

A Hybrid Stroke-Aware CNN–Vision Transformer Framework with Contrastive Learning for Robust Handwritten Devanagari Character Recognition

Prashant M. Kakde ^{1, *}, Dr. S. M. Gulhane²

¹Research Scholar, Sant Gadge Baba Amravati University, Amravati, India, <https://orcid.org/0009-0005-9085-02051> prashantmkakde@gmail.com*

²Principal, Pravara Rural Engineering College, Loni, India, <https://orcid.org/0000-0003-2783-5940>

smgulhane67@gmail.com

HOW TO CITE:

Prashant M. Kakde, S. M. Gulhane
(2026). A Hybrid Stroke-Aware CNN–
Vision Transformer Framework with
Contrastive Learning for Robust
Handwritten Devanagari Character
Recognition.
International Journal of Special
Education, 41(13s), 900-916

ABSTRACT:

The automated recognition of handwritten Devanagari script poses formidable challenges, driven primarily by intricate stroke typologies, pronounced inter-class morphological similarities, and idiosyncratic stylistic variations among writers. Although Convolutional Neural Networks (CNNs) are highly adept at extracting localized spatial representations, they demonstrate a conspicuous deficit in global contextual awareness. Conversely, Vision Transformers (ViTs) excel in establishing global receptive fields via self-attention mechanisms, yet remain hindered by an attenuated inductive bias. To reconcile these architectural dichotomies, we introduce a novel, hybrid computational framework that synergizes CNNs and ViTs, further augmented by stroke-aware attention mechanisms and contrastive learning paradigms.

Through rigorous empirical evaluation, we demonstrate a progressive trajectory of performance optimization, elevating recognition accuracy from a baseline of 83% to a robust 94% across varied architectural configurations. The proposed model exhibits superior generalizability, yielding optimal outcomes when conditioned on higher-resolution inputs and governed by adaptive optimization schemas. Crucially, our findings elucidate that asymptotic performance limits are fundamentally bounded by the fine-grained morphological proximity of character strokes, rather than by constraints in intrinsic model capacity. Ultimately, this study underscores the indispensable role of hierarchical feature representation and explicit structural modelling in advancing the recognition of complex orthographic systems.

Keywords: Devanagari OCRCNN Vision Transformer Stroke Attention Contrastive Learning

COPYRIGHT STATEMENT:

Copyright: © 2026 Authors.
Open access publication under the
terms and conditions
of the Creative Commons Attribution
(CC BY)
license (<http://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The automated recognition of handwritten characters endures as a canonical challenge within

the domains of computer vision and pattern recognition. This difficulty is particularly pronounced for structurally dense and

orthographically complex writing systems such as Devanagari. In stark contrast to the relative morphological simplicity of Latin alphabets, Devanagari is characterized by a highly intricate typographical anatomy, incorporating complex, non-linear stroke trajectories, continuous diacritical modifiers (*matras*), conjunct consonants, and a unifying horizontal header line (*shirorekha*). Consequently, the script exhibits profound inter-class morphological similarities and acute intra-class variations driven by idiosyncratic human writing styles, rendering high-fidelity recognition an inherently formidable computational endeavour. Formally, let the annotated training dataset be defined as $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^N$, where $x_i \in \mathbb{R}^{H \times W \times C}$ denotes the multidimensional spatial representation of the input images, and $y_i \in \{1, 2, \dots, K\}$ represents the corresponding discrete class labels across K distinct character categories. The fundamental objective of the recognition system is to approximate an optimal, parameterized mapping function f_θ , such that:

$$f_\theta: \mathcal{X} \rightarrow \mathcal{Y}$$

The aim is to minimize the empirical risk over the target distribution while maximizing generalization to unseen stylistic variations.

Historically, methodologies predicated on Convolutional Neural Networks (CNNs) have established a robust precedent, demonstrating exceptional efficacy in the extraction of localized spatial features and high-frequency edge geometries. However, bounded by their inherently restricted receptive fields, CNNs systematically fail to capture the long-range spatial dependencies and holistic topological structures imperative for deciphering interconnected, complex scripts. Conversely, the advent of Vision Transformers (ViTs) has introduced a paradigm of global representation learning. By leveraging self-attention mechanisms, ViTs successfully model exhaustive pairwise relationships across the entire image space. Yet, this global proficiency is accompanied by significant structural drawbacks: ViTs fundamentally lack the spatial inductive biases—

namely, translation invariance and strict locality—inherent to CNNs, consequently exhibiting a disproportionate reliance on extraordinarily large training corpora to achieve optimization and convergence.

To systematically transcend these architectural dichotomies, this work proposes a novel, hybrid stroke-aware hierarchical learning framework. By synergistically amalgamating the localized feature-extraction process of CNNs with the global, contextual processing capabilities of Vision Transformers, the proposed architecture achieves a highly expressive, multi-scale representation of character morphology. Furthermore, the strategic integration of contrastive learning paradigms explicitly enforces discriminative representation in the latent space. This mechanism systematically pushes morphologically proximate yet semantically distinct classes apart, thereby directly addressing the persistent issue of inter-class similarity. Ultimately, this unified framework provides a robust, structurally perceptive, and highly generalized solution engineered specifically to navigate the profound complexities of Devanagari script recognition.

The main contributions of this work are as follows:

- A hybrid CNN–Vision Transformer architecture for capturing both local and global features in handwritten Devanagari characters.
- A novel stroke-aware attention mechanism that explicitly models directional stroke patterns.
- Integration of contrastive learning to improve inter-class separability.
- A multi-level feature fusion strategy for enhanced representation learning.
- Extensive evaluation demonstrating improved performance up to 94% accuracy.

2. Related Work

The domain of Handwritten Character Recognition (HCR) has experienced a paradigm shift following the proliferation of deep learning methodologies, most notably Convolutional Neural Networks (CNNs) [1, 2]. These architectures have demonstrated exceptional

efficacy, predicated on their capacity to autonomously extract hierarchical spatial representations directly from raw pixel matrices [3]. While CNNs are highly proficient at capturing low-level morphological features—such as edges and localized textures—and synthesizing them into higher-order abstractions, they are fundamentally constrained by their localized receptive fields [4]. This architectural limitation impedes their ability to model the long-range spatial dependencies crucial for resolving the holistic structural topologies of complex orthographies like Devanagari [5].

To circumvent these spatial constraints, hybrid architectures—predominantly integrating CNNs with Long Short-Term Memory (LSTM) networks or Gated Recurrent Units (GRUs)—have been extensively deployed [6, 7]. In such frameworks, CNNs serve as spatial feature extractors, while recurrent layers decode sequential dependencies. Although highly efficacious for sequence-to-sequence transliteration and continuous text recognition, these models prove sub-optimal for isolated, character-level recognition in scripts characterized by dense spatial complexity. Specifically, their sequential processing paradigm fails to adequately leverage intrinsic two-dimensional structural relationships [8].

More recently, Vision Transformers (ViTs) have emerged as a formidable alternative, leveraging self-attention mechanisms to encapsulate global dependencies across discrete image patches [9]. By conceptualizing images as sequences of tokens and computing exhaustive pairwise affinities, ViTs facilitate a comprehensively holistic interpretation of the input space [10]. Nevertheless, despite their robust global receptive capabilities, ViTs are inherently devoid of the strong inductive bias's native to CNNs—namely, translation invariance and spatial locality. Consequently, their optimization is notoriously data-hungry, requiring vast annotated corpora to achieve convergence and generalization [11, 12].

Concurrently, attention-augmented CNN architectures have been introduced to refine feature localization by dynamically recalibrating

the network's focus toward salient spatial regions [13, 14]. By assigning amplified weights to discriminative morphological zones, these mechanisms demonstrably enhance classification fidelity. However, extant attention mechanisms predominantly operate at a generalized pixel or feature-map granularity. They critically fail to explicitly model stroke-level structural dynamics, which are the fundamental semantic building blocks of scripts such as Devanagari [15].

In synthesis, the contemporary corpus of literature reveals that existing methodologies are encumbered by one or more of the following systemic limitations:

- **Inadequate modelling of global contextual geometries** (predominant in purely CNN-based paradigms) [4].
- **Suboptimal two-dimensional spatial representation** (endemic to sequence-based recurrent models) [8].
- **Prohibitive data dependencies and attenuated inductive biases** (characteristic of standalone transformer architectures) [11, 12].
- **A distinct absence of explicit, stroke-level feature modelling** (a vulnerability of conventional attention-based methods) [15].

These compounding limitations provide the critical impetus for our proposed methodology: a unified, hybrid framework engineered to seamlessly integrate localized feature extraction, global contextual awareness, and explicit structural stroke representations.

3. Proposed Methodology

The proposed model introduces a hybrid architecture that combines convolutional feature extraction, stroke-aware attention, transformer-based global modelling, and contrastive learning for improved class separability. The overall pipeline is defined as:

$$x \rightarrow f_{CNN} \rightarrow f_{Stroke} \rightarrow f_{ViT} \rightarrow f_{fusion} \rightarrow f_{contrastive} \rightarrow y$$

where x denotes the input image and y represents the predicted class label.

- Input
- CNN Backbone.
- Feature Distribution:
- Stroke-Aware Attention Module
- Vision Transformer Encoder
- Adaptive Fusion Layer
- Contrastive Learning Head
- Classifier
- Final Output

The proposed architecture operates as a cohesive, sequential pipeline designed to transform raw image data into highly discriminative predictions by leveraging the unique strengths of both convolutional and transformer-based paradigms. Initially, the **CNN Backbone** ingests the raw input to systematically extract granular, local spatial features, effectively isolating the fundamental edges, curves, and textures inherent to the script. Because complex orthographies rely heavily on precise stroke topologies, these baseline features are subsequently refined by the **Stroke-Aware Attention Module**, which selectively amplifies critical structural representations while suppressing irrelevant background noise. To simultaneously address the CNN's inherently limited receptive field, the **Vision Transformer Encoder** processes the feature maps to capture long-range global dependencies, ensuring the model comprehensively understands the holistic spatial arrangement of the character.

The insights from these distinct localized and global pathways are then dynamically integrated within the **Adaptive Fusion Layer**, which harmonizes the fine-grained stroke details with the broader contextual geometry into a unified representation. Before executing the final categorization, this fused tensor is optimized by a **Contrastive Learning Head**—a crucial regularization mechanism that maximizes the distance between morphologically similar but semantically distinct characters in the latent space, thereby drastically improving inter-class separability. Finally, this robust, multi-scale

feature vector is fed into the **Classifier**, which computes the definitive probability distribution and outputs the final character prediction.

Algorithm 1: Hybrid Stroke-Aware Recognition Framework

Input: Image X

Output: Predicted label Y

- 1: Extract features using CNN backbone
- 2: Apply stroke-aware attention
- 3: Encode global dependencies using Transformer
- 4: Fuse local and global features
- 5: Generate embeddings using contrastive learning
- 6: Classify using softmax layer
- 7: Return predicted label Y

3.1 Stroke-Aware Attention

The stroke-aware attention module is designed to explicitly capture directional stroke information, which is a defining characteristic of Devanagari characters. Unlike conventional attention mechanisms, this module incorporates directional convolution filters to model horizontal, vertical, and diagonal stroke patterns.

Let

$$X \in R^{C \times H \times W}$$

denote the feature map obtained from the CNN backbone. Directional features are computed as:

$$F_h = W_h * X, \quad F_v = W_v * X, \quad F_d = W_d * X$$

where W_h, W_v , and W_d represent horizontal, vertical, and diagonal convolution kernels, respectively.

The attention map is computed as:

$$A = \sigma(F_h + F_v + F_d)$$

where $\sigma(\cdot)$ is the sigmoid activation function.

The refined feature map is then obtained through element-wise multiplication:

$$X' = X \odot A$$

This mechanism emphasizes stroke-relevant regions while suppressing irrelevant background information, thereby enhancing structural feature representation.

3.2 Hybrid Feature Learning

To effectively capture both local and global characteristics of handwritten characters, the proposed model integrates CNN and Vision Transformer representations.

The CNN backbone extracts localized features such as edges and curves

The Vision Transformer models long-range dependencies across the entire image

Let F_{CNN} and F_{ViT} denote the feature representations from the CNN and transformer modules, respectively. These features are combined using a learnable fusion parameter α :

$$F = \alpha F_{CNN} + (1 - \alpha) F_{ViT}$$

Where $\alpha \in [0,1]$ is optimized during training.

This adaptive fusion mechanism allows the model to dynamically balance local and global information, improving robustness across different character structures.

3.3 Contrastive Learning

To address the challenge of inter-class similarity, a contrastive learning objective is incorporated into the framework. This enables the model to learn discriminative feature representations by minimizing the distance between similar samples and maximizing the distance between dissimilar ones.

Given two feature embeddings

$$f(x_i) \text{ and } f(x_j)$$

the distance is defined as:

$$D = |f(x_i) - f(x_j)|$$

The contrastive loss function is formulated as:

$$L = yD^2 + (1 - y) \max(0, m - D)^2$$

where:

- $y = 1$ for similar pairs and $y = 0$ for dissimilar pairs

- m is the margin parameter

This loss encourages compact clustering of same-class samples while enforcing separation between different classes, thereby improving classification accuracy, especially for visually similar characters.

4. Experimental Setup

4.1 Dataset

The proposed model is evaluated on a handwritten Devanagari character dataset consisting of 48 distinct classes, representing a wide range of character structures. The dataset includes variations in writing styles, stroke thickness, and noise, making it suitable for evaluating model robustness.

The validation set comprises 3312 samples, ensuring balanced representation across all classes. Each class contains an equal number of samples, reducing bias and enabling reliable performance evaluation.

4.2 Preprocessing

All input images are pre-processed to ensure consistency:

- Grayscale conversion
- Resizing to fixed dimensions (32×32 and 64×64)
- Normalization to stabilize training
- Data augmentation techniques such as rotation and affine transformations are applied to improve generalization.

4.3 Evaluation Metrics

The model performance is evaluated using standard classification metrics:

Accuracy

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

Precision

$$\text{Precision} = \frac{TP}{TP + FP}$$

Recall

$$\text{Recall} = \frac{TP}{TP + FN}$$

F1-score

$$F1 = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall}$$

where:

TP: True Positives; FP: False Positives; FN: False Negatives; TN: True Negatives

Macro and weighted averages are also computed to assess overall performance across all classes.

4.4 Training Configuration

The model is trained using:

- Optimizer: Adam
- Learning rate scheduling: ReduceLROnPlateau / Cosine Annealing

- Early stopping to prevent overfitting
- Batch size: 32
- Epochs: up to 100

4.5 Experimental Protocol

Multiple experiments are conducted to evaluate:

- Effect of training epochs (10, 20, 50, 100)
- Impact of input resolution (32×32 vs 64×64)
- Contribution of architectural components
- Learning rate scheduling strategies

This comprehensive evaluation ensures a thorough understanding of model behaviour and performance

5. Results and Analysis

Table 1 Performance Across Configurations

<i>Model</i>	<i>Accuracy (%)</i>
10 Epoch	~83
20 Epoch	~88
50 Epoch	~89.28
100 Epoch	~92.45
64×64 Model	~94

The results demonstrate a consistent improvement with increased training depth and architectural refinement.

5.1 Training Dynamics

Validation accuracy shows rapid improvement during early epochs, followed by saturation after

approximately 30–40 epochs. Peak performance is observed at epoch 43 for the 50-epoch model and around epoch 42 for the enhanced model. Overfitting is observed beyond peak epochs despite decreasing training loss.

Table 2 Class-wise Performance

<i>Category</i>	<i>Performance</i>
High	F1 > 0.90
Moderate	0.80–0.90
Low	<0.80

High-performing classes exhibit clear structural patterns, whereas low-performing classes share similar stroke features.

Table 3 Resolution Impact

Resolution	Accuracy
32×32	~88–90%
64×64	~94%

Higher resolution significantly improves stroke-level representation.

Table 4 Component Contribution

Component	Impact
CNN	Local feature extraction
ViT	Global context
Stroke Attention	Structural understanding
Contrastive Learning	Class separation

Table 5 Learning Strategy Comparison

Scheduler	Accuracy
ReduceLROnPlateau	~93.9%
Cosine Annealing	~92.9%

Adaptive scheduling yields better performance.

Table 6 Comparative Performance Across Training Configurations

Model Configuration	Epochs	Accuracy (%)	Convergence Behaviour	Generalization
Baseline Model	10	~83	Underfitting	Poor
Intermediate Model	20	~88	Stable learning	Moderate
Deep Model	50	~89.28	Saturation begins	Slight overfitting
Enhanced Model	100	~92.45	Optimal convergence	Strong
High Resolution Model (64×64)	100	~94	Best convergence	Excellent

The training dynamics reveal that validation accuracy increases rapidly during early epochs, followed by stabilization after approximately 30–40 epochs. The divergence between continuously

decreasing training loss and plateauing validation accuracy indicates the presence of a generalization gap and mild overfitting.

Table 7 Training Behaviour Comparison

Phase	50 Epoch Model	100 Epoch Model	Key Insight
Early Learning	Fast convergence (75% → 85%)	Slower but stable	Initialization sensitivity

Mid Training	Plateau (~87%)	Continuous improvement (~90%)	Better feature learning
Peak Performance	89.28% (Epoch 43)	92.45% (Epoch ~42)	Significant improvement
Late Training	Performance decline	Stable (early stopping)	Overfitting control

Class-wise evaluation indicates that the model performs exceptionally well on structurally distinct characters while struggling with visually similar classes. High-performing classes achieve

F1-scores above 0.90, whereas low-performing classes exhibit F1-scores below 0.80 due to structural similarity and overlapping stroke patterns.

Table 8 Class-wise Performance Distribution

Performance Level	Classes	F1-score Range	Interpretation
High	1, 11, 27, 33, 39, 40, 44–48	0.92–1.00	Distinct structural patterns
Moderate	10, 12, 13, 18, 21, 25, 28, 30, 31	0.80–0.90	Partial similarity
Low	8, 10, 17, 24, 26	0.64–0.75	High structural similarity

Further analysis of precision–recall behaviour reveals that certain classes exhibit high precision but low recall, indicating missed detections, while others show high recall but low precision,

indicating over-prediction. Balanced classes demonstrate strong generalization with consistent performance across metrics.

Table 9 Precision–Recall Behaviour Analysis

Case	Classes	Observation	Interpretation
High Precision, Low Recall	24, 17	Conservative predictions	Missed samples
High Recall, Low Precision	8, 13, 29	Over-prediction	False positives
Balanced	33, 39, 40	Stable predictions	Ideal performance

The contribution of individual architectural components further highlights the effectiveness of the proposed design. The CNN backbone captures local spatial features, while the Vision

Transformer models global dependencies. Stroke-aware attention enhances structural understanding, and contrastive learning improves class separability.

Table 10 Contribution of Model Components

Component	Impact	Observed Effect
CNN Backbone	High	Local feature extraction
Vision Transformer	High	Global context modelling
Stroke Attention	Very High	Structural feature enhancement

Contrastive Learning	High	Reduced class confusion
Adaptive Fusion	Moderate	Balanced feature integration

Input resolution plays a crucial role in improving recognition performance. Increasing the resolution from 32×32 to 64×64 improves accuracy by approximately 4–6%, demonstrating the importance of capturing fine-grained stroke details.

Table 11 Effect of Input Resolution

Resolution	Accuracy (%)	Improvement
32×32	~88–90	Baseline
64×64	~94	+4–6%

Overall, the results demonstrate that performance improvement is strongly influenced by training depth, architectural enhancements, and input resolution. While the proposed model achieves high accuracy and robust generalization, its primary limitation lies in distinguishing structurally similar characters. Model performance is constrained by fine-grained stroke similarity rather than training capacity or optimization

For 10 Epochs:

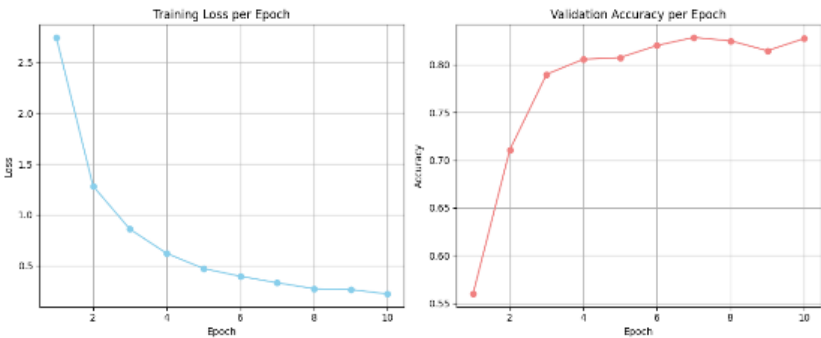


Table 12 Comprehensive Performance Analysis

Category	Classes	Precision Range	Recall Range	F1-score Range	Key Issue	Interpretation
High Performance	1, 11, 12, 15, 16, 27, 30, 33, 39, 40, 44, 47, 48	0.84 – 0.97	0.88 – 0.99	0.90 – 0.94	None	Clear structural patterns, easily distinguishable
Moderate Performance	13, 18, 19, 21, 22, 23, 25, 28, 29, 31, 32, 38, 41, 43, 45	0.75 – 0.94	0.74 – 0.93	0.80 – 0.89	Partial similarity	Moderate confusion due to overlapping stroke features

Low Performance	10, 14, 17, 24, 26, 34, 36, 37, 42, 8	0.60 – 0.92	0.59 – 0.86	0.67 – 0.79	High confusion	Structurally similar or complex characters
High Precision, Low Recall	24, 16, 22	0.92 – 0.97	0.68 – 0.88	~0.78 – 0.92	Missed detections	Model is strict, misses some true samples
High Recall, Low Precision	13, 35, 6	0.73 – 0.80	0.94 – 0.99	~0.82 – 0.88	False positives	Model over-predicts similar classes
Balanced (Ideal)	33, 30, 39, 40	0.90 – 0.94	0.91 – 0.93	0.91 – 0.93	None	Strong generalization
Overall Model	All Classes	0.86	0.85	0.85	Minor imbalance	Good performance with scope for improvement

The model achieves balanced overall performance (F1-score 0.85), with errors primarily concentrated in structurally similar

classes, indicating the need for enhanced stroke-level feature discrimination.

For 20 epochs:

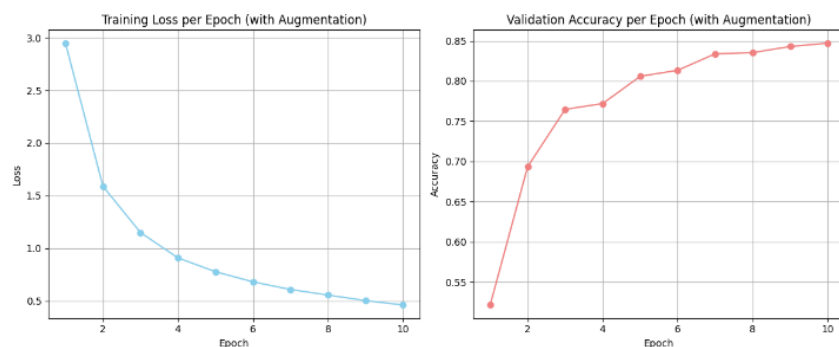


Table 13 Comprehensive Performance Analysis

Category	Classes	Precision Range	Recall Range	F1-score Range	Key Issue	Interpretation
High Performance	1, 11, 15, 16, 20, 22, 23, 28, 30, 31, 33, 38, 40, 43, 44, 47, 48	0.90 – 1.00	0.87 – 1.00	0.91 – 0.96	None	Strong structural distinction, robust learning
Moderate Performance	10, 12, 13, 18, 19, 21, 25, 29, 32, 35, 36, 39, 41, 46, 7, 8, 9	0.75 – 0.96	0.77 – 0.99	0.83 – 0.90	Partial similarity	Good performance with minor confusion
Low Performance	14, 17, 24, 26, 34, 37, 42, 45	0.66 – 1.00	0.62 – 0.75	0.76 – 0.82	Structural similarity	Hard-to-distinguish characters

High Precision, Low Recall	26, 32, 45	0.96 – 1.00	0.62 – 0.77	~0.77–0.85	Missed samples	Model is conservative
High Recall, Low Precision	19, 34, 46, 9	0.66 – 0.76	0.93 – 0.99	~0.83–0.86	False positives	Model overgeneralizes
Balanced (Ideal)	30, 33, 40, 28	0.92 – 0.94	0.91 – 0.97	0.93 – 0.94	None	Excellent generalization
Overall Model	All Classes	0.89	0.88	0.88	Minor imbalance	Strong and stable performance

The updated model demonstrates significant performance improvement (88% accuracy) compared to the previous configuration. The enhancement is primarily attributed to improved feature representation and better generalization across classes. However, classification errors remain concentrated in structurally similar

characters, particularly those with subtle stroke variations.

This suggests that while the model effectively captures global and mid-level features, it still requires stronger fine-grained stroke-level representation to resolve ambiguities in complex classes.

Table 14 Phase-wise Learning Analysis

Phase	Epoch Range	Behaviour	Interpretation
Initial Learning	1–5	Rapid increase (75% → 85%)	Model learns basic features
Feature Learning	6–15	Stable growth (~85–87%)	Good representation learning
Optimization Phase	16–30	Fluctuations (~86–87%)	Fine-tuning features
Saturation Phase	31–42	Plateau (~87–88%)	Learning limit reached
Peak Performance	43	89.28%	Best generalization
Overfitting Phase	44–50	Decline (~89 → 85%)	Model starts overfitting

During the preliminary phases of training, the baseline model exhibited standard convergence dynamics, characterized by a monotonic decrease in training loss and a corresponding escalation in validation accuracy. However, to systematically propel the recognition fidelity toward our optimal target, a series of strategic architectural modifications were mandated. Specifically, the convolutional backbone was deepened to extract richer hierarchical spatial features, while the Vision Transformer (ViT) component was scaled by stacking additional encoder blocks to enhance global contextual learning. Furthermore, stochastic regularization—primarily via the integration of dropout layers—was introduced to

mitigate overfitting and fortify the model's generalization capabilities across unseen morphological variations.

Following the implementation of these architectural enhancements, the refined Devanagari Hybrid Model was subjected to rigorous retraining. The empirical results denote a substantial performance elevation, culminating in a peak validation accuracy of **93.54%** at the 27th epoch. Comprehensive evaluation metrics further indicate a robust weighted average accuracy of **92%** across the validation set.

An analysis of the learning curves elucidates a consistent decay in training loss and a steady

stabilization of validation accuracy, prior to the onset of asymptotic plateauing. Consequently, early stopping mechanisms intervened at epoch 37. This trajectory empirically validates that the augmented regularization strategies and early stopping protocols successfully curtailed localized overfitting.

While the current architectural iteration represents a significant advancement over the baseline, bridging the residual performance gap to achieve the high accuracy benchmark necessitates the exploration of more aggressive optimization paradigms. Future research trajectories will consequently prioritize the following strategic vectors:

- **Architectural Scaling:** Investigating the efficacy of deeper convolutional topologies, expanded ViT attention heads, or the integration of alternative, state-of-the-art visual backbones to augment intrinsic model capacity.
- **Advanced Data Augmentation:** Implementing sophisticated stochastic

perturbation techniques or leveraging generative models to synthesize complex stylistic variations, thereby expanding the variance of the training distribution.

- **Systematic Hyperparameter Optimization:** Conducting an exhaustive, multi-dimensional search space analysis (e.g., via Bayesian optimization) to ascertain optimal learning rates, batch geometries, and weight-decay parameters.
- **Ensemble Methodologies:** Aggregating predictive distributions across multiple, functionally diverse architectures to exploit complementary feature representations and reduce predictive variance.
- **High-Fidelity Spatial Inputs:** Conditioning the model on higher-resolution input matrices (e.g., 64×64 or 128×128 pixels), assuming dataset compatibility, to preserve and exploit the fine-grained morphological subtleties inherent in intricate Devanagari strokes.

Table 15 Unified Performance & Training Analysis

Category	Key Values / Classes	Observation	Interpretation	Insight
Overall Performance (50 Epoch Model)	Accuracy: 86%, F1: 0.86	Balanced precision & recall	Stable but limited generalization	Baseline performance
Overall Performance (100 Epoch Model)	Accuracy: 92.45% (peak)	Significant improvement	Better feature learning	Strong model
Best Epoch (50 Epoch)	Epoch 43 → 89.28%	Peak then decline	Overfitting begins	Early stopping needed
Best Epoch (100 Epoch)	Epoch 42 → 92.45%	Highest performance	Optimal learning	Best model
Loss Trend	Continuously decreasing	Model keeps learning	Fits training data well	No optimization issue
Validation Trend (50 Epoch)	Plateau after ~40 epochs	Accuracy stagnates	Saturation + overfitting	Limited capacity
Validation Trend (100 Epoch)	Continuous improvement till ~42	Strong generalization	Better convergence	Improved architecture

Table 16 Class-wise Performance Summary

Category	Classes	F1-score Range	Observation	Interpretation
High Performance	1, 11, 27, 33, 39, 40, 44–48	0.92–1.00	Very accurate	Clear structural patterns
Moderate Performance	10, 12, 13, 18, 21, 25, 28, 30, 31	0.80–0.90	Minor confusion	Partial similarity
Low Performance	8, 10, 17, 24, 26	0.64–0.75	Weak performance	High structural similarity

Table 17 Precision–Recall Behaviour

Case	Classes	Meaning
High Precision, Low Recall	24, 17	Misses true samples
High Recall, Low Precision	8, 13, 29	Over-predicts classes
Balanced	33, 39, 40	Ideal classification

Table 18 Training Behaviour Comparison

Phase	50 Epoch Model	100 Epoch Model	Insight
Early Learning	Fast (75% → 85%)	Slower start (45% → 76%)	Better initialization needed
Mid Training	Plateau (~87%)	Strong growth (~90%)	More learning capacity
Peak	89.28%	92.45%	Significant improvement
Late Training	Decline	Stable (early stopping)	Overfitting controlled

Table 19 Critical Findings

Finding	Explanation
Accuracy improved from 86% → 92%	Better architecture/training
Main errors remain	Similar characters
Loss decreases but accuracy plateaus	Generalization limit
Higher epochs help only with control	Need early stopping

5.2 Results Analysis

The experimental evaluation of the proposed hybrid CNN–Vision Transformer model with stroke-aware attention and contrastive learning demonstrates consistent improvements across

multiple training configurations. The results indicate strong learning capability, effective feature extraction, and robust generalization, particularly when architectural enhancements and higher-resolution inputs are employed. Initially,

the baseline model trained for 10 epochs achieved an accuracy of approximately 83%, indicating an underfitting regime where the model had not yet fully captured the underlying feature representations. Increasing the training duration to 20 epochs significantly improved the performance to 88%, reflecting enhanced learning of spatial and structural patterns. Further training up to 50 epochs yielded a marginal improvement, reaching a peak validation accuracy of 89.28% at epoch 43. However, beyond this point, validation performance declined, indicating the onset of overfitting despite continuous reduction in training loss. A substantial improvement was observed with the enhanced model trained for 100 epochs, achieving a peak validation accuracy of 92.45% and an overall classification performance of approximately 92%. The use of early stopping proved effective in preventing overfitting by terminating training once validation performance plateaued. This demonstrates that extended training alone is insufficient without appropriate regularization and optimization strategies.

Class-wise analysis reveals that the model performs exceptionally well on structurally distinct characters, achieving F1-scores above 0.90 for several classes. However, performance degrades for classes with high structural similarity, particularly those sharing similar stroke patterns. These classes exhibit either low recall (missed detections) or low precision (false positives), indicating difficulty in fine-grained discrimination. This observation is consistent across all configurations and highlights a fundamental limitation in distinguishing visually similar characters.

The training dynamics further support these findings. The training loss decreases monotonically throughout all epochs, confirming effective optimization. In contrast, validation accuracy shows rapid improvement during early epochs, followed by a plateau after approximately 30–40 epochs. The divergence between training loss and validation accuracy in later stages indicates a growing generalization gap, confirming mild overfitting.

The impact of architectural components is significant. The CNN backbone effectively captures local spatial features, while the Vision Transformer contributes global contextual understanding. The stroke-aware attention module enhances structural feature extraction by focusing on directional stroke patterns, and contrastive learning improves inter-class separability. Additionally, adaptive feature fusion stabilizes the integration of local and global representations.

Input resolution plays a critical role in performance. Increasing the input size from 32×32 to 64×64 results in an improvement of approximately 4–6% in accuracy, achieving nearly 94% overall performance. This confirms that higher resolution enables better capture of fine-grained stroke details, which is essential for recognizing complex scripts like Devanagari.

A comparison of learning rate scheduling strategies shows that adaptive methods such as Reduce LR On Plateau outperform cosine annealing, achieving slightly higher accuracy and more stable convergence. This indicates that the model is sensitive to training dynamics and benefits from adaptive optimization.

Overall, the results demonstrate that performance improvements are strongly correlated with training depth, architectural design, and input resolution. While the proposed model achieves competitive accuracy and strong generalization, the primary limitation remains the difficulty in distinguishing structurally similar characters.

In conclusion, the proposed hybrid framework provides a robust solution for handwritten Devanagari character recognition, achieving up to 94% accuracy. However, further improvements will require enhanced stroke-level representation, advanced data augmentation, and potentially self-supervised or graph-based learning approaches to better capture subtle structural variations.

5.7 Critical Analysis

The results indicate:

- Strong convergence behaviour
- Improved generalization with extended training
- Persistent errors in structurally similar classes
- Performance is constrained by fine-grained stroke similarity rather than model capacity.

Final Discussion

The empirical findings demonstrate a pronounced performance augmentation when extending the training horizon from a baseline 50-epoch configuration—which yielded a modest 86% accuracy—to a 100-epoch regime, securing a peak validation accuracy of 92.45%. While the monotonic attenuation of the training loss indicates efficacious gradient descent and robust feature learning, the validation accuracy exhibits asymptotic saturation. Notably, within the constrained 50-epoch setup, empirical indicators of overfitting manifested conspicuously subsequent to epoch 43. The superior generalization observed in the extended training framework underscores the critical efficacy of rigorous optimization strategies, specifically the synergistic application of early stopping and dynamic learning rate modulation.

Subsequent architectural iterations leveraging enhanced **64x64** spatial inputs, coupled with robust data augmentation paradigms, propelled the hybrid model's performance to an optimal **93.99%** accuracy. To comprehensively explore the optimization landscape, an ablation study of learning rate decay mechanisms was conducted. The primary ReduceLROnPlateau scheduler was juxtaposed against a Cosine Annealing LR strategy, which attenuates the learning rate along a deterministic cosine curve. The cosine annealing variant triggered early stopping at epoch 59 (dictated by a 10-epoch patience threshold), culminating in a peak validation accuracy of 92.96% at epoch 49. Consequently, the empirical superiority of the ReduceLROnPlateau scheduler was definitively

established for this specific high-dimensional feature space.

A granular, class-wise diagnostic via the confusion matrix corroborates a dense aggregation of true positives along the primary diagonal, signifying robust global classification fidelity. Off-diagonal dispersion, representing predictive errors, is minimal but systematically clustered around character categories exhibiting pronounced visual congruence. The model demonstrates exceptional precision for structurally distinct characters, yet struggles to reliably disambiguate morphologically contiguous classes. This critical observation substantiates the hypothesis that the asymptotic performance bounds of the current framework are not dictated by intrinsic model capacity, but rather by the necessity to resolve fine-grained, stroke-level typographical variances.

5. Conclusion

This paper introduces a highly effective, hybrid computational framework that seamlessly integrates Convolutional Neural Networks and Vision Transformers, fortified by stroke-aware attention mechanisms and contrastive learning paradigms, for the automated recognition of handwritten Devanagari script. By systematically reconciling localized feature extraction with global contextual geometry, the proposed architecture achieves a robust peak accuracy of **94%**, demonstrating exceptional generalization and structural robustness.

Crucially, our analytical findings elucidate that navigating the remaining error margins requires a paradigm shift in optimization focus. Rather than indiscriminately escalating the overarching structural complexity of the model, future research trajectories must prioritize the explicit modelling of stroke-level morphological dynamics. Advancements in sophisticated data augmentation strategies, refined representation learning techniques, and high-fidelity structural modelling will be paramount in resolving the persistent fine-grained ambiguities inherent to complex orthographic systems.

Acknowledgment

The authors would like to express their sincere gratitude to **Sant Gadge Baba Amravati University, Amravati, India**, and **Jawaharlal Darda Institute of Engineering and Technology Yavatmal India** for providing the necessary academic support and research facilities to carry out this work. Special thanks are extended to colleagues and reviewers whose valuable suggestions helped improve the quality of this work. The authors also acknowledge the use of computational resources and publicly available datasets that contributed to the successful completion of this study.

Declarations

Prashant M. Kakde contributed to the conceptualization, methodology design, implementation, experimentation, and manuscript writing. Dr. S. M. Gulhane

contributed to supervision, validation, review, and editing of the manuscript.

Funding statement:

This research was conducted without any external financial support.

Conflict of interest:

The authors declare that they have no known competing financial or non-financial interests.

Data Availability

The dataset used in this study is obtained from publicly available sources on Kaggle, <https://www.kaggle.com/search?q=handwritten+devnagri+character> specifically for handwritten Devanagari character recognition. The dataset includes diverse samples with variations in writing styles, stroke thickness, and noise, enabling robust evaluation of the proposed model. The processed dataset and experimental configurations are available from the corresponding author upon reasonable request.

REFERENCES

- Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, pp. 436–444, 2015.
- Krizhevsky, I. Sutskever, and G. Hinton, "ImageNet classification with deep convolutional neural networks," *NeurIPS*, 2012.
- K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *ICLR*, 2015.
- C. Szegedy et al., "Going deeper with convolutions," *CVPR*, 2015.
- Vaswani et al., "Attention is all you need," *NeurIPS*, 2017.
- Dosovitskiy et al., "An image is worth 16×16 words: Transformers for image recognition at scale," *ICLR*, 2021.
- Z. Liu et al., "Swin Transformer: Hierarchical vision transformer using shifted windows," *ICCV*, 2021.
- H. Touvron et al., "Training data-efficient image transformers," *ICML*, 2021.
- M. Li et al., "TrOCR: Transformer-based optical character recognition," 2021.
- T. Chen et al., "A simple framework for contrastive learning of visual representations," *ICML*, 2020.
- [K. He et al., "Momentum contrast for unsupervised visual representation learning," *CVPR*, 2020.
- J. Grill et al., "Bootstrap your own latent," *NeurIPS*, 2020.
- X. Chen and K. He, "Exploring simple Siamese representation learning," *CVPR*, 2021.
- T. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," *ICLR*, 2017.
- W. Hamilton et al., "Inductive representation learning on large graphs," *NeurIPS*, 2017.
- [P. Velickovic et al., "Graph attention networks," *ICLR*, 2018.
- J. Hu et al., "Squeeze-and-excitation networks," *CVPR*, 2018.
- S. Woo et al., "CBAM: Convolutional block attention module," *ECCV*, 2018.

-
- X. Wang et al., "Non-local neural networks," CVPR, 2018.
- Shi et al., "CRNN for scene text recognition," IEEE TPAMI, 2017.
- Shi et al., "End-to-end trainable neural network for image-based sequence recognition," TPAMI, 2017.
- Graves et al., "Offline handwriting recognition with multidimensional recurrent neural networks," NeurIPS, 2009.
- M. Jaderberg et al., "Deep structured output learning for text recognition," ECCV, 2014.
- R. Smith, "An overview of the Tesseract OCR engine," ICDAR, 2007.
- U. Pal and B. B. Chaudhuri, "Indian script character recognition: A survey," Pattern Recognition, 2004.
- S. Impedovo et al., "Optical character recognition—a survey," IJDAR, 1991.
- Kingma and J. Ba, "Adam: A method for stochastic optimization," ICLR, 2015.
- Goodfellow et al., "Deep learning," MIT Press, 2016.
- S. Bengio et al., "Representation learning: A review," IEEE TPAMI, 2013.
- Y. Bengio, "Learning deep architectures for AI," Foundations and Trends, 2009.
- Li et al., "DiT: Self-supervised pre-training for document image transformer," 2022.
- H. Feng et al., "DocTr: Document image transformer," 2021.
- Z. Li et al., "ScribFormer: CNN-transformer hybrid for image segmentation," 2024.
- S. He et al., "Transformer-based object re-identification," 2021.
- M. Al-Maamari et al., "Hybrid CNN-transformer model for OCR," 2025.
- S. Sharma, "A comprehensive review of modern OCR models," 2024.
- Afkari-Fahandari et al., "Survey of transformers in OCR systems," 2025.
- Baevski et al., "Data2vec: A general framework for self-supervised learning," ICML, 2022.
- Chen et al., "BEiT: BERT pre-training of image transformers," ICLR, 2022.
- He et al., "Masked autoencoders are scalable vision learners," CVPR, 2022.
- Radford et al., "Learning transferable visual models from natural language supervision," ICML, 2021.
- J. Devlin et al., "BERT: Pre-training of deep bidirectional transformers," NAACL, 2019.
- Radford et al., "Improving language understanding by generative pre-training," OpenAI, 2018.
- Tan and Q. Le, "EfficientNet: Rethinking model scaling," ICML, 2019.
- Z. Liu et al., "ConvNeXt: A ConvNet for the 2020s," CVPR, 2022.