R.Vekeshhari (ch.sc.u4cys23051)

Yeshwanth Raj (ch.sc.u4cys23052)

Machine Learning in Cyber Security (20CYS215)

Image Feature Extraction and Classification Report

1. Introduction

Feature extraction plays a crucial role in image classification by reducing dimensionality while preserving essential image characteristics. This study evaluates and compares four widely used feature extraction techniques—Histogram of Oriented Gradients (HOG), Local Binary Pattern (LBP), Edge Detection (Canny), and Deep Learning (VGG16)—on the MNIST and EMNIST datasets.

The primary objectives include:

- Evaluating the effectiveness of different feature extraction methods for digit classification.
- Comparing classification performance using traditional machine learning models (Logistic Regression, K-Nearest Neighbours (KNN), Decision Trees, and Random Forests).
- Assessing robustness by introducing noise, rotation, and blur.
- Investigating model generalization using the EMNIST dataset.

3. Experimentation

Dataset:

- MNIST: 70,000 handwritten digit images (0-9), 28×28 pixels.
- EMNIST: Extended MNIST dataset with letters and digits, used for model generalization.

Feature Extraction Methods:

1. HOG (Histogram of Oriented Gradients)
2. LBP (Local Binary Pattern)
3. Canny Edge Detection
4. Deep Learning (VGG16 Pre-trained Model)

Classification Models:

- Logistic Regression (for HOG)
- KNN (for LBP)
- Decision Tree (for Edge)
- Random Forest (for Deep Learning Features)

4. Results and Analysis

4.1 Feature Extraction and Training Times

Method	Feature Extraction Time (s)	Training Time (s)
HOG (Logistic Regression)	60.20	13.08
LBP (KNN)	19.72	0.14

Method	Feature Extraction Time (s)	Training Time (s)
Edge Detection (Decision Tree)	27.21	31.44
Deep Learning (VGG16) (Random Forest)	858.24	116.07

- Deep learning took the longest for feature extraction and training due to the high computational complexity of convolutional networks.
- KNN was the fastest because it involves simple distance calculations rather than an extensive training process.

4.2 Classification Performance on MNIST

Method	Accuracy	Precision	Recall	F1-score
HOG (Logistic Regression)	0.9723	0.9724	0.9723	0.9723
LBP (KNN)	0.8326	0.8470	0.8326	0.8297
Edge Detection (Decision Tree)	0.7481	0.7474	0.7481	0.7475
Deep Learning (Random Forest)	0.9207	0.9207	0.9207	0.9205

Key Observations:

- HOG achieved the highest accuracy (97.23%), proving effective for digit classification.
- Deep Learning (VGG16) also performed well (92.07%), though slightly behind HOG.
- LBP and Edge Detection underperformed, with LBP at 83.26% and Edge Detection at 74.81%, indicating their limitations for digit recognition.

4.3 Robustness Analysis (Noisy, Rotated, and Blurred Images)

Method	Noisy Data	Rotated Data	Blurred Data
HOG (Logistic Regression)	0.1041	0.1334	0.1242
LBP (KNN)	0.1135	0.1135	0.1135
Edge Detection (Decision Tree)	0.3139	0.4829	0.5489
Deep Learning (Random Forest)	0.0982	0.2574	0.2391

Key Observations:

- HOG and Deep Learning suffered drastic performance drops under noise, blur, and rotation.
- Edge Detection was the most robust, showing higher accuracy (54.89%) on blurred images.
- LBP performed poorly across all distortions, showing minimal adaptation to noise, blur, or rotation.

4.4 Generalization to EMNIST (New Dataset)

Method	Accuracy on EMNIST
HOG (Logistic Regression)	0.1009
LBP (KNN)	0.1000
Edge Detection (Decision Tree)	0.0794
Deep Learning (Random Forest)	0.2067

Key Observations:

- Deep Learning generalized the best (20.67%), indicating it learned meaningful representations beyond MNIST.
- HOG and LBP had poor generalization, dropping to ~10%, suggesting overfitting to MNIST digits.
- Edge Detection performed the worst (7.94%), highlighting its lack of adaptability to new datasets.

5. Conclusion

This study evaluated four feature extraction methods for image classification using MNIST and EMNIST. HOG outperformed all others on MNIST, while Deep Learning showed the best generalization to EMNIST.

- For best accuracy, HOG with Logistic Regression is recommended for MNIST-like datasets.
- For generalization, Deep Learning (VGG16) is preferable.
- For robustness, Edge Detection showed resilience to noise, rotation, and blur.
- For speed, LBP and KNN are computationally efficient but at the cost of accuracy.