ECS 189G-001

Deep Learning

Winter 2024

Course Project: Stage 3 Report

Team Information

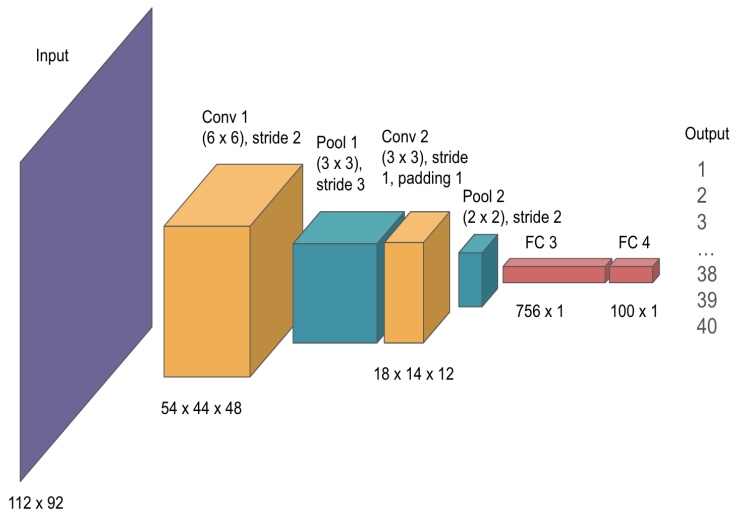
NeuralNinjas		
Student 1: Srihita Ramini	Student 1: 919221527	Student 1: sramini@ucdavis.edu
Student 2: Trishna Sharma	Student 2: 918135782	Student 2: tksharma@ucdavis.edu
Student 3: Sai Sindura Vuppu	Student 3: 918091017	Student 3: svuppu@ucdavis.edu
Student 4: Olivia Shen	Student 4: 919293157	Student 4: oshen@ucdavis.edu

Section 1: Task Description

The task is to develop individual CNN models for three distinct datasets: ORL, MNIST, and CIFAR-10.

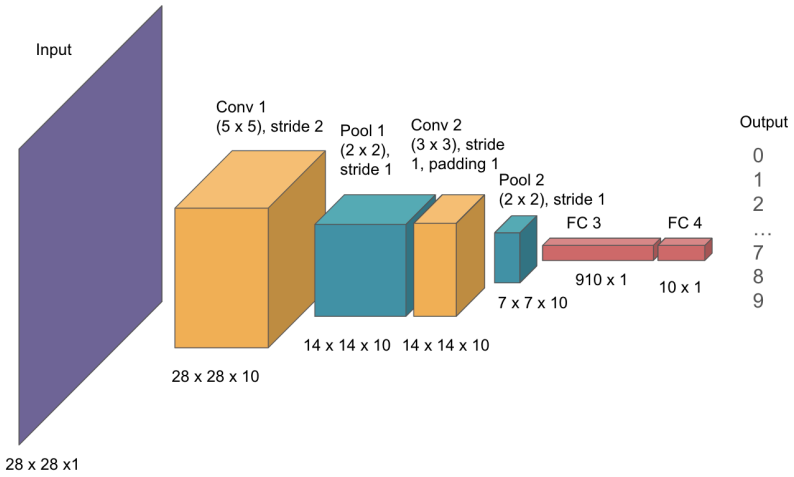
Section 2: Model Description

ORL:

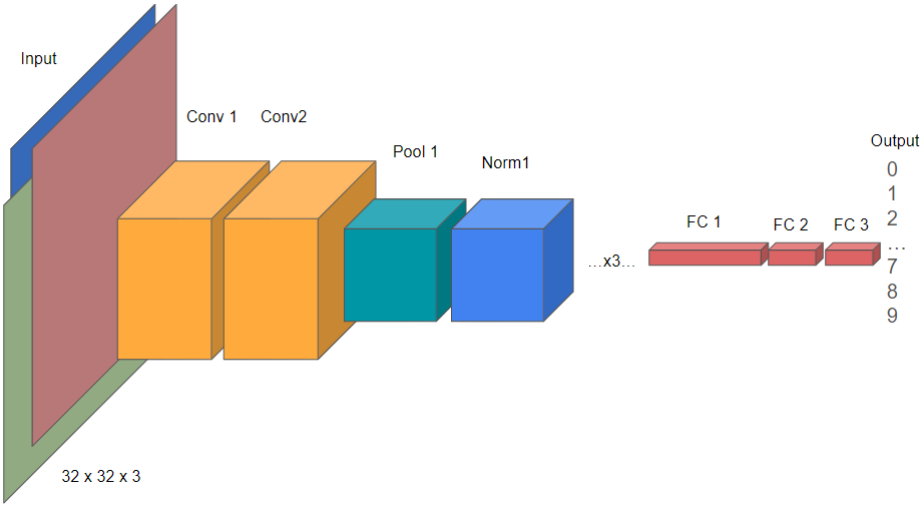


The ORL model is a CNN classifier that predicts the label 1-40 of an image of a human face. It has 2 convolutional layers, 2 pooling layers, and 2 fully connected layers. For activation functions, it uses ReLU on convolution layers, Tanh on FC3, and Softmax on FC4.

MNIST:



CIFAR:



The CIFAR model classifies colored images of a small dimension of 32 x 32 having 3 channels, RGB, into 10 values ranging from 0-9. It has 3 blocks of Conv layers, pooling, batch normalization layers, followed by 3 fully connected layers, having a total of 6 Conv layers, 3 Pooling, 3 Batch Normalization, 3 Fully Connected layers.

Section 3: Experiment Settings

3.1 Dataset Description

The ORL dataset contains 112 x 92 images of human faces. There are 40 different people in the dataset, each with 10 distinct images (9 for training, 1 for testing). Each image has a label {1, 2, …, 40}. The images have 3 channels set to the same values to produce grayscale. The dataset is divided into 360 training instances and 40 testing instances.

The CIFAR10 dataset contains of 50,000 colored images of size 32 x 32 x 3 (3 channels for RGB). There are 10 distinct labels {0,.., 9}. The images have 3 channels with different values for RGB.

The MNIST dataset contains images of handwritten digits ranging from the integers 0 - 9. The image is in the shape of a 28 x 28 matrix and the labels of the images are ranging from {0,1,2…9}. The images are in

greyscale and there is only one channel. There are 60,000 training images and 10,000 testing images in this dataset.

3.2 Detailed Experimental Setups

We designed separate CNN models for each dataset as follows:
For the ORL dataset, we will apply min-max normalization, then standardization with mean = 0.5, std = 0.5. This addresses the vanishing gradient problem that occurs in data with a large range. We will use a learning rate of 2e-3, 50 training epochs, cross-entropy loss, and Adam optimizer. The model uses 2 convolutional layers, each followed by a pooling layer for compression, and two fully connected layers. Conv1 learns 48 (6x6) kernels with stride 2. Conv 2 learns 12 (3x3) kernels with stride 1 and padding 1. For activation functions , layers 1 and 2 use ReLU, layer 3 uses Tanh, and layer 4 uses Softmax.

For the MNIST dataset, we initially designed the model architecture with 2 convolutional layers with 2 max pooling layers and ending with two FC layers. The first Conv Layer has a 5 x 5 kernel size with a stride of 2 and the second one has a 3 x 3 kernel size with a stride of 1. The model was trained with a learning rate of 2e-3, 50 training epochs, cross-entropy loss, and Adam optimizer. The second model architecture consisted of the same architecture as mentioned above, but with one extra convolutional layer with a kernel size of 3 x 3 and a stride of 1. It was training with the same learning rate, epochs, loss, and optimized as mentioned above. Various configurations of activation functions such as ReLU, PReLU, TanH, and Softmax were used in order to get the highest performance.

For the CIFAR dataset, we reference this link to build the model. We use a learning rate of 1e-3, 10 epochs, cross entropy loss and Adam optimizer. Our CNN network is more complex than the previous networks. We had a learning rate of 1e-3, 10 epochs, cross-entropy loss, and Adam optimizer. The model has 2 convolutional layers followed by a max pooling later that reduces the size by half and a batch normalization to normalize the values. This block is repeated 3 times. We then use a flatten layer and three fully connected layers to reduce it to 10 values in the output. We used mini-batch instead of full batch to do the training

3.3 Evaluation Metrics

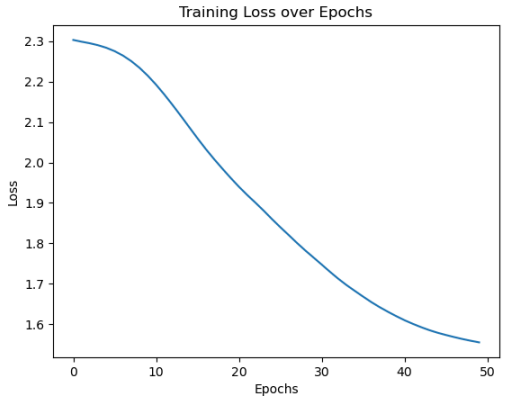
All of the experiments are multiclass classification problems. Accuracy, Precision, Recall, and F1-score are our evaluation metrics. We will take the macro average of Precision, Recall, and F1-score to accommodate multiple classes using Scikit-Learn. We have two Conv2d layers followed by a max pooling layer to reduce it by half and batch normalization layer to normalize the values in it. This block is repeated three times. We then flatten it and apply fully connected layers to get the output size to 10.

3.4 Source Code

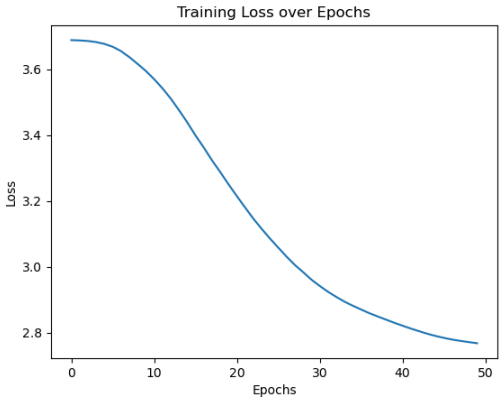
https://github.com/srihita123/189G-

3.5 Training Convergence Plot

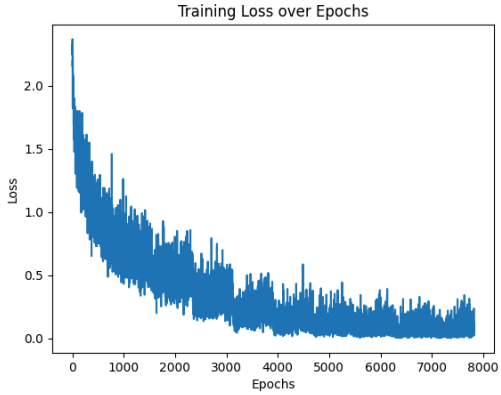
MNIST:



ORL:



CIFAR:



3.6 Model Performance

MNIST:

```
************ Overall Performance ************
Accuracy: Accuracy: 0.9433
Precision: 0.9425860325890627
Recall: 0.942793250712047
F1: 0.9426418069246522 +/- None
```

ORL:

```
************ Performance ************
Accuracy: 0.975
Precision: 0.9625
Recall: 0.975
F1: 0.9666666666666668
```

CIFAR

```
Accuracy: 0.6806
Precision: 0.69302259997076
Recall: 0.6806
F1: 0.6836272622413647
```

```
Epoch [1/10], Loss: 1.1454
Epoch [2/10], Loss: 0.6885
Epoch [3/10], Loss: 0.4969
Epoch [4/10], Loss: 0.3410
Epoch [5/10], Loss: 0.2220
Epoch [6/10], Loss: 0.1457
Epoch [7/10], Loss: 0.1131
Epoch [8/10], Loss: 0.0956
Epoch [9/10], Loss: 0.0799
Epoch [10/10], Loss: 0.0739
```

3.7 Ablation Studies

ORL

Changes to model	**Accuracy**	**Precision**	**Recall**	**F1 Score**
Conv1: 64 kernels, Conv2: 16 kernels	0.95	0.925	0.95	0.933
Max epochs: 40	0.925	0.8875	0.925	0.9
Learning rate: 1e-3	0.925	0.8875	0.925	0.9
Max epochs: 40, Learning rate: 3e-3	0.95	0.925	0.95	0.933
Conv1: kernel size: 3, stride:1, padding:1 Conv2: kernel size: 6, stride:2 FC3: 672 inputs	0.9	0.85	0.9	0.8667
Max epochs: 40 Remove FC3, FC4: 756 inputs	0.925	0.8958	0.925	0.9042

MNIST

Changes to model	**Accuracy**	**Precision**	**Recall**	**F1 Score**
(2 Conv layers, 2 FC layers)	0.699	0.630	0.687	0.634
Activation Functions Sigmoid/Tanh/Tanh/Softmax	0.854	0.781	0.845	0.801
Activation Functions ReLU/Tanh/ReLU/Softmax	0.945	0.944	0.944	0.944
Activation Functions ReLU/Tanh/Tanh/Softmax	0.860	0.787	0.850	0.814
Activation Functions ReLU/Tanh/PReLU/Softmax	0.923	0.922	0.920	0.920
(3 Conv layers, 2 FC layers) Activation Functions ReLU/Tanh/Tanh/PReLU/Softmax	0.793	0.745	0.782	0.728

ReLU/Tanh/Tanh/ReLU/Softmax				

CIFAR

Changes to model	**Accuracy**	**Precision**	**Recall**	**F1 Score**
2 Conv, 2 Pooling, 3 FC (kernel size = 6, stride = 2) (ouput channel 64 -> 128 -> 384 -> 192 -> 10)	0.1308	0.216	0.1308	0.09
2 Conv, 2 Pooling, 3 FC (kernel size = 5, stride = 1) (ouput channel 32 -> 64 -> 384 -> 192 -> 10)	0.2804	0.414	0.2804	0.2384
6 Conv, 3 Pooling, 3 Batch Normalization, 3 FC	0.6806	0.6930	0.6806	0.6836