Part1 Forward Pass

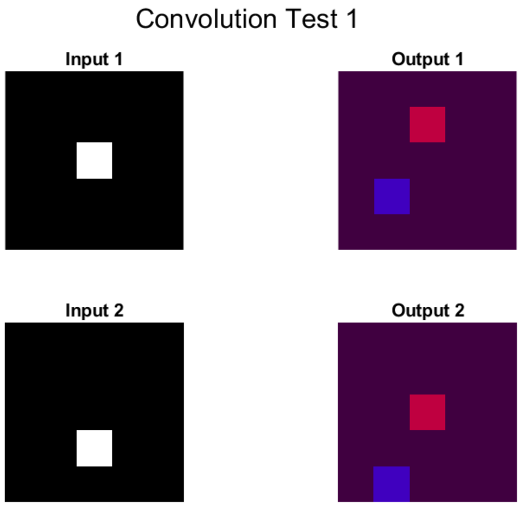
Q 1.1 Inner Product Layer Pooling Layer

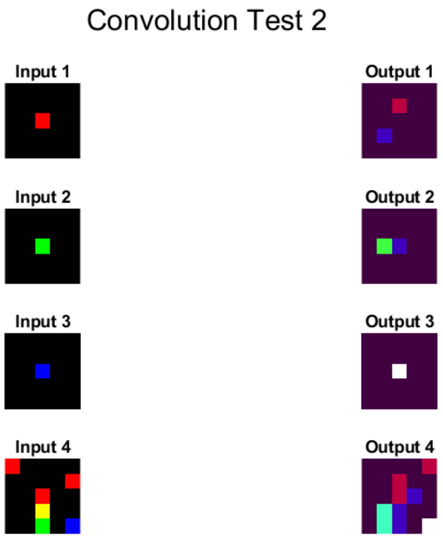
Q 1.2 Pooling Layer

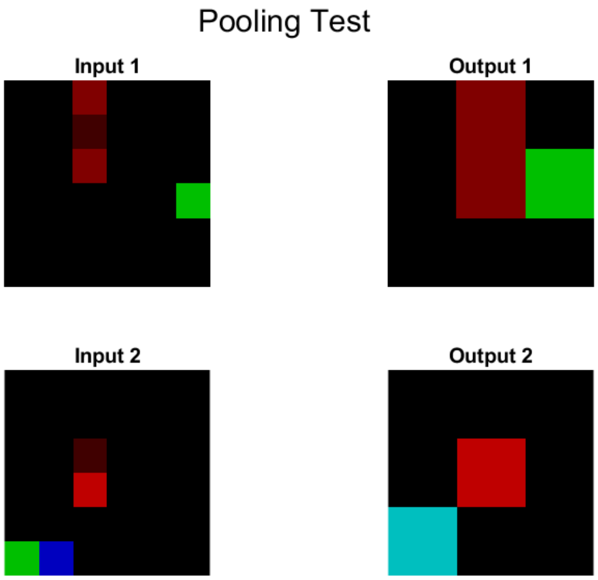
Q 1.3 Convolution Layer

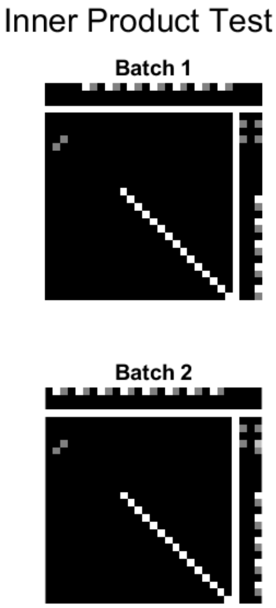
Q 1.4 ReLU

A forward pass network is implemented in this section, and the outputs are as follows:






Pooling Test
Input 1
Output 1
Input 2
Output 2


Inner Product Test
Batch 1
Batch 2

Part3 Training

Q 3.1 Training

By running train_lenet.m the below results were achieved:

```
>> train_lenet
cost = 0.273491 training_percent = 0.910000
cost = 0.279565 training_percent = 0.910000
cost = 0.176619 training_percent = 0.920000
cost = 0.127344 training_percent = 0.950000
cost = 0.191895 training_percent = 0.960000
test accuracy: 0.944000

cost = 0.192910 training_percent = 0.930000
cost = 0.131836 training_percent = 0.970000
cost = 0.115812 training_percent = 0.970000
cost = 0.103636 training_percent = 0.970000
cost = 0.124224 training_percent = 0.980000
test accuracy: 0.960000

cost = 0.111115 training_percent = 0.960000
cost = 0.113216 training_percent = 0.940000
cost = 0.134874 training_percent = 0.960000
cost = 0.067548 training_percent = 0.990000
cost = 0.095426 training_percent = 0.980000
test accuracy: 0.966000

cost = 0.086685 training_percent = 0.980000
cost = 0.106186 training_percent = 0.950000
cost = 0.034245 training_percent = 1.000000
cost = 0.048397 training_percent = 1.000000

cost = 0.060728 training_percent = 0.970000

test accuracy: 0.968000


cost = 0.069977 training_percent = 1.000000

cost = 0.068312 training_percent = 0.980000

cost = 0.063643 training_percent = 0.980000

cost = 0.084625 training_percent = 0.960000

cost = 0.083214 training_percent = 0.980000

test accuracy: 0.970000


cost = 0.083081 training_percent = 0.970000

cost = 0.026531 training_percent = 1.000000

cost = 0.044653 training_percent = 0.980000

cost = 0.056298 training_percent = 0.980000

cost = 0.049833 training_percent = 0.990000

test accuracy: 0.970000
```

Q 3.2 Test the network

By running test_network.m script the below confusion matrix was achieved.

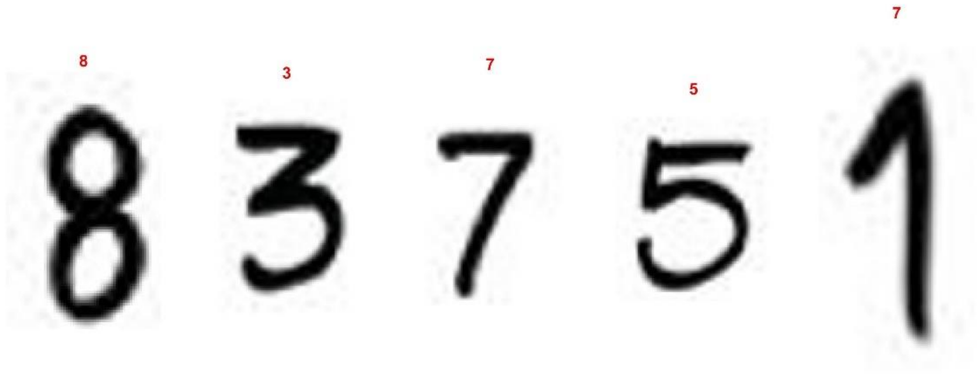
62	0	0	0	0	0	0	0	0	0
0	44	0	0	0	0	0	0	0	0
0	0	44	0	0	0	1	2	0	0
0	0	0	52	0	0	0	0	0	0
0	0	3	0	45	0	0	0	0	2
0	1	0	0	0	45	0	0	0	0
0	0	0	0	0	0	47	0	0	0
0	1	0	1	0	0	0	51	0	1
1	0	0	1	0	0	0	0	48	0
0	0	0	1	1	2	0	0	1	43

For the first class (zero), there is no confusing image.
Among the images for the 9 class (9), there are two that were confused with 9, which are 0 and 3.
For the last class (10), four images were confused with 10, which are 3,4,5,9.

Q 3.3 Real-world testing

A total of five images were downloaded, and then grayscaling and thresholding were used to provide them to the network.

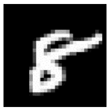
In the figure below, it can be seen that four of them are correctly recognized by the network.



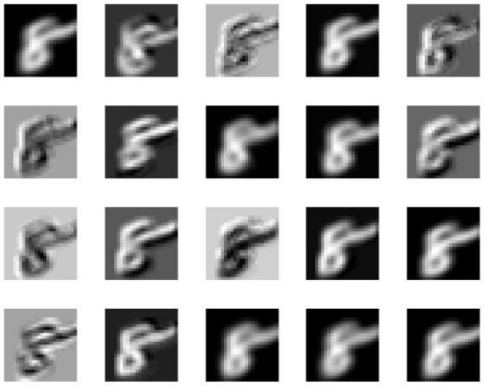
Part4 Visualization

Q 4.1

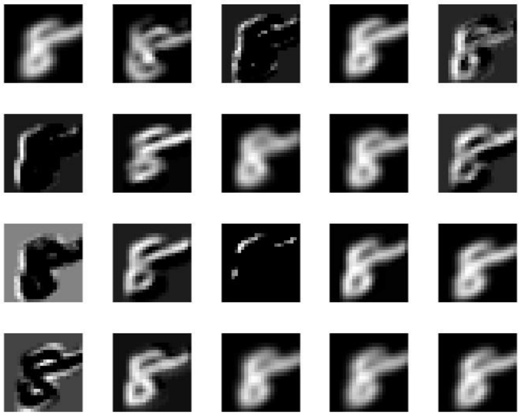
The original image:



Twenty images from the second layer which is conv



Twenty images from the third layer which is relu



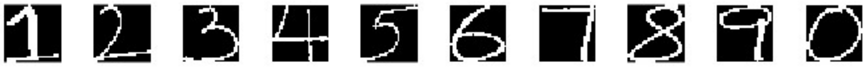
Q 4.2

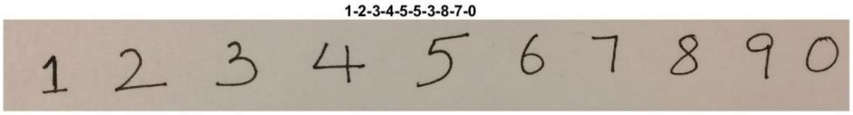
As a result of the convolution layer, the images are filtered, and they can become blurry or sharper in some cases.

In the relU layer, a thresholding operation has been performed to omit certain parts of the images.

Part5 Image Classification

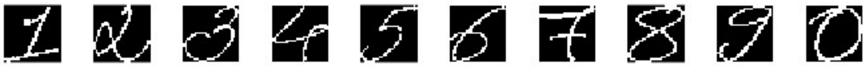
In this section a threshold was applied to the images, connected components were found, and the image was cropped using the minimum and maximum values, and then images were padded.





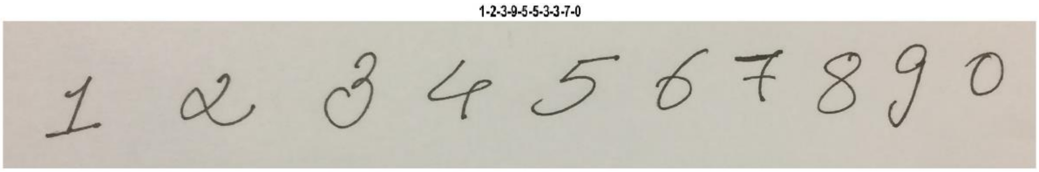
Result: 1-2-3-4-5-5-3-8-7-0

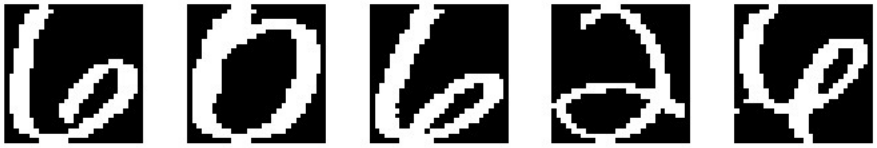
As can be seen seven classes were predicted correctly.

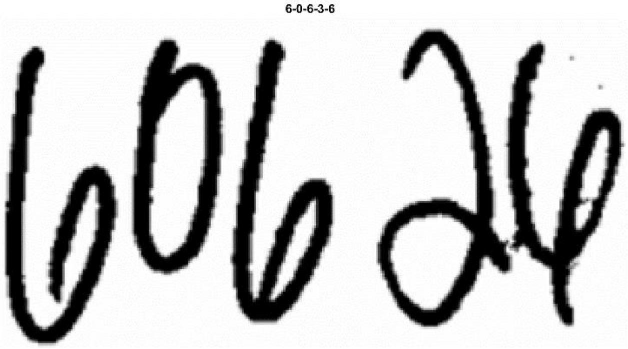


Result: 1-2-3-9-5-5-3-3-7-0

As can be seen five classes were predicted correctly.

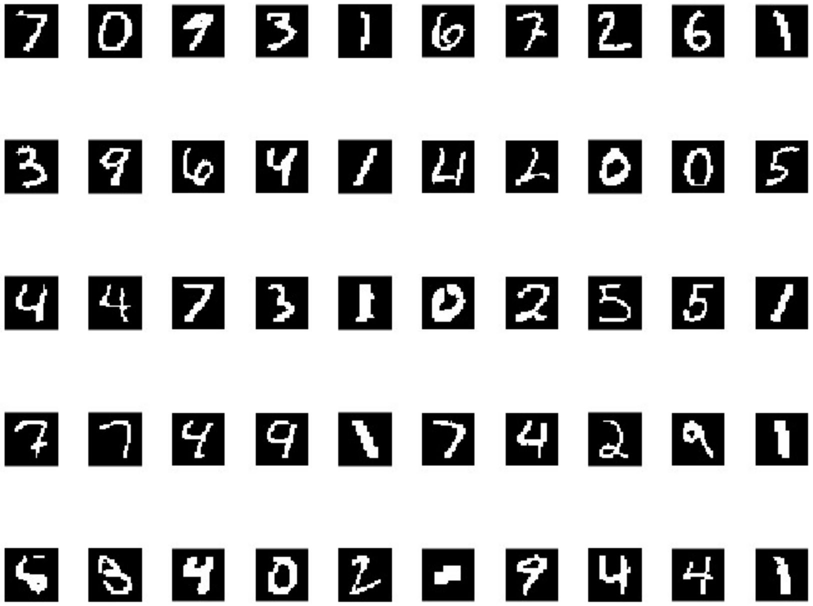


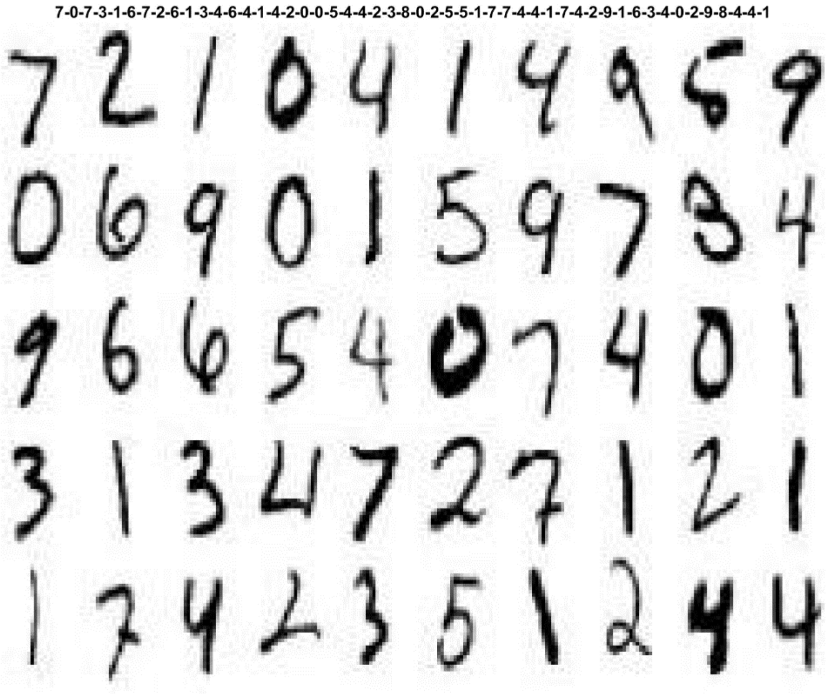




Result: 6-0-6-3-6

As can be seen four classes were predicted correctly.





Result: 7-0-7-3-1-6-7-2-6-1, 3-4-6-4-1-4-2-0-0-5, 4-4-2-3-8-0-2-5-5-1, 7-7-4-4-1-7-4-2-9-1, 6-3-4-0-2-9-8-4-4-1

As can been seen 44 images out of 50 images were predicted correctly.