ML - Neural Networks

Alexandru MIHAI, 341C3

January 9, 2025

1 MLP - Extracted Features

1.1 FashionMNIST

Batch size: 256, Training Epochs: 10, Learning Rate: 0.001, Loss function: Cross Entropy loss, Optimizer: Adam, Dropout factor: 0.2.

Layer	Output Shape
Input	63
Linear + ReLU	128
Dropout	128
Linear + ReLU	256
Dropout	256
Linear	10

Table 1: MLP on extracted features architecture - FashionMNIST

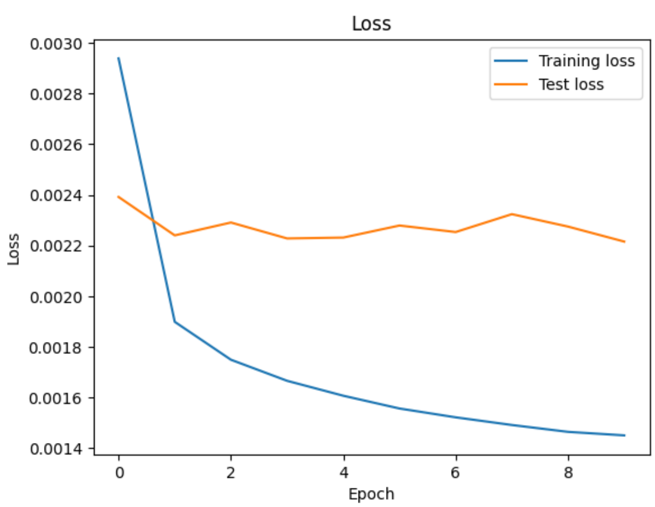

Figure 1: Loss Function for Feature-MLP - FashionMNIST

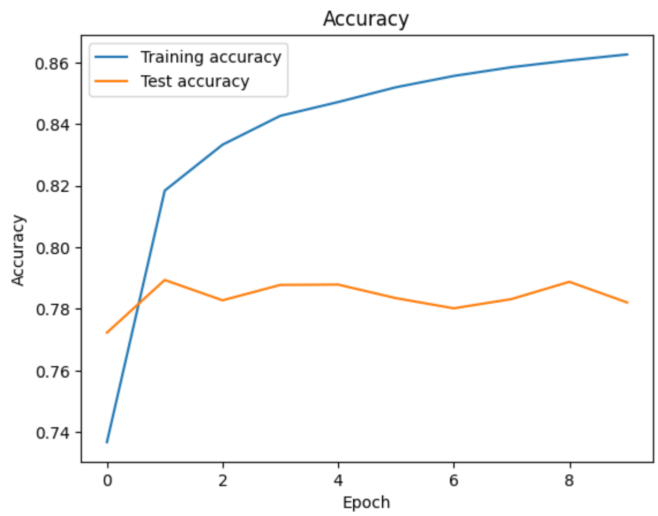

Figure 2: Accuracy for Feature-MLP - FashionMNIST

Feature MLP is the simplest and fastest model among the options. However, its accuracy is the lowest at 0.79, which indicates that it struggles to effectively capture the spatial relationships inherent in image data. This is expected as MLPs are not optimized for grid-like structures in data.

Compared to the classic ML algorithms used for the first assignment, Feature MLP for the FashionMNIST dataset performs worse than all algorithms (0.79 accuracy compared to SVM's 0.84). This outcome may have a couple of reasons: the number of features used (63 compared to 119 used for the other algorithms) might not fully describe the relevant information of each image and the used MLP architecture might be too shallow for each neuron to learn meaningful information.

1.2 Fruits 360

Batch size: 256, Training Epochs: 10, Learning Rate: 0.001, Loss function: Cross Entropy loss, Optimizer: Adam, Dropout factor: 0.2.

Layer	Output Shape
Input	110
Linear + ReLU	256
Dropout	256
Linear + ReLU	512
Dropout	512
Linear	141

Table 2: MLP on extracted features architecture - Fruits 360

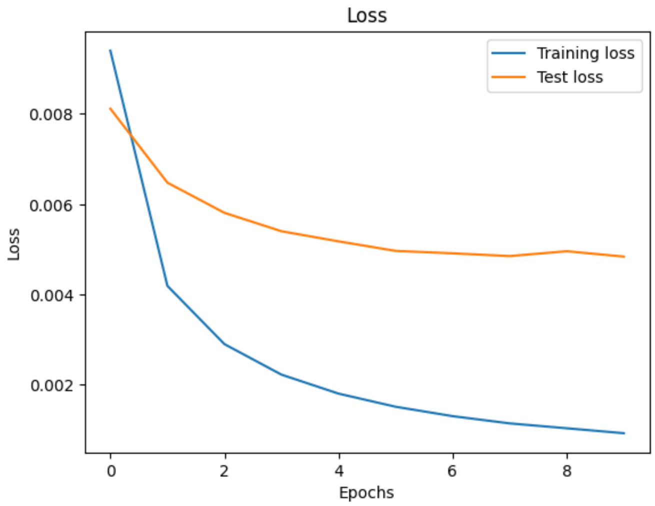

Figure 3: Loss Function for Feature-MLP - Fruits 360

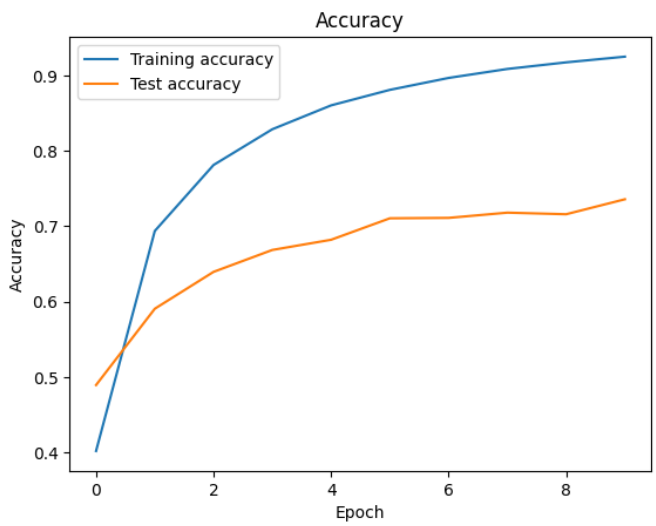

Figure 4: Accuracy for Feature-MLP - Fruits 360

Feature MLP achieves the lowest accuracy among the models, demonstrating its limited capacity to handle the complexities of image data. This is expected, as MLPs do not inherently exploit spatial features, a critical aspect of image classification tasks. However, its extremely low training time makes it viable for quick, low-resource applications where accuracy is not a priority.

Compared to classic ML algorithms, Feature MLP on the Fruits 360 dataset outperforms all algorithms except SVM (0.72 accuracy compared to 0.80). As the number of features extracted is higher compared to the Fashion-MNIST dataset, the MLP is able to better learn the meaning behind the extracted features and result in relevant labels. However, the architecture might limit the ability to fully extract information from the features, thus still performing worse than SVM.

2 MLP - Directly on images

2.1 FashionMNIST

Batch size: 256, Training Epochs: 5, Learning Rate: 0.001, Loss function: Cross Entropy loss, Optimizer: Adam, Dropout factor: 0.2.

Layer	Output Shape
Input	(28, 28, 1)
Flatten	28 * 28
Linear + ReLU	256
Dropout	256
Linear + ReLU	128
Dropout	128
Linear	10

Table 3: MLP directly on images architecture - FashionMNIST

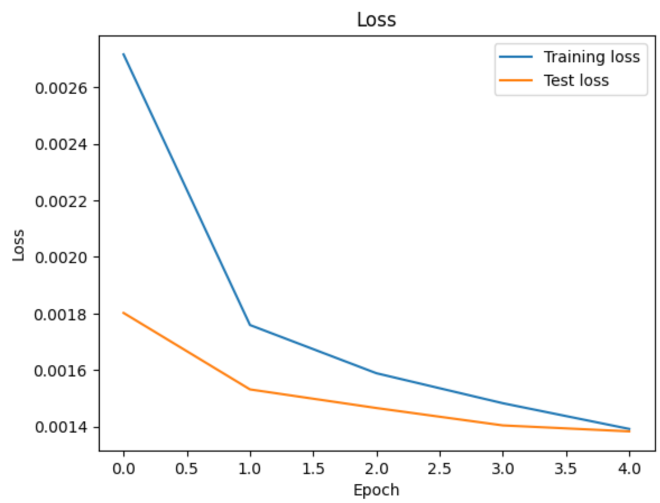

Figure 5: Loss Function for Direct-MLP - FashionMNIST

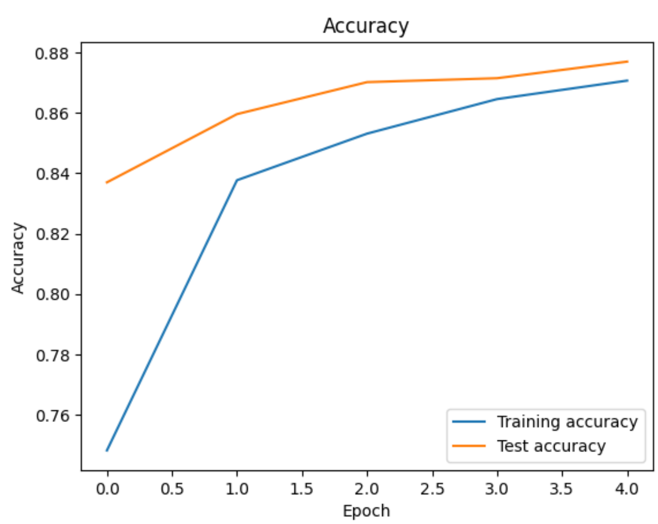

Figure 6: Accuracy for Direct-MLP - FashionMNIST

The Direct MLP, which operates directly on image data, shows a significant improvement in accuracy compared to the Feature MLP. However, this comes at the cost of increased computational time. Despite being a non-convolutional model, it performs reasonably well, likely due to the relatively simple nature of the FashionMNIST dataset.

By gradually reducing the number of neurons on each layer, Direct MLP is able to extract significantly more information than Feature MLP (0.88 for the Direct MLP compared to 0.79 for Feature MLP). Also, another relevant observation is the lack of overfitting of both MLP architectures, as we can clearly see that the accuracy and loss become smaller as the number of epochs increases, without spikes in their validation process.

2.2 Fruits 360

Batch size: 256, Training Epochs: 5, Learning Rate: 0.001, Loss function: Cross Entropy loss, Optimizer: Adam, Dropout factor: 0.2.

Layer	Output Shape
Input	(100, 100, 3)
Resize	(32, 32, 3)
Flatten	32 * 32 * 3
Linear + ReLU	1024
Dropout	1024
Linear + ReLU	256
Dropout	256
Linear	141

Table 4: MLP directly on images architecture - Fruits 360

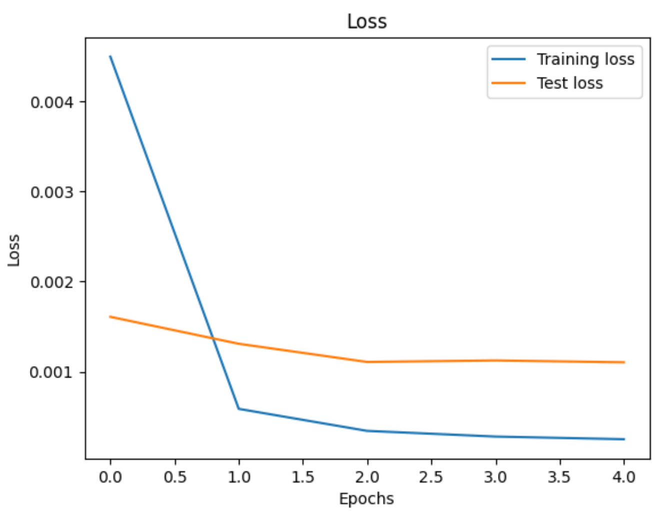

Figure 7: Loss Function for Direct-MLP - Fruits 360

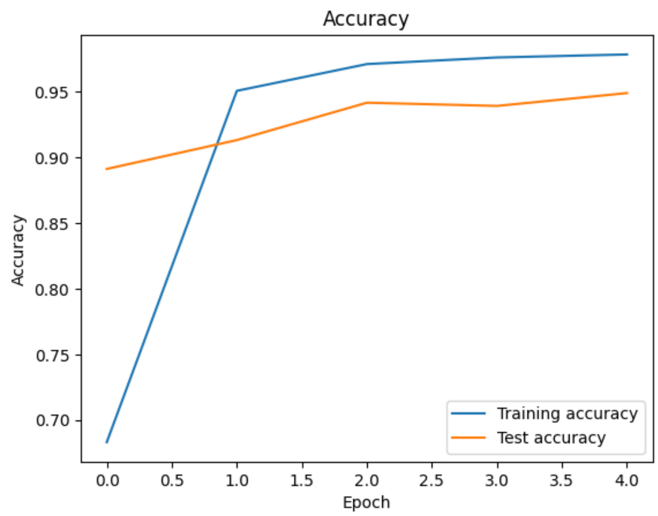

Figure 8: Accuracy for Direct-MLP - Fruits 360

The Direct MLP significantly outperforms the Feature MLP, showcasing the benefits of training directly on raw image data rather than relying on predefined features. Its accuracy surpasses even some convolutional models, such as DenseConvNet with augmentation, suggesting that it effectively learns to classify images in this dataset. While it takes longer than the Feature MLP, the computational cost is still relatively low compared to CNN-based models.

Similar to the FashionMNIST, Direct MLP on the Fruits 360 is able to better understand the meaning behind each image and create relevant labels. Also, another relevant observation is the lack of overfitting of both MLP architectures, as we can clearly see that the accuracy and loss become smaller as the number of epochs increase, without spikes in their validation process.

3 Dense CNN

3.1 FashionMNIST

Batch size: 256, Training Epochs: 20, Learning Rate: 0.001, Loss function: Cross Entropy loss, Optimizer: Adam.

Layer	Output Shape
Input	(28, 28, 1)
Resize	(32, 32, 1)
Convolution 2D	(16, 16, 16)
Batch normalization + ReLU	(16, 16, 16)
Convolution Block 1	(16, 16, 64)
Convolution Block 2	(8, 8, 128)
Convolution Block 3	(4, 4, 256)
Average Pooling	(1, 1, 256)
Flatten	256
Linear	10

Table 5: DenseConvNet architecture - Fruits 360

3.1.1 No augmented images

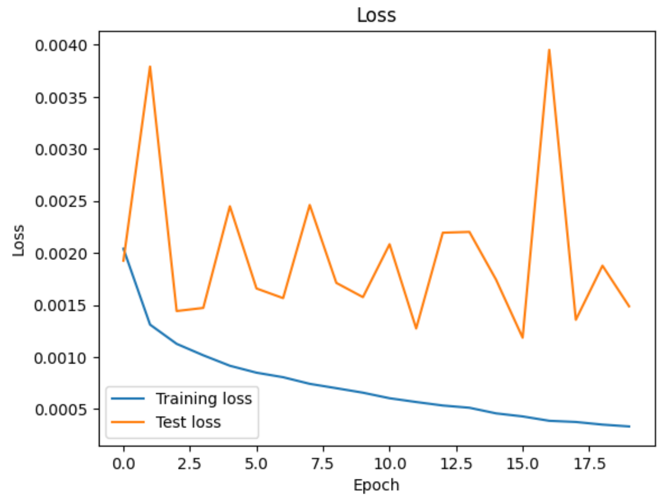

Figure 9: Loss Function for DenseConvNet (no augmented images) - Fruits 360

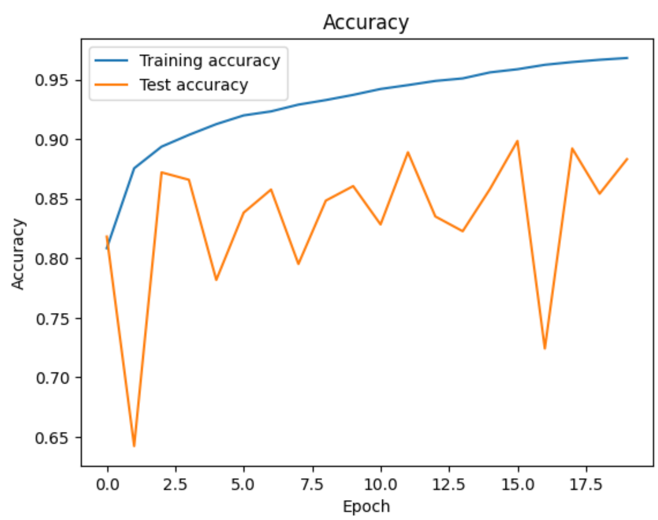

Figure 10: Accuracy for DenseConvNet (no augmented images) - FashionMNIST

3.1.2 Augmented images

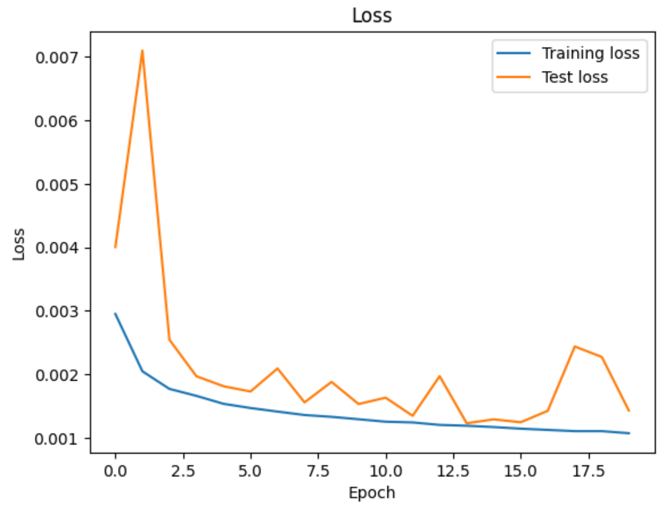

Figure 11: Loss Function for DenseConvNet (augmented images) - FashionMNIST

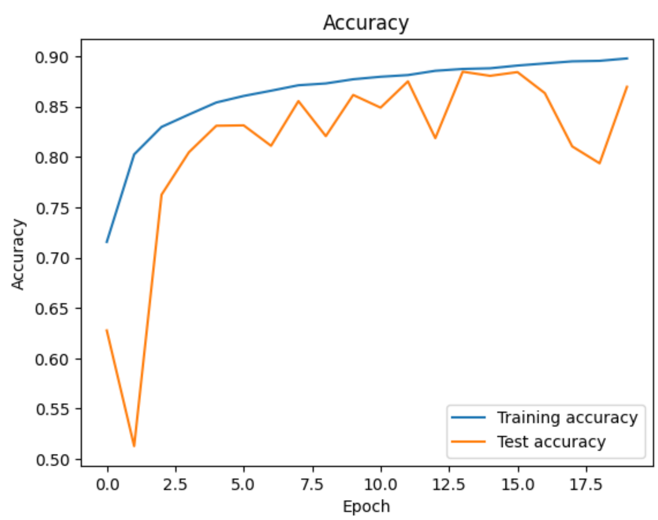

Figure 12: Accuracy for DenseConvNet (augmented images) - FashionMNIST

3.1.3 Comparison between the two approaches

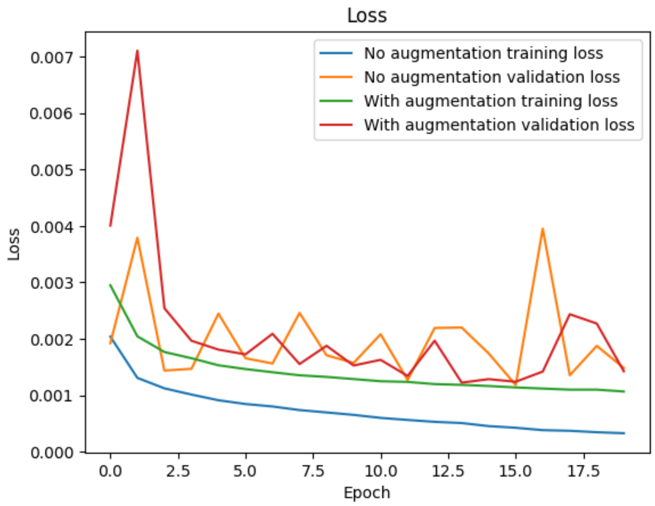

Figure 13: Loss Comparison - FashionMNIST

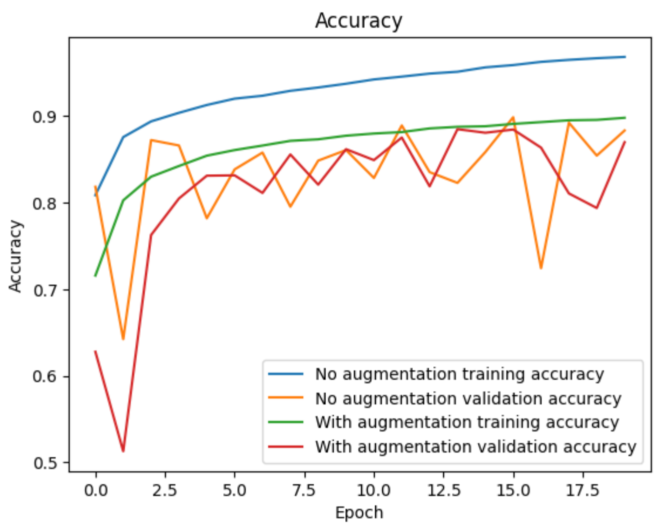

Figure 14: Accuracy Comparison - FashionMNIST

The data augmentation was done by adding random horizontal flips, random rotations, and random crops with padding.

DenseConvNet achieves slightly better accuracy than Direct MLP, benefiting from its convolutional architecture, which is better suited for image tasks. However, this improvement comes at a substantial computational cost. The lack of data augmentation might limit its generalization, leading to only a marginal increase in accuracy.

Interestingly, the addition of data augmentation in DenseConvNet reduces accuracy slightly compared to the no-augmentation variant. This counterintuitive result might be due to the network overfitting the original dataset in the no-augmentation scenario, while augmentation introduces variability that challenges the network to generalize better. The increase in training time reflects the computational overhead introduced by augmentation.

3.2 Fruits 360

Batch size: 256, Training Epochs: 20, Learning Rate: 0.001, Loss function: Cross Entropy loss, Optimizer: Adam.

Layer	Output Shape
Input	(28, 28, 1)
Resize	(32, 32, 1)
Convolution 2D	(16, 16, 16)
Batch normalization + ReLU	(16, 16, 16)
Convolution Block 1	(16, 16, 64)
Convolution Block 2	(8, 8, 128)
Convolution Block 3	(4, 4, 256)
Average Pooling	(1, 1, 256)
Flatten	256
Linear	141

Table 6: DenseConvNet architecture - Fruits 360

3.2.1 No augmented images

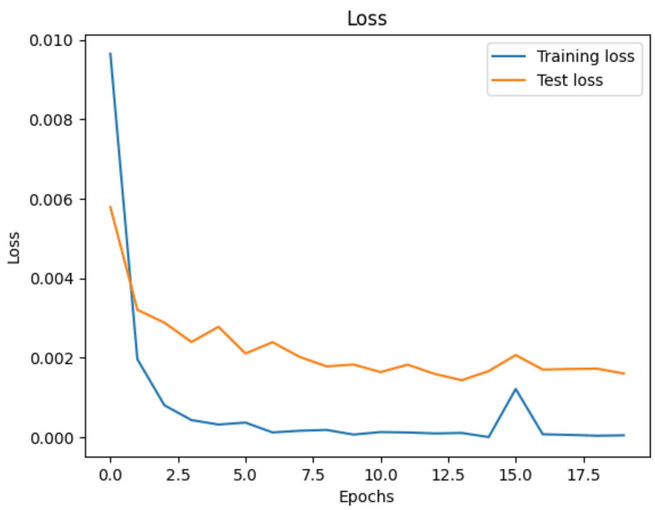

Figure 15: Loss Function for DenseConvNet (no augmented images) - Fruits 360

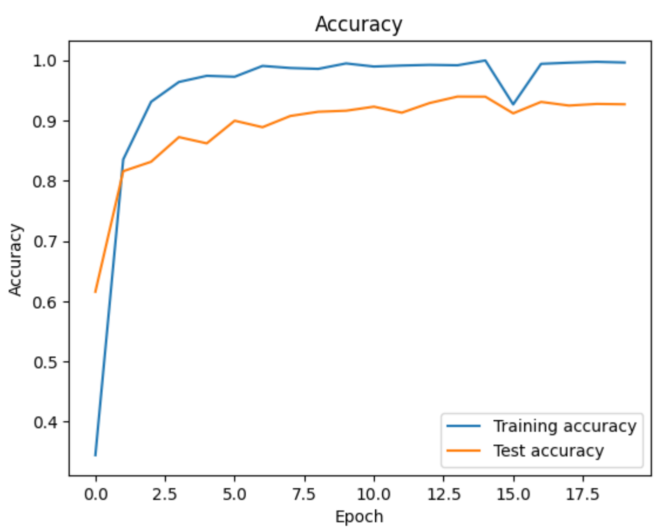

Figure 16: Accuracy for DenseConvNet (no augmented images) - Fruits 360

3.2.2 Augmented images

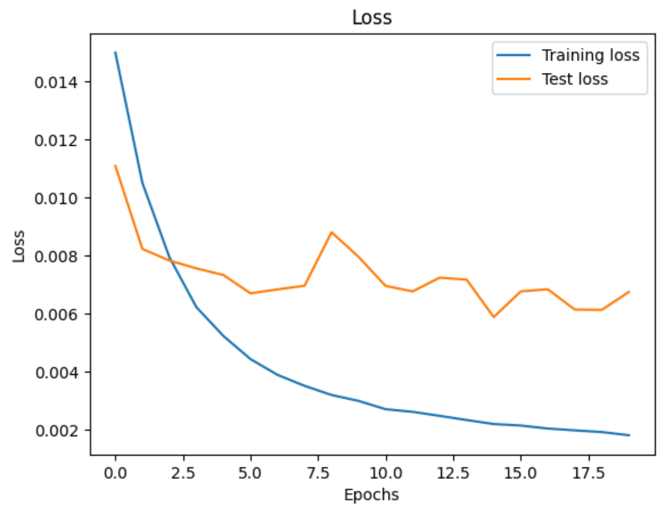

Figure 17: Loss Function for DenseConvNet (augmented images) - Fruits 360

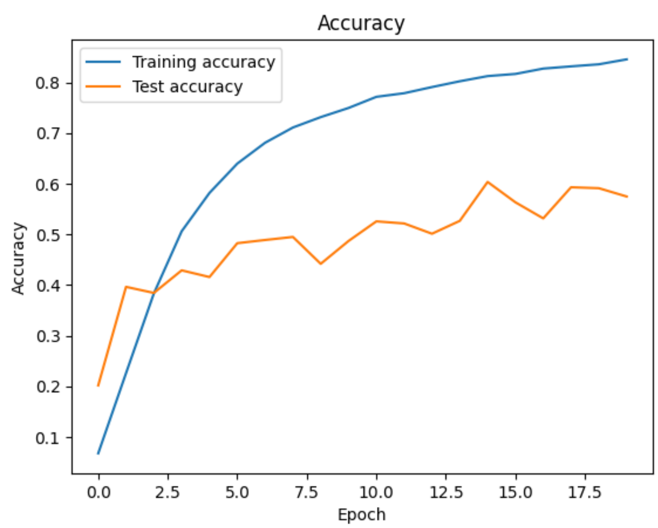

Figure 18: Accuracy for DenseConvNet (augmented images) - Fruits 360

3.2.3 Comparison between the two approaches

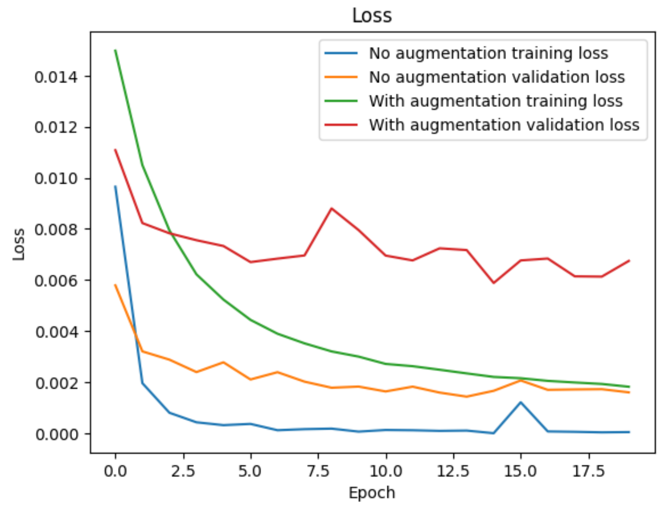

Figure 19: Loss Comparison - Fruits 360

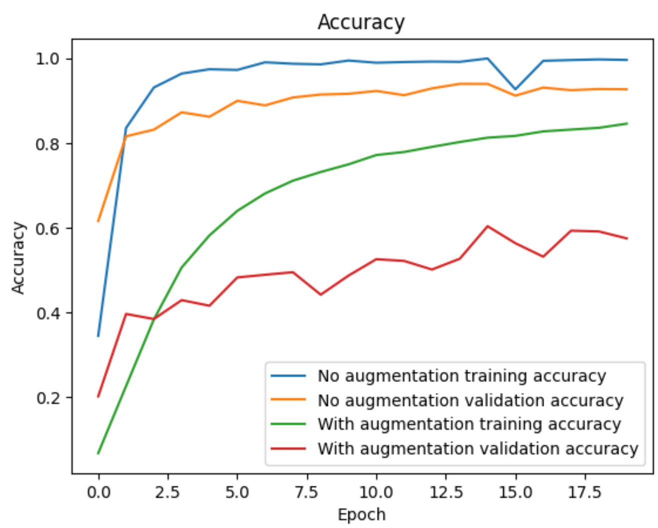

Figure 20: Accuracy Comparison - Fruits 360

The data augmentation was done by adding random horizontal flips, random rotations, and random crops with padding.

DenseConvNet (without augmentation) achieves reasonable accuracy, outperforming Feature MLP but lagging behind Direct MLP and ResNet. Its reliance on convolutional layers enables better feature extraction, but the lack of data augmentation may limit its ability to generalize. Its training time is considerably higher than the MLPs, making it less efficient for scenarios requiring quick iterations.

Unexpectedly, DenseConvNet with augmentation shows the lowest accuracy among all models, even lower than Feature MLP. This poor performance could stem from improper augmentation strategies, which might have introduced noise or distortions that made the learning process more challenging. The slight increase in training time reflects the additional computational cost of applying augmentations.

4 Resnet 18

4.1 FashionMNIST

Layer	Output Shape
Input	(28, 28, 1)
Resize	(32, 32, 1)
Resnet 18	256
Linear	10

Table 7: Resnet 18 architecture - FashionMNIST

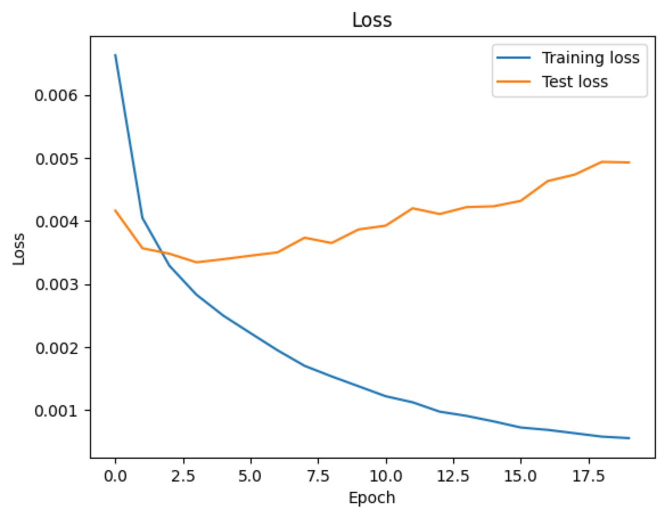

Figure 21: Loss Function for Resnet - FashionMNIST

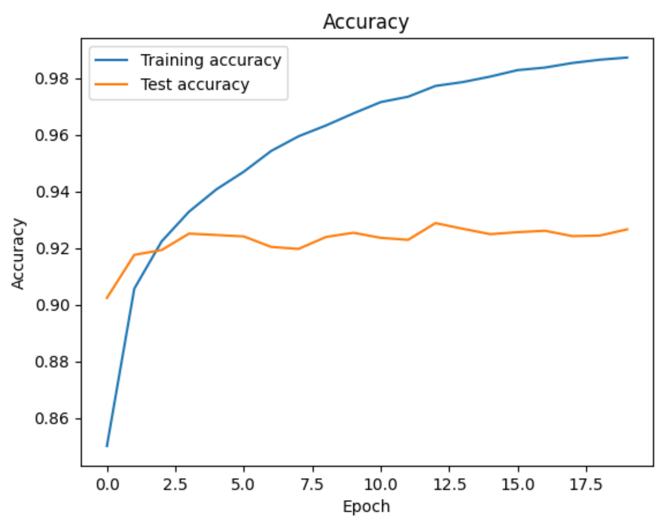

Figure 22: Accuracy for Resnet - FashionMNIST

ResNet achieves the highest accuracy of 0.93, outperforming all other models by a significant margin. Its use of residual connections likely aids in capturing complex features without vanishing gradients, resulting in superior performance. Despite its higher computational time compared to MLPs, it is more efficient than DenseConvNet, making it the best choice in terms of accuracy and efficiency trade-offs.

4.2 Fruits 360

Layer	Output Shape
Input	(28, 28, 1)
Resize	(32, 32, 1)
Resnet 18	256
Linear	141

Table 8: Resnet 18 architecture - Fruits 360

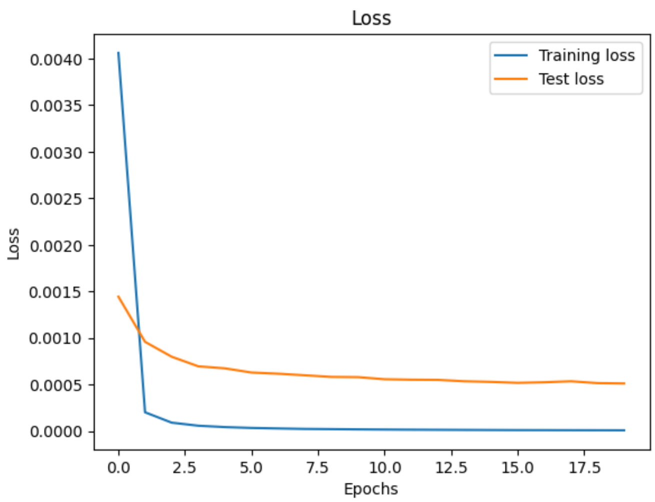

Figure 23: Loss Function for Resnet - Fruits 360

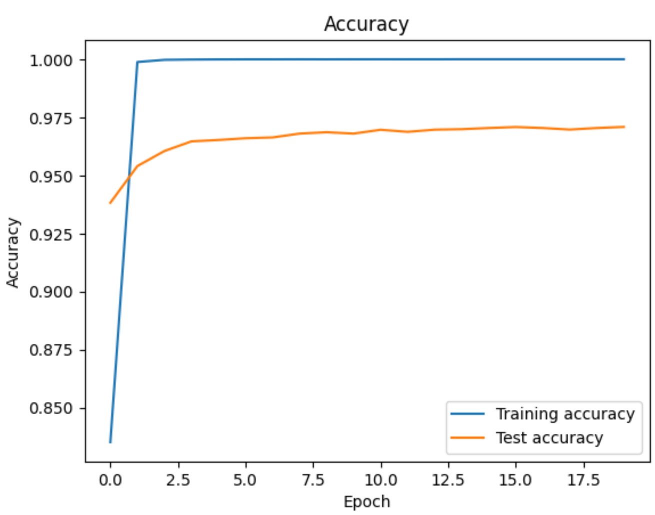

Figure 24: Accuracy for Resnet - Fruits 360

ResNet performs exceptionally well, achieving an accuracy of 0.93, nearly matching the Direct MLP. ResNet's use of residual connections allows it to effectively learn complex patterns without the vanishing gradient problem. While its training time is significant, it is more efficient than DenseConvNet, making it a strong candidate for high-accuracy applications.

5 Comparison

5.1 FashionMNIST

Architecture	Accuracy	Time (s)
Feature MLP	78.84%	10
Direct MLP	87.98%	26
DenseConvNet (no aug)	88.32%	38 * 60
DenseConvNet (aug)	86.97%	41 * 60
Resnet	92.66%	32 * 60

Table 9: Architecture comparison - test accuracy - FashionMNIST

ResNet outperforms all other architectures in terms of accuracy, followed by DenseConvNet (no augmentation) and Direct MLP. The Feature MLP performs poorly, highlighting the importance of convolutional architectures for image-based tasks.

Feature MLP and Direct MLP are significantly faster than convolutional models, making them suitable for scenarios where computational resources are constrained. ResNet offers the best balance between time and accuracy, being more efficient than DenseConvNet while achieving superior accuracy.

Data augmentation does not consistently improve performance in this experiment. Further hyperparameter tuning or a different augmentation strategy might be needed to realize its benefits.

For FashionMNIST, ResNet is the most effective architecture, delivering the highest accuracy while being computationally efficient relative to DenseConvNet. Feature MLP and Direct MLP provide faster alternatives but lack the performance required for high-accuracy applications. DenseConvNet, despite being effective, is outshined by ResNet in both accuracy and computational cost, suggesting that ResNet is the optimal choice for this dataset and task.

5.2 Fruits 360

Architecture	Accuracy	Time (s)
Feature MLP	72.89%	18
Direct MLP	93.19%	130
DenseConvNet (no aug)	88.49%	60 * 60
DenseConvNet (aug)	57.34%	66 * 60
Resnet	92.84%	52 * 60

Table 10: Architecture comparison - test accuracy - Fruits 360

The Direct MLP achieves the highest accuracy 0.931, slightly outperforming ResNet 0.928, and both significantly surpass DenseConvNet (no aug) and Feature MLP. DenseConvNet with augmentation performs surprisingly poorly, suggesting the need for better augmentation strategies or hyperparameter tuning.

Feature MLP and Direct MLP are the fastest models, with Direct MLP offering the best trade-off between accuracy and time. Among convolutional models, ResNet is more efficient than DenseConvNet, delivering high accuracy at a lower computational cost.

The negative impact of augmentation on DenseConvNet highlights the importance of carefully designing augmentation pipelines to enhance, rather than hinder, performance.

For the Fruits 360 dataset: Direct MLP is the most effective architecture, achieving the highest accuracy with relatively low training time, making it ideal for practical use cases. ResNet is a close second, offering a robust balance of accuracy and computational efficiency, particularly for scenarios requiring deep convolutional architectures. Feature MLP is suitable for rapid prototyping or low-resource applications where accuracy is less critical. The DenseConvNet results indicate potential for improvement, particularly with optimized augmentation strategies.