Multi-Input All-Weather Streetlight to Reduce Carbon Footprint of Illuminating Road Infrastructure

Cody Chen[1] and Weile Shen[#]

[1]Los Gatos High School, USA
[#]Advisor

ABSTRACT

Street lighting systems account for 15-40% of the total energy spent in cities worldwide. An ideal energy efficient street lighting system should ensure driver safety during the night with good visibility while reducing energy uses and thus costs. This project proposes a solution using a vision-based vehicle detection to respond to real-time road and traffic conditions. The system comprises of a vehicular detection model comprised of a computer vision object detection model that can be integrated into existing surveillance systems, such as a closed-circuit television (CCTV) camera, which allows for the automatic control and dimming of LED streetlights. The computer vision object detection model uses the YOLOv8 architecture, allowing for massively decreased computational requirements and model complexity while allowing for improved model accuracy. The developed model outperforms other models similarly trained in vehicular detection, such as a model from a 2023 paper based on YOLOX. The developed model has an 89% decrease in computation requirements, requiring 14.3 GFLOPS compared to 131.0 GFLOPS, and it has a 79.85% decrease in model complexity, with 11.14 MParams compared to 55.31 MParams. The developed model also performs better with a 4.12% percent increase in accuracy, with a mAP of 0.961 compared to 0.923. Decreased computational requirements and model parameter count allows the model to run on more affordable hardware, allowing a more widespread adoption in communities. More installations of the developed solution means that the energy consumption of streetlights can be rapidly decreased, thus providing long lasting environmental benefits.

Introduction

Street lighting is a vital part of city infrastructure, providing safety, security, and visibility for pedestrians and vehicles during nighttime. However, existing streetlights make up a large part of energy consumption in cities, taking up to 40% of the total energy spent in cities worldwide, meaning that their uses have massive economic and environmental implicants and should be carefully considered (Subramani et al. 2019).

With the global energy crisis and climate crisis on the rise in recent years, a sense of urgency has come around reducing energy usage and increasing energy efficiency. Tackling energy usage can reduce carbon emissions produced by the production of electricity. Recently, cities have been transitioning to light-emitting diode (LED) lamps from high intensity discharge (HID) lamps. HID lamps have several drawbacks, namely their significant lumen output deterioration, worse efficiency, and much shorter lifespans compared to LEDS. There has also been moves to develop intelligent dimmable streetlights, with Shende et al. developing a system consisting of a LED light controlled by a light dependent resistor (LDR) and passive infrared (PIR) sensor. However, PIR sensors aren't as effective outdoors in an uncontrolled environment, namely due to the fact that they work on the basis of ambient temperature, so sudden changes in ambient temperature, drafts, or direct sunlight can trigger false alarms. Furthermore, PIR sensors have reduced sensitivity to vehicles in heavy rain, snow, and dense fog, environments where streetlights are even more vital to ensure safety in low visibility environments.

The goal of my project to develop a low-maintenance all-weather streetlight in order to decrease the energy wasted and lower the carbon footprint of illuminating the road infrastructure. To address this need, this project looks

 Volume 13 Issue 3 (2024)

to computer vision and object detection. Object detection was chosen in order to quickly identify vehicles. Using a model based on the YOLOv8 architecture, the model was able to be lightweight and run quickly, allowing it to run on more affordable hardware, allowing a more widespread adoption in communities.

Methods

My developed system starts with the camera stream, which it then runs through the object detection model in order to identify any cars present in the location. Given the detection of a car, it then runs the scene through image classification models to detect the weather conditions and time of day, which it uses to determine an appropriate brightness level for the streetlight. The light remains on for at least a minute if it detects no cars following the initial detection, and it will stay on indefinitely until a period of prolonged inactivity on the road. The following design is shown in figure 1.

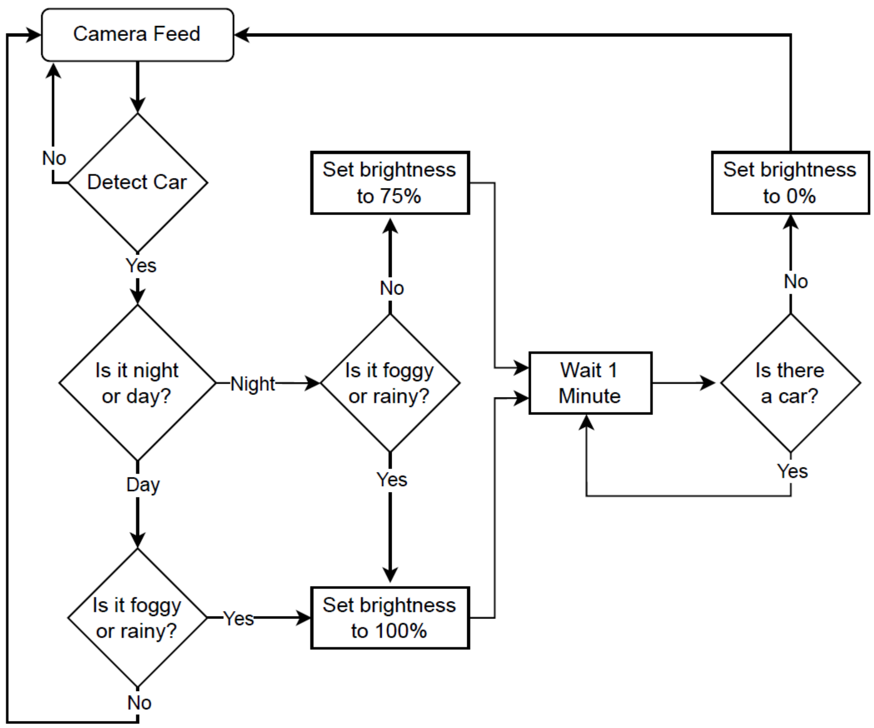

Figure 1. Flowchart of Developed System

Even during the day, in the event of rain or fog, streetlights turn to maximum brightness in order to provide maximum visibility. Otherwise, during clear conditions, streetlights only turn to 75% brightness in order illuminate the streets.

Object Detection

The object detection model, used for identifying instances of cars in the input, was based on the YOLOv8 architecture, the newest iteration in the You Only Look Once (YOLO) series of object detectors. The YOLOv8 architecture was picked for its real-time capabilities and high efficiency while maintaining accuracy. The YOLO series is renowned for its extremely fast speeds and high accuracy, namely due to the fact that it is a single-shot object detector. Single-shot object detectors, such as YOLO, use a single convolutional neural network (CNN) to simultaneously locate and classify objects in one go, rather than having to take multiple passes to detect object regions and then classify it. The backbone of the YOLOv8 architecture is shown in figure 2.

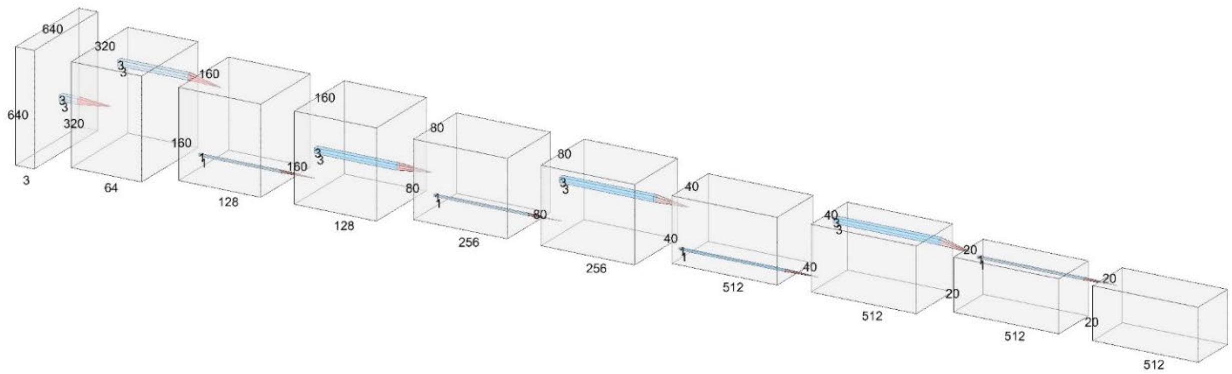

Figure 2. Backbone of YOLOv8 Architecture

The object detection model was trained using data from a Raspberry Pi Camera Module 3 NoIR. The Raspberry Pi Camera Module 3 NoIR lacks an IR cut filter, allowing it to see near infrared in addition to visible light, which normal cameras only see. Cameras without an IR cut filter, like the NoIR, can operate better in low light environments, because it allows the camera to become sensitive to more of the infrared spectrum, instead of just visible light. The longer wavelengths of the near infrared spectrum also allow it to penetrate clouds, fog, dust, smog, and smoke better than visible light, making it ideal for operating in a variety of different conditions. Furthermore, IR illuminators, which shine infrared light invisible to a normal camera, act as a flashlight and allow the camera to operate better in extreme conditions where the camera is unable to properly identify cars because of low visibility or complete absence of light.

The pictures were captured from September 2nd to September 3rd, 2023 from 9:00 PM to 1:00 AM, when the sun had fully set. The camera was aimed at an intersection in order to capture different angles: cars approaching and leaving, and cars passing by from the side. In order to save storage and only capture pictures of cars, the Raspberry Pi was running a simple motion detection script, in order to only capture images when motion occurred. The simple motion detection script compared the pixel intensity of the current frame to an initial frame of an empty road, and then found significant changes in pixel intensity values. If the values changed above a certain threshold, it would count it as motion being present, and then would take a photo every 0.2 seconds (5 pictures/second).

Of 14,000 images captured, 10,344 images, or 73.89%, were used for the training of the model and 1,060 images, or 7.57% were used for the validation. Images were labelled manually using MakeSense and bounding boxes were drawn around the vehicles. The model was then trained for 100 epochs with a batch size of 12 and an image size of 640 on an NVIDIA GeForce RTX 2060 6GB.

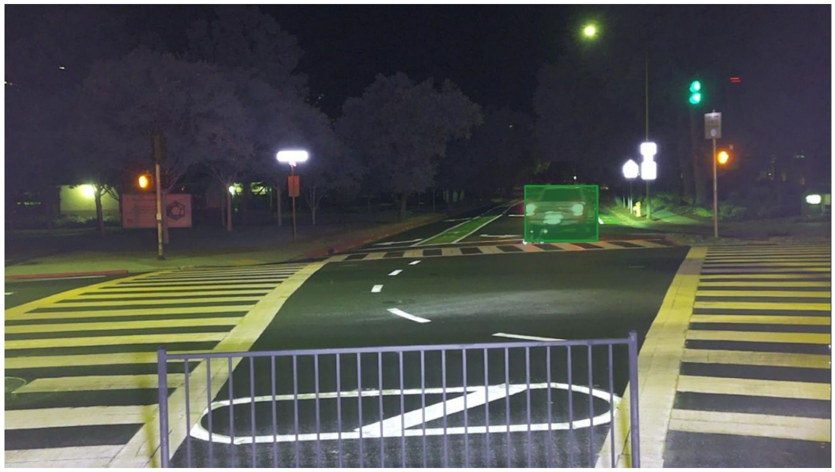

Figure 3. Example of Annotated Image

Weather Condition Classification

In order to classify current weather conditions, an image classification model based on the YOLOv8 architecture was trained. Videos from the top of a car during rainy and foggy conditions were taken using an iPhone 8 and split into images sequences for training. In total, 25,309 photos were collected, with 8,032 (31.76%) of foggy road conditions and 17,277 (68.26%) of rainy conditions. My own data was used in collection with other online datasets, including DrivingStereo, RaidaR, Berkeley's BDD100K, and CEIT-Foggy. The online datasets provided an additional 97,773 photos, supplementing my own photos for foggy and rainy conditions, and providing pictures for the cloudy and sunny classes. Out of the total 123,082 photos in the combined datasets, 99,421 (80.78%) were used for training and 23,661 (19.22%), were used for validation. The model was then trained for 30 epochs with a batch size of 12 and an image size of 640 on an NVIDIA GeForce RTX 2060 6GB.

Time Classification

In order to identify the current time, an image classification model based on the YOLOv8 architecture was trained. The model was trained on the "Daytime and Night Time Road Images" dataset, which contains 14,607 daylight images and 16,960 night images taken from the front camera of a car. Out of 31,567 image total images, 26,087 (82.64%) were used for training and 5,480 (17.36%) were used for validation. The model was then trained for 700 epochs with a batch size of 12 and an image size of 640 on an NVIDIA GeForce RTX 2060 6GB.

Results

Vehicular Detection

In order to evaluate the vehicular model's performance and effectiveness, the model needs to be evaluated on both its performance detecting cars and its ability to be run on affordable and accessible hardware.

Mean Average Precision (mAP)

One widely used metric in measuring the overall accuracy of object detection models is mean average precision, or mAP. Average precision is calculated as the weighted mean of precisions at each threshold, with the weight as the increase in recall from the prior threshold. Precision measures how accurate the predictions are, and recall measures how well the model is at identifying all of the objects in the input. The mean average precision achieved by my model is 0.963, outperforming similar vehicle detection models from other studies. Figure 4 showcases my model's precision-recall curve and figure 5 demonstrates its performance compared to other similar vehicle detection models.

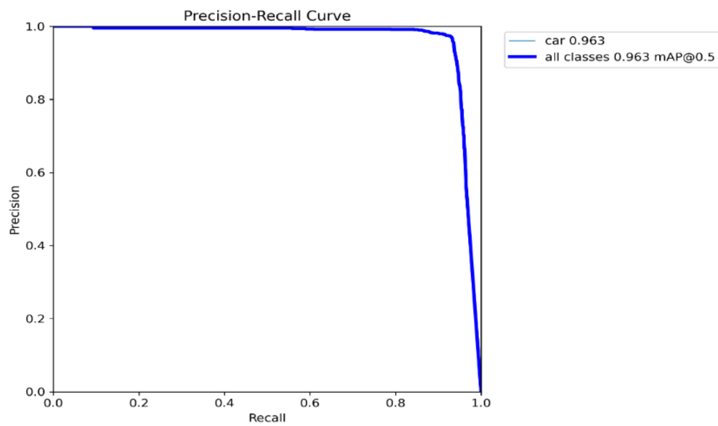

Figure 4. Precision-Recall Curve of Developed Model

HIGH SCHOOL EDITION
Journal of Student Research

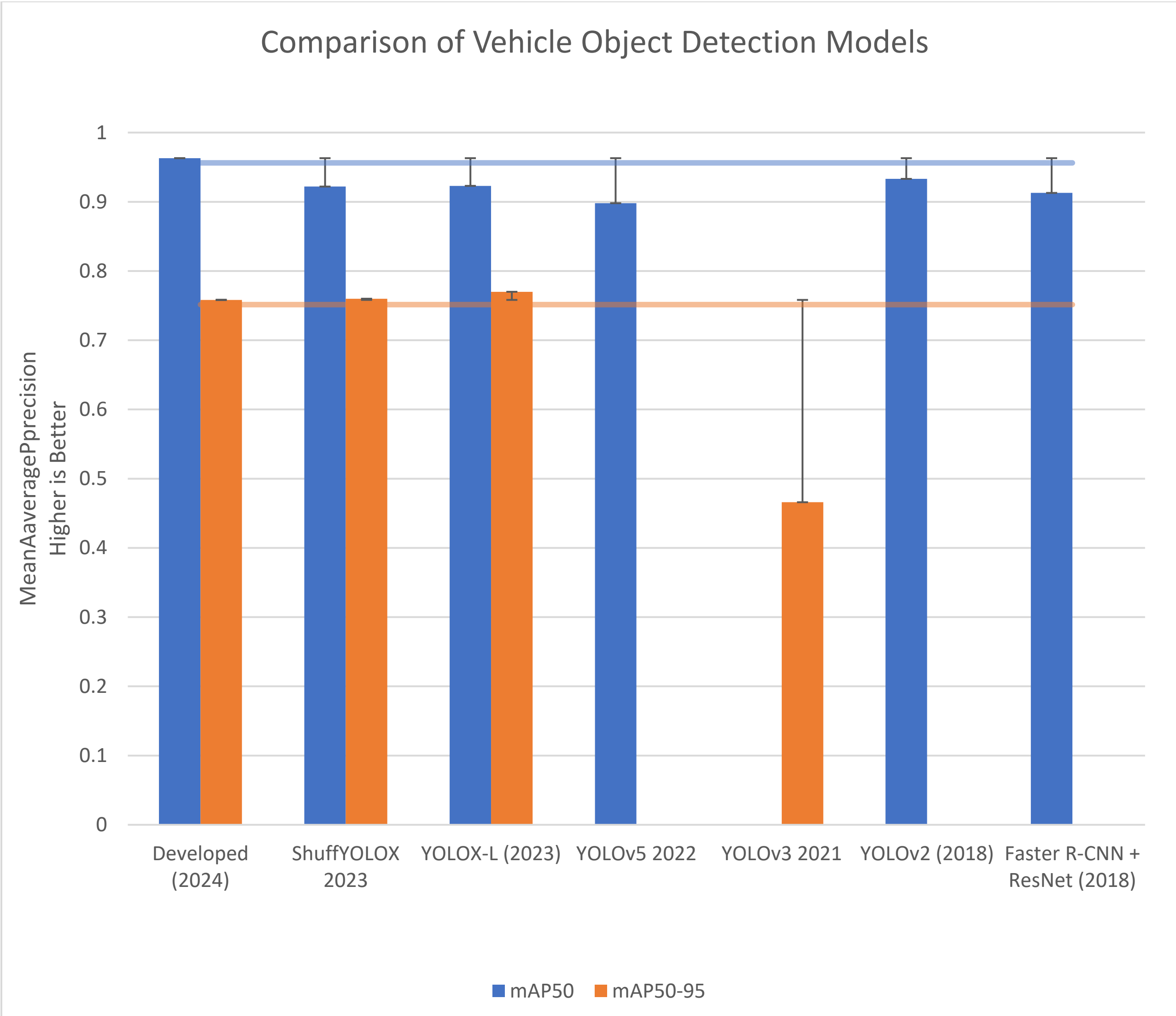

Figure 5. Comparison of Vehicle Object Detection Models in terms of mAP

Compute Requirements

Alongside with ensuring the accuracy of the model detecting cars, it's crucial to ensure that the model can be as lightweight as possible, allowing it to run on affordable and accessible hardware. Determining the FLOPs of the model is one popular metric of determining the computational requirements of the model. The developed model requires 14.323 GFLOPs, meaning it can run on more affordable hardware. Determining the number of parameters, which measures model complexity, gives a rough estimate of the amount of memory required to run the model. The developed model has 11.136 MParams. Figure 6 compares the developed model to other different architectures commonly used for vehicle detection.

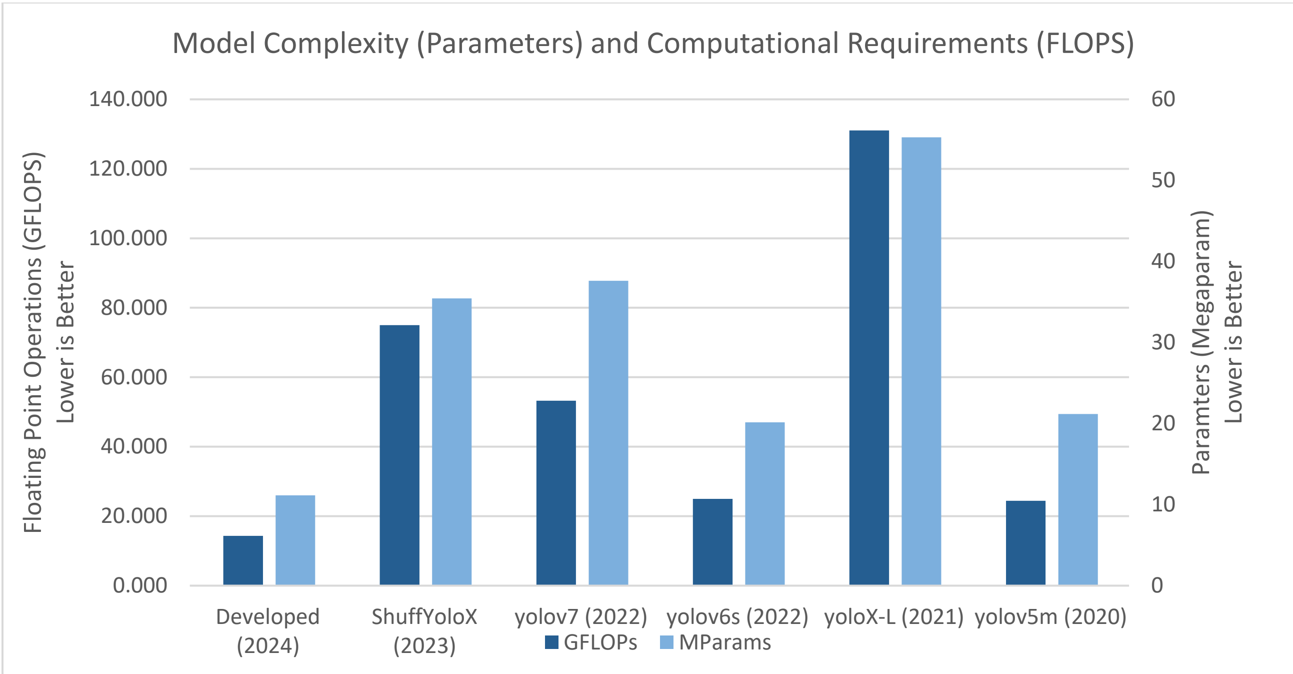

Figure 6. Comparison of Model Complexity and Computational Resources

Weather Classification

The image classification model was trained on a compilation of online datasets and collected images. The validation dataset was divided off from the initial dataset and was used to tune the model's hyperparameters. Images from the train dataset and the validation dataset both comprised of various road conditions taken from a front-facing camera's point of view from a car's perspective. In order to evaluate the model's real-world performance in the target context, which is as a camera looking at the road from an outside point of view, a test dataset was created, which involved videos looking at an intersection from the sidewalk. Using this test dataset to evaluate real-world performance, the model reached a top-1 accuracy of 92.91%. Figure 7 shows a confusion matrix of the model, which summarizes the performance of the model by displaying the percentages of true positives, false negatives, false positives, and true negatives produced.

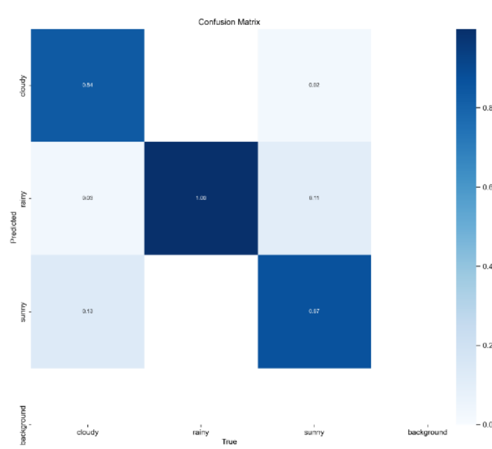

Figure 7. Confusion Matrix of Weather Classifier

Time Classification

Classifying time involves a similar process to the weather classifier. The weather classification model was trained on an online dataset of night and day images taken with the front camera of a car. The validation dataset was similar divided off of this dataset in order to tune the model's hyperparameters. Because the target perspective of the model is to face the road from an outside perspective, a test dataset was created using videos looking at an intersection from the sidewalk. Using this test dataset to evaluate real-world performance, the model reached a top-1 accuracy of 98.08%.

Real-World Energy Savings

In order to determine how much energy these lights might save in a real-world application, the Raspberry Pi Camera Module 3 NoIR and the IR illuminator were set up facing a fairly busy public road intersection. Following the design depicted in figure 1, the Raspberry Pi recorded the durations at which the light would've been at full brightness, 75% brightness, or off because of prolonged periods of inactivity. The device recorded the intersection for around 6 and a half hours, from January 1st 2024 6:05 PM to January 2nd 2024 12:38 AM.

The developed system uses an LED in order to accomplish the dimming of lights, so an equivalent LED must be found for the one used in the town. Town standards require the use of a high-pressure sodium (HPS) lamp, a type of HID lamp, with an average wattage of 150. A case study by Provident Procurement finds that an 88W LED street-light better meets the lumen requirement with a higher candela per square meter (cd/m2) than the 150W HPS lamp. Over the roughly 6 hour and a half period, the 150W HPS lamp would have used 984.67 Watt-Hours. Purely switching to an LED fixture would mean that only 577.67Wh would've been used over the same period. By dimming the lights based on vehicle activity, the developed system decreases the power consumption down to 373.13Wh over the same 6.5 hour period, a 62.11% decrease in energy consumption compared to the traditional HPS lamps.

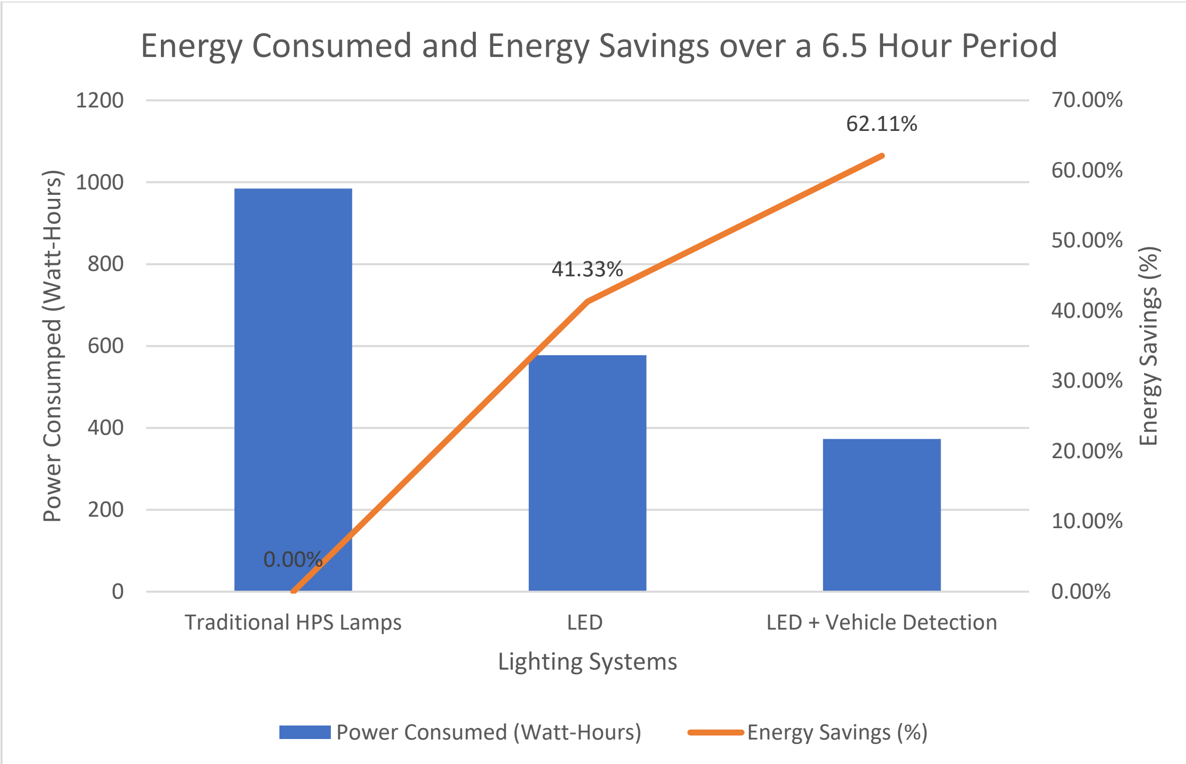

Figure 8. Energy Consumptions and Savings over a 6.5 Hour Period

Discussion

Vehicular Detection

The developed model performs better than other vehicular detection models for numerous reasons. One of the primary reasons is due to improvements in the YOLOv8 architecture. One major change is the shift away from the C3 module to the C2f module. The C2f module concatenates all outputs from two convolutional layers, allowing for it to combine high-level features with contextual information, improving detection accuracy. Furthermore, the architecture features a decoupled head, allowing for separating classification and localization, allowing each branch to focus on its task and improving accuracy. YOLOv8 also features a sigmoid function as an activation function, because of its nonlinear nature, allowing it to capture complex relationships between data. YOLOv8 also uses both CIoU and DFL loss functions in order to enhance bounding box regression, which improves object detection performance, especially with smaller objects.

My model also benefits from being more specialized and being trained on a custom dataset. Other models to detect vehicles, specifically those trained for Advanced Driver Assistance Systems (ADAS), differentiate between different types of road vehicles, such as trucks, cars, and buses, in order to respond to them accordingly. However, because a streetlight responds to all types of vehicles equally, the model does not need to differentiate between different types of road vehicles, meaning that all vehicles can be labelled under the same class. This increases the number of annotations for the one general vehicle class, allowing for the model to train with more examples, thus improving model accuracy.

Weather Classification

The weather classification model runs on the YOLOv8 image classification task. As the YOLOv8 image classification and object detection tasks share a common backbone, the image classification benefits from the same improvements and changes in feature attraction as the object detection does, such as the ones mentioned in the previous section. Like the vehicle detection model, the weather classification model also benefits from being trained on a specialized custom dataset. The training dataset was comprised of road images, allowing for the model to better understand the context of the roads, improving accuracy. Furthermore, the compilation of online datasets meant that the data came from a very diverse set of cars, conditions, and cityscapes. The improvements of the YOLOv8 architecture along with the highly specialized and diverse dataset allowed for improved model accuracy.

Time Classification

The time classification model similarly runs on the YOLOv8 image classification task. Again, as the YOLOv8 image classification and object detection tasks share a common backbone, the image classification benefits from the same improvements and changes in feature attraction as the object detection does. The model is also highly specialized, only having to differentiate between two classes, night and day. Again, the custom datasets contain road imagery, allowing the model to better learn the context of the city and road. The improvements of the YOLOv8 architecture along with the specialized custom dataset allowed the model to reach high accuracies when telling time and day.

Real-World Energy Savings

While the developed system may not seem to have much effect during earlier hours where there is much traffic, the system is much more effective at later hours, where there are less cars on the road and street lighting is simply not as needed. When analyzing the footage, it can be seen that the light rarely turns off in the first few hours, while there are still many cars on the road. The system starts turning off the light more often in the later hours of the footage, when it gets closer to 12am and there are only a couple cars every 10 minutes. It can thus be inferred that as it gets later and later, the system will become more and more effective and save more and more energy. Furthermore, the system may be more effective in the case of special events, such as holidays, where there might be less vehicles on the road because of vacation. The developed system works most efficiently at peak inactivity times, whether it be later hours or holidays, in order to dim or turn off the lights and save energy.

Conclusion

This research develops a working street lighting system that identifies and responds vehicle activity, brightening and dimming lights depending on whether they are needed, thus reducing energy consumption. The use of a convolutional neural network based on the YOLOv8 architecture is effective for creating a lightweight, real-time, and accurate model of recognizing networks. The YOLOv8 architecture was also used to train image classification models in order to classify weather conditions and time of day. The device, which uses a camera without an IR-cut filter in operate well in low-light conditions, can operate in junction with an IR illuminator in order to operate through extreme conditions, such as complete absence of light or extreme fog. When tested in the real world, the device was shown to decrease the power consumption of the street light by 62.11%. This device demonstrates a prototype of an all-weather streetlight that dims lights when they are not needed, in order to lower energy usage and thus carbon footprint of illuminating road infrastructure. In addition, future research can be carried out to determine the feasibility of integrating this system into already existing surveillance systems, such as CCTV systems. CCTV cameras traditionally send analog signals over a cable, rather than digitally, so the cost efficiency and viability of employing the developed solution, which is digital, would need to be researched. If implemented, however, this would allow for faster implementation of the developed solution, as new physical cameras would not need to be installed.

Acknowledgments

I would like to thank my advisor for the valuable insight provided to me on this topic.

References

Boshoff, D. W. (2019, February 18). *Case Study: LED vs HPS Street lights : Provident Procurement*. ProvProcure. https://provprocure.com/case-study-led-vs-hps-street-lights/

He, Q., Xu, A., Ye, Z., Wen, Z., & Cai, T. (2023). Object detection based on lightweight YOLOX for autonomous driving. *Sensors*, *23*(17), 7596. https://doi.org/10.3390/s23177596

Hong, S. G., Kim, N. S., & Kim, W. (2013). Reduction of false alarm signals for PIR sensor in realistic outdoor surveillance. *Etri Journal*, *35*(1), 80–88. https://doi.org/10.4218/etrij.13.0112.0219

Iparraguirre, O., Amundarain, A., Brazález, A., & Borro, D. (2021). Sensors on the move: onboard Camera-Based Real-Time traffic alerts paving the way for cooperative roads. *Sensors*, *21*(4), 1254. https://doi.org/10.3390/s21041254

Jin, J., Fatemi, A., Lira, W., Yu, F., Leng, B., Ma, R., Mahdavi-Amiri, A., & Zhang, H. (2021). RaidaR: A Rich Annotated Image Dataset of Rainy Street Scenes. *arXiv (Cornell University)*. https://doi.org/10.48550/arxiv.2104.04606

Redmon, J., Santosh, D. H. H., Ross, G., & Farhadi, A. (2015). You Only Look Once: Unified, Real-Time Object Detection. *arXiv (Cornell University)*. https://doi.org/10.48550/arxiv.1506.02640

Sang, J., Wu, Z., Guo, P., Hu, H., Xiang, H., Zhang, Q., & Cai, B. (2018). An improved YOLOV2 for vehicle detection. *Sensors*, *18*(12), 4272. https://doi.org/10.3390/s18124272

Shende, A., Gatpade, P., Marekar, S., & Bhaisare, S. (2022). Review of Various Technology Automatic Dimming Control of LED Street Light. *International Journal for Research in Applied Science and Engineering Technology*, *10*(4), 2016–2018. https://doi.org/10.22214/ijraset.2022.41642

Singh, R. (2020). *Day time and night time road images*. Kaggle. https://www.kaggle.com/datasets/raman77768/day-time-and-night-time-road-images/data

Subramani, C., Surya, S. M., Gowtham, J., Chari, R., Srinivasan, S., Siddharth, J. P., & Shrimali, H. (2019). Energy efficiency and pay-back calculation on street lighting systems. *AIP Conference Proceedings*. https://doi.org/10.1063/1.5112267

Terven, J. R., & Cordova-Esparza, D. (2023). A comprehensive review of YOLO: from YOLOV1 and beyond. *arXiv (Cornell University)*. https://doi.org/10.48550/arxiv.2304.00501

Yang, G., Song, X., Huang, C., Deng, Z., Shi, J., & Zhou, B. (2019). DrivingStereo: A Large-Scale Dataset for Stereo Matching in Autonomous Driving Scenarios. *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. https://doi.org/10.1109/cvpr.2019.00099

Yu, F., Chen, H., Wang, X., Xian, W., Chen, Y., Liu, F., Madhavan, V., & Darrell, T. (2020). BDD100K: A Diverse Driving Dataset for Heterogeneous Multitask Learning. *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. https://doi.org/10.1109/cvpr42600.2020.00271

Zhang, Y., Guo, Z., Wu, J., Tian, Y., Tang, H., & Guo, X. (2022). Real-Time Vehicle Detection Based on Improved YOLO v5. *Sustainability*, *14*(19), 12274. https://doi.org/10.3390/su141912274