Low Level Design (LLD)

VEHICLE DETECTION

Revision Number: 1.1

Vasanth P Swati Sinha Maitry Sinha

Document Version Control

|  |  |  |  |
| --- | --- | --- | --- |
| **Date**  **Issued** | **Versio n** | **Description** | **Author** |
| **18th Aug**  **2021** | 1.1 | First Draft | Vasanth P |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Contents

Document Version Control..................................................................................................... 2

Abstract................................................................................................................................. 4

1 Introduction..................................................................................................................... 5

1.1 Why this Low-Level Design Document?...................................................................5

1.2 Scope....................................................................................................................... 7

1.3 Constraints...............................................................................................................7

1.4 1.3 Risks.................................................................................................................. 7

1.5 1.4 Out of Scope......................................................................................................7

2 Technical specifications..................................................................................................8

2.1 Classifying the Images...................................................................................…………..8

2.1.1 Fashion Mnist Dataset overview.................................................................………..8

2.2 Predicting Images......................................................................................................... 9

2.3 Logging.......................................................................................................................... 9

2.4 Deployment................................................................................................................... 9

3 Technology stack.......................................................................................................... 10

4 Proposed Solution......................................................................................................... 11

5 Model training/validation workflow.................................................................................12

6 Exceptional scenarios................................................................................................... 12

Abstract

More than two decades machine learning techniques have been applied in multidisciplinary fields in order to find more accurate, efficient and effective solutions. In that specifically deep learning is a growing multi-layer neural network learning algorithm in the field of machine learning in recent years. This research tries to detect vehicles. SSD along with mobile detection is used for the detection task. Python programming language have been utilized as the development language for this model. For the evaluation purpose multiple techniques

are used in order to compare and identify the more accurate model. The primary goal of this project is to develop a system in which the system should be able to detect the vehicles automatically.

1 Introduction

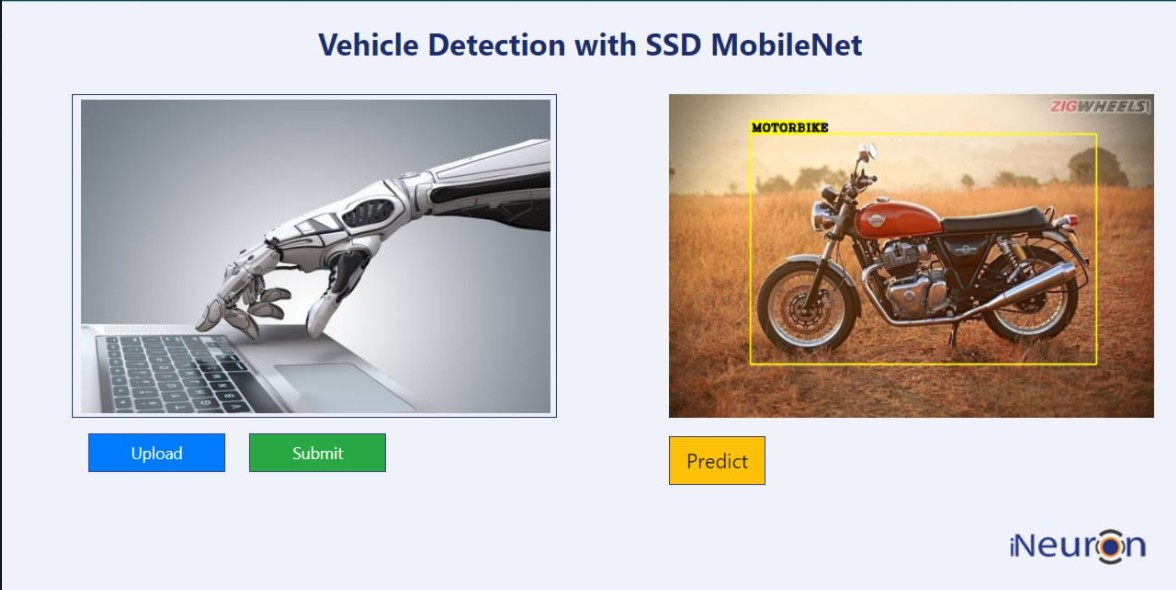
1.1 Why this Low-Level Design Document?

The purpose of this document is to present a detailed description of the Vehicle Detection system . It will explain the purpose and features of the system, the interfaces of the system, what the system will do, the constraints under which it must operate and how thesystem will react to external stimuli. This document is intended for both the stakeholders andthe developers of the system and will be proposed to the higher management for its

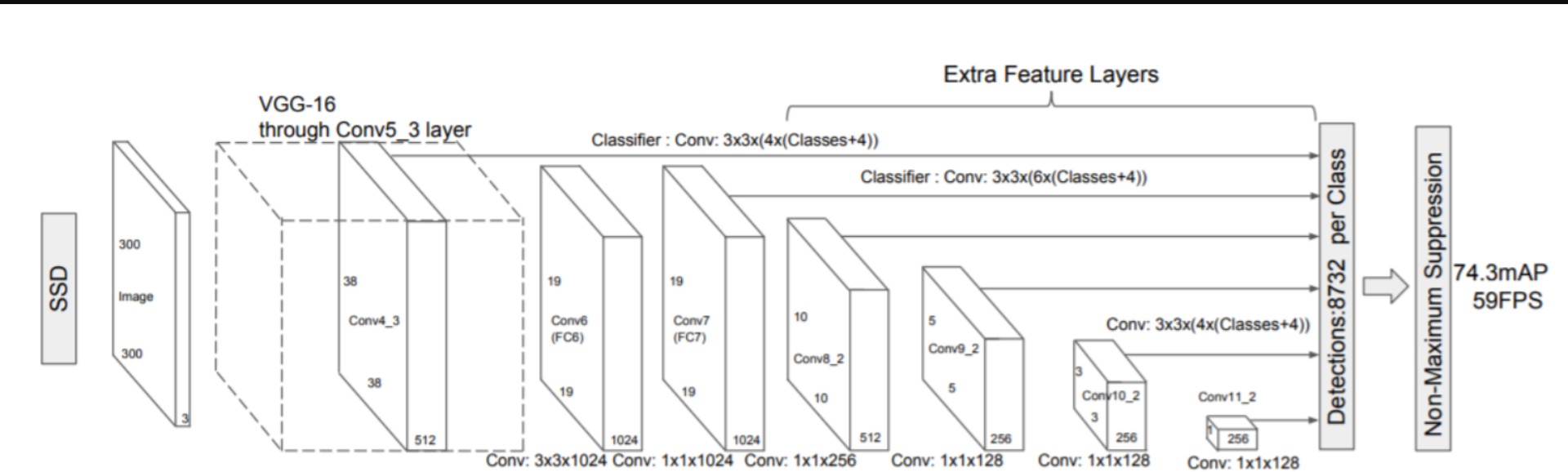
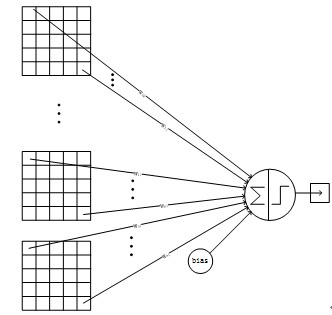
approval.

Vehicles in this world is now increasing day by day. Without vehicles most of the people don’t travel for their specific work be it personal or professional work. Now as it is increasing day by day there are some problems happening like all are to avoid traffic like moving in the wrong side of the road, running over time limit, overtaking and many more. So here we are doing an AI application which will detect the the vehicle and this AI can be applied in all sorts of the problem discussed above

The Vehicle Detection system is a web application which will detect the type of vehicles from the image which is being uploaded into the application and gives you a detection by giving a bounding box for the vehicle.



Name of Image with probability



1.2 Scope

This software system will be a Web application This system will be designed to detect the Image just by seeing the images, improved interventions, and more efficient product classification in which you can classify you different types of vehicles so that it will be easy to manage the traffic or even in autonomous car if needed to overtake this will help there because it shows the width and height of the vehicle so we can easily overtake and there

are lot other application scope for this system

1.3 Constraints

We will only be selecting vehicle types available in

1.4 Risks

Document specific risks that have been identified or that should be considered.

1.5 Out of Scope

Delineate specific activities, capabilities, and items that are out of scope for the project.

2 Technical specifications

2.1 Dataset

|  |  |  |
| --- | --- | --- |
| Dataset | Finalized | **Source** |
| COCO Dataset | yes | [https://cocodataset.or](https://www.kaggle.com/zalando-research/fashionmnist)  [g/](https://www.kaggle.com/zalando-research/fashionmnist) |

2.1.1 Fashion MNIST dataset overview

The datasets used task is COCO dataset:

The MS COCO (Microsoft Common Objects in Context) dataset is a large-scale object detection, segmentation, key-point detection, and captioning dataset. The dataset consists of 328K images.

Splits: The first version of MS COCO dataset was released in 2014. It contains 164K images split into training (83K), validation (41K) and test (41K) sets. In 2015 additional test set of

81K images was released, including all the previous test images and 40K new images

Annotations: The dataset has annotations for

• object detection: bounding boxes and per-instance segmentation masks with 80 object categories,

• captioning: natural language descriptions of the images (see MS COCO Captions),

• keypoints detection: containing more than 200,000 images and 250,000 person instances labeled with keypoints (17 possible keypoints, such as left eye, nose, right hip, right ankle),

• stuff image segmentation – per-pixel segmentation masks with 91 stuff categories, such as grass, wall, sky (see MS COCO Stuff),

• panoptic: full scene segmentation, with 80 thing categories (such as person, bicycle, elephant) and a subset of 91 stuff categories (grass, sky, road),

• dense pose: more than 39,000 images and 56,000 person instances labeled with DensePose annotations – each labeled person is annotated with an instance id and a mapping between image pixels that belong to that person body and a template 3D model. The annotations are publicly available only for training and validation images.

2.2 Predicting Images

 The system will ask you for enter image of product.

 The User will upload a images of product.

 The system will try to extract features from images and it will predict the

probability of image belonging to which class and make a bounding box surrounding the object

2.**3** Logging

We should be able to log every activity done by the user.

 The System identifies at what step logging required

 The System should be able to log each and every system flow.

 Developers can choose logging methods. You can choose database logging/ File logging as well.

 System should not be hung even after using so many loggings. Logging just because we can easily debug issues so logging is mandatory to do.

2.**4** Deployment

1. AWS



3 Technology stack

|  |  |
| --- | --- |
| Front End | HTML/CSS |
| Backend | Python/Flask |
| **Libraries** | Tensorflow, Keras, Numpy, OpenCV |
| Deployment | AWS |

4 Proposed Solution

Refer  [https://ieeexplore.ieee.org/document/9182346](https://arxiv.org/pdf/1811.04374.pdf)

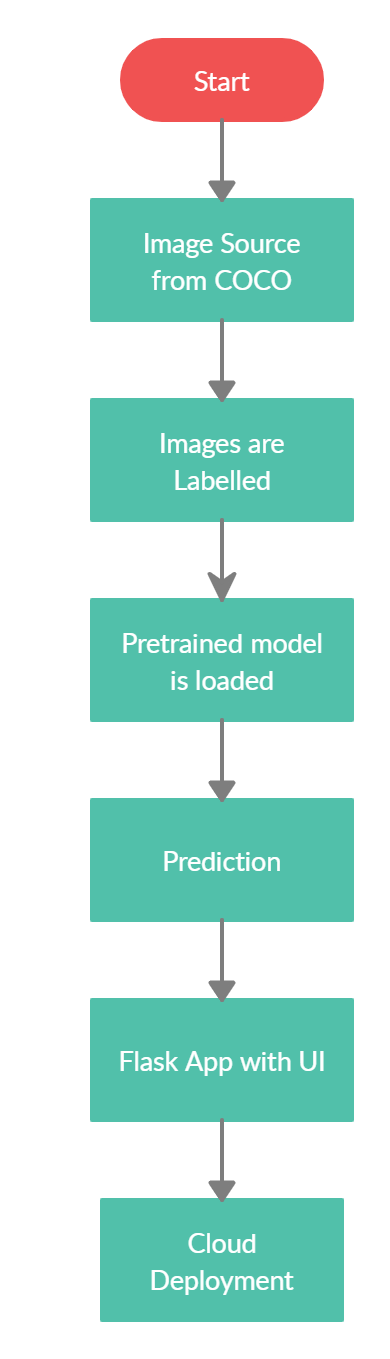
Over the past few years, deep learning has played a tremendous role in real time object detection, recognition and classification. Today, algorithms use deep learning not only for simple object detection but have also paved way for vehicle detection that is complex situation such as Indian roads. Different deep learning models can learn features of varied complexity and perform according to surrounding environment in which they are deployed.

Whereas SSD mobilenet V1 takes 31.89% less time than faster RCNN resnet101. On the flipped side, faster RCNN resnet101is 34.38% more accurate than SSD mobilenet v1.

Considering the realtime environment time complexity is given more importance by which

SSD with mobilenet is used

5 Model training/validation workflow



6 Exceptional scenarios

|  |  |  |  |
| --- | --- | --- | --- |
| **Step** | **Exception** | **Mitigation** | **Module** |
|  |  |  |  |
|  |  |  |  |

6 Test cases

|  |  |  |  |
| --- | --- | --- | --- |
| **Test case** | **Epochs to train** | **Module** | **Pass/Fail** |
|  |  |  |  |