

SAAM-ReflectNet: Sign-Aware Attention-Based Multitasking Framework for Integrated Traffic Sign Detection and Retroreflectivity Estimation

Joshua Kofi Asamoah^a (joshua.asamoah@ndsu.edu), Blessing Agyei Kyem^a (blessing.agyeikyem@ndsu.edu), Nathan-David Obeng-Amoako^b(obengamoako.n@northeastern.edu) Armstrong Aboah^a (armstrong.aboah@ndsu.edu)

^a North Dakota State University, 1410 14th Avenue Offerdahl North Building, CIE 201, Fargo-ND, U.S.A

^b Northeastern University, 216 Massachusetts Ave, Boston-MA 02115, U.S.A

Corresponding author at: Full address of the corresponding author, including the country name.

Armstrong Aboah
1410 14th Avenue Offerdahl North Building, CIE 201, Fargo-ND, U.S.A
Tel: +1 (701) 231 4144
Email: armstrong.aboah@ndsu.edu

SAAM-ReflectNet: Sign-Aware Attention-Based Multitasking Framework for Integrated Traffic Sign Detection and Retroreflectivity Estimation

Joshua Kofi Asamoah^a, Blessing Agyei Kyem^a, Nathan-David Obeng-Amoako^b, Armstrong Aboah^{a,*}

^a*North Dakota State University, 1410 14th Avenue Offerdahl North Building, CIE 201, Fargo, U.S.A*

^b*Northeastern University, 216 Massachusetts Ave Boston-MA 02115, U.S.A*

Abstract

Traffic sign retroreflectivity is essential for roadway safety, particularly in low-light and adverse weather conditions. Traditional methods, such as handheld retroreflectometers and nighttime inspections, are labor-intensive, costly, and unsuitable for large-scale implementation. To address these limitations, we developed SAAM-ReflectNet, a deep learning framework that unifies traffic sign detection, classification, and retroreflectivity estimation into a single automated pipeline. Our RetroNet backbone, developed as part of this study, extracts robust spatial and semantic features to enhance feature representation. The Sign-Aware Attention Module we designed prioritizes critical traffic sign regions, improving detection and classification accuracy by focusing on the most relevant areas. Additionally, our multimodal fusion layers seamlessly integrate RGB imagery with LiDAR intensity data, enabling reliable retroreflectivity estimation. ReflectNet achieved a mean Average Precision (mAP) of 0.635 at IoU=0.5 and 0.522 across IoU thresholds from 0.5 to 0.95, alongside Root Mean Squared Errors (RMSE) of 0.169 for foreground and 0.147 for background reflectivity. These findings underscore the reliability, scalability, and transferability of our approach, establishing ReflectNet as a transformative tool for intelligent transportation systems and proactive traffic sign maintenance.

Keywords: Traffic Signs, Retroreflectivity, Computer Vision, Object Detection, Autonomous Vehicles, LiDAR, Transportation Systems

* Armstrong Aboah.

Email addresses: joshua.asamoah@ndsu.edu (Joshua Kofi Asamoah), blessing.agyeikyem@ndsu.edu (Blessing Agyei Kyem), obengamoako.n@northeastern.edu (Nathan-David Obeng-Amoako), armstrong.aboah@ndsu.edu (Armstrong Aboah)

1. Introduction

Traffic signs play a critical role in road safety by providing drivers and autonomous vehicles with essential guidance. Their visibility, particularly in challenging conditions such as nighttime or adverse weather, depends significantly on retroreflectivity, a property that ensures that light is reflected back toward its source (Balali et al. (2015); Jamal et al. (2022); King & Tyrrell (2023)). This is especially important given that night driving, despite accounting for only 20% of total vehicle travel, results in three times more deaths than day driving (Bakutin (2024); Ferenchak & Abadi (2021)). Maintaining high retroreflectivity standards is thus imperative, as mandated by MUTCD Falk (2024). Traditional methods for ensuring retroreflectivity compliance by the MUTCD rely on a sequential two-stage process. First, traffic signs are identified and classified, either manually through field surveys or by basic image processing techniques (Laguna et al. (2014)). Then specialized equipment such as handheld retroreflectometers (Lindly et al. (2002)) or visual inspections (CathySatterfield (2014)) is used to measure retroreflectivity directly in the field. Although effective for localized assessments, these approaches are labor intensive, time-consuming, and prone to data misalignment between detection and retroreflectivity measurement phases, especially for large-scale or frequent assessments (Aldoski & Koren (2024)). Moreover, the reliance on physical retroreflectometers poses logistical challenges and limits scalability. Emerging technologies, such as Light Detection and Ranging (LiDAR), offer promising alternatives to the second stage, by capturing intensity data remotely, which correlates strongly with retroreflectivity. LiDAR has shown significant potential as a scalable, non-invasive alternative for retroreflectivity estimation, allowing for efficient data acquisition over larger areas(Ai & Tsai (2016); Che et al. (2019); He et al. (2023); Mahlberg et al. (2021)). However, its adoption in retroreflectivity assessment has been hindered by the absence of standardized preprocessing methods for large-scale point cloud data. The variability in data formats (Kashani et al. (2015)), noise levels (Ding et al. (2013)), and intensity calibration across LiDAR systems further complicates its integration into streamlined workflows, limiting its accessibility and scalability for widespread use. This lack of standardization, coupled with the absence of frameworks that unify LiDAR retro-intensity with critical visual features such as shape and color, perpetuates a fragmented approach to traffic sign assessment. Consequently, road authorities remain constrained to reactive, high-cost maintenance strategies that fail to fully harness modern

sensing technologies for proactive management. Together, these limitations underscore the pressing need for a scalable and efficient system capable of performing comprehensive traffic sign evaluations in a single step. Addressing these gaps requires a holistic solution capable of standardizing LiDAR preprocessing while simultaneously integrating retroreflectivity estimation with traffic sign detection and classification in a unified framework. To address these needs, we introduce SAAM-ReflectNet, a multitasking deep learning architecture that integrates traffic sign detection, classification, and retroreflectivity estimation into a unified system. ReflectNet is specifically designed to overcome the inefficiencies of traditional two-stage methods while addressing the challenges associated with processing LiDAR data for retroreflectivity estimation. At its core, SAAM-ReflectNet combines advanced feature extraction and multitasking capabilities with a streamlined LiDAR preprocessing pipeline, ensuring compatibility with large-scale, real-world datasets. By leveraging both critical visual attributes and LiDAR retro-intensity data, ReflectNet enables a comprehensive and scalable evaluation of traffic signs, effectively bridging the gap between traditional methods and modern sensing technologies. Our developed architecture incorporates key components tailored to address the challenges of multitasking in a unified system. Central to ReflectNet is the Sign-Aware Attention Module (SAAM), designed to allocate attention dynamically to features most relevant for each task—detection, classification, and retroreflectivity estimation. Unlike conventional multitasking models, which often face trade-offs in task accuracy due to overlapping feature priorities (Crawshaw (2020)), SAAM optimizes feature utilization by distinguishing between visual cues, such as shape and texture, and physical properties derived from Light Detection and Ranging (LiDAR) intensity data. LiDAR’s precise intensity measurements serve as an accurate, non-invasive proxy for retroreflectivity, enabling SAAM-ReflectNet to comprehensively assess both visual and physical characteristics of traffic signs, even in low-light or adverse weather conditions. To further enhance task performance, RetroNet, our customized backbone, addresses the limitations of traditional architectures like ResNet and CSPDarkNet in multitasking scenarios. These standard backbones, while powerful for single-task applications, are not equipped to handle the complex interplay between detection and retroreflectivity estimation. RetroNet introduces specialized refinement layers that isolate and optimize features for each task, reducing cross-task interference. This architecture ensures that both detection and retroreflectivity predictions maintain high precision, creating a balanced and efficient multitasking pipeline. While SAAM-ReflectNet introduces a transformative approach to unify detection and retroreflectivity estimation, it is essential to position its contri-

butions within the broader landscape of object detection. Existing state-of-the-art frameworks, such as YOLO (Redmon et al.), RetinaNet (Lin (2017)), and Faster R-CNN (Ren et al. (2016)), have made significant advancements in detecting and classifying objects based on visual features like shape, color, and texture. These models are optimized for tasks relying solely on visual cues, excelling in their respective domains. However, their inability to assess physical attributes, such as retroreflectivity, highlights a critical limitation—one that ReflectNet is uniquely designed to overcome. By integrating LiDAR’s precise intensity data as a surrogate measure for retroreflectivity, ReflectNet bridges the gap between visual analysis and physical property estimation. This seamless fusion not only enhances operational efficiency but also ensures that critical traffic sign attributes are evaluated holistically. Building on the identified limitations of existing systems and leveraging ReflectNet’s innovative design, this work makes the following key contributions to the field:

1. Propose ReflectNet, a unified framework for traffic sign detection, classification, and retroreflectivity estimation.
2. Introduce the Sign-Aware Attention Module (SAAM) to prioritize task-specific features dynamically, ensuring accurate multitasking performance
3. Design and integrate the RetroNet backbone, featuring specialized refinement layers to minimize task interference and enhance feature extraction
4. Leverage LiDAR intensity data as a scalable, non-invasive surrogate for retroreflectivity measurements, embedding it directly into the detection pipeline for seamless spatial and temporal alignment
5. Introduce a standardized preprocessing pipeline for LiDAR data, addressing challenges in handling large-scale point clouds and enabling seamless integration with deep learning models.

The rest of the paper is structured as follows: Section 2 reviews related work on traditional assessment and multi-tasking frameworks. Section 3 describes the SAAM-ReflectNet framework, including its methodology and retroreflectivity estimation process. Section 4 outlines the experimental setup and evaluation metrics. Section 5 presents the results, highlighting its performance. Finally, Section 6 concludes the study.

2. Related Studies

2.1. Traffic Sign Detection

In recent years, the emergence of deep learning and advanced computer vision techniques have significantly improved the first stage of traffic sign retroreflectivity assessment. These innovations have transformed traffic sign detection into a highly efficient and reliable process, addressing many challenges posed by traditional image processing methods. Among state-of-the-art (SoTA) models, YOLO (Redmon et al.), Faster R-CNN (Ren et al. (2016)), and RetinaNet (Lin (2017)) have emerged as prominent frameworks. YOLO, known for its real-time detection capabilities, has been extensively adapted for traffic sign detection tasks. For example, (Han et al. (2024)) proposed the YOLO-SG model, which incorporates SPD-Conv to mitigate feature loss during down-sampling, while (Mahadshetti et al. (2024)) introduced Sign-YOLO, enhancing detection precision with squeeze-and-excitation (SE) modules. Similarly, (Li et al. (2023)) improved small traffic sign detection accuracy by integrating channel attention mechanisms into YOLOv4's backbone and detection head. In parallel, Faster R-CNN has been effectively applied to traffic sign detection. (Gao et al. (2022); Li et al. (2022); Shao et al. (2019); Zuo et al. (2017)) have showcased its ability to handle complex traffic scenarios, such as overlapping signs and occlusions. In addition to these anchor-based models, recent detection transformer architectures have gained significant traction for their ability to handle multiscale detection tasks with greater precision and efficiency. For example, the TSD-DETR framework offers a lightweight approach to real-time traffic sign detection by addressing small sign sizes and complex backgrounds (Zhang et al. (2025)). Similarly, DSRA-DETR leverages a dilated spatial pyramid pooling model (DSPP) and a multiscale feature residual aggregation module (FRAM) to improve feature aggregation at various scales (Xia et al. (2023)). While the aforementioned SoTA models have revolutionized visual recognition tasks, they remain fundamentally limited to analyzing visual features. These models lack the capability to evaluate critical physical properties, such as retroreflectivity, which are essential for ensuring traffic sign visibility under adverse conditions like nighttime or poor weather. This limitation perpetuates the reliance on a second stage to evaluate retroreflectivity, introducing inefficiencies and challenges that hinder scalable, data-driven management solutions.

2.2. Assessing Traffic Sign Retroreflectivity

The transition from traditional methods, such as visual inspections and retroreflectometers, to non-contact, image-based approaches mark a significant advancement in the second stage of traffic sign retroreflectivity assessment. Unlike traditional methods, which are labor-intensive and limited in scalability, and image-based approaches utilize digital imaging to estimate retroreflectivity by analyzing reflective properties captured in photographs. Early studies by (Siegmann et al. (2008)) demonstrated the potential of correlating image intensity with retroreflectivity but struggled to account for varying illumination angles. (Khalilikhah et al. (2015)) refined these methods using daytime images, achieving strong correlations ($R^2 = 0.87$). However, these methods often fell short under diverse conditions, limiting their reliability for compliance assessments. Automated systems, such as the SMARTS van (Smith & Fletcher (2001)) and RetroView (Carlson & Higgins (2011)), aimed to streamline evaluations but faced challenges with consistency and implementation in real-world scenarios. In addition, European initiatives like VISUALISE (González et al. (2011)) and AMAC (Gutierrez et al. (2012)) introduced promising innovations, but their reliance on nighttime operations and limited validation hindered widespread adoption.

Building on the limitations of image-based methods, researchers have increasingly explored LiDAR technology as a promising alternative for traffic sign retroreflectivity assessment. LiDAR's ability to capture retro-intensity values, which reflect the material properties of objects, offers a scalable, non-contact solution for retroreflectivity evaluation. Early studies demonstrated its effectiveness across transportation applications, including roadway detection (Jiménez & Naranjo (2011)), segmentation and shaped-based classifications of traffic signs (Riveiro et al. (2016)), sight distance analysis (Castro et al. (2011)), and obstacle identification (Tsai et al. (2011, 2013)). Notably, (Ai & Tsai (2016)) pioneered the use of LiDAR retro-intensity data for estimating traffic sign retroreflectivity, achieving over 90% accuracy in compliance assessments. This research highlighted LiDAR's potential to enhance efficiency and safety without requiring physical contact with traffic signs.

Nevertheless, the absence of standardized preprocessing protocols continues to hinder reproducibility and limits the broader development of consistent retroreflectivity assessment models. This gap underscores the need for a robust, multitasking framework capable of unifying data processing with detection, classification, and reflectivity estimation into a single cohesive approach.

2.3. Multitasking Frameworks for Traffic Sign Analysis

Traditional two-stage methods for traffic sign retroreflectivity assessment are inefficient and difficult to scale. Multitasking frameworks provide a promising alternative by integrating detection, classification, and retroreflectivity estimation into a single architecture. Recent multitasking models have demonstrated success in combining related tasks such as object detection, lane segmentation, and depth estimation (Kendall et al. (2018)). Moreover, frameworks like MultiNet (Teichmann et al. (2018)) and CrossStitch Networks (Misra et al. (2016)) exemplify how shared architecture can enhance performance across semantic tasks, while (Uhrig et al. (2016)) highlight the benefits of learning semantic and instance segmentations jointly. These multitasking frameworks highlight the potential for improving efficiency and accuracy across multiple tasks within a single system. Extending these concepts to traffic sign analysis offers an opportunity to address inefficiencies in separate detection and retroreflectivity processes. However, multitasking models face unique challenges when combining tasks with distinct requirements. While detection relies on visual features like shape and color, retroreflectivity estimation depends on material properties, often requiring data from LiDAR. Integrating these tasks requires mechanisms to prioritize task-specific features without degrading overall performance. Recent advancements, such as attention mechanisms like squeeze-and-excitation networks (Hu et al. (2018)) and transformer-based approaches (Vaswani (2017)), have shown promise in effectively focusing on relevant features. Additionally, specialized refinement layers (Standley et al. (2020)) help reduce task interference in multitasking models. However, existing frameworks still face challenges in addressing the distinct demands of integrating visual detection and retroreflectivity estimation into a unified system. ReflectNet addresses these gaps by combining a Sign-Aware Attention Module (SAAM) for task-specific feature prioritization and a customized backbone, RetroNet, to minimize cross-task interference. By leveraging LiDAR retro-intensity data, it achieves comprehensive traffic sign evaluation with improved efficiency. The detailed architecture is explained in Section 3.

3. Methodology

3.1. Problem Formulation

To address the limitations of traditional two-stage methods that separately detect, classify, and then measure retroreflectivity, we propose a unified framework that not only identifies and

categorizes traffic signs but also estimates their retroreflectivity for both the foreground elements and the background surface in a single shot.

Specifically, let D represent a dataset of annotated traffic scenes, where each scene $s \in D$ consists of an image $I_s \in \mathbb{R}^{H \times W \times 3}$ and a corresponding LiDAR point cloud $P_s = \{p_i\}_{i=1}^{N_s}$, with each point $p_i = (x_i, y_i, z_i, R_i)$ including spatial coordinates (x_i, y_i, z_i) and an intensity value R_i serving as a proxy for retroreflectivity.

For each scene s , let $B_s = \{b_j\}_{j=1}^{M_s}$ denote the ground-truth bounding boxes of the traffic signs, $C_s = \{c_j\}_{j=1}^{M_s}$ their respective class labels, and $R_s = \{R_j^*\}_{j=1}^{M_s}$ the known retroreflectivity values corresponding to both foreground and background regions of each sign.

Our objective is to learn a parametric model f_θ that, given (I_s, P_s) , predicts bounding boxes \widehat{B}_s , class labels \widehat{C}_s , and retroreflectivity estimates \widehat{R}_s . To achieve this, we define a composite loss function:

$$L(\theta) = \lambda_{\text{det}} L_{\text{det}}(\widehat{B}_s, B_s) + \lambda_{\text{cls}} L_{\text{cls}}(\widehat{C}_s, C_s) + \lambda_{\text{ret}} L_{\text{ret}}(\widehat{R}_s, R_s), \quad (1)$$

where L_{det} is the bounding box loss, L_{cls} is the classification loss, and L_{ret} measures deviation between predicted and true retroreflectivity values. By selecting nonnegative coefficients $\lambda_{\text{det}}, \lambda_{\text{cls}}, \lambda_{\text{ret}}$ to balance these terms, we solve:

$$\theta^* = \operatorname{argmin}_{\theta} \frac{1}{|D|} \sum_{s \in D} L(\theta), \quad (2)$$

thereby learning a single integrated model that robustly detects and classifies traffic signs, while simultaneously estimating the retroreflectivity of both their foreground and background regions directly from LiDAR intensity data, ultimately streamlining and improving upon conventional workflows.

3.2. Overview of Architecture

The proposed architecture, shown in 1, consists of four key components: the RetroNet backbone, the sign-aware neck, the Sign-Aware Attention Module (SAAM), and task-specific prediction heads. The backbone extracts essential multi-scale features from the input data, serving as the foundation for subsequent processing. Then, to enhance feature representation, the neck refines and aggregates these features, preparing them for task-specific focus. Next, SAAM dynamically allocates attention to features critical for detection, classification, and retroreflectivity estimation,

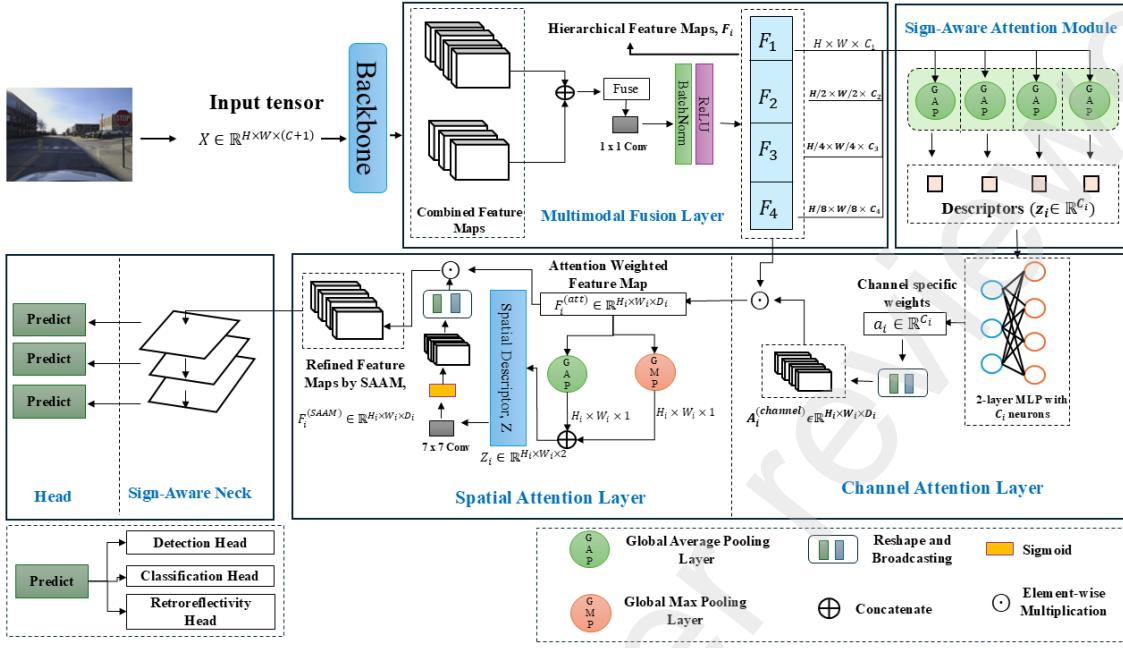


Figure 1: Overall Architecture of SAAM-ReflectNet depicting various components.

ensuring optimal task prioritization. Finally, the task-specific prediction heads utilize these refined features to produce accurate outputs, enabling efficient multitasking in a unified framework.

3.3. RetroNet Backbone

The RetroNet Backbone detailed in 2 is the core feature extraction module of our architecture. It is designed to process multimodal input, specifically, RGB images and corresponding retroreflectivity maps yielding a rich, unified feature representation that captures both visual and reflective properties essential for accurately estimating retroreflectivity and detecting traffic signs.

3.3.1. Input Representations

The input consists of an RGB image $I \in \mathbb{R}^{H \times W \times C}$, where H and W denote the image height and width, and $C = 3$ indicates the three standard RGB channels. In addition, a retroreflectivity map $L \in \mathbb{R}^{H \times W \times 1}$, spatially aligned with I , provides reflectivity information. Concatenating the two along the channel dimension produces a multimodal input:

$$X \in \mathbb{R}^{H \times W \times (C+1)}$$

Incorporating retroreflectivity data from the outset ensures that subsequent layers can exploit correlations between visual patterns and reflectivity without separate calibration or post-processing steps.

3.3.2. Convolution Feature Extraction

The RetroNet Backbone as shown in 2 processes the input tensor $X \in \mathbb{R}^{H \times W \times (C+1)}$, composed of RGB channels and a LiDAR intensity map, to extract progressively abstract features. Drawing inspiration from the Cross-Stage Partial (CSP) Networks, the backbone is designed to enhance gradient flow and reduce redundancy by partitioning feature maps and leveraging residual learning. Starting with $F^{(0)} = X$, the input is passed through a sequence of convolutional layers, each applying a transformation of the form:

$$F^{(l)} = \sigma \left(\text{BN} \left(W^{(l)} * F^{(l-1)} + b^{(l)} \right) \right), \quad (3)$$

where $W^{(l)}$ and $b^{(l)}$ are learnable parameters, $*$ denotes convolution, $\text{BN}(\cdot)$ is batch normalization, and $\sigma(\cdot)$ represents the ReLU activation function.

To refine features and further improve gradient propagation, Progressive Feature Refinement Blocks (PFRBs) are incorporated. Each PFRB splits the input feature map into two paths: the direct path, which reduces channel dimensions via a 1×1 convolution, and the residual path, which employs the Residual Split-Path Aggregation Block (Res-SPAB) for multiscale feature aggregation. The Res-SPAB aggregates features through two parallel paths, each consisting of 1×1 and 3×3 convolutions, followed by concatenation and residual addition:

$$F_{\text{out}} = F_{\text{in}} + \text{Concat} (\text{Conv}_{3 \times 3} (\text{Conv}_{1 \times 1} (F_{\text{in}}))). \quad (4)$$

This residual mechanism ensures the preservation of critical features while enhancing contextual and spatial richness. Downsampling is achieved at specific stages using convolutions with a stride of 2, progressively reducing spatial dimensions while increasing the depth of feature maps. The output of each stage produces hierarchical feature maps $\{F_1, F_2, F_3, F_4\}$ with dimensions:

$$F_i \in \mathbb{R}^{\frac{H}{2^i} \times \frac{W}{2^i} \times C_i}.$$

where $C_i \in \{64, 128, 256, 512\}$.

3.3.3. Multimodal Fusion Layers

The Multimodal Fusion Layers integrate visual and reflective information effectively by combining features extracted from both RGB channels and LiDAR intensity values. At each scale i , the backbone generates two separate feature maps $F_{\text{rgb},i} \in \mathbb{R}^{H_i \times W_i \times D_{\text{rgb},i}}$ and $F_{\text{lidar},i} \in \mathbb{R}^{H_i \times W_i \times D_{\text{lidar},i}}$, representing the features derived from the RGB image and LiDAR intensity map, respectively. These feature maps are concatenated along the channel dimension:

$$F_i = \text{Concat}(F_{\text{rgb},i}, F_{\text{lidar},i}). \quad (5)$$

To ensure efficient integration, the concatenated feature map is passed through a 1×1 convolution followed by batch normalization and a nonlinear activation:

$$F_i = \sigma(\text{BN}(W_{\text{fusion},i} * F_i + b_{\text{fusion},i})), \quad (6)$$

where $W_{\text{fusion},i}$ and $b_{\text{fusion},i}$ are the weights and biases of the convolution layer, and $\sigma(\cdot)$ is the ReLU activation function. This step reduces the dimensionality of F_i to D_i , ensuring computational efficiency while retaining discriminative information. This fusion process is applied hierarchically across all scales i , enabling the backbone to generate a series of fused feature maps $\{F_1, F_2, \dots, F_N\}$, where each $F_i \in \mathbb{R}^{H_i \times W_i \times D_i}$ captures multimodal information at a specific level of abstraction. These fused feature maps are then passed to the Sign-Aware Attention Module (SAAM) for further refinement.

3.4. Sign-Aware Attention Module (SAAM)

The Sign-Aware Attention Module (SAAM), shown in 3, refines the multimodal, hierarchical features obtained from the ReflectNet Backbone by emphasizing channels and spatial regions that are most indicative of traffic signs and their reflective properties. Given an intermediate feature map $F_i \in \mathbb{R}^{H_i \times W_i \times D_i}$ extracted at scale i , SAAM first computes a global descriptor $z_i \in \mathbb{R}^{D_i}$ using global average pooling:

$$z_i = \frac{1}{H_i W_i} \sum_{h=1}^{H_i} \sum_{w=1}^{W_i} F_i(h, w) \in \mathbb{R}^{D_i}. \quad (7)$$

This descriptor captures the overall channel-wise statistics, merging both visual and intensity cues present at this level of abstraction. To determine which channels contribute most to sign detection and retroreflectivity estimation, z_i is passed through a two-layer perceptron with nonlinear

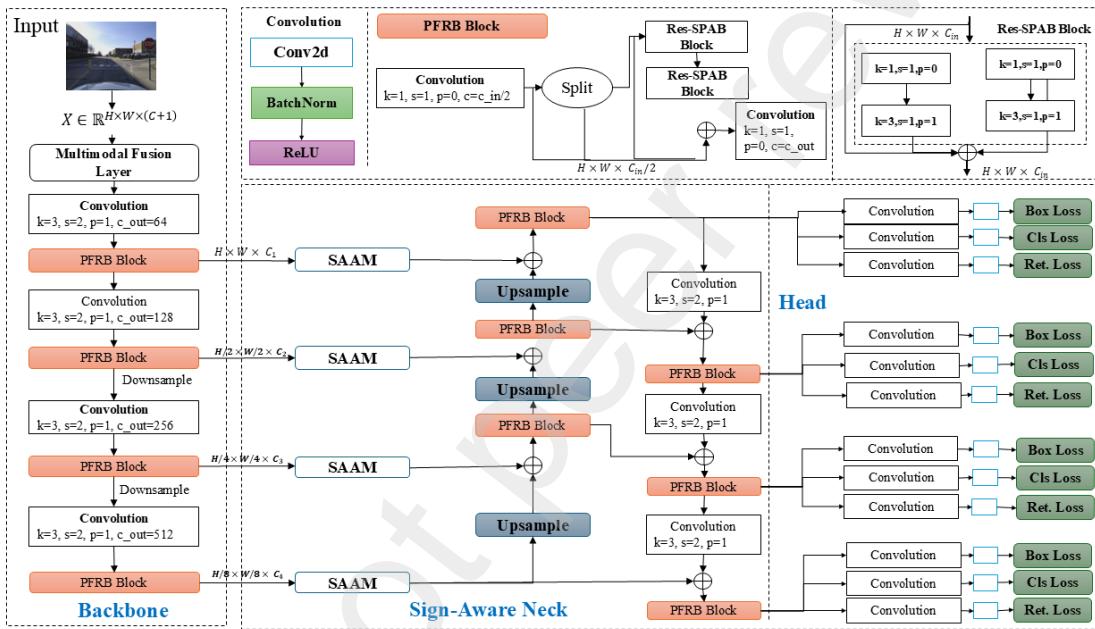


Figure 2: Detailed Architecture of the RetroNet Backbone, SAAM and the Prediction Heads

activations, producing attention weights:

$$a_i = \sigma \left(W_a^{(2)} \delta \left(W_a^{(1)} z_i + b_a^{(1)} \right) + b_a^{(2)} \right) \in [0, 1]^{D_i}, \quad (8)$$

where $\delta(\cdot)$ is a ReLU and $\sigma(\cdot)$ is a sigmoid ensuring bounded outputs. These learned weights highlight channels correlated with sign-specific contours, colors, and reflectivity gradients, while suppressing irrelevant background patterns.

The resulting vector \mathbf{a}_i is reshaped and broadcast spatially to form an attention map $A_c \in \mathbb{R}^{H_i \times W_i \times D_i}$, which is then applied element-wise to the original features.

$$\mathbf{F}_i^{(\text{att},c)} = A_c \odot F_i, \quad (9)$$

where \odot represents element-wise multiplication.

Following channel refinement, spatial attention identifies regions within the feature map most relevant to traffic signs. To compute the spatial attention map, global average pooling (GAP) and max pooling (MP) are applied along the channel dimension, resulting in two spatial descriptors:

$$F_{\text{avg}} = \text{AvgPool}(F_i), \quad (10a)$$

$$F_{\text{max}} = \text{MaxPool}(F_i), \quad (10b)$$

where $F_{\text{avg}}, F_{\text{max}} \in \mathbb{R}^{H_i \times W_i}$. These descriptors capture mean-based and extreme-value-based spatial features, respectively.

The pooled maps are concatenated along the channel dimension and passed through a 7×7 convolutional layer with a sigmoid activation to produce the spatial attention map:

$$A_s = \sigma(\text{Conv2D}(\text{Concat}(F_{\text{avg}}, F_{\text{max}}))) \in \mathbb{R}^{H_i \times W_i}. \quad (11)$$

The spatial attention map is then applied to the channel-refined features through element-wise multiplication:

$$\mathbf{F}_i^{(\text{att})} = A_s \odot \mathbf{F}_i^{(\text{att},c)}, \quad (12)$$

where \odot represents element-wise multiplication. This step ensures the network focuses on spatial regions where traffic signs are likely present, enhancing reflectivity-driven cues and reducing background noise.

The final output of SAAM,

$$\mathbf{F}_i^{(\text{att})},$$

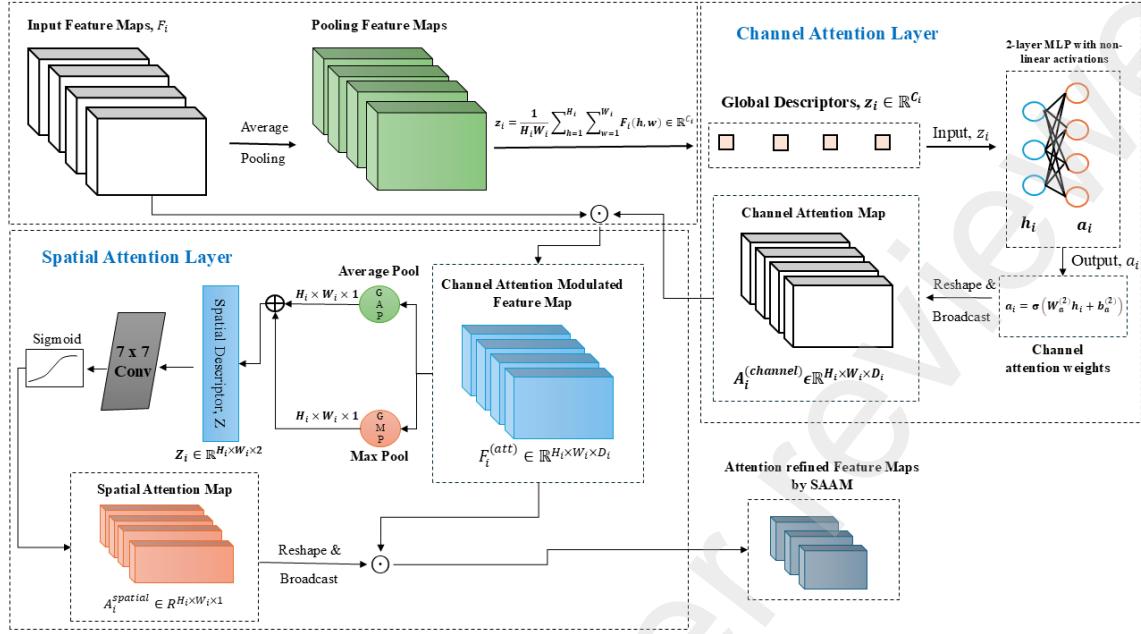


Figure 3: Detailed Architecture of the Sign-Aware Attention Module (SAAM)

represents a feature map where both channel and spatial refinements have been applied. This attention-weighted representation is passed to the Neck for multi-scale feature fusion and refinement, ensuring that the Heads receive highly targeted and interpretable features for detection, classification, and retroreflectivity estimation tasks.

3.5. The Sign-Aware Neck

The Neck integrates and refines the attention-weighted features across multiple scales, inspired by the feature pyramid concept but adapted to preserve intensity-driven information essential for retroreflectivity tasks. From the set $\{F_i^{(\text{att})}\}$, the Neck constructs multi-scale feature maps by combining representations from different levels:

$$G_i = \text{Fuse} \left(F_i^{(\text{att})}, F_{i+1}^{(\text{att})}, \dots \right), \quad (13)$$

where $\text{Fuse}(\cdot)$ includes lateral connections, upsampling layers, and 1×1 convolutions to align dimensions and resolutions. This yields $G_i \in \mathbb{R}^{H'_i \times W'_i \times D'_i}$, combining semantic richness from deeper layers with the spatial detail of shallower ones.

A refinement mapping $\phi(\cdot)$ further enhances the fused features:

$$G'_i = \phi(G_i), \quad (14)$$

where $\phi(\cdot)$ includes BatchNorm and nonlinear activations. This step ensures that each scale’s representation is well-structured, emphasizing edges, intensity cues, and color differences relevant for sign localization and reflectivity estimation.

The output of the Neck is a set of refined, multi-scale feature maps:

$$\{G'_1, G'_2, \dots, G'_M\}.$$

These maps maintain a balance between spatial resolution and semantic depth, providing a robust foundation for the final prediction tasks.

3.6. The Heads

Given the refined, multi-scale feature maps $\{G'_1, G'_2, \dots, G'_M\}$, each $G'_i \in \mathbb{R}^{H'_i \times W'_i \times D'_i}$, the Heads produce task-specific outputs for detection, classification, and retroreflectivity estimation. By applying these prediction layers at multiple scales, the framework captures both fine-grained local information and broader contextual cues.

3.6.1. Detection Heads

The Detection Head aims to localize traffic signs by predicting bounding boxes and associated objectness scores. For each spatial location (h', w') in a given scale G'_i , the head outputs parameters:

$$\{\widehat{x'_i}, \widehat{y'_i}, \widehat{w'_i}, \widehat{h'_i}\}, \widehat{p'_{obj,i}} = h_d(G'_i(h', w')),$$

where $\widehat{x'_i}, \widehat{y'_i}$ denote the relative offsets to the center of a candidate bounding box parameterized with respect to anchors, and $\widehat{w'_i}, \widehat{h'_i}$ specify the normalized width and height of the predicted box. The confidence $\widehat{p'_{obj,i}} \in [0, 1]$ represents the model’s estimated probability that a traffic sign is present at that location.

3.6.2. Classification Head

The Classification Head assigns a category label to each detected sign region. While the Detection Head identifies where a sign is, the Classification Head determines which sign type is present.

For each candidate region (as informed by the detection predictions), the head produces class probabilities:

$$\widehat{p'_{class,i}} = h_c(G'_i(h', w')) \in \mathbb{R}^K, \quad (15)$$

where K is the number of traffic sign classes, and $\widehat{p'_{class,i}}$ is passed through a SoftMax function to yield a normalized probability distribution over all classes. Because the input features G'_i reflect both visual (color, shape) and reflectivity properties, the Classification Head can leverage subtle cues. For instance, certain sign classes may exhibit distinctive reflective patterns under night-time conditions, while others may rely more on color and shape under daylight conditions. The joint training with reflectivity estimation ensures that class predictions are informed by the same integrated cues, reducing confusion among visually similar signs.

3.6.3. Retroreflectivity Head

A primary innovation of this methodology is the direct estimation of a sign’s retroreflective property, providing valuable information for assessing its visibility and maintenance needs. The Retroreflectivity Head provides continuous estimates of sign reflectivity. To capture both the foreground (content) and the background (sheeting) reflectivity properties, it outputs two values:

$$\left(\widehat{r'_{f,i}}, \widehat{r'_{b,i}} \right) = h_r(G'_i(h', w')),$$

where $\widehat{r'_{f,i}} \in \mathbb{R}$ corresponds to the estimated normalized retroreflectivity of the sign’s foreground elements (e.g., symbols, text), and $\widehat{r'_{b,i}} \in \mathbb{R}$ represents the estimated retroreflectivity of the sign’s background sheeting. This regression task is intrinsically linked to the retroreflectivity data integrated at the input stage and refined through the SAAM. By capturing reflectivity-related cues alongside shape and texture features, the Retroreflectivity Head aligns physical attributes (retroreflectivity) with spatial and semantic information, obviating the need for separate handheld measurements or external reflectivity assessments.

3.7. Loss Function and Optimization

The proposed architecture jointly optimizes three primary objectives i.e. detection, classification, and retroreflectivity estimation using a composite loss function designed to balance these interconnected tasks. Each head (detection, classification, and retroreflectivity) contributes a task-specific loss term, and these terms are combined into a single objective function.

3.7.1. Task-Specific Losses

Detection Loss. : To evaluate bounding box regression quality, the framework employs an IoU-based loss that directly optimizes spatial overlap between predicted and ground-truth boxes. Let \hat{b} and b represent the predicted and ground-truth bounding boxes, respectively, and $\text{IoU}(\cdot)$ a function that measures their overlap. A GIoU loss, $\mathcal{L}_{\text{GIoU}}$, is expressed as:

$$\mathcal{L}_{\text{GIoU}} = 1 - \text{IoU}(\hat{b}, b) + \Psi(\hat{b}, b), \quad (16)$$

where $\Psi(\cdot)$ is an additional regularization term ensuring gradient stability.

The detection objective also includes an objectness loss \mathcal{L}_{obj} , typically a binary cross-entropy term evaluating whether a given region corresponds to a traffic sign. Combining these components, the detection loss is:

$$\mathcal{L}_{\text{det}} = \alpha_{\text{GIoU}} \mathcal{L}_{\text{GIoU}} + \alpha_{\text{obj}} \mathcal{L}_{\text{obj}}, \quad (17)$$

with α_{GIoU} and α_{obj} as weighting factors.

Classification Loss. : For classification, the framework uses a categorical cross-entropy loss to compare predicted class probabilities $\widehat{p_{\text{class}}}$ against one-hot ground-truth labels y :

$$\mathcal{L}_{\text{cls}} = - \sum_{k=1}^K y_k \log (\widehat{p_{\text{class},k}}), \quad (18)$$

where K is the number of traffic sign classes, $y_k \in \{0, 1\}$, and $\sum_{k=1}^K y_k = 1$.

This loss encourages the model to accurately identify sign categories, leveraging both visual and reflectivity cues learned during the encoding stages.

Retroreflectivity Loss. : The retroreflectivity head estimates continuous values for foreground r_f and background r_b reflectivity. Let \widehat{r}_f and \widehat{r}_b be the predicted values, and r_f, r_b the ground truths. A mean squared error (MSE) loss is employed:

$$\mathcal{L}_{\text{ret}} = \beta_f \frac{1}{N} \sum_{i=1}^N (\widehat{r}_{f,i} - r_{f,i})^2 + \beta_b \frac{1}{N} \sum_{i=1}^N (\widehat{r}_{b,i} - r_{b,i})^2, \quad (19)$$

where β_f and β_b weight the contributions of the foreground and background reflectivity errors, respectively, and N is the number of sign instances in the batch.

3.7.2. Composite Loss Function

The final training objective is a weighted sum of the three task-specific losses:

$$\mathcal{L} = \lambda_{\text{det}} \mathcal{L}_{\text{det}} + \lambda_{\text{cls}} \mathcal{L}_{\text{cls}} + \lambda_{\text{ret}} \mathcal{L}_{\text{ret}}, \quad (20)$$

where λ_{det} , λ_{cls} , λ_{ret} control the relative importance of detection, classification, and retroreflectivity tasks. Selecting these weighting factors is guided by validation performance to ensure that no single task dominates optimization. A grid search was also employed to find an appropriate balance.

3.7.3. Optimization Procedure

The network parameters Θ (encompassing the RetroNet Backbone, SAAM, Neck, and Heads) are optimized by minimizing \mathcal{L} :

$$\Theta^* = \arg \min_{\Theta} \mathcal{L}(\Theta). \quad (21)$$

A stochastic gradient-based method, i.e., Adam, is used for optimization. Given a mini-batch of samples, gradients $\nabla_{\Theta} \mathcal{L}$ are computed via backpropagation, and parameters are updated iteratively:

$$\Theta \leftarrow \Theta - \eta \nabla_{\Theta} \mathcal{L},$$

where η is the learning rate. Learning rate schedulers (e.g., step decay, cosine annealing) and early stopping criteria based on validation metrics are used to stabilize training and prevent overfitting.

4. Experiments

The experiments are designed to evaluate the proposed framework across detection, classification, and retroreflectivity estimation tasks. This section describes the dataset, the preprocessing pipeline from data collection to annotation, the evaluation metrics, and the training protocols employed.

4.1. Dataset

The dataset used in this research was collected by (Zhang et al. (2024)) and comprises raw data in the form of LiDAR point clouds and a video log, captured using a vehicle-mounted Lidar sensing platform at the University of Missouri, MO, USA. The raw data were processed and adapted for this study, resulting in approximately 12,000 synchronized RGB images (1080p resolution) and LiDAR intensity maps. Each image was annotated with bounding boxes, class labels, and retroreflectivity

values for both the foreground (sign content) and background (reflective sheeting) regions. The dataset includes eight traffic sign classes defined in accordance with MUTCD guidelines, ensuring a representative set of sign types and reflective materials. For this study, the data were partitioned into training (70%), validation (15%), and testing (15%) subsets, maintaining balanced distributions across classes. The preprocessing steps, including synchronization of LiDAR and video data, sensor calibration, intensity normalization, and retroreflectivity computation, were conducted as detailed in Section 4.2.

4.2. Data Preprocessing

The preprocessing pipeline as shown in Figure 4 prepares the multimodal sensor data for subsequent detection, classification, and retroreflectivity estimation tasks. It comprises the following primary stages: (1) extraction of image frames and LiDAR point clouds from the raw sensor data, (2) detection of frames containing traffic signs using a custom detection algorithm, (3) synchronization of detected image frames with corresponding LiDAR scans based on timestamps, (4) calibration of sensor data, (5) normalization of LiDAR intensity values, and (6) computation of retroreflectivity measurements.

4.2.1. Data Synchronization

Initially, the video-log images and LiDAR point clouds were extracted from the raw camera-LiDAR bag files. A total of 15,000 image frames and 20,000 LiDAR scans were obtained from the 4-hour drive. To identify frames containing traffic signs, a custom detection algorithm was employed, achieving a detection success rate of 90%. Manual intervention was conducted to select additional frames that were initially missed by the detection algorithm.

The synchronization process pairs each detected image frame $I(t_i)$ with the nearest LiDAR scan $P(t_j)$ based on their timestamps t_i and t_j . The objective is to minimize the temporal difference $|t_i - t_j|$ while ensuring it remains within a predefined threshold Δt . Formally, the synchronization mapping $\phi : t_i \mapsto t_j$ is defined as:

$$\phi(t_i) = \arg \min_{t_j} |t_i - t_j|, \quad (22)$$

subject to:

$$|t_i - t_j| \leq \Delta t.$$

The algorithm utilized for data synchronization is detailed in Table 1.

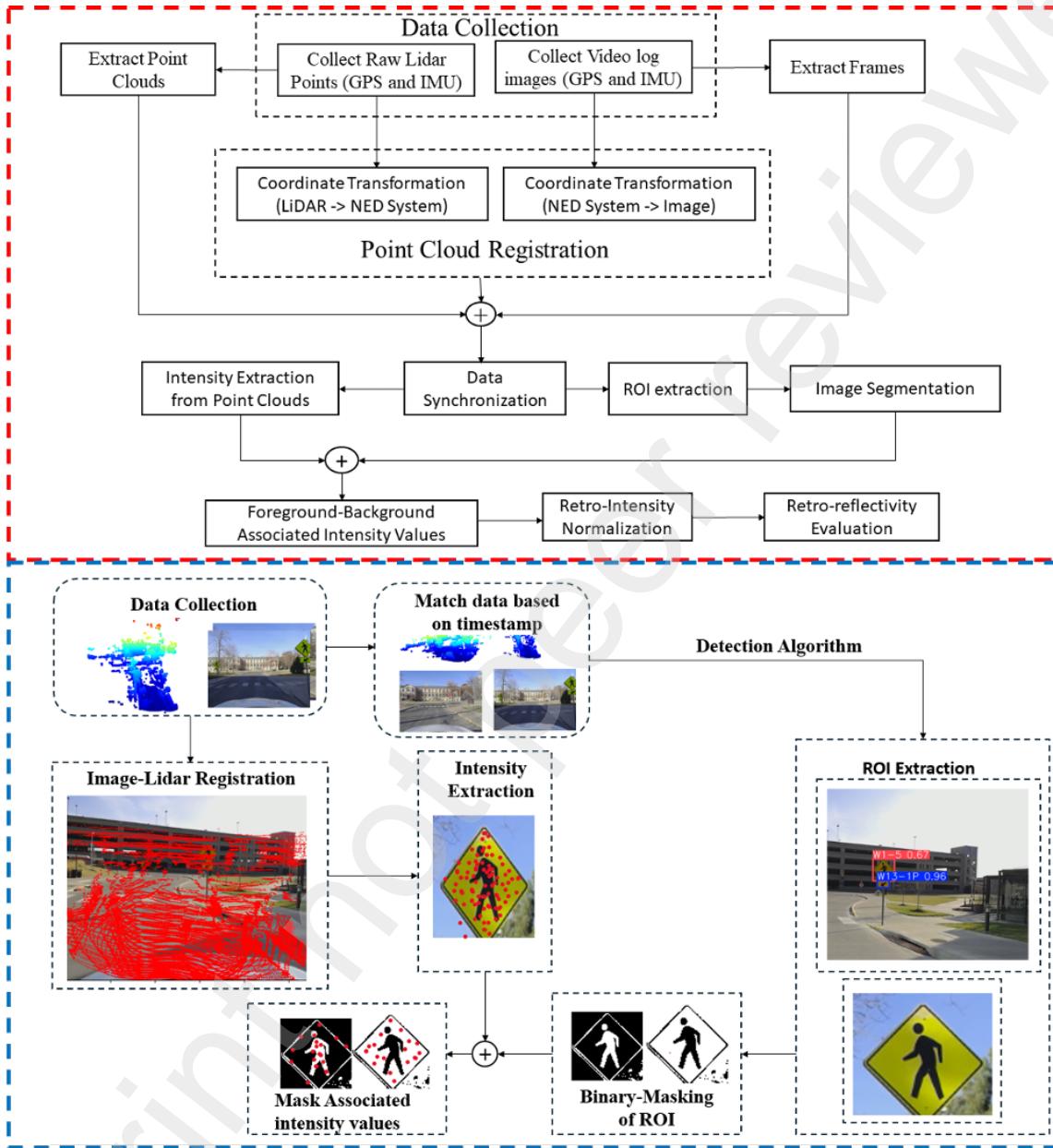


Figure 4: Our methodological preprocessing framework for the collected lidar and camera data

Table 1: Algorithm table for timestamp synchronization of frames and point clouds.

Step	Description
Input:	
<ul style="list-style-type: none"> • $\mathcal{I} = \{(t_i)\}_{i=1}^{N_I}$: Set of detected image frames with timestamps t_i. • $\mathcal{P} = \{(t_j)\}_{j=1}^{N_P}$: Set of LiDAR point clouds with timestamps t_j. • Δt: Maximum allowable time offset for synchronization. 	
Output:	
<ul style="list-style-type: none"> • $\mathcal{S} = \{(I(t_i), P(t_j))\}_{i=1}^{N_S}$: Set of synchronized image-LiDAR pairs. 	
Procedure:	
<ol style="list-style-type: none"> 1. Initialize an empty set $\mathcal{S} = \emptyset$. 2. For each image frame $I(t_i) \in \mathcal{I}$: <ol style="list-style-type: none"> (a) Find the LiDAR scan $P(t_j) \in \mathcal{P}$ that minimizes the temporal difference: $\phi(t_i) = \arg \min_{t_j} t_i - t_j$ (b) If $t_i - t_j \leq \Delta t$: <ul style="list-style-type: none"> • Add the synchronized pair $(I(t_i), P(t_j))$ to the set \mathcal{S}. (c) If $\mathcal{S} = N_S$: <ul style="list-style-type: none"> • Break the loop. 3. End For. 4. Return the synchronized set $\mathcal{S} = \{(I(t_i), P(t_j))\}$. 	

4.2.2. Traffic Sign ROI Extraction and Segmentation

For each detected traffic sign in an image frame $I(t_i)$, a bounding box $\mathcal{B} = (x, y, w, h)$ is obtained, where x and y represent the top-left corner coordinates of the bounding box, and w and h denote the width and height of the box, respectively. The bounding box defines the ROI $R \subset I$ that

encapsulates the traffic sign:

$$\mathbf{R} = \mathbf{I}(x : x + w, y : y + h). \quad (23)$$

The ROI extraction ensures that only the traffic sign region is processed further, reducing computational overhead and improving segmentation accuracy.

To segment the extracted ROI, \mathbf{R} , into foreground (traffic sign content) and background (sheeting), the Otsu thresholding method (Otsu et al. (1975)) is applied to the grayscale intensity values of the ROI. The Otsu method determines an optimal threshold T that minimizes the intra-class variance between two-pixel distributions: foreground (F) and background (B). The ROI \mathbf{R} is first converted to a grayscale image \mathbf{R}_g to simplify intensity segmentation:

$$\mathbf{R}_g = \text{Gray}(\mathbf{R}). \quad (24)$$

Then, Otsu's method computes the optimal threshold T that separates the intensity histogram into two classes. The threshold T is defined as:

$$T = \arg \min_{\tau} [w_F(\tau) \sigma_F^2(\tau) + w_B(\tau) \sigma_B^2(\tau)], \quad (25)$$

where:

- $w_F(\tau)$ and $w_B(\tau)$ are the weights (pixel proportions) of the foreground and background classes at threshold τ .
- $\sigma_F^2(\tau)$ and $\sigma_B^2(\tau)$ are the variances of the foreground and background pixel intensities, respectively.

Once T is determined, the binary segmentation of \mathbf{R}_g into foreground and background is performed as:

$$\mathbf{S}(p) = \begin{cases} 1, & \text{if } \mathbf{R}_g(p) \geq T, \\ 0, & \text{otherwise,} \end{cases} \quad (26)$$

where p represents a pixel in the ROI, \mathbf{S} is the resulting binary mask, 1 denotes the foreground, and 0 denotes the background.

4.2.3. Camera-LiDAR Projection

The LiDAR points are initially represented in the LiDAR coordinate system $\{L\}$, while the camera images reside in the camera coordinate system $\{C\}$. To align these modalities, a known set

of extrinsic parameters \mathbf{R} and \mathbf{t} (rotation and translation) is used to transform the LiDAR points from $\{L\}$ to the camera coordinate system $\{C\}$:

$$\begin{bmatrix} X_C \\ Y_C \\ Z_C \end{bmatrix} = \mathbf{R} \begin{bmatrix} X_L \\ Y_L \\ Z_L \end{bmatrix} + \mathbf{t}, \quad (27)$$

where (X_L, Y_L, Z_L) are coordinates of a LiDAR point in $\{L\}$, and (X_C, Y_C, Z_C) are the transformed coordinates in $\{C\}$.

Once points are expressed in the camera coordinate system, the intrinsic camera parameters (focal lengths f_x, f_y , and principal point offsets c_x, c_y) are applied to project the 3D coordinates onto the 2D image plane:

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \frac{X_C}{Z_C} \\ \frac{Y_C}{Z_C} \\ 1 \end{bmatrix}, \quad (28)$$

where (u, v) are the pixel coordinates corresponding to the 3D point (X_C, Y_C, Z_C) projected onto the 2D image.

Only points with $Z_C > 0$ and within the camera's field of view are considered valid for projection. Each valid LiDAR point thus maps to a specific pixel location (u, v) in the image. This association aligns retroreflectivity information carried by the LiDAR intensity data with the spatial and spectral details captured by the camera.

4.2.4. Retro-intensity Normalization

Laser-based retro-intensity ρ (i.e., the ratio of returned energy to emitted energy) is known to vary with beam distance D (Aytac & Barshan (2005)) and incidence angle θ (Ai & Tsai (2016); Wang & Glenn (2009)). If uncorrected, these dependencies can cause significant errors when evaluating traffic-sign reflectivity.

In this study, we adopt an empirical calibration approach (Voegtle & Wakaluk (2009)) to account for the dependence of retro-intensity ρ on beam distance D and incidence angle θ . First, we define a simplified model in which $\rho(\theta, D)$ is the product of two functions: $f(\theta)$, reflecting the angular dependence, and $g(D)$, capturing how intensity changes with distance. Drawing on prior work (Kamerman (1993); Foley (1996); Janecek & Moses (2008), we assume (proof in Appendix):

$$f(\theta) = [(1 - k_s(\theta)) \cos(\theta) + k_s(\theta)], \quad (29a)$$

and

$$g(D) = D^\alpha, \quad (29b)$$

where $k_s(\theta)$ can be expressed as a second-order polynomial:

$$k_s(\theta) = a + b\theta + c\theta^2 \quad (\text{for Type 1 engineer grade signs}) . \quad (30)$$

To determine the parameters $\{a, b, c, \alpha\}$, we performed two laboratory experiments. First, we conducted a distance test by placing the LiDAR sensor at multiple fixed distances from a stationary traffic sign while keeping $\theta \approx 0^\circ$. By plotting the measured retro-intensity against the distance shown in Figure 5a and fitting the resulting curve, we obtained α .

Next, we executed an incidence-angle test by holding the sensor distance constant and rotating the traffic sign from near 0° up to approximately 70° . Regressing the measured intensities against θ (see Figure 5b) allowed us to fit the polynomial in $k_s(\theta)$. The final model:

$$\rho(\theta, D) = [(1 - (a + b\theta + c\theta^2)) \cos(\theta) + (a + b\theta + c\theta^2)] \cdot D^\alpha, \quad (31)$$

provides the basis for normalizing field measurements so that distance and angle effects are mitigated, yielding consistent estimates of traffic-sign retro-intensity.

With the parameters $\{a, b, c, \alpha\}$ calibrated, we proceed to estimate the normalized retro-intensity for each LiDAR point associated with detected traffic signs. To achieve this, we first calculate the beam distance D and the incidence angle θ for each LiDAR point. Using the associated LiDAR point cloud, the normal direction \vec{n} of the traffic sign surface is obtained through Principal Component Analysis (PCA). The direction of the laser beam \vec{p} is determined from the known sensor position and the position of each point in the LiDAR point cloud.

The incidence angle θ between the laser beam and the surface normal is calculated using the following equation:

$$\theta = \arccos\left(\frac{\vec{p} \cdot \vec{n}}{\|\vec{p}\| \|\vec{n}\|}\right), \quad (32)$$

where \vec{p} represents the direction vector of the laser beam, \vec{n} is the surface normal vector of the traffic sign, and $\vec{p} \cdot \vec{n}$ denotes their dot product. The beam distance D is directly derived from the LiDAR point data.

Once the incidence angle θ and the beam distance D are obtained for each LiDAR point, the retro-intensity values are normalized using the calibrated model:

$$\rho_{\text{norm}}(\theta, D) = \frac{\rho_{\text{raw}}(\theta, D)}{[(1 - k_s(\theta)) \cos(\theta) + k_s(\theta)] \cdot D^\alpha}, \quad (33)$$

where $\rho_{\text{raw}}(\theta, D)$ is the measured retro-intensity value, and the denominator incorporates the angular and distance corrections derived from the earlier calibration steps.

4.2.5. Retroreflectivity Estimation

Retroreflectivity (R) is a critical parameter for assessing the visibility of traffic signs, and its estimation from normalized retro-intensity (ρ_{norm}) requires a well-defined relationship. Based on studies with similar experimental setups and materials (Ai & Tsai (2016)), we adopt a linear regression model expressed as:

$$R_{\text{EST}} = -285.9 + 392.3 \cdot \rho_{\text{norm}}. \quad (34)$$

The relationship was derived from prior research that established a strong correlation between normalized LiDAR retro-intensity and retroreflectivity for retroreflective materials commonly used in traffic signs. Given the similarity in traffic sign materials and LiDAR configurations in this study, the model provides a reliable framework to estimate retroreflectivity for each LiDAR point using the normalized retro-intensity values obtained in Section 4.2.4. To ensure consistent comparison and prevent the influence of outliers, the retroreflectivity assessment for each traffic sign considers the median value of retroreflectivity separately for the foreground and background.

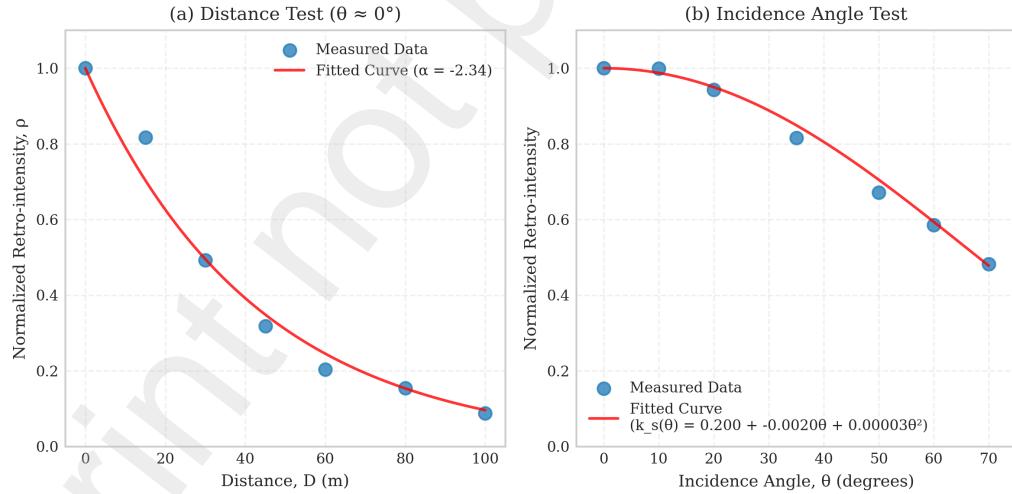


Figure 5: Calibration Curves for retro-intensity. (a) variation with beam distance D ; (b) variation with incidence angle

4.3. Evaluation Metrics

To comprehensively evaluate the performance of SAAM-ReflectNet, three sets of metrics were used, corresponding to the detection, classification, and retroreflectivity estimation tasks.

4.3.1. Detection Metrics

The performance of the detection task was evaluated using the Intersection over Union (IoU), which measures the overlap between predicted bounding boxes and ground truth bounding boxes. IoU is defined as:

$$\text{IoU} = \frac{\text{Area of Overlap}}{\text{Area of Union}} = \frac{B_p \cap B_g}{B_p \cup B_g}, \quad (35)$$

where B_p is the predicted bounding box, B_g is the ground truth bounding box, and \cap and \cup denote intersection and union, respectively. A prediction is considered a match if its IoU exceeds a predefined threshold (0.5 or 0.95).

Based on IoU, the Mean Average Precision (mAP) was computed at thresholds of 0.5 and 0.95 to assess the accuracy of the model's localization capabilities. Additionally, Precision and Recall were used to evaluate the detection performance:

$$P = \frac{\text{TP}}{\text{TP} + \text{FP}}, \quad (36a)$$

$$R = \frac{\text{TP}}{\text{TP} + \text{FN}}, \quad (36b)$$

where TP, FP, and FN represent the true positives, false positives, and false negatives, respectively.

4.3.2. Classification Metrics

For the classification task, Accuracy and the F1-Score were used. Accuracy measures the proportion of correctly classified traffic signs relative to the total number of samples, while the F1-Score provides a balanced evaluation by combining Precision and Recall:

$$\text{F1-Score} = 2 \cdot \frac{P \cdot R}{P + R}. \quad (37)$$

4.3.3. Retroreflectivity Estimation Metrics

The retroreflectivity regression task was evaluated using Mean Absolute Error (MAE) and Root Mean Square Error (RMSE). The MAE and RMSE are defined as:

$$\text{MAE} = \frac{1}{N} \sum_{i=1}^N |\hat{r}_i - r_i|, \quad (38\text{a})$$

$$\text{RMSE} = \sqrt{\frac{1}{N} \sum_{i=1}^N (\hat{r}_i - r_i)^2}, \quad (38\text{b})$$

where \hat{r}_i is the predicted retroreflectivity, r_i is the ground truth value, and N is the total number of samples.

4.4. Implementation Details

4.4.1. Hyper-parameter Configuration

The hyperparameters used in this study were selected through a systematic grid search approach to achieve optimal model performance. The best hyper-parameter combinations are shown in Table 2.

4.4.2. Training Setting and Procedure

The weights of the RetroNet Backbone, the Sign-Aware Attention Module (SAAM), the Neck, and the task-specific Heads were initialized using He initialization to ensure effective weight scaling for efficient learning.

During training, data augmentation techniques, including random horizontal flipping, random cropping to resolution, brightness and contrast adjustments within $\pm 30\%$, and Gaussian noise injection (mean 0, standard deviation 0.01), were applied to enhance generalization.

The model was then fine-tuned with a batch size of 32, optimized using the Adam optimizer with an initial learning rate of 1×10^{-4} , which was reduced by a factor of 0.1 every 50 epochs.

The composite loss function balanced detection, classification, and retroreflectivity estimation with weights $\lambda_{\text{det}} = 1.0$, $\lambda_{\text{cls}} = 1.0$, and $\lambda_{\text{ret}} = 1.0$. Training was conducted for a maximum of 1000 epochs, with early stopping applied if the validation loss did not improve for 100 consecutive epochs. Gradient clipping with a norm threshold of 5.0 stabilized the training, and the model checkpoint with the lowest validation loss was saved for final evaluation.

The entire training was conducted using PyTorch on an NVIDIA RTX 4090 GPU with 24 GB of VRAM.

Table 2: Best Hyperparameter combination for the SAAM-ReflectNet Architecture.

Hyperparameter	Value
Optimizer	Adam
Initial Learning Rate	1×10^{-4}
Learning Rate Decay	0.1 every 50 epochs
Batch Size	32
Weight Decay	1×10^{-5}
Maximum Epochs	1000
Early Stopping Patience	100 epochs

5. Results

The results presented in this section evaluate the performance of the proposed SAAM-ReflectNet framework in detecting, classifying, and estimating the retroreflectivity of traffic signs. The evaluation is based on a multimodal dataset of synchronized RGB images and LiDAR intensity maps, described in Section 4.1.

Metrics such as mean Average Precision (mAP), F1-score, and Root Mean Squared Error (RMSE) were utilized to assess the performance of detection, classification, and retroreflectivity estimation tasks, respectively. Qualitative results and ablation studies are included to validate the contribution of key architectural components.

5.1. Performance of SAAM-ReflectNet

The performance of the proposed ReflectNet architecture was evaluated across three primary tasks: detection, classification, and retroreflectivity regression. Table 3 provides a detailed comparative analysis between ReflectNet and state-of-the-art YOLO model variants for detection and classification tasks. As YOLO models lack retroreflectivity regression capabilities, this unique aspect remains an exclusive contribution of SAAM-ReflectNet.

In the detection task, ReflectNet demonstrated superior localization performance, achieving a mean Average Precision (mAP) of 0.635 at IoU=0.5 (mAP@0.5) and 0.522 at mAP@[0.5:0.95]. These metrics outperformed all compared YOLO variants, including YOLOv10-x (mAP@[0.5:0.95] = 0.462) and YOLOv11-x (mAP@[0.5:0.95] = 0.399), as summarized in 3. SAAM-ReflectNet’s performance gain is attributed to the RetroNet backbone and the integration of SAAM, which

Table 3: Performance Comparison Between ReflectNet and YOLO Model Variants for Detection and Classification Tasks.

Model	Backbone	mAP@0.5	mAP@0.5:0.95
YOLOv8n	CSP-Darknet	0.598	0.441
YOLOv8s	CSP-Darknet	0.525	0.400
YOLOv8x	CSP-Darknet	0.550	0.452
YOLOv10n	Enhanced CSPnet	0.583	0.416
YOLOv10s	Enhanced CSPnet	0.545	0.396
YOLOv10x	Enhanced CSPnet	0.550	0.462
YOLOv11n	SPPF + C2PSA	0.549	0.405
YOLOv11s	SPPF + C2PSA	0.552	0.395
YOLOv11x	SPPF + C2PSA	0.535	0.399
ReflectNet	RetroNet	0.635	0.522

enhances the model’s ability to focus on spatially relevant regions. Figure 6 further illustrates ReflectNet’s effectiveness, showcasing its robustness in localizing small-scale signs like R1-1 under cluttered and complex environments

For the classification task, SAAM-ReflectNet exhibited robust performance, particularly in handling visually similar classes. The normalized confusion matrix in Figure 7 highlights classification rates exceeding 0.98 for critical classes such as S1-1 and R1-1, surpassing YOLO variants that struggled with these subtle distinctions. Furthermore, the F1-confidence curves in Figure 8 further validate ReflectNet’s classification strength, showing F1 scores consistently exceeding 0.8 across most classes at higher confidence thresholds. This improvement is enabled by ReflectNet’s multi-modal fusion layer and Progressive Feature Refinement Blocks (PFRBs), which capture intricate intra-class variations even under adverse conditions, such as poor lighting and motion blur. SAAM-ReflectNet’s retroreflectivity regression task sets it apart from existing models.

Our architecture demonstrated robust performance across all traffic sign classes, achieving consistently low error metrics in retroreflectivity predictions, as summarized in Table 4. However, a few notable discrepancies were observed for specific signs. For instance, the R10-5 and W13-3 classes exhibited higher RMSE values, reaching 0.504 and 0.662, respectively. These outliers can be attributed to unique challenges, such as high variability in retroreflective materials and environ-



Figure 6: Representative outputs from ReflectNet showcasing detection, classification, and retroreflectivity predictions for various traffic sign classes. Each image displays the detected traffic sign with predicted retroreflectivity values for both foreground (FG) and background (BG)

mental conditions for R10-5 or partial occlusions for W13-3. These factors likely led to slight over- or under-predictions in retroreflectivity values. Despite these challenges, ReflectNet maintained a high level of accuracy overall, with average RMSE values of 0.169 for foreground and 0.147 for background retroreflectivity.

To demonstrate ReflectNet’s performance in retroreflectivity regression, Figure 6 presents result for six distinct traffic signs, including their predicted foreground (FG) and background (BG) retroreflectivity values. These predictions align closely with ground truth measurements, as validated in Table 4. Notably, YOLO models lack retroreflectivity prediction mechanisms, reinforcing ReflectNet’s unique contribution to the domain. Figure 9 presents a comprehensive comparison between predicted and ground truth retroreflectivity values for various traffic signs, illustrating ReflectNet’s effectiveness in accurately modeling reflective properties critical for traffic signs assessment. The bar charts depict the predicted retroreflectivity values for both foreground and

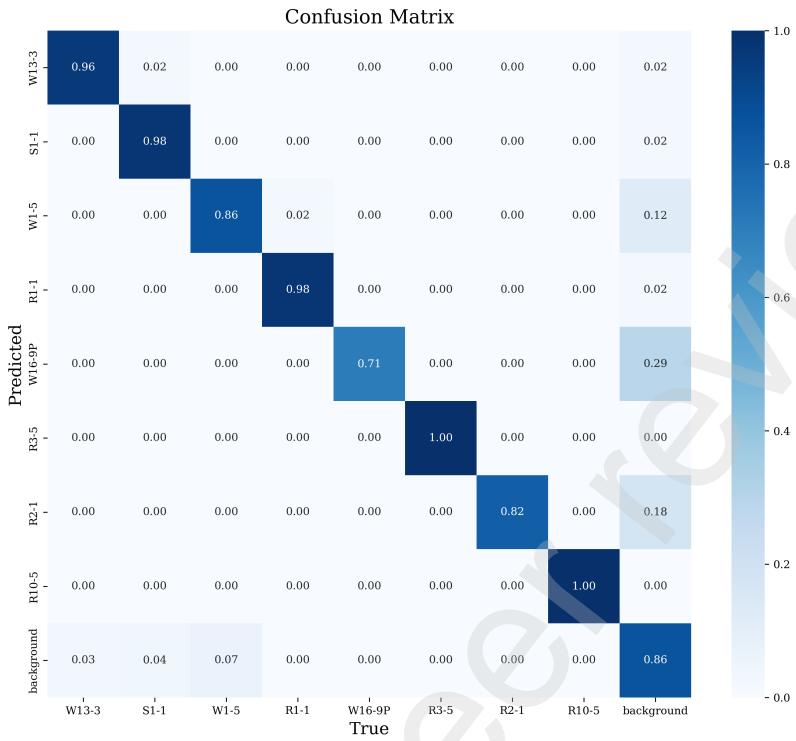


Figure 7: Confusion matrix illustrating ReflectNet’s classification performance across the evaluated MUTCD traffic sign classes.

Table 4: Performance metrics for retroreflectivity regression across different traffic sign classes.

Traffic Sign Class	FG RMSE	BG RMSE	FG MAE	BG MAE
R1-1	0.020	0.032	0.017	0.028
S1-1	0.025	0.034	0.021	0.039
W16-9P	0.050	0.047	0.042	0.032
R3-5	0.018	0.022	0.010	0.019
W1-5	0.033	0.026	0.029	0.020
R10-5	0.504	0.458	0.485	0.430
R2-1	0.036	0.033	0.022	0.025
W13-3	0.662	0.526	0.572	0.500
Average	0.169	0.147	0.150	0.137

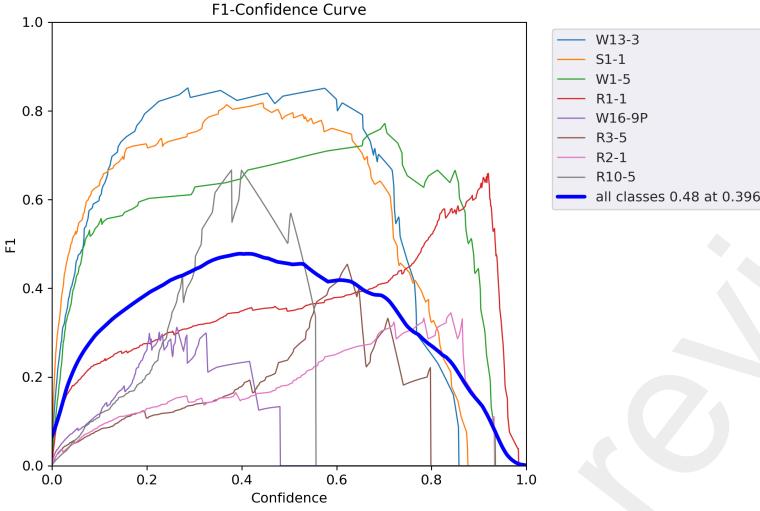


Figure 8: F1-confidence curve illustrating ReflectNet’s performance across varying confidence threshold

background regions alongside their ground truth counterparts, highlighting ReflectNet’s capability to align its predictions with real-world measurements across diverse sign classes and environmental conditions.

The training loss trends depicted in Figure 10 demonstrate ReflectNet’s stable convergence across all tasks. The box loss, object loss, and classification loss steadily decreased converging by 1000. Similarly, retroreflectivity losses for both foreground and background regions exhibited consistent reductions, reflecting the model’s efficiency in multitask optimization. These trends validate the effectiveness of ReflectNet’s architecture in achieving balanced learning across detection, classification, and regression tasks.

While ReflectNet generally outperformed YOLO models, certain challenging cases involving extreme occlusion or visually indistinct features led to minor misclassifications or missed detections. For instance, as shown in Figure 11, image350 includes the detection of an ‘R3-5’ sign with a low confidence of 0.4, which is notable because this sign was not part of the training dataset, indicating a false positive. Additionally, image2511 highlights a detection error where a ‘W13-3’ sign is erroneously detected in the sky, reflecting a failure to properly associate the detection with valid traffic sign regions. These challenges emphasize the need for further refinement of ReflectNet’s localization and classification mechanisms to minimize false positives and improve robustness in

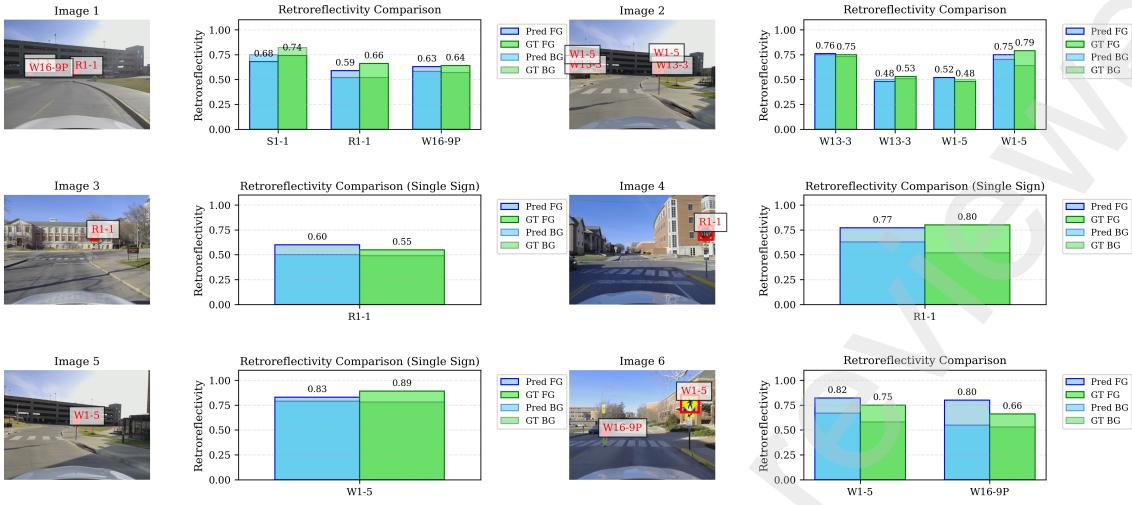


Figure 9: Retroreflectivity comparison for detected traffic signs. Images (left) show detected traffic signs, while corresponding bar plots (right) present predicted and ground truth retroreflectivity values for both foreground (FG) and background (BG) across multiple science and scenarios.

complex environments

5.2. SAAM Attention Map Visualization

The attention maps in Figure 12 highlight how the Sign-Aware Attention Module (SAAM) prioritizes traffic sign regions while suppressing background clutter. The first column shows detected traffic signs in complex environments, while the second column illustrates spatial attention maps, where brighter regions indicate areas of focus. The third column presents channel-weighted feature maps, emphasizing key features critical for detection, classification, and retroreflectivity regression. Finally, the fourth column displays weight distributions, showcasing SAAM’s adaptability across varying conditions. These visualizations demonstrate SAAM’s effectiveness in enhancing ReflectNet’s performance in challenging real-world scenarios.

5.3. Ablation Studies

To evaluate the contributions of ReflectNet’s architectural components, a series of ablation studies were conducted. These experiments systematically analyzed the impact of key features, including

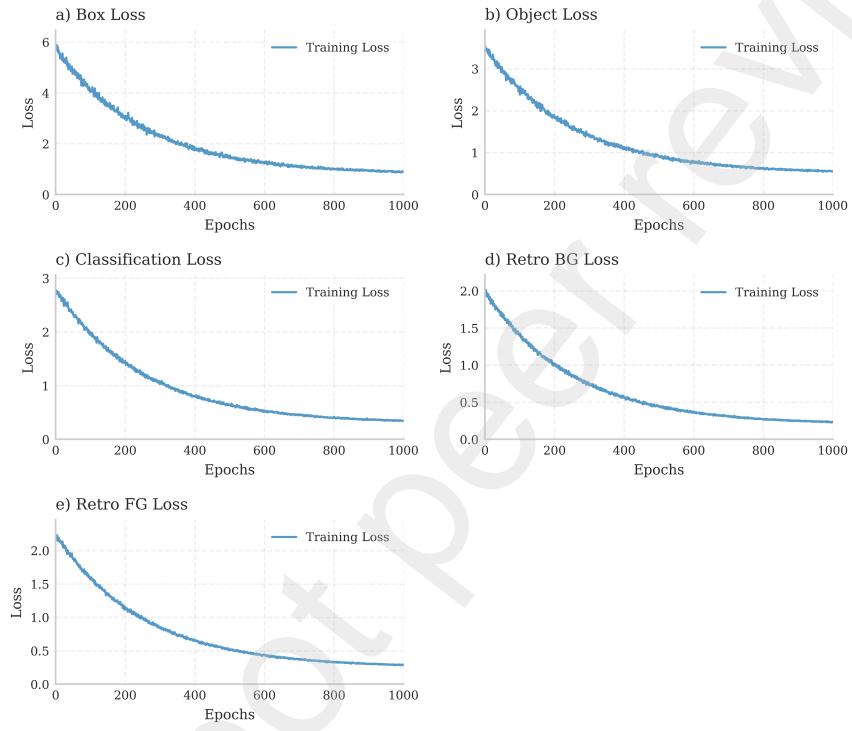


Figure 10: Training loss curves for different components of the model: (a) box loss, (b) object loss, (c) classification loss, (d) retroreflectivity background (Retro BG) loss, and (e) retroreflectivity foreground (Retro FG) loss

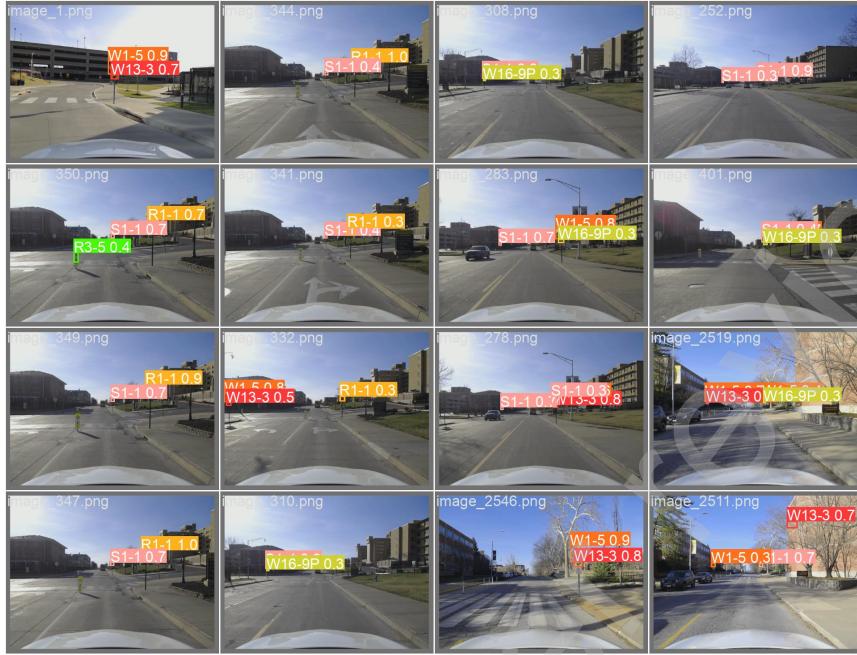


Figure 11: Sample Detection Output from SAAM-ReflectNet with Misclassifications

the RetroNet backbone with Progressive Feature Refinement Blocks (PFRBs), the SAAM module, the Multimodal Fusion Layer, and the retroreflectivity regression heads. By incrementally removing or modifying each component, the analysis quantified its specific role in detection, classification, and retroreflectivity regression tasks. Table 5 summarizes the performance of these configurations, quantifying their impact on detection, classification, and retroreflectivity regression tasks.

The RetroNet backbone, integrated with PFRBs, was shown to be essential for achieving competitive performance. By refining spatial and semantic features across scales, the PFRBs ensured the preservation of high-level contextual information alongside fine-grained details. When the RetroNet backbone was replaced with CSP-Darknet (YOLOv8x) or Enhanced CSPNet (YOLOv10x), significant drops in performance were observed. Specifically, the mAP@0.5 declined from 0.635 (RetroNet) to 0.550 for both YOLOv8x and YOLOv10x backbones.

Similarly, retroreflectivity regression accuracy deteriorated, with the RMSE for foreground retroreflectivity increasing from 0.169 to 0.312 and for background retroreflectivity from 0.147 to 0.381. Despite their lower computational cost (180.4 GFLOPs and 185.6 GFLOPs, respectively), these alternative backbones failed to match the superior performance of RetroNet, which operates

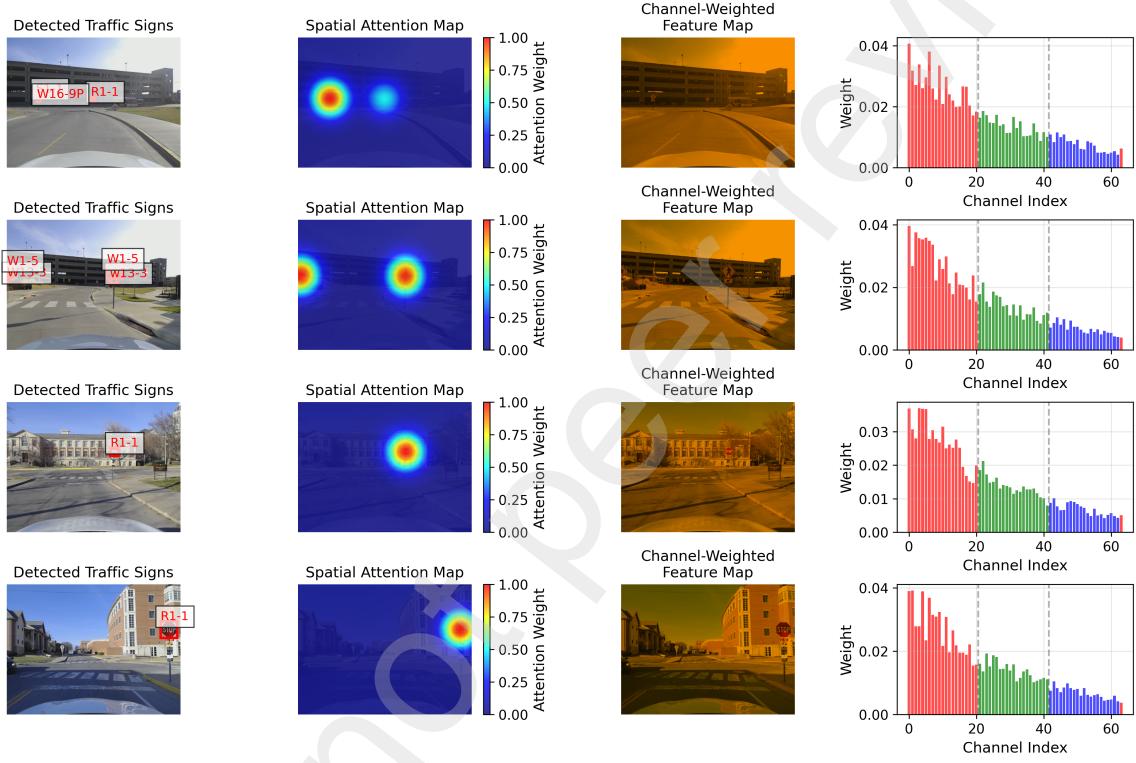


Figure 12: Output of SAAM with Detected traffic signs with corresponding spatial attention maps, channel-weighted feature maps, and channel attention distributions.

at 203.9 GFLOPs with 86.2 million parameters, as shown in Table 5.

The SAAM demonstrated significant improvements in both detection and classification. By prioritizing spatially relevant regions, particularly for smaller signs in cluttered scenes, SAAM enhanced ReflectNet’s localization and classification capabilities. When the SAAM was removed, the mAP@[0.5:0.95] dropped from 0.522 to 0.462, and classification confusion increased for similar classes like R1-1 and S1-1. These results underscore SAAM’s role in improving ReflectNet’s ability to handle complex environments. These findings highlight the importance of multimodal fusion in leveraging complementary information to improve performance across all tasks.

Table 5: Performance of SAAM-ReflectNet and its Variants.

Configuration	RetroNet	Sign-Aware	Multimodal	mAP	mAP	FG	BG	Params	GFLOPs
	Backbone	Attention	Fusion	@*0.5	@0.5	RMSE	RMSE	(M)	
:95									
Baseline	✓	✗	✗	0.568	0.415	1.435	1.600	70.3	184.5
Baseline + Fusion	✓	✗	✓	0.598	0.462	0.312	0.381	85.5	200.3
Baseline + SAAM	✓	✓	✗	0.570	0.470	1.215	1.300	74.0	188.9
YOLOv8x-Backbone	✗	✓	✓	0.550	0.452	0.312	0.381	68.4	180.4
YOLOv10x-Backbone	✗	✓	✓	0.550	0.462	0.312	0.381	72.3	185.6
ReflectNet (Ours)	✓	✓	✓	0.635	0.522	0.169	0.147	86.2	203.9

6. Conclusion

SAAM-ReflectNet represents a groundbreaking framework for traffic sign assessment, unifying detection, classification, and retroreflectivity regression into a single automated system. At the core of its success is the RetroNet backbone, which, along with Progressive Feature Refinement Blocks (PFRBs), refined spatial and semantic features across scales, achieving state-of-the-art performance while maintaining computational efficiency with 86.2 million parameters and 203.9 GFLOPs. The integration of Multimodal Fusion Layers significantly improved retroreflectivity regression by leveraging complementary RGB and LiDAR intensity data, achieving low RMSE values (foreground: 0.169, background: 0.147). The Sign-Aware Attention Module (SAAM) enhanced localization and classification accuracy by focusing on relevant regions, particularly small-scale or visually similar signs, ensuring robust performance in complex environments. Despite these advancements, ReflectNet currently relies on normalized retroreflectivity values, limiting direct comparison with established thresholds. Future work should incorporate conversion to actual retroreflectivity values

for comprehensive compliance assessments. Additionally, expanding ReflectNet's application to domains such as pedestrian detection or road surface quality assessment, and improving interpretability through visual explanations, could enhance its broader utility and adoption in safety-critical applications.

CRediT authorship contribution statement

First Author: Conceptualization, Methodology, Writing – original. **Second Author:** Methodology, Writing – review & editing. **Third Author:** writing-review and editing. **Last Author:** Supervision, Validation.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This research was supported by the Economic Diversifying Research Fund (EDRF) under the grant number (FAR0037938).

References

- Ai, C., & Tsai, Y. J. (2016). An automated sign retroreflectivity condition evaluation methodology using mobile lidar and computer vision. *Transportation Research Part C: Emerging Technologies*, 63, 96–113. doi:10.1016/J.TRC.2015.12.002.
- Aldoski, Z. N., & Koren, C. (2024). Assessment of traffic sign retroreflectivity for autonomous vehicles: A comparison between handheld retroreflectometer and lidar data. *Archives of Transport*, 70, 7–26. doi:10.61089/aot2024.qxy24g93.
- Aytac, T., & Barshan, B. (2005). Surface differentiation by parametric modeling of infrared intensity scans. *Optical Engineering*, 44, 67202. URL: <https://doi.org/10.1117/1.1931467>. doi:10.1117/1.1931467.

- Bakutin, Y. (2024). Visibility and safety of vehicle traffic in the dark (current reality, future projections). *Visegrad Journal on Human Rights*, (pp. 13–19). URL: <https://journals.uran.ua/journal-vjhr/article/view/312793>. doi:10.61345/1339-7915.2024.3.2.
- Balali, V., Sadeghi, M. A., & Golparvar-Fard, M. (2015). Image-based retro-reflectivity measurement of traffic signs in day time. *Advanced Engineering Informatics*, 29, 1028–1040. doi:10.1016/J.AEI.2015.08.003.
- Carlson, P. J., & Higgins, L. (2011). Research and recommendations for a statewide sign retroreflectivity maintenance program 5. report date project 0-6408 13. type of report and period covered project performed in cooperation with the texas department of transportation and the federal highway administration. project title: Development of a statewide sign maintenance program unclassified.
- Castro, M., Iglesias, L., Sánchez, J. A., & Ambrosio, L. (2011). Sight distance analysis of highways using gis tools. *Transportation Research Part C: Emerging Technologies*, 19, 997–1005. doi:10.1016/j.trc.2011.05.012.
- CathySatterfield (2014). Methods for maintaining pavement marking retroreflectivity.
- Che, E., Olsen, M. J., Parrish, C. E., Jung, J., & Peng, S. (2019). Pavement marking retroreflectivity estimation and evaluation using mobile lidar data. *Photogrammetric Engineering and Remote Sensing*, 85, 573–583. doi:10.14358/PERS.85.8.573.
- Crawshaw, M. (2020). Multi-task learning with deep neural networks: A survey, . URL: <http://arxiv.org/abs/2009.09796>.
- Ding, Q., Chen, W., King, B., Liu, Y., & Liu, G. (2013). Combination of overlap-driven adjustment and phong model for lidar intensity correction. *ISPRS Journal of Photogrammetry and Remote Sensing*, 75, 40–47. doi:10.1016/J.ISPRSJPRS.2012.09.015.
- Falk, K. (2024). The 11th edition of the mutcd-what it means to you. *Institute of Transportation Engineers. ITE Journal*, 94, 18–21.
- Ferenchak, N. N., & Abadi, M. G. (2021). Nighttime pedestrian fatalities: A comprehensive examination of infrastructure, user, vehicle, and situational factors. *Journal of Safety Research*, 79, 14–25. doi:10.1016/J.JSR.2021.07.002.

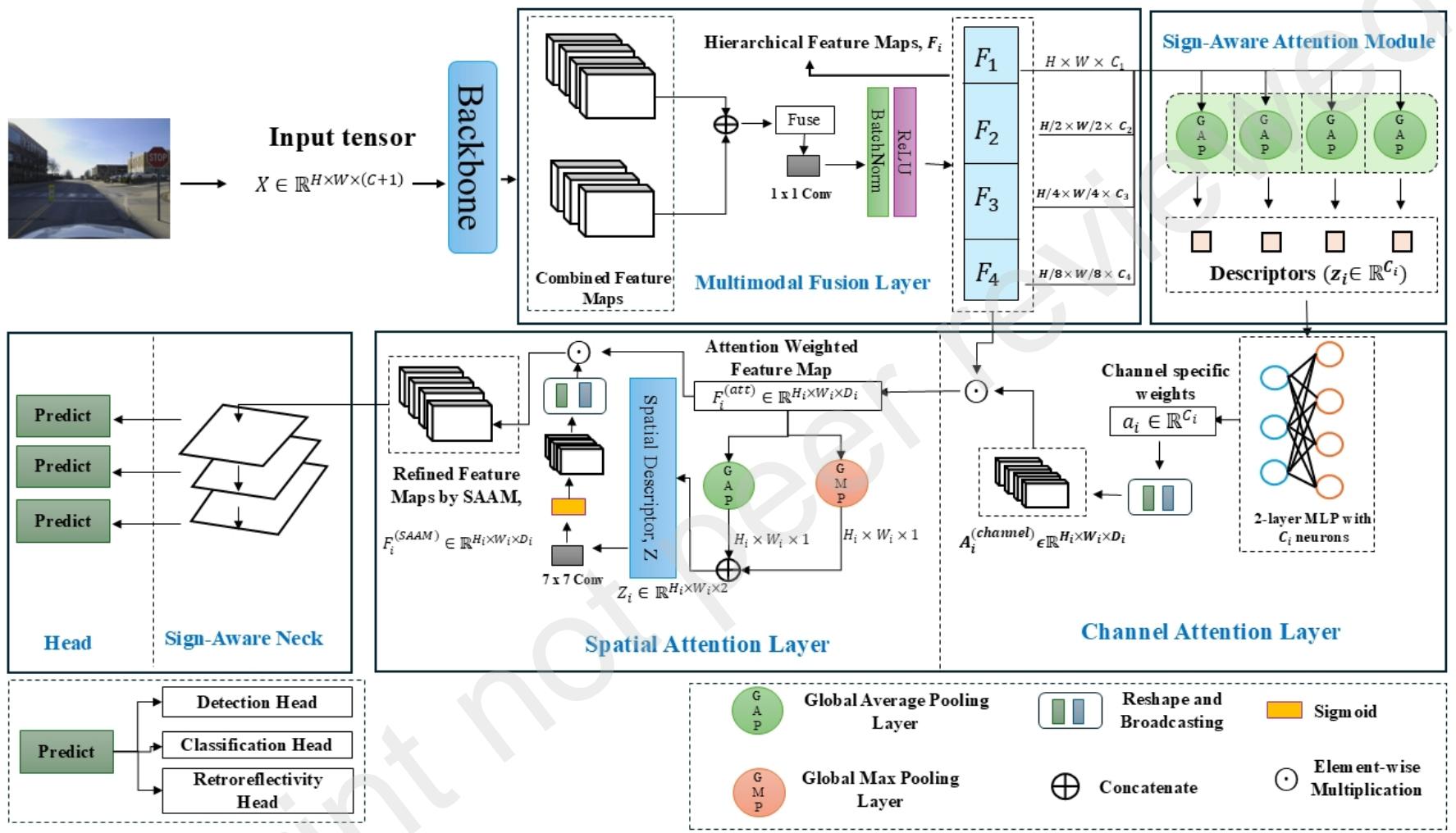
- Foley, J. D. (1996). *Computer graphics: principles and practice* volume 12110. Addison-Wesley Professional.
- Gao, X., Chen, L., Wang, K., Xiong, X., Wang, H., & Li, Y. (2022). Improved traffic sign detection algorithm based on faster r-cnn. *Applied Sciences (Switzerland)*, 12. doi:10.3390/app12188948.
- González, Á., Garrido, M. Á., Llorca, D. F., Gavilán, M., Fernández, J. P., Alcantarilla, P. F., Parra, I., Herranz, F., Bergasa, L. M., Sotelo, M. Á. et al. (2011). Automatic traffic signs and panels inspection system using computer vision. *IEEE Transactions on intelligent transportation systems*, 12, 485–499.
- Gutierrez, J. A., Lejarazu, D. O. D., Real, J. A., Mansilla, A., & Vizmanos, J. (2012). Dynamic measurement of traffic signluminance as perceived by a driver. *Lighting Research and Technology*, 44, 350–363. doi:10.1177/1477153511420049.
- Han, Y., Wang, F., Wang, W., Li, X., & Zhang, J. (2024). Yolo-sg: Small traffic signs detection method in complex scene. *Journal of Supercomputing*, 80, 2025–2046. doi:10.1007/s11227-023-05547-y.
- He, H., Xu, A., Han, X., Wang, H., Wang, L., & Su, W. (2023). Lidar perception and evaluation method for road traffic marking retroreflection. *Transportation Research Record*, 2677, 258–279. doi:10.1177/03611981221145135.
- Hu, J., Shen, L., & Sun, G. (2018). Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 7132–7141).
- Jamal, A., Reza, I., & Shafullah, M. (2022). Modeling retroreflectivity degradation of traffic signs using artificial neural networks. *IATSS Research*, 46, 499–514. doi:10.1016/J.IATSSR.2022.08.003.
- Janecek, M., & Moses, W. W. (2008). Optical reflectance measurements for commonly used reflectors. *IEEE transactions on nuclear science*, 55, 2432–2437.
- Jiménez, F., & Naranjo, J. E. (2011). Improving the obstacle detection and identification algorithms of a laserscanner-based collision avoidance system. *Transportation research part C: emerging technologies*, 19, 658–672.

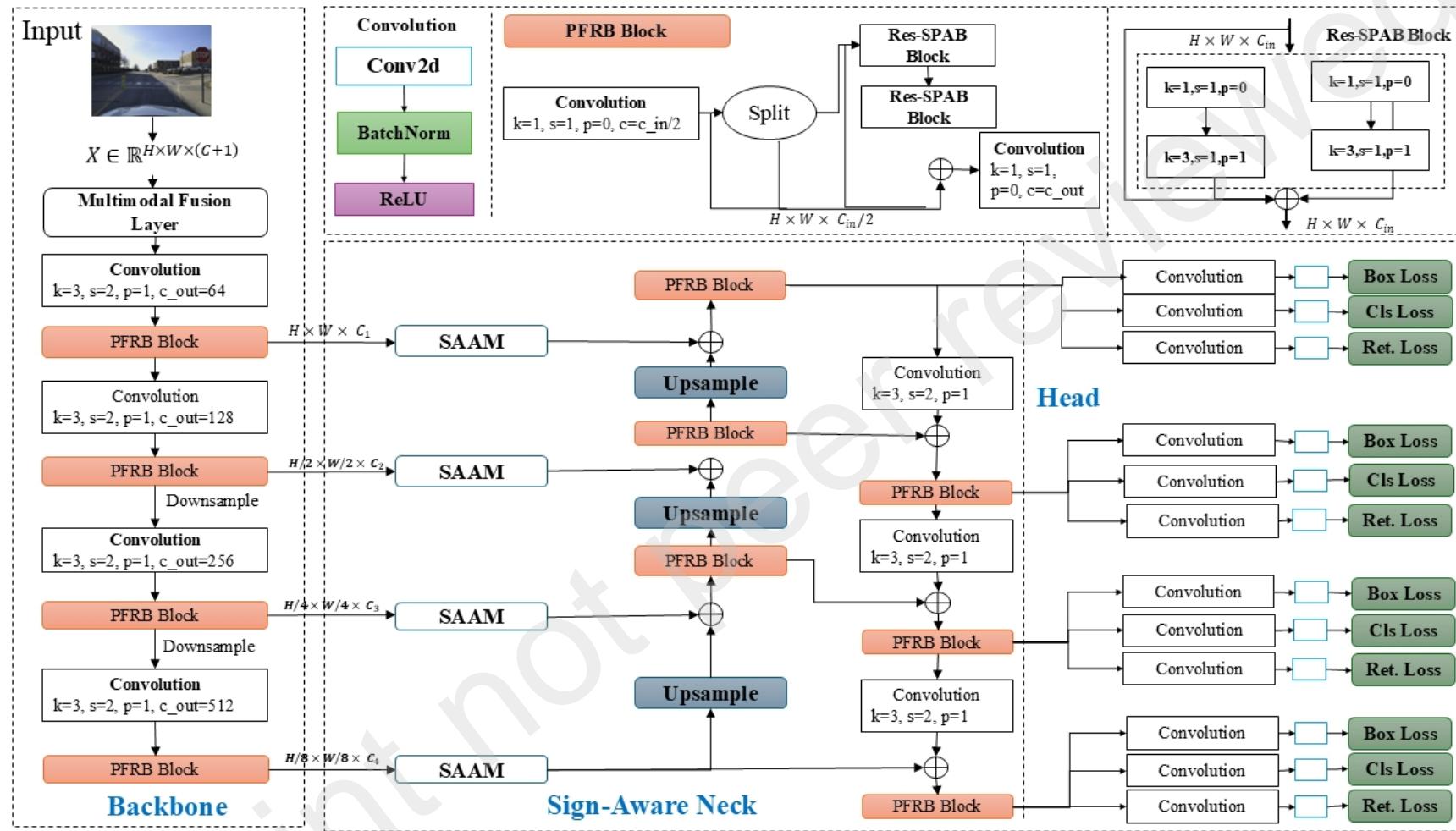
- Kameran, G. W. (1993). Active electro-optical systems. *The infrared and electro-optical system handbook*, 6.
- Kashani, A. G., Olsen, M. J., Parrish, C. E., & Wilson, N. (2015). A review of lidar radiometric processing: From ad hoc intensity correction to rigorous radiometric calibration. doi:10.3390/s151128099.
- Kendall, A., Gal, Y., & Cipolla, R. (2018). Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 7482–7491).
- Khalilikhah, M., Heaslip, K., & Song, Z. (2015). Can daytime digital imaging be used for traffic sign retroreflectivity compliance? *Measurement: Journal of the International Measurement Confederation*, 75, 147–160. doi:10.1016/j.measurement.2015.07.049.
- King, S. L., & Tyrrell, R. A. (2023). Demonstrations of retroreflectivity help road users appreciate retroreflective markings. In *Proceedings of the Human Factors and Ergonomics Society* (pp. 1228–1229). SAGE Publications Inc. volume 67. doi:10.1177/21695067231192524.
- Laguna, R., Barrientos, R., Blázquez, L. F., & Miguel, L. J. (2014). Traffic sign recognition application based on image processing techniques. *IFAC Proceedings Volumes*, 47, 104–109. doi:10.3182/20140824-6-ZA-1003.00693.
- Li, X., Xie, Z., Deng, X., Wu, Y., & Pi, Y. (2022). Traffic sign detection based on improved faster r-cnn for autonomous driving. *Journal of Supercomputing*, 78, 7982–8002. doi:10.1007/s11227-021-04230-4.
- Li, Y., Li, J., & Meng, P. (2023). Attention-yolov4: a real-time and high-accurate traffic sign detection algorithm. *Multimedia Tools and Applications*, 82, 7567–7582. doi:10.1007/s11042-022-13251-x.
- Lin, T. (2017). Focal loss for dense object detection. *arXiv preprint arXiv:1708.02002*, .
- Lindly, J. K., Yellapu, K., & Supriyasilp, T. (2002). Evaluation of retroreflectometers for the alabama department of transportation.

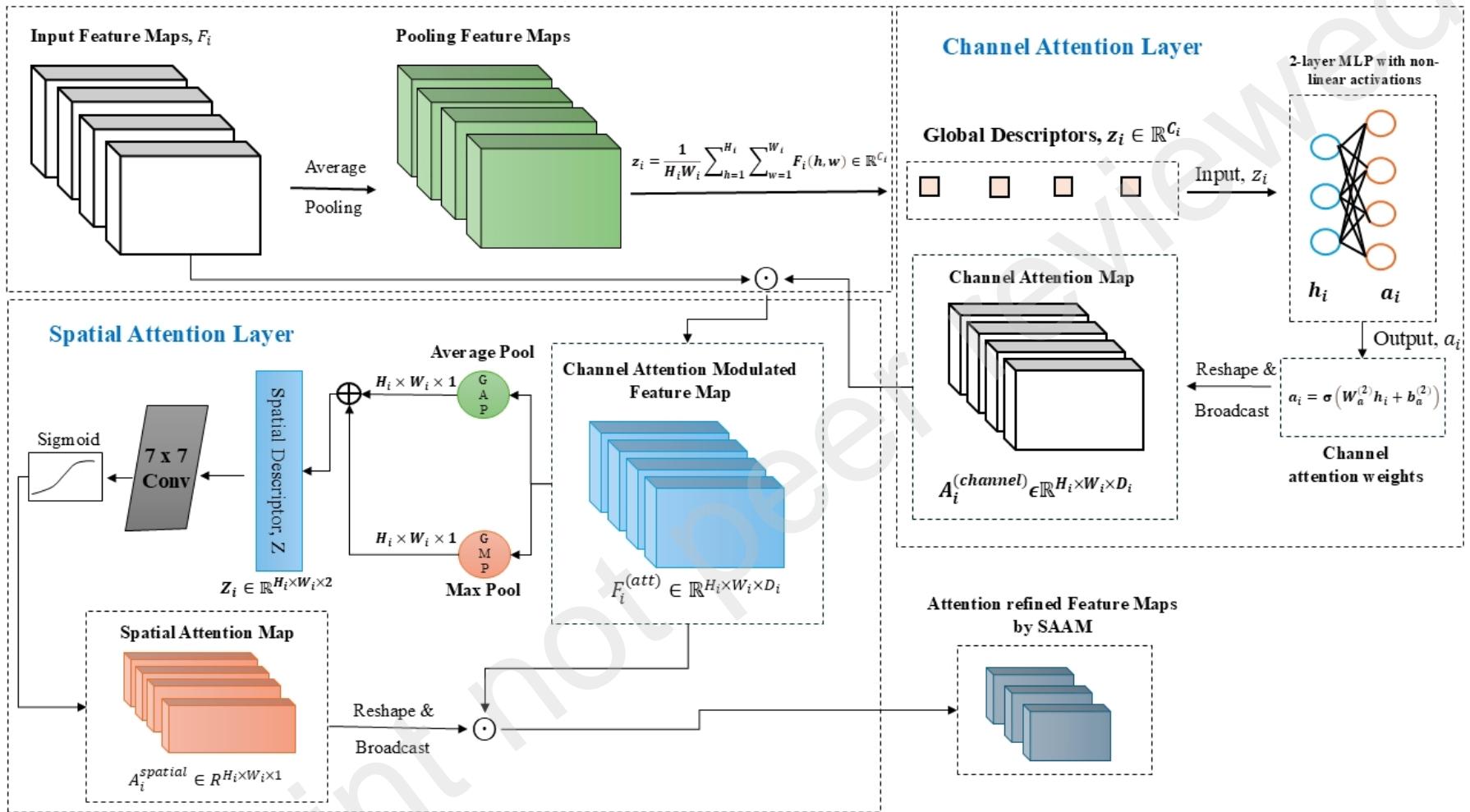
- Mahadshetti, R., Kim, J., & Um, T. W. (2024). Sign-yolo: Traffic sign detection using attention-based yolov7. *IEEE Access*, . doi:10.1109/ACCESS.2024.3417023.
- Mahlberg, J. A., Cheng, Y. T., Bullock, D. M., & Habib, A. (2021). Leveraging lidar intensity to evaluate roadway pavement markings. *Future Transportation*, 1, 720–736. doi:10.3390/futuretransp1030039.
- Misra, I., Shrivastava, A., Gupta, A., & Hebert, M. (2016). Cross-stitch networks for multi-task learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 3994–4003).
- Otsu, N. et al. (1975). A threshold selection method from gray-level histograms. *Automatica*, 11, 23–27.
- Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (). You only look once: Unified, real-time object detection. URL: <http://pjreddie.com/yolo/>.
- Ren, S., He, K., Girshick, R., & Sun, J. (2016). Faster r-cnn: Towards real-time object detection with region proposal networks. *IEEE transactions on pattern analysis and machine intelligence*, 39, 1137–1149.
- Riveiro, B., Diaz-Vilarino, L., Conde-Carnero, B., Soilan, M., & Arias, P. (2016). Automatic segmentation and shape-based classification of retro-reflective traffic signs from mobile lidar data. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 9, 295–303. doi:10.1109/JSTARS.2015.2461680.
- Shao, F., Wang, X., Meng, F., Zhu, J., Wang, D., & Dai, J. (2019). Improved faster r-cnn traffic sign detection based on a second region of interest and highly possible regions proposal network. *Sensors (Switzerland)*, 19. doi:10.3390/s19102288.
- Siegmund, P., Lopez-Sastre, R. J., Gil-Jimenez, P., Lafuente-Arroyo, S., & Maldonado-Bascon, S. (2008). Fundamentals in luminance and retroreflectivity measurements of vertical traffic signs using a color digital camera. *IEEE Transactions on Instrumentation and Measurement*, 57, 607–615. doi:10.1109/TIM.2007.911643.
- Smith, K., & Fletcher, A. (2001). Evaluation of the fhwa's sign management and retroreflectivity tracking system (smarts) van.

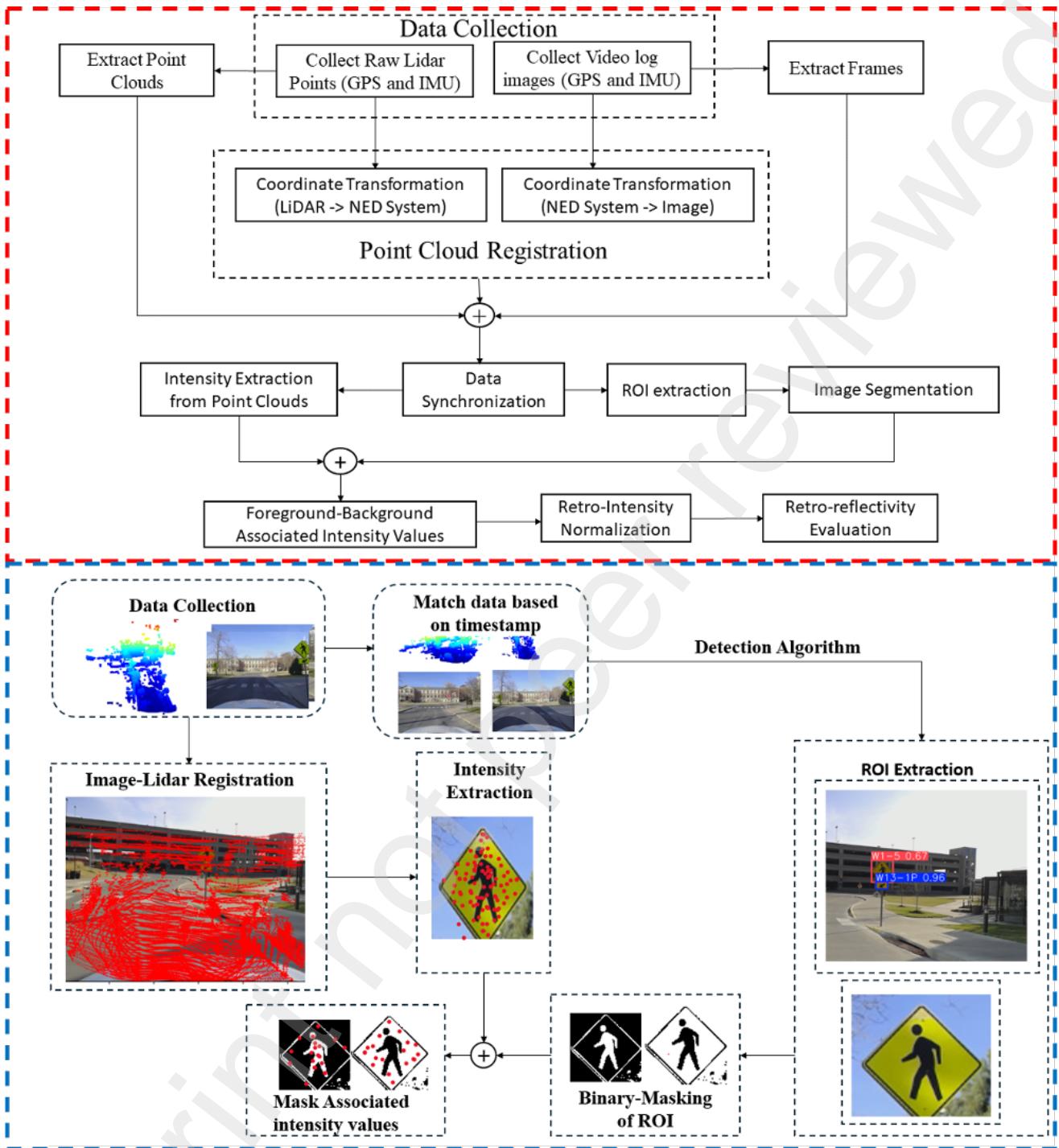
- Standley, T., Zamir, A., Chen, D., Guibas, L., Malik, J., & Savarese, S. (2020). Which tasks should be learned together in multi-task learning? URL: <http://taskgrouping.stanford.edu/>.
- Teichmann, M., Weber, M., Zöllner, M., Cipolla, R., & Urtasun, R. (2018). *MultiNet: Real-time Joint Semantic Reasoning for Autonomous Driving*. URL: <https://github.com/>. doi:10.0/ Linux-x86_64.
- Tsai, Y., Ai, C., Wang, Z., & Pitts, E. (2013). Mobile cross-slope measurement method using lidar technology. *Transportation Research Record*, (pp. 53–59). doi:10.3141/2367-06.
- Tsai, Y., Yang, Q., & Wu, Y. (2011). Use of light detection and ranging data to identify and quantify intersection obstruction and its severity. *Transportation Research Record*, (pp. 99–108). doi:10.3141/2241-11.
- Uhrig, J., Cordts, M., Franke, U., & Brox, T. (2016). Pixel-level encoding and depth layering for instance-level semantic labeling, . URL: <http://arxiv.org/abs/1604.05096>.
- Vaswani, A. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*, .
- Voegtle, T., & Wakaluk, S. (2009). Effects on the measurements of the terrestrial laser scanner hds 6000 (leica) caused by different object materials. *Proceedings of ISPRS work*, 38, 68–74.
- Wang, C., & Glenn, N. F. (2009). Integrating lidar intensity and elevation data for terrain characterization in a forested area. *IEEE Geoscience and Remote Sensing Letters*, 6, 463–466. doi:10.1109/LGRS.2009.2016986.
- Xia, J., Li, M., Liu, W., & Chen, X. (2023). Dsra-detr: An improved detr for multiscale traffic sign detection. *Sustainability (Switzerland)*, 15. doi:10.3390/su151410862.
- Zhang, L., Yang, K., Han, Y., Li, J., Wei, W., Tan, H., Yu, P., Zhang, K., & Yang, X. (2025). Tsd-detr: A lightweight real-time detection transformer of traffic sign detection for long-range perception of autonomous driving. *Engineering Applications of Artificial Intelligence*, 139. doi:10.1016/j.engappai.2024.109536.
- Zhang, L., Yu, X., Aboah, A., & Adu-Gyamfi, Y. (2024). 3d object detection and high-resolution traffic parameters extraction using low-resolution lidar data. *arXiv preprint arXiv:2401.06946*, .

Zuo, Z., Yu, K., Zhou, Q., Wang, X., & Li, T. (2017). Traffic signs detection based on faster r-cnn. In *Proceedings - IEEE 37th International Conference on Distributed Computing Systems Workshops, ICDCSW 2017* (pp. 286–288). Institute of Electrical and Electronics Engineers Inc. doi:10.1109/ICDCSW.2017.34.

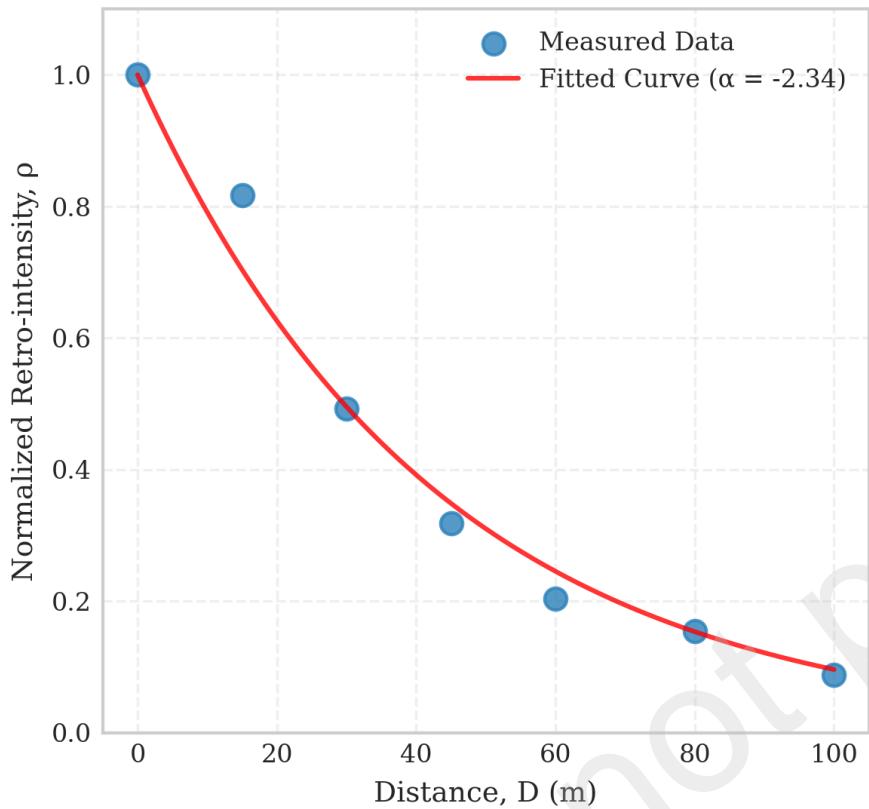




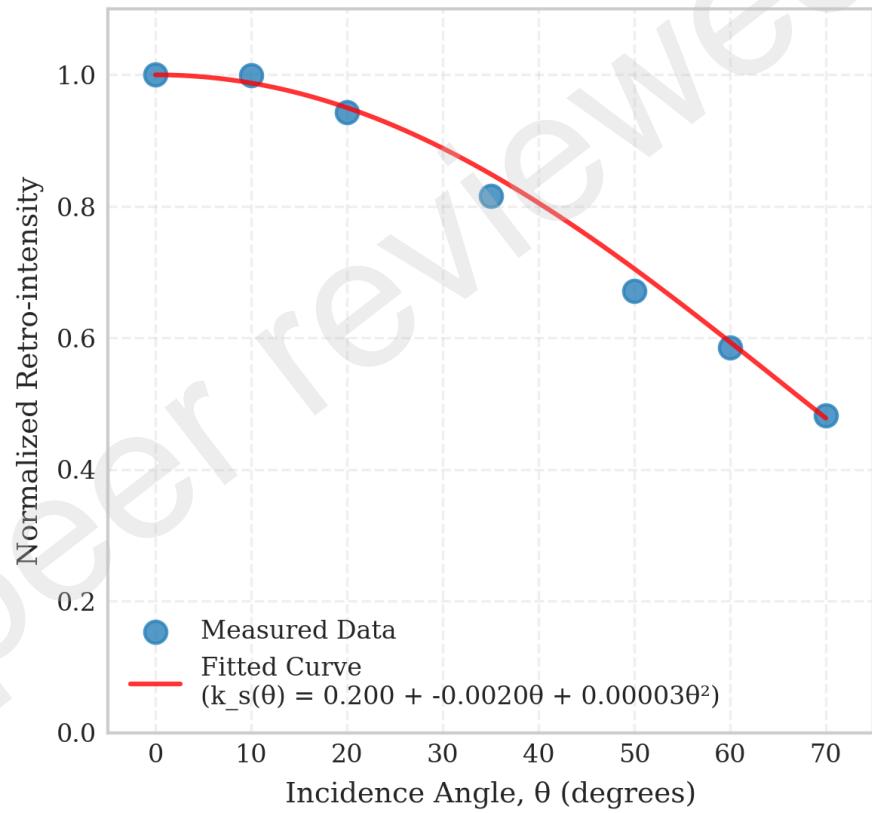




(a) Distance Test ($\theta \approx 0^\circ$)



(b) Incidence Angle Test





Confusion Matrix



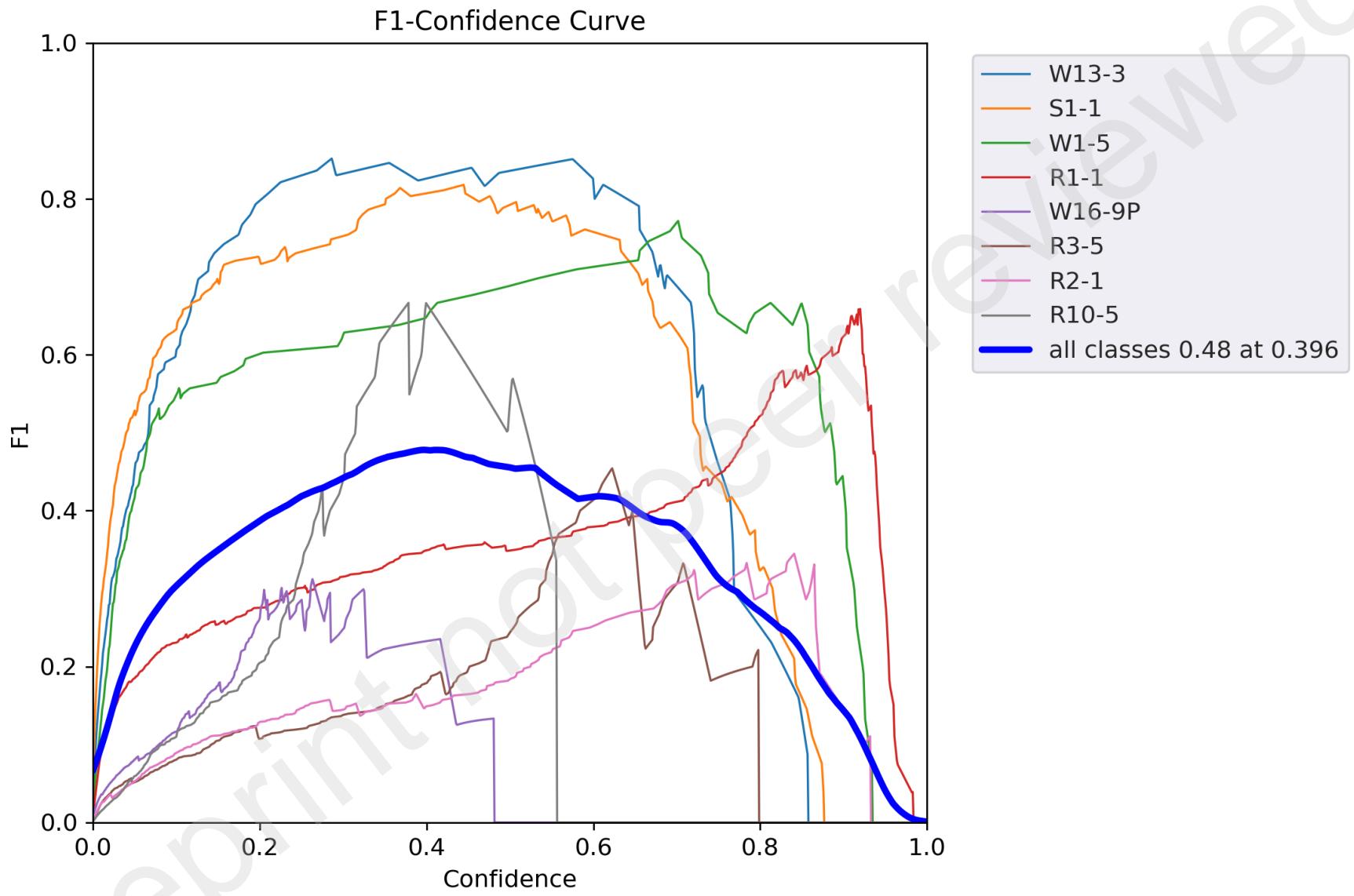


Image 1



Retroreflectivity Comparison

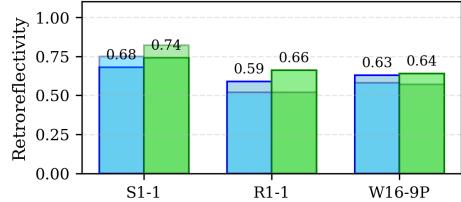


Image 2



Retroreflectivity Comparison

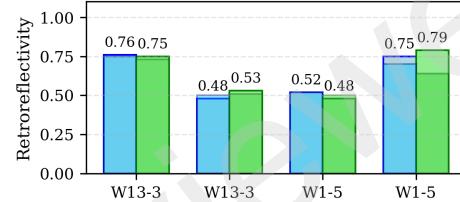


Image 3



Retroreflectivity Comparison (Single Sign)

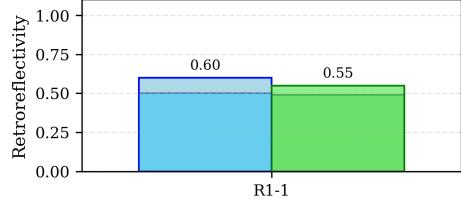


Image 4



Retroreflectivity Comparison (Single Sign)

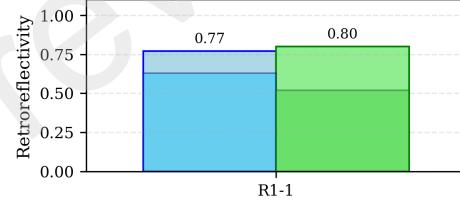


Image 5



Retroreflectivity Comparison (Single Sign)

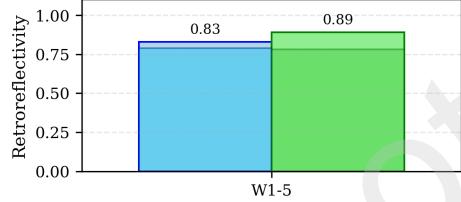
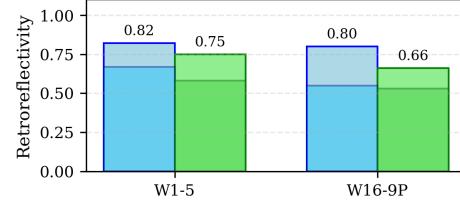


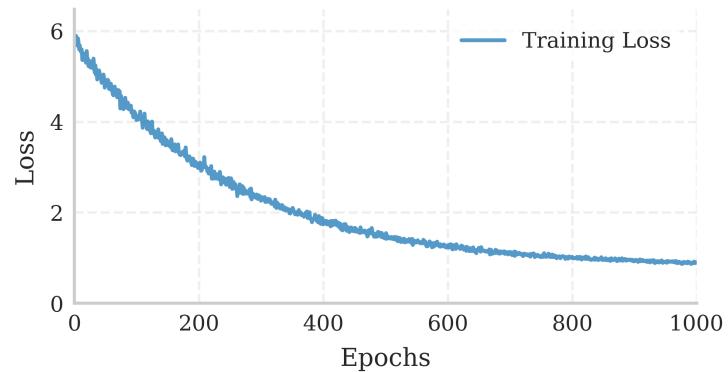
Image 6



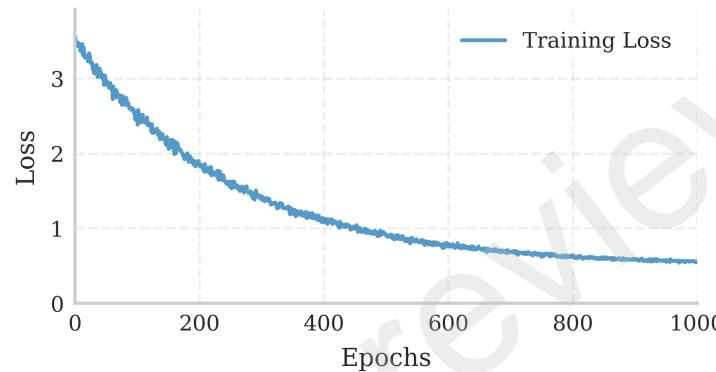
Retroreflectivity Comparison



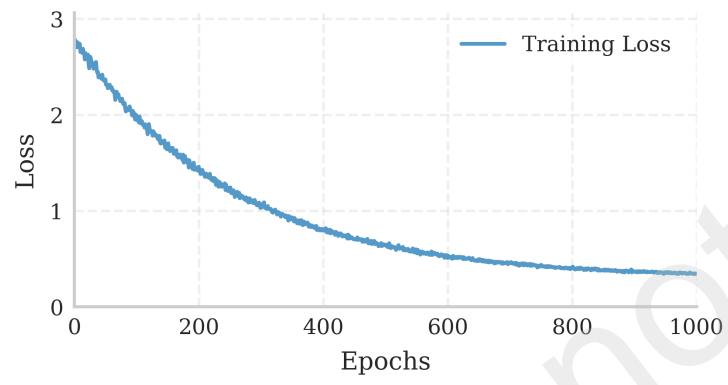
a) Box Loss



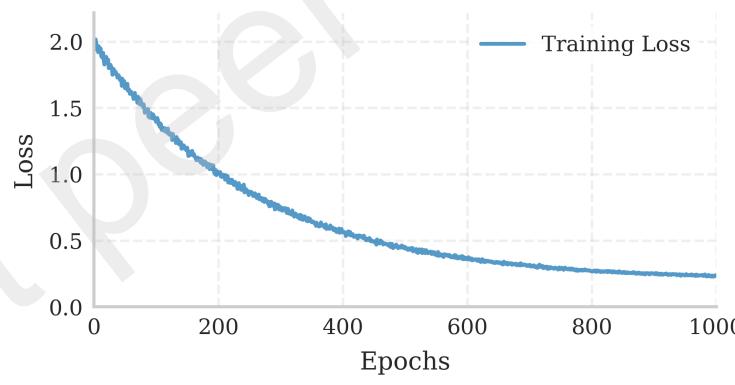
b) Object Loss



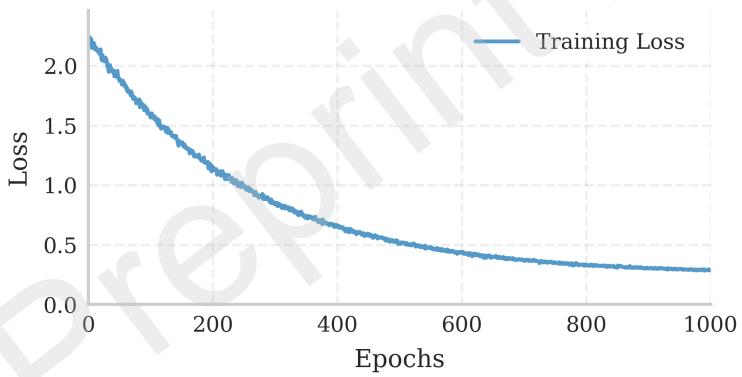
c) Classification Loss



d) Retro BG Loss



e) Retro FG Loss

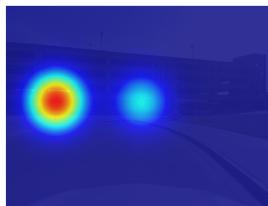




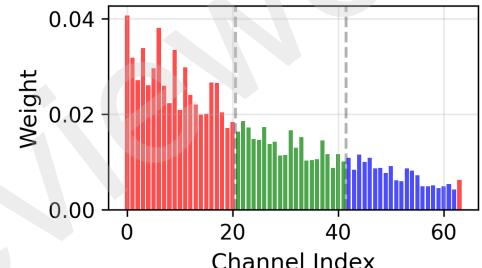
Detected Traffic Signs



Spatial Attention Map



Channel-Weighted Feature Map



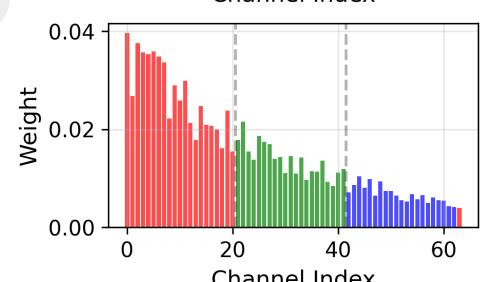
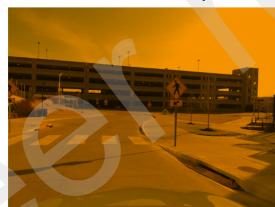
Detected Traffic Signs



Spatial Attention Map



Channel-Weighted Feature Map



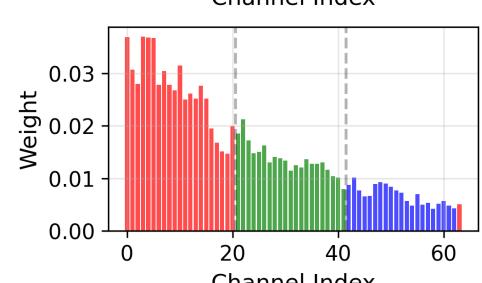
Detected Traffic Signs



Spatial Attention Map



Channel-Weighted Feature Map



Detected Traffic Signs



Spatial Attention Map



Channel-Weighted Feature Map

