Journal of Computing and Data Technology
Vol. 01, No. 01, June 2025, pp. 50-58.
DOI: https://doi.org/10.71426/jcdt.v1.i1.pp50-58

Homepage: https://review.journal-of-modern-technology.com/index.php/jmt/index

Enhancing Script Identification in Dravidian Languages using Ensemble of Deep and Texture Features

Satishkumar Mallappa[1*], Chandrashekar Gudada[2], P. Megana Santhoshi[3], Raghavendra[4]

[1*]Department of Mathematical and Computational Sciences, Sri Sathya Sai University for Human Excellence Kalaburagi, Karnataka, India, Email: satishkumar679@gmail.com , satish.k@sssuhe.ac.in , ORCID: https://orcid.org/0000-0001-9419-7606

[2]Department of Mathematical and Computational Sciences, Sri Sathya Sai University for Human Excellence Kalaburagi, Karnataka, India, Email: chandrashekar.vg@sssuhe.ac.in , chandrugudada@gmail.com , ORCID: https://orcid.org/0000-0003-2469-8274

[3]Department of Computer Science Engineering, KLM College of Engineering for Women, Kadapa-Pulivendula Road, YSR Kadapa Dt, Andhra Pradesh, 516003, India, Email: pmsanthoshi@gmail.com , ORCID: https://orcid.org/0000-0002-1376-7601

[4]College of Horticulture (UHSB), Munirabad, Koppal, Karnataka, India, Email: raghavendra.yadav@uhsbagalkot.edu.in , yadav156@gmail.com

Article Info

Article history:

Received: May 20, 2025
Revised: June 22, 2025
Accepted: June 26, 2025
First Online: June 30, 2025

Keywords:

Script languages
Data processing
Machine Learning
Convolutional Neural Network (CNN)
Histogram of Oriented Gradients (HOG)
Support vector machine (SVM)
Local binary patterns (LBP)

ABSTRACT

Dravidian languages, including Tamil, Telugu, Kannada, and Malayalam, have complex orthographic structures, making script identification challenging particularly for camera-based document images. This study proposes a hybrid approach that combines deep learning and texture-based methods for robust script recognition. The GoogLeNet convolutional neural network (CNN) model is used to extract deep features, while local binary patterns (LBP) and histogram of oriented gradients (HOG) capture texture characteristics. These features are fused and classified using support vector machine (SVM) classifier. Results show that CNN features alone achieve 84.50% accuracy, LBP achieves 85.90%, and HOG achieves 76.10%, while their fusion significantly improves accuracy to 92.10%. The combination of CNN and HOG features reaches 95.00% accuracy, demonstrating the effectiveness of integrating deep learning with texture-based approaches. This method has applications in OCR systems and assistive technologies for the visually impaired.

Corresponding Author:

Email address of corresponding author: satish.k@sssuhe.ac.in (Satishkumar Mallappa)

1. INTRODUCTION

The process of identifying text within a document is essential for handling digital documents, especially in OCR and content analysis. Noise, lighting variations, and complex orthographic morphology make script recognition from image-based documents challenging. Dravidian languages, primarily spoken in southern India, further complicate the task due to their intricate script designs. Most previous studies have focused on monolingual or bilingual script identification. Despite the strong feature extraction capabilities of CNNs, they tend to lose fine textural details. To address this, texture-based techniques such as LBP and HOG are employed [1] -[2]. Combining deep learning with texture analysis improves automation accuracy by capturing both global structures and local textural details. However, most existing techniques rely on controlled datasets and simple feature fusion methods, which are often impractical. Real-world variability is something many models fail to address. A more practical approach would be to merge deep features with texture features in a flexible way, making models more robust to different environmental conditions. Greater emphasis should be placed on advanced feature fusion techniques and testing on more diverse datasets to enhance generalization [3].

The research proposes an ensemble method aimed at improving script identification in South Indian languages—Tamil, Telugu, Kannada, and Malayalam—by combining deep learning and texture-based techniques. The approach integrates CNN features from the GoogLeNet model with texture descriptors such as LBP HOG. This represents a hybridization of two feature extraction techniques known to capture different aspects of a pattern: deep learning captures hierarchical structures, while texture analysis captures refined local details. Together, these techniques facilitate a scalable and precise recognition system [4]-[5].

The accurate recognition of Dravidian scripts faces added challenges due to the diversity of writing styles, distortions from varying fonts, poor lighting, and background noise—factors often encountered in real-world South Indian environments [6]. Traditional algorithms must employ robust machine learning techniques, as many fail to address these issues effectively. CNNs have shown remarkable performance in script recognition due to their ability to capture meaningful spatial features. The complex composition of strokes, curves, and character positions makes recognition difficult yet CNNs can manage this through their deep layered structures [7]. Although deep CNNs extract robust features, certain textural variations essential for distinguishing Dravidian scripts often remain unaddressed. Texture-based methods focus on patterns, edges, shapes, and fine-grained details that enhance character dissimilarity. Techniques such as LBP and Gabor filters, when applied alongside CNNs, improve the model's ability to capture both local and global variations, thereby boosting script recognition accuracy [8].

Blending deep CNN features with texture-based approaches yields better results than using either method alone. While CNNs capture high-level structural patterns, texture analysis detects finer details that enable the model to handle variations in stroke thickness, curvature, and orientation [4]. This fusion enhances recognition accuracy and boosts system performance in real-world scenarios where images are affected by environmental noise.

Furthermore, the combination of these techniques helps the model recognize different variants of Dravidian scripts, thereby reducing dependence on large, labelled datasets, which are tedious and expensive to create. This contributes to efficient and scalable script recognition systems that can be applied in automated document processing, digital archiving, and assistive technology for the visually impaired [5][6].

This motivated the present experiment, which aims to identify scripts originating from the southern part of India. Preliminary work has been conducted on bilingual script identification, which has been addressed in numerous studies. In the study presented in [7], the authors described a bilingual script identification method using images captured by a camera. They fused two texture features LBP with HOG and GLCM and fed them into KNN and SVM classifiers. Among all classifiers, SVM achieved the highest recognition accuracy of 95.59%.

Script identification in Indian document images has also been addressed using MobileNetV3, a lightweight and efficient CPU-based convolutional neural network. It processes scripts from six Indian languages—Bangla, Gurumukhi, Hindi, Kannada, Malayalam, Tamil—as well as English, with a reported accuracy of 98%, using center loss and cross-entropy loss [8].

In [9], the authors proposed WAFFNet, an attention-based feature fusion architecture for word-level multilingual scene text script detection. A recent advancement, SANet-SI, is a self-attention network for script identification from natural scene text images. It uses multi-scale feature decomposition and a style-based recalibration module. To improve efficiency and reduce model size, this method integrates both local and global features while replacing fully connected layers with a global average pooling layer. The cross-dataset validation results were exceptional, using RRC-MLT2017, SIW-13, and CVSI2015 as benchmarks [10], [16]-[17].

For script recognition from video text and handwritten text, the authors in [11] proposed the use of handcrafted texture features such as SRS-LBP and MLP. Their approach achieved recognition accuracies of 98.1% with SRS-LBP and 92.78% with MLP. In [12], the task of script recognition from camera-captured document images was addressed using statistical feature extraction methods along with template matching and contour signature techniques. Classification was performed based on the Hamming distance between validation and training data, achieving a recognition rate of 91.00%. An attention-based method using a convolutional LSTM network was described in [13] for script recognition in still images and video frames from natural scenes. After extracting local and global CNN features from the input images, a fusion method was applied. Using the CNN-LSTM network, the authors achieved recognition accuracies of 90.23% and 96.70% for the ICDAR-17 and MLe2e datasets, respectively. In [14], to facilitate script recognition from multi-script scene text captured by cameras, HOG, GLCM textures, and shape-based features were proposed. A new feature vector was constructed by concatenating HOG and GLCM features extracted from word images. This concatenated vector was classified using five well-known classifiers—SVM, multi-class SVM, Naïve Bayes, MLP, and another multi-class classifier [24]-[25]. Among them, the MLP classifier achieved the highest recognition accuracy of 90.00%. In [15], the recognition of Kannada and Malayalam scripts reached an accuracy of 97.05%. The authors performed bus signboard script recognition from input images by extracting Gabor, Wavelet, and Log-Gabor features.

Upon reviewing the literature [21]-[23], it is evident that relatively few studies focus on script recognition from camera-captured document images. Although some articles incorporate CNN features and custom-designed texture features for such tasks, none specifically address South Indian scripts. This gap in the literature has motivated the current research.

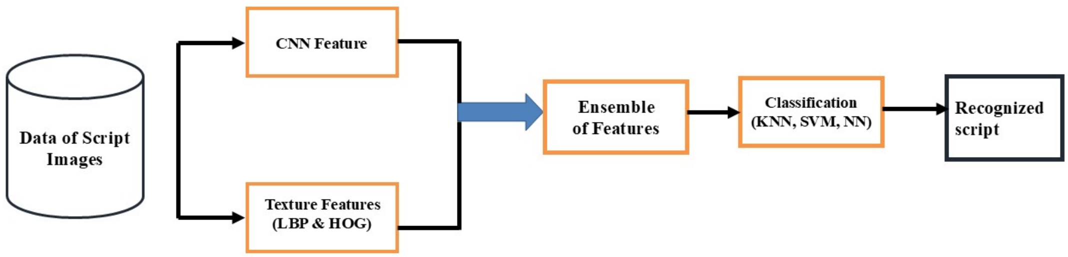
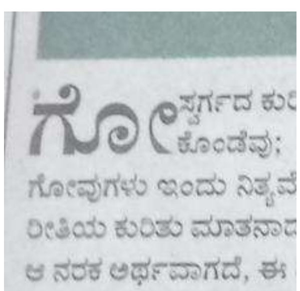
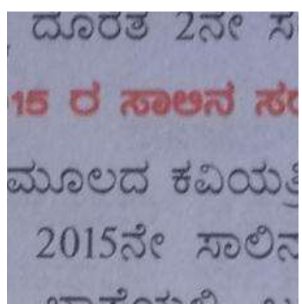
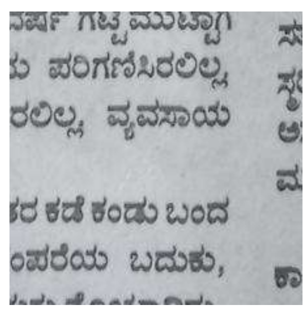
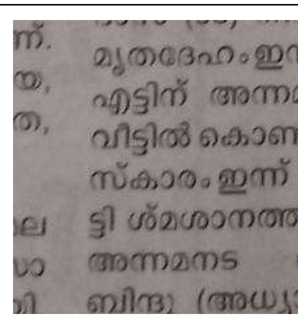


Figure 1: Block diagram of proposed system.

2. DATASET DETAILS

Because there is no other literature dataset that is appropriate for this work, a custom dataset is constructed based on South Indian scripts Kannada, Telugu, Tamil, and Malayalam. Each script is composed of 3000 images and all images were resized to 224x224 covering a total of 12000 images. These images were taken in controlled settings using a mobile phone with a 16 MP camera. The sample images of the dataset are shown in Figure 1. The dataset contains the images from various kinds of documents like newspapers, fiction and non-fiction novels, news magazines, printed documents and so on by using the Camera with 4920x3264 megapixel resolution.

Language	Samples		
Kannada			
Malayalam			

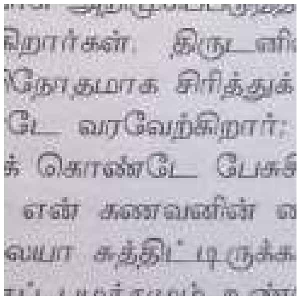
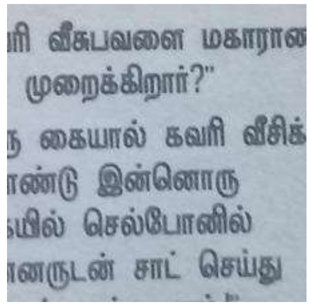
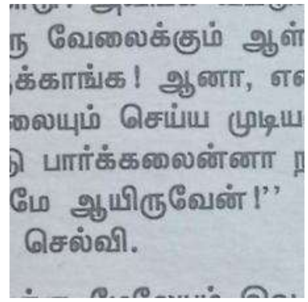
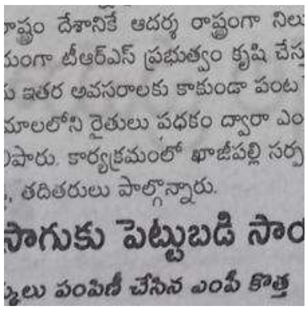
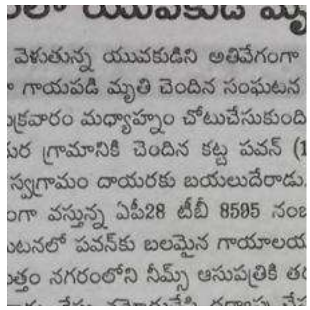
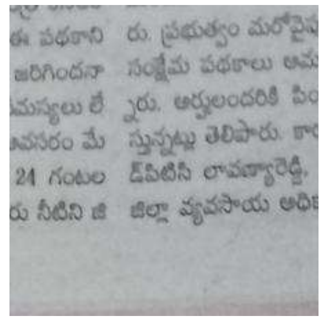
Tamil			
Telugu			

Figure 2: Sample images of different languages.

3. PROPOSED METHOD

This proposed approach is organized into three stages: initially, features are extracted using the pre-trained GoogLeNet CNN model; next, texture features are extracted using LB and HOG; finally, the CNN and texture features are merged.

3. 1 Pre-trained GoogLeNet Model:

GoogLeNet, also known as Inception v1, is a deep CNN that revolutionized computer vision due to its highly efficient architecture. It achieved outstanding accuracy and won the ILSVRC 2014 (ImageNet Large Scale Visual Recognition Challenge), while using significantly fewer parameters than traditional CNNs such as VGGNet.

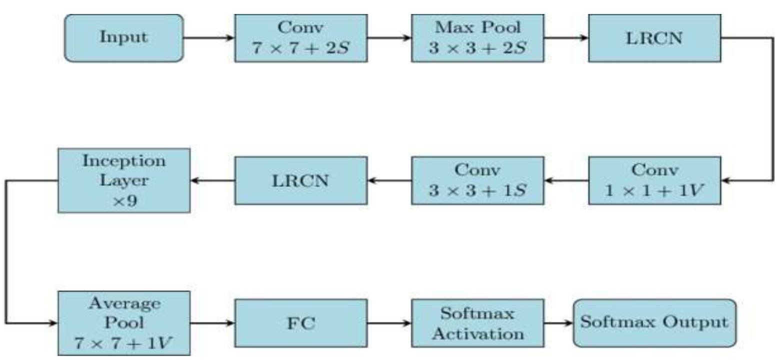


Figure 3: A simplified block diagram of the GoogLeNet Architecture [19].

3. 2 Deep Feature Extraction from GoogLeNet Model

The CNNs like GoogLeNet are designed to learn the hierarchical properties of input images automatically. The features extracted from a CNN represent varying levels of abstraction, from simple patterns in the shallow layers to complex semantic representations in the deeper layers. The GoogLeNet captures this hierarchy effectively. Early layers detect basic features such as edges and gradients, while intermediate layers combine these into object parts and textures. Deeper layers—such as pool5-drop_7x7_s1—encode high-level semantic information. This layer employs global average pooling to produce a 1024-dimensional feature vector that concisely summarizes the image's content. These features are compact, spatially invariant, and well-suited for tasks like classification, clustering, and transfer learning. As part of preprocessing, input images are resized to 224×224 pixels to ensure consistency across the dataset. The extracted features generalize well across various image types and are robust to spatial distortions, making GoogLeNet an efficient and reliable model for representing image semantics in downstream applications.

The primary objective of feature extraction from GoogLeNet is to encode compact, high-level semantic representations while maintaining robustness to positional variations in the input. These representations are then input into classifiers such as K-Nearest Neighbours (KNN), Support Vector Machines (SVM), or Neural Networks for further recognition and decision-making tasks.

3. 3 Texture Feature Extraction

Texture features capture the patterns, structures, and spatial arrangements within an image, providing crucial information about its surface properties. These features enable object identification based solely on texture, independent of colour or brightness, and are widely used in image processing, computer vision, and pattern recognition. In this work, texture feature extraction is considered due to its ability to represent and differentiate surface characteristics effectively. Accordingly, the LBP and HOG methods are employed, as both are well-established and widely used techniques in this domain.

3. 4 Local Binary Pattern

Images can have their features extracted using the LBP operator—a simple yet effective method for texture analysis. LBP works by thresholding the neighbourhood of each pixel relative to its center value and encoding the result as a binary number [20]. The LBP is represented by (1).

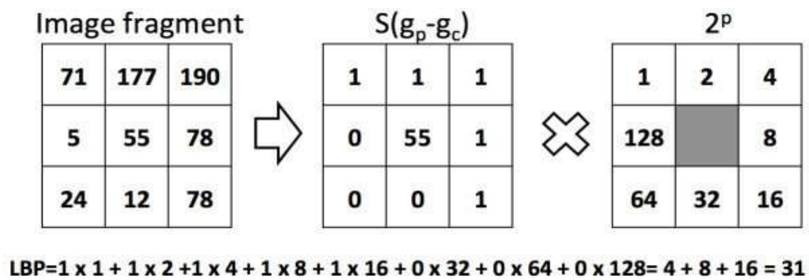


Figure 4: Illustration of the LBP process applied to a grayscale pixel with parameter P = 8 and R = 1 [21].

$$LBP_{P.R} = \sum_{p=0}^{P-1} S(g_p - g_c)2^p \tag{1}$$

In (1), PPP: Number of sampling points in the circular neighbourhood, RRR: Radius of the circular neighbourhood, gc: Intensity of the central pixel., gp: Intensity of the pth neighbouring pixel, s(x): Step function that thresholds the difference between the neighbouring and central pixel values and which is represented by (2).

$$s(x) = \begin{cases} 1, if\ x \geq 0; \\ 0, otherwise \end{cases} \tag{2}$$

3. 5 Histogram of Oriented Gradients

The HOG is a widely used feature descriptor in computer vision and image processing, particularly for object detection tasks such as pedestrian identification. It operates by analyzing the distribution of edge directions or intensity gradients within specific regions of an image.

The gradient of an image represents the change in intensity at each pixel, highlighting edges and transitions that define shapes and structures. Each gradient has both a magnitude, indicating edge strength, and an orientation, representing the edge direction. To extract HOG features, the image is divided into small cells (e.g., 8×8 pixels), within which a histogram of gradient orientations is computed. Each histogram bin corresponds to a specific orientation range (e.g., 0°–20°, 20°–40°), with gradient magnitudes contributing to their respective bins.

To enhance robustness, adjacent cells are grouped into larger blocks (e.g., 2×2 cells), and their histograms are concatenated and normalized. This normalization reduces sensitivity to lighting and contrast variations, making HOG an effective technique for capturing object shape and appearance.

Using this method, 81 texture features were extracted. HOG features effectively capture the local shape and appearance of an image, making them especially useful for object detection tasks—particularly when paired with linear classifiers like SVMs. As a key component in computer vision, HOG focuses on edge- and gradient-based features, providing a strong structural representation of an image.

3. 6 Ensemble of Features

An ensemble of features enhances the generalization, accuracy, and robustness of computer vision and machine learning models by integrating multiple feature descriptors. This strategy captures complementary information that individual descriptors may overlook, thereby improving overall performance.

In this study, deep and texture features are combined to exploit their respective strengths. Deep features are extracted using the GoogLeNet model, while texture features are derived from the HOG and LBP methods. The resulting feature sets include 1,000 deep features, 59 LBP features, and 81 HOG features. These are subsequently fed into classification models such as SVM, KNN, and Neural Networks, each yielding varying levels of recognition accuracy.

To maximize recognition performance, multiple combinations of features are explored. These include HOG with LBP, HOG with deep features, LBP with deep features, and a comprehensive combination of deep features with both HOG and LBP. The fusion process is carried out through straightforward feature concatenation, resulting in a unified and enriched feature descriptor.

Table 1. Details of the features with combinations.

Sl. No.	Features description	No. of Features
1	LBP	59
2	HOG	81
3	GoogLenet (Deep Features)	1000
4	LBP+HOG	140
5	LBP+Deep Features	1059
6	HOG+Deep Features	1081
7	LBP+HOG+Deep Features	1140

4. RESULTS AND DISCUSSIONS

An intensive experiment was conducted on a custom dataset to evaluate the effectiveness of the proposed method. This dataset includes four South Indian languages—Kannada, Telugu, Tamil, and Malayalam—with each language comprising 3,000 camera-captured block images of size 224×224 pixels. From these images, two types of features were extracted: texture features and deep features.

Table 2. Script identification using LBP and HOG features.

<table>
<tr><td colspan="6">Dravidian Language Script Identification using LBP and HOG features</td></tr>
<tr><td colspan="3">LBP (59)</td><td colspan="3">HOG (81)</td></tr>
<tr><td>KNN</td><td>SVM</td><td>NN</td><td>KNN</td><td>SVM</td><td>NN</td></tr>
<tr><td>75.5%</td><td>85.4%</td><td>81.5%</td><td>71.1%</td><td>76.1%</td><td>71.9%</td></tr>
<tr><td colspan="6">Combined LBP and HOG features (140)</td></tr>
<tr><td colspan="2">KNN</td><td colspan="2">SVM</td><td colspan="2">NN</td></tr>
<tr><td colspan="2">81.0%</td><td colspan="2">88.7%</td><td colspan="2">84.4%</td></tr>
</table>

To capture the image details, two widely used texture analysis methods were employed. The LBP were used to extract local texture information (micro-patterns), while HOG captured global shape information (edges and gradients). In addition, deep features were extracted using the pretrained GoogLeNet deep learning model. After feature extraction, the next step involved classification using three popular machine learning classifiers: K-Nearest Neighbours (KNN), SVM, and Neural Network (NN). These classifiers were implemented in MATLAB to evaluate the recognition performance of the extracted features. The table 2 represents the recognition accuracies achieved by each classifier.

Table 3. Script identification results from texture and deep features.

GoogLeNet Deep Features (1000)		
KNN	SVM	NN
73.5%	84.5%	81.4%
Combined Deep and LBP Features (1059)		
KNN	SVM	NN
76.2%	84.5%	81.4%
Combined Deep and HOG Features (1081)		
KNN	SVM	NN
90.4%	95.0%	94.2%
Combined Deep and Texture Features (1140)		
KNN	SVM	NN
80.1%	92.1%	89.1%

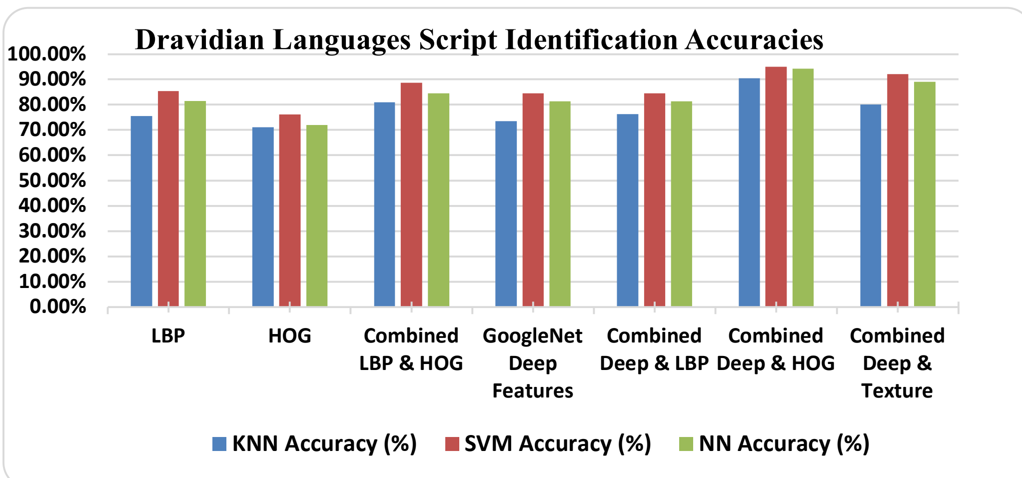


Figure 5. Graphical representation of accuracy of Dravidian languages scripts identification.

The results presented in the tables and graphs illustrate the effectiveness of the extracted features and classifiers in identifying South Indian scripts. Among the classifiers, the Support Vector Machine (SVM) consistently demonstrated superior performance compared to K-Nearest Neighbours (KNN) and Neural Networks (NN) when using texture features alone. Specifically, SVM achieved an accuracy of 85.4% with LBP features and 76.1% with HOG features. Combining both LBP and HOG further improved accuracy, with SVM reaching 88.7%. This enhancement underscores the complementary strengths of LBP and HOG in capturing distinctive texture patterns relevant to script identification.

Incorporating deep features significantly boosted classification performance. When using deep features extracted from GoogLeNet, SVM achieved 84.5% accuracy, demonstrating the effectiveness of deep learning in capturing hierarchical and semantic script representations. Performance improved further when deep features were combined with texture descriptors. The highest accuracy was observed with the combination of deep and HOG features: SVM reached 95.0%, NN achieved 94.2%, and KNN recorded 90.4%. Additionally, combining deep features with both LBP and HOG yielded strong results, with SVM attaining 92.1% accuracy. These findings highlight the advantage of integrating deep learning with traditional texture-based methods for robust script identification.

SVM proved to be the most reliable classifier, consistently outperforming KNN and NN across all scenarios. Its strength lies in its ability to handle high-dimensional feature spaces while maintaining strong generalization performance. The high accuracy achieved by deep feature-based approaches—particularly when fused with texture features—demonstrates their effectiveness in addressing the complexities of script identification.

Future research could explore alternative deep learning architectures or advanced feature fusion techniques to further improve classification accuracy. Moreover, evaluating this approach on a broader range of scripts and under real-world conditions, such as varying resolutions and noise, would provide deeper insights into its practical applicability and robustness.

3. 1 Comparative analysis

Conducting a comparative analysis is crucial for understanding advancements and challenges in script identification, particularly for camera-based Dravidian language scripts. The following table presents a detailed comparison of the proposed approach with existing methods.

Table 4: Comparative analysis.

Ref	Accuracy	Dataset	Remarks
[12]	91.00%	Camera-based document images	Use traditional statistical methods with moderate success. Real-world adaptability is limited by controlled conditions.
[13]	90.23% (ICDAR-17)	SIW-13, CVSI2015, ICDAR-17, MLe2e	Effective for dynamic scenarios like video frames but not tested on Dravidian scripts.
[14]	90.00%	Camera-based scene text	Focuses on multi-script scene text components with specific feature extraction techniques.
[15]	90.00%	Scene images	High accuracy for specific applications but lacks scalability to other contexts.
Proposed Method	95.00%	Custom dataset of South Indian scripts (12,000 images)	Demonstrates the advantage of combining deep learning and texture-based features for robustness and scalability.

4. CONCLUSION

This study explored an ensemble approach that combines deep learning features extracted from the GoogLeNet model with texture features such as LBP and HOG for South Indian script identification. The integration of deep and HOG features, when used with the SVM classifier, achieved a peak recognition accuracy of 95.0%, demonstrating the effectiveness of combining deep and texture-based descriptors. While CNNs effectively captured global structural patterns, texture features contributed critical edge and shape information, enhancing the overall recognition performance. Among the classifiers evaluated, SVM consistently outperformed KNN and NN, proving to be the most effective in handling multi-dimensional feature spaces.

Despite achieving high recognition accuracy, certain limitations were identified. The use of a self-constructed dataset collected under controlled conditions limits the method's applicability in real-world scenarios. Furthermore, combining features through simple concatenation may not fully exploit the complementarity of the feature types. Future work should address these limitations by:

- Utilizing real-world datasets that reflect varying imaging conditions,
- Investigating more sophisticated feature fusion strategies, and
- Expanding the study to include a wider range of scripts and multi-script environments.

Additionally, system robustness could be further enhanced by incorporating deep learning techniques designed to handle noise, illumination changes, and resolution variability in practical settings.

DECLARATIONS

Conflict of Interest: The authors declare that there is no conflict of interest.
Funding: This research received no external funding.
Availability of data and materials: No data is available in this article.
Publisher's note: The Journal and Publisher remain neutral about jurisdictional claims in published maps and institutional affiliations.

REFERENCES

[1] Steever SB. Introduction to the Dravidian languages. In The Dravidian languages 2019 Dec 18 (pp. 1-44). Routledge.
[2] Konya IV, Mare B. Adaptive methods for robust document image understanding (Doctoral dissertation, Universitäts-und Landesbibliothek Bonn). 10.1109/TKDE.2022.3192842

[3] Yang K, Yi J, Chen A, Liu J, Chen W, Jin Z. ConvPatchTrans: A script identification network with global and local semantics deeply integrated. Engineering Applications of Artificial Intelligence. 2022 Aug 1;113:104916. 10.1016/j.engappai.2022.104916
[4] Ghalati MK, Nunes A, Ferreira H, Serranho P, Bernardes R. Texture analysis and its applications in biomedical imaging: A survey. IEEE Reviews in Biomedical Engineering. 2021 Sep 27;15:222-46. 10.1109/RBME.2021.3115703
[5] Rane J, Mallick SK, Kaya O, Rane NL. Scalable and adaptive deep learning algorithms for largescale machine learning systems. Future Research Opportunities for Artificial Intelligence in Industry. 2024;4:39-92.
[6] Roy SK, Dubey SR, Chanda B, Chaudhuri BB, Ghosh DK. Texfusionnet: an ensemble of deep cnn feature for texture classification. InProceedings of 3rd International Conference on Computer Vision and Image Processing: CVIP 2018, Volume 2 2020 (pp. 271-283). Springer Singapore. 10.1007/978-981-32-9291-8_22
[7] Mallappa S, Dhandra BV, Mukarambi G. Hybridization of texture features for identification of Bi-lingual scripts from camera images at wordlevel. InComputer Vision and Machine Intelligence Paradigms for SDGs: Select Proceedings of ICRTAC-CVMIP 2021 2023 Jan 1 (pp. 113-124). Singapore: Springer Nature Singapore. 10.1007/978-981-19-7169-3_11
[8] Gupta MK, Dhawan S, Kumar A. Document Image Script Identification using Deep Network. In2024 11th International Conference on Signal Processing and Integrated Networks (SPIN) 2024 Mar 21 (pp. 174-179). IEEE. 10.1109/SPIN60856.2024.10511557
[9] Naosekpam V, Sahu N. A hybrid scene text script identification network for regional Indian languages. ACM Transactions on Asian and Low-Resource Language Information Processing. 2024 Aug 8;23(8):1-26. https://doi.org/10.1145/36494
[10] Li X, Zhan H, Shivakumara P, Pal U, Lu Y. SANet-SI: A new self-attention-network for script identification in scene images. Pattern Recognition Letters. 2023 Jul 1;171:45-52. 10.1016/j.patrec.2023.04.015
[11] Nicolaou A, Bagdanov AD, Gomez L, Karatzas D. Visual script and language identification. In2016 12th IAPR workshop on document analysis systems (DAS) 2016 Apr 11 (pp. 393-398). IEEE. 10.1109/DAS.2016.63
[12] Li L, Tan CL. Script identification of camera-based images. In2008 19th International Conference on Pattern Recognition 2008 Dec 8 (pp. 1-4). IEEE. 10.1109/ICPR.2008.4760965
[13] Bhunia AK, Konwer A, Bhunia AK, Bhowmick A, Roy PP, Pal U. Script identification in natural scene image and video frames using an attention based convolutional-LSTM network. Pattern Recognition. 2019 Jan 1;85:172-84. https://doi.org/10.1016/j.patcog.2018.07.034
[14] Jajoo M, Chakraborty N, Mollah AF, Basu S, Sarkar R. Script identification from camera-captured multi-script scene text components. InRecent Developments in Machine Learning and Data Analytics: IC3 2018 2019 (pp. 159-166). Springer Singapore. 10.1007/978-981-13-1280-9_16
[15] Fasil OK, Manjunath S, Aradhya VM. Word-level script identification from scene images. In Proceedings of the 5th International Conference on Frontiers in Intelligent Computing: Theory and Applications: FICTA 2016, Volume 2 2017 (pp. 417-426). Springer Singapore. 10.1007/978-981-10-3156-4_43
[16] Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, Erhan D, Vanhoucke V, Rabinovich A. Going deeper with convolutions. InProceedings of the IEEE conference on computer vision and pattern recognition 2015 (pp. 1-9). 10.1109/CVPR.2015.7298594.
[17] Li G, Muller M, Thabet A, Ghanem B. DeepGCNs: Can GCNs go as deep as CNNs?. In Proceedings of the IEEE/CVF international conference on computer vision 2019 (pp. 9267-9276). http://dx.doi.org/10.1109/ICCV.2019.00936.
[18] Touvron H, Cord M, Sablayrolles A, Synnaeve G, Jégou H. Going deeper with image transformers. InProceedings of the IEEE/CVF international conference on computer vision 2021 (pp. 32-42). https://doi.org/10.48550/arXiv.2103.17239
[19] Kishore A, Singh S. Natural language image descriptor. In 2015 IEEE Recent Advances in Intelligent Computational Systems (RAICS) 2015 Dec 10 (pp. 110-115). IEEE. 10.1109/RAICS.2015.7488398
[20] Ojala T, Pietikainen M, Maenpaa T. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. IEEE Transactions on pattern analysis and machine intelligence. 2002 Aug 7;24(7):971-87. 10.1109/TPAMI.2002.1017623
[21] García-Olalla O, Alegre E, Fernández-Robles L, García-Ordás MT, García-Ordás D. Adaptive local binary pattern with oriented standard deviation (ALBPS) for texture classification. EURASIP journal on image and video processing. 2013 Dec;2013:1-1. 10.1186/1687-5281-2013-31
[22] Dalal N, Triggs B. Histograms of oriented gradients for human detection. In2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05) 2005 Jun 20 (Vol. 1, pp. 886-893). IEEE. 10.1109/CVPR.2005.177
[23] Déniz O, Bueno G, Salido J, De la Torre F. Face recognition using histograms of oriented gradients. Pattern recognition letters. 2011 Sep 1;32(12):1598-603. 10.1016/j.patrec.2011.01.004
[24] Rane N, Choudhary SP, Rane J. Ensemble deep learning and machine learning: applications, opportunities, challenges, and future directions. Studies in Medical and Health Sciences. 2024 Jul 4;1(2):18-41. 10.48185/smhs.v1i2.1225
[25] H. K. Bhargav, Ambresh Bhadrashetty, K. Neelashetty, V. B. Murali Krishna, G. Manohar Bali. A Graph-Based and Pattern Classification Approach for Kannada Handwritten Text Recognition Under Struck-Out Conditions. IJCESEN [Internet]. 2025 Mar. 4 [cited 2025 Jun. 30];11(1). Available from: https://www.ijcesen.com/index.php/ijcesen/article/view/1021