5”) with class 9 (number is “8”), which has two “5” detected as “8”. The main reason for confusing detection is that the pairs of numbers have similar shapes (edges or corners). When the images are input into the model after resizing, some information may lose or add because of resizing and the model can mistake one number for another when detecting numbers. The different writing styles can also result in the errors.

Confusion matrix chart is shown as screenshot below:



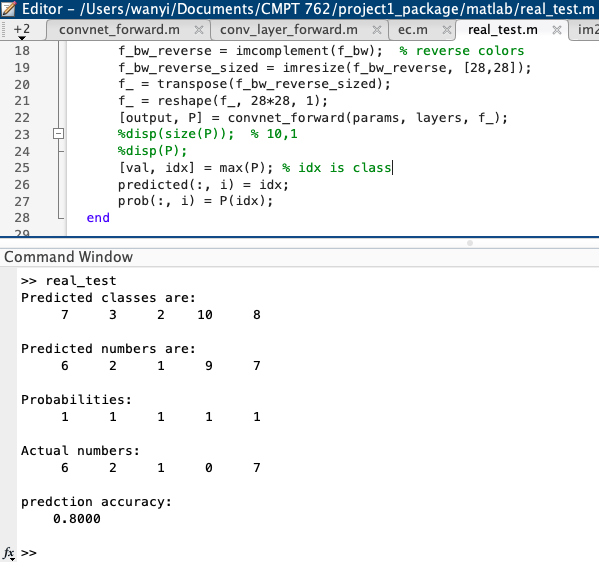
**Part 3.3**

Code is in **real\_test.m** file.

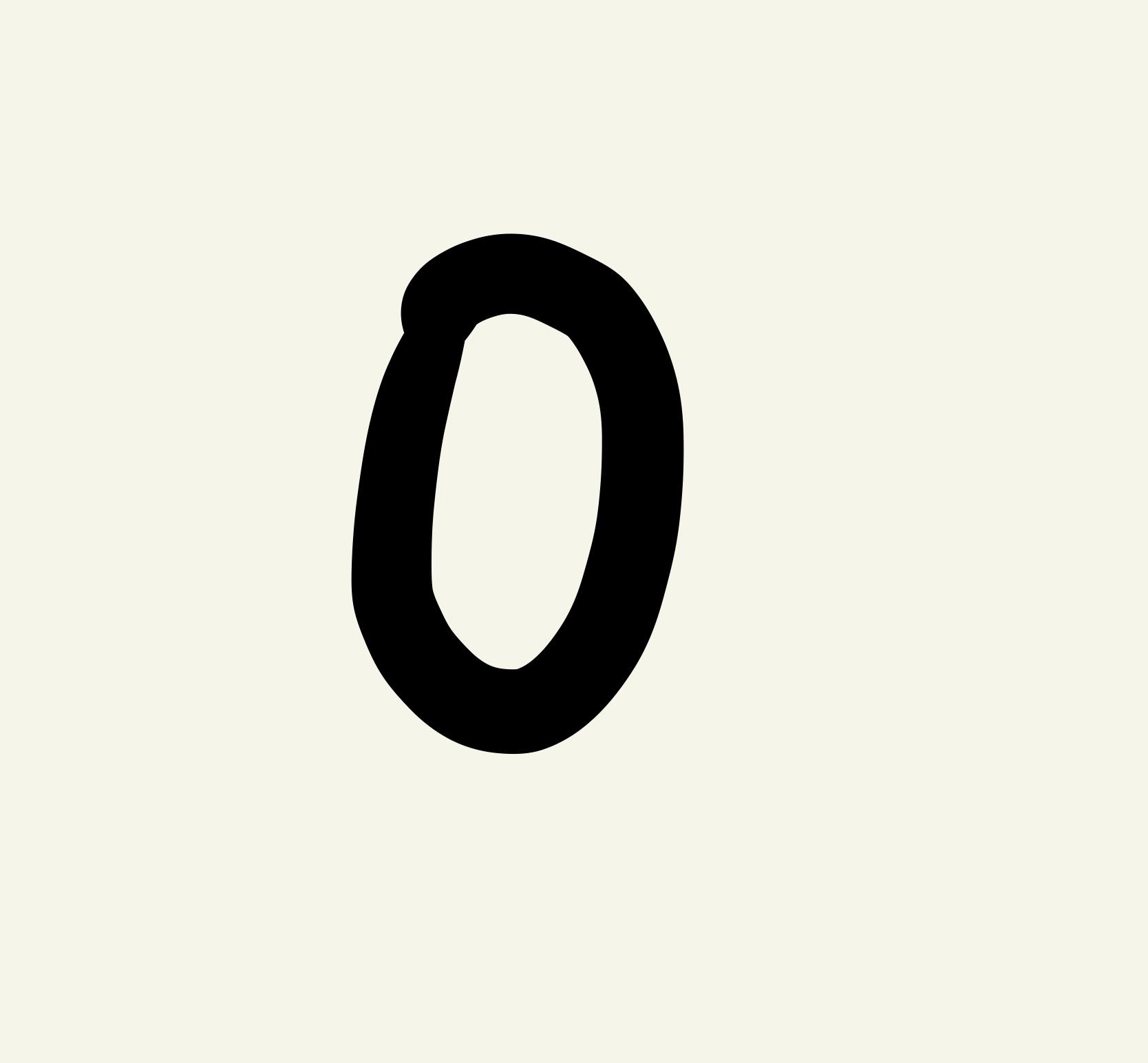
First resize the .jpg figures and turn them into black background and white characters pictures, then do the prediction. Finally, we correctly classify 4 out of 5 numbers I wrote, and the accuracy is 0.80.

The model misclassified number 0 as “9”, which is understandable, since the two numbers are similar. The model might detect some tiny pixels in the bottom right side of “0” and recognize it as “9”.

Screenshot of predicted numbers, corresponding probabilities, predicted accuracy are as follows:



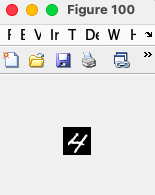
Input handwritten numbers are as follows:

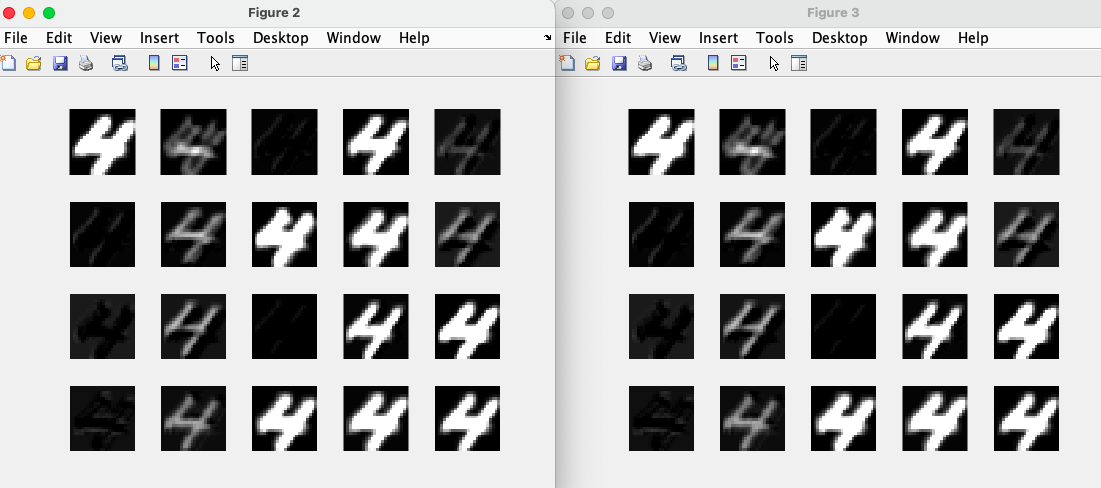


**Part 4**

**Part 4.1**

The 20 images from each of two layers are as below.





**Part 4.2**

The feature maps of 20 images from each of layer 2 as well as layer 3 and original image are quite similar in shape, but compared to original image, feature maps are blurred and are in different blur levels in 20 channels.

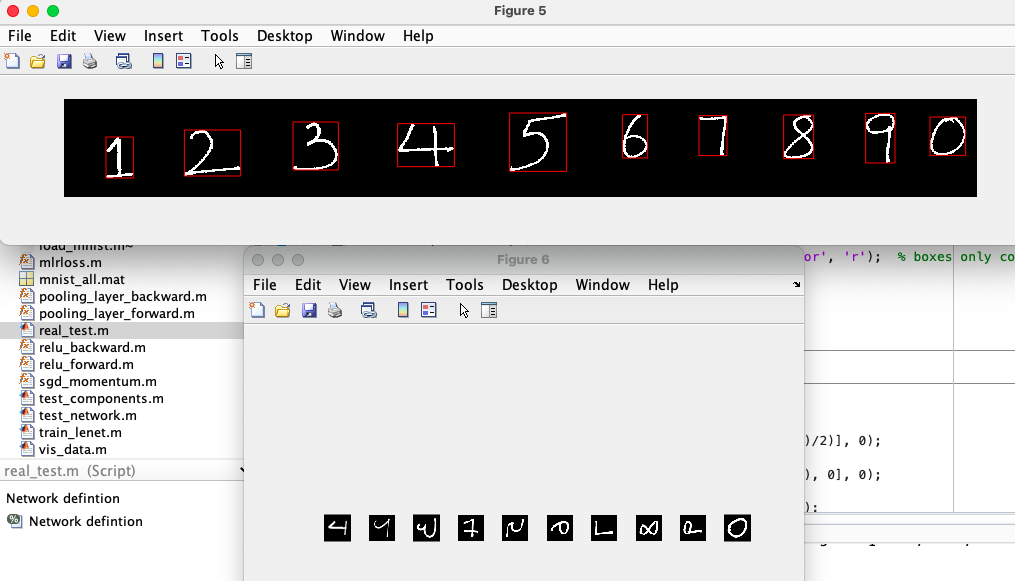
The reason can be as follows.

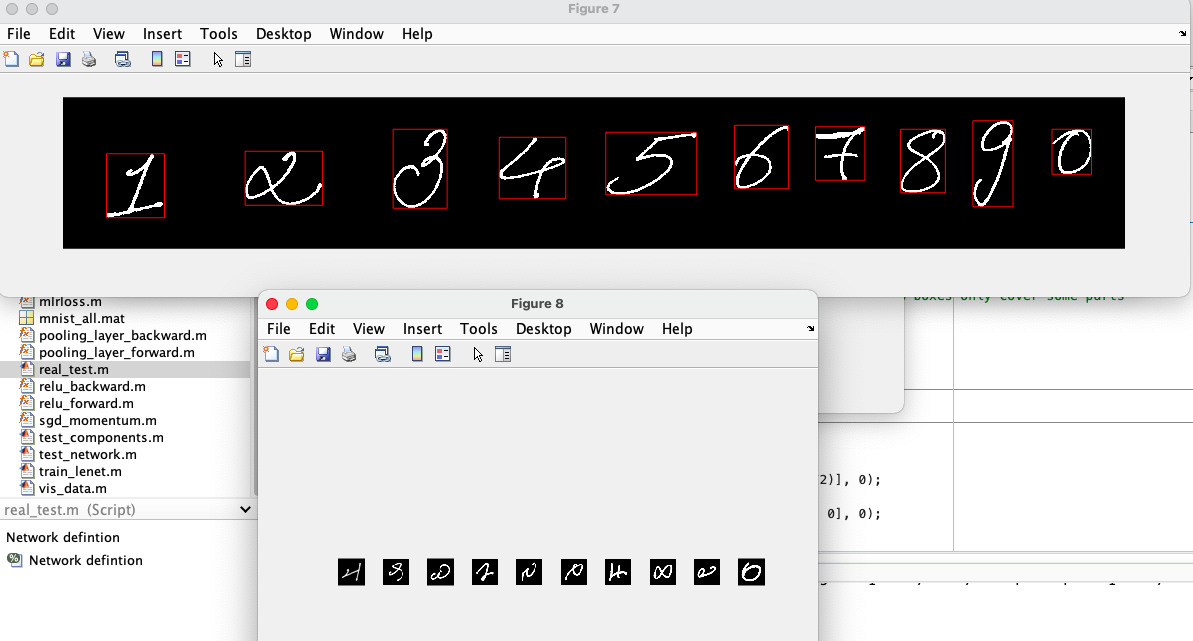
As we filter the original image into 20 channels in CONV and RELU layers, the feature maps in each channel can be blurred than the original image and have different light levels, while shapes remain similar since CONV layer can capture important features of the original image.

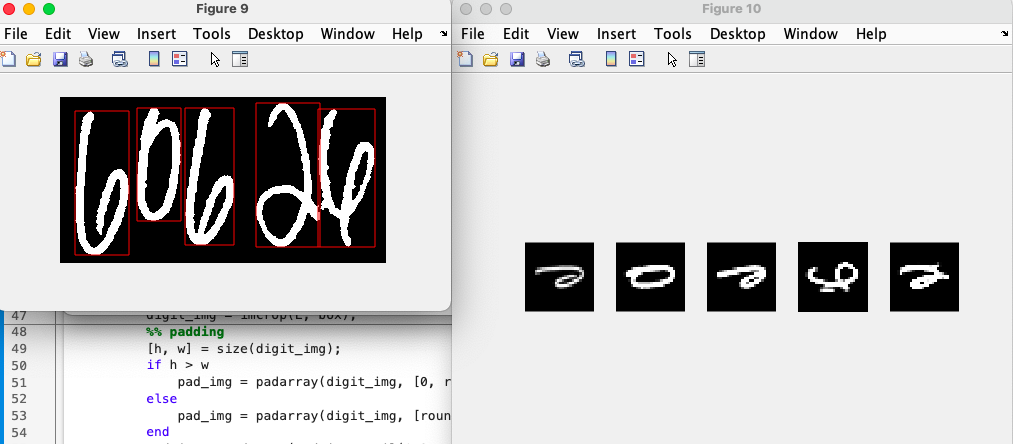
When the original image input first comes through CONV layer, there are both negative and positive features in the output. When we plot the output into 2D grayscale figures, negative numbers are treated as 0 while under image form. After CONV layer, data is input into RELU layer, where data is simply processed by MAX(0, data) function for each channel and only 0 and positive values are left. Therefore, the images of each channel from 1st layer of CONV and 2nd layer of RELU, which both have 20 channels, are similar to the original image input.

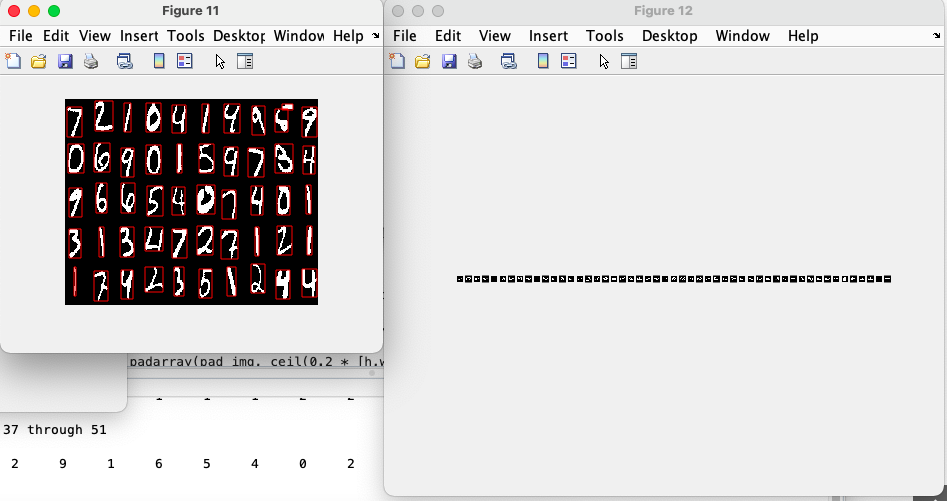
**Part 5**

Images with bounding boxes are as below:

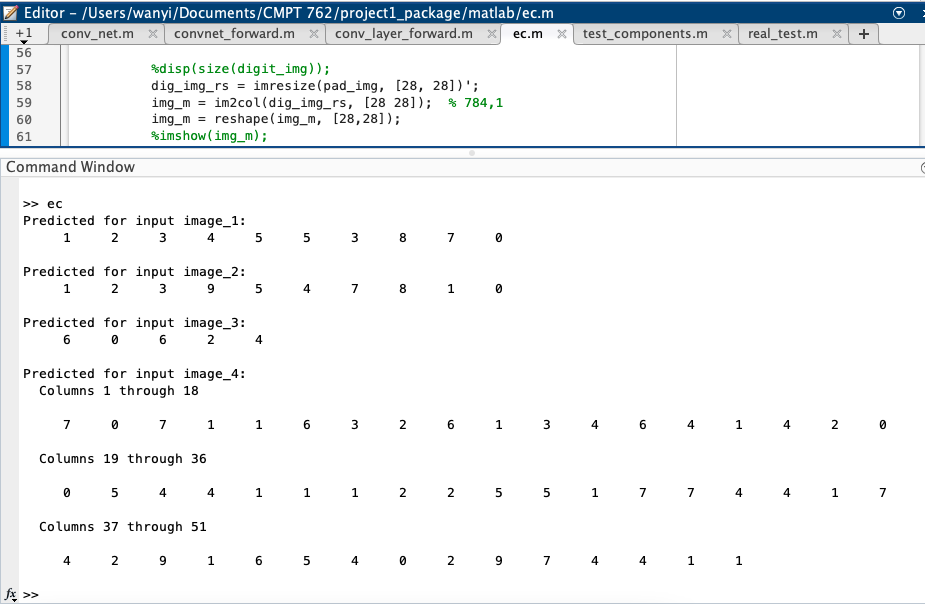








Predicted results are as follows:



Predicted results for the first two images are at 70% accuracy, and 100% accuracy for the third image. For the fourth image, since “5” are separated into two obvious parts (one horizontal line on the top and the remaining part of “5”), the model detects it as a separate object. This is why we got 51 objects detected in the result.

There are multiple reasons for errors of prediction here. First, the characters are written by hand and different people have different writing styles, which can cause bias when the model tries to predict numbers. Second, the input images are cropped and resized before being put into and detected by the model. The process can add some disturbance information that causes prediction errors.