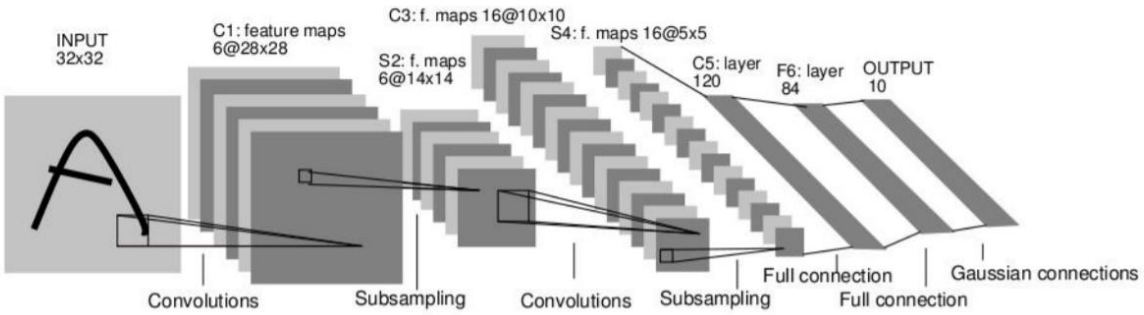
Semester Assignment 2: Convolutional Neural Network

Part 1

Here you will be tested on the understanding of a CNN. You will be asked to program in Python part of a CNN and without using the standard libraries.



Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner,
Gradient-based learning applied to document recognition, Proc. IEEE 86(11): 2278–2324, 1998.

The planned CNN that you are addressing should detect the letters “A” and “K” from the data set given in the appendix here. However, you are going to program just a part and test this part.

The part that will be your responsible for will consist of two layers that will be connected to each other. Each layer will consist of a convolution, a ReLU and Max pooling. The task at hand will include the following:

1. Organize the translation of the graphic letters for training and testing by creating a proper matrix. It does not have to be big or very fine resolution.
2. Basic convolution
    a. Determine padding, kernel size and stride
    b. You decide your own kernels for the purpose. You might consider 2-4. Tip: Check other kernels used for the same purpose.
3. The ReLU should process the output
4. Then do Pooling
    a. Decide again the kernel size and stride
5. Repeat the same thing for the second layer.

Test the signals that runs through the system and vary kernel size and stride. Create a set of charts that shows the effect of this in a way that you think is best.

Part 2

Set up Keras and Tensorflow and write the necessary Python code to interpret the different forms of A and K among the other characters given in the appendix. Apply the kernel size and other parameters that you decided on in Part 1. Preprocess the images at your own will. Measure MAPE. Evaluate and comment and make recommendations for improvement

Part 3

Repeat the effort for photos to interpret objects in the photos. Apply the following code to load the images:

```
from keras.datasets import cifar10

c10 = cifar10.load_data()

xtr,ytr,xte,yte = (b for a in c10 for b in a)
```

Test the system with different kernels and stride factors. Test also training with and without dropout. Comment on the results.

Write a short article (3-4 pages) about the work that you have done and the findings. Use the same type of structure as for Assignment 1:

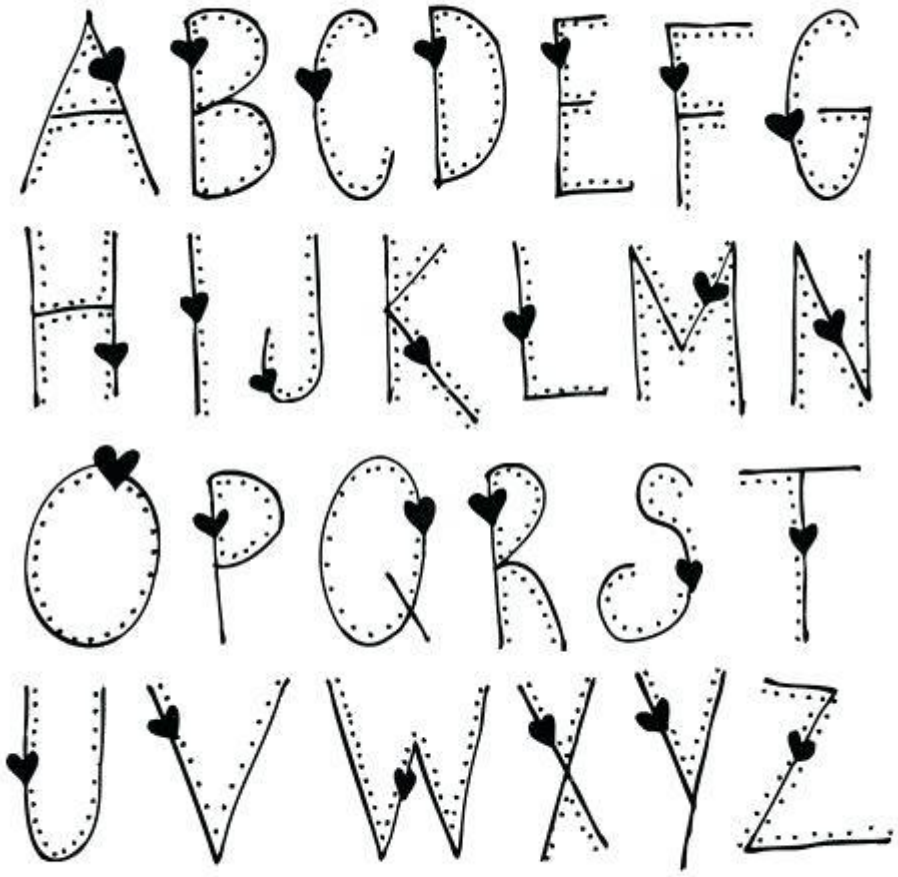
1. Introduction
2. Problem specification
3. Methods applied
4. Results
5. Discussion
6. Conclusions and recommendations for further work
7. References

Hand in the article and the code no later than November 6th. This effort will account for 40% of your final grade in the course.

APPENDIX

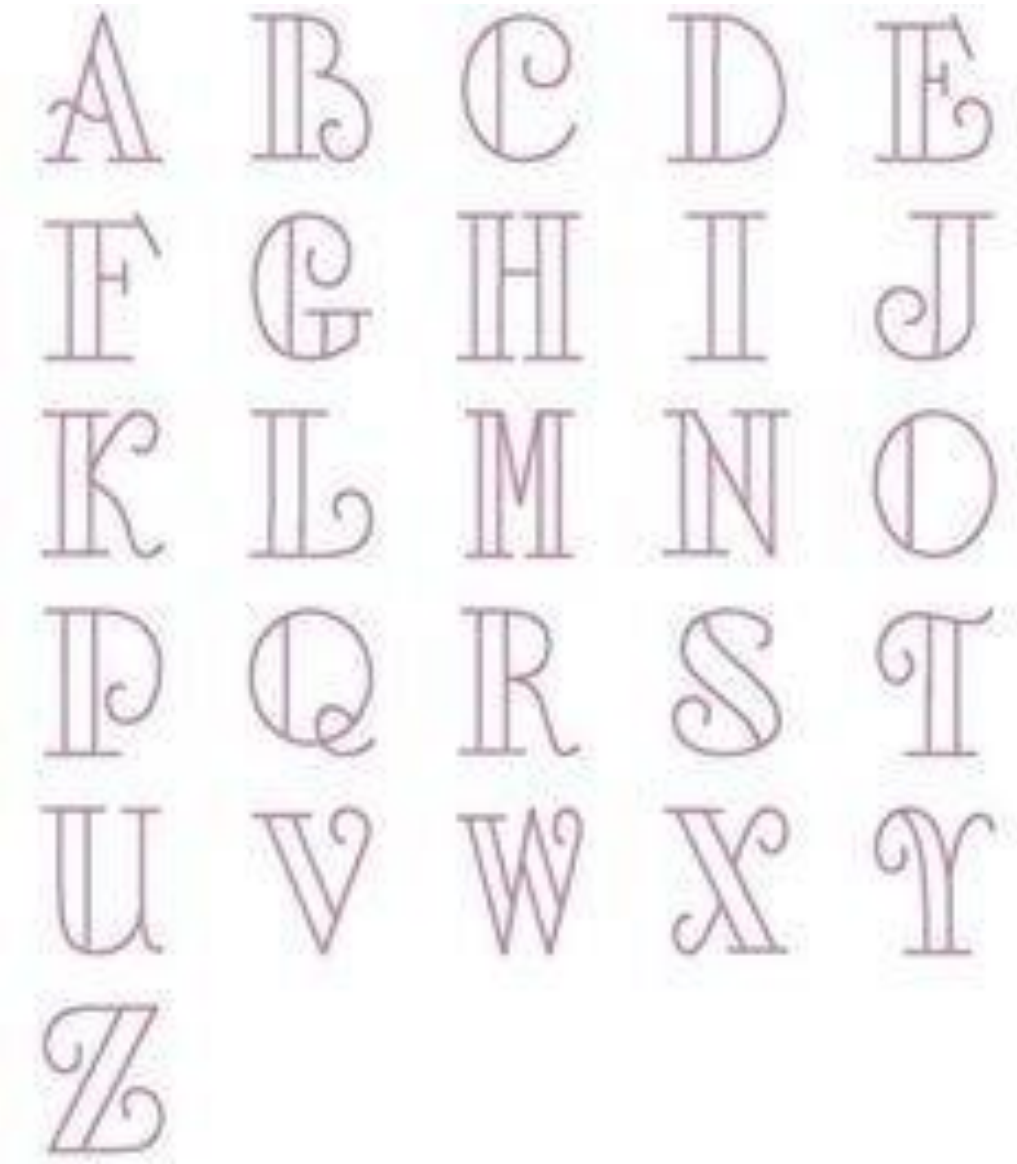
These are the characters that you will be preoccupied with in Part 1 and 2.


A B C D E
F G H I J
K L M N O
P Q R S T
U V W X
Y Z



LOVIE STAMPS {CU}


A B C D
E F G H
I J K L
M N O P
Q R S T
U V W
X Y Z


A B C D E
F G H I J
K L M N O
P Q R S T
U V W X Y
Z


A B C D E F G H I J
K L M N O P Q R S
T U V W X Y Z
a b c d e f g h i j k l m n o p q r
s t u v w x y z
1 2 3 4 5 6 7 8 9 0