Traffic Light Detection: Reducing Traffic Socializing with Traffic Light Detection

박지원, 정승민, 조태완

1 Abstract

This study aims to detect traffic lights, a crucial component in autonomous driving systems. To achieve this, we fine-tuned a pre-trained YOLOv5 model using a custom dataset. The dataset, provided by Roboflow, includes images and annotations for traffic lights. The model was trained with images resized to 360x360 pixels, a batch size of 256, and 50 epochs. The trained model accurately detects traffic lights and has demonstrated excellent performance in real-world tests. Notably, the model also excels in distinguishing the colors of traffic lights, making it useful for real-time traffic signal interpretation. This research contributes to the development of reliable and efficient traffic light detection systems, enhancing the safety and effectiveness of autonomous vehicles. Future research will explore integrating this model with vision-language models for improved contextual understanding.

2 Introduction

Traffic light detection is a critical component of autonomous driving systems, playing a vital role in ensuring the safety of both pedestrians and drivers. In many countries like Germany and the United States, traffic lights are typically positioned near pedestrian crosswalks. This placement allows for direct recognition of both the traffic signal and pedestrians, thereby facilitating a safer driving environment.

However, in South Korea, traffic lights are often installed at higher positions, making it difficult for drivers to simultaneously observe both the traffic signals and the pedestrians. This unique positioning presents a significant challenge in ensuring pedestrian safety, as drivers may miss the presence of pedestrians while focusing on the traffic lights.

To address this issue, we developed a model capable of autonomously detecting traffic lights and pedestrian crosswalks. By leveraging advanced computer vision techniques, our model aims to enhance driver awareness of pedestrians and improve overall traffic safety in South Korea. This paper presents our methodology, the dataset used, the training process, and the results of our traffic light detection model.

Figure 1: Sign location and size differences between South Korea and Germany

Figure 2: Difference in traffic light height between South Korea and Germany

3 Related Work

Research in the field of autonomous driving has made significant strides in the detection and recognition of various road elements, including pedestrians and crosswalks. One notable area of study is the development of technologies for pedestrian detection, which is crucial for ensuring the safety of individuals crossing the street.

Pedestrian Detection Technologies
Pedestrian detection systems utilize advanced computer vision techniques to identify and track individuals within the vehicle's vicinity. Methods such as Histogram of Oriented Gradients (HOG) combined with Support Vector Machines (SVM), and more recently, deep learning models like Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs), have been employed to enhance detection accuracy. For instance, the use of YOLO (You Only Look Once) models

has shown impressive results in real-time pedestrian detection, providing high accuracy and speed, which are critical for autonomous driving applications.

Integration in Autonomous Driving Systems
The integration of pedestrian detection and crosswalk recognition systems into autonomous driving platforms has been a key focus of recent research. Combining these technologies enables vehicles to make informed decisions about when to stop, slow down, or proceed, thus ensuring the safety of pedestrians. Moreover, studies have shown that using a unified framework for detecting both pedestrians and crosswalks can improve the overall efficiency and safety of autonomous navigation systems.

Challenges and Future Directions Despite significant progress, several challenges remain in the field of pedestrian and crosswalk detection. These include handling diverse environmental conditions, varying pedestrian behaviors, and the presence of occlusions. Future research is directed towards developing more adaptive and resilient models that can operate reliably under diverse scenarios. The incorporation of additional sensory data, such as LiDAR and radar, alongside visual information, is also being explored to enhance detection robustness.

4 Dataset Section

4-1 Dataset

For this research, we utilized a comprehensive dataset from Roboflow, specifically designed for traffic light detection and crosswalk recognition. This dataset contains a total of 5000 annotated images, making it a robust resource for training and validating our model.

4-2 Dataset Overview

The dataset, titled "Traffic-light Dataset," is an open-source collection provided by Roboflow and can be accessed here. It includes annotations for traffic lights in various states (red, yellow, green) and crosswalks, providing a diverse set of scenarios for model training.

4-3 Dataset Class

CrossWalk : Markgin on the road indicating pedestrian crossing areas

G_Signal : Green traffic light singals

R_Signal : Red traffic light singals

Y_Signal : Yellow traffic light singals

Figure 3 : The figure above shows a collection of annotated images from the Traffic-Light dataset provided by Roboflow. Each image is labeled with bounding boxes that indicate the location and class of the objects detected.

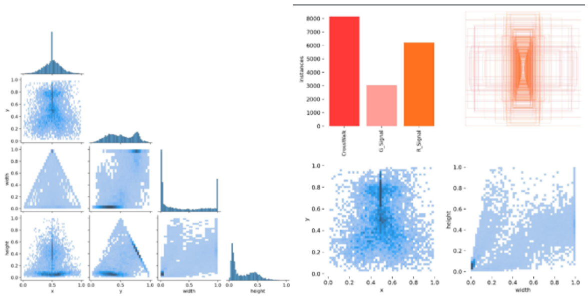


Figure 4 : Scatter Plots: Show the relationships between the x-coordinate, y-coordinate, width, and height of the bounding boxes. **Histograms**: Display the distribution of each feature (x, y, width, height) individually. **Class Distribution**: Indicates the number of annotations for each class: CrossWalk, G_Signal, R_Signal **Bounding Box Overlaps**: Visualizes the density of overlapping bounding boxes in the dataset.

5 Training

To begin training the traffic light detection model, we first cloned the YOLOv5 repository and installed the necessary dependencies.

Next, we navigated to the cloned directory and installed all required Python packages.

The dataset was split into training and validation sets. The dataset, provided by Roboflow, includes images and annotations for traffic lights and crosswalks. The annotations were formatted to be compatible with YOLOv5's expected structure. The training process involved fine-tuning a pre-trained YOLOv5 model.

- **Image Size (--img 360)**: The input images were resized to 360x360 pixels.

- **Batch Size (--batch 256)**: A batch size of 256 was used during training.

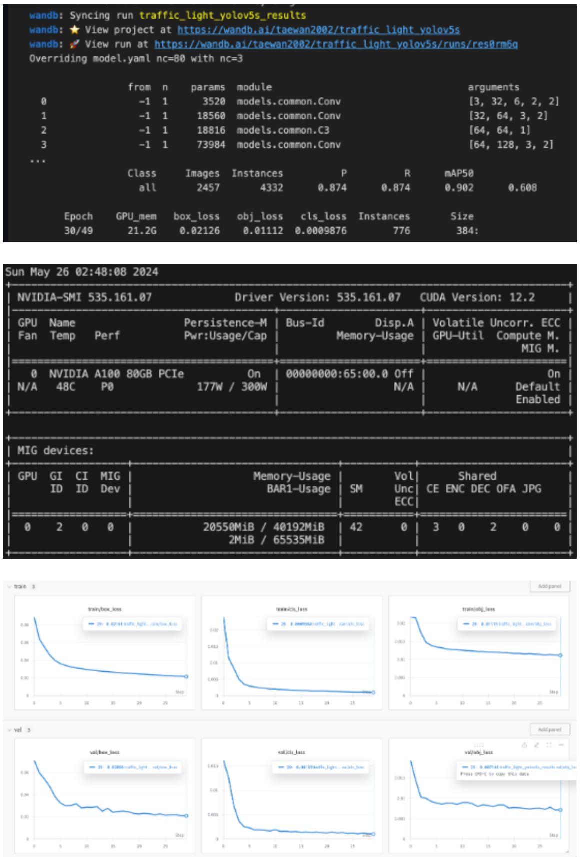
- **Epochs (--epochs 50)**: The model was trained for 50 epochs to ensure convergence and optimal performance.

- **Data Configuration (--data ../data.yaml)**: The path to the data configuration file, which contains the information about the dataset and its classes, was specified.

- **Model Configuration (--cfg models/yolov5s.yaml)**: The YOLOv5 small model configuration was used.

- **Pre-trained Weights (--weights yolov5s.pt)**: The model was initialized with pre-trained weights from the YOLOv5s checkpoint to leverage transfer learning.

- **Experiment Name (--name traffic_light_yolov5s_results)**: The experiment was named "traffic_light_yolov5s_results" for easy identification.



```
wandb: Syncing run traffic_light_yolov5s_results
wandb: ⭐ View project at https://wandb.ai/taewan2002/traffic_light_yolov5s
wandb: 🚀 View run at https://wandb.ai/taewan2002/traffic_light_yolov5s/runs/res0rm6q
Overriding model.yaml nc=80 with nc=3

                 from  n    params  module                                  arguments
  0                -1  1      3520  models.common.Conv                      [3, 32, 6, 2, 2]
  1                -1  1     18560  models.common.Conv                      [32, 64, 3, 2]
  2                -1  1     18816  models.common.C3                        [64, 64, 1]
  3                -1  1     73984  models.common.Conv                      [64, 128, 3, 2]
...
                 Class     Images  Instances          P          R      mAP50
                   all       2457       4332      0.874      0.874      0.902      0.608

      Epoch    GPU_mem   box_loss   obj_loss   cls_loss  Instances       Size
      30/49      21.2G    0.02126    0.01112  0.0009876        776        384:
```

```
Sun May 26 02:48:08 2024
+---------------------------------------------------------------------------------------+
| NVIDIA-SMI 535.161.07             Driver Version: 535.161.07   CUDA Version: 12.2     |
|-----------------------------------------+----------------------+----------------------+
| GPU  Name                 Persistence-M | Bus-Id        Disp.A | Volatile Uncorr. ECC |
| Fan  Temp   Perf          Pwr:Usage/Cap |         Memory-Usage | GPU-Util  Compute M. |
|                                         |                      |               MIG M. |
|=========================================+======================+======================|
|   0  NVIDIA A100 80GB PCIe          On  | 00000000:65:00.0 Off |                   On |
| N/A   48C    P0             177W / 300W |                  N/A |     N/A      Default |
|                                         |                      |              Enabled |
+-----------------------------------------+----------------------+----------------------+

+---------------------------------------------------------------------------------------+
| MIG devices:                                                                          |
+------------------+--------------------------------+-----------+-----------------------+
| GPU  GI  CI  MIG |                   Memory-Usage |        Vol|      Shared           |
|      ID  ID  Dev |                     BAR1-Usage | SM     Unc| CE ENC DEC OFA JPG    |
|                  |                                |        ECC|                       |
|==================+================================+===========+=======================|
|  0    2   0   0  |            20550MiB / 40192MiB | 42      0 |  3   0    2    0    0 |
|                  |               2MiB / 65535MiB |           |                       |
+------------------+--------------------------------+-----------+-----------------------+
```

Figure 5 : An illustration of how the model was trained

6 Result

The camera setup allowed us to evaluate the real-time detection capabilities of our model under various lighting and environmental conditions.

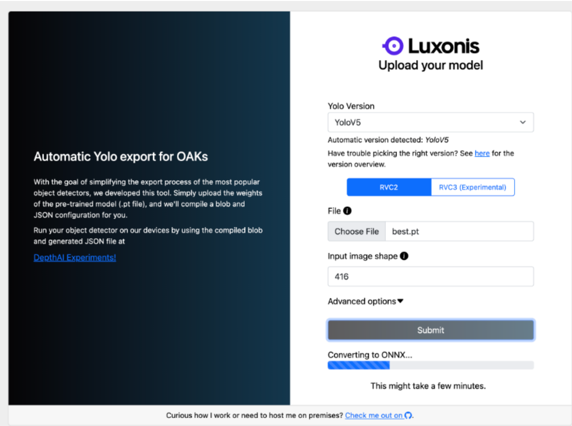


Figure 6: Testing the model with the OAK-D Lite camera

6-1 Model Conversion

After conversion, we applied the model using the DepthAI framework. The model was integrated into a Python application that allowed real-time video feed processing. The DepthAI framework facilitated the running of the compiled blob and generated JSON file on the device.

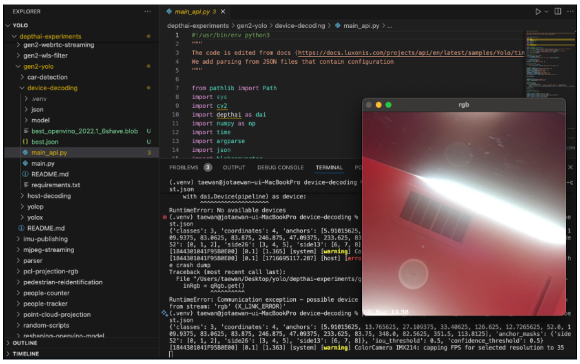


Figure 7 : Applying the model using DepthAI framework

6-2 Applying the Model

We conducted extensive tests using the camera to verify the model's performance. The model successfully detected traffic lights and crosswalks in real-time, demonstrating high accuracy and robustness.

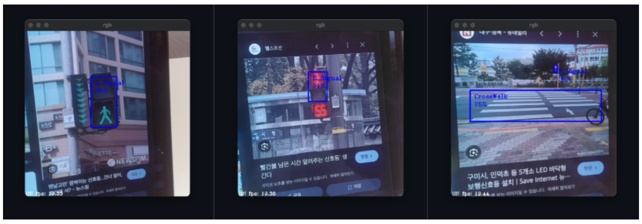


Figure 8: Real-time detection results

7 More Work

In future research, we plan to integrate advanced Vision-Language Models (VLMs) to enhance the contextual understanding of our traffic light detection system. One promising approach involves utilizing the PaliGemma-3b model, which combines image and text inputs to generate meaningful textual descriptions of visual scenes.

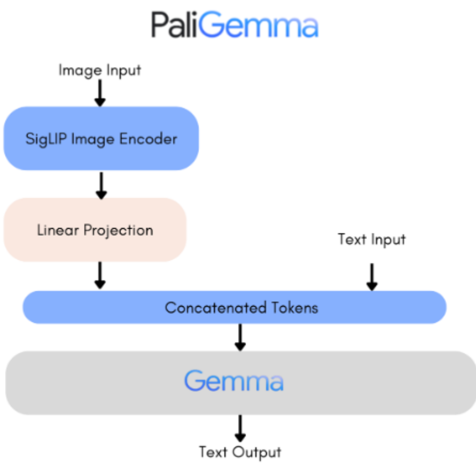


Figure 9: Vision-Language Model (VLM) Integration

The PaliGemma-3b model operates by processing image inputs through the SigLIP Image Encoder, which extracts significant features from the images. These features are then linearly projected and concatenated with text input tokens. The combined tokens are passed through the Gemma component to produce text outputs. This architecture enables the model to generate detailed and contextually rich descriptions based on both visual and textual information.

In conclusion, the integration of Vision-Language Models like PaliGemma-3b represents a significant step towards more intelligent and context-aware autonomous driving systems. Future research will focus on refining these integrations and exploring their impact on overall system performance and safety.