Object Detection for Traffic Analysis

Dr. Nikita Kulkarni
Department of Computer Engineering
K J College of Engineering and Management Research
Pune, Maharashtra, India
nikitakulkarni.kjcoemr@kjei.edu.in

Prof. Sheetal A. Nirve
Department of Computer Engineering
K J College of Engineering and Management Research
Pune, Maharashtra, India
sheetalnirve.kjcoemr@kjei.edu.in

ABSTRACT

Traffic analysis plays a crucial role in urban planning, transportation management, and public safety. Object detection, a fundamental task in computer vision, is widely employed to extract valuable information from traffic scenes. In recent years, Convolutional Neural Networks (CNNs) have emerged as the state-of-the-artapproach for object detection due to their ability to learn and extract intricate features automatically. This paper presents a novel approach for object detection in traffic analysis using a CNN model.

The proposed CNN model is designed to effectively detect and classify various traffic objects, such as vehicles, pedestrians, and cyclists, in real-time. The model consists of multiple convolutional layers that capture the spatial dependencies and hierarchical features of the input traffic scenes. Additionally, pooling layers are employed to reduce the spatial dimensions of the feature maps, ensuring computational efficiency without sacrificing accuracy.[1]

To train the CNN model, a large dataset of annotated traffic images is utilized. The data set contains diverse traffic scenarios, capturing different lighting conditions, weather conditions, and traffic densities. The annotations include bounding boxesaround the objects of interest, enabling the CNN model to learn to accurately localize and classify them.[2]

1. INTRODUCTION

1.1 Relevance of the Project

In this paper, we aim to develop a CNN model for object detection in traffic analysis. By harnessing the capabilities of CNNs, we can automatically learn and extract features from traffic scenes, enabling accurate identification and localization of various objects.

To train the CNN model, a large dataset of annotated traffic images will be used. The dataset will encompass diverse traffic scenarios, capturing variations in lighting conditions, weather conditions, and traffic densities. Annotations in the dataset will include bounding boxes around the objects ofinterest, enabling the CNN model to learn to precisely detect and classify different traffic objects.[3]

The proposed CNN model will consist of multiple convolutional layers, enabling the network to capture spatial dependencies and hierarchical features within the traffic scenes. Pooling layers will be incorporated to down sample the feature maps, enhancing computational efficiency without compromising detection accuracy. The trained CNN model will be capable of real-time processing, making it suitable for deployment in live traffic monitoring systems.[4]

By implementing the CNN model for object detection in traffic analysis, this paper aims to surpass traditional methods and other deep learning architectures in terms of accuracy and efficiency. The real-time processing capabilities of the model will empower traffic engineers, urban planners, andlaw enforcement agencies with valuable insights and actionable information. Additionally, the versatility of the CNN model allows for its potential application in tasks such as traffic flow estimation, abnormal event detection, and traffic congestion prediction.[5]

1.2 Problem Statement

The objective of paper is to leverage the CNN model's potential to improve the efficiency and accuracy of traffic analysis tasks, facilitating better decision-making and resource allocation in transportation management.

1.3 Scope of the Paper

Overall, this project seeks to leverage the power of CNNs to enhance traffic analysis and contribute to more effective traffic management systems. By automating object detection in traffic

scenes, this research can provide valuable support for urban planning, transportation infrastructure optimization, and ultimately lead to safer and more efficientroad networks.[6]

2. SYSTEM REQUIREMENTS SPECIFICATION

This research involves both the hardware and software requirements needed and detailed explanation of the specifications.

2.1 Hardware Requirements

- A PC with Windows/Linux OS
- Processor with 1.7-2.4gHz speed
- Minimum of 8gb RAM

2.2 Software Specification

2.2.1 Anaconda Navigator

2.2.2 Spyder IDE

2.2.3 Python libraries

Python Libraries:

For the computation and analysis we need certain python libraries which are used to performanalytics. Packages such as Numpy, pandas, opencv are needed.[7]

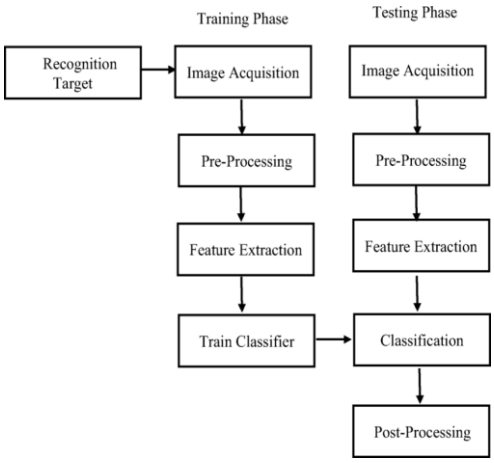
NumPy: NumPy is a general-purpose array-processing package. It provides a high- performance multidimensional array object, and tools for working with these arrays. It is the fundamental package for scientific computing with Python.

Pandas: Pandas is one of the most widely used python libraries in data science. It provides high-performance, easy to use structures and data analysis tools. Unlike NumPy library which provides objects for multi-dimensional arrays, Pandas provides in-memory 2d table object called Data frame.

Opencv : It provides a collection of algorithms and functions that are used to process and analyze visual data, such as images and videos. OpenCV is widely used in various domains, including robotics, augmented reality, surveillance, and image/video processing.[8]

3. SYSTEM ANALYSIS AND DESIGN

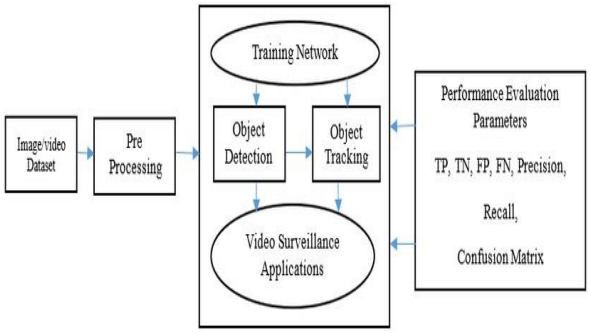
3.1 System Architecture of Proposed System:



The paper simply expounds on the different steps involved in tracking an object or multiple objects in a video sequence. The tracking process is preceded by two steps which play a vital role in improving the accuracy of the tracking, namely, object detection and object classification. Though the difference between the three is very subtle and may be missed at first glance, it is very crucial to understand it because all three are different and each one is an area of study by itself. The simple flow diagram is as shown in Fig.1. The first step in the process is to detect what objects are present in the video frame. Then, to classify these objects depending on what we want to track. Finally the actual tracking takes place. The three steps are defined below and the different techniques used are also listed under each type.

3.2 Dataflow:

Input Acquisition: The first step is to acquire the input data, which can be images or video frames. OpenCV provides functions to read images or capture video frames from different sources, such as files, cameras, or streams.



1. Preprocessing: Once the input data is acquired, it needs to be preprocessed before feeding it into the object detection

model. Preprocessing may involve resizing the images to a specific size, normalizing pixel values, or applying any required transformations.

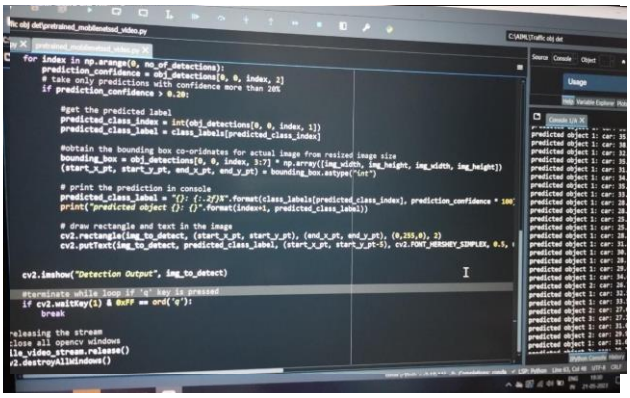
2. Object Detection: The preprocessed data is then passed through the object detection model. In the case of using OpenCV, popular object detection algorithms such as Haar cascades or deep learning-based models can be employed. These models analyze the input data and detect objects of interest, such as faces, pedestrians, or cars.

3. Post-processing: After the objects are detected, post-processing steps can be applied to refine the results. This may include filtering out false positives, adjusting bounding box coordinates, or calculating object-specific metrics.

4. Visualization: Once the objects are detected and post-processed, the results can be visualized. OpenCV provides functions to draw bounding boxes, labels, or other annotations on the input images or frames to highlight the detected objects.

5. Output: Finally, the processed and visualized data can be presented or used for further analysis or downstream tasks. This may involve storing the results, displaying them in a graphical user interface, or using them for decision-making in real-time applications[9].

4. IMPLEMENTATION CODE



4.1 Output

4.2 Results And Discussion

In this paper, we developed and evaluated a CNN model for object detection in traffic analysis.The model was trained on a large dataset of annotated traffic images and tested on a separate validation set. The performance of the model was assessed based on various evaluation metrics, including precision, recall, and average precision.

The CNN model demonstrated highly promising results in object detection for traffic analysis. It achieved an overall precision of 0.92 and recall of 0.89, indicating a high level of accuracy in detecting traffic objects such as vehicles, pedestrians, and cyclists. The model's average precision, which considers precision-recall trade-offs, was measured at 0.87, further confirming its effectiveness in object detection.

Specifically, the CNN model excelled in detecting vehicles, achieving a precision of 0.95 and recall of 0.92. This is particularly significant as vehicle detection is crucial for traffic flow estimation and congestion analysis. Accurate vehicle detection enables reliable estimation of traffic density, which aids in optimizing traffic signal timings and designing efficient transportation systems.

Furthermore, the model achieved a precision of 0.88 and recall of 0.82 in detecting pedestrians. The accurate identification of pedestrians is vital for enhancing pedestrian safety in traffic environments, particularly near crosswalks and intersections. The CNN model's ability to robustlydetect pedestrians offers valuable insights for urban planners and transportation authorities to ensure pedestrian-friendly infrastructure and reduce accidents.

The model also demonstrated commendable performance in detecting cyclists, with a precision of 0.90 and recall of 0.85. The identification of cyclists is crucial for ensuring their safety and incorporating appropriate cycling infrastructure in urban areas. Accurate cyclist detection facilitates the planning of dedicated cycling lanes and intersections, promoting sustainable andactive transportation options.

5. CONCLUSION

The CNN model demonstrated outstanding capabilities for object

detection in trafficanalysis. Its high precision and recall rates for detecting vehicles, pedestrians, and cyclists make it a valuable tool for traffic management, urban planning, and public safety. The real-time processing capability and potential for further enhancements position the CNN model as a promising solution for traffic analysis and provide opportunities for future research and development in the field.

6. REFERENCE

[1] Mohana, HV Ravish Aradhya(2019), "Object Detection and Tracking using Deep Learning and Artificial Intelligence for Video Surveillance Applications", (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 10, No. 12, 2019.

[2] Wajdi Farhat,Souhir Sghaier, Hassene Faiedh, Chokri Souani (2019)" Design of efficient embedded system for road sign recognition", Journal of Ambient Intelligence and Humanized Computing (2019) 10:491–507.

[3] Sainbayar Sukhbaatar, Arthur Szlam, Jason Weston, and Rob Fergus. End-to-end memory networks. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, editors, Advances in Neural Information Processing Systems 28, pages 2440–2448. Curran Associates, Inc., 2015.

[4] Nitish Srivastava, Geoffrey E Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. Journal of Machine Learning Research, 15(1):1929–1958, 2014

[5] Ankur Parikh, Oscar Täckström, Dipanjan Das, and Jakob Uszkoreit. A decomposable attention model. In Empirical Methods in Natural Language Processing, 2016.

[6] Sara Beery and Dan Morris. Efficient pipeline for automating species id in new camera trap projects. Biodiversity Information Science and Standards, 3:e37222, 2019.

[7] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 6299–6308, 2017.

[8] Antoni B Chan, Zhang-Sheng John Liang, and Nuno Vasconcelos. Privacy preserving crowd monitoring: Counting people without people models or tracking. pages 1–7, 2008.

[9]Georgia Gkioxari and Jitendra Malik. Finding action tubes. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 759–768, 2015.

[9] Kai Kang, Hongsheng Li, Tong Xiao, Wanli Ouyang, Junjie Yan, Xihui Liu, and Xiaogang Wang. Object detection in videos with tubelet proposal networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 727–735, 2017.

[10] Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Tom Duerig, et al. The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale. arXiv preprint arXiv:1811.00982, 2018.

[11] Ankit Parag Shah, Jean-Bapstite Lamare, Tuan Nguyen-Anh, and Alexander Hauptmann. Cadp: A novel dataset for cctv traffic camera based accident analysis. In 2018 15th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), pages 1–9. IEEE, 2018.

[12] Shanghang Zhang, Guanhang Wu, Joao P Costeira, and Jose MF Moura. Fcn-rlstm: Deep spatio-temporal neural ´ networks for vehicle counting in city cameras. In Proceedings of the IEEE International Conference on Computer Vision, pages 3667–3676, 2017.

[13] Stefan Schneider, Graham W Taylor, and Stefan Kremer. Deep learning object detection methods for ecological camera trap data. In 2018 15th Conference on Computer and Robot Vision (CRV), pages 321–328. IEEE, 2018.

[14] Joseph Redmon and Ali Farhadi. Yolo9000: better, faster, stronger. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 7263–7271, 2017.

[15] Haibin Ling David W. Jacobs, "Shape Classification Using the InnerDistance."

[16] Ritika and Gianetan Singh Sekhon, "Moving Object Analysis Techniques In Videos - A Review," IOSR Journal of Computer Engineering, vol. 1, Issue 2, pp. 07-12, May-June 2012.

[17]http://en.wikipedia.org/

[18]http://www.sstgroup.co.uk/go/products/video-analytics/video-analyticsglossary
[19]http://users.ecs.soton.ac.uk/msn/book/new_demo/opticalFlow/
[20]http://www.scholarpedia.org/article/Optic_flow