Signature Recognition Using CNN and Manual Feature Extraction Techniques

Syed Muhammad Murtaza Kazmi
Department of Computer Science
National University of Computer and Emerging Sciences
Islamabad, Pakistan
i210685@nu.edu.pk

***Abstract*—This paper presents a comparative study on signature recognition using Convolutional Neural Networks (CNN) and manual feature extraction techniques such as Histogram of Oriented Gradients (HOG) and Scale-Invariant Feature Transform (SIFT). The study involves preprocessing signature images, segmenting them into individual signatures, and classifying them using CNN and Artificial Neural Networks (ANN) trained on manually extracted features. The models are evaluated using precision, recall, F-measure, and accuracy metrics. The results indicate that the CNN model, while effective, is outperformed by the ANN model using HOG features in terms of accuracy. The study highlights the importance of feature selection and data preprocessing in signature recognition tasks.**

***Index Terms*—Signature Recognition, Convolutional Neural Network, HOG, SIFT, Image Processing, Machine Learning**

I. Introduction

Signature verification and recognition are critical components in various security and authentication systems, especially in financial institutions, legal documents, and identity verification processes. Handwritten signatures are widely accepted as a biometric attribute for identity verification due to their uniqueness and ease of acquisition [1]. Automating the process of signature recognition not only enhances security but also increases operational efficiency by reducing manual workload.

Traditional signature recognition systems rely on manual feature extraction methods, which require domain expertise and may not capture the intricacies of individual signatures. With the advent of deep learning, Convolutional Neural Networks (CNN) have shown remarkable performance in image recognition tasks by automatically learning hierarchical features from raw pixel data [2]. However, CNNs require large amounts of data for training, which can be a limitation in scenarios with limited datasets.

This study aims to develop a signature recognition system that processes images of signatures, segments them into individual signatures, and classifies them to identify the corresponding person. We compare the performance of a CNN-based approach with traditional manual feature extraction techniques, specifically Histogram of Oriented Gradients (HOG) and Scale-Invariant Feature Transform (SIFT), followed by training an Artificial Neural Network (ANN) for classification.

The source code for this project is available at our GitHub repository [7].

II. Methodology

A. Dataset Description

The dataset used in this study consists of scanned images containing handwritten signatures. Each image contains multiple signatures arranged in a tabular format. The dataset includes signatures from multiple individuals, with each individual having a varying number of signature samples.

B. Data Preprocessing

Data preprocessing is a crucial step to ensure the quality and consistency of the input data for model training. The following steps were undertaken:

1) Image Alignment: All images were checked for alignment to ensure that signatures are properly oriented. Misaligned images were corrected using rotation transformations based on their skew angle.

2) Noise Removal: Noise artifacts present in the images, such as specks or background patterns, were removed using median and Gaussian filters. This step helps in enhancing the quality of the signatures for better feature extraction.

3) Grayscale Conversion: The color images were converted to grayscale to simplify the computational complexity and focus on the intensity variations that are essential for signature recognition.

4) Thresholding: Grayscale images were converted to binary images using thresholding techniques. A global threshold value was applied to distinguish the signature (foreground) from the background.

5) Contour Detection: Contours in the binary images were detected using the OpenCV library. Contour detection helps in identifying the boundaries of each signature within the image.

6) Segmentation: Each detected contour corresponding to a signature was extracted and cropped from the original image. The segmented signatures were resized to a standard size of 128×128 pixels to ensure uniformity.

7) Class Organization: The segmented signature images were organized into folders corresponding to each individual. Each folder was named with a unique ID representing the person to whom the signatures belong.

C. Handling Class Imbalance

An initial analysis revealed that some classes (individuals) had insufficient samples, with some having only one sample.

Classes with fewer than two samples were removed to ensure that the models have enough data to learn meaningful patterns.

To address the remaining class imbalance, data augmentation was performed on classes with fewer than four samples. Augmentation techniques included rotation, shifting, zooming, and flipping to generate additional synthetic samples.

D. *Feature Extraction*

1) Convolutional Neural Network (CNN): The CNN model was designed to automatically learn features from the raw pixel data of the signature images. The architecture includes convolutional layers, pooling layers, and fully connected layers.

2) Histogram of Oriented Gradients (HOG): HOG is a feature descriptor used in computer vision and image processing for object detection. It captures edge and gradient structures that are characteristic of local shapes. The HOG features were extracted from the preprocessed images and used as input to an ANN classifier.

3) Scale-Invariant Feature Transform (SIFT): SIFT is another feature descriptor that detects and describes local features in images. It is invariant to scale and rotation, making it useful for matching keypoints between different images. SIFT features were extracted and used to train an ANN classifier.

E. *Model Architectures*

1) CNN Model: The CNN model architecture is as follows:

- **Input Layer:** Accepts images of size $128 \times 128 \times 1$.
- **Convolutional Layers:** Three convolutional layers with filter sizes of 32, 64, and 128, respectively. Each layer uses a kernel size of 3×3 and ReLU activation.
- **Pooling Layers:** Max pooling layers with a pool size of 2×2 follow each convolutional layer to reduce spatial dimensions.
- **Flatten Layer:** Converts the 2D feature maps into a 1D feature vector.
- **Dense Layers:** A fully connected layer with 128 neurons and ReLU activation, followed by a dropout layer with a rate of 0.5 to prevent overfitting.
- **Output Layer:** A dense layer with softmax activation corresponding to the number of classes.

2) ANN Model: The ANN model used for classification with manual features consists of:

- **Input Layer:** Size depends on the length of the feature vector (HOG or SIFT features).
- **Hidden Layers:** Three dense layers with 512, 256, and 128 neurons, respectively, each followed by a dropout layer with a rate of 0.5.
- **Activation Function:** ReLU activation is used for hidden layers.
- **Output Layer:** A dense layer with softmax activation for multi-class classification.

F. *Training and Hyperparameters*

1) Training Parameters: The models were trained with the following parameters:

- **Optimizer:** Adam optimizer was used for its adaptive learning rate capabilities.
- **Loss Function:** Categorical Crossentropy, suitable for multi-class classification.
- **Batch Size:** 32
- **Epochs:** CNN model was trained for 30 epochs; ANN models were trained for up to 50 epochs with early stopping.
- **Early Stopping:** Monitored validation loss with a patience of 5 epochs to prevent overfitting.

2) Data Splitting: The dataset was split into training and testing sets using stratified sampling to maintain the class distribution. A 50% split was used due to the limited number of samples per class.

G. *Evaluation Metrics*

The models were evaluated using the following metrics:

- **Accuracy:** The overall correctness of the model.
- **Precision:** The ability of the model to return only relevant instances.
- **Recall:** The ability of the model to identify all relevant instances.
- **F1-Score:** The harmonic mean of precision and recall.

III. Results

A. *Data Preprocessing Outcomes*

After filtering out classes with insufficient samples, 163 classes remained with a total of 647 images. Data augmentation increased the number of training samples to 652, balancing the class distribution.

B. *CNN Model Performance*

1) Training and Validation Accuracy: The CNN model achieved a training accuracy of 85.72% and a validation accuracy of 30.25% after 30 epochs, as shown in Fig. 1.

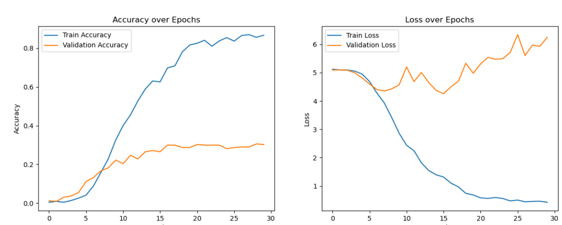

Fig. 1. CNN Training and Validation Accuracy and Loss

2) Evaluation Metrics: The CNN model's performance metrics are summarized in Table I.

TABLE I
CNN Model Performance Metrics

Metric	Precision	Recall	F1-Score
Average	28%	30%	27%

C. *ANN Model with HOG Features*

1) Training and Validation Accuracy: The ANN model trained on HOG features showed an improvement in validation accuracy over epochs, as depicted in Fig. 2.

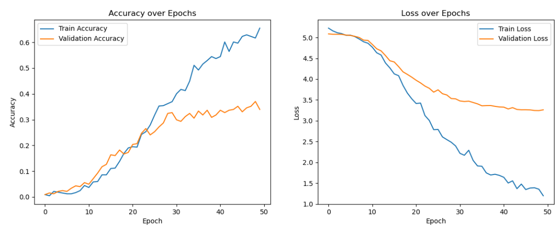

Fig. 2. HOG ANN Training and Validation Accuracy and Loss

2) Evaluation Metrics: The ANN model trained on HOG features achieved a test accuracy of 37.04%, outperforming the CNN model. The performance metrics are provided in Table II.

TABLE II
ANN MODEL WITH HOG FEATURES PERFORMANCE METRICS

Metric	Precision	Recall	F1-Score
Average	33%	37%	33%

D. *ANN Model with SIFT Features*

1) Training and Validation Accuracy: The training and validation accuracy for the ANN model with SIFT features remained low throughout the training process, as shown in Fig. 3.

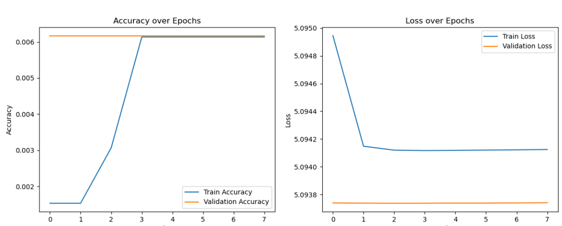

Fig. 3. SIFT ANN Training and Validation Accuracy and Loss

2) Evaluation Metrics: The ANN model trained on SIFT features performed poorly, with a test accuracy of only 0.62%. The model failed to learn meaningful patterns from the SIFT features for this dataset.

E. *Performance Summary*

Table III summarizes the performance of the three models. The ANN model with HOG features achieved the highest accuracy, while the SIFT-based model underperformed.

TABLE III
PERFORMANCE METRICS COMPARISON

Model	Accuracy	Precision	Recall	F1-Score
CNN	30.25%	28%	30%	27%
ANN (HOG)	**37.04%**	**33%**	**37%**	**33%**
ANN (SIFT)	0.62%	N/A	N/A	N/A

IV. DISCUSSION

A. *Effectiveness of CNN vs. Manual Features*

While CNNs are powerful in automatically extracting features, they require large amounts of data to generalize well. In this study, the limited dataset may have hindered the CNN's ability to learn robust features, leading to overfitting.

On the other hand, HOG features provided a more structured representation of the signatures, capturing edge and gradient information effectively. The ANN model trained on HOG features performed better, suggesting that manual feature extraction can be advantageous when dealing with small datasets.

B. *Challenges Faced*

1) Class Imbalance: The dataset had a significant class imbalance, with some classes having very few samples. Despite data augmentation, the limited variability in the signatures may have affected the models' ability to generalize.

2) Overfitting: Both the CNN and ANN models exhibited signs of overfitting, as indicated by the divergence between training and validation accuracy. Regularization techniques and more data could mitigate this issue.

3) Data Quality: Variations in handwriting styles, pressure, and pen strokes introduce a high degree of variability, making signature recognition challenging. Preprocessing steps aimed to reduce this variability but may not have fully addressed the issue.

C. *Improvement Strategies*

1) Collecting More Data: Acquiring more signature samples per individual would provide the models with more information to learn from and improve generalization.

2) Advanced Data Augmentation: Applying more sophisticated augmentation techniques, such as elastic distortions, could simulate the natural variations in handwriting.

3) Transfer Learning: Using pre-trained models and fine-tuning them on the signature dataset could leverage learned features from larger datasets.

4) Ensemble Methods: Combining the predictions of multiple models could improve overall performance by mitigating individual model weaknesses.

V. CONCLUSION

This study compared the performance of CNN-based feature extraction and manual feature extraction techniques (HOG and SIFT) for signature recognition. The ANN model trained on HOG features outperformed the CNN model, achieving an accuracy of 37.04% compared to the CNN's 30.25%. The

SIFT-based model did not perform well, indicating that SIFT features may not be suitable for signature recognition tasks.

The results highlight the importance of feature selection and data preprocessing in signature recognition. Manual feature extraction using HOG can be effective, especially when dealing with limited data. Future work could explore integrating CNNs with manual features, applying transfer learning, and collecting more data to enhance model performance.

VI. Prompts

- How can advanced data augmentation techniques improve model performance in signature recognition?
- Would integrating CNNs with manual feature extraction techniques yield better results?
- How does the quality and quantity of data affect the performance of deep learning models in biometric recognition tasks?

Code Availability

The source code and dataset used in this study are available at the GitHub repository: https://github.com/smmk47/signature-recognition.

VII. References

References

[1] E. N. Zois and V. Anastassopoulos, "Fusion of Correlation and Morphological Features for Signature Verification," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 14, no. 06, pp. 907–920, 2000.

[2] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, pp. 436–444, 2015.

[3] N. Dalal and B. Triggs, "Histograms of Oriented Gradients for Human Detection," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2005, pp. 886–893.

[4] D. G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[5] L. Perez and J. Wang, "The Effectiveness of Data Augmentation in Image Classification using Deep Learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 4, pp. 867–880, 2018.

[6] S. J. Pan and Q. Yang, "A Survey on Transfer Learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345–1359, 2010.

[7] A. Author, "Signature Recognition Source Code," GitHub repository, 2024. Available: //github.com/smmk47/signature-recognition.