Indian Institute of Technology, Jodhpur

Assignment 1:

Stress-Testing of Convolutional Neural Networks

Submitted By: Group 25

Members:

M25MAC004 Deepak Kumar
M25MAC008 Sakshi Singh
M25MAC014 Vidushi Mittal
M25MAC016 Zobiya Khalid

Course Name: Deep Learning

Instructor: *Dr. Angshuman Paul*

Date of Submission: *15 February, 2026*

1. INTRODUCTION

This project requires performing experimentation on different CNN models and selecting one for performing stress testing from a given set of datasets. We are using the Fashion-MNIST Dataset and comparing with 15 epochs using RESNET, VGG16, VGG19, and a custom CNN model that we developed. Further, we Performed Failure detection and Explainability analysis using Grad-CAM. The constrained optimization is performed using data augmentation, more on it later in the report.

2. DATASET

We used the Fashion-MNIST dataset, a benchmark image classification dataset designed as a slightly challenging alternative to the original MNIST digit dataset. Fashion-MNIST consists of 70,000 grayscale images of size 28×28 pixels, categorized into 10 clothing classes: T-shirt/top, Trouser, Pullover, Dress, Coat, Sandal, Shirt, Sneaker, Bag, and Ankle boot.

The dataset is provided with an official split of 60,000 training images and 10,000 test images, which was followed in all experiments.

```
Dataset Name: Fashion-MNIST - Details of the Train data
Number of training samples: 60000
Image dimensions: torch.Size([1, 28, 28]) (C, H, W)
Number of classes: 10
Classes: ['T-shirt/top', 'Trouser', 'Pullover', 'Dress', 'Coat', 'Sandal', 'Shirt', 'Sneaker', 'Bag', 'Ankle boot']
```

We are using 50,000 samples for training and all 10000 samples for testing split.

3. BASELINE MODEL ARCHITECTURE

Before finalizing the baseline architecture, we conducted controlled experiments using deeper standard models, namely VGG-16, VGG-19, and ResNet-18, each trained from scratch keeping the weights as none for 15 epochs on Fashion-MNIST using identical optimization settings. While these architectures achieved reasonable test accuracies (VGG-16: 92.03%, VGG-19: 91.78%, ResNet-18: 90.38%), However, these models exhibited overfitting behaviour (ResNet 18). Given the relatively small input resolution (28×28) and moderate dataset complexity, the high-capacity architecture appeared unnecessary and reduced the interpretability of failure patterns. Therefore, a custom CNN with controlled capacity was selected as the baseline model for all subsequent experiments.

Preliminary Architecture Experiments

While these architectures achieved reasonable test accuracies (VGG-16: 92.03%, VGG-19: 91.78%, ResNet-18: 90.38%), their training dynamics revealed clear overfitting behavior. In particular, ResNet-18 exhibited a widening gap between training and validation loss after the initial epochs, with validation loss increasing despite continued improvement in training accuracy, as shown in figure 2,3,4.

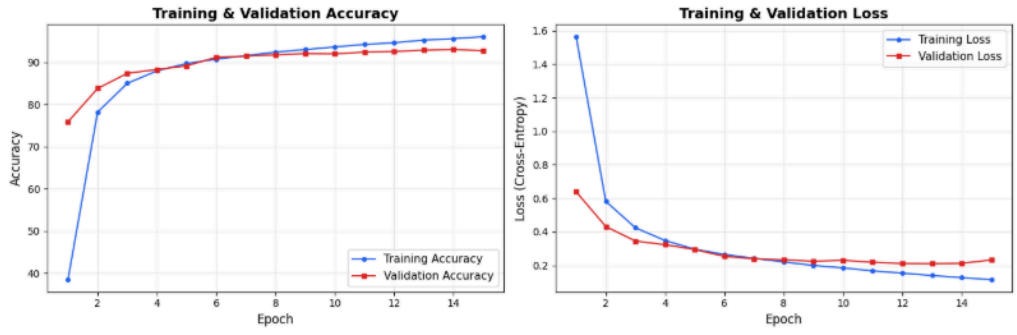


Figure 2. VGG16 on Fashion MNIST- Test accuracy(92.3%)

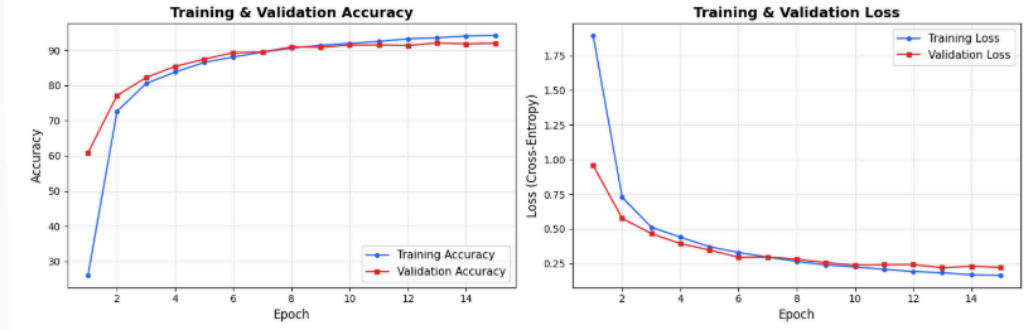


Figure 3. VGG19 on Fashion MNIST- Test accuracy(91.78%)

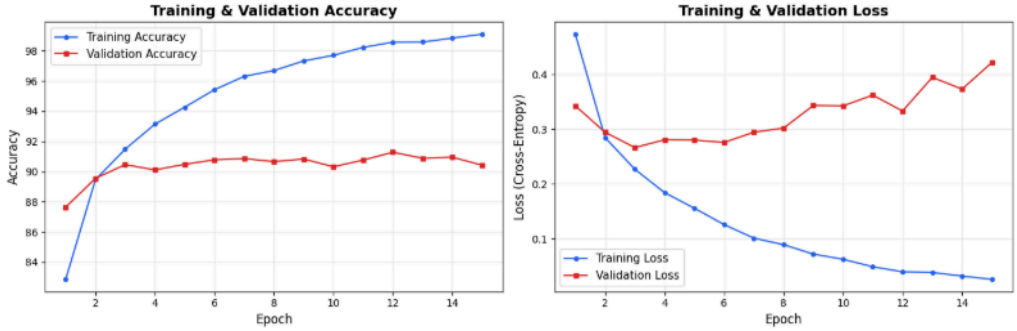


Figure 4. ResNet18 on Fashion MNIST- Test accuracy(90.38%)

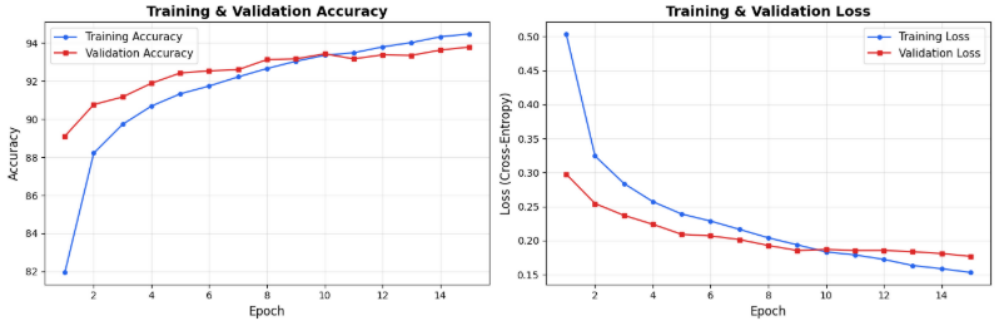


Figure 5. Custom baseline CNN on Fashion MNIST- Test accuracy(92.97%)

This suggests that the high model capacity was not well matched to the relatively low-resolution (28×28) grayscale dataset. The high-capacity architecture appeared unnecessary and reduced interpretability of failure patterns. We selected a custom CNN- (test accuracy 92.97%) with controlled capacity as the baseline architecture for all subsequent experiments.

Custom CNN Baseline Model Architecture:

Stage,Operations,Output Shape,Parameters / Key Features

Block 1,"2× (3x3 Conv, 32 filters) + MaxPool + Dropout","(32,14,14)",Captures low-level spatial features.

Block 2,"2× (3x3 Conv, 64 filters) + MaxPool + Dropout","(64,7,7)",Extracts high-level semantic patterns.

Classifier, Flatten + Dense (256 units) + Dropout + Dense (10),"(10,)","Maps 3,136 features to class probabilities.".

4. BASELINE TRAINING RESULTS

The baseline custom CNN was trained for 15 epochs using the official Fashion-MNIST training split. Figure 5 shows the training and validation accuracy curves and the corresponding loss curves. All plots were generated using our implementation.

The model achieved a **final test accuracy of 93.21%**, with a corresponding test loss of 0.2390.

Analysis:

Training accuracy increased steadily from 83.69% in the first epoch to 97.44% by epoch 30. Validation

Overfitting:

A visible gap emerges between training and validation accuracy after approximately epoch 12. By epoch 15:

- Training Accuracy ≈ 97.44%
- Validation Accuracy ≈ 93.84%
- Test Accuracy = 93.21%

The ~3-4% gap suggests moderate overfitting, but not severe memorisation. The use of Batch Normalization and Dropout likely mitigates stronger overfitting effects.

Validation accuracy remains stable rather than collapsing, indicating that the model retains reasonable generalization ability despite increasing training performance.

5. FAILURE CASE DISCOVERY

The objective of this section is to systematically analyze instances where the baseline CNN produces incorrect predictions. Out of 10,000 test images, we found 703 incorrect predictions. Rather than focusing solely on aggregate accuracy, we have mentioned individual failure cases with the highest uncertainty values, and some high confidence in incorrect predictions. For each case, we present the input image, ground-truth label, predicted label with confidence score, and a hypothesis explaining the likely cause of failure based on model behaviour and dataset characteristics.

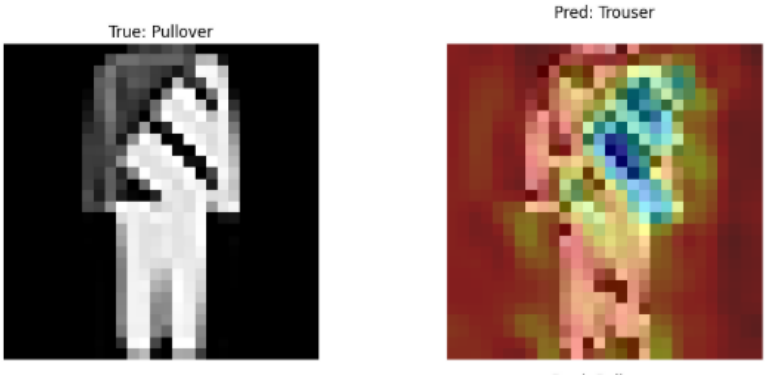


Figure: Failure case 1

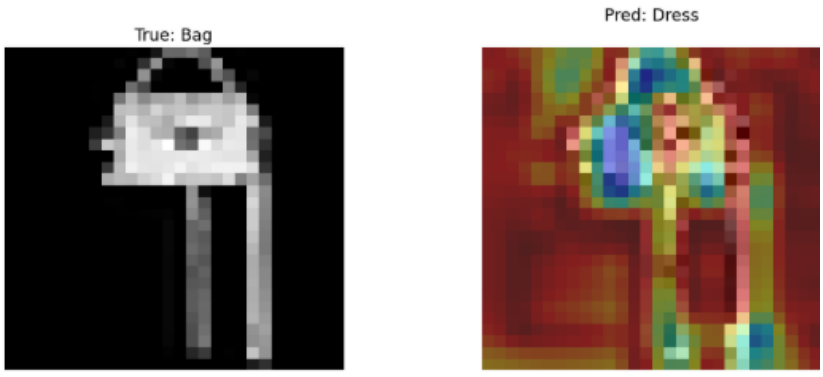


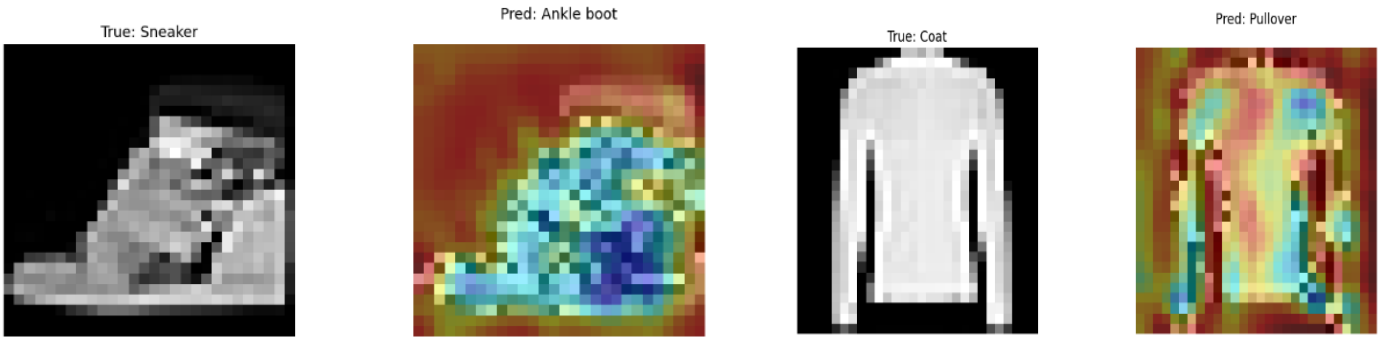
Figure: Failure case 2

Section 1: Case Details	Section 2: Observed Model Behaviour	Section 3: Failure Hypothesis
Case 1: Pullover → Trouser Fig 1 • Confidence: **99.00%** • Entropy: 0.0652	**Overconfident Error:** Model shows near-total certainty despite distinct categories. Saliency shows activation on **central vertical regions** (lower half).	**Silhouette Reliance:** Model relies on dominant vertical contours/global shape rather than semantic cues (sleeves/neckline) which are lost in 28×28 resolution.
Case 2: Bag → Dress • Confidence: **19.54%** • Entropy: 2.0499	**High Uncertainty:** Lowest confidence and highest entropy. The network was effectively "guessing" among plausible categories.	**Shape Abstraction:** Handbag handles resemble the vertical silhouette of a dress. Loss of texture/thickness cues causes confusion between vertically structured objects.

Case 3: Sneaker → Ankle boot • Confidence: **99.86**	**Feature Misalignment:** Focus is on the **sole and mid-body** rather than ankle height. Extremely peaked probability distribution. • Entropy: 0.0108%	**Geometric Overlap:** Sneaker and boots share similar lower geometry. Model prioritizes horizontal features over the vertical extension cues needed to distinguish them.
Case 4: Coat → Pullover • Confidence: **98.98%** • Entropy: 0.0648	**Silhouette Collapse:** Saliency shows widespread activation across torso/shoulders with no focus on lapels or button placement.	**Information Loss:** Low resolution suppresses layering/depth cues. Model encodes both as "generic long-sleeve upper garments" based on global contour.

Figure: Failure Case 3

Figure Failure Case 4:



6. EXPLAINABILITY ANALYSIS

To better understand the failure patterns identified in Section 5, we applied **Grad-CAM (Gradient-weighted Class Activation Mapping)** to visualize which spatial regions influenced the model's predictions. While the baseline CNN achieves high overall accuracy, several high-confidence errors suggest that the network may rely on simplified structural cues rather than semantically meaningful garment features. Grad-CAM was therefore used to verify whether the model focuses on discriminative regions or dominant silhouettes

6.1 Methodology and Mathematical Formulation

Grad-CAM generates a class-specific localization map by combining feature maps from the final convolutional layer with gradients of the target class score. Let Ak denote the k th feature map of the selected convolutional layer and yc represent the score for class c. The importance weight for each channel is computed as:

$$\alpha_k^c = \frac{1}{Z} \sum_i \sum_j \frac{\partial y^c}{\partial A_{ij}^k}$$

where Z is the spatial size of the feature map. The Grad-CAM heatmap is then obtained by:

$$L_{\text{Grad-CAM}}^c = \text{ReLU}\left(\sum_k \alpha_k^c A^k \right)$$

In our implementation, Grad-CAM was applied to the final convolutional stage (Block-2) of the custom CNN, since deeper layers encode higher-level semantic features while retaining spatial structure.

6.2 Observed Attention Patterns

The generated heatmaps reveal attention behaviors consistent with the failure analysis:

- **Silhouette-Dominant Focus:** For garment confusions such as Pullover vs Trouser or Coat vs Pullover, activation concentrated along central torso or elongated vertical regions rather than

sleeve or layering details.

- **Footwear Geometry Bias:** In Sneaker vs Ankle boot errors, attention primarily highlighted the sole and mid-body regions, while ankle height received limited emphasis.

- **Diffuse Attention in Uncertain Cases:** High-entropy predictions showed broader and less localized heatmaps, indicating weaker feature representations.

7. CONSTRAINED IMPROVEMENT ANALYSIS: DATA AUGMENTATION

To address the observed overfitting behavior and systematic confusion between structurally similar classes, we introduced a single constrained modification: data augmentation applied exclusively to the training set. No architectural changes were made to ensure that improvements could be attributed solely to the augmentation strategy.

The modified training transform included:

- Random horizontal flipping (p = 0.5)
- Random rotation (±10 degrees)
- Standard normalization

7.1 Quantitative Results

After 30 epochs of training with augmentation, the model achieved:

- Final Test Accuracy (Baseline): 92.97%
- Final Test Accuracy (With Augmentation): 93.24%

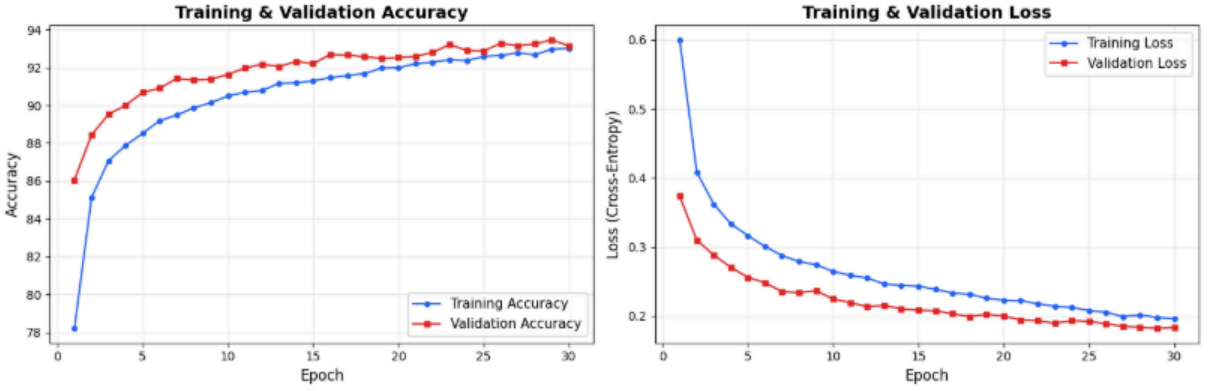


Figure 10. Custom baseline CNN + Augmentation -Test accuracy(93.24%)

7.2 Effect of Augmentation on Previously Identified Failure Cases

To evaluate the impact of data augmentation beyond aggregate accuracy, we re-examined the previously identified failure cases under the augmented model. While overall test accuracy improved modestly , the effect on individual high-confidence misclassifications was limited. Some ambiguous garment confusions exhibited slightly reduced prediction confidence, indicating improved uncertainty calibration. However, systematic footwear confusions (e.g., Sneaker vs Ankle boot, shirt vs coat) persisted, particularly in cases where silhouette similarity dominates. This suggests that augmentation enhances tolerance to minor geometric variations but does not fundamentally alter the learned structural representations responsible for class overlap.

A key trade-off observed was slower convergence during early epochs, as the augmented model required additional training time to achieve comparable training accuracy. Nevertheless, this behavior reflects improved regularization rather than degradation. Overall, augmentation provides incremental robustness gains without increasing model complexity, but structural feature ambiguity remains an inherent challenge at low resolution.

7.3 Interpretation

Data augmentation provides measurable but limited improvement in overall performance. The modest accuracy gain suggests that the primary limitation may not be solely due to overfitting, but rather due to intrinsic information loss in low-resolution grayscale images. Nevertheless, augmentation improves training stability and slightly enhances generalization capacity without increasing model complexity.

9. CONCLUSION

This study investigated CNN behavior on Fashion-MNIST beyond aggregate accuracy metrics. While the custom baseline model achieved strong performance (≈93% test accuracy), systematic failure analysis revealed consistent structural confusions, particularly among footwear and upper-body garment categories. Explainability techniques showed that the network often relies on coarse silhouette and dominant contour features, while underemphasizing fine-grained structural cues such as ankle height or garment layering. A constrained improvement through data augmentation provided modest gains in generalization but did not eliminate class overlap caused by low-resolution ambiguity. Overall, the results demonstrate that high accuracy alone does not guarantee robust or semantically grounded decision-making, highlighting the importance of failure analysis and interpretability in evaluating deep learning models.

10. Git Repository

Repo link: https://github.com/Vidushi-Mittal/Stress-Testing-of-Convolutional-Neural-Networks_A1_DL