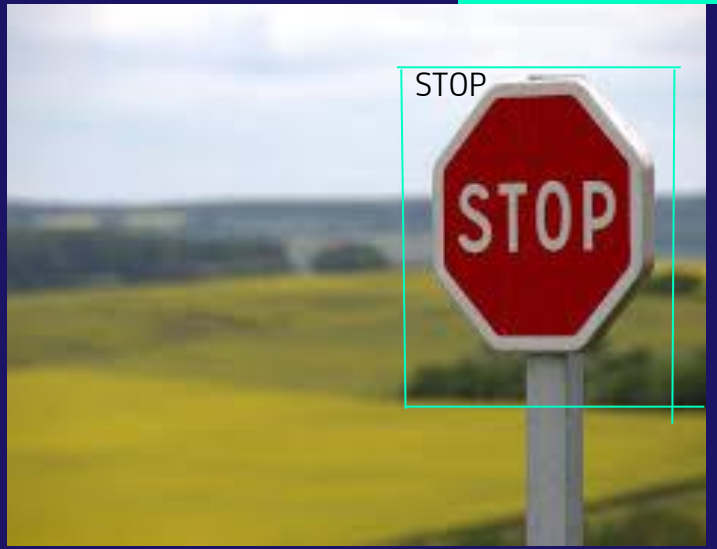
TRAFFIC SIGNS CLASSIFICATION

GUIDE - PROF. SWATI HIRA

Akshay Mishra (33)
Amey Bhutada (35)
Anand Sharma (37)
Ananya (39)
Nakul Tibdewal (50)
Sarthak Pande (86)

INTRODUCTION

- Trending Technology.
- Basic building block of self driving cars.
- Safety and Accident prevention.
- Area of research.



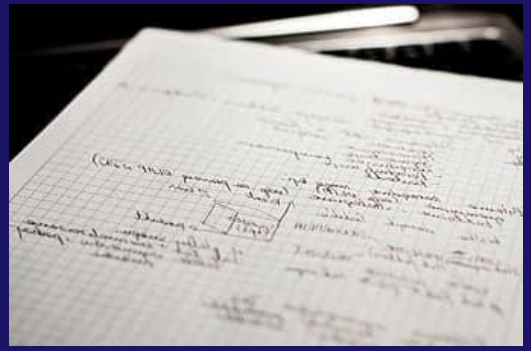
“Self-Driving cars are the natural extension of active safety and obviously something we should do.”

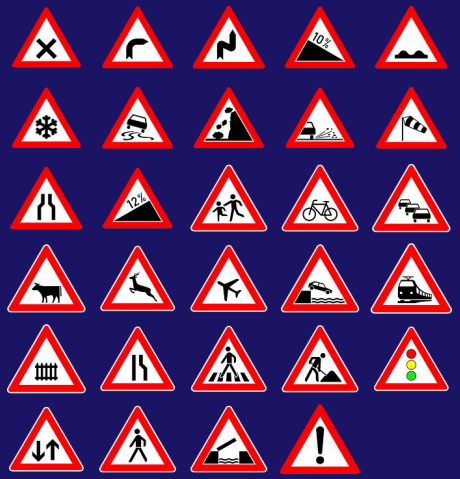
–ELON MUSK

01

Literature Survey and Data Collection

Datasets





GTSRB Dataset

Research Papers

Malaysian Dataset

Dataset for YOLO-V3

DATASET

Prohibitory
Signs which impose restrictions.

Danger
Signs that denote danger.

Mandatory
Regular signs for general purpose.

Others
Pluto is now a dwarf planet

02

Planning and Methodology

TIMELINE

Step 1

Literature Survey and data collection

Step 2

Planning of project structure

Step 3

Understanding Yolo-V3 and Darknet Architecture

Step 4

Learning R-CNN and object detection

Step 5

Training our custom data and building a model

Step 6

Integrating it with a GUI

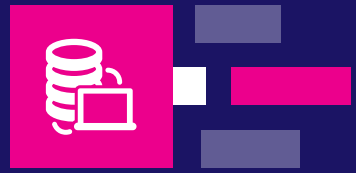
TECH-STACK

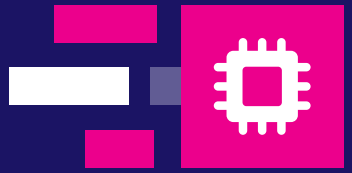
Python

Base language for coding purpose.

PyQt5

Designing of Graphical User Interface.



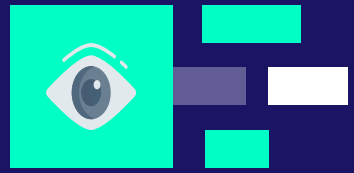


YOLO-V3 and Darknet

For implementing Real time Object Detection.

OpenCV

For interacting with images, videos and webcam.



03

Yolo-V3 and Darknet Architecture

YOLO-V3

	Type	Filters	Size	Output
	Convolutional	32	3 × 3	256 × 256
	Convolutional	64	3 × 3 / 2	128 × 128
1×	Convolutional	32	1 × 1	
	Convolutional	64	3 × 3	
	Residual			128 × 128
	Convolutional	128	3 × 3 / 2	64 × 64
2×	Convolutional	64	1 × 1	
	Convolutional	128	3 × 3	
	Residual			64 × 64
	Convolutional	256	3 × 3 / 2	32 × 32
8×	Convolutional	128	1 × 1	
	Convolutional	256	3 × 3	
	Residual			32 × 32
	Convolutional	512	3 × 3 / 2	16 × 16
8×	Convolutional	256	1 × 1	
	Convolutional	512	3 × 3	
	Residual			16 × 16
	Convolutional	1024	3 × 3 / 2	8 × 8
4×	Convolutional	512	1 × 1	
	Convolutional	1024	3 × 3	
	Residual			8 × 8
	Avgpool		Global	
	Connected		1000	
	Softmax			

Figure 1. Architecture of YOLO-3

- YOLO stands for “You Only Look Once” and uses convolutional neural networks (CNN) for object detection.
- When YOLO works it predicts classes’ labels and detects locations of objects at the same time. That is why, YOLO can detect multiple objects in one image.
- YOLO divides image into regions, predicts bounding boxes and probabilities for every such region.
- YOLO also predicts confidence for every bounding box showing information that this particular bounding box actually includes object, and probability of included object in bounding box being a particular class.

WHAT IS DARKNET?

- Darknet is an open source neural network framework.
- It is a fast and highly accurate **framework for real time object detection.**
- Accuracy for custom trained model depends on training data, epochs, batch size and some other factors.
- Also can be used for images.

04

Preprocessing Dataset

Task at Hand

- Convert Images from .ppm to .jpg
- Convert annotation file to YOLO
- Create .txt file with annotations next to every image that will have the same names as images files.

04

Training Model on custom data

DATA FILE

EVERY DATA FILE CONTAINS:

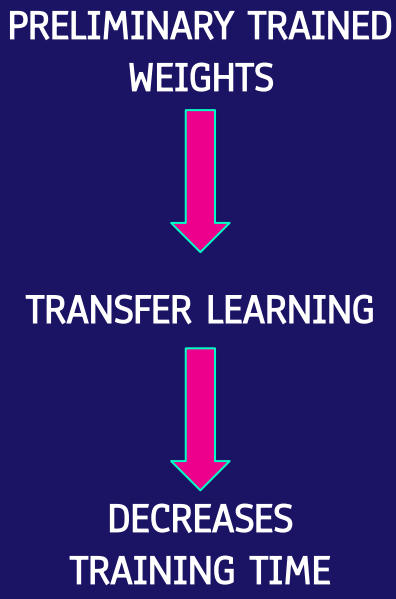
- Number of classes
- Full path to train.txt file
- Full path to test.txt file
- Full path to classes.names file.
- Full path to location where trained weights are to stored.

CONFIGURATION FILE

- Parameters for Training file
  - Learning Rate
  - Angle and Saturation
  - Exposure and Hue
- Structure of CNN Layers
  - Filters
  - Activation
  - Size and stride
- Last 3 YOLO layers

SOME ADDITIONAL CONFIGURATIONS

- FILTERS = (classes + coordinates + 1)*masks,
  = (4 + 4 + 1)*3 = 27
- MAX_BATCHES ⟶ (Total number of iterations for training.)
  = classes*2000 (but not less than 4000)
  = 4*2000 = 8000
- STEPS ⟶ (Number of iterations after which learning rate will be multiplied by scale factor.)
  = 80% of MAX_BATCHES, 90% of MAX_BATCHES
  = 6400, 7200
- BATCH ⟶ (Number of images that will be processed during One iteration.)
  =64



COMPLETE IDEA

READING RGB IMAGE

GETTING BLOB

LOADING YOLO V3 NETWORK

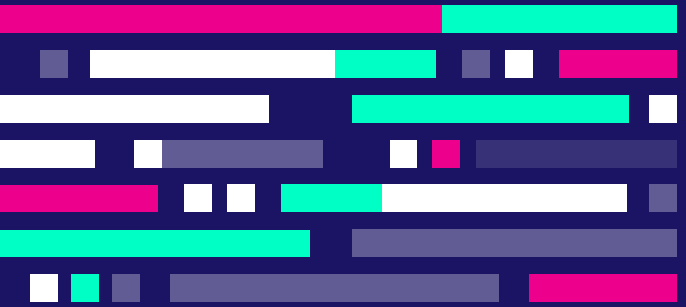
IMPLEMENTING FORWARD PASS

GETTING BOUNDING BOXES

NON-MAXIMUM SUPRESSION

DRAWING BOUNDING BOXES WITH LABELS

SAMPLES



TO-DO

- Integrating the Trained Model with a GUI.
- Increase the number of classes for prediction.
- Train and test the model on Google Colab for Real-time classification.


THANK YOU!
Questions are welcomed...