Advancing Vehicle Classification in Bangladesh: A YOLOv8 Exploration of the 'Poribohon-BD' Dataset

Khondokar Riaz Mahmud
0122310008
Md. Hridwanur Rahman
0122320009
CSE 6023 : Machine Learning

Abstract

This paper presents an exploration of the 'Poribohon-BD' dataset for vehicle classification in the context of Bangladesh, utilizing the state-of-the-art You Only Look Once (YOLO) version 8 (YOLOv8) model. The dataset, comprising 9058 labeled images of 15 native Bangladeshi vehicles sourced from smartphone cameras and social media, serves as a rich resource for advancing computer vision applications in traffic surveillance, accident avoidance, and intelligent transportation systems. Human face blurring and data augmentation techniques have been applied to address privacy concerns and ensure dataset diversity. Our work involves the implementation of YOLOv8 for training on the 'Poribohon-BD' dataset, incorporating the latest advancements in object detection. The model's performance is evaluated against this unique dataset, considering the specific challenges and characteristics of Bangladeshi traffic scenarios. We discuss the customization of YOLOv8 configuration to accommodate the 15 vehicle classes and the subsequent training process, emphasizing the adaptability of YOLOv8 to diverse datasets.

Results from our experiments are analyzed, highlighting the strengths and potential areas for improvement of the YOLOv8 model on the 'Poribohon-BD' dataset. The paper concludes with insights into the implications of our findings for vehicle classification in the context of Bangladesh, as well as the broader applicability of YOLOv8 in diverse geographical and cultural settings getting mAP 89.1%, Precision 89.9%,Recall 82.1% .

1 Introduction

In the pursuit of advancing vehicle classification within the unique context of Bangladesh, this paper leverages the 'Poribohon-BD' dataset and employs the cutting-edge You Only Look Once (YOLO) version 8 (YOLOv8) model. This dataset, containing 9058 labeled images of 15 native Bangladeshi vehicles, is curated from smartphone cameras and social media platforms, providing a nuanced and varied representation of the country's traffic scenarios. Privacy measures, including human face blurring, and data augmentation techniques have been meticulously applied during dataset creation.

1.1 Literature review:

[1]This study identifies the YOLOV5x variant as the most suitable for vehicle detection in Bangladesh, outperforming other YOLO architectures by 7-4 percent in mAP and 12-8.5% in accuracy.

[2]The study presents a deep learning-based Bangladeshi vehicle classification model, outperforming state-of-the-art techniques, with an accuracy of 97.06%, precision of 97.17%, recall of 97.24%, and F1-score of 97.12%.

[3] This study introduces a new vehicle detection algorithm, SSB-YOLO based on the YOLOv8 model , which enhances detection accuracy and reduces computational cost, achieving a 1.6% increase in mAP@50.

[4]The Robotaxi-Full Scale Autonomous Vehicle Competition project aims to develop a traffic sign recognition system using YOLOv8, improving road safety by aiding autonomous vehicles and human drivers.

[5]This study evaluates the performance of regression-based algorithms, YOLOv5 and YOLOv8, in detecting vehicles and license plates in urban mobility, with YOLOv8 showing slightly higher accuracy and precision.

Our approach involves customizing the YOLOv8 model to accommodate the specific characteristics of the 'Poribohon-BD' dataset. The training process is detailed, emphasizing the adaptability and efficiency of YOLOv8 in handling diverse datasets. Evaluation metrics are employed to assess the model's performance, taking into account the complexities inherent in Bangladeshi traffic, such as varying vehicle sizes and unique transportation modes.

The results of our experiments shed light on the efficacy of YOLOv8 in the context of vehicle classification on the 'Poribohon-BD' dataset. We discuss the strengths exhibited by the model as well as potential areas for refinement, providing valuable insights for future research in computer vision and intelligent transportation systems tailored to the specific needs of Bangladesh.

1.2 Our Contribution

We transform the "Poribohon-BD" dataset for the YOLOv8 model, focusing on vehicle classification in Bangladesh. The results reveal elevated mean average precision (mAP), precision, and recall, showcasing the model's efficiency in real-time object detection. Our study not only highlights the effectiveness of the model but also provides valuable insights for guiding future research in this domain.

2 Material and Methods

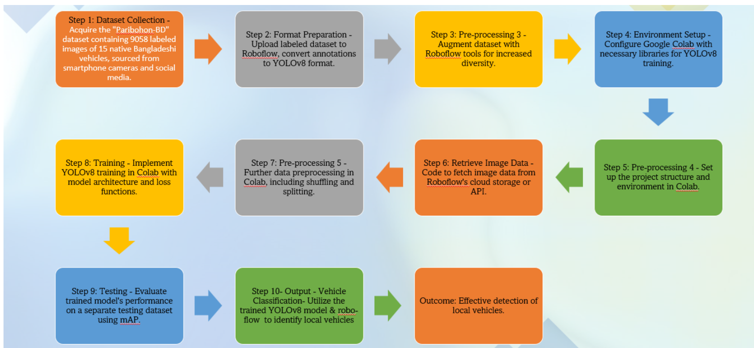

Figure 1: Methodology

Step 1: Dataset Acquire the "Paribohon-BD" dataset containing 9058 labeled images of 15 native Bangladeshi vehicles, sourced from smartphone cameras and social media. Step 2: Image Upload to Roboflow and YOLOv8 Format Preparation Upload your labeled dataset to the Roboflow platform. Convert the annotations into the YOLOv8 format, providing normalized coordinates of bounding boxes along with class indices. Step 3: Roboflow Pre-processing Utilize Roboflow's tools to apply further preprocessing if necessary. This could involve augmenting the dataset with transformations like rotation, flipping, and adjusting brightness/contrast to increase the diversity of the training data. Step 4: Start Coding Using Google Colab Set up a Google Colab environment. Import necessary libraries and dependencies required for working with YOLOv8 and training deep learning models. Step 5: Colab Environment Setup In Colab, configure your environment by installing required packages, cloning the YOLOv8 repository (if not already done), and organizing your project structure. Step 6: Image Data Code Writing in Roboflow Write code in Colab to retrieve image data directly from Roboflow's cloud storage or API, ensuring you're working with the latest labeled data. Step 7: Data Pre-processing in Colab Further preprocess the data within Colab as needed. This could involve data shuffling, splitting into training and validation sets, and performing any final data transformations. Step 8: Training Using YOLOv8 Implement the training process using the YOLOv8 architecture in Colab. This involves setting up the model architecture, defining loss functions, optimizing hyperparameters, and initiating the training loop.

Step 9: Testing the Trained Model Evaluate the trained YOLOv8 model's performance on a separate testing dataset. Calculate metrics such as mean average precision (mAP) and intersection over union (IoU) to assess the model's accuracy in vehicle classification.

Step 10: Output - Vehicle Classification Use the trained YOLOv8 model to analyze images and identify the types of vehicles present. The model's output will include predicted bounding boxes around vehicles, along with corresponding class labels.

2.1 Dataset Description

[6]Poribohon-BD is a vehicle dataset of 15 native vehicles in Bangladesh, including bicycles, boats, buses, cars, and more. The dataset contains 9058 images and annotation files, with a high diversity of poses, angles, lighting, and backgrounds. The dataset is divided into 15 folders, with the 16th folder titled 'Multi-class Vehicles' containing images and annotation files of different vehicle types. The original format was in xml for the annotations, which was later converted to YOLO text format annotation. Some of the classes are over and some are underrepresented in the dataset, after checking all the annotations it was observered.

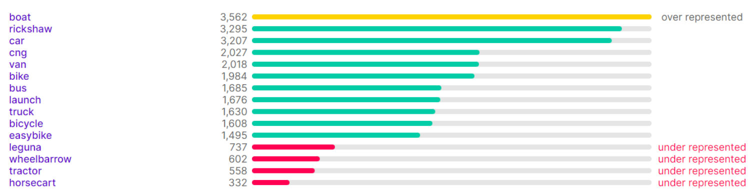

Figure 2: Annotation imbalance

2.2 YOLOv8: An Overview

YOLOv8 is a cutting-edge advancement in object detection algorithms, particularly renowned for its object detection, image classification, and instance segmentation capabilities. Developed by Ultralytics, the same team responsible for the influential YOLOv5 model, YOLOv8 introduces a host of architectural enhancements and developer-friendly features that surpass its predecessors.

Being in an active development phase, YOLOv8 constantly incorporates new features and improvements based on community feedback. Ultralytics commits to providing long-term support for their models, ensuring a continuous evolution towards excellence.

The Evolution of YOLO into YOLOv8:

The YOLO series has garnered widespread recognition in the computer vision realm. Known for its potent accuracy and compact model size, YOLO models are accessible for single-GPU training, making them suitable for deployment across a spectrum of applications, from edge devices to cloud platforms.

YOLO's journey began in 2015 with Joseph Redmond's first version. Over time, YOLOv8's author, Glenn Jocher at Ultralytics, transitioned the YOLOv3 repository to PyTorch, a deep learning framework from Facebook. This transition laid the groundwork for YOLOv5's emergence, which quickly gained prominence due to its Pythonic structure, enabling flexibility, and easy sharing of improvements across the community.

Parallelly, YOLO spawned various models like Scaled-YOLOv4, YOLOR, YOLOv7, YOLOX, and YOLOv6, each contributing novel techniques to enhance accuracy and efficiency.

Ultralytics' diligent research led to YOLOv8's inception, which was officially launched on January 10th, 2023, as the latest pinnacle of YOLO models.

Advantages of YOLOv8:

Here are key reasons to consider incorporating YOLOv8 into your computer vision projects:

High Accuracy: YOLOv8 boasts impressive accuracy, as demonstrated by its results on both the COCO benchmark and the Roboflow 100 benchmark. It achieves top-tier mean average precision (mAP) values while maintaining competitive inference speeds.

Developer Convenience: YOLOv8 offers a suite of developer-centric features, including an intuitive command-line interface (CLI) and a well-structured Python package. These enhancements streamline the training process and coding experience.

Strong Community Support: The extensive community around YOLO and the growing support for YOLOv8 mean you'll have ample resources and assistance available when working with the model.

Key Architectural Updates in YOLOv8

While there isn't a published paper on YOLOv8 yet, insights gathered from the repository shed light on the model's key updates:

Anchor-Free Detection: YOLOv8 adopts an anchor-free approach, directly predicting object centers instead of offsets from predefined anchor boxes. This simplifies predictions and improves post-processing steps like Non-Maximum Suppression.

New Convolutions: YOLOv8 introduces changes in convolutional layers, adopting 3x3 convolutions and revising the main building block. These architectural adjustments contribute to improved performance.

Mosaic Augmentation: YOLOv8 refines the training routine, incorporating mosaic augmentation. This technique stitches together multiple images, exposing the model to diverse scenarios and enhancing its ability to detect objects effectively.

Accuracy of YOLOv8

On the COCO benchmark, YOLOv8 achieves remarkable accuracy, particularly the medium-sized model (YOLOv8m) with a 50.2% mAP. Furthermore, YOLOv8 surpasses its predecessors on the Roboflow 100 benchmark, demonstrating superior performance across various task-specific domains.

In conclusion, YOLOv8 represents a significant leap forward in object detection capabilities. With its robust accuracy, developer-friendly features, and strong community support, YOLOv8 emerges as a compelling choice for advancing computer vision projects.

2.3 Performance Evaluation

Accuracy and Performance Metrics:
One of the key indicators of the YOLOv8 and Robofolow 3 object detection model's performance is its accuracy. This can be measured using metrics such as Mean Average Precision (mAP). These metrics provide insights into how well the model can detect and localize diseases in rice plants. mAP: The mAP score reflects the average precision of the model across different disease classes. A higher mAP indicates better overall performance. It's crucial to assess both the mAP for individual classes and the mean mAP across all classes to understand where the model excels and where it may need improvement.

3 Experimental Analysis

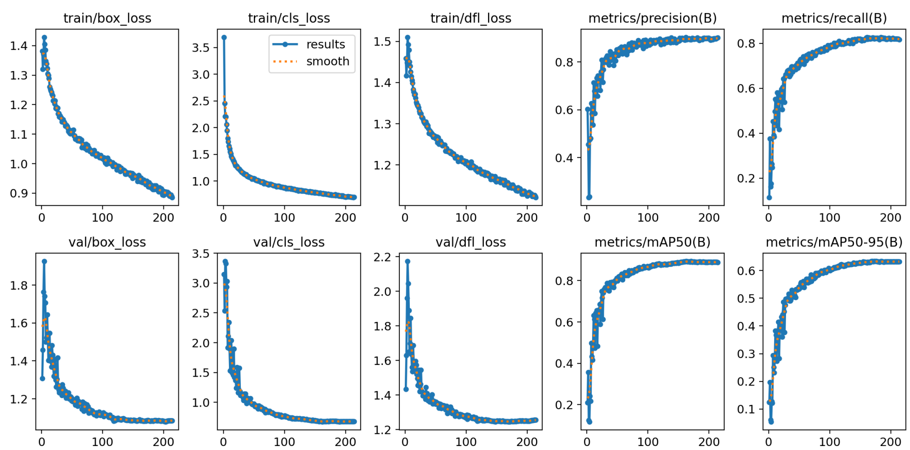

Figure 3: Results

The model got mAP 89.1%, Precision 89.9%,Recall 82.1% .
Which is phenomenal for real time object detection meaning the deployed model will very precisely predict and label the input image or video in real time. which was proven true, when the model was tested in real life scenario.

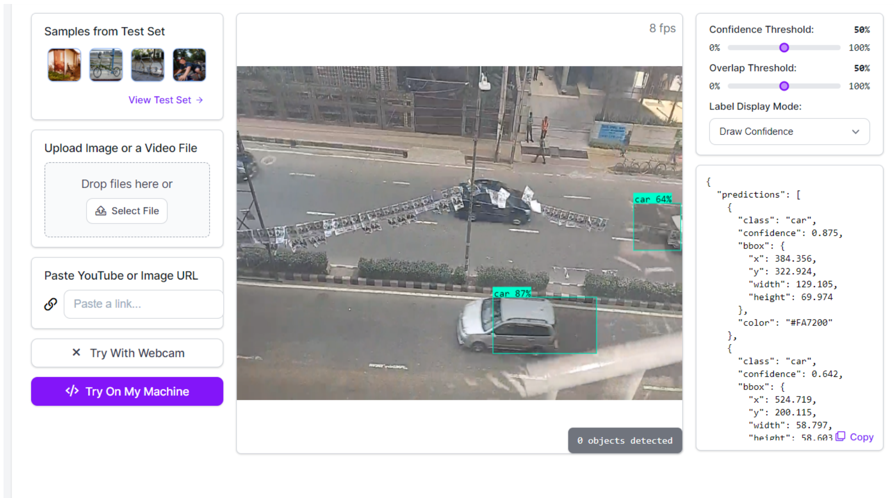

Figure 4: Testing of the model

4 Conclusion

In conclusion, this paper presents a comprehensive exploration of the "Poribohon-BD" dataset for vehicle classification. The experiments show YOLOv8 outperforms other models in Bangladeshi traffic scenarios, achieving impressive metrics like mAP of 89.1While the study provides valuable insights, it is important to acknowledge its limitations. The "Poribohon-BD" dataset, though rich, may still have certain class imbalances, as observed in the annotation analysis. Addressing these imbalances and further enhancing the dataset's diversity could contribute to even more robust model performance. Future research in computer vision is crucial for improving vehicle classification in Bangladesh. This includes comparing models, exploring hybrid approaches, optimizing for real-time applications, and addressing privacy concerns to enhance vision solutions. In summary, this study lays the foundation for future research endeavors in the field of computer vision, emphasizing the potential of YOLOv8 in vehicle classification and its broader applicability in diverse contexts.

References

[1] Refaat Alamgir, Ali Shuvro, Mueeze Mushabbir, Mohammed Raiyan, Nusrat Rani, Md Rahman, Md Kabir, and Sabbir Ahmed. Performance analysis of yolo-based architectures for vehicle detection from traffic images in bangladesh. pages 982–987, 12 2022.

[2] Tanvir Ahmed, Farzana Sharmin Mou, and Abu Ibn Samad. A deep learning based bangladeshi vehicle classification using fine-tuned multi-class vehicle image network (mvinet) model. pages 1–6, 06 2023.

[3] Luyuan Ren. Mingda Wang. Ssb-yolo: A vehicle object detection algorithm based on improved yolov8, 13 december 2023, preprint (version 1) available at research square [https://doi.org/10.21203/rs.3.rs-3743453/v1]. 12 2023.

[4] Emel Koçak and Tuncay Soylu. A performance comparison of yolov8 models for traffic sign detection in the robotaxi-full scale autonomous vehicle competition. *Multimedia Tools and Applications*, pages 1–31, 08 2023.

[5] Matheus Afonso, Eduardo Teixeira, Mateus Cruz, Guilherme Aquino, and Evandro Vilas Boas. Vehicle and plate detection for intelligent transport systems: Performance evaluation of models yolov5 and yolov8. 10 2023.

[6] Shaira Tabassum, Md Ullah, Nakib Al-Nur, and Swakkhar Shatabda. Poribohon-bd: Bangladeshi local vehicle image dataset with annotation for classification. *Data in Brief*, 33:106465, 12 2020.