Approach to Signature Verification System

Objective

The goal of this task is to develop an intelligent signature verification system that can distinguish between genuine human signatures and invalid hand-drawn inputs. Invalid inputs may include scribbles, geometric shapes, symbols, numbers, or any non-signature-like marks. The challenge is to generalize well beyond simple classification and ensure that the system does not overfit dataset-specific patterns.

Solution Approach

To address the problem, I designed a hybrid solution consisting of:

1. **Autoencoder for Feature Learning** – To capture signature patterns and detect anomalies.
2. **Convolutional Neural Network (CNN) Classifier** – To classify signatures as valid or invalid.
3. **Anomaly Detection Models** – To detect outliers based on learned representations.

Dataset Preparation

- **Proper Signatures**: A dataset containing genuine human signatures with natural variations.
- **Invalid Inputs**: A dataset with random scribbles, geometric shapes, numbers, and non-Latin characters.
- **Synthetic Data Augmentation**: Using noise augmentation and random stroke generation to enhance the robustness of the model.

Preprocessing Steps

- **Grayscale Conversion**: Convert images to grayscale for uniformity.
- **Contrast Enhancement**: Apply histogram equalization to improve stroke visibility.
- **Noise Removal**: Use Gaussian blur to smoothen images.

- **Adaptive Thresholding**: Enhance signature strokes while eliminating background noise.
- **Morphological Operations**: Remove small noise components.
- **Resizing and Normalization**: Convert images to (224x224) size and normalize pixel values between [0,1].

Model Architecture

Autoencoder (Feature Learning)

- Encoder with convolutional layers to extract signature patterns.
- Bottleneck latent representation (128-dimensional vector).
- Decoder to reconstruct the original signature image.
- **Loss Function**: Combination of binary cross-entropy and Structural Similarity Index (SSIM) loss for better reconstruction quality.

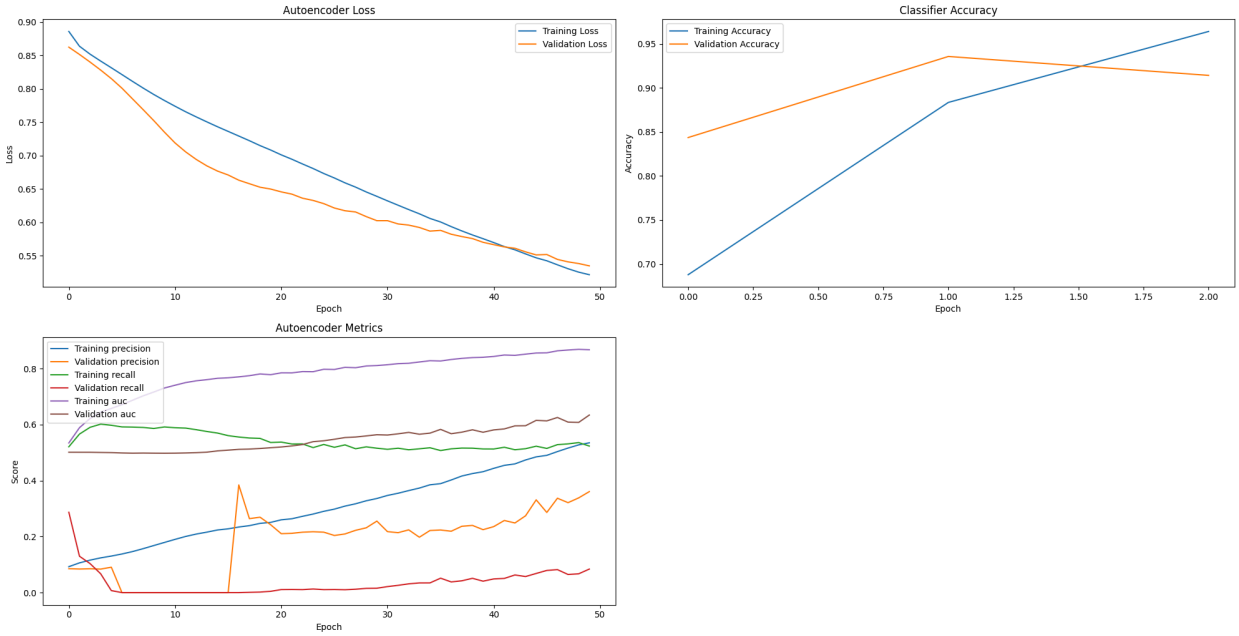
CNN Classifier

- A convolutional network trained on labeled data to classify signatures as valid or invalid.
- **Architecture:**
    - Convolutional layers for feature extraction.
    - Max pooling for dimensionality reduction.
    - Fully connected layers for classification.
    - Dropout for regularization.

Anomaly Detection Models

- **Isolation Forest**: Trained on valid signatures to detect anomalies.
- **One-Class SVM**: Learns decision boundaries for signature validity.

Training Results



False Positives: And when invalid curves are more like that synthetic valid data and the model reconstructs it .

False Negatives: As I included a lot of synthetic_valid signatures that are made with curves. So in case of complex signatures. So in some complex signatures the autoencoder model struggles to reconstruct it.