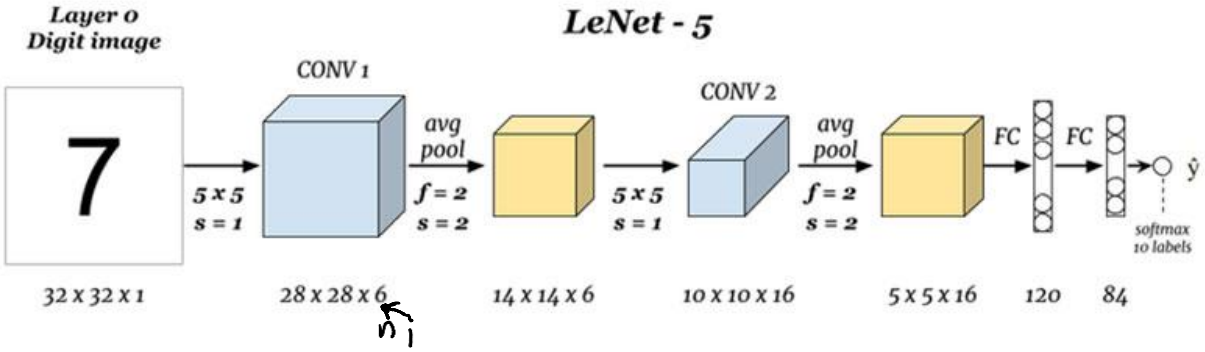
Assignment 2

Use the ReducedMNIST which is a reduced version of the MNIST data set.

- **ReducedMNIST training**: 1000 examples for each digit.
- **ReducedMNIST test**: 200 examples for each digit.

a. Use multilayer perceptron (MLP) (it is also called Feedforward neural network (FFNN)) with 1, 3, or 5 hidden layers to solve this problem. You may use any of the features used in assignment 1. You are free to use any hyper-parameters of your own choice.

b. a. Use the ReducedMNIST data to train CNN from the images without using a feature extraction step. Use the following structure to start with.
(Hint: you must adjust your parameters to fit 28x28 images instead of 32x32, and use **ReLU** function as an activation function).



c. Make some (at least two) variations in the hyper-parameters of your choice and check the network performance, then give your comments. These changes may be in the number of filters in any convolutional layer, the activation function, adding or removing any layer…etc.

d. Compare the results that you have obtained in this assignment with those you have obtained in Assignment 1; regarding the accuracy, the training time, and the testing time (fill the following table).

		Features						
		DCT		PCA		AutoEncoder		
		Accuracy *	Processing Time**	Accuracy	Processing Time	Accuracy	Processing Time	
Classifier								
K-means Clustering	1							DCT and PCA features from assignment 1
	4							
	16							
	32							
SVM***	Linear							
	Nonlinear:							
Multi-layer Perceptron (MLP)								
		DCT		PCA		AutoEncoder		
	Variations	Accuracy*	Processing Time**	Accuracy	Processing Time	Accuracy	Processing Time	
MLP	1-Hidden							Assignment 2
	3-Hidden							
	5-Hidden							
In the CNN Model no Features are needed								
	Variations	Accuracy	Training time		Testing time			
CNN****	Variation1:							
	Variation2:							
	Variation3:							
	Variation4:							

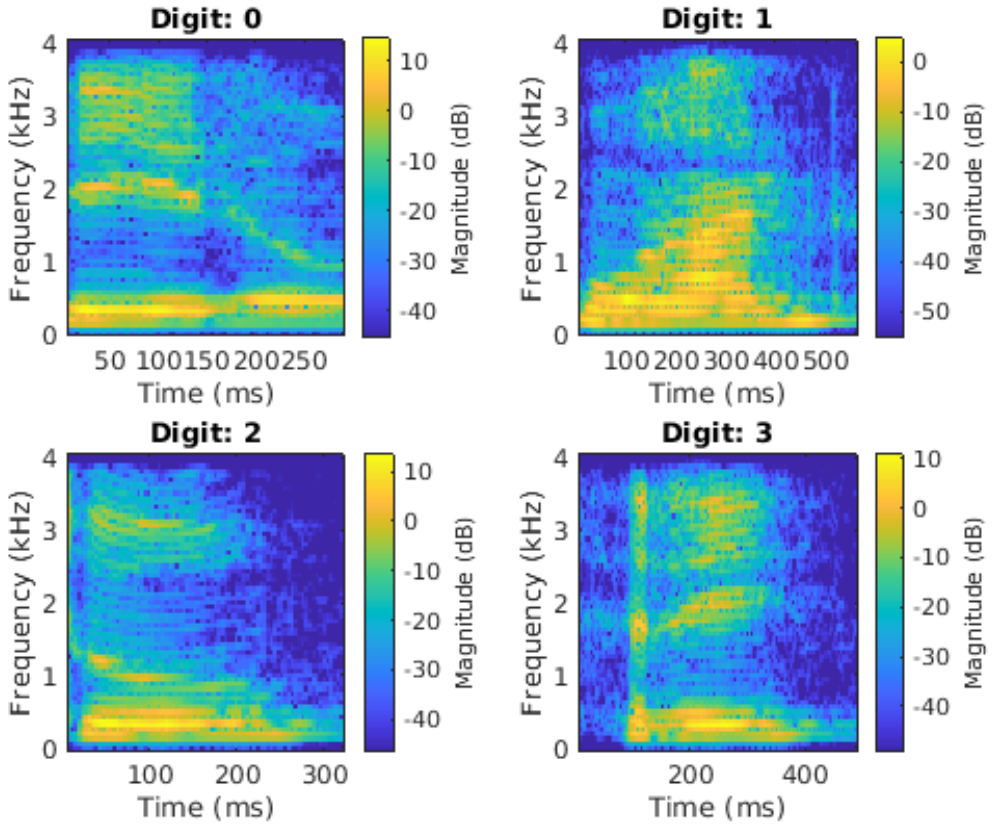
* Accuracy to be as % with one friction digit like 89.5% not as 89.51111111%

** Processing time in milli-seconds like 10.3 msec. not as 10.3333333 msec.

*** mention the kernel name and its specs

**** Describe each variation in short in the corresponding cells

4. **Speech Recognition from Speech Spectrum with Augmentation Experience:**
   a. Given the speech data (train and test) for the 10 digits uttered by many speakers, develop and train a network to recognize any given new digit using the spectrogram of each digit as the training and testing data. You may convert the speech data to images and deal with the speech problem as if it is an object recognition problem in images. **(Hint: you may start from the network in problem 1 and enhance by making some alternations in the hyper-parameters or start from any of the networks that have been used in the ImageNet problem).**
   b. Repeat "a" above after making data augmentation for the speech data; for example: speed up 5% and down speed 5%, and you may add some speech noise.
   c. Repeat "a" above after making data augmentation for the spectrum images; for example: squeeze (horizontally) the image by 5%, expand (horizontally) the image by 5%, and you may add some noise to the images.
   d. Repeat "a" above after making speech and image augmentation as done in "b" and "c" above.



Examples of spectrum images for some digits