ASSIGNMENT 2: SIAMESE NETWORK

VINAY HN - N10180893

RAHUL RAJENDRAN - N10145281

KARTHIKEYAN JAYAVEL - N10124004

1. Introduction

1.1. Experiment Context:

Dr Fredric Maire and his students have identified that a Siamese Network based hierarchical convolution Neural Networks is capable to discriminate homographic images with a wide range of transformation. Their research is based on the Atomic verification system for Marine biologist to monitor the marine life such as manta rays. The System can verify the manta rays with different angels.

Figure 1: Two poses of one Manta ray

1.2. Objective:

We need to examine the pairs from MNIST digits and compute their accuracy. If the 2 handwritten digits are same then we will get more accuracy. Using the Siamese Network we determine whether the 2 images are similar or not. We need to train the MNIST digits pairs such as [2,3,4,5,6,7]. We are not training the digit pairs [0,1,8,9]. We need to test the training and testing pairs and compute the accuracy.

We need to test the capability of our network by doing the below experiments.

a. We need to test the digits [2,3,4,5,6,7] as training and [2,3,4,5,6,7] as testing pairs.
b. We need to test the digits [0,1,2,3,4,5,6,7,8,9] as training and [0,1,2,3,4,5,6,7,8,9] as testing pairs.
c. We need to test the digits [0,1,8,9] as training and [0,1,8,9] as testing pairs.

Based on the above experiments, we need to plot the training and validation errors vs time.

1.3. Siamese Network and CNN

Siamese network are a special type of neural network architecture. Instead of a model learning to classify its inputs, the neural networks learns to differentiate between two inputs. It learns the similarity between them

There are two sister networks, which are exact neural network which are having the same weight and bias. Each image from the MNIST handwritten digit pairs are feed into one of these networks. We will get loss based on the distance between the 2 vectors and we need to minimise the loss function using the contrastive loss function

The objective of the Siamese network is not classify of the two images but to find the Difference between the two input images.

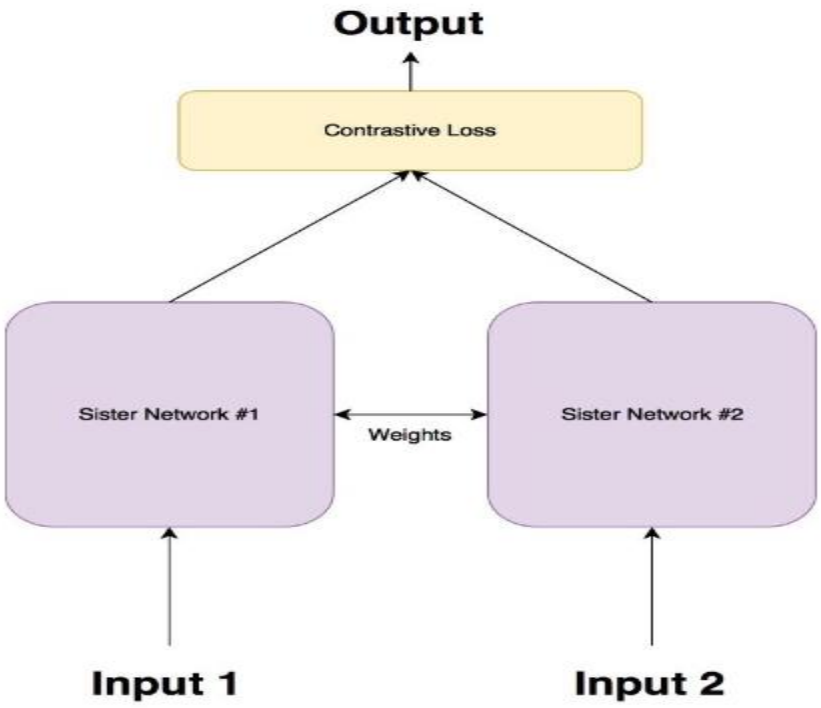


Figure2: Siamese Network

Convolution Neural Network (CNN) are the current state of art architecture for the image classification. Which are mainly used for face recognition, driverless cars, Social Medias like Facebook (image tagging), etc.

We are using the MNIST handwritten digits and we need to find the accuracy for the training as well as testing pairs. MNIST is a dataset, which consists of images of handwritten digits from 0 to 9. Each image is a monochrome, 28*28 pixel. The below image is the sample of MNIST digit.

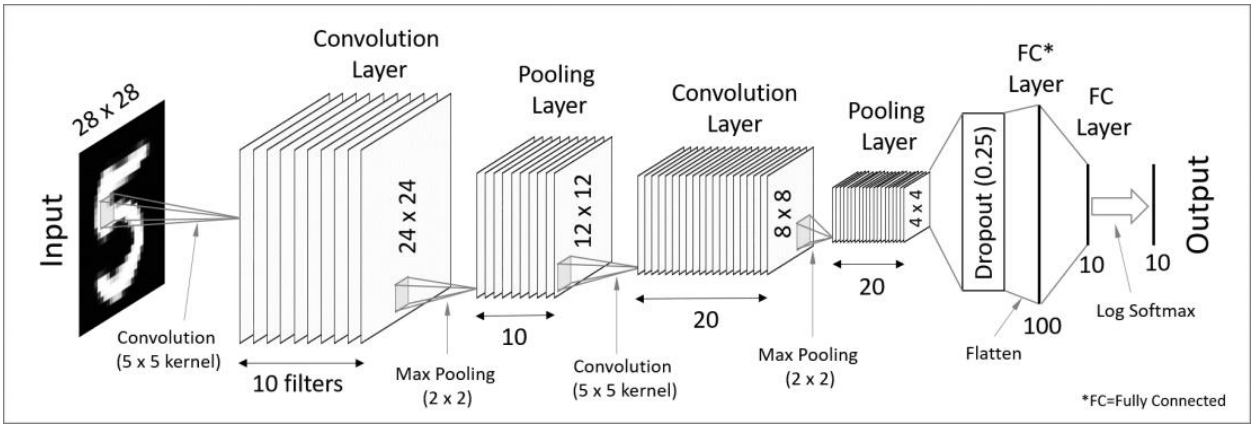


Figure 3: Convolution Neural Network

2. Experimental Methodology:

2.1. Computing Environment:

Processor: AMD A6-9220 RADEON R4, 5 COMPUTE CORES 2C+3G 2.50 GHz
RAM: 8.00 GB (7.47GB usable)
System type: 64-bit operating system, 64-based processor
Windows edition: Windows 10

2.2. Experiment Approach:

2.2.1. Major Functions:

euclidean_distance():
which will compute the Euclidian distance between the 2 vectors. The distance is less for Positive pairs of images and the distance is more for the negative pairs of the images.

contrastive_loss():
Which takes the input from the actual image and predicted image and produce the variation between those images. There are 2 set of images positive and negative pairs. The value is 1 for positive pair and o for the negative pair. The trained output y value is the distance output of the Siamese network.

Compute_accuracy():
Which will compute the accuracy between the pairs of trained and tested images. If the accuracy is more then the predicted image is same as the actual output. These values are computed by the Siamese network.

create_pairs():
It will create the positive pairs and the negative pairs in an alternative passion.

If the two MNIST digits are similar then the image is positive else the image is negative.

create_Siamese_network(input_dim):
Siamese Network consists of 4 layers of the convolution network. We are using “relu” activation function.

train_test_evaluate(epochs):

This function will train the Siamese network to predict whether 2 input images corresponds to the same digits (Positive pairs) or not (Negative pairs).

2.2.2. **Train the Network**:

We are using the MNIST handwritten digits. Has 60000 training samples and 10000 test samples. We need to keep 80%of the [2,3,4,5,6,7] digit pairs for training and remaining for the testing. After building the Siamese network, it is ready to implement with experiments. Accuracy for the [234567] is more because they already trained and Accuracy for the pair [0189] is less because they are not trained.

We need to do the following process step by step, in order to find the similarity between the training and testing MNIST digits.

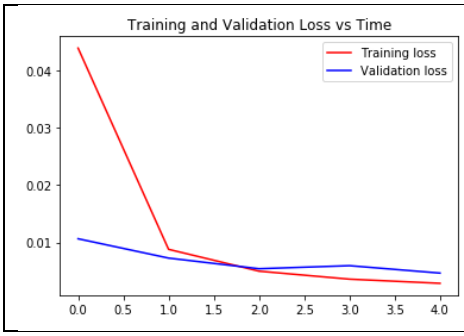
- Load the MNIST data set using the Keras.
- Create the Paired Images.
- Define the Model
- Input the 2 MNIST images
- Implementing the Input to the Siamese Network.
- Compute the distance between the two vectors.
- Compile the Model
- Fit the Model.
- Evolute the Model
- Plot the Loss Function
- Compute the Accuracy
- Time associated to compute the accuracy for different Epochs.

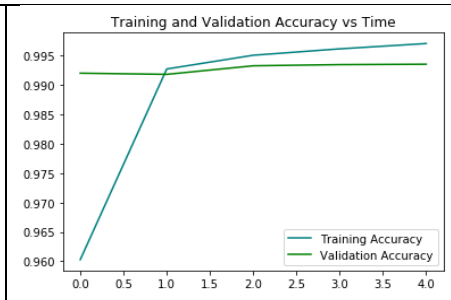
3. Experimental Results:

Epochs – 5

Accuracy of [234567]:	Accuracy of [0123456789]:	Accuracy of [0189]:
98.98%	86.98%	71.85%

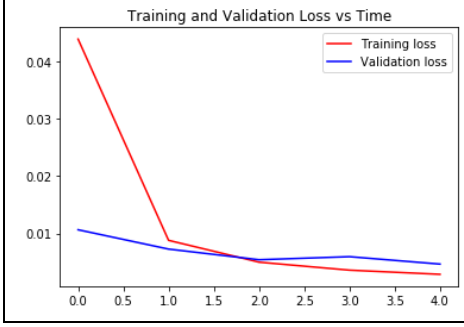
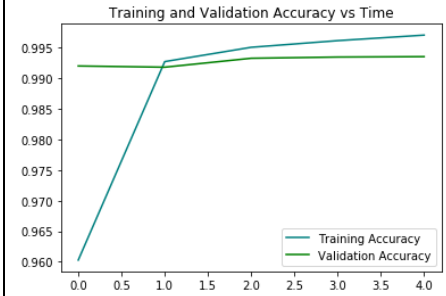
Figure 4: Training and Validation Error vs Time	**Figure 5: Training and Validation Accuracy vs Time**





Epochs – 10

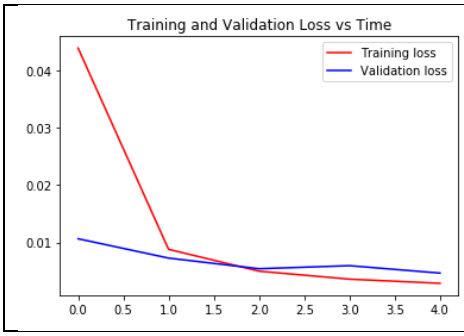
Accuracy of [234567]:	Accuracy of [0123456789]:	Accuracy of [0189]:
99.36%	86.60%	69.10%

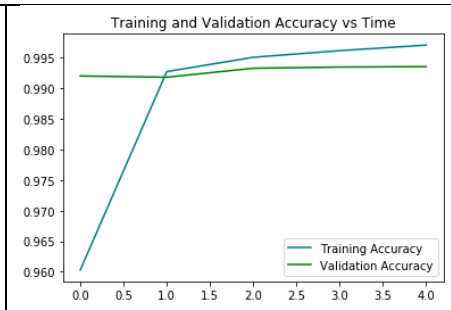
Figure 6: Training and Validation Error vs Time	**Figure 7: Training and Validation Accuracy vs Time**
	

Epochs – 15

Accuracy of [234567]:	Accuracy of [0123456789]:	Accuracy of [0189]:
99.27%	87.95%	66.77%

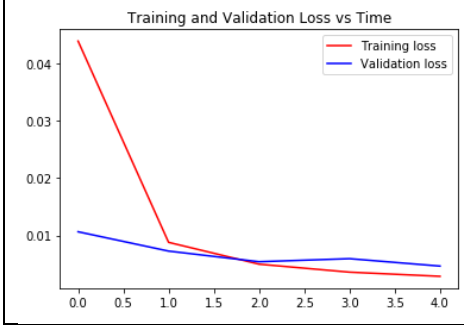
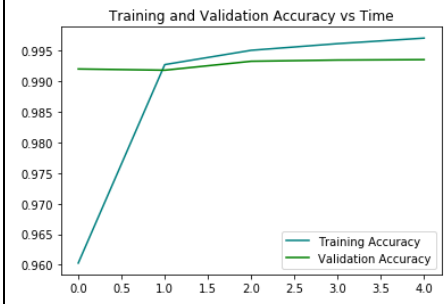
Figure 8: Training and Validation Error vs Time	**Figure 9: Training and Validation Accuracy vs Time**





Epochs – 20

Accuracy of [234567]:	Accuracy of [0123456789]:	Accuracy of [0189]:
99.12%	86.65%	68.60%

Figure 6: Training and Validation Error vs Time	**Figure 7: Training and Validation Accuracy vs Time**
	

Accuracy Prediction vs Epochs.

The below diagram represents the Predicted Accuracy over all the Epochs. We used 4 Epochs which are 5,10,15,20. Which is shown below.

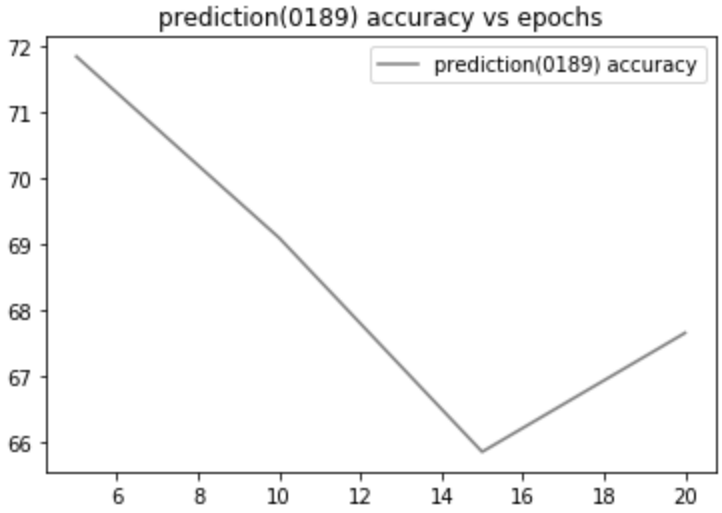


Figure 10: Accuracy [0189] pairs vs Time