Developing an OEDR Module For Autonomous Vehicle Using Simulators

Souvik Mandal
Tatavarthy Venkat Nikhil
Gourinath Banda

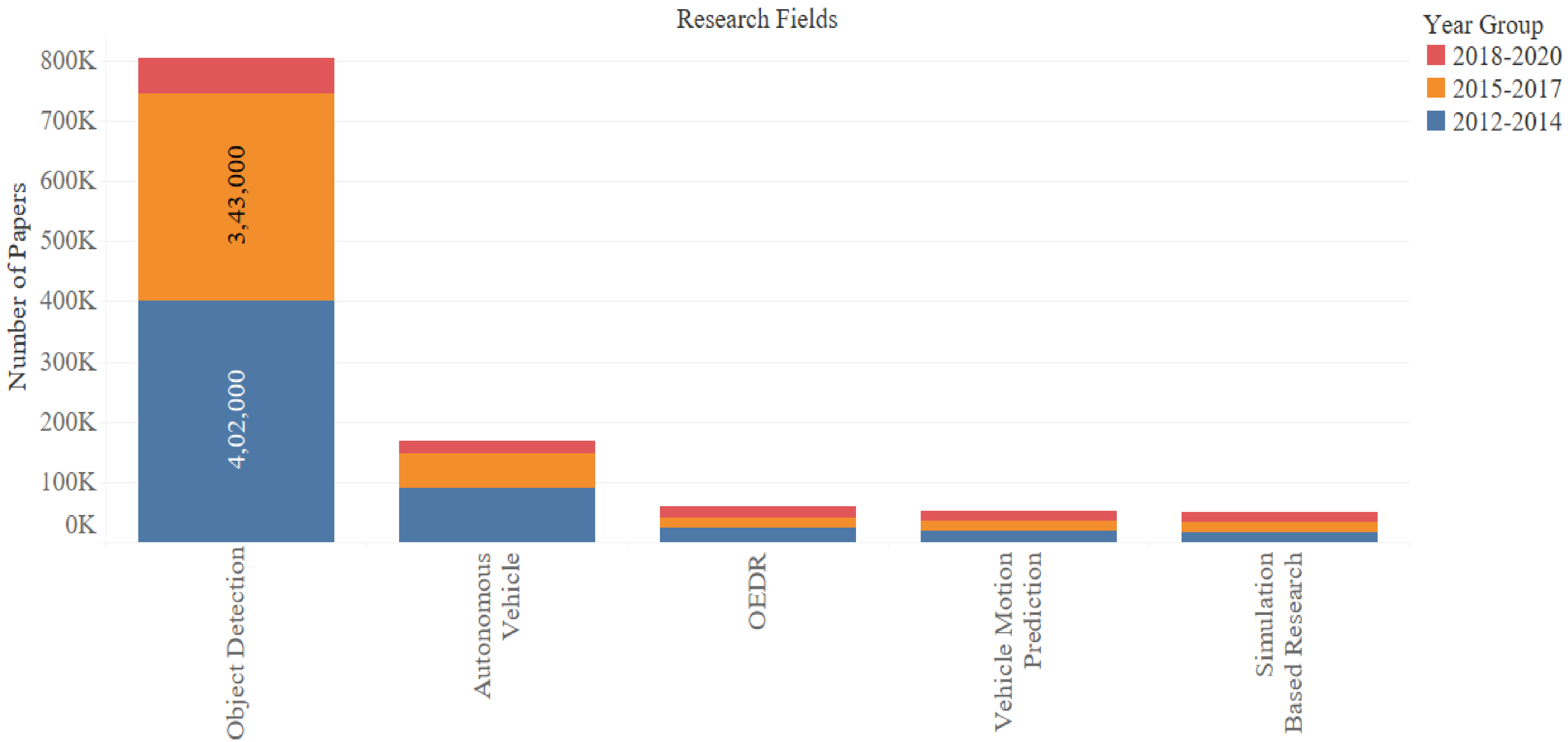
Discipline of Computer Science and Engineering
Indian Institute of Technology Indore

Motivation

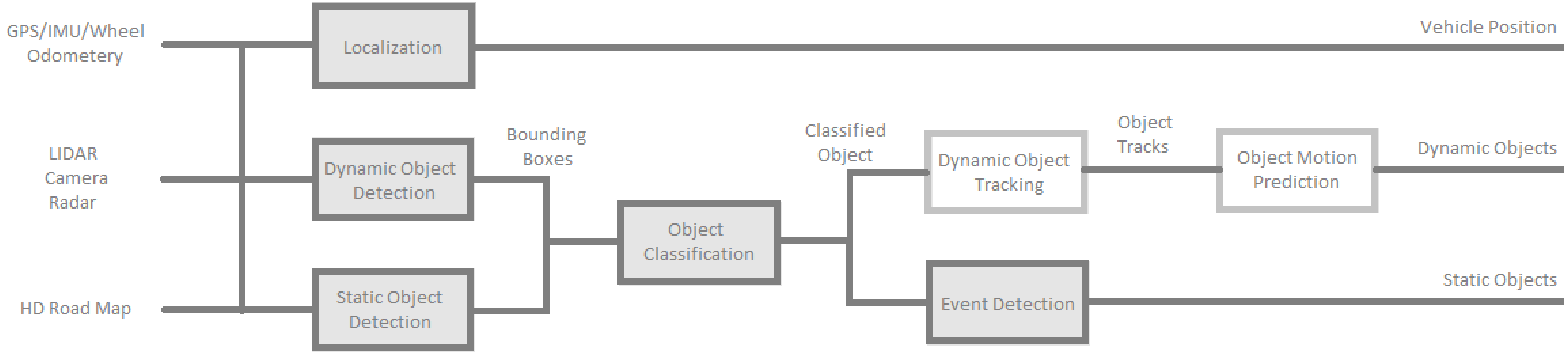
- ➢ Getting started with research in autonomous vehicles
  - in the absence of a true autonomous driving environment in an academic institute
  - and to form a special interest group for autonomous driving at IIT Indore.
- ➢ Simulators make it possible for virtual training and testing.
- ➢ IPG-CarMaker simulator lets to define complex scenarios.
- ➢ We wanted to develop an OEDR module
  - OEDR module is essential as a perception unit for autonomous vehicles.

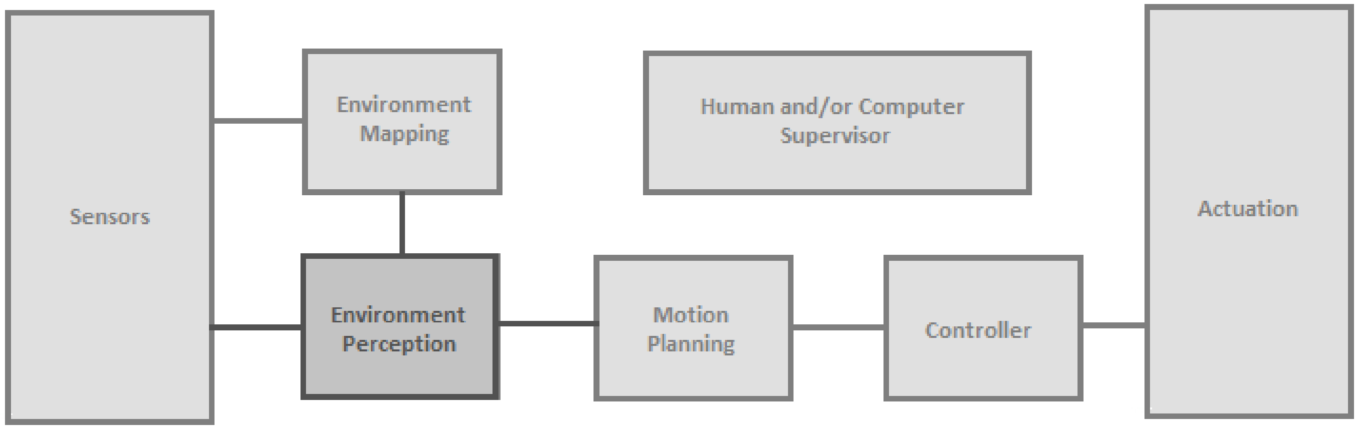
Status of Research

Autonomous Vehicle, Object Detection, OEDR and Simulation



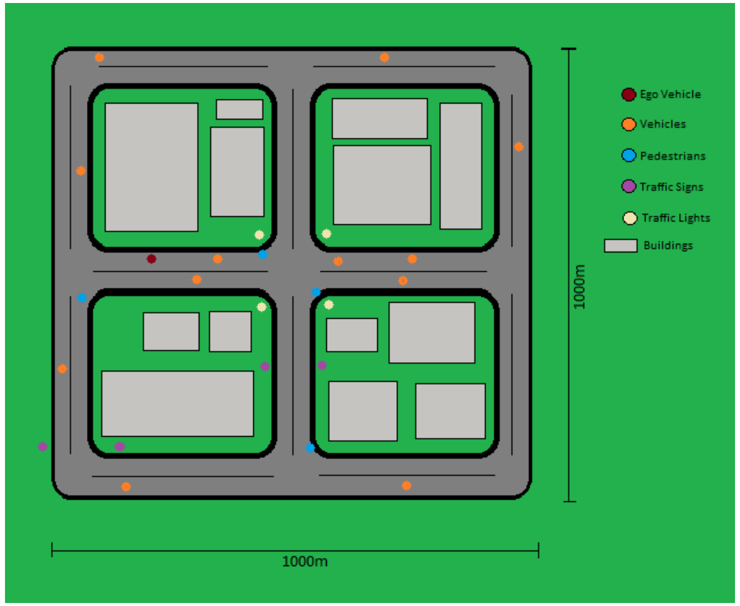
Autonomous Vehicle (Block Diagram)





Traffic Scenarios Modelled in IPG-CarMaker

We have created over 20 traffic scenarios containing both static and dynamic entities such as cars, buses, bicycles, motorcycles, pedestrians, trucks, traffic signs, traffic lights, terrain and other miscellaneous objects in CarMaker.



Object Detection and Classification Module

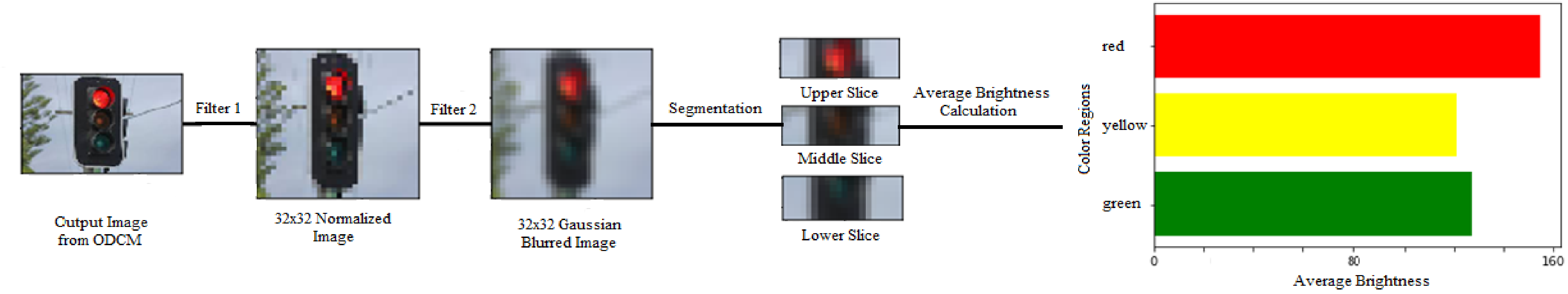
- Object detection algorithm: You Only Look Once version 3
  - YOLOv3 is extremely fast and reasonably accurate.
  - Fully Convolutional with skip connection and up-sampling.
  - Small objects are detected with higher accuracy compared to its predecessors for its three-level detection and classification.

ODCM - Dataset

- Urban traffic sign dataset used for training: RoViT Urban Object Detection Dataset
  - Small dataset (~23 GB).
  - Our work is restricted to university-level development and as large or medium-sized dataset require significant processing power, so we have used the RoViT dataset, which is significantly smaller in size.

Traffic Light Classification

- The traffic light classifier runs on the traffic lights annotated by the Object Detection and Classification Module.
- This subsystem is used for identifying the state of the traffic light by comparing the average brightness corresponding to the regions of the three lights.



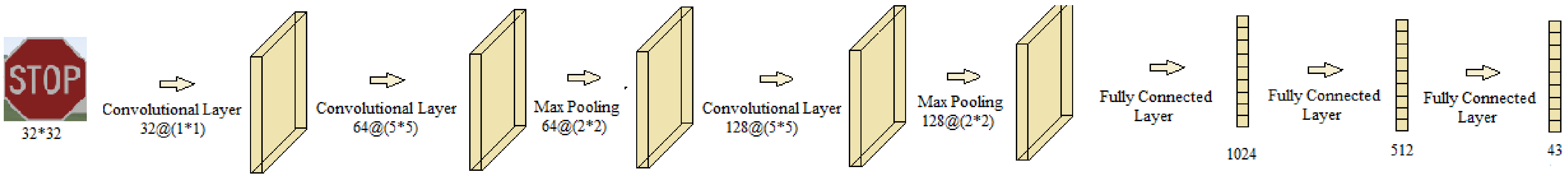
Filter 1 - SizeNormalization Filter
Filter 2 - Gaussian Filter

Traffic Light Classification

- Average Time for computation is 3.3 ms.
- Testing Images for red Traffic light illuminated: 723.
- Testing Images for green traffic light illuminated: 429.
- Testing Images for yellow traffic light illuminated: 44.
- 100% accuracy for Red traffic light illuminated.
- 100% accuracy for Yellow traffic light illuminated.
- 99% accuracy for Green traffic light illuminated.

Traffic Sign Classification

- The traffic sign classification subsystem also uses the traffic signs annotated by the Object Detection and Classification Module.
- We have implemented the traffic sign classifier based on a convolutional neural network and trained with the German traffic sign dataset.

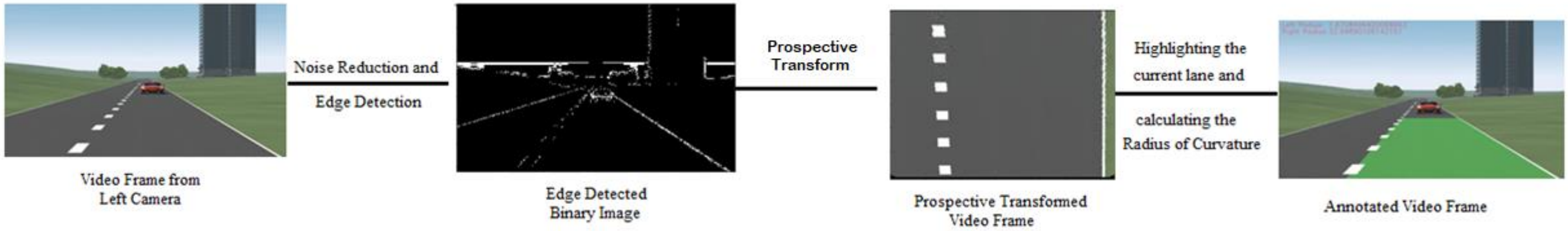


Traffic Sign Classification

- Total Number of Classes: 43.
- Training Images 34799.
- Testing Images 10000.
- Accuracy of the Model is 97%.
- Average Time for computation is 3.3 ms.

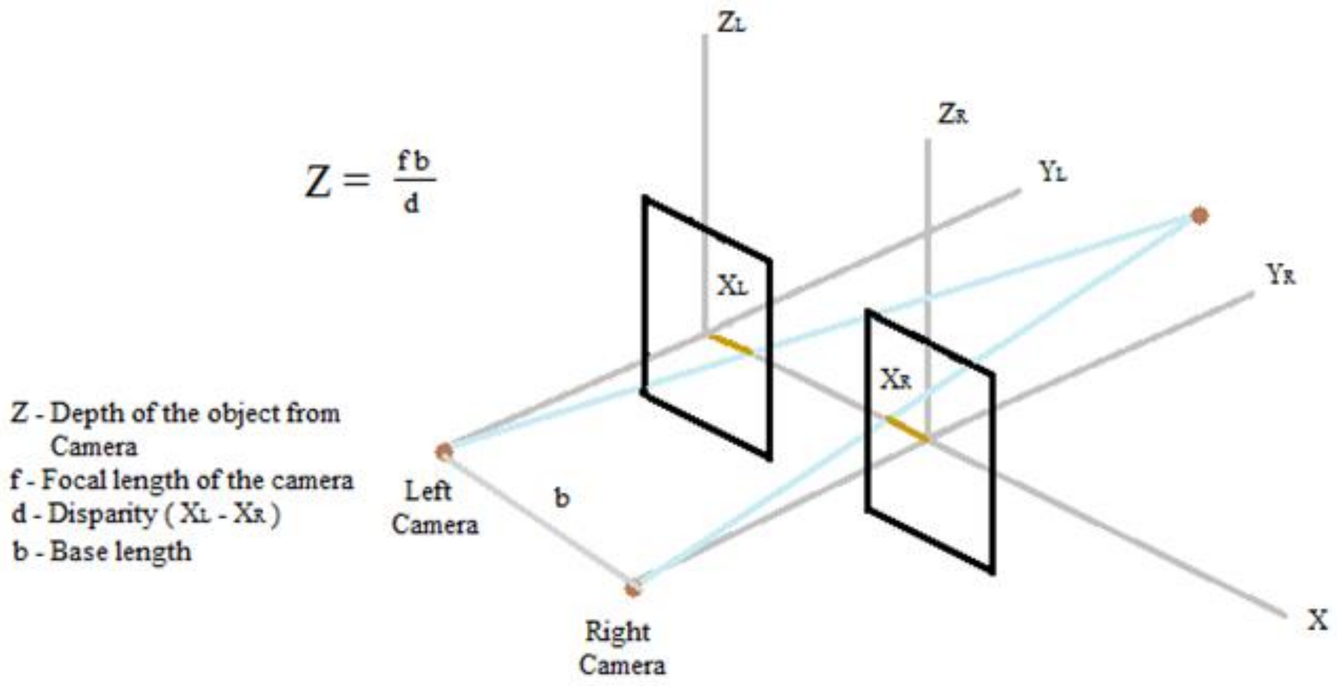
Lane Detection

Lane detection subsystem estimates lanes coordinates and lane radius of curvature using the ego vehicle's primary camera sensor data.



Depth Estimation

- The depth estimation subsystem, unlike previous subsystems, uses the right camera sensor data along with the annotated output generated by the ODCM.
- It employs stereo depth estimation techniques.



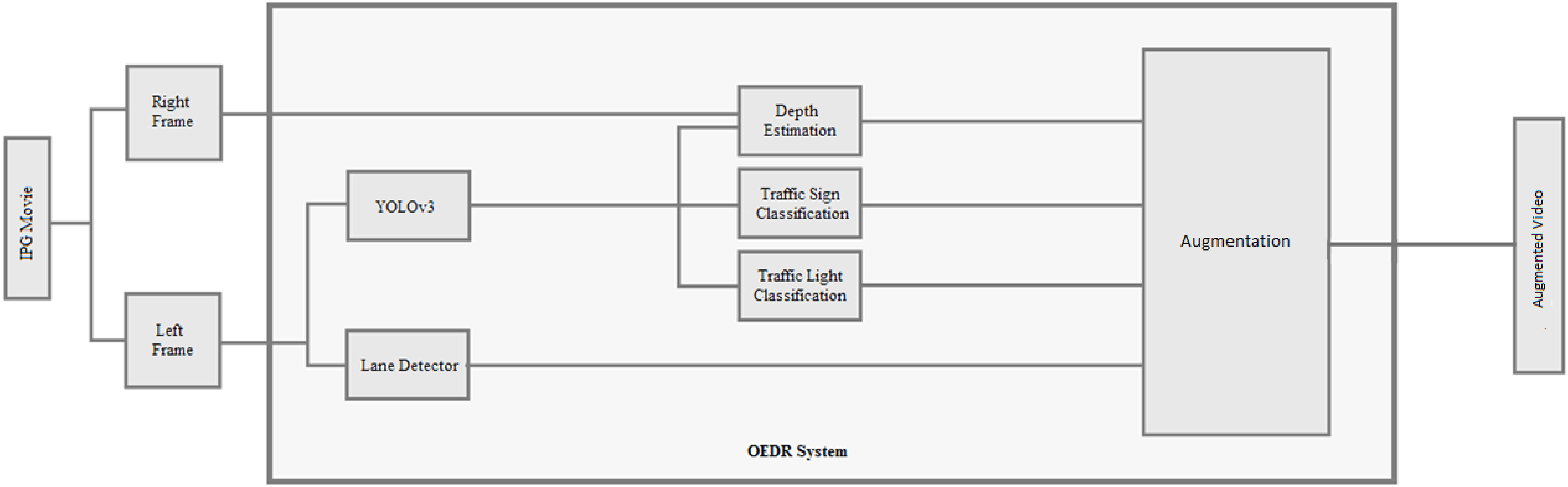
Depth Estimation

- Accuracy of depth estimation is directly proportional to the area of the image overlap.
- We can improve accuracy by increasing the field of view of camera sensors.
- Minimum distance for estimation from the camera: 3m.

Subsystem Performance Indices

Traffic Light Classification					
Detection Rate		Error Rate		Miss Rate	
96.58%		3.42%		0.00%	
Traffic Sign Classification					
Detection Rate		Error Rate		Miss Rate	
94.25%		4.60%		1.15%	
Lane Detection					
Detection Rate		One Side Detected		Miss & Error Rate	
84.38%		14.80%		0.82%	
Depth Estimation					
Ego Lane			*Outside of Ego Lane*		
Detection Rate	Error Rate	Miss Rate	Detection Rate	Error Rate	Miss Rate
73.91%	6.21%	19.88%	70.42%	2.82%	26.76%
Object Detection and Classification					
Ego Lane			*Outside of Ego Lane*		
Detection Rate	Error Rate	Miss Rate	Detection Rate	Error Rate	Miss Rate
98.77%	0.00%	1.23%	72.47%	0.00%	27.53%

Integrated OEDR System Architecture



Observations

- While considering a simulator's camera sensor, we have to keep in mind that various anomalies can be introduced when the system is imported to a real ego-vehicle.
- The accuracy of a machine learning model is highly dependent on the diversity between training data and testing data.
- These potential limitations and their solutions are highly dependent on the simulator being used in the development process and may vary accordingly.

Conclusion

- It is possible to implement and integrate essential subsystems of an autonomous vehicle to a simulator to validate its efficiency in the absence of a real working environment.
- The usage of a simulator can help develop an economical and efficient state of the art system for self-driving cars.
- It is possible to research and development into autonomous vehicles at the university level without requiring the expensive infrastructure.
- Please visit https://github.com/souvik3333/OEDR-for-autonomous-vehicle for any further information on our project.

Acknowledgement

- We would like to express our special thanks of gratitude to IPG Automotive for their continued support during the project.
  - We are grateful for the multiple academic licenses of IPG-CarMaker provided to us over this research's lifespan.

THANK YOU