Tuijin Jishu/Journal of Propulsion Technology
ISSN: 1001-4055
Vol. 47 No. 02 (2026)

A Comprehensive Preprocessing and Feature Extraction Methodology for Traffic Sign Detection on Indian Roads Using the IRTSD-Datasetv1

Harminder Singh Deosi[1,2], Gurpreet Singh[1], Narinder Kumar[2]

[1] University Institute of Computing, Chandigarh University, Gharuan, Mohali, 140413

[2] Department of Statistics, Panjab University, Sector 14, Chandigarh, 160014

Abstract

Traffic sign detection is a fundamental component of intelligent transportation systems (ITS) and autonomous driving technologies. The accuracy of detection models is heavily dependent on the quality of input data and the effectiveness of preprocessing and feature extraction pipelines. Indian road environments present unique challenges, including diverse sign designs, variable illumination, weather-induced degradation, partial occlusions, and cluttered backgrounds. This paper presents a comprehensive preprocessing and feature extraction framework specifically designed for the Indian Road Traffic Sign Detection Dataset (IRTSD-Datasetv1), which comprises 5,141 images spanning 37 traffic sign classes collected from over 90 cities across India. The proposed framework integrates a multi-stage preprocessing pipeline encompassing image resizing and normalization, Contrast Limited Adaptive Histogram Equalization (CLAHE), Gaussian noise filtering, and morphological operations, followed by an advanced hybrid feature extraction methodology combining Histogram of Oriented Gradients (HOG), Local Binary Patterns (LBP), colour histogram analysis in HSV space, and Gabor filter-based texture features. The pre-processed dataset is further enriched through a systematic data augmentation strategy incorporating geometric transformations, photometric distortions, and GAN-oriented synthetic sample generation to address class imbalance. Experimental evaluation demonstrates that the proposed preprocessing pipeline achieves a 14.7% improvement in feature discriminability measured through Fisher's Discriminant Ratio, a 23.2% enhancement in Signal-to-Noise Ratio (SNR), and a 97.8% structural similarity index (SSIM) preservation. When integrated with a baseline Convolutional Neural Network (CNN), the pre-processed features yield an accuracy improvement from 89.3% to 96.7%, confirming the efficacy of the proposed framework. The findings establish a robust foundation for subsequent GAN-based traffic sign detection model development.

Keywords: Traffic Sign Detection; Image Preprocessing; Feature Extraction; CLAHE; HOG; LBP; Data Augmentation; IRTSD Dataset; Indian Roads; Intelligent Transportation Systems

1. Introduction

Traffic signs constitute a critical component of road infrastructure, serving as visual communication tools that regulate traffic flow, warn drivers of hazardous conditions, and provide navigational guidance. The automated detection and recognition of these signs have emerged as indispensable capabilities for intelligent transportation systems (ITS), advanced driver assistance systems (ADAS), and autonomous vehicle navigation platforms. The efficacy of such automated systems is fundamentally contingent upon the quality of input data and the robustness of the preprocessing and feature extraction methodologies employed in the computational pipeline. The global push toward autonomous driving has catalysed significant research activity in traffic sign detection, with numerous benchmark datasets and detection algorithms emerging in recent years. However, the majority of existing research has focused on traffic signs from European and North American contexts, utilizing datasets such as the German Traffic Sign Recognition Benchmark (GTSRB), the Belgian Traffic Sign Dataset (BTSD), and the Tsinghua-Tencent 100K (TT100K) dataset. Indian traffic environments present a fundamentally distinct set of challenges that render direct application of models trained on foreign datasets suboptimal. These challenges include: non-standardized sign designs with regional variations across states; multilingual text content on signboards; tropical weather conditions causing rapid degradation and colour fading; cluttered urban environments with dense signage; variable illumination conditions spanning harsh sunlight to monsoon-induced low visibility; and partial occlusions from vegetation, other vehicles, and infrastructure.

The Indian Road Traffic Sign Detection Dataset (IRTSD-Datasetv1), introduced by Jackulin Mahariba and Aryan (2024), represents a significant contribution toward addressing the scarcity of publicly available Indian traffic sign datasets. Comprising 5,141 images across 37 traffic sign classes collected from over 90 Indian cities under varying distances and lighting conditions, this dataset captures the real-world complexity of Indian road environments. However, the raw dataset, collected using mobile phone cameras, exhibits inherent variability in image quality, resolution, and environmental conditions that necessitate systematic preprocessing before it can be effectively utilized for training robust detection models.

Data preprocessing and feature extraction form the foundational stages of any computer vision pipeline. Preprocessing operations such as noise reduction, contrast enhancement, and normalization serve to standardize input data and mitigate the adverse effects of acquisition variability. Feature extraction, whether through handcrafted descriptors or learned representations, transforms raw pixel data into discriminative feature vectors that encode semantically meaningful information about traffic sign characteristics. The synergy between effective preprocessing and robust feature extraction directly determines the upper bound of subsequent detection and classification model performance.

2. Related Work

2.1 Traffic Sign Detection: An Overview

Traffic sign detection and recognition have been extensively studied in the computer vision literature. Early approaches relied on handcrafted features such as colour-based segmentation and shape analysis. De La Escalera et al. (1997) pioneered road sign detection using colour and geometric features. Subsequent developments incorporated machine learning classifiers, with the combination of Histogram of Oriented Gradients (HOG) features and Support Vector Machines (SVM) becoming a widely adopted pipeline for traffic sign recognition. However, these traditional methods exhibited significant sensitivity to illumination variations, shadows, and adverse weather conditions, limiting their robustness in complex real-world scenarios.

The advent of deep learning, particularly Convolutional Neural Networks (CNNs), revolutionized traffic sign detection by enabling automatic extraction of hierarchical discriminative features. Architectures such as LeNet-5, VGGNet, ResNet, and EfficientNet have demonstrated remarkable performance on standard benchmarks. Zaibi et al. (2021) proposed an enhanced LeNet-5 model achieving 99.84% accuracy on GTSRB through preprocessing techniques including histogram equalization, grayscale conversion, and normalization. More recently, object detection frameworks including YOLO variants (YOLOv5 through YOLOv11) and RT-DETR have dominated the field, offering real-time detection capabilities with high accuracy.

2.2 Preprocessing Techniques for Traffic Sign Detection

Image preprocessing constitutes a critical stage in traffic sign detection pipelines. Common preprocessing operations include image resizing, normalization, contrast enhancement, noise reduction, and colour space transformation. Contrast Limited Adaptive Histogram Equalization (CLAHE) has emerged as a particularly effective technique for traffic sign images. Unlike global histogram equalization, CLAHE divides the image into localized tiles and performs adaptive histogram equalization within each tile, preventing over-amplification of noise in homogeneous regions through a clip limit mechanism. Research has demonstrated that CLAHE integration with detection algorithms significantly improves performance in low-light and variable illumination conditions, with studies achieving up to 76.2% mAP (mean Average Precision) improvement in nighttime detection scenarios when combined with YOLOv9.

Gaussian and median filtering have been widely employed for noise reduction in traffic sign images. An et al. (2024) demonstrated that the combination of median filtering, histogram equalization, and vertical and horizontal histogram features with a multilayer perceptron achieved 98.72% classification accuracy. Colour space transformations, particularly conversion from RGB to HSV or Lab colour spaces, facilitate more robust colour-based segmentation by decoupling luminance from chrominance information.

2.3 Feature Extraction Methods

Feature extraction methods for traffic sign detection can be broadly categorized into handcrafted and learned approaches. Among handcrafted methods, HOG captures edge orientation distributions through local gradient computation, providing shape-discriminative features. LBP encodes local texture patterns through binary comparison of pixel intensities with their neighbourhood, offering rotation-invariant texture descriptors. Gabor filters extract frequency-domain texture features at multiple orientations and scales, capturing both fine and coarse structural patterns.

Deep learning-based feature extraction has largely superseded handcrafted methods in recent years. However, hybrid approaches that combine handcrafted features with deep features have shown promise, particularly in scenarios with limited training data. The multi-scale feature extraction capabilities of modern architectures such as Feature Pyramid Networks (FPN) and Bidirectional Feature Pyramid Networks (BiFPN) have proven especially effective for detecting traffic signs at varying scales.

2.4 Data Augmentation and GAN-based Approaches

Data augmentation plays a vital role in addressing the class imbalance and limited data availability inherent in traffic sign datasets. Traditional augmentation techniques include geometric transformations (rotation, flipping, scaling, cropping), photometric distortions (brightness adjustment, contrast modification, colour jittering), and synthetic noise injection. Advanced augmentation strategies such as Mosaic and MixUp have been incorporated into modern detection frameworks.

Generative Adversarial Networks (GANs) have emerged as powerful tools for synthetic data generation in traffic sign recognition. Deep Convolutional GANs (DCGANs) and Conditional GANs (CGANs) can generate realistic traffic sign images that improve classifier generalization. Pix2Pix architectures have been applied for symbolic-to-real image translation of traffic signs, while more recent approaches have incorporated GAN-based augmentation for low-visibility conditions through controllable data synthesis techniques.

2.5 Indian Traffic Sign Detection

Research specifically addressing Indian traffic sign detection remains relatively limited compared to European and Chinese contexts. Several studies have highlighted the unique challenges of Indian road environments, including non-standardized sign designs, multilingual content, and diverse environmental conditions. A Refined Mask R-CNN model evaluated on 6,480 Indian traffic sign images across 87 categories achieved 97.08% precision with data augmentation. YOLOv8-based transfer learning models have demonstrated 81.3% mAP for Indian traffic sign detection on datasets comprising 22,400 images. The IRTSD-Datasetv1 was evaluated using YOLOv8, YOLOv10, and RT-DETR algorithms, achieving up to 98.25% mAP, establishing it as a valuable benchmark for Indian traffic sign research.

2.6 Research Gap

Despite the growing body of literature on traffic sign detection, a systematic investigation of preprocessing and feature extraction techniques specifically optimized for Indian traffic sign datasets is conspicuously absent. Most existing studies apply generic preprocessing pipelines without consideration of the unique visual characteristics and environmental challenges of Indian road environments. Furthermore, the interplay between different preprocessing operations and their cumulative effect on feature discriminability has not been rigorously analysed.

3. Proposed Methodology

The proposed framework comprises four principal stages: dataset analysis and characterization, multi-stage preprocessing pipeline, hybrid feature extraction, and data augmentation with GAN-oriented synthesis.

3.1 Dataset Description and Analysis

The IRTSD-Datasetv1 (Indian Road Traffic Sign Detection Dataset, Version 1) serves as the primary dataset for this study. The dataset was introduced by Jackulin Mahariba A. and Aryan (2024), published through IEEE DataPort with DOI: 10.21227/ap4b-c439. The key characteristics of the dataset are summarized in Table 1

Parameter	Description
Total Images	5,141
Number of Classes	37 traffic sign categories
Collection Source	Over 90 cities across India
Capture Device	Mobile phone cameras
Image Format	PNG
Annotation Format	TXT (YOLO format bounding boxes)
Conditions	Varying distances, lighting, and weather
Train/Test Split	Not pre-divided (user-defined)

Table 1: Characteristics of the IRTSD-Datasetv1 Dataset

A preliminary analysis of the dataset revealed several characteristics demanding preprocessing intervention: variable image resolutions ranging from approximately 640×480 to 4000×3000 pixels; significant class imbalance with overrepresentation of common mandatory and prohibitory signs and underrepresentation of informatory signs; illumination inconsistency attributable to different capture times (dawn, daylight, dusk, artificial lighting); presence of motion blur in images captured from moving vehicles; and complex backgrounds including vegetation, buildings, and overlapping signage.

3.2 Multi-Stage Preprocessing Pipeline

The preprocessing pipeline is designed as a sequential cascade of operations, each addressing specific quality degradation factors. The pipeline is formalized as follows:

Let I_{raw} denote the original input image. The preprocessing transformation T is defined as:

$$I_{preprocessed} = T_{morph} \circ T_{denoise} \circ T_{CLAHE} \circ T_{norm} \circ T_{resize}(I_{raw})$$

3.2.1 Image Resizing and Resolution Standardization

All images are resized to a uniform resolution of 640×640 pixels using bilinear interpolation. This standardization is essential for batch processing compatibility and ensures consistent spatial feature extraction across the dataset. The aspect ratio is preserved through center-cropping with zero-padding for non-square images, preventing geometric distortion of sign shapes. The resizing operation is defined as:

$$I_{resized} = Resize(I_{raw}, 640, 640, interpolation = bilinear)$$

3.2.2 Pixel Normalization

Pixel values are normalized from the integer range [0, 255] to the floating-point range [0.0, 1.0] through min-max normalization. Subsequently, channel-wise mean subtraction and standard deviation division are applied using statistics computed from the entire training set:

$$I_{norm}(c) = (I(c) - \mu_c) / \sigma_c$$

where c ∈ {R, G, B} represents the colour channel, μ denotes the channel mean, and σ denotes the channel standard deviation computed over the training partition.

3.2.3 Contrast Limited Adaptive Histogram Equalization (CLAHE)

CLAHE is applied to enhance local contrast, which is particularly critical for images captured under non-uniform illumination conditions prevalent on Indian roads. The algorithm operates on the luminance channel after converting the image to the LAB colour space, thereby preserving colour information while enhancing contrast:

Step 1: Convert the normalized RGB image to LAB colour space: L, A, B = RGB2LAB(I).

Step 2: Divide the L channel into non-overlapping tiles of size 8×8.

Step 3: Compute the histogram for each tile and clip values exceeding the clip limit (CL = 2.0).

Step 4: Redistribute excess histogram bins uniformly across all bins.

Step 5: Compute the cumulative distribution function (CDF) for each tile.

Step 6: Apply bilinear interpolation to eliminate tile boundary artifacts.

Step 7: Reconvert to RGB: I_enhanced = LAB2RGB(L_CLAHE, A, B).

The clip limit parameter CL = 2.0 was empirically determined through grid search optimization over the range [0.5, 5.0] with step size 0.5, selecting the value that maximized the mean Structural Similarity Index (SSIM) while minimizing noise amplification measured through Peak Signal-to-Noise Ratio (PSNR).

3.2.4 Noise Reduction

A two-stage denoising approach is employed. First, a Gaussian filter with kernel size 3×3 and σ = 0.5 is applied for initial smoothing. Second, a bilateral filter with diameter d = 9, colour sigma σ_colour = 75, and spatial sigma σ_space = 75 is applied to preserve edge information while suppressing remaining noise. The bilateral filter is mathematically expressed as:

$$BF[I](x) = (1/W_p)\ \Sigma\ G_{\sigma s}(||x - x_i||) \cdot G_{\sigma r}(|I(x) - I(x_i)|) \cdot I(x_i)$$

where G denotes the Gaussian function, σs and σr are the spatial and range parameters respectively, and Wp is the normalization factor.

3.2.5 Morphological Operations

Morphological opening (erosion followed by dilation) with a 3×3 elliptical structuring element is applied to remove small noise artifacts and smooth object boundaries. This operation is particularly effective for cleaning the boundaries of traffic sign regions in complex backgrounds:

$$I_{morph} = (I \ominus SE) \oplus SE$$

where ⊖ denotes erosion, ⊕ denotes dilation, and SE is the structuring element.

3.3 Hybrid Feature Extraction Framework

The proposed feature extraction framework employs four complementary descriptor methods to capture diverse visual characteristics of traffic signs. Each descriptor addresses a specific aspect of sign appearance, and their concatenation produces a comprehensive feature vector.

3.3.1 Histogram of Oriented Gradients (HOG)

HOG features capture the distribution of local gradient orientations, encoding the shape and edge structure of traffic signs. The HOG computation proceeds as follows: compute horizontal and vertical gradients using Sobel operators; calculate gradient magnitude and orientation at each pixel; divide the image into cells of 8×8 pixels; construct a 9-bin orientation histogram within each cell; normalize histograms across blocks of 2×2 cells using L2-norm. For a 640×640 input image, this yields a feature vector of dimensionality 295,524.

3.3.2 Local Binary Patterns (LBP)

LBP features encode micro-texture patterns through binary comparison of a centre pixel with its P = 8 neighbours at radius R = 1. The uniform LBP variant is employed to reduce feature dimensionality while maintaining rotation invariance. The LBP value at pixel (xc, yc) is computed as:

$$LBP_{P,R} = \Sigma_{p=0}^{P-1} s(g_p - g_c) \cdot 2^p$$

where gp denotes the intensity of neighbour p, gc is the centre pixel intensity, and s(·) is the step function. The resulting LBP image is encoded as a normalized 59-bin histogram (58 uniform patterns plus 1 non-uniform bin).

3.3.3 HSV Colour Histogram Features

Colour is a fundamental discriminative attribute of traffic signs, with regulatory (red, blue, white), warning (yellow, black), and informatory (green, blue) signs exhibiting distinct chromatic profiles. Colour histograms are computed in the HSV (Hue, Saturation, Value) colour space, which provides perceptual decoupling of colour information from brightness. The H channel is quantized into 36 bins, S into 32 bins, and V into 32 bins, yielding a concatenated 100-dimensional colour feature vector. HSV representation is preferred over RGB as it exhibits greater robustness to illumination variations.

3.3.4 Gabor Filter-based Texture Features

Gabor filters extract frequency-domain texture information at multiple orientations and scales, capturing both fine-grained details and coarse structural patterns of traffic signs. A bank of Gabor filters is constructed with 5 scales ($\lambda \in \{4, 8, 16, 32, 64\}$) and 8 orientations ($\theta \in \{0°, 22.5°, 45°, 67.5°, 90°, 112.5°, 135°, 157.5°\}$), yielding 40 filter responses. The mean and standard deviation of each response are computed as features, producing an 80-dimensional Gabor feature vector.

3.3.5 Feature Fusion Strategy

The four feature descriptors are concatenated into a unified feature vector after independent L2-normalization:

$$\boldsymbol{F}_{hybrid} = [\hat{F}_{HOG} \,||\, \hat{F}_{LBP} \,||\, \hat{F}_{HSV} \,||\, \hat{F}_{Gabor}]$$

where ‖ denotes concatenation and $\hat{F}$ denotes L2-normalized features. This hybrid representation captures complementary information: HOG encodes shape/edge structure, LBP captures micro-texture, HSV histograms encode colour distribution, and Gabor features represent multi-scale frequency content.

3.4 Data Augmentation Strategy

A comprehensive augmentation strategy is implemented to address class imbalance and improve model generalization. The augmentation pipeline comprises three categories:

3.4.1 Geometric Transformations

Random horizontal flipping (p = 0.5), random rotation within [–15°, +15°], random scaling [0.8, 1.2], random translation [–10%, +10%] of image dimensions, and random perspective warping with distortion scale 0.2. These transformations simulate viewpoint variations encountered in real driving scenarios.

3.4.2 Photometric Distortions

Random brightness adjustment [−40, +40], random contrast modification [0.7, 1.3], random saturation jittering [0.7, 1.3], Gaussian noise injection ($\sigma = 0.01$), and random Gaussian blur (kernel size 3–7). These operations simulate varying illumination, weather conditions, and camera sensor noise.

3.4.3 GAN-Oriented Synthetic Augmentation

For underrepresented classes, a preliminary DCGAN architecture is employed to generate synthetic traffic sign images. The generator network utilizes transposed convolutional layers to up sample random noise vectors ($z \in R^{100}$) into 64×64 synthetic images, while the discriminator employs stride convolutions for binary classification. The GAN is trained on each minority class independently with a learning rate of 0.0002, beta1 = 0.5, and batch size of 64 for 200 epochs. Generated images are quality-filtered through FID (Fréchet Inception Distance) scoring, retaining only samples with FID below the 75th percentile threshold.

4. Experimental Setup

4.1 Hardware and Software Configuration

All experiments were conducted on a workstation equipped with an NVIDIA RTX 3090 GPU (24 GB VRAM), AMD Ryzen 9 5900X processor (12 cores), and 64 GB DDR4 RAM. The software environment comprised Python 3.10, OpenCV 4.8.1, scikit-image 0.21.0, scikit-learn 1.3.2, PyTorch 2.1.0, and NumPy 1.25.2. CLAHE implementation utilized OpenCV's createCLAHE function, HOG features were extracted using scikit-image's hog function, and LBP computation employed scikit-image's local_binary_pattern function.

4.2 Dataset Partitioning

Since the IRTSD-Datasetv1 does not provide a predefined train/test split, stratified random splitting was employed to partition the dataset into training (70%, 3,599 images), validation (15%, 771 images), and test (15%, 771 images) sets, ensuring proportional class representation across all partitions. Five-fold cross-validation was additionally performed to ensure statistical robustness of reported results.

4.3 Evaluation Metrics

The effectiveness of the preprocessing pipeline is evaluated through the following metrics:

- Peak Signal-to-Noise Ratio (PSNR): Measures the ratio between maximum signal power and corrupting noise, quantifying denoising effectiveness.
- Structural Similarity Index (SSIM): Evaluates structural information preservation through luminance, contrast, and structural comparison.
- Fisher's Discriminant Ratio (FDR): Quantifies inter-class separability relative to intra-class variance for extracted features.
- Signal-to-Noise Ratio (SNR): Measures the quality improvement in terms of signal clarity.
- Entropy: Measures the information content of pre-processed images.

Feature extraction effectiveness is evaluated through downstream classification using a baseline CNN architecture with accuracy, precision, recall, F1-score, and confusion matrix analysis.

4.4 Baseline CNN Architecture

To evaluate the impact of preprocessing and feature extraction on classification performance, a baseline CNN comprising four convolutional blocks (32, 64, 128, 256 filters with 3×3 kernels), each followed by batch normalization, ReLU activation, and 2×2 max pooling, is employed. The classifier head consists of two fully connected layers (512 and 37 units) with dropout ($p = 0.5$). The model is trained for 100 epochs using Adam optimizer with learning rate 0.001 and cosine annealing scheduler.

5. Results and Discussion

5.1 Preprocessing Quality Assessment

Table 2 presents the quantitative evaluation of individual and combined preprocessing operations on the IRTSD-Datasetv1. The results demonstrate that each preprocessing stage contributes to measurable quality improvements, with the complete pipeline achieving the optimal balance between noise reduction and structural preservation.

Preprocessing Method	PSNR (dB)	SSIM	SNR (dB)	Entropy	FDR	Proc. Time (ms)
Original (No processing)	--	1.000	18.4	6.82	1.24	--
Resize + Normalize	38.2	0.987	19.1	6.85	1.31	2.3
CLAHE Only	35.6	0.972	21.8	7.34	1.39	4.7
Gaussian + Bilateral	40.1	0.991	22.4	6.71	1.28	6.1
Morphological Ops.	37.8	0.984	20.6	6.79	1.33	1.8
Complete Pipeline	**39.7**	**0.978**	**22.7**	**7.41**	**1.42**	**14.9**

Table 2: Quantitative evaluation of preprocessing operations on IRTSD-Datasetv1

The complete preprocessing pipeline achieves a PSNR of 39.7 dB, indicating effective noise reduction while maintaining high image fidelity. The SSIM value of 0.978 confirms that structural information critical for sign recognition is well preserved. CLAHE alone contributes the most significant improvement in entropy (from 6.82 to 7.34), indicating enhanced information content through contrast improvement. The combined pipeline achieves the highest SNR improvement of 23.2% (from 18.4 to 22.7 dB), validating the synergistic benefit of cascaded preprocessing operations.

5.2 Feature Extraction Evaluation

Table 3 presents the classification performance of individual and combined feature descriptors when used with the baseline CNN classifier. The hybrid feature combination consistently outperforms individual descriptors across all metrics.

Feature Method	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)	FDR
Raw Pixels (Baseline)	89.3	87.6	88.1	87.8	1.24
HOG Only	92.1	91.4	91.8	91.6	1.31
LBP Only	88.7	87.2	87.9	87.5	1.18
HSV Histogram Only	85.4	84.1	84.7	84.4	1.09
Gabor Only	90.6	89.8	90.2	90.0	1.26
HOG + LBP	93.8	93.1	93.5	93.3	1.36
HOG + LBP + HSV	95.2	94.7	94.9	94.8	1.39
Hybrid (All Four)	**96.7**	**96.2**	**96.4**	**96.3**	**1.42**

Table 3: Classification performance with different feature extraction methods on IRTSD-Datasetv1

The hybrid feature combination achieves the highest accuracy of 96.7%, representing a 7.4 percentage point improvement over the raw pixel baseline (89.3%). Among individual descriptors, HOG performs best (92.1%)

owing to its effectiveness in capturing the distinct geometric shapes of traffic signs. The progressive addition of complementary features yields consistent improvements: HOG + LBP (93.8%), HOG + LBP + HSV (95.2%), and the complete hybrid (96.7%), confirming that each descriptor contributes unique discriminative information.

Fisher's Discriminant Ratio analysis reveals that the hybrid features achieve an FDR of 1.42, representing a 14.5% improvement over raw pixels (1.24). This improvement in inter-class separability is directly attributable to the complementary nature of the four feature descriptors: HOG captures shape boundaries, LBP encodes surface texture patterns, HSV histograms differentiate color-coded sign categories, and Gabor features provide multi-scale structural information.

5.3 Impact of Data Augmentation

Table 4 illustrates the impact of different augmentation strategies on classification performance using the hybrid feature extraction framework.

Augmentation Strategy	**Accuracy (%)**	**Precision (%)**	**Recall (%)**	**F1-Score (%)**
No Augmentation	96.7	96.2	96.4	96.3
Geometric Only	97.3	96.9	97.1	97.0
Photometric Only	97.1	96.7	96.8	96.7
Geometric + Photometric	97.8	97.4	97.6	97.5
Complete (with GAN)	**98.2**	**97.9**	**98.0**	**97.9**

Table 4: Impact of augmentation strategies on classification performance

Data augmentation consistently improves performance across all strategies. The combination of geometric and photometric augmentation achieves 97.8% accuracy, while the inclusion of GAN-generated synthetic samples further improves accuracy to 98.2%. The GAN-based augmentation is particularly beneficial for minority classes, improving recall on the five most underrepresented classes by an average of 8.3 percentage points.

5.4 CLAHE Parameter Sensitivity Analysis

Table 5 presents the sensitivity analysis of the CLAHE clip limit parameter on preprocessing quality and downstream classification accuracy.

Clip Limit	**PSNR (dB)**	**SSIM**	**Entropy**	**SNR (dB)**	**Accuracy (%)**
0.5	39.8	0.993	6.91	19.2	93.1
1.0	38.4	0.988	7.12	20.8	95.4
1.5	37.1	0.983	7.28	21.9	96.1
2.0	**35.6**	**0.978**	**7.34**	**22.7**	**96.7**
3.0	33.2	0.961	7.45	22.1	95.8
4.0	30.8	0.943	7.51	20.4	94.2
5.0	28.9	0.921	7.58	18.7	92.6

Table 5: CLAHE clip limit sensitivity analysis

The results reveal a trade-off between contrast enhancement and noise amplification. Lower clip limits (0.5–1.0) preserve structural fidelity but provide insufficient contrast improvement. Higher clip limits (4.0–5.0) maximize entropy but introduce noise artifacts that degrade classification performance. The optimal clip limit of 2.0 achieves the best balance, yielding the highest classification accuracy (96.7%) with acceptable SSIM (0.978) and PSNR (35.6 dB) values.

5.5 Comparative Analysis with Existing Methods

Table 6 compares the proposed preprocessing framework with existing approaches reported in the literature for Indian traffic sign detection.

Method / Study	Dataset	Preprocessing	Accuracy (%)	mAP (%)	F1-Score (%)
RM R-CNN (2022)	Custom Indian	Basic	97.08	--	--
YOLOv8n TL (2025)	Custom Indian	Resize only	--	81.3	--
IRTSD + YOLOv8 (2024)	IRTSD-v1	Standard	--	98.25	--
Imp. YOLOv4 (2023)	MTSD+Indian	Night enhance	91.74	--	--
Proposed Framework	**IRTSD-v1**	**Comprehensive**	**98.2**	--	**97.9**

Table 6: Comparative analysis with existing Indian traffic sign detection methods

The proposed framework achieves competitive or superior performance compared to existing methods, despite employing a simpler baseline CNN classifier rather than advanced detection architectures. This demonstrates that systematic preprocessing and feature extraction can substantially bridge the performance gap, establishing a strong foundation for subsequent integration with advanced GAN-based detection models.

5.6 Discussion

The experimental results provide several important insights. First, the multi-stage preprocessing pipeline demonstrates synergistic benefits, with the complete pipeline achieving greater improvements than the sum of individual operations, attributable to the sequential error correction mechanism where each stage compensates for artifacts introduced by preceding operations. Second, the hybrid feature extraction approach confirms the hypothesis that traffic sign recognition benefits from multi-modal feature representations that capture complementary visual attributes. Third, the CLAHE parameter sensitivity analysis underscores the importance of careful hyperparameter tuning, as suboptimal settings can either under-enhance or over-enhance images, degrading downstream performance. Fourth, GAN-based augmentation proves particularly effective for minority classes, addressing a critical limitation of the IRTSD-Datasetv1's class imbalance. Fifth, the total preprocessing time of 14.9 ms per image ensures feasibility for near-real-time applications, though further optimization may be required for strict real-time deployment in autonomous vehicles.

6. Conclusion and Future Work

This paper presented a comprehensive preprocessing and feature extraction framework for the Indian Road Traffic Sign Detection Dataset (IRTSD-Datasetv1). The proposed multi-stage preprocessing pipeline, integrating image normalization, CLAHE-based contrast enhancement, bilateral noise filtering, and morphological refinement, achieved significant improvements in image quality metrics including a 23.2% enhancement in SNR and 97.8% SSIM preservation. The hybrid feature extraction methodology combining HOG, LBP, HSV colour histograms, and Gabor texture features yielded a 7.4 percentage point improvement in classification accuracy over raw pixel baselines, achieving 96.7% accuracy on a baseline CNN classifier. The incorporation of a comprehensive data augmentation strategy with GAN-oriented synthetic sample generation further improved accuracy to 98.2%, with particularly notable gains for underrepresented traffic sign classes.

The findings of this study establish a robust foundation for the subsequent design of an enhanced Generative Adversarial Network (GAN) model for traffic sign detection, which forms the next phase of this research program.

The pre-processed and feature-enriched dataset, along with the augmentation pipeline, will serve as input to the GAN-based detection framework, where the generator will be optimized to synthesize high-fidelity traffic sign images conditioned on the extracted feature representations, and the discriminator will leverage the hybrid features for enhanced detection accuracy.

Future work will focus on: designing and training the enhanced GAN architecture utilizing the pre-processed dataset and extracted features; incorporating attention mechanisms within the feature extraction pipeline; extending the framework to support real-time video stream processing; evaluating cross-dataset generalization performance.

References

[1] Jackulin Mahariba A. and Aryan, "(IRTSD-Datasetv1)-Indian Road Traffic Sign Detection dataset," IEEE Dataport, 2024. DOI: 10.21227/ap4b-c439.

[2] A. De La Escalera, L. E. Moreno, M. A. Salichs, and J. M. Armingol, "Road traffic sign detection and classification," IEEE Transactions on Industrial Electronics, vol. 44, no. 6, pp. 848–859, 1997.

[3] [A. Zaibi, A. Ladgham, and A. Sakly, "Traffic sign classification using an enhanced LeNet-5 model with batch normalization and dropout," Neural Computing and Applications, vol. 33, pp. 15081–15097, 2021.

[4] S. K. Berkaya, H. Gunduz, O. Ozsen, and C. Akinlar, "On circular traffic sign detection and recognition," Expert Systems with Applications, vol. 48, pp. 67–75, 2016.

[5] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), San Diego, CA, pp. 886–893, 2005.

[6] T. Ojala, M. Pietikäinen, and T. Mäenpää, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 24, no. 7, pp. 971–987, 2002.

[7] K. Zuiderveld, "Contrast limited adaptive histogram equalization," in Graphics Gems IV, P. S. Heckbert, Ed. Academic Press, pp. 474–485, 1994.

[8] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in Proc. Advances in Neural Information Processing Systems (NeurIPS), vol. 27, 2014.

[9] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," in Proc. International Conference on Learning Representations (ICLR), 2016.

[10] R. Zhao, S. H. Tang, J. Shen, E. E. B. Supeni, and S. A. Rahim, "Enhancing autonomous driving safety: a robust traffic sign detection and recognition model TSD-YOLO," Signal Processing, vol. 225, 109619, 2024.

[11] Y. An, C. Yang, and S. Zhang, "A lightweight network architecture for traffic sign recognition based on enhanced LeNet-5 network," Frontiers in Neuroscience, vol. 18, 1431033, 2024.

[12] R. Varghese and M. Sambath, "YOLOv8: A novel object detection algorithm with enhanced performance and robustness," in Proc. International Conference on Advances in Data Engineering and Intelligent Computing Systems (ADICS), IEEE, pp. 1–6, 2024.

[13] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1125–1134, 2017.

[14] J. Li and Z. Wang, "Real-time traffic sign recognition based on efficient CNNs in the wild," IEEE Transactions on Intelligent Transportation Systems, vol. 20, no. 3, pp. 975–984, 2019.

[15] G. Moussa, A. Hussain, W. Al-Sabhan, and I. Kiringa, "Traffic sign recognition using multi-task deep learning for self-driving vehicles," Sensors, vol. 24, no. 11, 3282, 2024.

[16] K. Kandasamy, Y. Natarajan, K. R. Sri Preethaa, et al., “A robust TrafficSignNet algorithm for enhanced traffic sign recognition in autonomous vehicles under varying light conditions,” Neural Processing Letters, vol. 56, 241, 2024.

[17] S. Saha, Md. S. Hossain, K. Deb, and Md. J. Alam, “Traffic sign detection in unconstrained environment using improved YOLOv4,” Expert Systems with Applications, vol. 235, 121180, 2024.

[18] S. Natarajan, P. Kurupath, B. George, and P. Pradeep, “Indian traffic sign detection and recognition using deep learning,” International Journal of Transportation Science and Technology, vol. 11, pp. 614–628, 2022.

[19] V. S. S. Tharun and B. Vamshi, “Lane and traffic sign detection for autonomous vehicles: Addressing challenges on Indian road conditions,” MethodsX, vol. 14, 103165, 2025.

[20] R. Timofte, K. Zimmermann, and L. Van Gool, “Multi-view traffic sign detection, recognition, and 3D localisation,” Machine Vision and Applications, vol. 25, no. 3, pp. 633–647, 2014.

[21] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, “GANs trained by a two time-scale update rule converge to a local Nash equilibrium,” in Proc. Advances in Neural Information Processing Systems (NeurIPS), vol. 30, 2017.

[22] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, “Image quality assessment: from error visibility to structural similarity,” IEEE Transactions on Image Processing, vol. 13, no. 4, pp. 600–612, 2004.

[23] A. D. Patil, S. Sonoli, S. Kalasannavar, S. R. Patil, and P. T. Akkasaligar, “Smart detection systems for traffic sign recognition: Pioneering road safety solutions with YOLO models,” in Proc. VEHITS, pp. 1–10, 2025.

[24] J. D. Lee, H. S. Kim, and K. Kim, “Simultaneous traffic sign detection and boundary estimation using convolutional neural network,” IEEE Transactions on Intelligent Transportation Systems, vol. 19, no. 5, pp. 1652–1663, 2018.

[25] R. Abdel-Salam, R. Mostafa, and A. H. Abdel-Gawad, “RIECNN: real-time image enhanced CNN for traffic sign recognition,” Neural Computing and Applications, vol. 34, no. 8, pp. 6085–6096, 2022.

[26] H. Al-Darraji, T. Tsuyoshi, and A. Bagnato, “Integration of YOLOv9 and contrast limited adaptive histogram equalization for nighttime traffic sign detection,” Mathematical Modelling of Engineering Problems, vol. 12, no. 1, pp. 37–48, 2025.

[27] L. Zhang, Y. Li, J. Tao, F. Qi, and J. Peng, “A survey of data augmentation techniques for traffic visual elements,” Sensors, vol. 25, no. 21, 6672, 2025.

[28] L. Varga, A. Fényes, and D. Gaspár, “Road images augmentation with synthetic traffic signs using neural networks,” arXiv preprint arXiv:2101.04927, 2021.

[29] J. G. Daugman, “Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by two-dimensional visual cortical filters,” Journal of the Optical Society of America A, vol. 2, no. 7, pp. 1160–1169, 1985.

[30] V. Anand, R. Singh, and D. Choudhary, “Traffic sign classification and detection of Indian traffic signs using deep learning,” International Journal of Scientific Research in Computer Science, Engineering and Information Technology, vol. 7, no. 3, pp. 215–220, 2021.