Traffic Sign Classification Under Varying Lighting Conditions in the Philippines Using Transfer Learning with ResNet50 and Zero-DCE

Traffic Sign Classification Under Varying Lighting Conditions in the Philippines

Dr. John Paul Q. Tomas, Carlo Miguel P. Legaspi, Karl Anthony S. Dalangin, Gabriel Paul Q. Lim
Mapúa University School of Information Technology, Philippines

***Abstract*—This study presents a multi-stage transfer learning approach for improving traffic sign recognition performance under both normal and low-light conditions, addressing the gap between existing datasets and the real-world road environments of the Philippines, where poor lighting, faded signs, and unstructured roads are common. A curated local dataset of 7 commonly encountered traffic sign classes comprising approximately 5,000 manually localized images was constructed and split into training, validation, and test sets (70–10–20 ratio). Five model configurations were developed and compared: a VGG-inspired baseline trained from scratch, a standard ResNet50 transfer learning model, a multiphase ResNet50 model pretrained on the GTSRB dataset, and two corresponding variants enhanced using Zero-DCE low-light preprocessing. The baseline achieved 92.17% accuracy, while the standard ResNet50 models performed similarly with and without Zero-DCE (92.10–92.45%). The multiphase ResNet50 significantly improved accuracy to 96.43% by leveraging domain-aligned pretraining, and the highest performance was achieved by its Zero-DCE-enhanced counterpart at 98.21%, showing more balanced metrics and improved recognition stability. These results indicate that low-light enhancement alone does not guarantee better performance, but becomes highly effective when paired with a feature extractor already specialized in traffic sign features. Overall, the proposed multiphase, Zero-DCE–assisted pipeline provides a strong and scalable solution for traffic sign recognition in low-visibility Philippine conditions, with potential applications in ADAS and autonomous driving systems.**

Keywords*—*Traffic Sign Recognition (TSR); Traffic Sign Classification (TSC); Advanced Driver-Assistance System (ADAS)

I. Introduction

Computer vision has become a core component of modern smart systems, powering a wide range of real-world applications from facial recognition and autonomous driving to medical imaging and industrial automation [1][2]. In the context of intelligent transportation systems, one crucial application of computer vision is traffic sign classification. This task plays a vital role in enabling vehicles, especially those equipped with Advanced Driver-Assistance Systems (ADAS) or fully autonomous capabilities, to understand and respond to road environments effectively [3].

While significant progress has been made in this field, many existing models struggle when exposed to challenging real-world conditions [4]. Many studies in this field rely heavily on popular datasets such as the German Traffic Sign Recognition Benchmark (GTSRB), BDD100K, or TT100K [5][6][7]. While these datasets are useful for benchmarking, they are often limited in terms of lighting diversity and geographic context. When models trained on these datasets are applied to other domains like local roads in different countries or low-light conditions, they often experience a notable drop in accuracy and reliability. This domain shift is not just a possibility; it is a known issue that poses a serious challenge for deploying these systems in real-world settings [8].

To address this gap, deep learning techniques, particularly transfer learning combined with low-light image (LLI) enhancement methods, are to be employed on a dataset gathered in chosen cities in NCR, Philippines. These approaches allow models to adapt better to difficult visual conditions and perform more consistently across varied scenarios, including both daytime and nighttime settings. Incorporating diverse datasets in both training and evaluation further strengthens the robustness of classification models. This study aims to improve traffic sign classification under varying conditions by leveraging transfer learning and an appropriate low-light image preprocessing technique. Unlike object detection tasks that focus on locating signs within an image, this research focuses purely on the classification of already-localized traffic signs. The study will make use of a locally gathered dataset from the Philippines, reflecting the real-world conditions of the country's roads, particularly in low-visibility scenarios.

A. Background of the Study

Traffic Sign Recognition (TSR) systems are designed to interpret road signs from visual data captured by vehicle-mounted cameras. The general TSR pipeline typically involves four stages: image acquisition, where real-time visual data is collected; preprocessing, which enhances image quality and normalizes input formats; detection, where the system identifies the location of potential traffic signs; and classification, where each sign is categorized based on its type and meaning. Among these stages, classification plays a key role in ensuring that signs are correctly understood and acted upon.

Traditionally, TSR systems relied on handcrafted features and conventional computer vision techniques such as edge detection, color thresholding, and template matching [9][10]. However, these methods often failed to handle the complexity and variability of real-world road environments. With the rise of

deep learning, particularly Convolutional Neural Networks (CNNs), traffic sign classification has significantly improved. CNNs can automatically learn rich, hierarchical features from image data, making them more robust to variations in scale, orientation, and background clutter compared to older methods.

Most TSR research to date has been based on datasets such as the German Traffic Sign Recognition Benchmark (GTSRB), Berkeley Deep Drive (BDD100K), and Tsinghua Tencent (TT100K). These datasets are widely used for benchmarking but were collected primarily in developed countries with consistent infrastructure, clear signage, and favorable lighting conditions [11]. As a result, they are limited in their representation of diverse and challenging environments like those found in the Philippines. Using models trained on these datasets in a different setting, especially one with frequent low-light scenarios and less-maintained traffic signage, can lead to a noticeable drop in performance.

Lighting is a critical factor in visual recognition tasks. Poor lighting conditions reduce image clarity, diminish contrast, and obscure key visual features, making classification more difficult. Environmental factors such as streetlight glare, shadows, and low-contrast backgrounds further contribute to the problem. These conditions introduce noise that directly impacts a model's ability to extract meaningful features [12]. One known approach to tackle this is through Low-Light Image (LLI) preprocessing techniques, which enhance visual quality by adjusting brightness, contrast, and detail levels before feeding the images into a classification model.

In the context of the Philippines, several challenges make traffic sign classification even more difficult. There is currently no standardized, publicly available local dataset of Philippine traffic signs, let alone one that includes low-light scenarios. Road signs across cities may vary in shape, condition, and placement, adding another layer of complexity. A model trained solely on a dataset like GTSRB is unlikely to generalize well in this environment due to the domain differences and absence of relevant lighting conditions in its training data.

Given the lack of a large, high-quality local dataset, this study proposes the use of transfer learning as a practical solution. By leveraging a pretrained model such as ResNet50 and fine-tuning it on a custom dataset collected from selected areas in Metro Manila, the model can adapt to the unique characteristics of local traffic signs. Combining this approach with an effective LLI preprocessing technique has the potential to significantly improve classification accuracy in real-world varying conditions found across Philippine roads.

B. Research Gap

Despite recent advancements in deep learning, most traffic sign classification models are still optimized for well-lit, structured, and non-local environments [5][6][7]. Many of these models assume consistent visibility, standardized signage, and ideal conditions or assumptions that do not always reflect real-world driving environments, especially in developing countries like the Philippines.

High-performing models trained on datasets such as the German Traffic Sign Recognition Benchmark (GTSRB) have demonstrated impressive performance under controlled conditions [5]. However, these datasets are primarily composed of high-resolution images captured in structured European settings. As a result, the trained models often struggle to generalize when applied to different geographic and environmental contexts. This includes unstructured roads, poorly-maintained signage, and particularly, low-light conditions scenarios that are frequently encountered in many parts of the Philippines [13].

The base paper referenced in this study, "Traffic Sign Detection and Recognition using Deep Learning", utilizes the VGG architecture and achieves high accuracy on the GTSRB dataset [5]. While the results highlight the potential of deep learning in traffic sign recognition, the paper does not address domain adaptation or performance in low-visibility and geographically distinct environments. The study's effectiveness is therefore limited to the dataset and conditions it was designed for.

In the Philippine context, the challenges are even more pronounced. Road infrastructure varies significantly across regions, and low-light environments such as poorly lit highways, inner-city streets, and rural areas are common [13]. In addition, there is no publicly available, large-scale traffic sign dataset that captures the unique characteristics of Philippine Road signs, especially under low-light scenarios. Without a localized and lighting-aware approach, foreign-trained models are likely to suffer from domain shift and underperform in this setting.

Given these limitations, there is a clear need to:

- Evaluate the performance of a baseline deep learning model, such as VGG, trained on well-lit images, when applied to Philippine traffic signs captured in varying conditions.
- Propose a model architecture that addresses these limitations by incorporating transfer learning (e.g., ResNet50) and low-light image enhancement techniques.
- Provide empirical results that support the development of traffic sign classifiers that are context-aware and resilient to lighting variability, with a specific focus on Philippine roads.

C. Research Questions

This study seeks to investigate how deep learning-based traffic sign classification models can be improved for the varying lighting in real-world scenarios in the Philippines. With the integration of low-light image enhancement and transfer learning, the research aims to assess whether these techniques can bridge the performance gap seen when models trained in ideal conditions are applied to challenging environments. Specifically, the study focuses on evaluating the performance of a novel pipeline using Zero-DCE for image enhancement and multiphase trained ResNet50 for classification.

- To what extent does the integration of Zero-DCE as a low-light enhancement technique improve traffic sign classification accuracy compared to traditional preprocessing methods?

- How significantly does the full pipeline of Zero-DCE for enhancement and Multiphase ResNet50 for classification improve overall model performance and robustness under varying conditions, relative to a baseline model trained with unenhanced data?

The answers to these questions will help determine whether a lighting-aware and transfer learning-based pipeline can serve as a more practical and reliable solution for traffic sign classification in varying Philippine road environments.

D. Research Objectives

To address the research questions, this study sets out two main objectives. These objectives align with the broader aim of building a more adaptable and accurate traffic sign classification model that is suitable for the lighting and environmental conditions of the Philippines. Through comparative evaluation and experimental testing, the study aims to understand the strengths and limitations of using Zero-DCE and ResNet50 in the classification pipeline.

- To evaluate the impact of Zero-DCE on image quality and its effect on classification accuracy relative to traditional preprocessing techniques.
- To develop and assess a Zero-DCE + Multiphase ResNet50 pipeline, and determine its effectiveness compared to the baseline VGG model approach.

By achieving these objectives, the study aims to contribute valuable insights into the development of lighting-resilient and context-aware classification systems for traffic signs in developing countries.

E. Significance of the Study

This study contributes to the field of computing by addressing the technical challenge of traffic sign classification under varying visibility conditions, specifically between daytime and nighttime, using a deep learning-based approach. It explores the integration of image enhancement (Zero-DCE) with a pretrained convolutional neural network (ResNet50), providing a practical workflow for improving classification accuracy in inconsistent visual environments. The research offers value to computer vision and machine learning practitioners by demonstrating how preprocessing techniques can be optimized to enhance the performance of classification models without requiring large-scale data collection or computationally intensive retraining. From an academic perspective, this study highlights the limitations of applying generic, foreign-trained models to local contexts and emphasizes the importance of building datasets that reflect the visual characteristics of Philippine road environments. It also adds to the growing body of work on localized AI solutions, serving as a reference for future research in intelligent transportation systems, image processing under varying conditions, and domain adaptation within computer vision.

F. Scope and Delimitations

This study focuses solely on the task of traffic sign classification under varying visibility conditions, including such as those encountered during daytime and nighttime or in poorly lit areas, with specific attention to real-world scenarios in the Philippines. It aims to enhance classification accuracy using a deep learning-based approach that combines transfer learning and low-light image enhancement. The study does not cover traffic sign detection tasks, such as identifying the location of signs within an image, and instead assumes that signs are already localized prior to classification. In this experimental setup, signs were manually localized and tightly cropped to the sign's boundaries, ensuring the input images focus exclusively on the sign's features to maintain consistency with the GTSRB format. Furthermore, the research does not attempt to implement the model in real-time systems or integrate it with vehicle sensors or autonomous driving platforms. The goal is to conduct a thorough evaluation of the proposed approach in a research context rather than to develop a fully deployable system. Data collection was specifically conducted within cities in Metro Manila. However, it is important to note that because the dataset is geographically constrained to this urban region, variations in local traffic sign styles, environmental factors, and regional lighting characteristics may limit the generalizability of the proposed approach to rural or significantly different geographical areas outside the scope of this study.

II. Review of Related Literature

A. Comparative Overview of CNN Architectures for Image Classification

Deep learning has revolutionized the field of image classification, with Convolutional Neural Networks (CNNs) playing a central role in this progress [14]. Over the years, numerous CNN architectures have been developed and tested across a variety of domains. Two of the most widely recognized models in this space are the VGG family (VGG16 and VGG19) and ResNet50. These architectures have become foundational in many computer vision tasks, offering reliable performance and serving as benchmarks in both research and applied settings [15].

VGG networks, introduced by the Visual Geometry Group at the University of Oxford, gained popularity due to their straightforward and uniform architecture, consisting of sequential convolutional layers followed by fully connected layers [16]. VGG16 and VGG19, named after their depth, are known for their simplicity and effectiveness, particularly on well-structured datasets like ImageNet. However, one of their major limitations is the large number of parameters, which makes them computationally expensive and prone to overfitting when used on smaller or less diverse datasets [17]. ResNet50, introduced by Microsoft Research, brought a major breakthrough by addressing the degradation problem in deep neural networks [18]. Unlike VGG, ResNet uses skip connections, or residual connections, that allow gradients to flow more easily during backpropagation, enabling the successful training of much deeper networks without performance degradation. This architectural innovation allows ResNet models to generalize better and perform more robustly across varied and challenging datasets [19].

A comparative study that evaluated VGG16, VGG19, and ResNet50 on a real-world product classification task involving a dataset from a regional retailer offers useful insights. Under identical training setups using the Keras API with Softmax

activation and categorical cross-entropy loss, all three models achieved high accuracy levels. The training was conducted over 20 epochs with a batch size of 32. While all models performed well, the best results were observed between epochs 15 and 20, after which overfitting started to occur. By epoch 20, VGG16 reached a test accuracy of 96.67%, VGG19 slightly improved on this with 97.07%, and ResNet50 achieved the highest accuracy of 97.33%. The study concluded that ResNet50 not only outperformed the VGG models but also showed better generalization, particularly in real-world classification scenarios [20].

These findings are consistent with broader literature, which often highlights ResNet50's improved performance in terms of both accuracy and robustness, especially when working with datasets that exhibit variation in lighting, background, or object orientation [19]. Given its ability to learn deeper representations without overfitting, ResNet50 is well-suited for more complex or context-sensitive image classification tasks, such as traffic sign recognition in low-light environments, which is the focus of this study.

B. Foundational Study and Transfer Learning Using ResNet Architectures for Traffic Sign Recognition

To establish a meaningful comparison for this study, it is essential to reference a strong baseline model that demonstrates reliable performance in traffic sign classification. A high-performing work provides the foundation for evaluating architectural improvements, domain adaptation strategies, and performance shifts under challenging conditions. For this purpose, a well-recognized study utilizing a VGG-inspired Convolutional Neural Network serves as the primary baseline, offering a solid benchmark for accuracy, training methodology, and dataset usage.

One notable study adopted a VGG-based CNN framework trained on the German Traffic Sign Recognition Benchmark (GTSRB), a widely used dataset containing over 50,000 labeled images across 43 traffic sign categories. The model followed the characteristic VGG design of stacked convolutional layers with small 3×3 filters, followed by fully connected layers for classification. This architecture allowed the network to learn hierarchical visual features effectively while maintaining interpretability and structural simplicity. Through consistent preprocessing, data augmentation, and careful hyperparameter tuning, the study achieved an accuracy of 97.86% on the GTSRB dataset [5]. This strong performance, combined with the stability of VGG-style architectures, positions the study as an appropriate baseline for comparison, especially for evaluating improvements introduced by deeper or more adaptive models such as ResNet50.

Transfer Learning has been widely recognized as a powerful technique for improving deep learning models by utilizing pretrained architectures [21]. Another notable example other than VGGNet is the use of Residual Networks (ResNet), which allow deep models to efficiently learn feature representations while mitigating training difficulties in very deep networks. Building on this approach, Barodi et al. [22] proposed a traffic sign classification system that leveraged a CNN-based model with Transfer Learning to extract visual features. Their study demonstrated that this method achieved higher accuracy compared to traditional machine learning techniques and some Advanced Driving Assistance Systems (ADAS), highlighting the effectiveness of Transfer Learning in traffic sign recognition tasks.

Another study focusing on the detection and classification of Mexican traffic signs using deep learning involved creating a dataset of 1,284 images containing 1,426 traffic signs from Ciudad Juárez and Monterrey [23]. The study compared the performance of Region-based Convolutional Neural Networks (R-CNN) and You Only Look Once (YOLO v3) for detection, and a modified ResNet50 for classification. Results indicated that the RCNN/ResNet50 combination achieved a mean Average Precision (mAP) of 95.33%, while the YOLO v3/ResNet50 yielded 90.33%. The classification accuracy was 99.00%, demonstrating the effectiveness of using deep learning techniques for traffic sign recognition. Additional tests confirmed the model's robustness even with occluded or randomly positioned signs.

In addition to its core architectural benefits, ResNet also supports advanced training strategies such as multiphase fine-tuning, which enhances transfer learning performance in domains with limited or specialized data. A study by Ko et al. [24] introduced a multiphase fine-tuning method, where network layers are unfrozen and fine-tuned in sequential phases rather than all at once. Originally applied to sign language recognition, this approach allowed better adaptation of pretrained weights to the target task. The authors reported that multiphase fine-tuning not only improved accuracy but also reduced the number of training epochs required. This demonstrates ResNet's flexibility in adapting to various domains and tasks, including scenarios like traffic sign classification, where data distribution may differ from large-scale pretraining datasets.

In a related domain, [25] a study conducted by Rao et al., explored the application of ResNet-based models for traffic light detection and classification, emphasizing the importance of Intelligent Transportation Systems (ITS) in enhancing road safety and traffic management. Their study proposed a novel methodology for real-time traffic light recognition using ResNet, addressing challenges such as adverse weather, varying illumination, and occlusions. The ResNet architecture's deep learning capabilities enabled the system to discern intricate patterns and features, showcasing its adaptability to real-world traffic scenarios. The study highlighted the significance of accurate traffic light detection and classification, not only as a technical necessity but also for optimizing traffic flow, minimizing congestion, and enhancing road safety in urban environments. The project demonstrated the potential of ResNet-based models to improve ITS applications, paving the way for smarter and safer urban mobility systems.

In another study, Jaiswal et al. conducted an empirical analysis of traffic sign recognition using various iterations of the Residual Neural Network (ResNet) architecture [26]. The study utilized the German Traffic Signs Recognition Benchmark (GTSRB) dataset, which contains 43 different classes of traffic signs. The authors evaluated the performance of multiple ResNet architectures, including ResNet50, ResNet50V2, ResNet101, ResNet101V2, ResNet152, and ResNet152V2,

based on metrics such as accuracy, precision, recall, and F1 score. Their results showed that ResNet50V2 achieved the highest accuracy of 99.78%, while ResNet152 achieved the lowest accuracy of 99.49%. The study also highlighted that the accuracy of ResNet50V2 improved with the depth of the architecture. To enhance performance, image preprocessing techniques such as resizing, zero-centering, and normalization were applied to the traffic sign images before feeding them into the ResNet models. The reshaped matrix of the preprocessed images was then used for training. The study demonstrated that the combination of ResNet architectures, image preprocessing, and matrix reshaping significantly improved the performance of traffic sign recognition systems, making them more robust and efficient.

C. Low-Light Image Preprocessing Techniques for Traffic Sign Classification

Image preprocessing is a critical step in any computer vision pipeline, particularly for tasks such as traffic sign classification, where the quality and clarity of input images directly affect model performance [27]. Traditional preprocessing techniques often involve simple, rule-based methods such as resizing, histogram equalization, brightness adjustment, and normalization. These methods are computationally efficient and can help prepare images for neural networks by standardizing input formats and enhancing contrast. However, they often fall short in more complex or degraded visual environments, such as night-time or low-light road conditions, where global adjustments may not be enough to restore image visibility or preserve important features [28].

Given the study's focus on improving classification performance under low-light conditions, traditional methods alone are insufficient [29]. Traffic signs captured in poorly lit scenarios often suffer from reduced contrast, loss of detail, and high visual noise, which are problems that require more adaptive and learned enhancement strategies. Instead of relying solely on traditional preprocessing, this research explores model-based low-light image enhancement techniques capable of learning complex adjustments directly from data. Specifically, the study considers and evaluates two state-of-the-art models: MIRNet and Zero-DCE.

A study by [30] compared the performance of MIRNet and Zero-DCE in enhancing lowlight images using both the LoL dataset and actual CCTV footage. The evaluation focused on two widely used image quality metrics, Peak Signal-to-Noise Ratio (PSNR) and Mean Squared Error (MSE). Zero-DCE slightly outperformed MIRNet in this comparison, achieving a PSNR of 28.055 dB and an MSE of 10.204, compared to MIRNet's PSNR of 27.86 dB and MSE of 10.595. While the difference in metrics was marginal, the study also assessed the qualitative strengths of each model. Unlike MIRNet's heavy multi-scale approach, Zero-DCE operates by learning a set of pixel-wise, high-order transverse curves to iteratively adjust the dynamic range of the input [30]. This curve estimation is achieved through a zero-reference framework, using non-reference loss functions such as spatial consistency and exposure control to enhance images without requiring paired ground-truth data.

Given the classification-oriented nature of this study, where contrast and clarity of traffic sign features are essential for accurate recognition, Zero-DCE was chosen as the preferred preprocessing model. Its core principle of estimating pixel-specific enhancement curves allows it to brighten dark regions while preserving the structural integrity of traffic sign symbols. Its ability to brighten images effectively and reduce noise makes it well-suited for enhancing traffic signs captured in dimly lit urban environments or poorly illuminated roads. Moreover, Zero-DCE's lightweight, end-to-end structure allows it to be easily integrated into the classification pipeline without significant computational overhead.

The Zero-DCE algorithm was implemented using the original version provided in the official GitHub repository [29]. All images from the local dataset were resized to 32×32 pixels before enhancement, and the default parameters specified in the repository were used for curve estimation and adaptive brightness adjustment. No additional modifications were made to the model architecture or optimization settings. This setup ensures that the preprocessing pipeline can be faithfully reproduced by other researchers using the same dataset and official implementation.

This decision aligns with the study's objective of building a robust and context-aware classification system. By selecting a data-driven image enhancement model like Zero-DCE, the research ensures that the visual quality of input images is sufficiently improved to support accurate and reliable traffic sign classification in Philippine low-light conditions. This decision aligns with the study's objective of building a robust and context-aware classification system. By selecting a data-driven image enhancement model like Zero-DCE, the research ensures that the visual quality of input images is sufficiently improved to support accurate and reliable traffic sign classification in Philippine low-light conditions.

D. Recent Discussions in Low-Light Traffic Sign Recognition

Several recent studies were able to address and discuss traffic sign recognition in low-light environments, each with their own deep learning approaches and methods. One study [31] addresses the topic and developed a YOLOv8 algorithm trained on the ZND dataset, a combination of GTSRB, GitHub, and locally obtained images. Other assessments were done on traffic signs, such as retroreflectivity measurements, video data, and human evaluations. Among the results discussed in the study is the comparison of detection rates between daytime and nighttime. Both conditions show reliable performance from the model. However, it is revealed that 10% of traffic signs were missed in nighttime conditions but were detected during the day.

In a different study [32], a traffic sign recognition method is introduced. The image enhancement algorithm uses an adaptive approach of gamma correction. The network uses a single-stage target detection (SSD) with ResNet and VGG as the backbone, and it also includes a feature difference (FD) model. After optimization, the ResNet + FD network obtained an accuracy of 96.33%, while VGG + FD obtained 94.52%.

While the discussed studies provide valuable insights into traffic sign recognition under low-light conditions, this research

introduces distinct methodological steps aimed specifically at the unique environmental challenges of the Philippines, focusing purely on a multi-stage classification pipeline for already-localized signs. This training, combined with Zero-DCE enhancement, addresses the specific domain shift and visibility issues common on local Philippine roads, a comprehensive integration not fully explored in the referenced literature.

E. Conceptual Framework

The training framework for the proposed model, as shown in Fig. 1, adopts a two-phase strategy to improve traffic sign recognition performance under both standard and low-light conditions. In Phase 1 (Feature Learning), the model is trained on the German Traffic Sign Recognition Benchmark (GTSRB) to learn general traffic sign features. The dataset undergoes standard preprocessing steps, including resizing and normalization. A ResNet50 model, initialized with pretrained ImageNet weights, is trained using categorical cross-entropy loss and the Adam optimizer for 20 to 30 epochs, with early stopping applied based on validation loss. After training, the model weights, excluding the final fully connected (FC) classification layer, are saved for transfer learning.

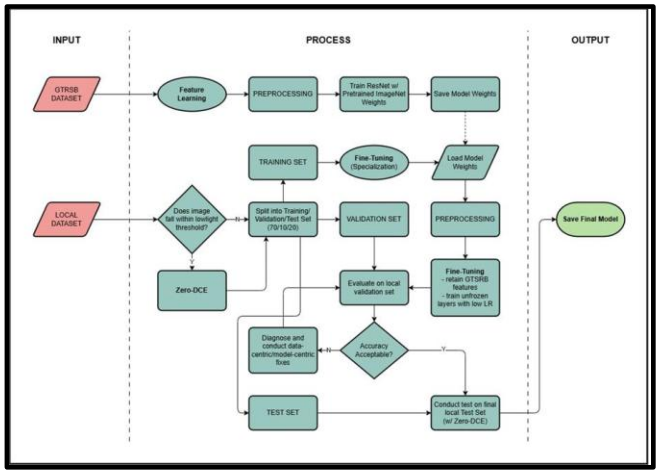

Fig. 1. Conceptual framework.

In Phase 2 (Fine-Tuning), the pretrained model is adapted to a locally collected dataset that mirrors the GTSRB in terms of class distribution and brightness variation. The previously trained weights from Phase 1 are loaded, and the fully connected (FC) layer is replaced to match the number of local traffic sign classes. The local dataset undergoes preprocessing steps, including resizing and normalization consistent with Phase 1 to maintain input compatibility and stable convergence. During fine-tuning, the first 30 to 40 layers of the model are frozen to retain the generalized visual representations learned from GTSRB, while the remaining layers are trained with a lower learning rate to promote adaptation to local features and environmental contexts specific to Philippine traffic signs.

To evaluate robustness under low-light conditions, Zero-DCE (Zero-Reference Deep Curve Estimation) enhancement is applied to the dataset prior to training and evaluation. This ensures that the model's generalization capability is measured against enhanced low-visibility samples. The fine-tuned model's performance is assessed on the local validation and testing sets using accuracy, precision, and F1-score as primary evaluation metrics.

If the evaluation results are unsatisfactory, diagnostic refinement is performed. This may involve data-centric adjustments such as expanding the dataset, improving label quality, or recalibrating the Zero-DCE preprocessing parameters; or model-centric improvements, including unfreezing additional layers, tuning learning rates, extending training epochs, or introducing regularization techniques like dropout or L2 penalties. Through this iterative refinement process, the framework ensures the model effectively generalizes across diverse lighting conditions and maintains high recognition accuracy in real-world scenarios.

III. Methodology

In order to build on existing work, introduce improvements, and objectively compare model performance under varying conditions, the study adopts a structured five-part methodology. Each part corresponds to a distinct model configuration designed to evaluate how architecture depth, pretraining strategy, and low-light enhancement influence traffic sign recognition accuracy. The design of this methodology is incremental, allowing the study to clearly observe how each added component, such as deeper architectures, specialized pretraining, or low-light enhancement, contributes to overall performance. By training and testing each model separately under the same evaluation procedures, the study ensures a fair comparison and a clear view of how performance evolves across models, ultimately identifying which pipeline offers the most reliable results for Philippine traffic sign conditions.

The first part, the Baseline Evaluation Phase, establishes a foundation/baseline using a VGG-inspired Convolutional Neural Network. This model follows the classic VGG structure that stacks multiple 3×3 convolution layers before each pooling layer, allowing it to learn fine-grained visual patterns. It is trained from scratch using the local traffic sign dataset without any external pretraining. Standard augmentation techniques: rotation, zoom, brightness shifts, and noise, are applied to improve generalization. This baseline provides a reference point for assessing the benefit of deeper architectures and more advanced training strategies.

The second part employs a ResNet50 backbone initialized with ImageNet weights. Transfer learning is used by freezing the early layers and retraining the higher layers on the local dataset. The model benefits from deep residual connections that help stabilize training and capture richer feature representations compared to the baseline. After preprocessing and augmentation, the model is trained end-to-end with a classification head tailored to the dataset. This setup represents a conventional transfer learning approach commonly used in modern traffic sign recognition studies.

The third part adopts a two-phase training strategy. In the first phase, the model is trained on the German Traffic Sign Recognition Benchmark (GTSRB) to learn traffic-sign-specific features that go beyond ImageNet's general object knowledge. The weights from this phase are saved and transferred to a second architecture with matching dimensions but a new classification head. In the second phase, the pretrained model is

fine-tuned on the local dataset using a lower learning rate to preserve the learned traffic sign features while adapting to local characteristics. This multiphase setup aims to reduce domain shift and enhance recognition performance in Philippine environments.

The fourth part uses the same transfer learning pipeline as Part 2, but all dataset images undergo Zero-DCE low-light enhancement before training. Zero-DCE adjusts exposure and contrast to improve visual clarity in dim or uneven lighting. By combining enhancement with the standard ResNet50 transfer learning setup, this part assesses whether preprocessing alone can significantly improve model robustness, especially for signs captured in poorly lit conditions.

The fifth part integrates the multiphase training strategy with Zero-DCE enhancement. All GTSRB and local images are processed through Zero-DCE before use. The model first learns enhanced traffic sign features during GTSRB pretraining, then undergoes low-learning-rate fine-tuning on the enhanced local dataset. This pipeline evaluates whether combining targeted feature pretraining with low-light correction offers the strongest resistance to visibility challenges commonly found in Philippine roads.

A. Baseline Evaluation Phase

The baseline evaluation phase aims to replicate and fairly assess the performance of the traffic sign classification model proposed in the foundational study. The primary goal is to understand how the baseline approach, originally trained and tested on the GTSRB dataset, performs when applied to a local Philippine context with varying lighting conditions. To achieve this, the training procedure of the recreated baseline model will strictly follow the methodology prescribed in the base paper, ensuring fidelity to the original architecture and settings. Evaluation, however, will be conducted on a Philippine test set that has been carefully curated to be evenly distributed across different brightness levels, providing a more realistic assessment of the model's performance under conditions not captured in the original GTSRB dataset.

1) GTSRB-based local dataset recreation: Since this study focuses exclusively on the task of classification, traffic signs were manually localized and cropped from raw field images prior to dataset construction. Each sample in the recreated dataset contains a single, centered traffic sign with the background minimized, ensuring the model learns features specific to the sign's geometry and symbols without external interference. No automated detection algorithms were employed for this process; instead, bounding boxes were manually defined to maintain a standardized Region of Interest (ROI) that mirrors the tightly-cropped format of the original GTSRB dataset.

To avoid improper adaptation and biased evaluation that could result from training the baseline model on a foreign dataset (GTSRB) and directly testing it on a local Philippine dataset, a localized, smaller-scale version of GTSRB will be recreated for training purposes. This recreated dataset will focus only on the traffic sign classes that are both present in the GTSRB and commonly found in Metro Manila, reflecting the practical limitations of data collection within the study. By mirroring the original GTSRB in terms of the number of images per selected class and preserving the mean brightness distribution across images, the recreated dataset ensures that the baseline model is trained under conditions comparable to the original study, while still enabling fair evaluation in the Philippine context.

The selection of traffic sign classes for the recreated dataset will be guided by both compatibility with the GTSRB dataset and relevance to the Philippine traffic environment. Only signs that exist in Germany's dataset and are also commonly encountered on Philippine roads will be included, ensuring that the training data reflects traffic signs the model is likely to process in real-world conditions. To maintain comprehensive coverage, the chosen signs will be grouped into three main functional categories: regulatory, warning, and informational. Regulatory signs convey mandatory instructions that drivers are required to follow, warning signs provide alerts regarding potential hazards or changes in road conditions, and informational signs deliver guidance or directional information to assist navigation. As both the Philippines and Germany are signatories to the Vienna Convention on Road Signs and Signals, there is an established baseline of international standardization in terms of sign shapes, colors, and meanings [33]. This standardization enables the adaptation of GTSRB classes to the Philippine context while preserving the fidelity of the baseline model replication. The selected traffic sign classes are as follows:

- 20 km/h speed limit
- 60 km/h speed limit
- Bike lane
- Children crossing
- Do not enter
- Pedestrian crossing
- Stop

Fig. 2. Illustration of selected traffic sign classes in the recreated dataset.

Representative images of each selected traffic sign class from the recreated dataset are shown in Fig. 2, providing a clear visual reference for the types of signs included in this study. This visualization ensures that the recreated baseline dataset mirrors the class structure and distribution of the original GTSRB dataset while remaining relevant and applicable to the Philippine traffic environment.

To ensure that the recreated local dataset accurately mirrors the characteristics of the GTSRB dataset, several dataset-level metrics were examined and used as reference points. These include the number of images per class, which specifies the total image count for each selected traffic sign category, and the overall class balance, which reflects the proportional distribution of samples among classes to prevent overrepresentation or underrepresentation of specific signs. Additionally, the brightness distribution per class was analyzed to capture the lighting variability present in the GTSRB dataset, ensuring that the recreated dataset maintains a comparable visual diversity.

To ensure the integrity of the evaluation, a strict separation between training and testing data was maintained. No individual image was used in both the training and evaluation phases. Furthermore, to minimize potential data leakage, the test set was curated to include a diverse range of unique sign captures across varying backgrounds and locations within Metro Manila, ensuring that the model's performance reflects generalized feature learning rather than the memorization of specific image samples.

Table I presents the corresponding number of images available per selected class in the GTSRB training dataset, which serves as the foundation for determining the image count and balance of the recreated Philippine dataset.

TABLE I. TOTAL NUMBER OF IMAGES PER CLASS (GTSRB TRAINING SET) [34].

Class	Number of Images
20 km/h Speed Limit	211
60 km/h Speed Limit	1411
Bike Lane	271
Children Crossing	541
Do Not Enter	1111
Pedestrian Crossing	241
Stop	781

An analysis of the class distribution in the GTSRB dataset reveals noticeable variations in the number of available images per selected class, indicating a degree of class imbalance. As shown in Table I, some classes such as 60 SPEED LIMIT (1,411 images) and DO NOT ENTER (1,111 images) are heavily represented, while others such as 20 SPEED LIMIT (211 images) and PEDESTRIAN CROSSING (241 images) contain significantly fewer samples. This uneven distribution can lead to biased model learning, where the classifier tends to favor more frequent classes during training, potentially reducing recognition accuracy for underrepresented signs.

Another important metric to be referenced in recreating the local dataset is the brightness distribution of the original GTSRB images. Since the primary focus of this study is to evaluate performance under varying lighting conditions, maintaining a similar brightness profile across classes is essential to ensure fairness and consistency when comparing the recreated Philippine dataset to the original. The mean brightness values of all images in each class were computed using OpenCV, allowing the researchers to identify the distribution of image illumination levels within the GTSRB dataset.

The brightness distribution per class is illustrated in Fig. 3 to Fig. 9, where each figure corresponds to a specific traffic sign category, providing a clear overview of how each class is represented across different lighting intensities. This information serves as the foundation for reconstructing a local dataset with comparable lighting characteristics, thereby allowing the baseline model to be trained under similar visual conditions while still reflecting the Philippine traffic environment.

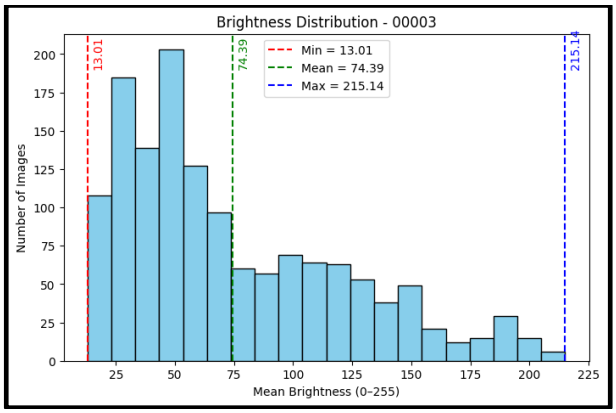

Fig. 3. '20 km/h Speed Limit' Training Set Mean Brightness Distribution Histogram [34].

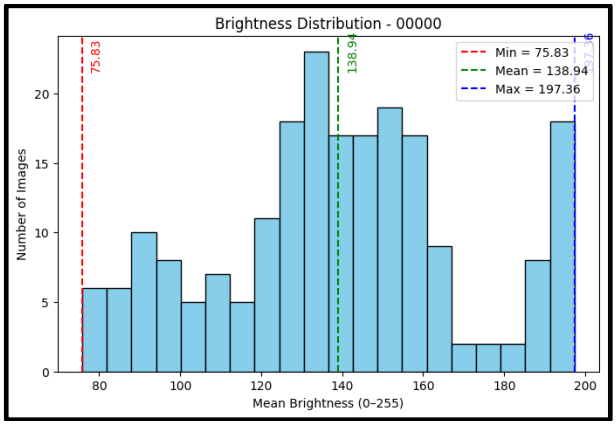

Fig. 4. '60 km/h Speed Limit' Training Set Mean Brightness Distribution Histogram [34].

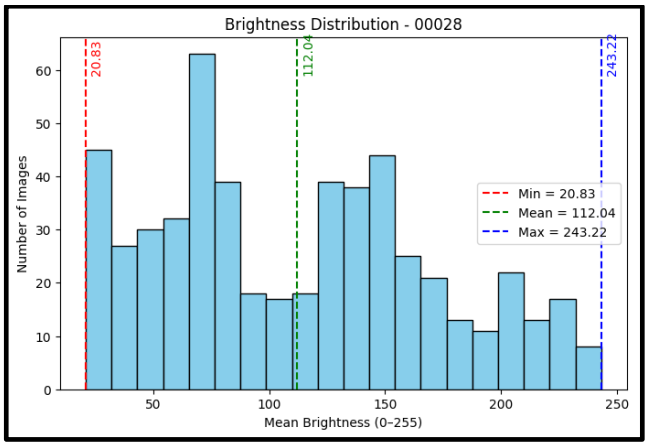

Fig. 5. 'Bike Lane' Training Set Mean Brightness Distribution Histogram [34].

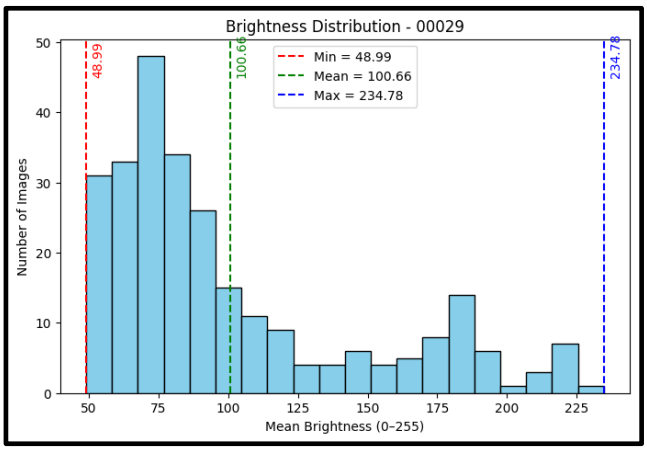

Fig. 6. 'Children Crossing' Training Set Mean Brightness Distribution Histogram [34].

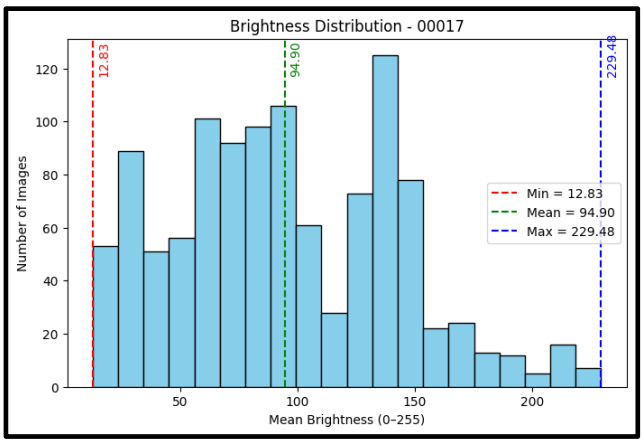

Fig. 7. 'Do Not Enter' Training Set Mean Brightness Distribution Histogram [34].

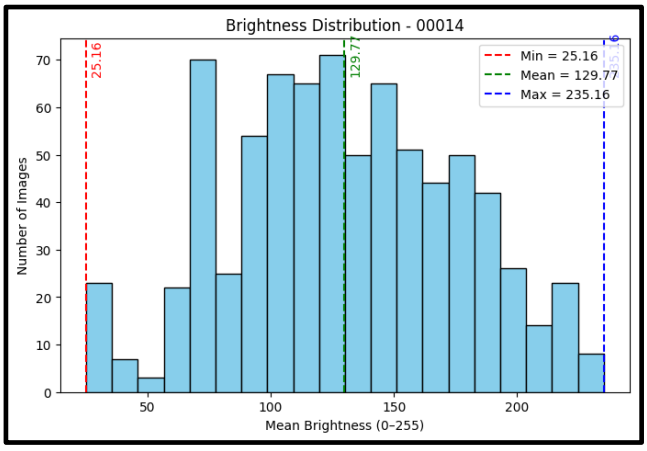

Fig. 8. 'Pedestrians Crossing' Training Set Mean Brightness Distribution Histogram [34].

Upon examining the brightness distribution across classes, it was observed that most categories exhibit significant variability and random gaps in their mean brightness values. Some classes contain a higher concentration of well-lit images, while others have a relatively larger proportion of dimly lit or overexposed samples. This imbalance in brightness distribution introduces potential bias during training, as the model may overfit to dominant lighting conditions and fail to generalize effectively across diverse illumination scenarios.

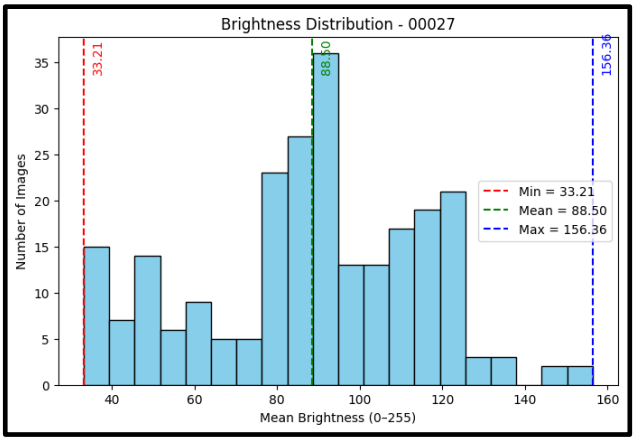

Fig. 9. 'Stop' Training Set Mean Brightness Distribution Histogram [34].

Such inconsistencies also make it more difficult to interpret the model's performance fairly across classes. For instance, a class with predominantly bright samples might appear to perform better, not because of stronger feature learning, but simply due to clearer visual conditions. Conversely, classes with limited representation in darker environments might suffer from poor recognition accuracy in real-world situations. Addressing this imbalance is crucial to ensure that the recreated dataset provides a balanced learning environment, one that fairly represents the varying conditions under which traffic signs are encountered in the Philippines.

To recreate a balanced local dataset that accurately reflects the illumination characteristics of the GTSRB, a stratified approach based on brightness levels was employed. The mean pixel intensity of each image was computed to represent its overall brightness, serving as the basis for stratification. These brightness values were divided into defined intervals, or strata, each covering a range of 20 intensity levels (e.g., 0–19, 20–39, and so on).

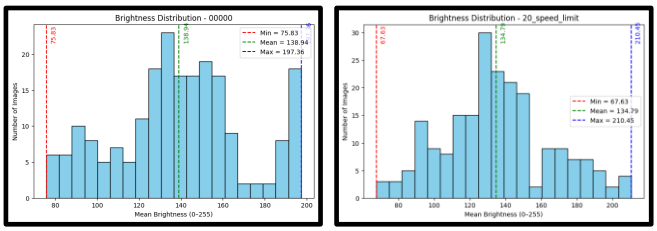

Fig. 10. '20 km/h Speed Limit' Training Set Mean Brightness Distribution Histogram: GTSRB (*left*) [34], Recreated Dataset *(right)*.

Images in the local dataset were then selected or captured to match the proportion of samples within each brightness stratum observed in the GTSRB dataset. An image was considered part of a particular stratum if its mean brightness value fell within the corresponding range. This stratification process ensured that the recreated dataset maintained the same general distribution of lighting conditions as the original dataset, without requiring an exact match for each image. By following this structured approach, the resulting dataset achieves a realistic balance of

lighting variation while remaining feasible to reproduce under local conditions. The resulting histograms, as shown in Fig. 10 to Fig. 16, demonstrate that the recreated dataset retains a realistic variation in brightness while preserving the overall balance achieved through stratification.

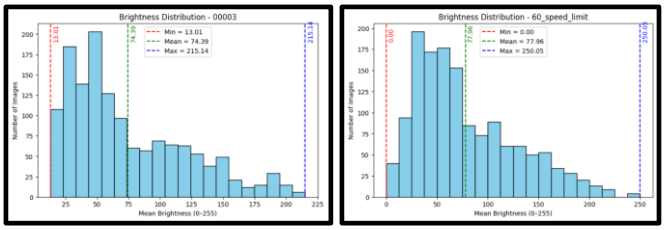

Fig. 11. '60 km/h Speed Limit' Training Set Mean Brightness Distribution Histogram: GTSRB (left) [34], Recreated Dataset (right).

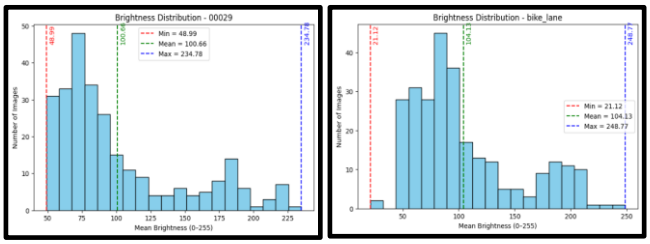

Fig. 12. 'Bike Lane' Training Set Mean Brightness Distribution Histogram: GTSRB (left) [34], Recreated Dataset (right).

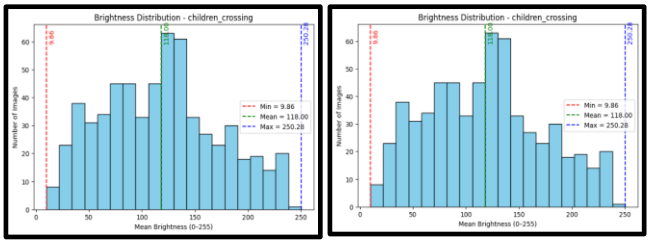

Fig. 13. 'Children Crossing' Training Set Mean Brightness Distribution Histogram: GTSRB (*left*) [34], Recreated Dataset (*right*).

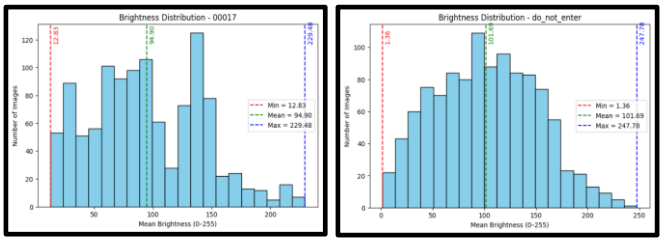

Fig. 14. 'Do Not Enter' Training Set Mean Brightness Distribution Histogram: GTSRB (*left*) [34], Recreated Dataset (*right*).

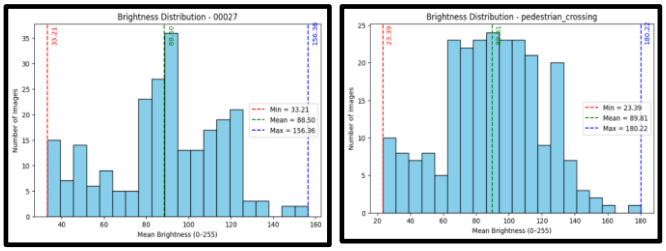

Fig. 15. 'Pedestrian Crossing' Training Set Mean Brightness Distribution Histogram: GTSRB (*left*) [34], Recreated Dataset (*right*).

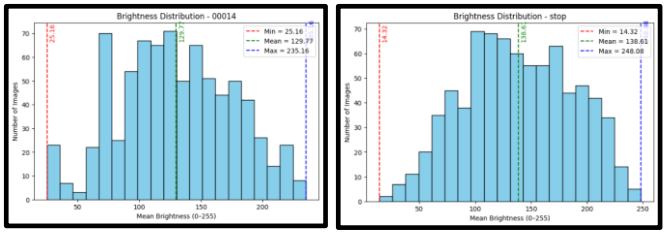

Fig. 16. 'Stop' Training Set Mean Brightness Distribution Histogram: GTSRB (*left*) [34], Recreated Dataset (*right*).

To ensure that the model's evaluation accurately reflects its capability to handle images under varying lighting conditions, both the validation and test sets were carefully constructed to be balanced across brightness levels. This means that the number of images in each brightness range, spanning from low-light to well-lit conditions, was made approximately equal. By doing so, the evaluation process avoids bias toward specific lighting conditions, such as consistently bright or dark images, which could otherwise lead to misleading performance results.

A balanced brightness distribution allows the model's performance to be assessed uniformly across all illumination strata, providing a fair and comprehensive measurement of its robustness. This approach ensures that improvements observed during the study are not a result of overrepresentation of certain brightness levels but are instead indicative of genuine adaptability to diverse lighting scenarios, including those commonly encountered on Philippine roads.

To illustrate this preparation process, visual examples of both the validation and test sets are provided in Fig. 17 and Fig. 18. Each figure includes a histogram representing the mean brightness distribution of one selected class, alongside sample images from different brightness ranges. The histogram demonstrates the uniformity of the distribution across brightness levels, while the example images provide visual confirmation of the dataset's diversity in lighting conditions.

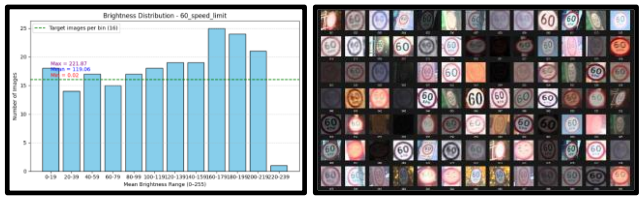

Fig. 17. '60 km/h Speed Limit' Local Validation Set Mean Brightness Distribution (*left*), sample images (*right*)

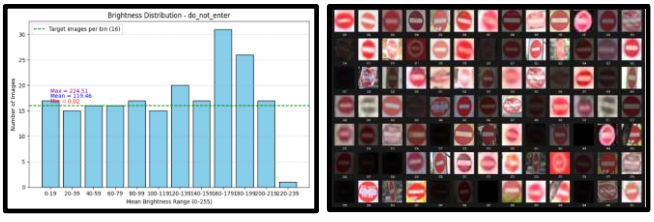

Fig. 18. 'Do Not Enter' Local Test Set Mean Brightness Distribution (*left*), sample images (*right*).

2) Baseline data preprocessing: All images were resized to 32×32 pixels (RGB) to match the input dimensions of the base model. The preprocessing steps were implemented in Python

using the NumPy, Pandas, Matplotlib, and OpenCV libraries. Each image was normalized by scaling pixel values between 0 and 1, ensuring numerical stability and facilitating faster convergence during model training.

To enhance generalization and minimize overfitting, data augmentation was applied using Keras' ImageDataGenerator. The augmentation parameters were configured to replicate the diversity introduced in the foundational study and to simulate real-world variations in lighting, viewing angles, and sign positioning [5]. The transformations applied include:

- Random rotation up to ±15°
- Zoom up to 20%
- Shear transformation
- Brightness adjustments

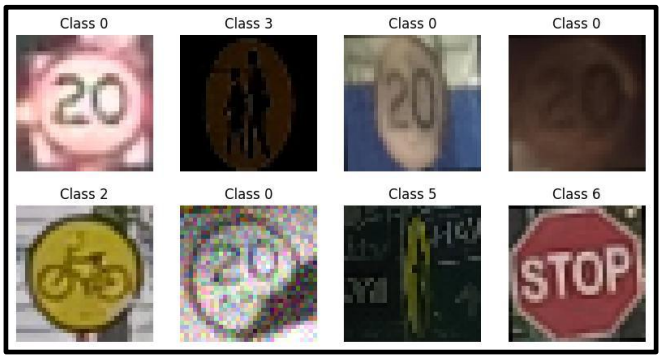

Fig. 19. Baseline preprocessing augmented data sample.

This augmentation process effectively expanded the dataset; a sample of the augmented data is presented in Fig. 19 by generating synthetic variants at a 1:1 ratio, doubling the total number of training samples. Following augmentation, class weights were computed using the "balanced" heuristic to address the persistent class imbalance. By calculating weights based on the final augmented counts, the model's loss function was adjusted to assign higher importance to underrepresented classes. Table II details the final distribution of the training set after augmentation and the corresponding weights utilized during the training phase.

TABLE II. AUGMENTED DATASET DISTRIBUTION AND COMPUTED CLASS WEIGHTS.

Class	Number of Images	Augmented Samples (1:1)	Class Weights
20 km/h Speed Limit	211	422	3.7143
60 km/h Speed Limit	1411	2822	0.4147
Bike Lane	271	542	2.8467
Children Crossing	541	1082	1.1272
Do Not Enter	1111	2222	0.5869
Pedestrian Crossing	241	482	2.9545
Stop	781	1562	0.9630

3) Baseline model creation and training configuration: The recreated model was implemented using TensorFlow/Keras and designed as a VGG-inspired Convolutional Neural Network (CNN), as illustrated in Fig. 20, closely following the configuration outlined in the foundational study. The architecture is composed of four convolutional blocks, each containing a sequence of operations that promote hierarchical feature learning while maintaining computational efficiency.

Each convolutional block consists of a 3×3 convolution layer, followed by Batch Normalization, ReLU activation, MaxPooling, and Dropout for regularization. The use of small receptive fields (3×3) enables the model to capture fine-grained spatial details across multiple layers, while Batch Normalization stabilizes learning and accelerates convergence. Dropout was applied at increasing rates throughout the network to minimize overfitting.

After the convolutional stages, the extracted feature maps were flattened and passed through fully connected (dense) layers with ReLU activations, leading to a final Softmax output layer corresponding to the number of selected traffic sign classes. The Adam optimizer was employed with a learning rate of 1×10^{-4}, while categorical cross-entropy was used as the loss function. Training was conducted over 30 epochs with a batch size of 32, following the same configuration as prescribed in the base paper to preserve fidelity and ensure a fair performance comparison. The detailed layer configuration of the recreated model is presented in Table III.

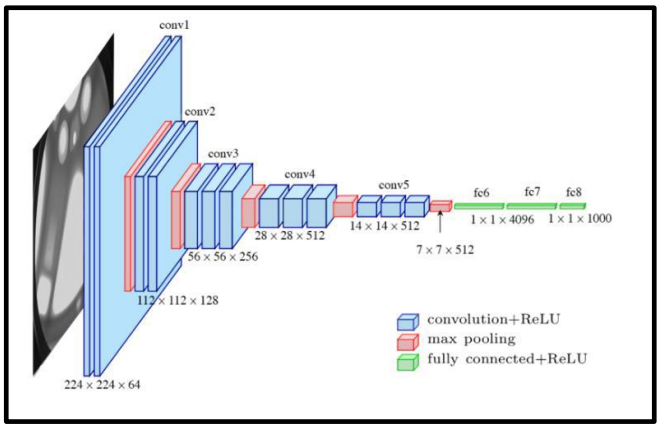

Fig. 20. VGG - 16 network architecture [35].

TABLE III. BASE MODEL RECREATION ARCHITECTURE [5]

Layer Type	Details
Input Layer	32×32×3 RGB image
Conv2D (Block 1)	32 filters, 3×3 kernel, ReLU
Conv2D (Block 2)	64 filters, 3×3 kernel, ReLU
Conv2D (Block 3)	128 filters, 3×3 kernel, ReLU
Conv2D (Block 4)	256 filters, 3×3 kernel, ReLU
Batch Normalization	2×2, after each convolutional block
Dropout	After each convolutional layer
Flatten	0.25 in convolutional blocks, 0.5 in dense layers
Dense Layer 1	Converts feature maps to a vector
Dense Layer 2	256 units, ReLU activation
Output Layer	128 units, ReLU activation

This architecture mirrors the VGG-style progression of gradually increasing filter depth and stacked fully connected layers, consistent with the configuration of the base model. Batch Normalization was incorporated to enhance training stability and speed up convergence, while Dropout was utilized as a regularization technique to mitigate overfitting.

To maintain methodological consistency with the foundational study, the training setup including optimization strategy, learning rate, and other hyperparameters was aligned with those used in the original model, ensuring a fair and reliable performance comparison. A summary of the training configurations of the recreated base model is presented in Table IV below.

TABLE IV. BASE MODEL RECREATION'S TRAINING CONFIGURATIONS

Parameter	Setting
Optimizer	Adam
Learning Rate	0.001
Batch Size	32
Epochs	30
Loss function	Categorical Cross-Entropy
Class Weights	Computed via compute_class_weight('balanced')

To address the issue of class imbalance within the dataset, class weights were calculated and integrated into the training process to ensure that underrepresented classes contributed proportionally to the model's learning. Additionally, EarlyStopping and ReduceLROnPlateau callbacks were employed to enhance training efficiency and model generalization. EarlyStopping halted training once the validation loss stopped improving, while ReduceLROnPlateau automatically lowered the learning rate when progress plateaued, allowing for finer adjustments during later training stages. The training configurations used for the base model recreation are summarized in Table IV.

B. Standard ResNet50 Transfer Learning Model

The second stage of the methodology focuses on developing a baseline ResNet50 model using a standard transfer learning workflow. This model serves as an important comparison point because it represents a widely adopted approach in image classification studies, especially those involving limited datasets. By leveraging ImageNet-pretrained weights, the model begins with a strong feature extraction capability, enabling it to identify edges, shapes, textures, and other low- to mid-level visual patterns before being fine-tuned on the local Philippine traffic sign dataset. This setup allows the study to evaluate how a well-established deep architecture performs under typical training conditions without additional enhancements or multiphase adjustments.

Before training the ResNet50 model, all images undergo a consistent preprocessing pipeline to ensure uniformity and to help the network better adapt to the characteristics of traffic sign images. The preprocessing steps applied in this stage include:

- Image Resizing: All images were resized to 128×128×3 pixels, as this represents a balanced middle ground that preserves sufficient visual detail while still allowing effective reuse of ImageNet-pretrained weights, and reduces computational overhead compared to the original 224×224 input size.
- Normalization: Pixel values were normalized using ImageNet statistics to maintain feature compatibility:
  - Mean = [0.485, 0.456, 0.406]
  - Std = [0.229, 0.224, 0.225]
- Data Augmentation:
  - Random rotation up to ±15°
  - Zoom up to 20%
  - Shear transformation
  - Brightness adjustments
  - Horizontal flipping
  - Addition of Gaussian noise

The model is constructed by loading the ResNet50 architecture with ImageNet weights while excluding the original top classification layers. A custom classification head is then added, typically composed of a global average pooling layer followed by one or more dense layers leading to the final softmax output. This approach allows the pretrained backbone to retain its general feature extraction capabilities while adapting the final layers to the specific classes found in the dataset.

To ensure reproducibility and to make the training procedure clear, the model is trained using a fixed set of hyperparameters and optimization settings. These configurations are summarized in Table V.

TABLE V. STANDARD RESNET MODEL TRAINING CONFIGURATIONS

Layer Type	Details
Input Layer	128×128×3 RGB image
Base Model	ResNet50 (pretrained on ImageNet, include_top=False)
Global Average Pooling	Reduces spatial dimensions
Dense Layer 1	512 units, ReLU, Batch Normalization, Dropout (0.5)
Dense Layer 2	256 units, ReLU
Output Layer	7 units (Softmax activation)
Class Weights	Computed via compute_class_weight('balanced')

C. ResNet50 Multiphase Model

The ResNet50 Multiphase Model introduces a two-stage training pipeline aimed at enhancing the robustness of traffic sign classification models under varying lighting conditions, encompassing both daytime and nighttime environments. This approach leverages transfer learning using the ResNet50 architecture, enabling the model to first acquire general traffic sign recognition features from a large-scale benchmark dataset and subsequently adapt to the visual characteristics of Philippine Road signs. The workflow ensures that the model develops a strong foundational understanding of traffic sign patterns before undergoing localization and fine-tuning with a dataset that reflects real-world conditions in the Philippines.

1) Pretraining on GTSRB dataset: The first phase focuses on pretraining the model using the German Traffic Sign Recognition Benchmark (GTSRB) dataset to transfer fundamental knowledge about traffic sign features and structure. The objective of this phase is to establish a robust feature extractor that can later be fine-tuned for local adaptation, reducing the need for large-scale local data collection.

The GTSRB dataset comprises over 50,000 RGB images across 43 traffic sign classes, including various regulatory, warning, and informational signs. Each image exhibits natural variations in brightness, rotation, and scale, allowing the model to learn from diverse visual conditions. This variability makes GTSRB an ideal source for initial pretraining, as it provides a rich foundation for recognizing traffic signs under a wide range of real-world circumstances.

To ensure compatibility with the network and optimize computational efficiency, the preprocessing steps implemented prior to training are the same as the standard/base ResNet50 model.

A ResNet50-based model was utilized as the backbone architecture for feature extraction, initialized with ImageNet pretrained weights to leverage learned visual representations from large-scale image data. The network was modified to accept 32×32×3 input dimensions, aligning it with the preprocessing pipeline and ensuring compatibility with the GTSRB dataset.

The original fully connected layers of ResNet50 were excluded to allow task-specific adaptation, and a custom classification head was added, consisting of dense layers and dropout. The detailed layer configuration of this model is presented in Table VI.

TABLE VI. RESNET FIRST PHASE MODEL ARCHITECTURE

Layer Type	Details
Input Layer	32×32×3 RGB image
Base Model	ResNet50 (pretrained on ImageNet, include_top=False)
Global Average Pooling	Reduces spatial dimensions
Dense Layer 1	512 units, ReLU, Batch Normalization, Dropout (0.5)
Dense Layer 2	256 units, ReLU
Output Layer	43 units (Softmax activation)

During the initial training phase (Phase 1A), the ResNet50 backbone was frozen to retain the pretrained ImageNet feature representations, while allowing the newly added classification layers to learn task-specific features relevant to traffic sign recognition. This strategy ensures that the foundational visual features remain intact while the upper layers adapt to the new domain. To mitigate the effects of class imbalance inherent in the dataset, class weights were computed and applied during training. Furthermore, data augmentation was implemented through Keras' ImageDataGenerator, introducing variations such as rotation, zoom, shear, brightness adjustment, and horizontal flipping to enhance the model's generalization capability.

Once the model achieved convergence, fine-tuning (Phase 1B) was performed by unfreezing the last 30 layers of the ResNet50 backbone, corresponding approximately to the final convolutional stage (conv5_x block). This step allowed higher-level feature representations to be refined specifically for traffic sign characteristics, while the earlier convolutional layers responsible for low-level feature extraction remained stable. This progressive unfreezing strategy enabled the network to balance feature preservation and domain adaptation, improving its ability to recognize a wide variety of traffic signs under different visual conditions. The detailed training configurations for Phase 1A and Phase 1B are summarized in Table VII.

TABLE VII. TRAINING CONFIGURATIONS PHASE1A + PHASE1B

Layer Type	Details
Optimizer	Adam
Learning Rate	0.001 (Phase 1A), 0.0001 (Phase 1B)
Batch Size	32
Epochs	25 (Phase 1A) + 30 (Phase 1B)
Loss Function	Categorical Cross-Entropy
Class Weights	Computed via compute_class_weight('balanced'),
Callbacks	EarlyStopping, ReduceLROnPlateau, ModelCheckpoint
Input Size	32x32x3

The model was trained and validated using a 70–10–20 data split ratio for training, validation, and testing sets, respectively. This allocation ensured that sufficient data was available for learning while maintaining separate subsets for unbiased validation and performance evaluation. Upon completion of training, the optimized model weights were saved as resnet50_gtsrb_pretrained.h5, which served as the initialization point for the subsequent local dataset fine-tuning phase. This approach ensured continuity in the model's learning process while preserving the generalized traffic sign recognition features obtained from the GTSRB dataset.

2) Phase 2: Fine-tuning on local dataset (7 classes): After pretraining on the GTSRB dataset in Phase 1, the pretrained model was subsequently adapted and fine-tuned using a custom 7-class local dataset. This dataset represents traffic sign categories commonly observed in the Philippines and serves to evaluate the model's ability to generalize from the GTSRB domain to Philippine-style signs.

The local dataset consists of RGB images grouped into the following seven categories: 20 Speed Limit, 60 Speed Limit, Bike Lane, Children Crossing, Do Not Enter, Pedestrian Crossing, and Stop. To enable effective transfer learning from the pretrained model, each class was mapped to its semantically closest GTSRB counterpart. This mapping ensured that the visual and functional characteristics of the signs such as shape, color, and meaning remained consistent across both datasets. The complete mapping between the local classes and their corresponding GTSRB class IDs is presented in Table VIII. Through this process, the pretrained model's learned feature representations from GTSRB could be directly leveraged and refined for the local traffic environment.

TABLE VIII. LABEL MAPPING

Local Class	Mapped GTSRB Class ID	Description
20 Speed Limit	0	Speed Limit 20 km/h
60 Speed Limit	3	Speed Limit 60 km/h
Bike Lane	29	Bicycles Crossing
Children Crossing	28	Children Crossing
Do Not Enter	17	No Entry
Pedestrian Crossing	27	Pedestrians
Stop	14	Stop Sign

The dataset images were resized to 64×64 pixels to maintain a balance between visual detail and computational efficiency. The dataset was then split into Training, Validation, and Testing subsets, each containing the same seven class folders to preserve class distribution across all splits. To enhance the model's ability to generalize to unseen data, the training set underwent in-memory data augmentation, with each image randomly subjected to one of several transformations. The complete architecture and setup of the Phase 2 model are summarized in Table IX.

TABLE IX. PHASE 2 MODEL SETUP

Layer Type	Details
Input	64×64×3 RGB image
Base Model	ResNet50 (include_top = False, initialized using pretrained weights from Phase 1 via layer-wise weight transfer)
Global Average Pooling	
Dense (512, ReLU) + Batch Normalization + Dropout(0.5)	
Dense (256, ReLU)	
Output Layer	7 units, Softmax

Although convolutional layers in ResNet50 are generally input-size agnostic, the change in input resolution from 32×32 to 64×64 affects the overall feature map dimensions and prevents direct full-model weight loading. To address this, weights obtained from Phase 1 were transferred to the new model through layer-wise matching based on layer names. Only layers with compatible dimensions were assigned pretrained weights, while incompatible layers primarily within the classification head were reinitialized. This approach preserves learned feature representations while allowing architectural adaptation to higher-resolution inputs.

All layers except the final 30 layers of the network were frozen. This configuration enables fine-tuning of the high-level layers of the ResNet50 architecture, along with the custom classification head, while preserving earlier convolutional features. The detailed training configuration for Phase 2 is summarized in Table X.

Training was performed with class weighting to mitigate imbalance across categories and real-time validation on the held-out set. Model checkpoints were saved whenever validation accuracy improved.

TABLE X. PHASE 2 TRAINING CONFIGURATION

Parameter	Setting
Optimizer	Adam
Learning Rate	0.0001
Batch Size	32
Epochs	30 w/ Early Stopping
Loss Function	Categorical Cross-Entropy
Class Weights	Computed via compute_class_weight('balanced'),
Callbacks	Early Stopping, ReduceLROnPlateau, ModelCheckpoint
Input Size	64×64×3

D. Standard ResNet50 with Zero-DCE Preprocessing Model

The fourth phase of the methodology introduces a low-light enhancement component by integrating Zero-DCE preprocessing into the standard ResNet50 transfer learning pipeline. This model follows the same architectural setup, training procedure, and evaluation workflow described in Section B (Standard ResNet50 Transfer Learning Model). The only difference lies in the dataset used for both training and testing, where images undergo enhancement to simulate real-world low-visibility conditions.

To maintain consistency across all datasets used in this model variant, every image, regardless of whether it belongs to the training, validation, or testing subset, is processed using the Zero-DCE algorithm before being fed into the pipeline. Zero-DCE applies an adaptive curve-based enhancement to each image, adjusting brightness and contrast in a data-driven way without the need for paired ground-truth references. An example of Zero-DCE preprocessing is illustrated in Fig. 21, showing an unprocessed image (left) and its enhanced version (right). This preprocessing rule ensures that all inputs follow the same enhancement procedure, allowing the study to assess how a fully Zero-DCE-processed dataset affects ResNet50's classification performance under a unified condition.

Fig. 21. Zero-DCE Enhancement: Unprocessed Image (*left*), Zero-DCE Processed Image (*right*).

Through this selective enhancement process, the model is exposed to conditions that more closely resemble real-world driving environments where visibility can vary significantly. By applying Zero-DCE only to images with a mean brightness value below the defined threshold, the preprocessing step specifically targets low-light samples while avoiding unnecessary modifications to images that are already well-exposed. This prevents potential distortions in brightness, contrast, or color balance that could occur if enhancement were applied uniformly

across all images. This configuration allows the study to evaluate whether applying low-light enhancement as a preprocessing step improves classification performance compared to the standard ResNet50 model. The results from this model also provide an important reference when analyzing the combined effects of illumination enhancement and multiphase fine-tuning in the subsequent methodology stage.

E. Multiphase ResNet50 with Zero-DCE Preprocessing Model

This model follows the same multiphase training strategy outlined in Section C (ResNet50 Multiphase Model), but with one key modification: dataset images are subjected to Zero-DCE enhancement prior to entering the training pipeline. Similar to the preprocessing rule established in the previous model configuration, images are first evaluated based on brightness level. Any image with a mean brightness value below 40 is processed using the Zero-DCE algorithm before being used for training, validation, or testing. This approach ensures that darker samples are enhanced while images with sufficient illumination remain unchanged, maintaining a consistent rule-based preprocessing strategy across the study.

ResNet50 is initialized with ImageNet weights and adapted to traffic sign recognition using the GTSRB dataset. The final classification layer is replaced to match the 43 GTSRB classes, while earlier layers remain frozen to preserve foundational visual features. Training is performed until convergence, and intermediate pretrained weights (excluding the last fully connected layer) are saved for transfer. All other training hyperparameters follow the configurations reported in Table VI to Table VII.

In Phase 2, the pretrained weights are loaded into a new ResNet50 model configured for the seven-class Philippine traffic sign dataset. Early layers (approximately the first 30–40 layers) are kept frozen to retain core feature representations, while the remaining layers, together with the custom classification head, are fine-tuned using a reduced learning rate. The dataset images are resized to 64 × 64 and undergo the same preprocessing pipeline consisting of brightness-based Zero-DCE enhancement, normalization, and augmentation. This consistent preparation process allows the model to adapt from the large-scale GTSRB feature space to the characteristics of locally collected traffic signs.

All other hyperparameters, including learning rate schedules, batch size, number of epochs, loss function, class weighting, and callbacks, follow the configurations summarized in Table V to Table X. Maintaining these identical training settings ensures that any observed differences in model performance can be attributed to the integration of multiphase transfer learning and Zero-DCE preprocessing rather than changes in experimental parameters.

By incorporating brightness-aware Zero-DCE enhancement within a multiphase training framework, this model configuration allows the study to analyze how illumination enhancement interacts with staged fine-tuning. The results obtained from this variant help determine whether combining domain-specific pretraining with adaptive low-light enhancement can produce more robust traffic sign classification performance under conditions representative of real-world Philippine driving environments.

F. Model Evaluation

To assess each model's performance, we will use standard evaluation metrics:

- Accuracy - Measures how often the model makes correct predictions out of all predictions made.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (1)$$

- Precision and Recall - Precision measures how many of the instances that the model predicted as traffic signs are correct. Recall measures how many actual traffic signs the model correctly identified.

$$Precision = \frac{TP}{TP+FP} \quad (2)$$

$$Recall = \frac{TP}{TP+FN} \quad (3)$$

- F1 score - Balanced measure of Precision and Recall.

$$F1\ Score = 2 \times \frac{Precision \times Recall}{Precision+Recall} \quad (4)$$

- Confusion Matrix - A table that shows how well the model distinguishes between different class.

IV. Results and Discussion

A. VGG-Inspired Baseline Model Results

The recreation of the baseline model, designed as a VGG-inspired convolutional neural network, was trained from scratch on the 7-class traffic sign dataset. To mitigate class imbalance, class weighting was applied during training, and a learning rate scheduler progressively reduced the learning rate after the 20th epoch to facilitate stable convergence. Early stopping was also used to prevent overfitting.

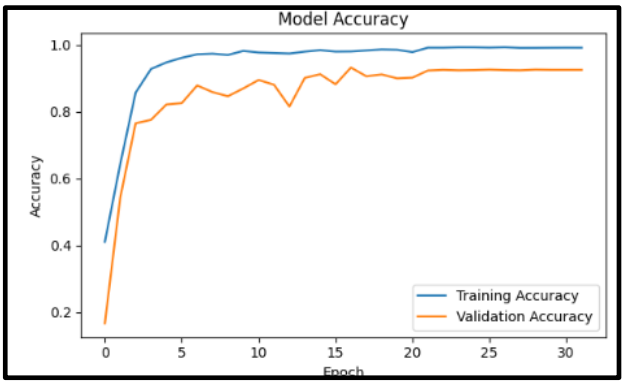

Fig. 22. Baseline model training history.

The model exhibited steady performance improvements across epochs. Initial training accuracy was 41.11%, increasing rapidly to 85.76% by the third epoch, and reaching over 93% by epoch 15. Validation accuracy followed a similar trend, improving from 16.69% in the first epoch to 91.28% by epoch 15. After the learning rate reduction at epoch 22, the model stabilized, with minimal fluctuation between training and validation performance, indicating good generalization. The

training and validation performance over epochs is shown in Fig. 22. The best recorded validation accuracy was 92.65%, with a validation loss of approximately 0.34.

Upon doing further evaluation on the unseen test-set, the model achieved a final test accuracy of 92.17%, confirming robust generalization capability on unseen data. The detailed classification report is presented in Table XI.

TABLE XI. BASELINE MODEL CLASSIFICATION RESULT

Class	Precision	Recall	F1 - Score
20_speed_limit	0.92	0.95	0.93
60_speed_limit	0.86	0.76	0.81
bike_lane	0.84	0.98	0.90
children_crossing	0.98	0.98	0.98
do_not_enter	0.93	0.91	0.92
pedestrian_crossing	1.00	0.97	0.99
stop	0.94	0.90	0.92

The per-class metrics indicate that the model performed exceptionally well on categories with distinct geometric and semantic features, such as children_crossing and pedestrian_crossing, which achieved F1-scores of 0.98 and 0.99 respectively. These high scores suggest that the model effectively captured the unique triangular shapes and human-figure silhouettes characteristic of these classes.

In contrast, the 60_speed_limit class exhibited the lowest performance with a recall of 0.76 and an F1-score of 0.81. An example of the class's failure cases as presented in Fig. 23, reveals that this is primarily due to high inter-class similarity with the 20_speed_limit category. Both classes share an identical 'red-circle' global feature and a common '0' digit, forcing the VGG-inspired network to rely entirely on the fine-grained internal features of the '6' versus the '2'.

Fig. 23. Failure cases (VGG-inspired baseline model).

Additionally, several misclassifications can be attributed to poor image visibility, particularly in low-light or underexposed conditions. Since this baseline model does not incorporate any dedicated enhancement technique such as Zero-DCE, it is more sensitive to illumination variations, resulting in degraded feature extraction for darker images. This limitation is illustrated in Fig. 24, which shows representative failure cases for dark images. The figure highlights the baseline model's reduced robustness to challenging lighting conditions and motivates the inclusion of enhancement-based approaches in subsequent model variants.

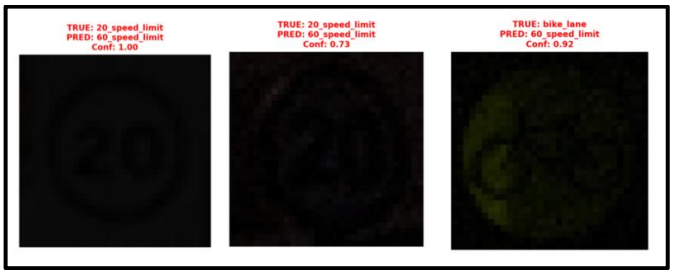

Fig. 24. Dark images failure cases (VGG-inspired baseline model).

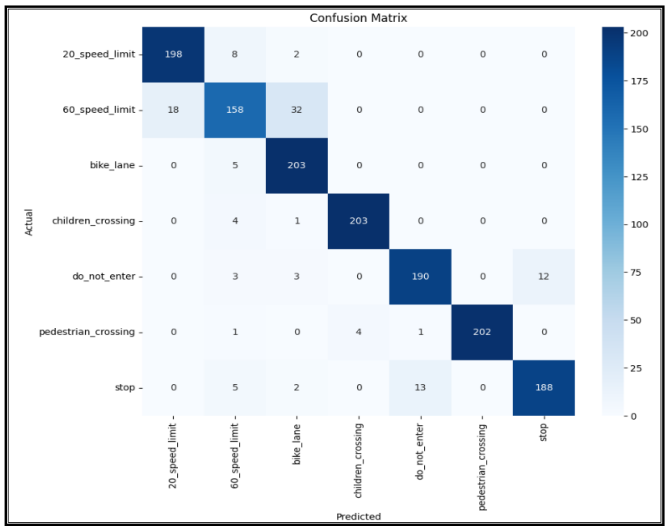

Fig. 25. VGG-inspired baseline model confusion matrix.

The overall classification performance across all classes is summarized in the confusion matrix, presented in Fig. 25, providing a comprehensive view of inter-class misclassifications and model accuracy. The results demonstrate that the VGG-inspired model effectively captured discriminative visual features, achieving strong generalization despite being trained from scratch without pretrained weights. This model serves as a solid performance baseline for subsequent experiments and enhancements, aligning with the methodological framework proposed in the base study.

B. Standard ResNet50 Transfer Learning Model Results

The Standard ResNet50 Transfer Learning Model showed steady improvement throughout training, with its performance stabilizing toward the later epochs. By epoch 28, the model reached a validation accuracy of 91.99%, indicating that the pretrained ImageNet features were successfully adapted to the local seven-class traffic sign dataset. The per-class precision, recall, and F1-scores of the Standard ResNet50 Transfer Learning Model are summarized in Table XII, providing a detailed view of the model's classification performance across all seven classes.

The per-class precision, recall, and F1-scores reveal a generally strong and balanced performance, with most classes achieving scores near or above 0.90. Signs with distinct shapes and colors such as bike_lane and do_not_enter were classified with perfect or near-perfect precision, reflecting the model's ability to capture well-defined visual patterns.

TABLE XII. STANDARD RESNET50 TRANSFER LEARNING MODEL CLASSIFICATION RESULT.

Class	Precision	Recall	F1 - Score
20_speed_limit	0.90	0.90	0.90
60_speed_limit	0.93	0.86	0.89
bike_lane	1.00	1.00	1.00
children_crossing	0.86	1.00	0.92
do_not_enter	1.00	0.88	0.93
pedestrian_crossing	0.97	0.86	0.91
stop	0.85	0.99	0.91

However, a critical area of confusion was observed between the pedestrian_crossing and children_crossing categories. While children_crossing achieved a perfect Recall of 1.00, its Precision (0.86) was lower, indicating a high rate of False Positives. Conversely, pedestrian_crossing showed high Precision (0.97) but lower Recall (0.86).

To further examine class-level performance, a confusion matrix was generated, illustrating how accurately the model classified each of the seven traffic sign categories. This confusion matrix is presented in Fig. 26, revealing both the model's strengths and the specific classes where misclassifications occurred, and offering insight into potential areas for refinement in future iterations.

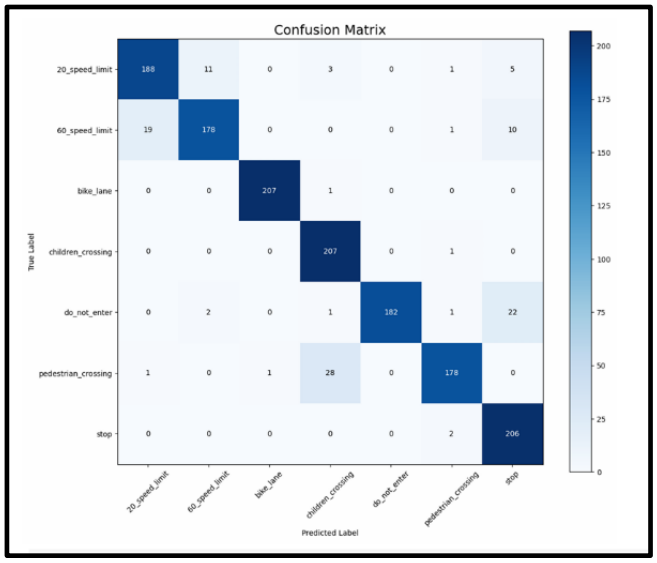

Fig. 26. Standard ResNet50 transfer learning model confusion matrix.

The confusion matrix confirms that the model frequently misclassified pedestrian_crossing samples as children_crossing. This suggests that the ResNet50 backbone successfully identified the 'person' silhouette and triangular yellow frame features shared by both classes but struggled to resolve the fine-grained semantic differences between a single figure and a multi-figure iconography. These misclassifications are illustrated in Fig. 27, highlighting specific failure cases for the Standard ResNet50 model without Zero-DCE. This 'silhouette grouping' error points to a limitation in the model's ability to distinguish highly similar semantic icons without additional spatial or illumination enhancements.

Fig. 27. Failure cases (Standard ResNet50 without Zero-DCE).

While numeric confusion was also noted between 20_speed_limit and 60_speed_limit, a significant portion of these failure cases involved extreme low-light conditions where visual features were nearly indiscernible to the naked eye, similar to Fig. 24. Because this standard ResNet50 model lacks a dedicated enhancement stage, these samples were prone to misclassification based on chromatic noise rather than structural features. This gap in performance underscores the necessity of the proposed illumination-enhancement phase (Zero-DCE) explored in subsequent stages of this study.

C. Multiphase ResNet50 Model Results

1) Pretraining on the GTSRB dataset: In the first phase of the methodology, a ResNet50 backbone initialized with ImageNet weights was trained on the German Traffic Sign Recognition Benchmark (GTSRB) dataset to develop a generalized feature extractor for traffic sign classification. The model was initially trained for 25 epochs with a learning rate of 0.001. During early epochs, the model showed steady improvement from an initial validation accuracy of 50.9% at epoch 1 to 78.5% by epoch 25, accompanied by a decrease in validation loss frm 1.50 to 0.62.

Since 78% was still not satisfactory to us, we decided to add 10 more training epochs, maintaining a reduced learning rate of 0.0001 This added training epochs led to significant improvements in both training and validation performance. Validation accuracy progressively increased from 89.0% (epoch 21) to a final value of 91.15% (epoch 30), while validation loss decreased from 0.32 to 0.26. The model demonstrated stable learning with no signs of overfitting throughout this period.

Final evaluation on the GTSRB test set yielded a test accuracy of 91.23% and a test loss of 0.2661, confirming that the pretrained model achieved strong generalization performance across all 43 traffic sign categories. These results demonstrate that phase 1 was successful in producing a robust pretrained model that captured the distinguishing features of traffic signs in the GTSRB dataset.

Conceptually, this phase served as the foundation for domain adaptation. Leveraging the ImageNet-initialized ResNet50 architecture and retraining it on the GTSRB dataset, the model successfully transferred general visual representations. This will serve as the initialization point for the 2nd phase of the training wherein it will undergo fine-tuning on the 7-class local dataset that was also used for the baseline model recreation for a fair comparison. This strategic transfer of learned representations is expected to enhance convergence speed, stability, and recognition accuracy during fine-tuning.

2) Fine-tuning on the 7-class local dataset: During the second phase, the pretrained ResNet50 model underwent fine-tuning using a specialized, 7-class dataset of local traffic signs. The objective was to transfer the model's general knowledge from the GTSRB benchmark to a more context-specific domain. To mitigate overfitting and class imbalance, the training regimen employed data augmentation techniques and class-weighted loss functions. The per-class precision, recall, and F1-scores of the fine-tuned model are summarized in Table XIII, showing strong performance across all seven traffic sign categories.

TABLE XIII. MULTIPHASE RESNET50 MODEL CLASSIFICATION REPORT

Class	Precision	Recall	F1 - Score
20_speed_limit	0.99	0.92	0.96
60_speed_limit	0.91	0.96	0.93
bike_lane	0.96	1.00	0.98
children_crossing	0.99	0.99	0.99
do_not_enter	0.92	0.99	0.96
pedestrian_crossing	1.00	0.99	0.99
stop	1.00	0.91	0.95

The model was fine-tuned for up to 30 epochs, but it converged rapidly, achieving 94.57% validation accuracy by the 10th epoch. This quick learning is due to the effective use of its pretrained features, the model only needed to adjust its higher-level layers to specialize for the new traffic signs, as it already understood basic visual patterns from its initial training.

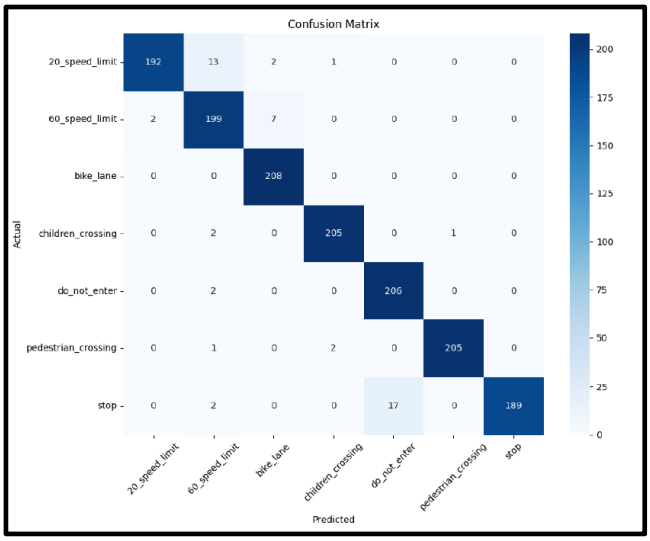

Fig. 28. Multiphase ResNet50 model confusion matrix.

Final testing confirmed the model's robust generalization capability, with a test accuracy of 96.43% and a loss of 0.2569. As detailed in Table XIII, the model demonstrated balanced performance across all classes, with mean precision, recall, and F1-scores all approximately 0.95. It showed particular proficiency in classifying the 'children crossing,' 'pedestrian crossing,' and 'stop' signs, achieving F1-scores greater than or equal to 0.95, which indicates a strong ability to distinguish between both regulatory and warning signs. The confusion matrix of the multiphase ResNet50 model is presented in Fig. 28, providing a detailed view of the model's predictions and misclassifications across all classes.

Despite the high overall accuracy, a qualitative analysis of the remaining misclassifications reveals that the model's performance is primarily constrained by environmental degradation rather than architectural limitations. These low-light failure cases are illustrated in Fig. 29, highlighting instances where the model struggled to correctly classify underexposed images.

Fig. 29. Dark images failure cases (Multiphase ResNet50 without Zero-DCE).

The majority of failure cases observed in the stop sign (Recall: 0.91) and 60_speed_limit (Precision: 0.91) occurred in samples with extreme underexposure. In these instances, the Stop sign's characteristic red hue and octagonal border become indistinguishable from the background and the slight precision dip in the 60_speed_limit class suggests that in low-light, the model occasionally over-identifies circular features as "60".

D. Standard ResNet50 with Zero-DCE Preprocessing Model Results

The Standard ResNet50 model trained with Zero-DCE enhanced data reached its highest validation accuracy of 90.59% at epoch 17. This performance indicates that the model was able to adapt well to the modified visual characteristics introduced by the enhancement process, despite the increased contrast and amplified brightness present in all samples. The per-class precision, recall, and F1-scores of the Zero-DCE enhanced model are summarized in Table XIV, showing the classification performance across all seven traffic sign categories.

TABLE XIV. STANDARD RESNET50 WITH ZERO-DCE PREPROCESSING MODEL CLASSIFICATION REPORT.

Class	Precision	Recall	F1 - Score
20_speed_limit	0.93	0.89	0.91
60_speed_limit	0.91	0.93	0.92
bike_lane	1.00	1.00	1.00
children_crossing	0.82	1.00	0.90
do_not_enter	0.99	0.86	0.92
pedestrian_crossing	1.00	0.77	0.87
stop	0.85	0.99	0.92

The integration of Zero-DCE preprocessing with the Standard ResNet50 architecture yielded a final test accuracy of 92.10%, representing a marginal improvement over the non-

enhanced baseline ResNet50 model of 91.99%. While the model successfully adapted to the modified visual characteristics such as increased contrast and amplified brightness, the incremental gain suggests a complex trade-off between recovered visibility and newly introduced digital noise.

As shown in Table XIV, the impact of Zero-DCE enhancement varies significantly across categories. Highly distinct geometric signs, such as the bike_lane class, maintained perfect F1-scores, proving that Zero-DCE does not distort robust structural features. Notably, the 60_speed_limit class saw a recovery in recall compared to the standard baseline. This indicates that by "unmasking" the internal digits in low-light conditions, Zero-DCE allowed the ResNet50 kernels to resolve numeric details that were previously lost to shadow.

Examples of misclassified low-light images under Zero-DCE preprocessing are presented in Fig. 30, illustrating how certain enhancement artifacts can still challenge the model. The overall classification performance and misclassification patterns are summarized in the confusion matrix shown in Fig. 31, providing a detailed view of class-level successes and remaining errors.

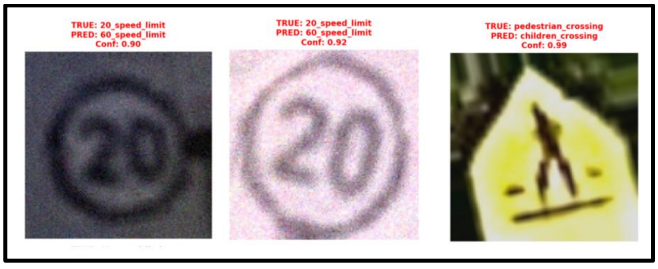

Fig. 30. Failure Cases (Standard ResNet50 with Zero-DCE).

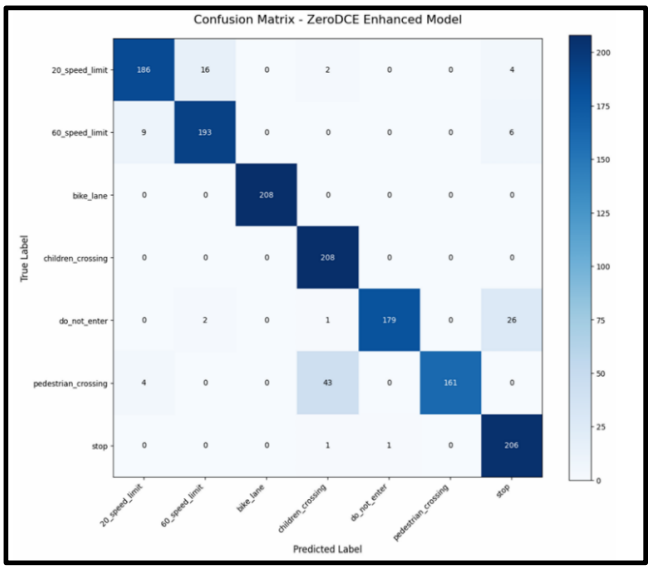

Fig. 31. Standard ResNet50 with Zero-DCE preprocessing model confusion matrix.

Despite these gains, the model exhibited nuanced shifts in categories with high semantic similarity. Pedestrian crossing achieved flawless precision but suffered a noticeable decrease in Recall (0.77). This suggest that the enhancement process occasionally amplifies background textures or edge artifacts to a degree that confuses the model's "person" detection filters, causing it to fail on valid samples. The failure cases in Fig. 30 confirm that while Zero-DCE successfully rescues visibility in extreme low-light samples, allowing the classification of previously invisible signs; the benefit is partially neutralized by enhancement artifacts.

E. Multiphase ResNet50 with Zero-DCE Preprocessing Model Results

To test the effectiveness of using image enhancement in image classifications, specifically, local traffic signs, we fine-tuned the Phase 1 model to a local dataset that underwent Zero-DCE image enhancement.

Similar to Phase 2a, the model was also fine-tuned for 30 epochs, but converged rapidly, converging faster than Phase 2a's model. At the first epoch, the model was able to achieve a training accuracy of 85.6% and a validation accuracy of 92.45%, which is superior compared to Phase 1a's first epoch results of a training accuracy of 84.33% and a validation accuracy of 90.25. The model then went off to achieve a final 95.33% validation accuracy at the 6th epoch with a training accuracy of 99.55%. The per-class precision, recall, and F1-scores of the Zero-DCE enhanced multiphase ResNet50 model are summarized in Table XV, highlighting strong performance across all seven traffic sign categories.

TABLE XV. MULTIPHASE RESNET50 WITH ZERO-DCE PREPROCESSING MODEL CLASSIFICATION REPORT.

Class	Precision	Recall	F1 - Score
20_speed_limit	0.99	0.98	0.99
60_speed_limit	0.98	0.96	0.97
bike_lane	0.96	1.00	0.98
children_crossing	0.99	1.00	0.99
do_not_enter	0.97	0.99	0.98
pedestrian_crossing	1.00	0.99	0.99
stop	1.00	0.97	0.98

Final testing upon an unseen test set was done and the model demonstrated excellent performance with 98.21% overall accuracy and a macro F1-score of 0.98, indicating strong and consistent generalization across all seven sign categories. The primary errors occurred between visually similar red circular signs, suggesting that performance could be further improved with more targeted data or feature enhancement. This improvement demonstrates the effectiveness of combining domain-aligned pretraining with illumination enhancement, allowing the model to learn both robust traffic sign features and improved visibility representations simultaneously. The confusion matrix, shown in Fig. 32, confirms the strong performance of the Multiphase ResNet50 model with Zero-DCE preprocessing.

Analysis of the remaining 1.79% of errors indicates that a subset of these cases is attributable to extreme image degradation. As shown in Fig. 33, some samples captured under severe low-light conditions exhibit significant sensor noise and loss of structural detail, resulting in high-frequency artifacts that obscure critical features. In such cases, even Zero-DCE

enhancement is unable to recover meaningful visual information, as the original signal is insufficient.

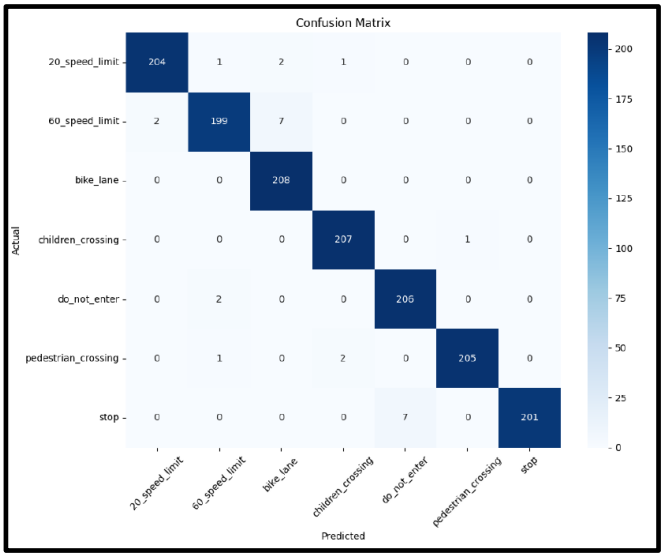

Fig. 32. Multiphase ResNet50 with Zero-DCE preprocessing model confusion matrix.

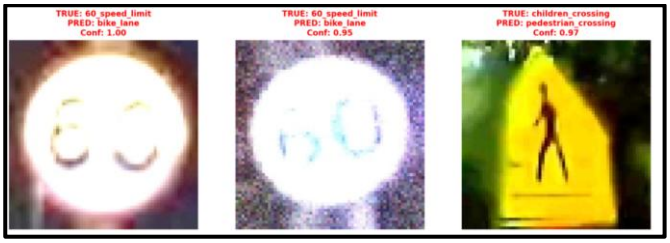

Fig. 33. Failure Cases (Multiphase ResNet50 Model with Zero-DCE).

Additionally, minor misclassifications persist among visually similar classes, particularly red circular signs, where distinctions rely on fine-grained internal patterns such as numerical digits or symbol variations. These findings suggest that while the proposed model significantly improves robustness to illumination, performance is ultimately constrained by the quality of the captured input data and inherent inter-class similarity.

V. Conclusion and Recommendations

This study compared five different model configurations to understand how architecture depth, transfer learning, and low-light enhancement influence traffic sign recognition performance on a 7-class local dataset. The results highlight clear differences in each model's learning capability and show how combining large-scale pretraining with illumination enhancement leads to the strongest overall performance.

The VGG-inspired baseline achieved a 92.17% test accuracy, establishing that even a compact CNN trained from scratch can learn meaningful visual patterns. However, its performance also reflected the limitations of small datasets, especially when distinguishing between signs with similar shapes or colors. This model served as an important benchmark for evaluating the benefits of more advanced training strategies.

The Standard ResNet50 Transfer Learning Model slightly improved performance to 92.45%, demonstrating the value of leveraging ImageNet features. While the gain over the baseline was modest, it confirmed that deeper architectures with pretrained filters can offer better generalization, especially for texture-heavy signs like bike_lane or pedestrian_crossing.

The Multiphase ResNet50 Model produced a substantial jump to 96.43%. Training first on the GTSRB dataset gave the model a strong traffic-sign-specific feature backbone, which translated effectively to the local dataset. This result illustrates how domain-aligned pretraining provides more relevant priors than generic ImageNet features, allowing the network to recognize sign patterns with greater confidence.

Interestingly, the Standard ResNet50 model with Zero-DCE enhancements achieved 92.10%, slightly lower than its non-enhanced counterpart. This outcome suggests that while Zero-DCE improves brightness and contrast, applying it to all images can also alter the natural appearance of well-lit samples. For the standard transfer learning setup, where the model learned from a mix of lighting conditions, this uniform enhancement introduced small inconsistencies between training and evaluation samples, leading to subtle misclassifications.

In contrast, the Multiphase ResNet50 with Zero-DCE achieved the highest performance at 98.21%, outperforming all other models by a wide margin. In this setup, Zero-DCE complemented the traffic-sign-specific features learned during Phase 1. Because the multiphase model already had a strong understanding of structural sign patterns, the enhanced visibility provided by Zero-DCE amplified relevant edges and shapes without confusing the feature extractor. This synergy resulted in cleaner gradients, faster convergence, and more balanced class-wise metrics, especially for signs that are typically harder to see in low-light scenarios.

Overall, the results show that illumination enhancement alone is not guaranteed to boost performance. Its effectiveness depends heavily on the strength of the underlying feature extractor. Zero-DCE becomes most beneficial when paired with a domain-trained model, as seen in the multiphase pipeline, where the improvement from 96.43% to 98.21% represents the largest gain across all experiments. This final model stands as the most reliable and robust solution for traffic sign recognition under Philippine conditions, especially in environments where visibility is limited.

References

[1] A. Voulodimos, N. Doulamis, A. Doulamis, and E. Protopapadakis, "Deep learning for computer vision: a brief review," *Computational Intelligence and Neuroscience*, vol. 2018, no. 1, pp. 1–13, Feb. 2018, doi: https://doi.org/10.1155/2018/7068349.

[2] E. Dilek and M. Dener, "Computer vision applications in intelligent transportation systems: a survey," *Sensors*, vol. 23, no. 6, p. 2938, Mar. 2023, doi: https://doi.org/10.3390/s23062938.

[3] S. B. Wali et al., "Vision-based traffic sign detection and recognition systems: current trends and challenges," *Sensors*, vol. 19, no. 9, p. 2093, May 2019, doi: https://doi.org/10.3390/s19092093.

[4] D. Cireşan, U. Meier, J. Masci, and J. Schmidhuber, "Multi-column deep neural network for traffic sign classification," *Neural Networks*, vol. 32, pp. 333–338, Aug. 2012, doi: https://doi.org/10.1016/j.neunet.2012.02.023.

[5] Rudri Mahesh Oza, A. Geisen, and T. Wang, "Traffic sign detection and recognition using deep learning," Sep. 2021, doi: https://doi.org/10.1109/ai4i51902.2021.00012.

[6] X. Zhang, "Research on automatic driving safety image recognition based on deep learning," *2024 IEEE 7th International Conference on Automation, Electronics and Electrical Engineering (AUTEEE)*, pp. 457–463, Dec. 2024, doi: https://doi.org/10.1109/auteee62881.2024.10869683

[7] J. Luo and Z. Wang, "A low latency traffic sign detection model with an automatic data labeling pipeline," *Neural Computing and Applications*, vol. 34, no. 18, pp. 15499–15512, Apr. 2022, doi: https://doi.org/10.1007/s00521-022-07253-x

[8] H. Zhang, H. Singh, M. Ghassemi, and S. Joshi, "'Why did the model fail?': attributing model performance changes to distribution shifts," *arXiv.org*, 2022. https://arxiv.org/abs/2210.10769

[9] S. Lefèvre, E. Aptoula, B. Perret, and J. Weber, "Morphological template matching in color images," *Lecture Notes in Computational Vision and Biomechanics*, pp. 241–277, Dec. 2013, doi: https://doi.org/10.1007/978-94-007-7584-8_8.

[10] B. Sanyal, R. K. Mohapatra, and R. Dash, "Traffic sign recognition: a survey," *IEEE Xplore*, Jan. 01, 2020. https://ieeexplore.ieee.org/abstract/document/9072976 (accessed Jul. 02, 2023).

[11] O. T. Nartey, G. Yang, S. K. Asare, J. Wu, and L. N. Frempong, "Robust semi-supervised traffic sign recognition via self-training and weakly-supervised learning," *Sensors*, vol. 20, no. 9, p. 2684, May 2020, doi: https://doi.org/10.3390/s20092684.

[12] L. Vieira, "Comparing performance of preprocessing techniques for traffic sign recognition using a HOG-SVM," *arXiv.org*, 2025. https://arxiv.org/abs/2504.09424.

[13] S. Somvanshi, R. Sheley, S. A. Shuvo, A. Rafe, and S. Das, "A survey on automated vehicles in low visibility and infrastructure-limited roadway settings," 2025, doi: https://doi.org/10.2139/ssrn.5387394.

[14] W. Rawat and Z. Wang, "Deep convolutional neural networks for image classification: a comprehensive review," *Neural Computation*, vol. 29, no. 9, pp. 2352–2449, Sep. 2017, doi: https://doi.org/10.1162/neco_a_00990.

[15] A. Khan, A. Sohail, U. Zahoora, and A. S. Qureshi, "A survey of the recent architectures of deep convolutional neural networks," *Artificial Intelligence Review*, vol. 53, Apr. 2020, doi: https://doi.org/10.1007/s10462-020-09825-6.

[16] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, doi: https://doi.org/10.48550/arXiv.1409.1556

[17] Y. Zhou, H. Chang, Y. Lu, X. Lu, and R. Zhou, "Improving the performance of VGG through different granularity feature combinations," *IEEE Access*, vol. 9, pp. 26208–26220, Jan. 2021, doi: https://doi.org/10.1109/access.2020.3031908.

[18] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, Jun. 2016, doi: https://doi.org/10.1109/cvpr.2016.90.

[19] S. Bianco, R. Cadene, L. Celona, and P. Napoletano, "Benchmark analysis of representative deep neural network architectures," *IEEE Access*, vol. 6, pp. 64270–64277, 2018, doi: https://doi.org/10.1109/access.2018.2877890.

[20] S. Mascarenhas and M. Agarwal, "A comparison between VGG16, VGG19 and ResNet50 architecture frameworks for image classification," *IEEE Xplore*, Nov. 01, 2021. https://ieeexplore.ieee.org/document/9687944

[21] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345–1359, Oct. 2010, doi: https://doi.org/10.1109/tkde.2009.191.

[22] Anass Barodi, Abderrahim Bajit, M. Benbrahim, and A. Tamtaoui, "Improving the transfer learning performances in the classification of the automotive traffic roads signs," vol. 234, pp. 00064–00064, Jan. 2021, doi: https://doi.org/10.1051/e3sconf/202123400064.

[23] R. Castruita Rodríguez, C. Mendoza Carlos, O. O. Vergara Villegas, V. G. Cruz Sánchez, and H. de J. Ochoa Domínguez, "Mexican traffic sign detection and classification using deep learning," *Expert Systems with Applications*, vol. 202, p. 117247, Sep. 2022, doi: https://doi.org/10.1016/j.eswa.2022.117247.

[24] N. Sarhan, M. Lauri, and S. Frintrop, "Multi-phase fine-tuning: a new fine-tuning approach for sign language recognition," *KI - Künstliche Intelligenz*, Feb. 2022, doi: https://doi.org/10.1007/s13218-021-00746-2.

[25] Mr, N. Rao, T. Archana, S. Maram, T. Vishnuvardhan, and S. Suguru, "Traffic light detection and classification using RESNET50." *International Journal of HRM and Organizational Behavior*, 2024. https://ijhrmob.org/index.php/ijhrmob/article/view/180

[26] A. Jaiswal, Deepali, and N. Sachdeva, "Empirical analysis of traffic sign recognition using ResNet architectures," *2023 3rd International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE)*, pp. 280–285, May 2023, doi: https://doi.org/10.1109/icacite57410.2023.10183247.

[27] Z.-Q. Zhao, P. Zheng, S.-T. Xu, and X. Wu, "Object detection with deep learning: a review," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 30, no. 11, pp. 3212–3232, Nov. 2019, doi: https://doi.org/10.1109/tnnls.2018.2876865.

[28] K. G. Lore, A. Akintayo, and S. Sarkar, "LLNet: A deep autoencoder approach to natural low-light image enhancement," *Pattern Recognition*, vol. 61, pp. 650–662, Jan. 2017, doi: https://doi.org/10.1016/j.patcog.2016.06.008.

[29] C. Guo *et al.*, "Zero-reference deep curve estimation for low-light image enhancement," *arXiv.org*, Mar. 22, 2020. https://arxiv.org/abs/2001.06826.

[30] Purbandini, C. Fatichah, and B. Amaliah, "Digital image enhancement using MirNet and Zero-Deep Curve Estimation (Zero-DCE)," *2023 IEEE 7th International Conference on Information Technology, Information Systems and Electrical Engineering (ICITISEE)*, pp. 290–295, Nov. 2023, doi: https://doi.org/10.1109/icitisee58992.2023.10404895.

[31] Z. N. Aldoski and C. Koren, "Traffic sign detection and quality assessment using YOLOv8 in daytime and nighttime conditions," *Sensors*, vol. 25, no. 4, pp. 1027–1027, Feb. 2025, doi: https://doi.org/10.3390/s25041027.

[32] Y. Yan, C. Deng, J. Ma, Y. Wang, and Y. Li, "A traffic sign recognition method under complex illumination conditions," *IEEE access*, vol. 11, pp. 39185–39196, Jan. 2023, doi: https://doi.org/10.1109/access.2023.3266825.

[33] Economic Commission for Europe-Inland Transport Committee, "Convention on road signs and signals," *United Nations Treaty Series*, vol. 1091, no. 3, 1968.

[34] "German traffic sign benchmarks," *benchmark.ini.rub.de*. https://benchmark.ini.rub.de/gtsrb_dataset.html

[35] "Fig. A1. The standard VGG-16 network architecture as proposed in [32]...," *ResearchGate*. https://www.researchgate.net/figure/Fig-A1-The-standard-VGG-16-network-architecture-as-proposed-in-32-Note-that-only_fig3_322512435