IJFANS INTERNATIONAL JOURNAL OF FOD AND NUTRITIONAL SCIENCES
ISSN PRINT 2319 1775 Online 2320 7876
Research Paper Journal Volume 13, Iss 04, 2024

SIGNATURE FORGERY DETECTION USING DEEP LEARNING

G.Shiva prasad[1], Ch.Navya[2], G. Sahithi[3], S. Shivamani[4], K. Rakesh[5], K.Shobhanbabu[6]

[2,3,4,5]BTech Student, Department of CSE, Balaji Institute of Technology and Science, Laknepally, Warangal, India

[1,6] Assistant Professor, Department of CSE, Balaji Institute of Technology & Science, Laknepally, Warangal, India

ABSTRACT

Signature forgery detection is a critical aspectof security, providing a reliable method for verifying authenticity. This process analyses features such as stroke patterns, length, continuity, and thickness. Variations in these characteristics, even for signatures from the same individual, make forgery detection a challenging task. The system processes signature images by extracting key features using convolutional neural networks (CNNs). Preprocessing steps such as grayscale conversion and binarization are applied to enhance the clarity of features. The CNN model analyses the input signature and compares it against predefined criteria to determine its authenticity. Effective feature extraction and model training are vital to ensure accurate and reliable results. This method simplifies the verification process while maintaining precision and efficiency.

1. INTRODUCTION

Handwritten signatures are a widely trusted form of biometric authentication, especially in finance, where accurate identity verification is essential. However, natural variations in signatures caused by factors like mood, health, and environmental conditions make it difficult to differentiate genuine signatures from forgeries.

Forgeries can be particularly deceptive when created by skilled individuals. While such forgeries may closely resemble the original, they often lack the natural fluidity and unique stroke patterns. These limitations make traditional manual verification methods prone to errors and inefficiencies.

This project aims to develop an automated offline signature verification system that uses machine learning techniques to address these challenges. The system analyzes geometric and textural features of signatures to assess their authenticity. Neural networks are trained on preprocessed datasets containing both genuine and forged signatures, leveraging Python libraries such as NumPy, Matplotlib, and scikit-learn.

The primary goal is to enhance security by automating the verification process, thereby reducing the likelihood of accepting forged signatures as genuine. By emphasizing precision and efficiency, this project seeks to improve the reliability of signature-based authentication systems, particularly in critical areas like financial transactions.

IJFANS
International Journal of Food And Nutritional Sciences
Official Publication of International Association of Food and Nutrition Scientists

CLASSIFICATION

Deep Learning

Deep learning is an advanced technique in machine learning that employs artificial neural networks with multiple layers to process complex data. These networks are inspired by the human brain's architecture, allowing the system to improve its learning and predictive abilities as more data is fed into it. While simpler tasks can be accomplished with single-layer networks, multiple layers enable the detection and analysis of intricate patterns, making deep learning particularly effective for tackling more complicated challenges.

Convolutional Neural Networks (CNNs)

Convolutional Neural Networks (CNNs) are a particular type of deep learning model, specialized for processing image data. CNNs use convolutional layers to detect and extract critical features—such as edges, shapes, and textures—from images. These features are passed through several layers of the network for further processing, which allows the system to accurately classify and interpret images. CNNs are highly effective in a wide range of image-related applications, including object detection, facial recognition, handwriting analysis, and offline signature verification, which is a central focus of this project.

2.LITERATURE SURVEY

1. JiveshPoddara et al. [1]:

Proposes a signature classification method combining Crest Trough and CNN techniques. After classification, forgery detection is performed using the Harris Algorithm followed by the Surf Algorithm. The dataset consists of 1320 images, including 25 unique pictures in non-grayscale format.

2. Vincent Christlein et al. [2]:

Explores the detection of copy-move forgeries, where sections of an image are duplicated and altered. The study evaluates algorithms and feature sets for identifying manipulations at various processing stages and levels.

3. Ms. Manjula Subramaniam etal.[3]:

Focuses on a CNN-based system to detect forged signatures in offline scenarios, such as documents, cheques, and passports. The research tackles signature variability over time, enhancing accuracy for offline signature verification using deep learning.

4. Sandesh Kandel et al. [4]:

Presents a CNN-based system for verifying handwritten signatures, accounting for natural variations. It aims to improve performance over traditional methods, with applications in areas like loan processing and legal documentation.

5. Eman Alajrami et al. [5]:

Highlights an offline signature verification model using CNNs that achieves high accuracy and speed. The paper compares this method to online biometric systems, emphasizing the advantages of automated verification for large-scale document processing.

6. Navya V K et al. [6]:

Introduces a comprehensive signature verification system integrating image processing, OCR, and machine learning. The study addresses challenges in detecting forged signatures and enhances the accuracy and security of verification systems.

7. JiaXin Ren et al. [7]:

Describes a method where local structural patterns are used to extract features from processed images. Verification is performed by matching the extracted features against stored data.

8. Kshitij Swapnil Jain et al. [8]:

Proposes a CNN-based approach for signature verification across platforms, addressing both offline and online methods. The paper emphasizes the importance of signature verification in mitigating identity fraud and ensuring security in financial and legal sectors.

3.PROBLEM STATEMENT

The need for efficient and trustworthy methods to authenticate handwritten signatures is critical in sectors such as finance, law, and business. Traditional methods of signature verification, which rely on manual comparison, have limitations, especially with the increasing sophistication of modern forgery techniques.

This project utilizes a Convolutional Neural Network (CNN)-based machine learning model to classify signatures as either genuine or forged, significantly enhancing accuracy and reducing the risks of human error. The system includes a web-based interface developed using Flask, HTML, and CSS, which allows users to easily upload and compare signature images. By combining advanced deep learning with a user-friendly interface, this project aims to provide a more reliable and accessible solution for signature verification.

4.EXISTING SYSTEM

Signature verification systems that use K-means clustering rely on unsupervised learning to group signatures based on their similarities. In this system, features of the signatures are clustered into predefined categories, such as genuine or forged, based on how close they are to the cluster centroids. However, K-means struggles with accurately capturing the complex and subtle variations in handwritten signatures. The system depends on manually selected features, such as pixel intensities or geometric shapes, which may not be sufficient for distinguishing between real and forged signatures.

K-means also faces challenges when dealing with large datasets or more complex signatures, as it does not scale well with high-dimensional data. Additionally, it does not consider the spatial relationships within the signature, which limits its ability to detect more sophisticated forgery techniques. As a result, systems relying on K-means can have a higher rate of false positives and false negatives, making them less reliable for critical applications like financial transactions.

5.PROPOSED SYSTEM

The proposed system improves upon the existing method by using Convolutional Neural Networks (CNNs), which are specifically designed to analyze and learn complex patterns in images. CNNs automatically learn relevant features from the data, which makes them more effective at identifying even subtle differences between genuine and forged signatures. This approach eliminates the need for manual feature extraction and allows the system to adapt and improve as it processes more data. CNNs also scale better with larger datasets, providing improved accuracy and efficiency.

By passing signature images through convolutional layers, pooling layers, and fully connected layers, CNNs can detect small discrepancies between genuine and forged signatures, making the verification process more accurate and reliable compared to traditional methods like K-means clustering.

6.METHODLOGY

Data Preparation

The initial phase involves preparing a dataset of 2,149 signature images obtained from Kaggle. This dataset contains both authentic and forged signatures from 69 distinct individuals. The preprocessing steps for the data are as follows:

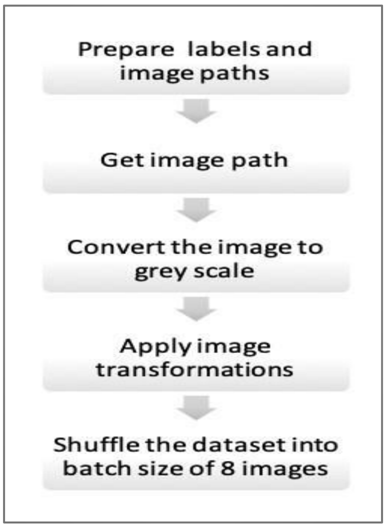

Fig:1 Data Preparation Flow Diagram

1. Labeling and Image Path Assignment: Each image is labeled as either genuine or forged, with corresponding paths assigned to link the dataset to the signature types.

2. Grayscale Conversion: To reduce processing demands, the images are converted to grayscale, highlighting the key features needed for analysis.

3. Resizing: The images are resized to a uniform size of 100x100 pixels, ensuring consistency in the input fed into the neural network.

4. Shuffling: The images are randomly shuffled into small batches to eliminate bias during the training process.

7.FEATURE EXTRACTION

Feature extraction is essential for transforming the raw signature images into usable representations for the classification process. The Convolutional Neural Network (CNN) utilizes convolution layers to identify fundamental patterns like edges and textures through specialized filters. Max pooling layers help reduce the size of the feature maps, retaining critical information. The resulting feature maps are flattened into a one-dimensional vector, which is then passed through fully connected layers to determine whether the signature is genuine or forged. ReLU activation functions are used to add non-linearity, enabling the model to learn complex patterns in the data.

MODEL DEVELOPMENT

The CNN used for this signature verification project comprises multiple convolution layers followed by pooling layers for feature extraction. Fully connected layers are then employed for the classification task. The model is trained on the prepared data and optimized using a loss function and optimization algorithm to minimize classification errors.

Model Evaluation

Following the training phase, the model's performance is evaluated using a separate test set. Key metrics such as accuracy and loss are analyzed to determine how effectively the model distinguishes between real and forged signatures. Visualization tools are utilized to monitor the model’s progress during training.

System Deployment

The trained CNN model is integrated into a web-based application developed with the Flask framework. This application allows users to upload signature images for real-time verification. The Flask backend handles the image processing and classification tasks, providing users with results that indicate whether the uploaded signature is genuine or forged.

8.PROJECTREQUIREMENTS

Software Requirements

1. Python: The primary programming language used in the project.

2. Visual Studio Code (VS Code): The IDE used for writing and managing the project code.

3. Libraries and Frameworks

- PyTorch: Used for implementing and training the CNN model.

- Flask: A framework for building the web application interface.

- Pillow: A Python Imaging Library for image operations such as resizing, converting, and opening images.
- NumPy: Utilized for numerical operations, particularly with image data handling.
- Matplotlib: A plotting library to visualize model training progress, including accuracy and loss curves.

Dataset

The dataset includes signature images, organized appropriately for both training and testing purposes.

Development Environment Setup

1. VS Code Configuration: Set up VS Code with necessary Python extensions for coding and debugging.
2. Virtual Environment: Establish a Python virtual environment to manage project dependencies and avoid conflicts.
3. Library Installation: Install the necessary libraries through pip or using a requirements.txt file.

9.SYSTEM DESIGN

The Signature Verification System utilizes Convolutional Neural Networks (CNNs) to verify handwritten signatures. It consists of a user interface built with Flask, a backend to handle image processing, and a machine learning model for classification.

Components:

1. Frontend: Built using Flask to provide an interface for users to upload signature images and view the results.
2. Backend: Python-based backend processes uploaded images, performs preprocessing, and communicates with the CNN for signature classification.
3. Machine Learning Model: The CNN model is trained to distinguish between authentic and forged signatures using PyTorch.
4. Dataset: Includes both genuine and forged signature images to train and test the model.

Development Process

a. Frontend Development:

- User Interface: Created using HTML and CSS, designed for an intuitive experience, enabling users to upload images for verification.
- File Upload: Signature images are uploaded through HTML forms.
- Result Display: Flask handles form submissions and displays the signature verification results.

b. Backend Development:

1. Flask Application: The Flask app manages HTTP requests, file uploads, image preprocessing, and invokes the CNN model for inference.

2. Image Preprocessing: Images are converted to grayscale, resized to 100x100 pixels, and normalized for input into the CNN model.

3. Model Integration: The pre-trained CNN, implemented in PyTorch, processes the uploaded signature images to classify them as genuine or forged.

c. Machine Learning Model:

1. Network Architecture

Convolutional Layers: These layers extract significant features from the images through convolution operations.

Activation Functions: ReLU activations introduce non-linearity, enabling the model to learn complex patterns.

Pooling Layers: Reduce dimensionality while preserving essential features.

Fully Connected Layers: Perform classification based on features extracted by the convolutional layers.

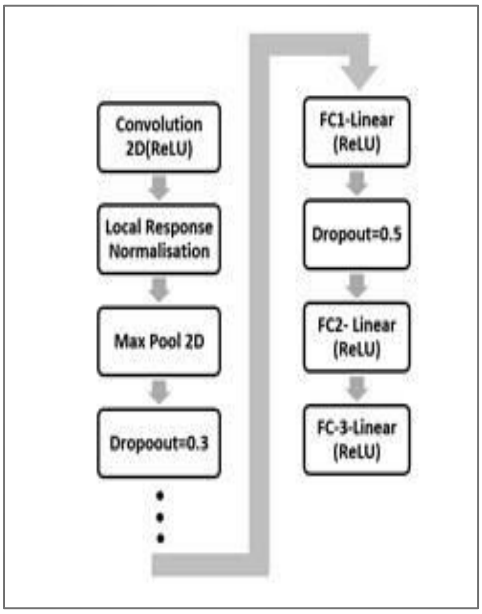

Fig:2 Network Architecture

2. Training

1. The dataset is split into training and testing sets, with data augmentation applied to increase the model's robustness. Cross-entropy loss and the Adam optimizer are used to minimize errors in classification.

d. Testing and Deployment:

1. Testing

The model is tested on a separate dataset to ensure it can accurately classify genuine and forged signatures.

2. Deployment

The Flask application is deployed on a web server or cloud platform for users to verify signatures in real-time.

User Interaction:

Users upload two images: one genuine and the other suspected of being a forgery. The system processes both images using the trained CNN, compares their features, and returns a result indicating whether the signatures match (genuine) or differ (forged).

Verification Process:

Upon uploading the signature images, the system compares their features. Based on the analysis, it provides a result, indicating whether the signatures are authentic or if a forgery is detected.

10.Future Scope for Signature Verification System

The Signature Verification System has significant potential for further development. Advancing its performance through the implementation of more complex deep learning models and expanding the dataset to encompass a variety of signature styles will enhance both accuracy and adaptability. This will allow the system to generalize more effectively to diverse real-world scenarios. Additionally, optimizing the system to operate in real-time is essential for faster verification, particularly in high-demand applications.

The inclusion of other biometric modalities, such as fingerprint or facial recognition, could provide enhanced security and strengthen the overall verification process. A more intuitive user interface would also improve accessibility and the user experience.

Exploring innovative machine learning techniques for feature extraction could further improve the system's efficiency and reliability. Moreover, expanding the system to manage larger datasets and ensuring adherence to data protection regulations will be vital as the system scales, particularly in sectors like banking and legal industries, where security is paramount.

These improvements will help the Signature Verification System stay ahead of new forgery methods and meet the increasing demand for secure signature verification in various industries.

11.CONCLUSION

This project successfully developed a Signature Forgery Detection System based on Convolutional Neural Networks (CNNs). By leveraging CNN's capability to process and analyze image data, the system effectively distinguishes between genuine and forged signatures. The model's architecture, which includes multiple convolutional and fully connected layers, along with preprocessing steps like resizing and grayscale conversion, ensures accurate and reliable signature verification.

Overall, the project demonstrates the potential of CNNs to enhance security by automating handwritten signature verification. Future work may include expanding the dataset, fine-tuning the model for better accuracy, and adding real-time verification features to further improve performance and extend the system's applicability.

REFERENCES

1. Jivesh Poddara, Vinanti Parikha, Santosh Kumar Bhar. "Offline Signature Recognition and Forgery Detection Using Deep Learning." *International Journal of Computer Sciences and Engineering*.

2. Christlein, Vincent, Riess, Chrisan, Jordan, Johannes, Riess, Corinna, and Angelopoulou, Elli. "An Assessment of Well-Known Copy-Move Forgery Detection Methods." IEEE.

3. Subramaniam, Manjula. "Peer-Reviewed, Fully Refereed, Open-Access International Research Journal of Modernization in Engineering Technology and Science."

4. Kandel, Sandesh. "Machine Learning-Based Signature Forgery Detection in Document Authentication Systems."

5. Alajrami, Belal A. M., Ashqar, Bassem S., Abu-Nasser, Ahmed J., Khalil, Ahmed J. "Signature Verification."

6. V K, Navya, Sarkar, Abhilasha, Viswanath, Adi, Koul, Akshita, Srivastava, Amipra. "Signature Forgery Detection System."

7. Wen, Jing, Chen, MoHan, Ren, JiaXin. "Off-Line Signature Verification Based on Local Structural Pattern Distribution Features." *Pattern Recognition*, 2014, Volume 484, Springer.

8. Jain, Kshitij Swapnil, Patel, Udit Amit, Kheni, Rushab. "Handwritten Signatures Forgery Detection."

BIBLIOGRAPHY

I am Navya Chanda from the Department of Computer Science and Engineering. Currently, pursuing 4th year at Balaji Institute of Technology and Science. My research is done based on "Signature Forgery Detection Using Deep Learning".

I am Sahithi Gudimalla from the Department of Computer Science and Engineering. Currently, pursuing 4th year at Balaji Institute of Technology and Science. My research is done based on "Signature Forgery Detection Using Deep Learning".

I am Shivamani Sirimalle from the Department of Computer Science and Engineering. Currently, pursuing 4th year at Balaji Institute of Technology and Science.My research is done based on "Signature Forgery Detection Using Deep Learning".

I am Rakesh Kannam from the Department of Computer Science and Engineering.Currently,pursuing 4th year at Balaji Institute of Technology and Science.My research is done based on "Signature Forgery Detection Using DeepLearning".