TECHNISCHE UNIVERSITÄT
CHEMNITZ

Detection of Road Conditions Using Image Processing and Machine Learning Techniques for Situation Awareness

Master Thesis

Submitted in Fulfillment of the
Requirements for the Academic Degree
M.Sc. in Automotive Software Engineering

Dept. of Computer Science
Chair of Computer Engineering

Submitted by:	Md Asaduzzaman
Student ID:	431651
Date:	29.10.2019

Supervising Professor: Prof. Dr. Dr. h. c. Wolfram Hardt
Supervising Tutor : M.Sc. Julkar Nine

Acknowledgments

I would first like to thank Prof. Dr. Dr. h. c. Wolfram Hardt, the Chair of Computer Engineering at Technische Universität Chemnitz for letting me perform the internal master thesis under his valuable supervision. I would also like to thank my supervisors Dr. Ariane Heller and M.Sc. Julkar Nine for their enormous support throughout the thesis. The door to him was always open whenever I ran into a trouble spot or had a question regarding my thesis and report. Without his support and guideline, it would have been impossible to finish my thesis. Moreover, I also like to thank my beloved daughter, Md Sumaia for showing her patient without me. Finally, I must express my gratitude to my parents and my wife for their support and encouragement throughout my study as well as writing this report. This accomplishment would not have been possible without them. Thank you.
Author

Abstract

In this modern era, land transports are increasing dramatically. Moreover, self-driven car or the Advanced Driving Assistance System (ADAS) is now the public demand. For these types of cars, road conditions detection is mandatory. On the other hand, compared to the number of vehicles, to increase the number of roads is not possible. Software is the only alternative solution. Road Conditions Detection system will help to solve the issues. For solving this problem, Image processing, and machine learning have been applied to develop a project namely, Detection of Road Conditions Using Image Processing and Machine Learning Techniques for Situation Awareness. Many issues could be considered for road conditions but the main focus will be on the detection of potholes, Maintenance sings and lane. Image processing and machine learning have been combined for our system for detecting in real-time. Machine learning has been applied to maintains signs detection. Image processing has been applied for detecting lanes and potholes. The detection system will provide a lane mark with colored lines, the pothole will be a marker with a red rectangular box and for a road Maintenance sign, the system will also provide information of maintenance sign as maintenance sing is detected. By observing all these scenarios, the driver will realize the road condition. On the other hand situation awareness is the ability to perceive information from it's surrounding, takes decisions based on perceived information and it makes decision based on prediction.

Keywords: Lane detection, Maintenance sign detection, Pothole detection, Situation Awareness, Image Processing, Machine Learning.

Contents

List of Figures

List of Tables

List of Abbreviations

GTSDB	German Traffic Sign Detection Benchmark
ABS	An anti-lock braking system
KKN	K Nearest Neighbor
ANN	Artificial Neural Network
CNN	Convolutional Neural Network
DNN	Deep Neural Network
ABS	Anty-lock Breaking System
TSC	Tracktion Control System
SCS	Stability Control System
SVM	Support Vector Machine
SCU	Stability Control Unit
GCU	Gear Control Unit
HSV	Hue, Saturation and Valuet
ANN	Artificial Neural Network
ECU	Electronic control unit
OS	Operating System
RSS	Time to Collision
TTC	resident set size
VMS	Virtual Memory Size
YOLO	You Look Only Once
Psutil	Python System and process utilities
ROI	Region of Interest
DLL	Dynamic-link library
ML	Machine Learing
IM	Image Processing
IDE	Integrated DevelopmentEnvironment

1 Introduction

Stop the accident before they stop you. A dynamic traffic situation will be detected by this Situation Awareness System. This system includes three subsystems which are pothole detection, Lane detection, and Maintenance Sign detection. Potholes are occurred due to not to maintain properly. It may occur due to natural disgusted and road accidents. Potholes are considered more than 150 mm diameter hole is considered as potholes [6]. Potholes, lane and maintenance signs will be detected using this system. Safety and security are the major issues in our automotive domain. Driving is a risky job, for this needs high train then also many accidents occur daily due to potholes, lane marks, and maintenance signs. Though there are many causes of a road accident, these are the major causes of the road accident. Detection of Road Conditions will help to reduce the accident and it will reduce the driver's brain work also. Automotive industries are trying to move for a self-driven car or Advanced Driving Assistance system as this is the public demand. The detection system is mandatory for these types of cars. For developing such a system, various types of image processing and machine learning techniques will be applied.

1.1 Motivation

Today, cars are fast enough and convenient transport for every household. Now public demand for self-driven cars and Advanced Driving Assistance System(ADAS). So, Automotive industries are investing a lot in research to develop a Self-driven car or ADAS. Road condition detection is the compulsory part for developing a self-driven car or advanced driving system. Nearly 1.25 million people die in road accidents in road crashes each year. On average 3,287 people deaths a day. An additional 20-50 million are injured or disabled [7]. This system will help to solve these issues. The subsystems of the detection system are given one by one. Pothole detection subsystem will help to avoid to use ABS which will save our car life as well as reduce accidents. Many serious accidents accrue due to crossing the lane. The lane detection subsystems will play a big important role to reduce such accidents. Maintenance signs also an important issue. Causes of many accidents worldwide. Drivers need to follow bypass or need to reduce the speed limit for narrowing the street whenever appear the maintenance sign. This subsystem will detect the maintenance and will help to follow the instruction. This situation awareness system help to reduce accident like cars crashes which will safe the driver, pedestrian and passenger.

1.2 Objective

Gear Control Unit (GCU), Stability Control Unit (SCU), etc which are directly involved in active safety. Various technologies are already in the automotive domain for detection and recognition. For example radar, leaser, LID, depth sensor, etc. The camera is the cheapest and most effective for other purposes also. A camera can be used for observing the surrounding area of the vehicle which is most important for security and safety. The same camera can be used for the detection of potholes, lane, and maintenance sign detection. So, only need software then no more extra cost for hardware. The detection of road conditions is directly related to active safety. This safety is needed for the semi-automotive or self-driven car. It will help to reduce the accident as well as car maintenance issues which are the main objective. Image processing is the computational resources hunger, target to apply the algorithms which will reduce computation resource, and the second target to reduce the processing time so that detection can be done in real-time or near to it.

1.3 Structure of the thesis

This thesis is in 7 chapters. Chapter 1, is about the interdiction, objective and motive of the thesis. Chapter 2, is about the state of arts, theoretical and technical background of recent worked related to this project. Chapter 3, talks about concept and mythology, theoretical analysis of methods that will be applied in the project. Chapter 4, Implementation, This is about hardware, software environment, implementation techniques, programming in details. Chapter 5, Result and evaluation, it will contain the results and achievement. chapter 6, Expectation and future work, here achievement, cause of fail and future suggestion will be discussed. Finally, chapter 7, the thesis will end with a conclusion.

2 State of the Art and Theoretical background

In this chapter, the State of arts and the theoretical background which are strongly related to this project will be described. Industries are investing more and more for researching in this sector. Now, cars are fast enough. Now people want self-driven cars. But it is not yet publicly operable in the street. Many techniques and hardware are used for this kind of system. For example, various cameras, laser, radar, depth sensors, etc. Using leaser and radar, the system can get a more accurate result but these do not work properly due to road surface level and bad road condition. A combination of sensors and cameras are also expensive. Single-camera can be used for various purposes. The proposed system based on the image. Automotive industries are investing a lot in these types of applications. Various types of Image processing and machine learning techniques will be given below which are already applied for various detecting systems in the automotive domain.

2.1 Related Work

In this chapter, related work will be described in detail. This situation awareness system is the combination of three subsystems, Pothole Detection, Lane Detection and Maintenance Sign Detection. All the relater work with pothole will be presented. First, Pothole Detection, second, lane detection and the last, Maintenance Sign Detection. Safety is a major issue for the automotive domain. The potholes are the cause of many accidents. So, many researchers and industries developed detection systems using hardware, like Laser, Infrared, lidar, Radar, depth sensor, etc. Some of these can be described later. The main focus of this project is to develop a system using a simple camera. Road conditions detection is the mandatory issue for self-driven cars or the Advanced Driving Assistant System. Many actions of a car depend on road conditions. For example, if a car detects a big pothole then the system decides to press the brake and or adjust control over Anti Breaking System.

If the lane mark is not on the street then the system has to alert the driver. In the case of maintenance sign detection and speed limit detection, the system needs to make many decisions. For example, a car should follow an alternative way or need to speed down. Depending on the detection condition many actions need to be performed. May need to take a turn, put on the turning light, breaking for slow down and control to ABS also. These are the mandatory scenario for driving a car. Any kind of the wrong detection may cause of massive car accident. Many technologies are available for detecting road conditions. Currently used some technology based on image processing, machine learning, and deep learning will be disused for detecting potholes, lane mark and maintenance sign for situation awareness systems.

2.1.1 Situation Awareness

Situation awareness is a system that can precept the surrounding environment and event concerning space and time and avail to understand their meaning and depending on all these issues, it provides future state. So, the system will greater the information of the surrounding environment and based on this information, will predict the future and take the action. In a word, it states the actual awareness of the situation. It is required for any kind of high-risk job, even many low-risk works can also lead to a high-risk. For making the work pace safe and secure, the situation awareness system is essential. Driving is a high-risk job, it needs high skill for being a driver then also every year a huge number of people are killed in a road accident. So, a perfect situation awareness system is required for land transport to reduce the number of roads accident. Situation awareness system observer the situation and take necessary action. The situation awareness system can be defined into 3 levels. First level, system observes the information from the environment. In the second level, the system takes meaningful information and finally in the third level, section system predicts the future based on the meaningful information and takes the necessary actions [8]. These are the key parts of Situation Awareness System.

Situation Awareness System has been using for many years in many industries. Especially using in the avionic industry in the USA for several decades. Endsley mentions in her publication, "Theory of Situation Awareness in the Dynamic System", several accidents are also recorded but no doubt situation awareness helps a lot to reduce accident and It has made the operation easy also[1]. It has been using all over the world for avionic systems. There are many situation awareness system applications nowadays all over the world in many sectors. For example, in the chemical industry, building industry, digital economy, electronic, microtechnology, energy, financial sector, automotive, and aviation. Even, it has been using for detecting terrorists and to find the behavior of the terrorist, still, it is not in perfect condition. Researchers are working hard to build a perfect situational awareness system for many sectors. Automotive industries are investing a lot in developing high-quality situation awareness systems and researchers are also working hard for finding a solution to reduce the accident by developing a high-quality situation awareness system. Situations Awareness System is essential to reduce the number of accidents. It can predict the environment and can decide for action. Land transport is now a convenient transport for every household. The number of roads is not increasing comparing with the number of transports. It is not also possible. Day by day traffic hazard is increasing, leading to more and more accidents. The software solution is the only alternative for solving the issue. Situation awareness system can bring the solution to reduce the accident and the brain work.

Motus and other states, the process of making a complex situation awareness system based on the model.For example model for ciber security, model for interaction etc. The priority among them and works together for solving the issue in the current the situation [9]. An Expert is the most important part of the Awareness System. It takes all decision and sends the command to the controllers. It observes the information and then takes the necessary action. One of the expert systems is a rule-based system which can take fast decision based on the base of the rules and information. It helps to take the necessary action. The expert system takes the information as input such as traffic light, sign and sent the output for the controllers. The output is the decision of the expert system. The expert system provides a decision based on rules. Such a rule-based system can be built using many algorithms[10]. Rete algorithm and backward chaining algorithm are prominent of them. Let have a look, backward chaining algorithm starts work with the last element of the chain till the last element. It starts with a goal and ends with a factor that leads to the new goal. Depth-fast, a search algorithm is used for backward chaining. It starts with the data adjacent node and moves to the deeper. Eventually, it provides the answer [11]. On the other hand, rete is a rule-based algorithm which works with the massive collection of pattern and objects. Objects and patterns are in the tree structure or an index. These trees contain a huge number of objects and patterns. This system is very efficient for comparing with these massive data and find the solutions. It works based on pattern matching. It takes the information and matches with the rules whenever it finds a matching set, an action is a fire for the complete set that matches. It stores the matches factor by joining among them. It used memory very efficient way. It finds all possible solutions for the single input information.

Model

Here the model of a situation awareness system is shown in the [1] 2.1. The model describes the components of situation awareness. The core of the model is three levels. Level 1, Perception of the element in the current situation, Level 2, Comprehension of the current situation and level 3 Projection of the future Status.

Level 1 Perception of current situation - System precept information from the environment and or from the action of the actuator. All the meaningful information is stored in memory.

Level 2, System works on the available information in the memory. In this level, the system only understands the environment from the loaded in the memory or from sensors.

Level 3, At this level, the system works for future action. It can be a rule base system or can be a neural network or any kind of algorithm that can perform a correct output.

Other influences also have been taken into consideration in the decision section. In figure 2.1 has been shown system ability, interface, workload and complexity of tasks influence the Decision unit. The decision also can be changed by other influences like the Information process machine, the goal of the objective. According to the decision of this unite, the action is taken by the performance of the action unit. the performance also depends on system properties. There is an influence of ability, experience, and training through the Information processing machine.

A general overview of an awareness system for land transport is given here. With this example, it will be clear how the awareness system works. Here is a general overview of expert system 2.2, [2].

In this figure, the driver watches an expert system or get an alarm to take the necessary action. The expert system also makes many decisions and send to direct to the controller according to the situation. There are many types of sensor can be used in such a system. For example, speed mature, depth sensor, radar, Proximity, etc. Even, it could be a device also, like a camera and or another system. The camera sends an image to the computer vision unit. computer vision and another system can also send to a command to the control unit direct according to the logical operation. All of these subsystems can also send data to the expert system and the

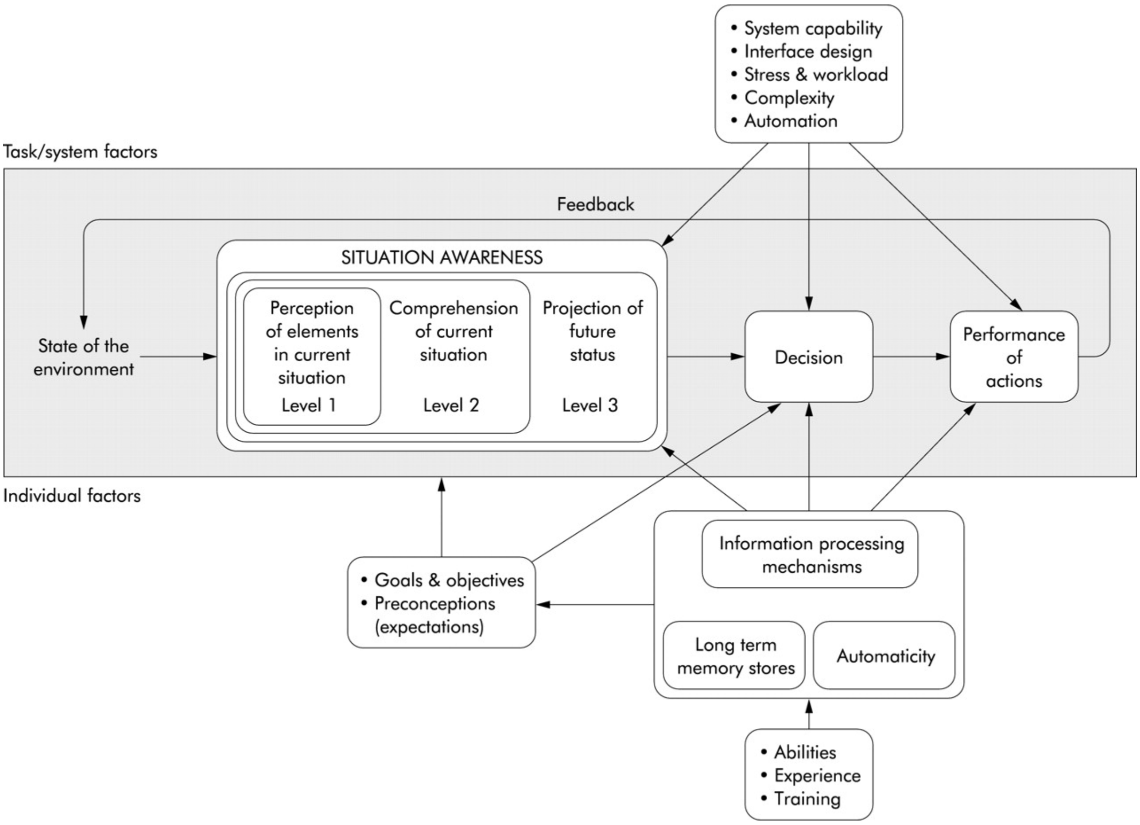


Figure 2.1: Model of Situation Awareness [1]

expert system send the command to the control unit. The expert system takes the decision based on the rule and input information.

A Situation Awareness System integrates many subsystems. Road Condition Detection for Situation Awareness System consists of 3 subsystems, Lane Detection, Pothole Detection, and Maintenance Singe Detection. These are the major issues for safety, security, and comfort. The system warns, inform and makes suggestions for the driver depending on available information. For example, whenever vehicles go out of the lane or touching the lane, the lane detection system will pass the signal to the awareness system. Same way, if any pothole or maintenance sign is found then these subsystems will pass the signal to the awareness system. The awareness system will alert the driver. Visual Technology has improved a lot. The driver or operator can watch the scenario which will help to reduce the accident. It will increase safety and security. The system can give a warning based on a real senior.

Classification

The situation awareness system can be classified in several ways. For example, most situation awareness system does not perceive pothole and traffic sign detection. One of the situation awareness systems is augmented reality based on the driving situation awareness system. The system consists of two mainly two subsystems. One is the input subsystem which consists of a sensor module, recognition module. The second one is the output subsystem consists of a vehicle status information module driving information module, time to collision (TTC) module, dangerous level decision module, warning strategy module, and display module. Various sensors and cameras have been used for detecting president, lane and front cars. The system can detect all these in the rainy weather also. It detects the danger level and collusion time also. The system warns the drive-by calculating braking time, safe distance and car speed. It also provides driving information like speed, fuel labels, etc. The system also provides various suggestions for the driver. Surely, such an awareness system will increase the security, safety, and comfort for all [12].

Traffic light detection is the most important feature of land transport. An awareness system is required for any kind of intelligent transport for a traffic light. Nine and Other mention in their publication about a situation awareness system that has been developed using a monocular camera, Raspberry Pi, PiCan and ECU. PiCan and the Pi are the components of "EC-Box" [2]. The figure is given for such a system. Here is the figure:2.2. An onboard video camera has been added to the system. Each frame has been taken from the video for detecting traffic light sign, is there or not, using a traffic light detection subsystem. Once the light is detected as red or yellow or green, the subsystem sends the signal to the ESU by CAN bus. ECU recognizes the light color as send via PiCan and this signal can be used for several actions by ECU. This is a very effective process to recognize traffic light for land transport, but it provides some false-positive results in some cases [2].

2.1.2 Pothole

Potholes can be indifference shape and size from small to large. This is usually happened because weaken of the road surface. Road weaken is happened due to several reasons, For example, movement of the uncomfortable vehicle and for the uncontrolled natural issues. Potholes could be a cause of danger if not notice prior. These could be deep and can lead to unexpected consequences. Example, the vehicle can be damaged and or hitting the people or other objects. So. these are the

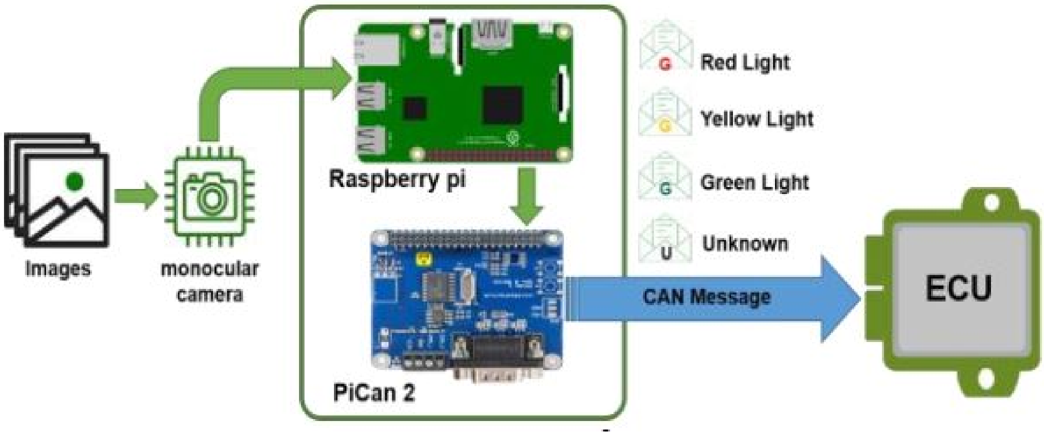


Figure 2.2: System architecture of Situation Awareness [2]

most critical issues on car life expectancy and most of the causes of the accident. Therefore, potholes detection is the most important for road condition detection and to prevent an accident. Many approaches have been implemented for potholes detecting. Some of these approaches are presented one by one. Using black box camera pothole has been detected. The frame has been converted from the video. Later on, converted to grace scale, find the region of interest and cropped. From the dark region, the area has been calculated for finding the pothole. Here is more difficulty we do not know about the camera specification. The author got 71% to 88% are the correct result. The main reason for not accurate is manhole and shade. So, we have found some difficulties here. Such a camera specification and false detection [13]. Another approach for potholes detection. From the video, frames are taken. K-means clustering approaches than by the threshold potholes are detected. This process is real-time but has been used as a Pentium computer. So, it is resource hunger. It can not be fit for a tiny device like PI. A pothole can be detected using RGB color space Image Segmentation. A pothole can be detected form different angle, irregular shapes, number of potholes also. Region of interest is only potholes area which is detected by RGB color value. One B value has been taken for detecting RGB value. Potholes are detected using four steps. 1. Prepossessing-Image resize, Image segmentation. 2.Region of interest- Here B value of RGB is used for calumniation to find the region of interest. The computational expensive algorithm has been used, need to find the standers deviation of an image pixel. 3. from the ROI interest potholes are cropped. Finally detected image is displayed. This is a resource hunger process. This test has been done in the Intel core 2.80 Gb processor with 8 GB RAM. Detection rate 82% accurate [14]. This resource hunger method is not a perfect one for implementing in tyne device Pi. Detection and Counting of Pothole using Image Processing Techniques, Image Reduce, Grayscale Conversation, Median Filtering, Difference of Gaussian -Filtering(DoG), Edge Detection Techniques Threshing techniques based on Otsu's Method. K-Means based Image clustering techniques. Fuzzy C-Means based Image clustering techniques [15], Canny edge detection, Erosion, Dilatation, Contour detection, counter draw have been used. Various clustering algorithms also are used for pothole detection. For example K nearest, SVM, Harr Cascade[14, 13, 15] "Pothole detection base on SVM in the pavement Distress Image" [16] Support vector machine is one of the popular methods for detecting potholes. For detecting potholes 80 images are taken. 50 images taken for training and 30 for the test. Each image size is 60X60 pixels. The satisfactory result has been found in their tow classes. Crack and potholes are found. Images have been converted to grayscale as color information is not required for feature extraction. In this simple method, the result is very efficient but the author did not mention the noisy image. The author also mentions correctly to identify the pothole need to train with more images which he has left as future work [16]. So, more images make the system more perfect in the case of a machine learning.

"Detection and counting pothole using Image processing Technique"[17]. In this paper k-means and Gaussian filter, image segmentation methods have been used. This is a better approach by comparing results. Image detection can be done using image segmentation, clustering, and edge detection. In this project k-means and fuzzy c means has been used and the result is evaluated by performance measure like accuracy, computation time. In the pre-processing step, the original image has been resized due to reduce computation time without losing the required information. RGB image has been changed to grayscale. Computation in the color image is much more expensive. Moreover for feature detection and shape detection, there is no need for color information also. For smoothing the image, Median filter and Gaussian filter have been used for comparison. The Gaussian filter has been applied to remove noise. Canny edge detection has been applied to the grayscale image but this process does not produce batter results. Many noises produce alone with potholes' image. But filtering with Gaussian, it produces a satisfactory result. In this paper, they consider the Gaussian filter and canny edge detector are the better options for detection. For making the black and white image, Otsu's threshold has been applied. After threshold, K-means based image clustering has been applied.

Using this method images are put in the cluster using mean value. But it also produces some grouped error. Later on, Fussy C-means based image clustering technique has been used. It helps to cluster one object which belongs to two or more clusters. Such an object input is in the center cluster with a degree coefficient which represents distance with the actual value. if the value, weight <1, consider in the center of the cluster. All the images are now in binary format. The black and white convex hull is applied to the images to find out the number of potholes in the image. It produces the black and white components, potholes individually. Finally by checking the parameter using pixel value pothole is detected. Remarked has been applied to the pothole holes. K-Means clustering segmentation produces the fastest computation and edge detection but some errors are there. [17]."Pothole Detection: An Efficient vision Based Method using Color Space Image Segmentation"[18]. In their approach RGB color image taken for segmentation. B value of RGB has taken consideration. Image segmentation and resize have been done as per-process. Arithmetic means and stander deviation has done. After applying the white picture is found with dotted black marks which are considered as seeds level. Again stander deviation is done to detect more accuracy. From here, pothole has been found and make the boundary as marking as potholes. This process is highly complicated, and the result is 82% correct and moreover, images are taken from google. According to paper, weather conditions should be sunny for detecting potholes. Matlab version 7.11.0 is used in Core to 2.80 GHz CPU and 8 GB ram. So, it indicates that it also computational resource hunger [18]. It is not perfect for our system as the target deployment device is Pi. So, let not go to more details.

2.1.3 Lane

Day by day the number of vehicles is increasing dramatically but the number of roads is not increasing, as a result, many accidents occurred daily. Many of these accidents are the cause of road lane detection. By looking at a statistic, it can realize the importance of road lane detection. This analysis data takes from the research institute of Benz, they have categorized the road accident type into 5 categories. Categories of accidents and percent are given here. Road Departure is (19%), Lane Change and Merge are (4%), Intersection is (29%), Rear End is (26%), and other reasons account for about 22%. The study also shows that over 80% accidents a relationship with the driver's careless or dozing, and if 0.5 seconds awoke can be provided, 60% possible to avoid, just need driver's awareness moreover it is possible to reduce up to 90% accident.[19]. From the statistic, we can see, 23 % accident is related to lane line management. Road departure will not happen if the driver does not go out of the lane. So, we can come to decision, road lane line detection is the core and most basic requirement of the safety system of vehicles. Day by day due to safety has become a serious issue. For this, lane detection has come a hot topic nowadays in computer vision. An autonomous car or Advanced Driving assistance system must have to have Lane detection. When a vehicle is running it is important to get the accurate detection of the lane in real-time. Image processing is one of the best ways to detect lane as the high computational resources are available. Lidar sensors are widely used but it has some limitations one of them is given here. It can not detect the object's lower height. So small object is ignored but using a camera any type of object possible to using identify alone with lane. For detecting shape and feature, color is not important. since lane detection is possible for the low regulation image. however more computational resources are needed for that. Many Image process techniques are used for detecting lane. Such as Hough line detection, Edge detection etch. some of them are given here. "Simple robust road lane detection algorithm" [20]. Canny edge detection and Hough Transformation have been used for lane detection. Canny edge detection has been used for extracting feature and Hough transformation is used for making the lane. It consists of 3 steps, pre-processing step, post-processing step and lane modeling. In the pre-processing stage, the image has been converted to a gray-scale. Erosion has been applied for making computation easy and smoothing the image. In the post-processing stage, canny edge detection has been applied for getting the feature and then Hough Transformation has been used for finding the lines. Finally in the last stage, from the Hough Transformation lined parameters are found. Grouping similar lines and averaging those values, the line is generated. This line has been attached to the original image and lane lines have been detected. Using this algorithm left and right lane marks have been detected successfully.[20].

Hough transformation is widely used as it can detect perfectly and in real-time. "A Lane Detection Method for Lane Departure Warning System" [21]. In this paper for detecting lane 3 steps have been taken. They are the Canny edge detection algorithm, lane marking, and region fo interest. Firstly they have discus several edge detection methods, like Sobel operator, Prewitt operator, Kirsch operator, GaussLaplace operator, and Canny operator. Sobel operator is widely used as its accuracy is fast and it has two kernels that work with horizontal and vertical. One kernel affects horizontal edge detection and other is a vertical edge detector. It makes the edge smooth. It is simple, efficient, processing fast and no error also. So, it has real practical implications for real-time for an autonomous car. On the other hand, Canny edge detection is also simple and effective. So, the canny edge direction has been used for their project. Secondly for detecting lane Hough transformation has been used as it has detected straight lines. This algorithm has good tolerance and robust also. Among these lines, some of them are lane lines. For finding the lane lines, the angel of the line, length of line and distance between the lane lines have been taken to the consideration. Last step, the region of interest, forgetting the whole road profile information, the whole image has been taken fast and then the region of interest has been fixed for lane detection. It also accelerates the detection process. Region of interest is a critical issue. In the case of the wrong region of interest setting, the system will be failed to detect the lane line. [21].

”Double Lane Line Edge Detection Method Based on Constraint Conditions Hough Transform” [22]. In this paper, the author has given a piece of serious information about china. Perhaps this situation is in most of the developing and overpopulated countries in the world. China is one of the most populated countries in the world. Many accidents happen due to lack of following lane lines. Around 50% of the total accident. 60% can avoidable if the drivers are alert.[19]. In this paper, double lane lines have been detected, mainly two kinds of detection methods are found. One of them is model-based and the other is feature-based. Model-based is applied for extracting the lane line and geometrical calculation alone with the driving way. Almost the same technology has been used. On the other hand, feature-based detect lane lines are using through the low-level feature, for example, grayscale conversion, finding edge point etc. For finding the edges, canny edge detection has been used. Some mathematical calculations have been done for straight line detection. RGB color values have been used for finding the region of interest. Hough transformation has been applied for finding the straight line. Polar angel has been used form -90° to 90 °. The authors have been implemented in Mathlab 2016 and the resolution of the image is 640*480. 1000 straight lines have been found for 64.5 ms video. 98.5% of them are correct and for carve line 500 carve lane line image have taken for analysis. 470 lane lines are correctly identified that means 94% of the total is correct.[22]. Bhope and others mention in their paper. From the various paper, it has been seen, no is perfect enough for real-world application. Authors mention in there papers, accuracy is more but speed is slow or vice-versa.

2.1.4 Road Maintenance Sign

Automatic traffic sign detection and recognition is essential for self-driven car or Advanced Driving Assistant Systems. Researchers are working for many years but not yet in perfect condition to detect traffic signs. Road maintenance sing is one of the most important ones among them. Anywhere in the world traffic sign provides the most important information about the road like restriction, warning, prohibited, limitation, etc. These signs are most important to avoid the accident but the driver can overlook the traffic sign which is the cause of many accidents. So, automatic traffic sign detection is the most important for the human-driven car and it is mandatory phenomena for the self-driven ca. Road maintenance sign is one of the most important for driving a car because it makes the journey safer and easier. Some applications have been developed and can recognize sing successfully but still, many challenges remain. Such as faded sign due aged, light condition, noises on the sings, etc. According to the Vienna Convention 1986 traffic sign should be 8 types and 52 countries agreed on this issue and 31 from Europe but many of them are not following the rule. Even, Vienna is also violating the rule in their treaty. It easy for a human to read and understand but not easy for machines.[23]. For the Road Condition Detection System, maintenance sing has a big role. This subsystem will detect maintenance signs and a speed limit sign also. The speed limiter is always related to maintenance sign. Whenever the maintenance sing is seen, the speed limit sign is there or a sign for following an alternative road. There are many reasons for the rapidly growing vehicle number. The vehicle has come convenient transport for every house. So, more and more cars on the road. It has become a more and more complicated issue for the traffic system. People expecting such a Driving Assist System that ensures traffic safety. Many applications have been developed but not yet sufficiently accurate. Not yet publicly is operated anywhere. Road condition detection is the most complicated thing for various scenarios. Researchers are trying to improve in this field. Many ways to detect the traffic sign. First of all traffic sign categories. There are two types of road signs, 1. Main sign and auxiliary sign. The main sign also divided into many subcategories. For example warning sing, prohibition signs, mandatory sign, guide sign, tourist sign, and road construction safety sign. Traffic maintenance sign or construction sign is the mandatory sign. For the proposed project, more emphasis has given for detecting traffic maintenance signs. Whenever a traffic sign is detected, traffic limit signs are also there. For the proposed project main target to detection traffic maintenance sign. Traffic sign detection is related to active security. In the case of detection fail or false detection can lead to an unpredicted consequence. A system is a needed which will give high accuracy in real-time is required for this type of detection system.

There are many approaches to detect traffic signs. Detecting from the image there are three types of techniques, 1. deep learning 2. Machine learning and 3.image processing. This project is about machine learning. Why machine learning is the perfect selection. Let's discuss this issue. Deep learning takes a long time for training, the algorithm takes a long time level data or unlevel data and also resource hunger. It can be done with unlevel data also then also it has some complections. Deep learning works with unlabeled data as it can differentiate from the input and eventually it finds the out from its feeds dada. It learns from the errors. So, this type of algorithm is more intelligent than machine learning. Such an algorithm is needed for the complex type of data. For example, finding cats and dogs form animal images. It is really difficult for its size, color, shape, and environments but for traffic sign detection is not that much complicated. Shape and color are the same for one category. If it is different then also not that much. All data can be possible to label in a good way. So, machine learning is the best option for this project. Detecting road maintenance sign is the only category. It is not so complex to level all possible different images for different classes. For this reason, machine learning has been chosen. It is possible also using image processing. Need to use template matching but many computational is required for each image. Computations are needed to find out the triangular from each image. Images of traffic maintenance signs are not the same due to different background and noise. Unpredictable noise on the rectangular. The image can be faded also. Many parts of the image can not visible due to a tree branch or another obstacle. So, all these issues are needed in consideration for pre-processing the image. All these are required expensive calculations for each frame. After pre-processing with a clear maintenance image is possible to find the maintenance sign. For these reasons, image processing is needed for this project. The machine, K nearest neighbor KNN has been taken for developing the maintenance sign detection subsystem.

Some machine learning algorithms are presented here for comparing among them. There are various algorithms for detecting traffic sign have applied already. The support vector machine is the most popular one. Using AdaBoost for color segmentation and Hough circle detection has been applied for finding circle and Support vector machine have been used to filter the circlet. For preposing, the image they have converted RGB to HSV color space and the marge H and S implied as a result of the output image is a binary image. The proper circle is fit for the image and by using the voting techniques. The same way triangular also has been detected. With this boundary box, the inner edge image has been developed. The circle and triangle inner image are also taken for consideration. By using these techniques more than 95% is detected correctly. For testing, they have taken 200 images. The authors have been used Del Latitude E6400 computer. For separate image they have found the different sing they have found the different result . 2 sets of the images have been taken for testing first set warning sign, 111 images and second set prohibition signs, 103 images have been taken. For the first set they have been found 98.4% and for the second set 97.0% correct. This approach provides very good accuracy but real-time processing is also an important issue which is not mentioned in the paper.[24]. So,the processing time to detect lane is not clear.

SVM is used widely for traffic sign recognition. For color segmentation images have been converted to RGBN to eliminate the noise. It just the average of RGB value for each color which is given below. on using thresholding has been converted to binary images. SVM classifier has been applied. A total of 163 images have been for testing and training. 90% for training ,10 % for testing. The author has found 100% accurate results and comparing with another algorithm, this is the best algorithm [25] but the images are selected. So, it does not apply to the unpredicted environment. Even it might not work in many cases.

Several methods have been presented above with their comparative for all 3 subsystem. These also have been discussed. Images have been changed to grayscale and then Gaussian filter or median blur can be used for smoothing the images and to reduce the noise. The maximum part of the images is not needed for detecting lane. So, the region of interest also a big factor for reducing calculation. It is clear that using edge detection techniques, edges can be detected and a Hough transformation algorithm can be applied on the edged image to detect lane. For cross-checking and finding the other lane, the width value can be added with the detected lane. Many papers have been discussed. Most of the authors have been used canny edge detector for edge detection and Hough Transformation has been used for straight-line detention.

2.2 Theoretical background

Computer vision and image processing are both involved in doing computation on images, where image processing techniques are used to transform images as required like sharpening, highlighting the edges, on the other hand, computer vision generally focuses on giving decisions based on the processed images. This chapter will describe the theoretical background of the above topics. It will give an overview of the field of computer vision and provide details about OpenCV. Also, it will state the different image processing techniques, Machine learning and little bit about Deep learning. With the rapid development of image processing techniques, digital images are using more and more in various types of applications. Such as medical science, law, and force, automotive domain, for various types of security and safety systems. Image processing is playing a big role in the intelligent vehicle system. Image processing is a method that is applied to an image to carry out some information or and another type of image. Image processing usually is done for carried out some important pieces of information which are needed. There are two types of image processing analog and digital image processing. Analog image processing like printout and photograph. This type of analysis deals with a hard copy. In this case, the digital image is processed by the computer. Nowadays image processing technology is growing fast. Various types of image processing techniques have been used for Lane, potholes. Maintenance signs detection subsystems subsystem has been developed using machine learning which is widely used all over the world. Machine learning will be discussed in another section. Details of various image processing techniques that are related to detect Potholes, Lane are given below in a separate section. In the last section, A bit about the Neural network which is related to deep learning but it will help to understand for chosen machine learning for traffic Sign detection. Among all libraries, OpenCV is the most vital part of this project which will be discussed in more detail along with Computer vision in the next section.

2.2.1 Computer Vision and OpenCV

Computer Vision is the science and technological study of acquiring and analyzing images to produce outputs for decision making. Using the computer vision application, computers can understand and differentiate their surroundings. The computer vision applications already have started their journey with everyday life. For example, as smart vehicles can detect traffic signs and control the car, Mobile phone cameras can detect objects and set the focus to the object, Many computer applications differentiate between different objects like dog and cats. But still there is a lot of scopes where a human can understand the situation, yet computer struggles to do so. Because of the very genuine image processing and decision making by the human eye and brain respectively, people can distinguish between objectives and take decisions very easily. Researchers are trying hard to overcome these differences in the field of Computer Vision, hoping a computer will be able to perform as well or better than humans. Such effort and success in this field can bring independent applications to reduce human effort in a different situation. These applications can enrich lives, improve productivity, and even help protect peoples. Even though computer vision application can manage to provide many advantages, the utilization of resources and code complexity for real-time implementation of these applications are increasingly high. Also, limitations increased by the use of a hardware platform upon which computer vision applications would run. Real-time implementations need more frame rate processing depending on the field, for which the applications are developed. To overcome the difficulties and limitation, many computer vision project has been introduced and most popular till now is Open source Computer Vision project, OpenCV, which allows developers to explore the applications that use the results of computer vision while at the same time allowing researchers to continue to improve the underlying implementations. Many functions have been used from OpenCV in this project. In the next section, OpenCV will be discussed.

OpenCV

There are many computer vision libraries out there, yet most popular in use which has plenty of algorithms and convenience methods in-built is none other than the Open Computer Vision or OpenCV library. Even though the first stable version of the library is publically released in 2006 with the collaboration of Intel and Computer Vision researchers [26], beforehand it was initiated by Intel in 1999 and has released a few beta versions from 2001 to 2005. The most used and popular version of this library, OpenCV2, was released in October 2009. Currently, it is owned by OpenCV.org and released it's 4.1.1 version which contains almost all the necessary techniques and algorithms to perform several image and video processing tasks. As an open-source library many authors contributing to its development which helping the library to gradually improving more and more.

OpenCV contains different modules that are dedicated to different areas of computer, Specially it has more than 2500 algorithms related to Computer Vision and Machine Learning, and it is expanding day-by-day. The library was originally written in optimized C/C++ and can take advantage of multi-core processors. For attracting more users, several wrappers available to it for other programming languages such as Python, Java and MATLAB/OCTAVE, C# and Perl. Even accelerating the functions in hardware, many attempts have already taken. A CUDA-based and An OpenCL-based GPU interface have been in progress since 2010 and 2012 [27, 28]. Xilinx also has published its synthesizable library of OpenCV for easy implementation and accelerations of the OpenCV functions in the FPGA.

2.3 Image Processing Techniques

Image Processing is a technique to improve raw images received from cameras or sensors placed on satellites, space probes, and aircraft, or pictures taken in normal life for various applications. [29]. Image processing divided into two subdomains analog and digital. The main concentration on this work related to digital image processing, which has been on the focus of many researchers for quite sometimes. The digital image is transformed into another image by playing with pixel-wise operations (contrast) and the local operations (extraction or scaling). The resultant image can be further used by a computer vision algorithm to generate more sensible output. There is a lot of techniques for image processing, the following paragraphs describe only a few of them which are related to this work.

2.3.1 Geometric Transformations of Images

The transformation of images plays an essential role in the reprocessing for image processing applications. This includes image scaling, rotation, translation image to different positions or perspectives. Often images need to be oriented or resized to a specific resolution to be processed by the applications, this processing comes handy in those situations. This technique has been implemented in this research project.

Resizing image

Image Resizing acts similar processing like scaling the image. The size of the output image for a given image can be defined either as a scaling factor or manually which may introduce distortion in the output image. Remapping is kind of similar process as image resizing which used for correcting the lens distortion or rotating an image. This process most widely used in various applications and has has been used in this project also. This different conversion of image resizing from a one-pixel grid to another shows the effect of image interpolation. Interpolation applies to determine the unknown pixel in the images by using the data of surrounding pixels. It estimates the approximated pixel's intensity based on nearby data so can it provide the best result for a smooth transaction between the images. There are different types of interpolation available and each type uses a different algorithm to process the data. The two most used interpolation algorithms are Area interpolation and linear interpolation, used for shrinking and zooming an image respectively. Shrinking reduces the number of available pixels in the images whereas zooming increases the pixels to provide more details of the image. Adaptive and non-adaptive interpolation are the main categories for image interpolation. The non-adaptive method uses the same method for processing each pixel, but on the other hand, the adaptive method always changes based on what they are interpolating.

2.3.2 Conversion of Colorspaces

Images are stored in memory in various color spaces. Though the most used color space is RGB, one might need different color spaces which is better suited for the purpose of the implemented application. Example as the use of grayscale images to simplify the processing. For most of image processing applications colorspace conversions is the primary step to go on. Two of them have been used in this project. Some of the most popular colorspaces are described below.

Grayscale

A grayscale image most simplified image where only intensity information is available. An image can have 256 distinct shades of gray, each one just a little bit lighter than the previous one. So, for each pixel of the images only need one byte (8-bit) memory to cover all shades. Grayscale images are presented as a 2D array, where only information of width and height are presented. This array can be described as a channel where the intensity of white is stored. Most of the image processing operations do not need full-color information. It is just sufficient to have a single channel only. Grayscale images are used widely even it has less property, but it processes faster and contains the necessary information to give a sufficient output.

RGB

Red, green and blue being primary colors can be mixed and different proportions to form any color. RGB (Red, Green, Blue) is the most widely used colorspace, which has 3 channel. Each channel contains the intensity information for Red, Green, and Blue. So, it needs a total of 3 bytes for 3 colors to represent each pixel, thus it can illustrate more than 16 million distinct colors.

BGR

This colorspace is the same as RGB except, the order of each color. OpenCV load images in this format. This is a simple issue but needs to keep in mind to avoid an error. Accept this naming convention, There is no different from RGB [3].

HSV

The HSV color space also has 3 channels but they represent Hue, the Saturation and the Value, or intensity of the image. The Hue channel represents the "color". For example, "red" would be a color. Light red/dark red would not be a color. This channel has wide uses in image processing. The saturation channel is the "amount" of the color it has, light red and dark red have different saturation. And finally, the intensity is the brightness of the color e.g. light green or dark green are different). This is very important for many color research projects or applications but this is not required in this project. An example model of colors representation in different colorspaces has shown in Figure 2.3 [30]

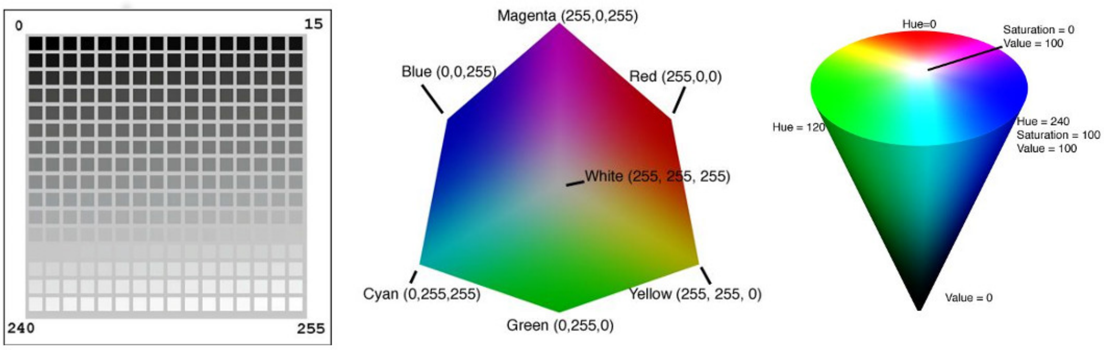


Figure 2.3: Color representation in Grayscale, RGB and HSV colorspace [30]

2.3.3 Image Filtering and Blurring

Image filtering and blurring processes play a very important role in image processing applications. It may produce realizable outputs by the human eye, but those images have significant importance for further processing. One-dimensional images also can be filtered with various low-pass filters (LPF) to remove noise or blurring the image. It removes high-frequency content (e.g: noise, edges) from the image resulting in edges being blurred when this filter is applied. High-pass filters (HPF) work in reversibly to find edges in an image. Image filtering can be applied in different ways, but most useful one by which many types of filters, blurring effect and even sharpening can be achieved is 2D convolution filter. Convolution is the process of adding each element of the image to its local neighbors, weighted by a kernel, small convolution matrix. Depending on the kernel matrix, many filtering, blurring, sharpening, embossing effects can be easily achieved.

Averaging

This is done by convolving the image with a normalized box filter. It simply takes the average of all the pixels under the kernel area and replaces the central element with this average. A 3x3 normalized box filter would look like this:

$$k = \frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \tag{2.1}$$

Gaussian Filtering

Gaussian filtering is highly effective in removing noise from the image. It is also used to blur images. For this filter, the Gaussian kernel is used. Gaussian kernel coefficients are sampled from the 2D Gaussian function.

$$G(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \tag{2.2}$$

Where σ is the standard deviation of the distribution. Here is the equation for stander deviation 2.2. The distribution is assumed to have a mean of zero. The Gaussian filter is a non-uniform low pass filter. The kernel coefficients diminish with increasing distance from the kernel's center.It most widely used filtering technique. Gaussian filters might not preserve image brightness.

Median Filtering

Median filtering computes the median of all the pixels in an original image under the kernel window and the central pixel is replaced with this median value. This is another most frequently used filtering technique. This is highly effective in removing salt-and-pepper noise. The central element of the kernel is always replaced by some pixel value in the image which reduces the noise effectively.

2.3.4 Image Segmentation

In computer vision, image segmentation is the process of partitioning a digital image into multiple segments (sets of pixels, also known as image objects). The goal of segmentation is to simplify and or change the representation of an image into something that is more meaningful and easier to analyze [31]. Segmentation partition images into smaller parts that strongly co-related with each other. Segmentation is generally based on either discontinuity of intensity where intensity changes abruptly or similarity of intensity. With the first one, it locates boundaries like lines, curves, etc. Mostly output of image segmentation is a set of contours.

Hough Transformation

Hough Transformation is a feature extraction method. It is used for line and shape detection. It detects the line between two edge points. It can detect the edge even the edges are So, for getting this line. Edge detection must be applied before starting to apply Hough transformation. It is applied after the edge detection algorithm. For example, after applying a canny edge detector or Sobel edge detector algorithm, It is used. Hough transformation is widely used. It is the most important image processing technique and widely used. Visiting every pixel in the image for finding the line is an expensive computation. Hough transformation is a voting processing that helps to find the prominent line from the image. Hough transformation is based on line detection. Here is xy is the line. The angle of the line or detection is also possible to find by some mathematical calculations. Here is the figure: 2.4 [3].

Thresholding

Thresholding is the simplest method of image segmentation. From a grayscale image, thresholding can be used to create binary images [31]. Several popular methods for thresholding are used including the maximum entropy method, balanced histogram thresholding, Otsu's method (maximum variance), and k-means clustering. Most algorithms involve establishing a threshold level of a certain parameter. Correct thresholding leads to better segmentation. The simplest thresholding set a specific point, below which the image pixel counts as black, whereas for greater than that value considered as white. It provides a binary image as output.

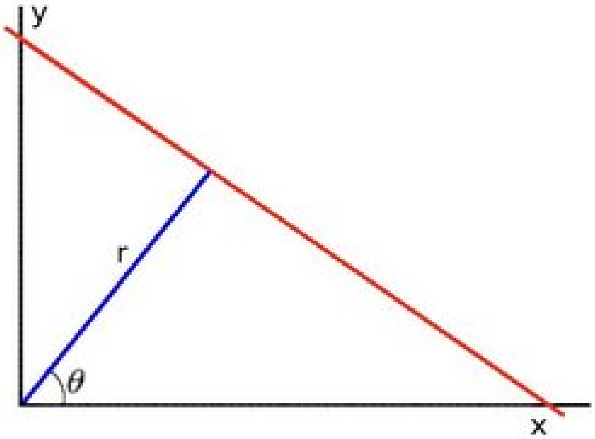


Figure 2.4: Hough Line Transformation[3]

Absolute Difference

It produces a binary image like thresholding, but the image generated from differences between two images. The absolute difference is a measure of the dissimilarity between images. It is calculated by taking the absolute difference between each pixel of the first image to the corresponding pixel in the second image. These differences are summed to create a simple binary image. The absolute differences used for different purposes like object recognition, motion detection, presence of objects, motion estimation for video compression.

2.3.5 Edge detection

Edge detection is most important to identify the feature in the image. Edges are a huge local change in the image which is used to analyze the feature. This usually occurs on the boundary between the two regions. This is the very first step for getting feature information from the image. There are many types of edge detection algorithms available but due to its importance, it is an active research area. Edges are significant local change intensity in the image. In real-world roof or step can be an example for one dimension edge profile but due to sensitive device, such an image is not possible. Photographic devices are capturing a lot of low frequencies also. As a result, many edge lines detected So, edge detector can be defined as it is an algorithm that produces a set of edges from an image. Many edge detection algorithms have been developed. Every one has its pros and cons. For example, some of these algorithms can miss a few edges but no error. On the other hand, another algorithm is detecting with the error or false edge but no missing. Here is the most popular edge detecting algorithms are given, Canny edge detector, Sobel operator. Prewitt operator, Laplacian operator, etc. Among these, Sobel and canny edge detector are most widely used. Sobel operator is faster than the canny edge detector, but it is a more tolerating algorithm and producing more accurate results. A figure of canny edge detection is given below 2.5, [3].

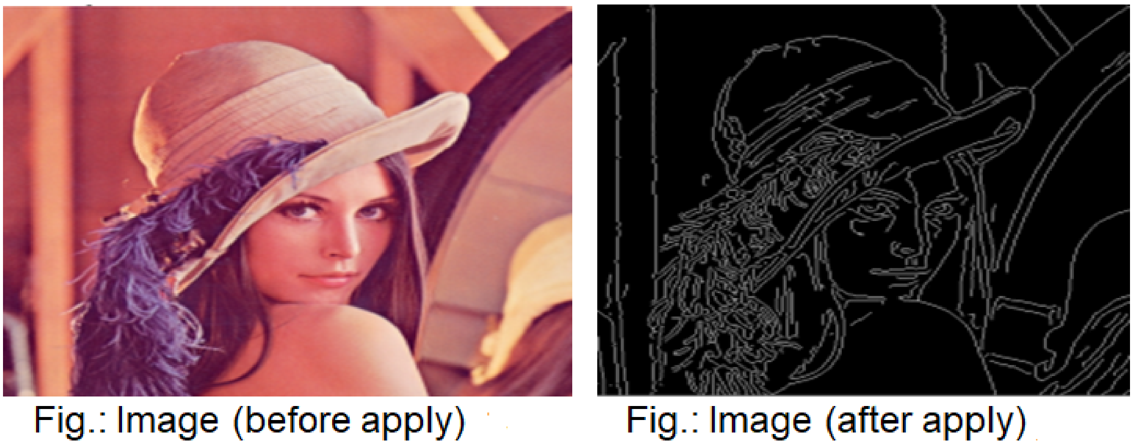


Fig.: Image (before apply)

Fig.: Image (after apply)

Figure 2.5: Canny Edge Detection [3]

2.3.6 Morphological Operations

The resultant images from image enhancement processing or image segmentations like thresholding can have more noises and imperfections. Morphological operations help to remove this type of imperfections that arises complications in further processing. Morphological operations done relative ordering of the pixel rather than relying on their numerical values. A small binary image provided as the kernel to work on the main images. This kernel is called the structuring element. Based on this structuring element the output image is shaped. Morphological operations are mostly applied in binary images. Sometimes it is used in grayscale images as well but results are not that much fruitful. The primary morphological operations are erosion and dilation. These are the most widely and frequently used techniques.

Dilation and Erosion

Dilation process adds pixels to the boundaries of the objects in an image, which makes objects more visible and fills in small holes in objects [32]. As the dilation process adds more pixels to the edge, it affects the intensity and creates a blur effect. Erosion removes pixels from the object boundaries and makes an object's boundaries shrink and invisible. Erosion is used to remove the noisy and gives a sharpening effect. The number of pixels is added or removed from the objects in an image depends on the size and shape of the structuring element used to process the image. In the morphological dilation and erosion operations, the state of any given pixel in the output image is determined by applying a rule to the corresponding pixel and its neighbors in the input image. After the dilation process, generally, a basic erosion process is applied to get a noise-free state, the process is called opening. Similarly, after erosion, a basic dilation process is applied, which is called closing. Both of them are morphological operations. Here is the figure:2.6,[3]. In the figure, a,b,c are for original image, Erosion and dilatation respectfully

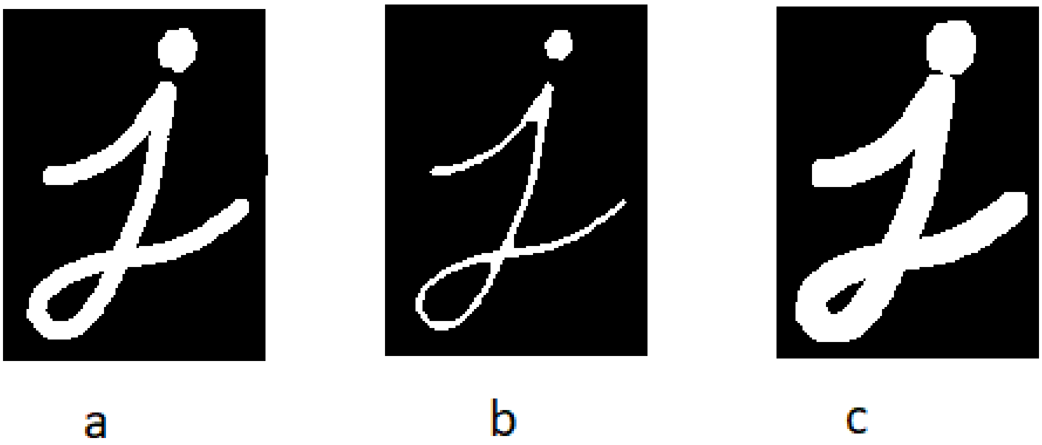


Figure 2.6: Morphological Operation [3]

2.3.7 Contours Detection

Contours detection is an essential process to detect objects, image segmentation or any other kind of image processing. Contours can be described as the curve or edges having the same color or intensity, joining all the continuous points resulting in a polygon. The contours are a useful tool for shape analysis and object detection and recognition. Contours detection is just not the edge detection, the algorithm behind that does indeed find edges of images but it also puts the joined edges or polygons in an array. From that array, any outer borders of objects detected in images can be called to do other image processing, calculation or make a decision using that.

2.4 Machine Learning

Professor Emeritus Arthur L. Samuel was the pioneer of Gaming and artificial Intelligent. He introduced the term ”machine Learning” coined the term Machine learning. It is the field of study that gives the ability the computers to learn without explicitly program [33]. Now it is part of computer science. The machine learning is used for pattern recognition and computational learning theory. This uses data by applying the algorithm and Model to predict something. The main idea for machine learning is to understand the pattern and structure of data so that in future unknown data can be matched with one of these. In this modern era, the machine learning application is widely used. Now it is not the new field of study. The basic concept of machine learning is old but it was not applied due to lack of powerful computation. Algorithms take huge computation power. So, it was a big problem in an early time. Nowadays, the hardware is powerful and not very costly also. So, Machine learning is getting popularity. Though many projects are developed using deep learning also. In every case, many procs and corns are there. Machine learning and Image processing have been considered for the proposed system development.

General overview of Machine

Machine learning can be classified into three categories, Supervised, Unsupervised and reinforce. Again, supervise can be classified into two types, Regression and Classification. Unsupervised only clustering or grouping and reinforcement. All these will be presented below one by one. Here is the picture of the classification of machine learning, figure : 2.7 which will give overall idea approaches.

2.4.1 Machine Learning applications and Approaches

Here are some applications for understanding it’s importance and then approaches. Machine learning is using in the different fields of the automotive domain. Many subsystems of all self-driven cars have been developed using many Machine learning algorithms. Some most important machine learning algorithms will be listed below. Machine learning is playing a big role in the modern digital security system. For example, we can mention fingerprint, face detection.It is using for tracking objects by combine cameras and electronic devices like radar, leaser, and other sensors, In the medical sector, many applications successfully applied. using machine learning identify and diagnose disease perfectly. Many applications identify disease by analysis of the images. Deep learning is also part of artificial intelligence. Many

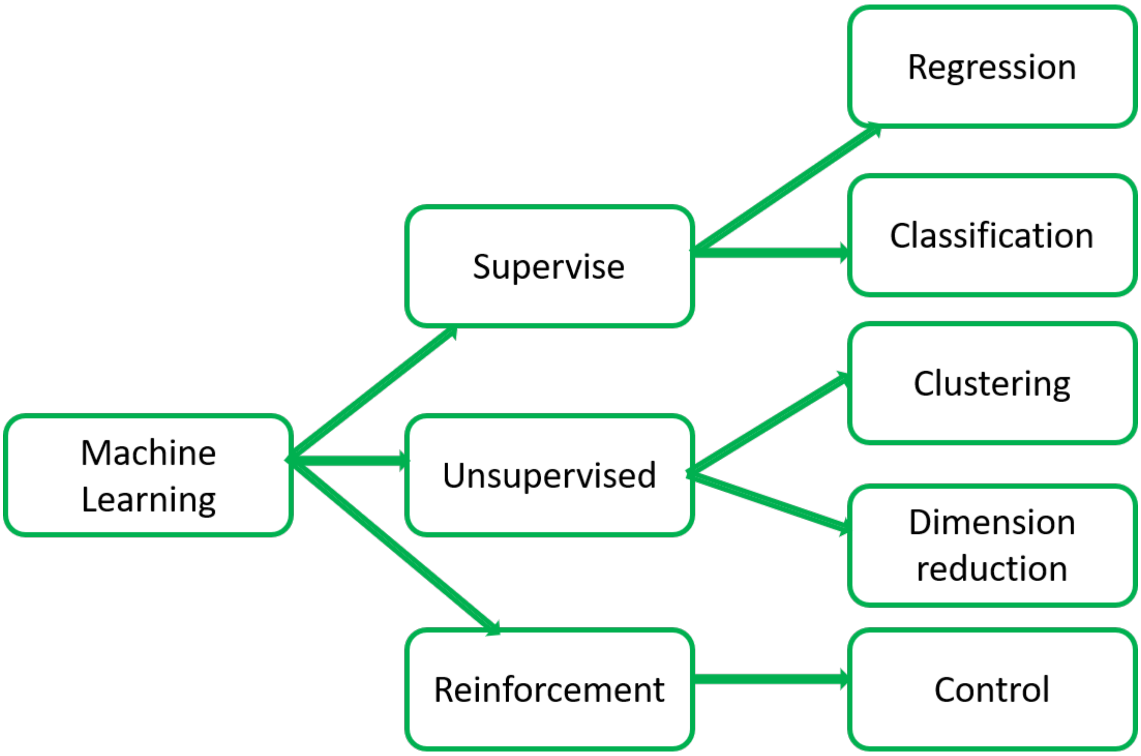


Figure 2.7: Classification of Machine Learning

applications using deep learning. Microsoft face detection engine, google translator, etc. It is also used in the financial market. Previous data is used to predict future opportunities. It is a great application for the investor. In the computer vision field, it has been used for many years. To classify an object, color, shape, distance between the object and to many others, Many applications use in the oil and gas industry for finding a new mine. It cut off a huge cost. Machine learning method or techniques teach the computer to do what come naturally like a human. The learning of machines or computers can be based on data, Image or something else. There are mainly three major techniques. 1.Supervised learning, 2. Unsupervised learning and 3.Reinforce learning.

2.4.2 Supervised learning method

The supervised learning method is the most popular method. In this method, the model is built from a training data set. The training data set contains features and possible solutions. The main idea behind this, the system has some solution and patter answer for a future problem. The system predicts new input data from the training data. A new problem comes to the system then it starts to analyze and predict possible answers form the training data. Data collection and labeling is part of data preparation. Model is the generating from this data labeling. Now new data can be predicted using the model. The workflow is given in figure 2.8.

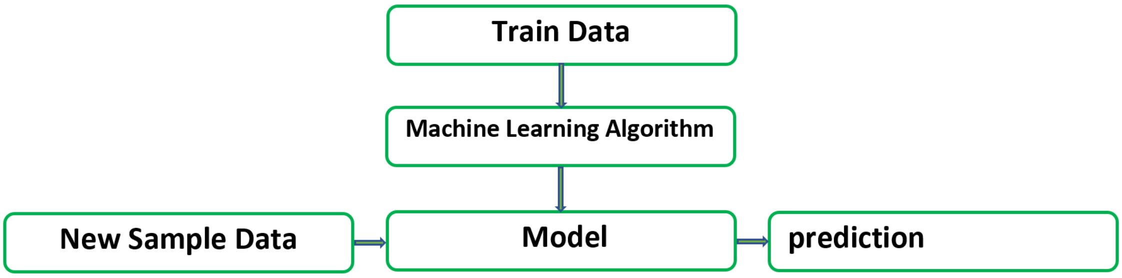


Figure 2.8: Workflow of Supervised learning

Classifications

At the time of training, the system learns about the training data and its classes. Whenever new data cames it finds the match with classes and predicts data. For example, if a system is trained with orange, apple and charry then if a new data come then the system predicts whether it is an orange or charry or an apple.

Regression

This technique is based on statistics. For example, the Price of land is based on locality and the amount of area. The output depends on the prediction function. Here, the price depends on the prediction function. Regression is normally based on a decision tree, fizzy classification, and artificial neural networks.

2.4.3 Unsupervised Learning

This technique does not need a leveled training data set. It works with unlabeled data at the time of training. The algorithm finds the hidden pattern of the data set. There are 2 types of unsupervised learning.

Clustering

In clustering data are grouped into meaningful categories. The idea behind clustering to make various categories for covering the pattern. These clusters are arranged in a meaning full way. There are few clustering algorithms, for example, artificial neural network, k-means, genetic algorithm, Gaussian mixer models, etc.

Dimension Reduction

Every element has various dimensions. In this method, the system takes as little as possible dimension without losing the information of the correct element or object. It can be used at the taring time of the support vector which speeds up the training task [34]. Alone with this, it has many other applications in image processing.

2.4.4 Reinforcement

This algorithm learns from the environment changes. This learns from the experience. It takes whatever positive from the output. The result this is going to be added by the policy and negative result also going to be reduced by the policy. So the system is getting more and more intelligent, how to give positive feedback from the previous result. This helps the system to put the provable answer. This system is controlled by the policy, not by the pre-leveled classified data or images. It takes the input frame and produces the output and from the output, the system gets knowledge about the issue. whether the action is positive or negative. Just need to control the network with positive data. This is a continuous learning approach.

2.4.5 important Algorithms

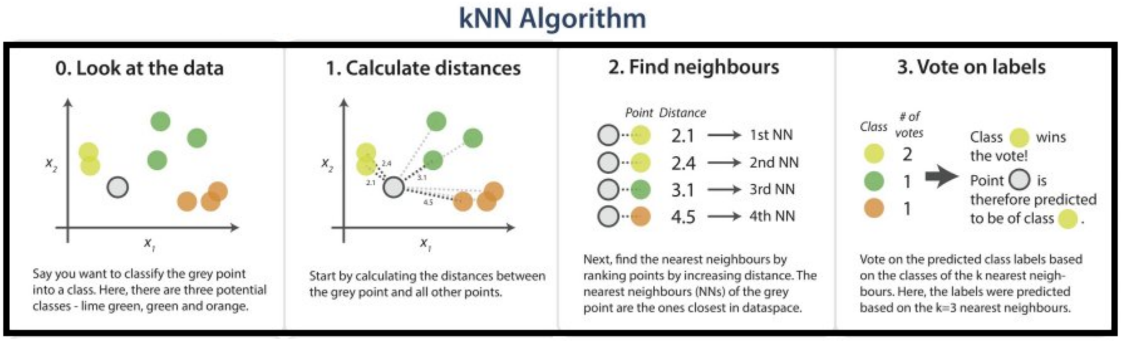
One of the simplest unsupervised algorithms is K Nearest Neighbor but this is the most popular and widely used. Support Victor machine can be mentioned as supervised machine learning which is also the most popular and widely used. Both of these algorithms will be presented below in detail.

K Nearest Neighbor

It is commonly known as KKN, stands for K Nearest Neighbor. For this algorithm, the K value needs to provide. K value is not fixed, we need to put according to accuracy value at the time of teasing and test. This procedure has been used for implementing the subsystem, Traffic Maintenance Sign Detection for Road Condition detection system. So, the details of this algorithm are given below.

Function

For example, 3 color balls can be considered for 3 classes which are in the figure: 2.9, [4]. This will make the understand very easy. Let's consider a new ball color is gray. KNN will define, its color class. KNN algorithm will do in three steps. Step 1. It will calculate distance any two points from the new ball. 2, Next the system will find the nearest neighbor among these pairs, step 3. Finally, the majority vote on a class based on the nearest neighbor list. In the figure it is clear. There are 3 colors ball, yellow, pink and blue which are considered as 3 classes, in the figure 2.9. Here it is clear, the distance from gray color to yellow color balls is respectfully 2.1, 2.4 which is the shorter distance. The majority,2 of them voted for yellow. So, the new ball belongs to the yellow color class group [4] .



Say you want to classify the grey point into a class. Here, there are three potential classes - lime green, green and orange.

Start by calculating the distances between the grey point and all other points.

Next, find the nearest neighbours by ranking points by increasing distance. The nearest neighbours (NNs) of the grey point are the ones closest in dataspace.

Vote on the predicted class labels based on the classes of the k nearest neighbours. Here, the labels were predicted based on the k=3 nearest neighbours.

Figure 2.9: KNN Algorithm [4]

Pseudo Code

This is a very simple and powerful algorithm for application. Here are a few lines for applying to develop a project.

∗ Load the training dada ∗ K value is defined the more accurate value ∗ Need to put level clustering level ∗ Need to take more than 70% for training ∗ Need to take rest for test ∗ Calculate the accuracy for the given tast data. ∗ Adjust the K value for the better value

∗ pass the new data

∗ The result will be carried out by matching the clustering level

pros and cons

Some advantages are presented here. This is the most important and simple algorithm in the unsupervised algorithm. This is a very simple, easy algorithm to implement. It can be used for classification and regression problems. This can be used for classification and regression problems. KNN can do not explicitly make a model. It has instant based learning capacity.

Some disadvantages are also given. The biggest problem with KNN is choosing the optimum number of number Neighbor. It does not deal with the mission value problem. K-NN algorithm is very sensitive to outliers as it simply chose the neighbors based on distance criteria.

Support Vector Machine

Support Vector Machine provides a powerful mechanism for clustering. Clustering is done based on no prior knowledge. It is known as extreme class. The support vector machine is easy to understand for 2 classes. A hyper-plane or line is drawn which is based on the distance of attributes. It works based on the attributes of the classes. For example, cats and dogs are 2 classes. A decision line needs to be put in between them. But it should be in an optimum way. If it is not in an optimum way then incorrectly classification will be for a dog or a cat class. Here Blue circle has been considered as a cat and pink circle as a dog. The figure is given as an example, 2.10, [5] This is the most important for the support vector machine and training example is ignorable. If an unknown dada comes then we need to find the short distance to the vector hyperplane. The nearest value to the plane will decide the classification.The kernel equation is given here, 2.3. This is a linear support vector machine. It is possible to draw in line with 2 classes of objects. Just we need to break down one-dimensional tow dimensions, but it is difficult for the multiclass. Need to use polynomial function and possible to get the hyperplane Linear kernel.

$$K(x,y) = xXy \tag{2.3}$$

Let's have a look at the multidimensional cases. It needs to change 2 dimensions to 3 dimensions. But it is computationally expensive. The kernel is used to reduce the computation. This is done by using kernel function which provides us nonlinear to liner type conversion. Some popular kernel type is given here. Polynomial kernel, the equation is here,2.4, Radial Basis function kernel, Sigmoid Kernel, etc. It usage a decision tree that is memory efficient. There is a way to use multi-kernel learning which shows very good performance for many applications but it is computationally expensive. Kernel makes 2D plane to 3D object which is given in the fig: 2.11, [5].

$$K(x,y) = (x * y + 1)^d \tag{2.4}$$

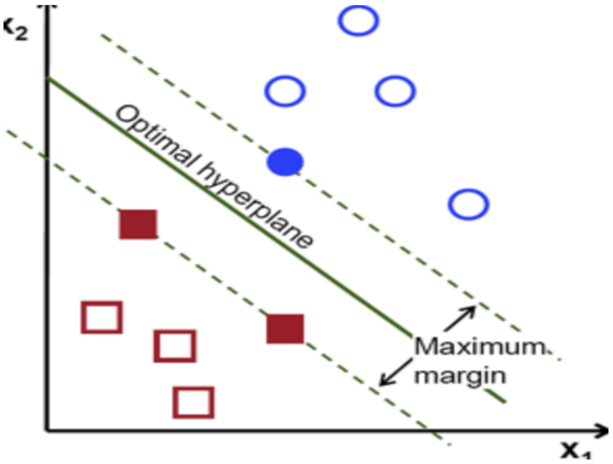


Figure 2.10: Classification of 2 dimensional [5]

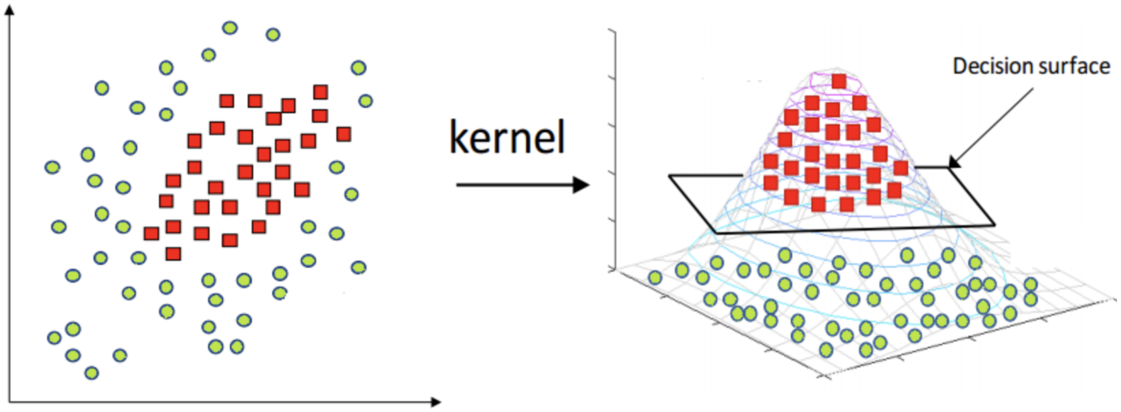


Figure 2.11: Kernel 2D to 3D [5]

Radial Basis Function RBF kernel

$$K(x,y) = e^{-p(||x-y||^2} \tag{2.5}$$

Random Forest Algorithm

Random Forest is the most popular and most powerful supervised algorithm. It is used for reintegration and classification tasks. It learns from the experiences. This algorithm is based on decision trees. Normally more tree the more robust the prediction which produces higher accuracy. In this algorithm, each tree gets an attribute and belongs to a class. The tree goes under a classification by voting of the other trees of the forest. Take the average of the different trees[35].

Applications

It has many application. Some of the important applications are mentioned here. Broadcasting SMS to customers. Finding medicine with the correct combination of chemical. Also used to identify diseases by analyzing the patient record. In the stock market, it is used to identify stock behavior. In E-commerce, it is used for commanding the customer product. In computer vision, this algorithm is used for image classifications. Assume of cases in the train set in N. Then, a sample of these N cases is taken at random but with replacement. if there are M input variables, a number m ¡ M specified such as each node selected variable m out of M.The best split of m is used to split the node. The value is m is constant while we grow the forest. Each tree is growing. It does not create any problem with the algorithm. Predict the now data aggregation(Majority vote of classification, the average for largest registration)

2.5 Deep learning

Deep learning is one of the hot topics at this time for various image analysis research projects. It is also using for the various detection system. Maintenance sign detection is the most complicated subsystem in this project. YOLO is one of the widely used deep learning tools. It contains a framework, is called darknet which is based on ANN. For labeling and classification of images, a tool is used which is called BBOXtool. It is a very efficient way of developing a detection system but for this project serious a complication arise to implementation, all these complications will be described in the Exception and Future perspective chapter.

2.5.1 Artificial Neural Network

An artificial neural network is the most important and widely used concept in this modern era, It has been used for the past few decades. Day by day it's getting more popularity. The idea is based on human brain neurons. This system build layer by layer. It is built on some input layers, some hidden layers, and one output layer. Hidden layers are optional. It is widely used to find patterns and recognition. This concept is proposed in 1940 but it was in the mathematical theory. Due to huge computation resources lacking it was not the par of computer science. No neural network is widely used all over the world.

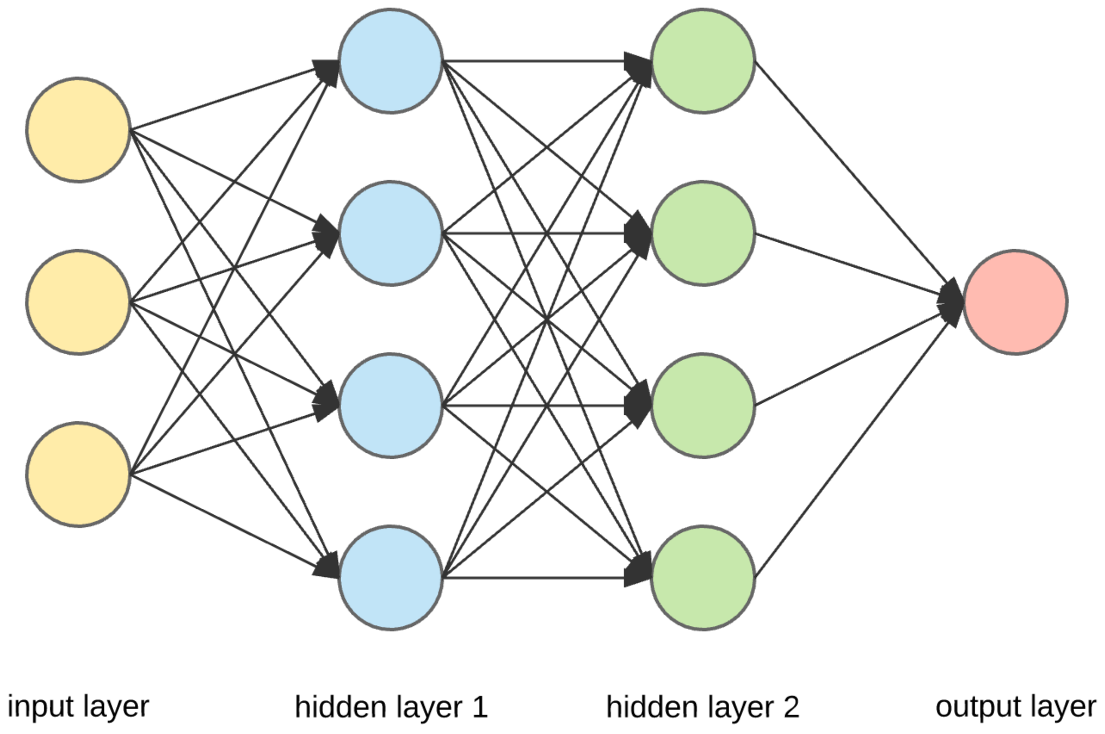


Figure 2.12: Artificial Neural Network

Structure of ANN

The artificial neural network is based on human brain stricture. It has 3 layers. An input layer, hidden layer, and output layer. The hidden layer is optional. Here is a figure of an artificial Neural Network figure: 2.12.

Input Layer The input layer is the first layer where input data are store then feed to the hidden layer of the neural network.

Hidden Layer The hidden layer lies between the input and output layers. The hidden layer breaks down the complex problem and small part and makes the solution for out later. The hidden layer performs the most difficult task. As for a simple example calculating a large number. The hidden layer makes small parts and solve the issues and sum up them for a final result. The hidden layer is the key factor for different types of neural networks, FFN, RNN, CNN. For most of the complex problem needs to design for the hidden layer. This is the most important part of NN. The number of the hidden layer also depends on the type of complicity.

Out Layer This is the last layer of ANN which provides the output data. The active function is added to this layer. This is the most important part of the output.

Feed forward Neural Network

Another name of the feed-forward neural network is the Multi-layered Network. It is the first step of deep neural networks. The goal of the neural network is an approximation of some functions. For example, for ca classifier Y=f*(X) map the input X to a category Y. Function Y=(X, W) here W is a weight parameter for X, System learns the value for W parameter and the result is the best matching result. In the feed-forward neural network, data are forwarded for farther calculation. When data back to the input line node from the output to make a farther calculation, this type of system is called a recurrent neural network. This algorithm is used for natural language processing. Intermediate nodes take parts for computation.

Convolution Neural Network

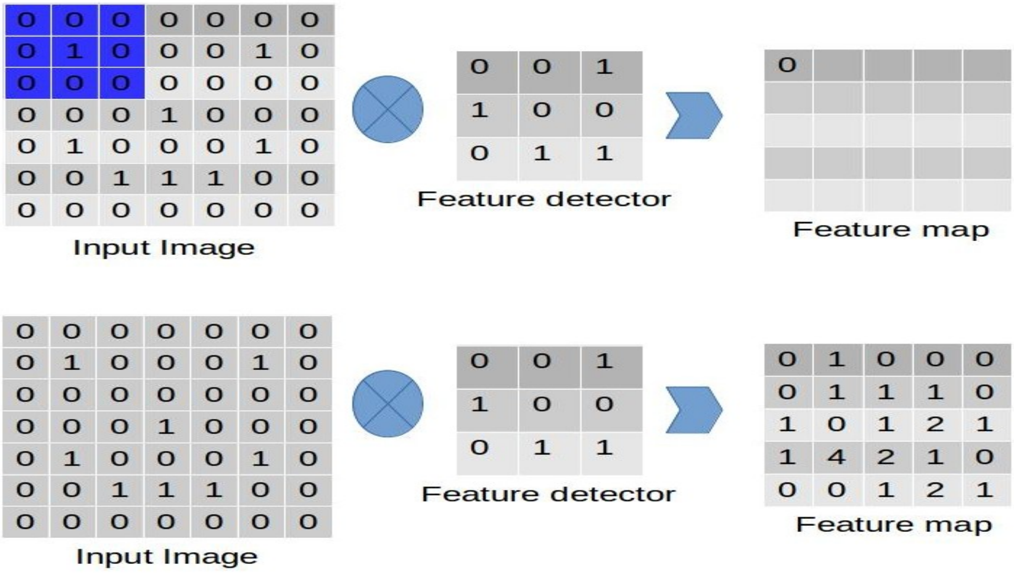
CNN is the most widely used neural network and it is very effective in the area of image classification and recognition. It has been used in the area of face recognition, traffic light detection, analyzing an image, describing an image [35, 36, 37, 38]. The word Convolution comes from mathematics. It mainly does the cross correlational between two-dimension input and filter.This algorithm is widely used in many applications.The filter slides over all possible input pixel and the output size is smaller than the input size. The mathematical equation is given below.

$$g(i,j) = \sum_{k=K}^{k} \sum_{l=K}^{k} [(i+k, j+l) * h(k,l)] \tag{2.6}$$

In this equation, (i,j) represents a pixel where i is the row index of the input and j is the column index of the input. f is the input two-dimensional input function and h is the filter. This operation computes the dot product or element-wise multiplication between the sub-image and the filter. An example is presented here 2.13.

Architecture of CNN

The architecture of Convolution Neural Network is quite different from the neural network but the main workflow is the same as it is a feed-forward network. Similar to the Neural network except for the conversational layer. Input is passed through the hidden layer. Each layer's output is the input to the next layer. Each layer is



0	0	0	0	0	0	0
0	1	0	0	0	1	0
0	0	0	0	0	0	0
0	0	0	1	0	0	0
0	1	0	0	0	1	0
0	0	1	1	1	0	0
0	0	0	0	0	0	0

0	0	1
1	0	0
0	1	1

0				

0	0	0	0	0	0	0
0	1	0	0	0	1	0
0	0	0	0	0	0	0
0	0	0	1	0	0	0
0	1	0	0	0	1	0
0	0	1	1	1	0	0
0	0	0	0	0	0	0

0	0	1
1	0	0
0	1	1

0	1	0	0	0
0	1	1	1	0
1	0	1	2	1
1	4	2	1	0
0	0	1	2	1

Figure 2.13: Convolution Operation

fully connected to the next layer. The input layer is divided into 3 levels. for the image, it takes height, weight, and depth. It accepts 3 dimensions. There is another major difference it performs the confabulation operation in the convolution layer.

Convolution layer

Convolution layer is the main part of the Convolution Neural Network. This is the feature information of the input image is extract using the layer. This layer is the basis of CNN. A feature map is generated form this. A convolution operation occurs between the input image and filter. It is a combination of mathematical functions. This result of the constitutional operation is sent to the active function [37].

Summary

Many authors have been enlightened with many ideas in the related work sections. Various approaches have been discussed which are related to this project. The traffic sign detection subsystem is a critical part of the system. To solve this issue, deep learning also has been taken into consideration for better output. The problem with deep learning will be described in the Exception and Future prescriptive.

3 Concept and Methodology

In this chapter, theoretical concepts will be discussed which are strongly related to my project. This chapter will be discussed in mainly 3 subsections, Pothole detection, Lane detection, and Maintenance sign detection. For detecting all these, various hardware has been used so far but our project based on camera and image only. So no additional cost is required for the hardware. Moreover, the same camera can be used for some other purposes also. For example to prevent thief, Face recognition techniques can be used to start the engine. The surrounding environment can be observed for many purposes. So, Image processing is the best option to make the cheapest and better outcome for many purposes with the same camera. Nowadays the camera works perfectly in the night also. Our biggest challenge everything should be in real-time and perfectly. Any fault can be the cause of great danger. The dangerous accident may happen due to delay detection also. Machine learning and image processing have been used for developing the project. For potholes and lane detection image processing techniques have been used. For detecting Maintenance signs, machine learning has been used. Image processing techniques take less computational resources and faster than machine learning. This is the reason for using image processing and on the other hand, machine learning been used for maintenance signs as a lot of noises on the maintenance sign. For example, almost all of the time maintenance signs are not on the main street. Beside the road, a lot of noises are there for trees and buildings. Pattern matching techniques of image processing, the template will not detect the sings due to huge noise. To overcome this issue, machine learning has been chosen for traffic maintenance sign detection.

This project contains three subsystems, Potholes Detection, Lane Detection and Road Maintenance Sign. Pothole Detection and lane Detection have been implemented using image processing and Traffic Maintenance Sign has been implemented using machine learning but Traffic maintenance sign also needs image processing for pre-processing the input image. Here is the overall design flow of the project 3.1.

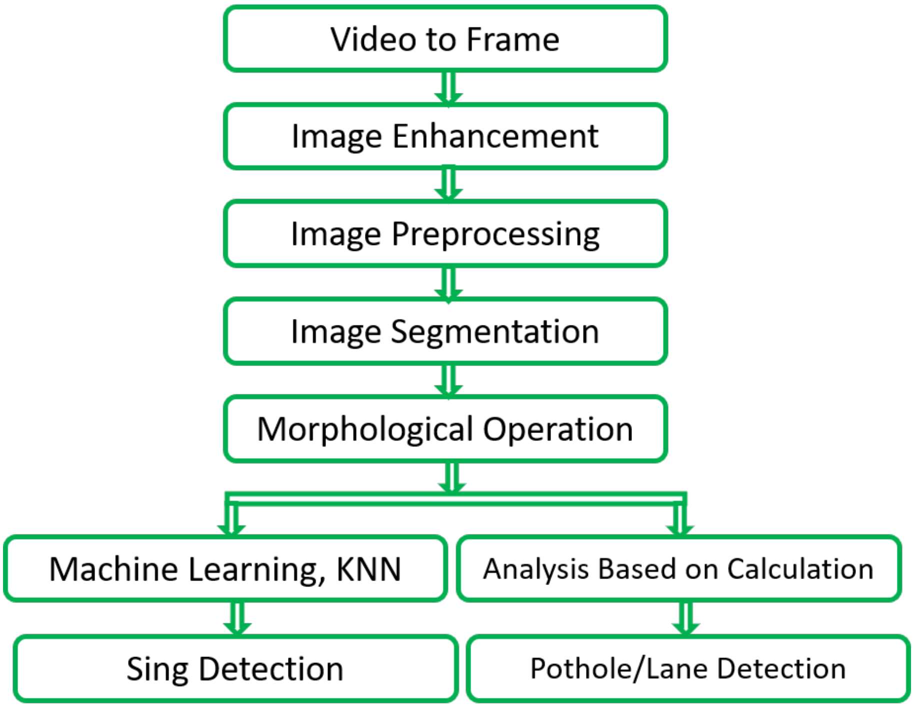


Figure 3.1: Design Overview

3.1 Pothole

Road condition detection mainly depends on potholes. Road conditions can be considered as good if the road surface is plane or contains small potholes which are not considerable. Road conditions can be defined as bad, depending on the number of potholes and the size of potholes. These are the two major criteria for finding the road condition. Due to time limitations, pothole size has been considered in this project. The image processing techniques which can be used for potholes detection are given one by one. Many challenges have to faced to develop such a subsystem. For example, where to mount the camera. The camera can be mounted on the front top of the car but it will detect a lot of noise. The camera should be placed in a lower position so that it can cover the surface of the road only to reduce the noise. The output of this subsystem must be correct and in real-time. The original color image contains unnecessary more information which is not required for shape and structure detection but needs more computational resources. The frame has been separated from the video. Considering this issue image restoration has done. The frame also needs to resize or need to take the region of interest. The image has been changed to gray also. Considering the issue, the required information should not be cut off. After resize or finding the region of interest. After all this prepossessing, main operations have been applied for detecting potholes. First, the median blur has to apply. so that, many small objects and small crack can be omitted. Next, Basic morphological operation, erosion has been used for removing some extra pixels around the potholes. It is a very usual matter, potholes contain extra small stones around them. So for removing these small stone erosion has been applied and then canny edge detection algorithm to find the line around the potholes in the images. For finding Contours and their areas, Contour detection and find contour area method have to be applied. Finally, a red-colored rectangular has to place over the contour area depending on the size of the area.

3.1.1 Algorithm

This algorithm works in real-time with webcam also. For convenient of this project, a recorded mp4 video has been taken. This algorithm is can be divided into 3 parts. The pre-processing unit, the processes for detection, and finally, detection. First, video to the frame has been converted. Then resize has been done due to reducing properties. These are prepossessing parts. Next part the image processing for pothole detecting stat with median blur the erosion. Finally, contour methods have been applied for finding the potholes. If contour size is less then 1500 pixels, are not considered as potholes. The last and third part is detection, In this part, only the bigger than this size is considerable for potholes.

The median filter has been used to remove unwanted noise. It brings smoothness to the image. As a result, many unwanted small objects removed from the image which helps to draw proper counter. Without the median filter, many unwanted contours will appear in the image which is needed extra time and computational resources to filter again. In the project, median kernel value, 5 have been taken for applying to median function. All the blurring techniques smooth the images but why the median blur has been used here. The main reason for using the median blur is to smooth the edge also with the same median value. In the case of Gaussian, It has a great impact on edge smoothing of the image but It works on comparing the neighborhood weight also. So, it will produce an extra small counter area. Figure 3.2 has given an overall idea of the process flow. Erosion, one of the basic operation of morphological operation which has been applied to structuring the element and removing the noise. Erosion produces smaller structure elements with the same shape. It removes small scale details from the binary images. Many small partials around the potholes which should not be considered as potholes. To eliminate such unwanted particles erection has been used. Especially to element the additional

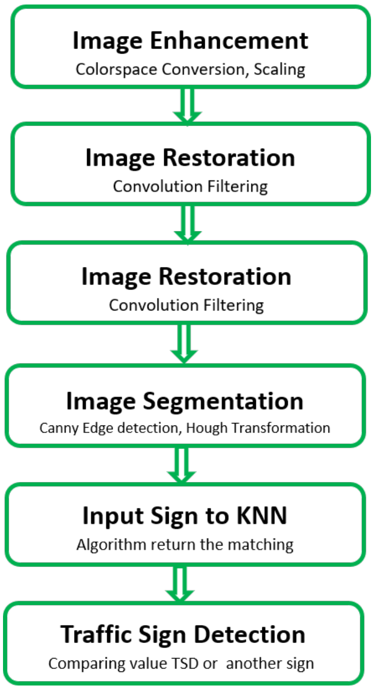


Figure 3.2: Process flow

thing around the pothole, this operation has been taken. A Canny edge detector has been applied to the eroded image. It produces the edge in perfect condition. By applying the canny edge detector, it is possible to find the all edged element. Most of them are crack and small tiny things. Contour detection and contour area calculating methods have been used for finding the pothole on the eroded image. Curves can be found by joining the contour moments. By joining all curves, the pothole's image is visible perfectly but alone with potholes many other small objects like a small stone or other small objects can be visible from the image after applying contour. To elemental, all these small objects, areas of contour have been calculated. If the contour size is less than 1500 pixel has not been considered as a pothole.

3.2 Lane

This is the most essential application for an intelligent vehicle. The most difficult issue is noise for detecting lanes. The output of this application should be in real-time also. Accuracy and real-time detection are the two mandatory criteria for lane tacking. Lane detection system is mandatory for the self-driven car but also most important for the Driving Assistance System. Statistic tales, cause of the accident for Road Departure is (19%), Lane Change and Merge are (4%) [19]. The reason for these both accidents is lane detection fail in time. So, 23% of the total accident is the cause of lane detection. The importance of lane detraction is clear from this statistic. The working procedure has been presented in the algorithm section.

Algorithm

To detect lane the most difficult issues are to remove noise and tacking in real-time. An unnecessary portion of the image and color information can be removed the very first time. It will be very helpful to avoid unnecessary calculations which will save huge time. Color information is not required for finding the feature. So, the original image has been changed to grace scale image. It will help to fast calculation. Lane width is fixed. The car will detect only one lane. For this reason, the region of interest has been fixed only for the lane area. This is the most critical part to fix the lane area. As the camera is in a fixed position, the lane area will not be changed. Incorrect selection region will cause a failure which can lead to unpredictable consequences. Only lane area portion has been taken in Region of Interest for the farther operation. Using thresholding and masking only black and white image has been taken. Many unassay pieces of information have been removed for making the system faster. Now, the optimized image is used for detecting lane. Canny edge detection has been applied for finding the optimum edges than other edge detection algorithm. This is the main reason for using a canny edge detection algorithm. There are many other reasons also for chosen this algorithm. This algorithm makes the image smooth fast which is the cause of noise reduction. It produces a very low error rate. People know this algorithm as an optimized edge detector [21]. On this edged image, Hough transformation has been applied for finding the lines. It is the best algorithm for finding straight lines. Many researchers use canny edge detection and Hough transformation algorithms for lane detection [21, 33, 15]. Using figure : 3.3, an idea about the Lane Detection subsystem has been presented which will be applied for implementation.

3.3 Traffic Maintenance Sign

Traffic Sign Detection plays a very important role in the Autonomous car or Driving Assistance System. The maintenance sign is one of the most important. Many approaches for detecting maintenance sign but still detection system is not perfect enough. Some of the approaches can detect perfectly but not in real-time. Many of

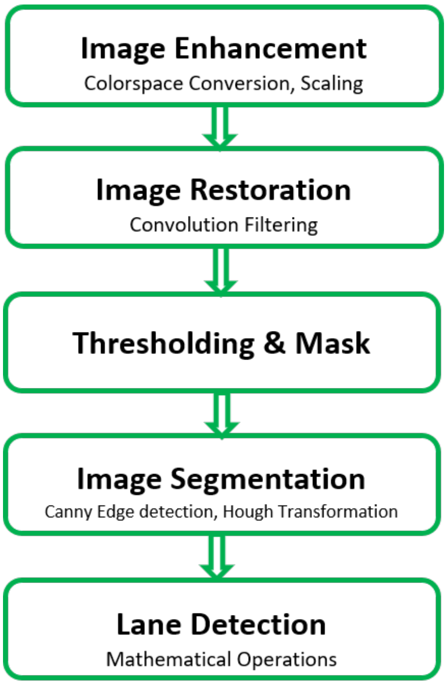


Figure 3.3: Lane Process Flow

them can detect in real-time but many false-positive also or can detect only in sunny weather conditions but not in nigh or in dark. These senior is already described in the related work section. So, many researchers are still working for perfection. Many applications for Traffic sign detection are already developed using image processing and machine learning. All these approaches also have the same issues. Considering the pros and cons of many algorithms, KNN has been chosen for this subsystem, Traffic Maintenance sign detection. This is the most effective classifier algorithm.

3.3.1 Algorithm

In many cases, Maintenance signs are not on the road. It is placed outside of the road. So, it is difficult to find a region of interest. It is needed to scan the whole image and the noise type is also not the same. Traffic maintenance sign is almost noise-free on the road but noise is unpredictable outside of the road. For solving this issue, the input image is converted to a gray image. It will help to reduce mathematical calculations. Gaussian blur has been applied which reduces the noise from the image. Then canny edge detection is applied for detecting an edge in the image. This edge detected image is resized to 20 by20 Pixels which is

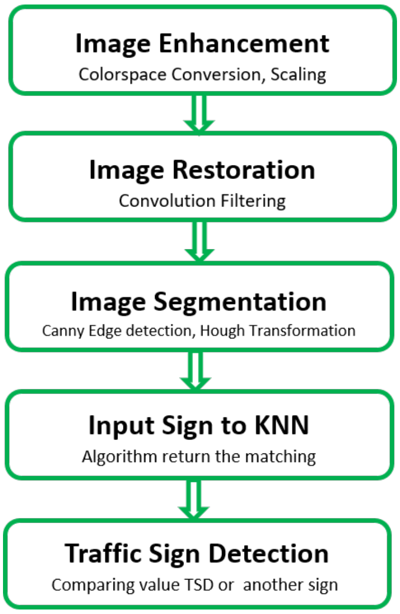


Image Enhancement
Colorspace Conversion, Scaling
Image Restoration
Convolution Filtering
Image Segmentation
Canny Edge detection, Hough Transformation
Input Sign to KNN
Algorithm return the matching
Traffic Sign Detection
Comparing value TSD or another sign

Figure 3.4: Maintenance Sign Process Flow

the same size as the training image. This small image sends to the KNN algorithm. It matches with the training images and returns the value. Comparing the return value, maintenance or other sign is detected. The process flow is given here for maintenance sign detection, Figure: 3.4.

Summary All the algorithms have been described separately in detail which has been used at the time of implementation and also has been described in the following Chapter 4, Implementation. Process flow for all subsystems has been given in all algorithm sections. This chapter will help a lot to implement this project.

4 Implementation

The implementation of the thesis will be discussed in detail in this chapter which has been derived from the concept. The libraries and tools are the most important for implementation. These will be discussed in separate subsections. The algorithms of the subsystems will be described in the algorithm subsections. The overall architecture of the system will be presented below in the Layered Architecture subsection. Various tools, libraries, other hardware and software resources will be described in the Toll and Resources subsection. Image processing has been used for potholes and lane detection. Image processing and machine learning have been used for maintenance sign detection in this situation awareness project.

4.1 Layered Architecture

Here is the general overview of the system layered architecture which will provide an overall design factor of the project. Figure is added here 4.1.

User Interface Users can start the system and the system takes input through the camera. For the convent of development process recorded video has been taken. This input is sent to the application layer for processing. After processing the image, the system sends feedback from the application layer to the interface.

Application Layer In this layer, images are processed to detected pothole, lane and Traffic sign. This layer contains all the libraries and tools for the applications. This layer also communicates with the Driver Management layer and the Support layer. For example, the camera needs to talk to its driver and some time with the operating system also. Many DLL files also need to talk with OS and applications. These are the shared files. DLL(Dynamic-link library) is a shared file of the operating system. These DDL file talks to the application layer and Operating system layer and a few more examples are here. Simple example, a printer which connect from an API(Application Programming Interface). Other example video player,

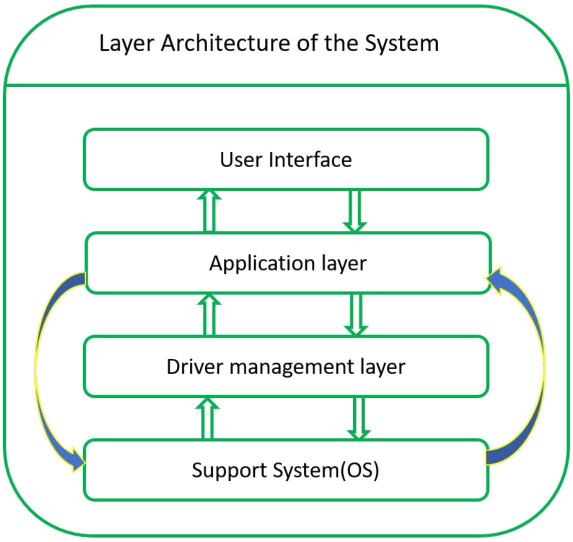


Figure 4.1: Over All Layered Architecture

GDI(Graphical User Interface), etc. The working procedure of this layer has been shown in the figure very clearly. The figure is presented here, Figure : 4.1.

Driver Management System This layer contains all the drivers for the system. It communicates with the application layer and the OS layer. For example, the camera takes video. The camera needs driver's help and it communicates with OS also.

System Support Layer Operating System lays on this layer. It would not be possible to make an application without an OS. The database also is in this layer but for this project no database system. Video file has been stored in this layer and has been used by the application layer. Operating System is the key part of this layered. All other layered are based on the operating system. It provides the facility to install various developing tools. The deployment facility is also provided by OS. User access and all most all features are for comfortable and user secure provided by OS. This is the major and important part of the support system layer.

4.2 Tool and Resources

To implement the concept of this master thesis, software and hardware components have been used. Let's have a look for the hardware and software components separately. Software is the most important for developing this research project. Details of the software implementation will be given one by one.

4.2.1 Hardware

For developing this project a laptop has been used which is manufactured by the Toshiba and model Satellite C855. It contains 12 GB memory and inlet Core i 5 processor of 2.60 GB. The developed software will be transferred to the pi3 model B. Mobile phone iPhone 5S and Huawei pro 30 have been used for video recording.

4.2.2 Software

several tools and libraries have been used for this project. These make the software development life cycle easier to reach the solution. let's have a look at the software component and then go for All the tools and libraries that have been used are listed below including a short introduction about each. Several libraries have been used which are given one by one. In addition to that, some of the functionalities which are most important in this project will be presented also in detail. For example, OpenCV will be discussed in detail which is the key part of this project. Besides OpenCV other libraries also describe such as Numpy, time and Psutil. All these libraries will be described in a separate section. Last of this section, several tools like Spyder, Qt Designer and python will be described.

Operating System

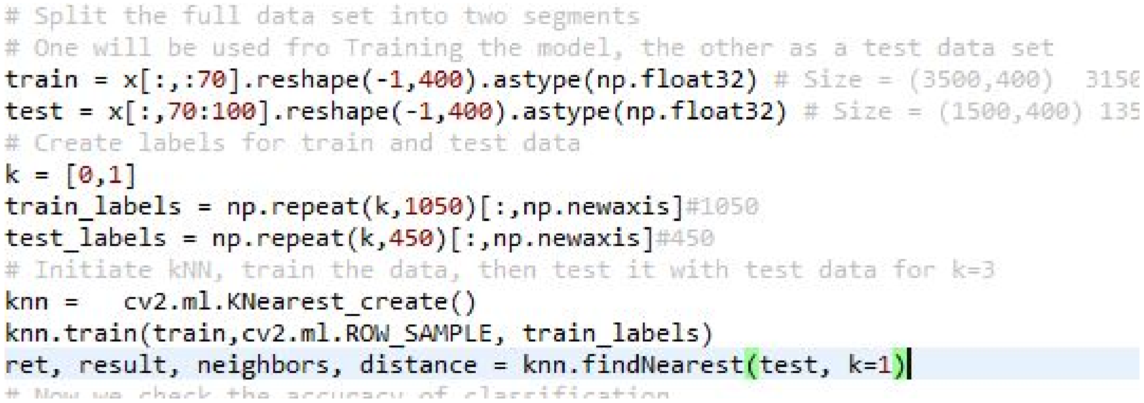
Windows 10 and Ubuntu have been used for developing this project. After developing this project has to be deployed into Raspberry pi. It will be easy to deploy to Raspberry pi as Ubuntu and Raspberry pi are similar. For personal comfortable Windows 10 has been used.

OpenCV

The description of openCV is already in the theoretical background section. The functionality of OpenCV and the used methods in this project will be presented in this section. This is the main library for this project. OpenCV is the short form of Open Source Computer Vision Library(opencv.org). It is a machine learning library and computer vision library. This library has been developed for providing a reliable foundation for computer vision and machine learning applications. It boosts the usages of image processing and machine learning algorithms in the industrial label by providing by opening the easily accessible API for all. More than 2500 optimized computer vision and machine learning algorithms have been developed in this vast library using C/ C++. It is the most widely used image processing and machine learning library. It has a huge user community [3]. Any problem to use the function or parameters we can get help from the community. Its documentation is also so rich and well decorated. The functions can be select from the selection button according to the selected version. It is so easy to find any solution from the documentation. This library is build using c, c++ but it has an interface for c#, c/ c++ and python. we can get huge help from the community. Around forty-seven thousand people are involved in this. It's a huge amount of computer vision algorithms are used to detect an object, object movement, identify a face, form 3D model of an object and so on. It has implemented some statistical machine learning algorithms like Random Forest, Support Vector Machine (SVM), KNN and it has support for Deep Neural Network (DNN). The most important feature of the OpenCV is DNN module. A programmer can easily train the model using some other popular library, Caffe, TensorFlow and then can load the model using OpenCV. Another important reason for using OpenCV is it has a python interface for programming in Windows, Linux and raspberry pi platform. These are the two reasons for using OpenCV as a machine learning library also. Many methods from the OpenCV library have been implemented. Some of the most important are given here. These are K Nearest Neighbour, Median blur, Gaussian blur, Contour, erosion, dilation. canny edge detection, rectangular, etc. Let's go through one by one a few of them.

K-Nearest Neighbour This algorithm has been used for traffic sign detection. This is the simplest supervised machine-learning algorithm. The main idea behind this algorithm is to find out the closest match with the test data. It matches test data by maturing distance and by voting of the classes. There are 2 classes in the project. One traffic maintenance sign and another is the speed limit 50. 3000 images have been for training and testing. 70% of them for training and remind for testing. Half of these images for traffic maintenance sign and rest of the image for the speed limit. Later on, the test image will be sent to matching with the trained image by KNN. Code is given mainly for KNN tarin 4.2. KNN will find the match and return the value. By comparing the return value traffic maintenance is detected.

Blur Two blur techniques have been implemented in this project. One of these two is Gaussian blur which has been used for traffic maintenance sign, Lane Detection and another median blur which has been used for pothole detection subsystems. The main difference between these two blur techniques is its kernel working techniques. For the median blur, it takes the median value of the given matrix. For example, 3, 3 size kernel has been taken then the algorithm will find the median value of 9



```
# Split the full data set into two segments
# One will be used fro Training the model, the other as a test data set
train = x[:,:70].reshape(-1,400).astype(np.float32) # Size = (3500,400)  3150
test = x[:,70:100].reshape(-1,400).astype(np.float32) # Size = (1500,400) 135
# Create labels for train and test data
k = [0,1]
train_labels = np.repeat(k,1050)[:,np.newaxis]#1050
test_labels = np.repeat(k,450)[:,np.newaxis]#450
# Initiate kNN, train the data, then test it with test data for k=3
knn =   cv2.ml.KNearest_create()
knn.train(train,cv2.ml.ROW_SAMPLE, train_labels)
ret, result, neighbors, distance = knn.findNearest(test, k=1)
# Now we check the accuracy of classification
```

Figure 4.2: Code:KNN algorithem.

numbers and apply to all pixels. Here is an example 4.1 for the median blur which is presented in the 3 by 3 matrix.

40	50	60
70	80	90
100	110	120

=40,50,60,70,80,90,100,110,120

Table 4.1: Median Blur

In the table, it can be seen that 80 is the median of 9 elements. This technique has been applied for pothole detection which helped a lot to remove the noise. In this way, the pixel value has been changing with the median value.

On the other hand, in the case of Gaussian blur, it will deal with the stander deviation value, apply them to the corresponding pixel. Say, 3 by 3 kernel has been taken. First, place the kernel on the first pixel of the image. For finding this average value. First need to calculate the stander deviation of pixels which has been discussed in the chapter State of Arts, subsection, Image processing. The equation is given in that chapter 2.2, The action for finding and then the average value of the currently used kernel will be applied on the first pixel of the image. In this way kernel, the average value will be applied all over the image until the last point.

Erosion Erosion is one of the fundamental morphological operations. It helps to bring the proper shape of the object in the image. It removes the pixel from the boundary of an object. Considering this issue, to remove some noise around potholes, this technique has been applied. As a result, some very small stone on the boundary of the pothole is not detected as a part of potholes. This operation has been done before applying the Canny edge detection algorithm to find the pothole boundary edges. This is a very effective way to remove noise. From the figure, it is clear that many small marks have gone disappear especially around the pothole. This helps to remove many unwanted small spots. This is the most effective way to remove the noise aground the object.This algorithm has a big impact for finding potholes. Here is the picture of before erosion and after erosion 4.3.

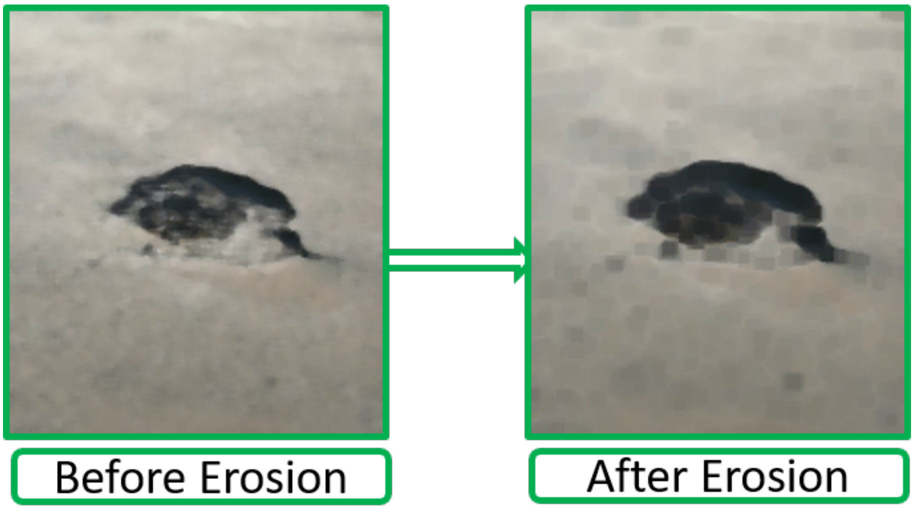


Figure 4.3: An Image before and after erosion

Canny edge detection The majority of the people consider the Canny is the ultimate edge detection algorithm. Most of the people use Canny for detecting edges in the image. It has two key parameters. One of these is the upper threshold and another is the lower threshold. The upper bound is used for the edge marks and lower than the threshold value is used to discarded pixels. Edge is detected if the value is upper then the upper value. In between these two threshold values are classified as edge and no-edge all the subsystem. This method has been used for all the subsystems but these values are not the same for all these subsystems. This method has been used in all subsystems. It has been presented below for all detection subsystems. Code has been given from the lane line detection. Here is the picture for canny before and after applying canny, Figure: 2.5.

Contour The contour contains a closed shape. It joins all the points, those are having the same pixel value or intensity. The contour function in Opencv can detect shape inside the shape and object inside the object. Outer object or shape is the

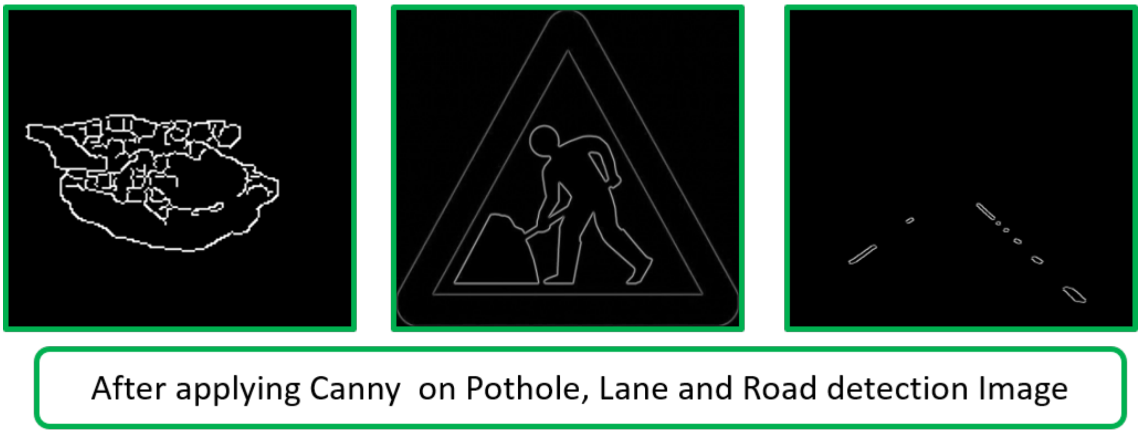
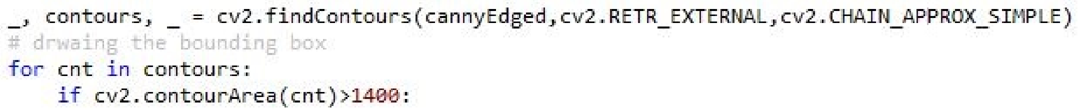

After applying Canny on Pothole, Lane and Road detection Image

Figure 4.4: An Image after applying canny



```
_, contours, _ = cv2.findContours(cannyEdged,cv2.RETR_EXTERNAL,cv2.CHAIN_APPROX_SIMPLE)
# drwaing the bounding box
for cnt in contours:
    if cv2.contourArea(cnt)>1400:
```

Figure 4.5: Code for Contour

parent and the inner object is called a child. There are several functions related to contour, e.g., For finding contour, area of the Contour, draw counter etc. Contour detection has been used in the pothole and traffic maintenance sign detection. In pothole detection, the contour is used for calculating the pothole area. Finally, a box has been fixed on the real pothole. ”Findcontour” function contains 3 returns value and it is applied after canny edge detection. It has three operations modes for finding contour. For example in this project, one of them, RETR-EXTERNAL mode has been used. It will show the outer contour. For drawing a line, all contours not needed, just need ends point the line. For this Counter approximation method, cv2. CHAIN-APPROX-SIMPLE has been used. It is applied for saving the memory as it removes all the recant points and compresses the contour. Here is the contour detection a few lines code for finding the potholes 4.5. Find contour and contour area have been applied for in pothole detection and find contour is also used for traffic sign detection. Draw counter function has been used in the Traffic Maintenance sign. The find contour is the basic operation among the contour operations. Once contour is detected then the area of the contour is possible to find by the find contour area method. Draw contour another important method. cv.drawContours (img, contours, 3, (0,255,0), 3) [3].It is the most important and most widely used.

Numpy

NumPy, short for Numerical Python, is a very popular python package. Primarily introduced in 2006 as an independent package for large scale numerical data creation and manipulation. Among many simple and complex computations, it also used for easy Fourier transform and random number capabilities. This library can be used for storing multi-directional generic data which opens integration opportunities to various databases. A NumPy array is a grid of values having the same type and is indexed by a tuple of nonnegative integers. OpenCV library also uses NumPy arrays for storing and operating on image data. It makes very convenient to access each basic element of the images as they are represented as three-dimensional arrays. Also, it simplifies the complex algorithm and workflows. These are the reasons for the popularity of this library and have been applied in this project. This is widely used for Numerical analysis, Linear algebra and Matrix computation. Numpy creates multidimensional matrix and array which are very effective for presenting matrix. This matrix manipulation is required for various operations. NumPy provides many mathematical functions that can be easily applied for image processing. Actually, Image is a numerical presentation of the same type of data. Numpy provides the facility multidimensional matrix of the same type of data. So, the operation of image processing has gone very easy and effective. It is has been applied in every subsystem of this project. Especially in the Lane Detection Detection for plating and masking. Here is an example code using numPy. Without numPy it would be a very complex and many lines code that has been done with 3 line codes which are given as listed below. Here is the NumPy library that is imported as nu. Here NumPy has given efficient access to the image which is the region of interest. This plotting point has been used on masked image and finally, ROI has been found. Without this, it will be a complex calculation of multidimensional array which is the most difficult and more time-consuming task. It is an open-source python package that can use as an alternative of Matlab with python. The usage of this package is scientifically operations. It contains a huge number of mathematical, algebraic and transformation functions. Here are a few lines code using NumPy but without NumPy it will be very hard and many lines.

```
mask = np.zeros((frame.shape[0], frame.shape[1]), dtype="uint8")
pts = np.array([[200, 190], [275, 50], [380, 50], [575, 190]],
dtype=np.int32)
cv2.fillConvexPoly(mask, pts, 255)
```

Time

Time library is another popular library in python programming. It handles many time-related conversions and representations in python. It mostly used to convert a numerical expression to time constraints, printing current time and conversion of timing systems and zone. Also using the library, profiling of the functions is possible based on the time. A time type variable is stored as the number of seconds and current time calculated from total seconds from 12:00 am, January 1, 1970. It is also possible to access the individual fields of date, year, month, etc.) rather than seconds. The time module defines struct time for holding date and time values with components broken out so they are easy to access. This library helps to find the various issue issues. The various time-related function which is most essential for tracking the system behavior. Detection should be on time. In any case, false detection is dangerous. Same way, late detection also the case of many unpredictable consequences. This module is essential for testing as well as in production also. Some of the common functions of this library are waiting, sleep, calendar, time, etc. In this project, some of these functions have been applied for calculating the time-related issue. Example- Image pre-processing time, Post-processing time, total execution time, processing of each frame or per second processing number, etc. Time is the most important factor for the real-time system or the system which is related to active safety. All the modules of this project need to do in real-time or near real-time. So, this library helps a lot for analysis al the time-related issue which will be presented in the Result and Evaluation cheater.

Psutil

Psutil(Python System and process utilities) is a cross-platform library. This library is mainly used for memory and CPU utilization monitoring in python. This library contains many functions for finding memory utilization as well as CPU utilization. This is the key to control the memory and keep the system safer from memory-related issues. It takes the process id from the OS(Operating system then) then python code finds the RSS, VMS and other paging issues information. Though its arguments are not the same in the different operating systems, it provides all the necessary information. Especially it provides RSS and VMS for all operating systems. This is a very powerful tool for memory profiling. This library has been used in this project in every subsystem to find memory utilization. The detection system must use the optimize memory. If the system used more memory then more paging will be happened due to the main memory lacking. So, we can put the threshold for giving the worming so that users can stop unnecessary processes or systems that can stop the optional process. For example, memory lacking worming is counted then the system will kill video player automatically. In any application related to the active safety of the automotive domain, memory management is a vital issue to predict the system behavior. It is a very easy process and simple step but most important for automotive or any kind of awareness system where the video is taking as input. The contentious process of video to frame and other image processing steps are resource hunger. Continuously adding faction of a millisecond of a single frame processing will create many seconds which can lead to an unpredictable consequence. So, memory and CPU utilization is a big factor to make a safety awareness system. In this project, these issues have been taken into consideration.

spider 3

Every IDE has some pros and cons. Spyder also has some pros and cons but still the most popular IDE for python all over the world. It is a lightweight IDE, for this reason, it is faster than PyCharm and others. It uses very fewer system resources. It works well with an anaconda environment also. This is very easy to install in almost all platforms an also configurable. On the other hand, PyCharm is also a very good programming IDE for remote access like ssh. It connects with GitHub and updates very efficient. Code analysis also very powerful, works with cross modules. PyCharm also allows many other plugin tools. All these seem to be better for developing but all these extra features are not required for this project. For example, no need for remote access for this project. It is not open also for all. So no need to keep in GitHub or other ssh remote access. For frontend design, QtPy designer has been used which works very good with Spyder also. This project is developing for tine device pi 3 B. It is also possible to install Spyder in this tine device as it is lightweight. Any error is found in Pi then, it will help to debug. Considering all these issues Spyder ide has been considered for the development environment.

Qt Designer

It is the Qt tools for designing user graphical interface (GUIs) with widgets. It has rich and nice tools for making the user interface. It works well also with the python code. Very easy to design the interface using tools. Qt automatic converts this to code. It allows adding the plugin. This is the quest way to design interface. Various types of applications can possible to developed using Qt designer. Farther more it has an open-source module that is good enough for this project. It is also free to use. It supports all most platform, in case of not support QT support team will support their users. In that case, Qt commercial version is needed. It also provides a solution for various types of the display using the method DPI(Dot per Inch). It works with high-resolution images and paint factors. It has some fantastic features like auto screen scaling factor, adjustment depending on the device resolution.

4.2.3 The reason for using Python

It is not an easy task to say, what is the best language? It can be defined based on some criteria. There are many factors to define the best. Examples, learning curve, syntax, construction, execution time, memory management or easy to make particulate tools. So many factors can be considered for choosing a language as the best language. Python can be considered as the best language based on simplicity, job market and of course this is easy to learn. One of the famous software quality company, TIOBE has served on computer languages. The current position of python is 3 among 20 languages[39]. They did this survey based on many issues over many popular companies. The most positively changing language is python among the 20 languages.

Python is an objected-oriented, high-level an interpreted language. It is an attractive language because it has reduced the programming language maintenance, easy to learn and very simple to write due to readable easy syntax. Less chance to make mistake also as it is so simple. Python interpreter and the standard library are in source code or binary format. Anyone can freely distribute and use it for personal or commercial. Python interpreter does not catch the error but it realizes the error and prints the stack trace which helps to catch the exact error and fix the issue. The python debugger is written in Python itself. Debugger inspects local, global variables, and evaluation of an expression. It is also possible to put the break mark on the code. The quickest way to debug just writing a few print statements on the code. So, many advantages over other languages and widely to users all over the world. Besides all of these reasons, there is another interesting reason for using python which is memory management. Every variable is treated as an object. For example, if x=12, x=y and if x value is increased by 2 then x will refer to 14 and y value will be remaining 12. several variables locate the same object if they have the same value. It is a very optimized way. So, if new variable z is assassin a value 12 then it will refer the variable y. Its garbage collection also impressive. Python interpreter looks the which are not refereed or unused and getting aged then that is collected by the garbage collector. So, in a word, memory management is very impressive.

4.3 Algorithms

One picture says many things, which helps to better understand. These algorithms have been presented using pictures. In the subsections, algorithms have been given for detecting Pothole , Lane Detection and Maintenance sign.

4.3.1 Pothole Detection

Here is the pictorial representation of the pothole detection algorithm, Figure: 4.6. Every frame of the video is considered as input and sends to the remaining process for finding the potholes. Each of the frames has been resized to 275,180. The main reason behind this is to make the system faster and secondly, for better fitting in the display screen. Median blur has been applied for removing the small objects from the image which has been considering as noise. After that, a morphological operation, erosion has been applied for removing the small stone around the pothole. So that, proper size of the pothole can be found. It also helps to remove small crack. A canny edge detection algorithm has been applied for getting the edges of the potholes. After finding the edges, contour detection algorithms, find-contour and contour-area have been applied. This is an important issue because, after blur and morphological operations, some noises are still there, like the small holes or cracks which are less than 150 mm. For this, contour-area has been applied for finding the area. The total area should be bigger than 1400 pixels. Video has been taken using Huawei 30 Pro mobile camera. It is depending on the camera resolution. In this way, the pothole is detected. This process is faster. A red color rectangular has fixed over the main image wherever pothole is detected.Here is the figure for pothole detection algorithm 4.6. For the awareness system, pothole size can be used for alert messages or other necessary actions. It can take the necessary action depending on the size of the potholes and position. Details of each process have been given in separate paragraphs.

Video to frame This process has been done for all the subsystems. Here is the picture that shows how the frame is separated from the recorded video 4.7. For this operation used coded also is given side by side.

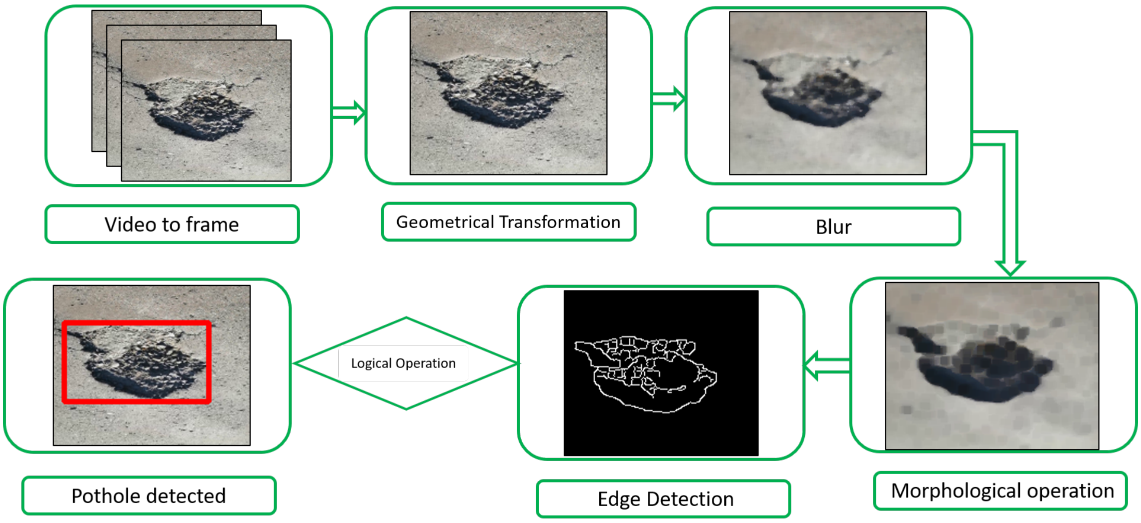
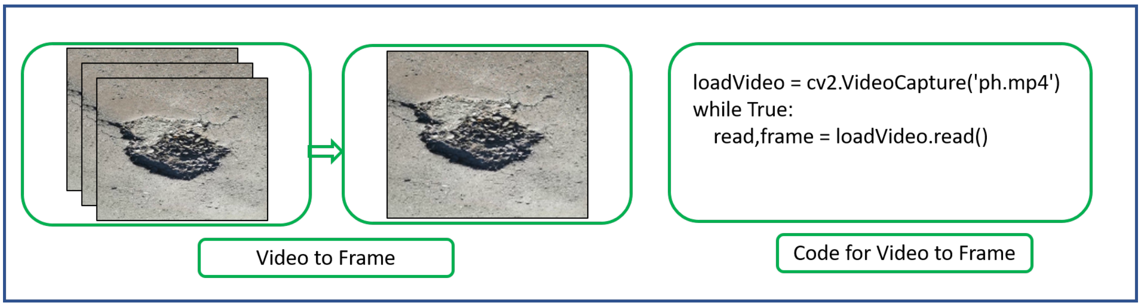


Figure 4.6: Pothole Detection Algorithm

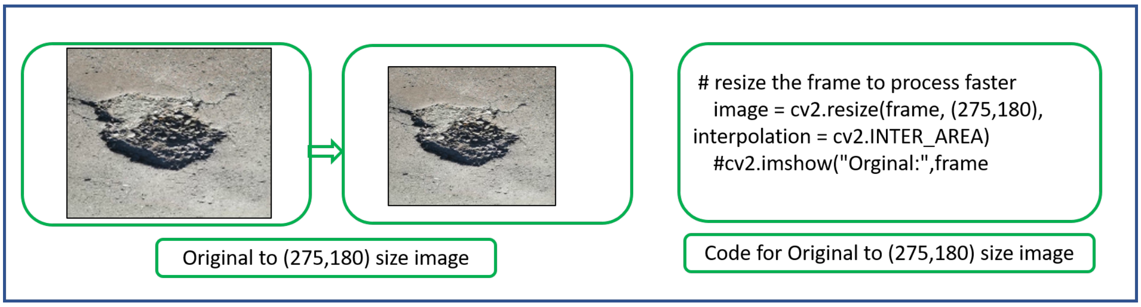


```
loadVideo = cv2.VideoCapture('ph.mp4')
while True:
    read,frame = loadVideo.read()
```

Figure 4.7: Video to Frame

Geometrical Transformation Each frame has been resized to 275 by 180. Video size may differ due to change to the camera but the resized image will remain the same. Here is the figure of the original and a resized image with the python code, Figure: 4.8.

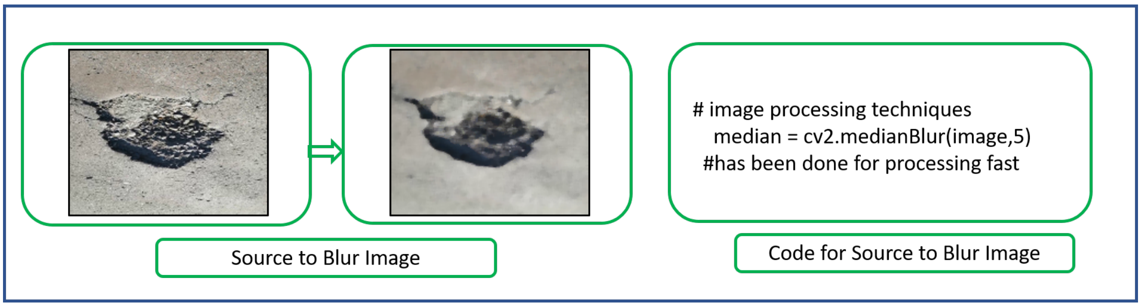
Blur Blur has been used in all of the detection subsystems because it reduces the properties which are not needed for structural information. It makes the calculation easier. One example is presented here from the pothole detection subsystem. Here are the picture source image and blurred images with python code side by side, Figure: 4.9.



```
# resize the frame to process faster
   image = cv2.resize(frame, (275,180),
interpolation = cv2.INTER_AREA)
   #cv2.imshow("Orginal:",frame
```

Original to (275,180) size image
Code for Original to (275,180) size image

Figure 4.8: Original to resized image



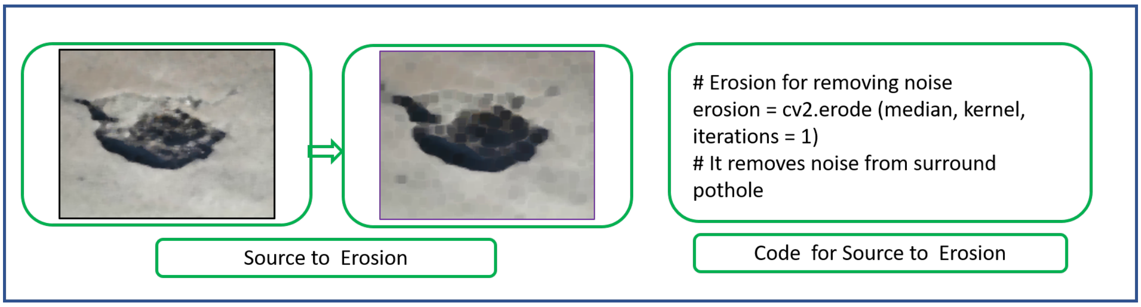
```
# image processing techniques
   median = cv2.medianBlur(image,5)
 #has been done for processing fast
```

Figure 4.9: Source to Median Blur Image

Morphological Operation One of the basic morphological operations, Erosion has been done. Pictorial representation of the source image and image after erosion given here with relevant code, Figure:4.10 . It has been used to remove the noise from surrounding of the pothole. Here is the source image is a blurred image.

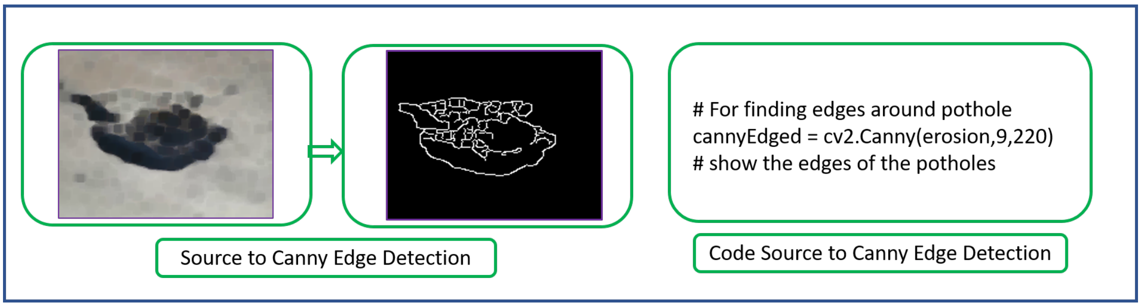
Edge detection Canny is the most popular edge detection algorithm. Canny edge detection has been applied to the eroded image. It has been applied to all of the subsystems but the source image is not the same. Here is the eroded image for pothole but the detection of Lane and Traffic Sign Detection source image is Gaussian blurred. Pictorial representation of the source of eroded image.An image after and before applying canny edge detector is given here with relevant code, Figure:4.11.

Pothole Detection This is the final operation. A contour detection operations are done on the canny edged image. Contours are detected as potholes but all of them are not potholes. Example crack, the small holes which are not considered as potholes. So, it is needed to find the area to remove the small crack and small holes from the detected contours. For this reason, the area of the contours has been



```
# Erosion for removing noise
erosion = cv2.erode (median, kernel,
iterations = 1)
# It removes noise from surround
pothole
```

Figure 4.10: Morphological Operation, Erosion


Source to Canny Edge Detection
Code Source to Canny Edge Detection

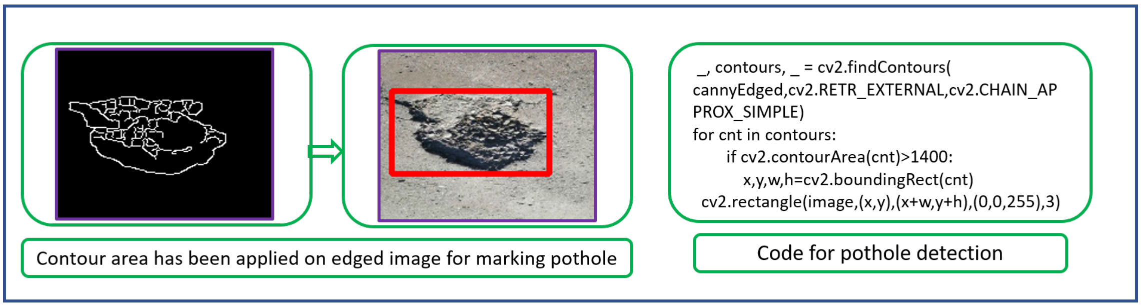
```
# For finding edges around pothole
cannyEdged = cv2.Canny(erosion,9,220)
# show the edges of the potholes
```

Figure 4.11: Edge Detection Operation, Canny

calculated and excluded from the pothole list. Remain grater than the threshold areas are consider as potholes. Areas have been calculated based on the number of pixels for the area. It is depending on the number of pixels which is fully depended on the camera resolution. Here is the figure for pothole detection 4.12.

4.3.2 Lane Detection

Lane detection is one of the most sensitive applications for any kind of land transport. Any kind of false detection or fail to detect will cause unpredictable consequences. Here is the picture of lane detection algorithm,4.13. In the figure, there are 6 steps for this algorithm but many mathematical calculations among them. let's discuss all of these one by one. Step 1, frames have been taken from the video. Each frame has gone for all next steps for detecting the lane . Here is the Figure :4.14 for the video to the frame. Step 2. Region of Interest has been fixed as the full image also is not needed. Moreover, it will reduce the computation. This the most difficult task. 4 points have to choose correctly otherwise the system will not find the lane line.2 of them will be in from the front and 2 others from behind. These points will



```
_, contours, _ = cv2.findContours(
cannyEdged,cv2.RETR_EXTERNAL,cv2.CHAIN_AP
PROX_SIMPLE)
for cnt in contours:
      if cv2.contourArea(cnt)>1400:
          x,y,w,h=cv2.boundingRect(cnt)
  cv2.rectangle(image,(x,y),(x+w,y+h),(0,0,255),3)
```

Contour area has been applied on edged image for marking pothole
Code for pothole detection

Figure 4.12: Canny to Pothole Detection

make the region of interest. It is seen as a white color in the image. It has been shown in the figure. Step 3, Black and white image, this has been done by bitwise AND operation between ROI image with the original image then the image has been changed to gray so that some noise will be eliminated. For example, very short edge lines will be eliminated for canny edge detection. Moreover, it will be helpful to reduce the calculation also. Next step 4, Blurred, Gaussian blur has been applied for removing the noise and for better edge detection. Step 5, Canny edge detection has been applied for detecting edges that help to detect the line by using Hough transformation. Step 6, in this step Hough line detection has been applied and then left and right-side lines are kept in the multidimensional array and form there left and right lanes have been detected. From the left line keeping array and right line keeping array are merged with the original image for displaying the detected lane.

Video to Frame [ht] It is the very beginning of the lane and pothole detection. It is needed to convert the frame to process the individual image. It is the same as the pothole procedure. Just for the continuing algorithm and better understanding pictorial representation is given below. Here is the figure: 4.14

Region of Interest ROI is the basic operation for removing and unnecessary calculations. The maximum part of the frame is not necessary. Only lane area is counted for ROI but if it is not correctly detected then the system can not find the lane from the frame or video. At the same time masking also has done. Here is the pictorial represented with code. The maximum part of the image has been discarded for farther calculation.Here is the figure: 4.15.

White and Black image A Black and white image has been taken by AND operation with the masked image and ROI where only the one lane in presented. so,

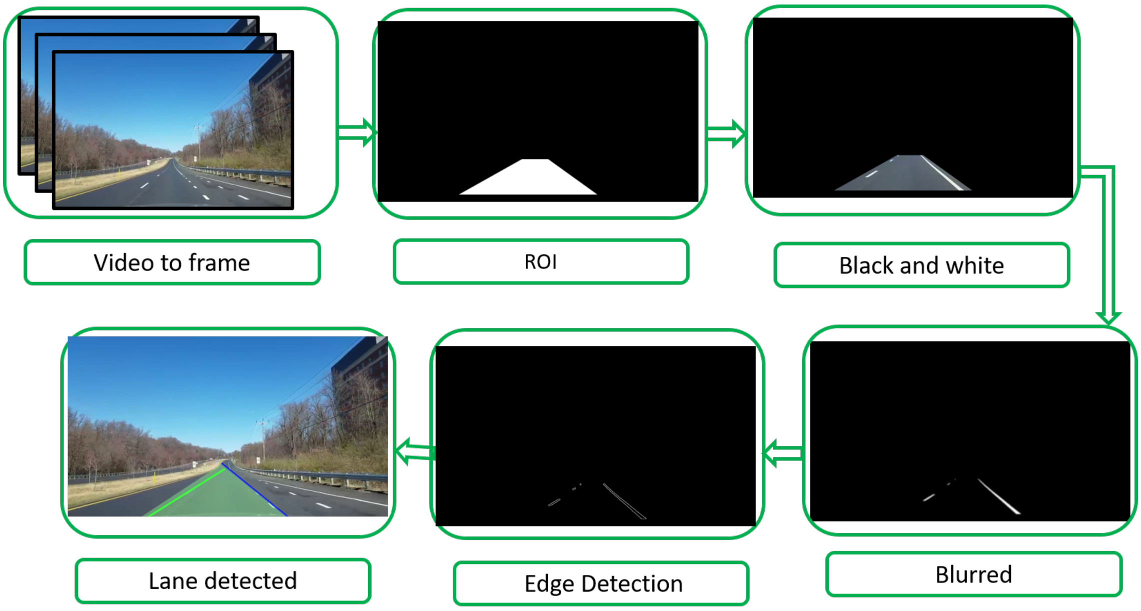
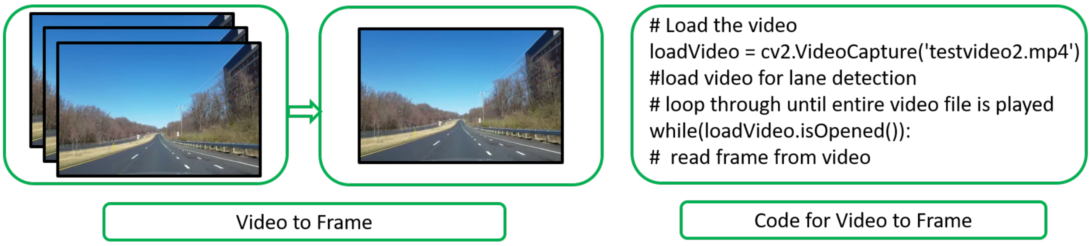


Figure 4.13: Lane Detection Algorithm

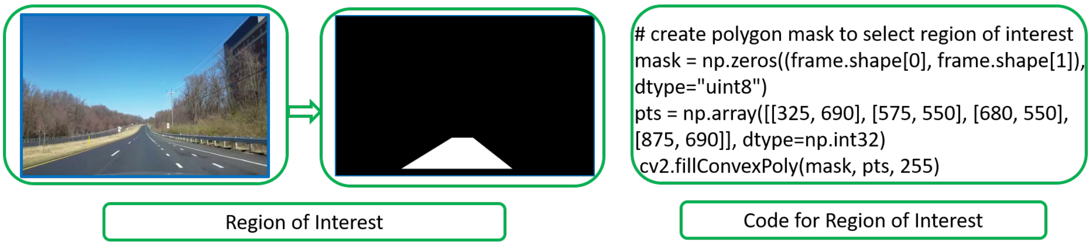


```
# Load the video
loadVideo = cv2.VideoCapture('testvideo2.mp4')
#load video for lane detection
# loop through until entire video file is played
while(loadVideo.isOpened()):
#  read frame from video
```

Figure 4.14: Video to Frame

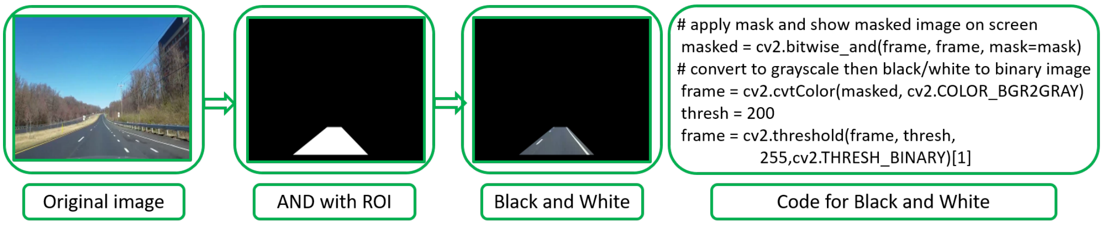
no more extra noise and lane line is there. Here is the image with code in the given figure: 4.16.It will make the calculation easy.

Blurred and Edge Detection Black and white images still contain the original color in the lane line. So, for detecting lane no need to color information and it is needed extra calculation. So Black and white image has been changed to blurred using Gaussian blur and then Canny edge detection has been applied for finding the edges. Canny edge detection and blur operation also have been seen in pothole detection. For this reason, this pictorial representation is not given here separately. It is given combine with lane detection which is in the next para.



```
# create polygon mask to select region of interest
mask = np.zeros((frame.shape[0], frame.shape[1]),
dtype="uint8")
pts = np.array([[325, 690], [575, 550], [680, 550],
[875, 690]], dtype=np.int32)
cv2.fillConvexPoly(mask, pts, 255)
```

Figure 4.15: Region of Interest and Mask



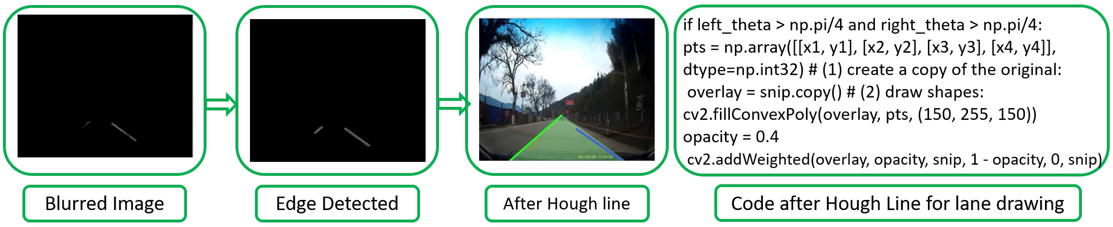
```
# apply mask and show masked image on screen
 masked = cv2.bitwise_and(frame, frame, mask=mask)
# convert to grayscale then black/white to binary image
 frame = cv2.cvtColor(masked, cv2.COLOR_BGR2GRAY)
 thresh = 200
 frame = cv2.threshold(frame, thresh,
                255,cv2.THRESH_BINARY)[1]
```

Figure 4.16: Black and Withe Image

Lane detection Hough line detection has been applied on the edge detected image for finding the lines. The values of the detected line are kept in the separated array, called the left and right array. In the appendix, section code will be added. From the array pixel positions are taken for drawing a convex polygon over the original image. This convex polygon contains the lane lines which are presented by blue and green lines. Here is the figure : 4.17 for finally lane detection.

4.3.3 Road Maintenance Sign Detection

Road Maintenance Sign Detection one of the mandatory sign detection. Whenever this sign is detected, many operations are needed for controlling the car. For example, maybe needed to follow another road and or need to slow down the car. For



```
if left_theta > np.pi/4 and right_theta > np.pi/4:
pts = np.array([[x1, y1], [x2, y2], [x3, y3], [x4, y4]],
dtype=np.int32) # (1) create a copy of the original:
 overlay = snip.copy() # (2) draw shapes:
cv2.fillConvexPoly(overlay, pts, (150, 255, 150))
opacity = 0.4
 cv2.addWeighted(overlay, opacity, snip, 1 - opacity, 0, snip)
```

Figure 4.17: Lane detected image

detecting Maintenance Sign, Machine learning has been used. One of the simple and most effective algorithms, KNN has been chosen. This algorithm needs to train with rich data set for getting better output. Data preparation is important for training and testing the system. The data preparation and the working procedure has been discussed below.

Data preparation

Data has been taking form the Institution of Neuro information, German[40]. 1500 images have been taken for each type of sign. The picture size is not the same. In this project, only one file will be used for training and test and test. So, different size images will be very difficult to find from a file. For this project, images have been resized to 20 by 20 pixels. After that Images have been arranged into 100 columns and 15 rows into an image for each class. Just have to add 1500 images using 2 loops for columns and rows. More data is required for getting a better result. Anyone can add new data set just by appending by increase the loop size. It is a scale-able system for farther improvement.

Work procedure

RRoad Maintenance Sign Detection is the most difficult task as its position is not fixed. Machine learning, K near Neighbor has been used for train and later this model has been used for recognizing the traffic maintenance sign. It has 3 parts, one is Processioning, second training and test and the last one is an individual image for testing its match. This one image has been used for the test. Pictorial algorithm is given hare,4.18

The second part is the test and training. From the grouped image, 70% image is taken for training and the rest 30% have been used for the test. Accuracy also has been done by using the KKN algorithm. At the time of training 20 by 20 space of the image has been considered as a picture.

From the third section, the Individual image has been sent for matching with the training data. The system takes a new image and sends to the training image. From the training image, the match value is returned. Comparing with this return value, a sign name can be found. But this input image contains a lot of noise and size of the image also not like train image size. For removing noise, the input image has been blurred using Gaussian blur then the image has been changed to 20 by 20 as like as the input image. Before making the same size the image covered to black and white using the threshold. All these processes have been done due to removing

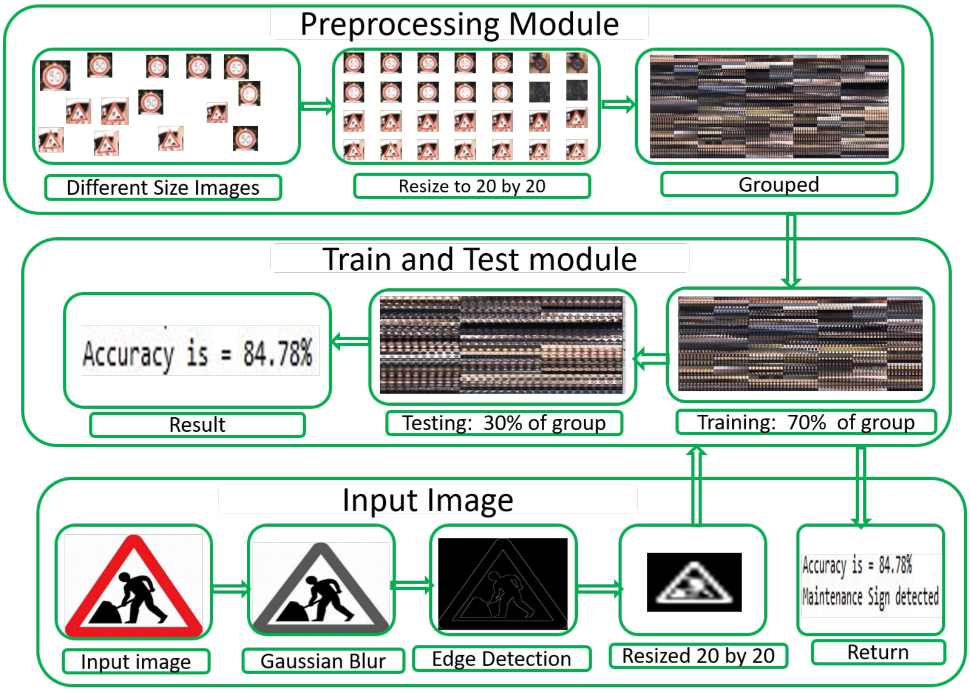


Figure 4.18: Maintenance Sign Detection Algorithm

the noise. After al these preprocess, the image has been sent to the KNN algorithm for matching the with the train data. KNN finds the match value and sends back to the calling function. The calling function compares the value and prints the input image content, traffic sign or not.

Summary Used technologies in this project have been presented very clearly alone with the process flow of each detection subsystem. Few important implemented python code also have been added for better understanding. All the resources have been described in detail.

5 Result and Evaluation

Textual and graphical representation presentation of the result of each subsystem will be presented separately.

5.1 Pothole

The memory occupied in every 3 seconds for the pothole detection is approximately 120 MB but it has little bit fluctuation which has been given in figure 5.1. The 13 seconds video has been taken which contains 389 frames. It takes 0.00035s MS for loading the video. The performance matrix of this video has been given here. Table :5.1 .The Pothole detection has been run in windows 10 OS using Core i5, CPU speed 2.60 GH. It has taken on an average 48% of the CPU usage.

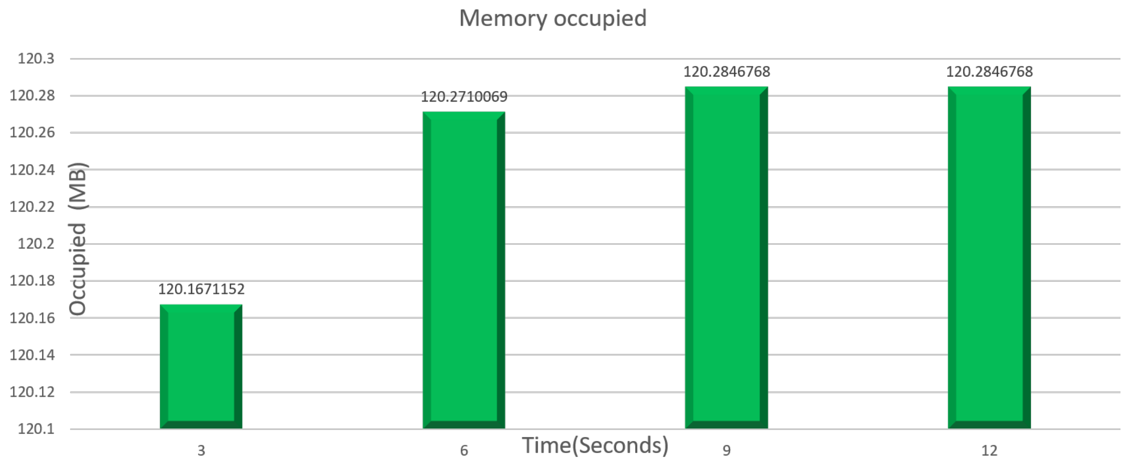


Figure 5.1: Memory consumption in every 3 seconds

Potholes are detected in a dynamic situation. The figure has been shown for one pothole and multiple potholes. In any case, it can detect pothole. It is shown in the figure:5.2. It can detect various size and shape images.

Here is the another table time which contains the total execution time ,single frame execution time and total number of frame for per second.

Performance Matrix	Value
Total time for complete execution	10.56 s
Single Frame processing Time	0.016 ms
Frame per Second	31,05

Table 5.1: Performance Matrix for Pothole

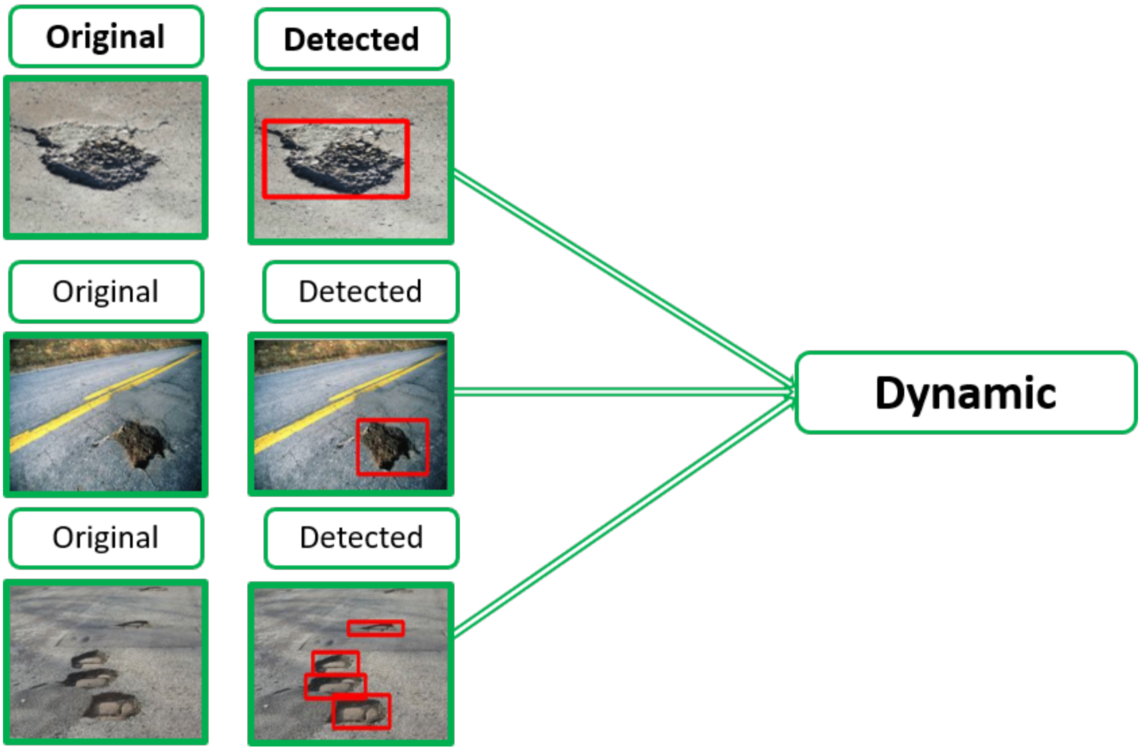


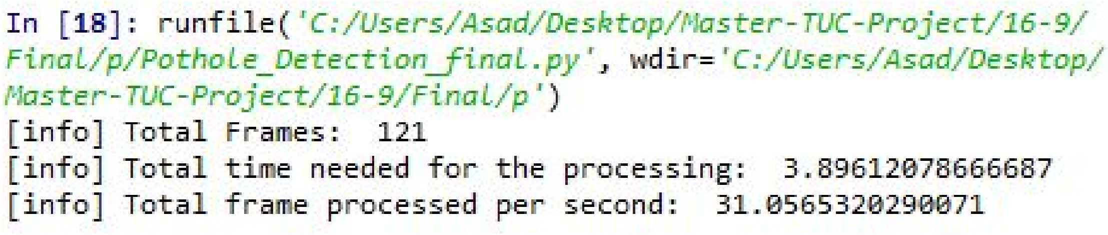
Figure 5.2: Dynamic detection of Pothole.

From the figure 5.3, It is clear that frame rate 31 which can be consider as real time output. It is clear that the average around 30 ms. This is good enough to say a real time operation.

Here is the table which give all the major image processing operations performance including average memory usage and CPU , Figure: 5.4

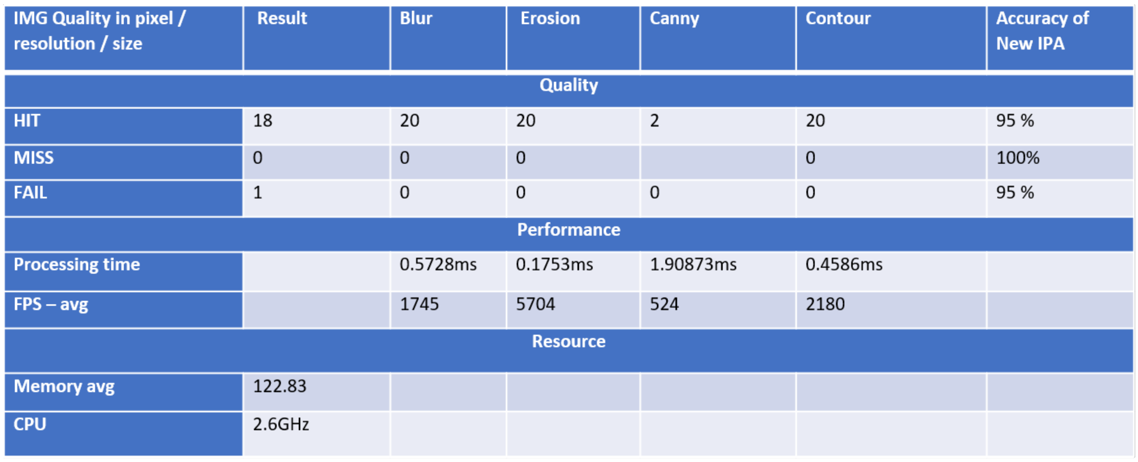
5.2 Lane Detection

Lane detection is a critical issue so many mathematical calculations are required for line detection. The performance is a big issue. Here is the performance matrix table for the lane detection 5.2.This has taken huge computational power. The lane



```
In [18]: runfile('C:/Users/Asad/Desktop/Master-TUC-Project/16-9/
Final/p/Pothole_Detection_final.py', wdir='C:/Users/Asad/Desktop/
Master-TUC-Project/16-9/Final/p')
[info] Total Frames:  121
[info] Total time needed for the processing:  3.89612078666687
[info] Total frame processed per second:  31.0565320290071
```

Figure 5.3: Result:Pothole.



IMG Quality in pixel / resolution / size	Result	Blur	Erosion	Canny	Contour	Accuracy of New IPA
Quality						
HIT	18	20	20	2	20	95 %
MISS	0	0	0		0	100%
FAIL	1	0	0	0	0	95 %
Performance						
Processing time		0.5728ms	0.1753ms	1.90873ms	0.4586ms	
FPS – avg		1745	5704	524	2180	
Resource						
Memory avg	122.83					
CPU	2.6GHz					

Figure 5.4: Pothole: IM operation performance overview .

detection has been run in windows 10 OS using Core i5, CPU speed 2.60 GH. It has taken on an average 60% of the CPU usage.

Performance Matrix	Value
Total time for complete execution	43.88 s
Single Frame processing Time	0.066 ms
Frame per Second	20.25

Table 5.2: Performance Matrix for Lane

Memory is a big issue. For testing, a video of 981 frames has taken to detect lane. It takes 0.04996538162231445 MS for loading the total video. Memory utilization has been tested for every 5 seconds. The statistic is given here in the figure:5.5 .In this chart, the fluctuation of memory utilization is found. The minimum usage of memory is 116.85 MB and the maximum usage of the memory is 120.738 MB.

Here is the table which give all the major image processing operations performance including average memory usage and CPU power, Figure: 5.8

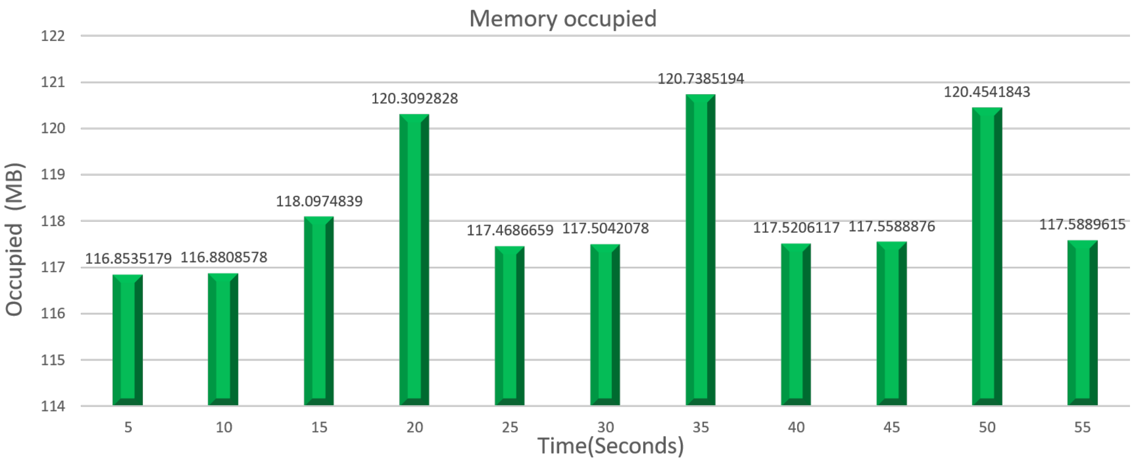


Figure 5.5: Memory consumption of Lane in every 5 second.

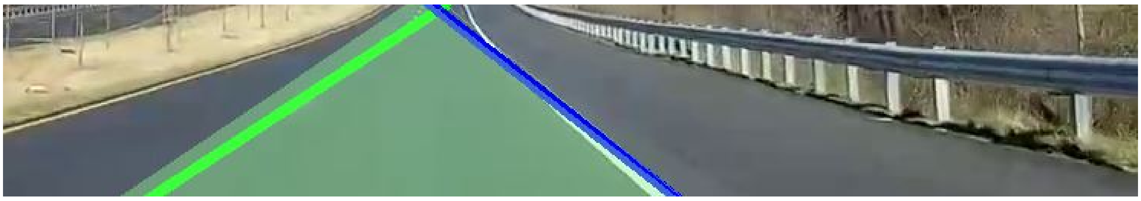


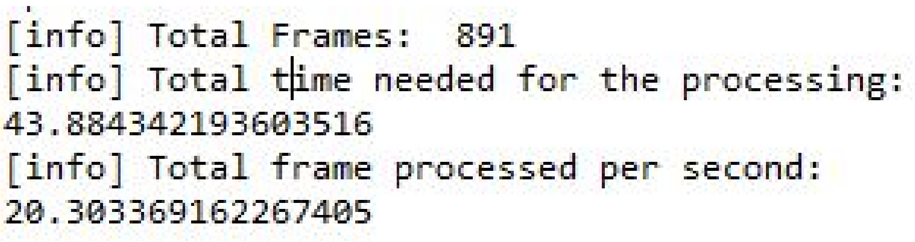
Figure 5.6: Lane detected

5.3 Maintenance Sign

Traffic sign has been tested with several pictures. The result has been added to the evaluation table. For this subsystem, machine learning has been used but some issues have been found. All these issues have been listed in the exception and future perspective chapter. The input and output result also has been given here. Here is the input image in the figure: 5.9. Accuracy is also a big issue for detecting the traffic maintenance sign. Here is the figure for accuracy in percent. 1500 images are used for training and testing. 70% of the image has been used for training and the rest of the image for testing. KNN algorithm has been for training and testing. the result is around 84% which is shown in the figure: 5.10.Overall performance is also given in the table 5.11.

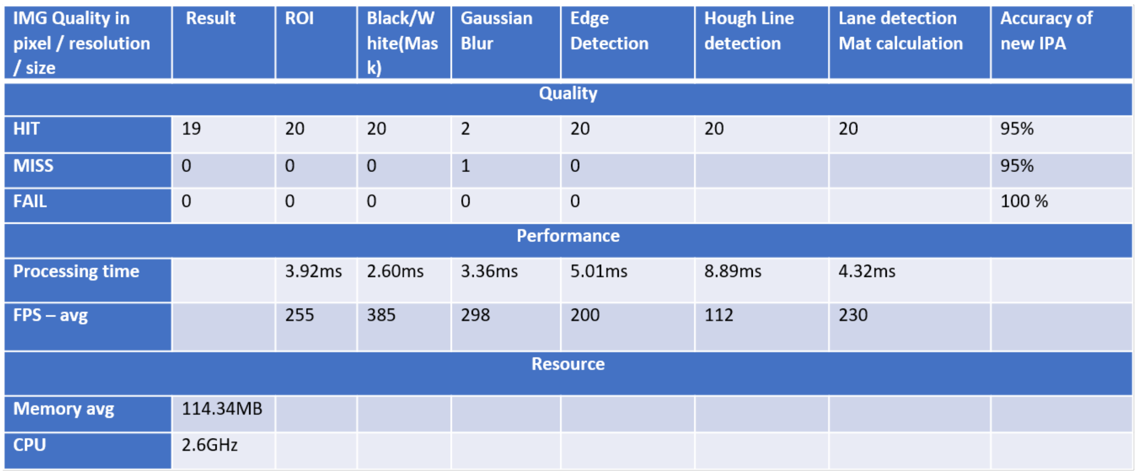
Here is the table which give all the major image processing operations performance including average memory usage and CPU power, Figure: 5.11

Summary: This system can nicely detect potholes, lanes and Traffic maintenance signs. Performance matrix tables have been added for understanding the useful re-



```
[info] Total Frames:  891
[info] Total time needed for the processing:
43.884342193603516
[info] Total frame processed per second:
20.303369162267405
```

Figure 5.7: Result:Total processing time, frame and frame per second for Lane detection



IMG Quality in pixel / resolution / size	Result	ROI	Black/White(Mask)	Gaussian Blur	Edge Detection	Hough Line detection	Lane detection Mat calculation	Accuracy of new IPA
Quality								
HIT	19	20	20	2	20	20	20	95%
MISS	0	0	0	1	0			95%
FAIL	0	0	0	0	0			100 %
Performance								
Processing time		3.92ms	2.60ms	3.36ms	5.01ms	8.89ms	4.32ms	
FPS – avg		255	385	298	200	112	230	
Resource								
Memory avg	114.34MB							
CPU	2.6GHz							

Figure 5.8: Lane Detection: IM operation performance overview .

sources and this application. There are performance overview tables for the detection of pothole, lane and Maintenance sign. Anyone can easily understand this project for farther improvement where and what needs to add or remove just by watching these tables. Some other statistics have been given using charts and screen prints about the usage of resources and performance.

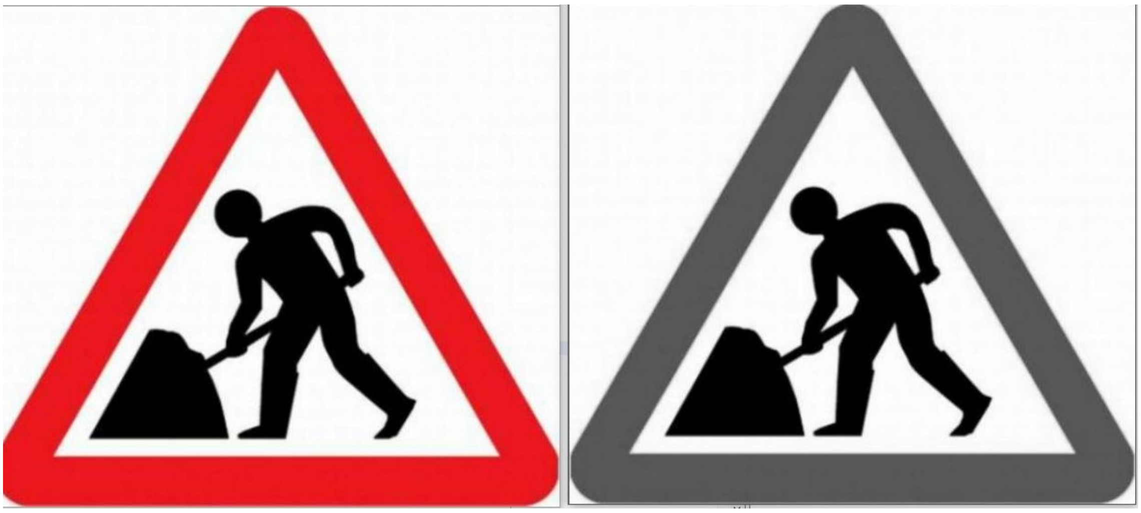
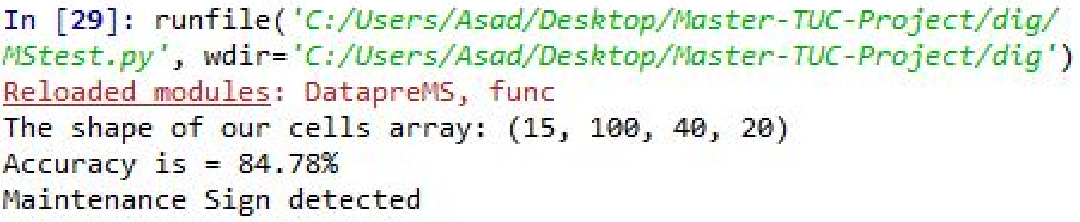
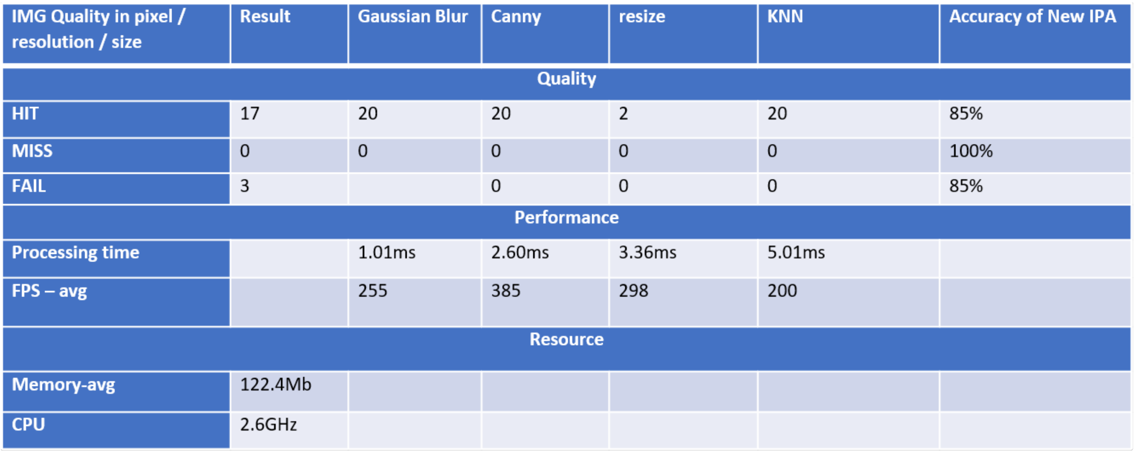


Figure 5.9: Input Maintenance sign.



```
In [29]: runfile('C:/Users/Asad/Desktop/Master-TUC-Project/dig/
MStest.py', wdir='C:/Users/Asad/Desktop/Master-TUC-Project/dig')
Reloaded modules: DatapreMS, func
The shape of our cells array: (15, 100, 40, 20)
Accuracy is = 84.78%
Maintenance Sign detected
```

Figure 5.10: Result:Maintenance Sign for accuracy.



IMG Quality in pixel / resolution / size	Result	Gaussian Blur	Canny	resize	KNN	Accuracy of New IPA
Quality						
HIT	17	20	20	2	20	85%
MISS	0	0	0	0	0	100%
FAIL	3		0	0	0	85%
Performance						
Processing time		1.01ms	2.60ms	3.36ms	5.01ms	
FPS – avg		255	385	298	200	
Resource						
Memory-avg	122.4Mb					
CPU	2.6GHz					

Figure 5.11: Performance overview for Maintenance Sign Detection.

6 Expectation and Future Perspective

In this cheater, complications of the modules and possible solutions have been presented in three subsections.

6.0.1 Pothole

The system can detect pothole in the various situation in near to real-time but still have some issues which can be deal in the future. The problem and the possible solution have been given in the separate section. Moreover, a suggestion also will be added for future work. In this project, the pothole is detected not as an object so, tracking is difficult. For tacking, we have to consider the detecting pothole as an object. A pothole area depends on pixels number of a contour area which is depending on the camera resolution. So, pixel and counter area are two issues.

Pixel The pothole area depends on the number of pixels. So, a higher resolution camera will increase the number of the pixel for the same picture. So, the number of the pixel has to increase for detecting the same potholes. Some for the lower regulation camera, need to reduce the number of pixels. the pixel value is needed to change depending on camera regulation. Using iPhone 5s which camera is 8 Megapixel counter area consider as bigger than 1000 pixels on the other hand for Huawei 30 Pro mobile has 20 megapixels camera which needs 1500 pixels to find the same pothole. This seems to be a big issue but the camera of the car will not be changed frequently. If the camera change also there is an alternative solution which will be described in the future work subsection.

Contour Area The contour area is the final considering issue for detecting pothole. The counter area is the pothole area. By filtering these contour areas, small holes remove from the potholes list. This area is also counted based on pixels. So, for higher resolution camera need to change the number of pixels for the area. The same issue for the lower resolution camera, need to reduce the pixel number for the contour area.

Future work

The pothole is detected perfectly but still, there is a problem for counting the number of potholes. It can be solved by treating the pothole area as an object. Then it will be easy for tacking.

The second problem of camera resolution. This can be solved in two ways. The first solution, A configuration file will be set for this module. where the user can set the camera resolution and from this resolution minimum pothole area will be calculated. Just by taking the current resolution and new camera regulation. The equation is given below. Lets, consider current resolution using the camera is Cu and image capturing capacity is Pu and using pixels number pa for pothole area. The new camera is Cn and the image capturing capacity is Pn. Then the pothole area, New-area will be changed as given in the equation:(6.1).This is can be used as a camera of the car is not frequently change but it may be difficult for some of the users.

$$NewArea = (Pu/Pn) * Pa \tag{6.1}$$

Another solution whenever the application will start working will read image properties for finding the camera resolution. If anything changes then it will call a function to change the area. This function will also work according to the given equation:(6.1). This procedure will be easier for the user but it be the small extra calculation every start time for the system. It is the best option for the user.

Suggestion Father improvement can be possible by cross-examining the pothole's image with machine learning techniques for classification as pothole and others. The detected pothole is not examined with any other method, the father cross-checks is possible with machine learning. It will not take too much time as only pothole images will pass to a machine learning module.

6.0.2 Lane

Region of interest Fixing the region of interest in the image is the most difficult. For different videos need to change the ROI. In case of wrong selection of a region of interest, the system will not find the lane. In the current system, it is done manually but it can possible to find the region of interest using color tacking.

It is also possible to solve by using the color segment. Considering the color on the left and the right, both lanes are possible to find out. But some time lane mark is missing. In that case, this procedure will not work. Details of this issue will be described in feature work sections. This is a big issue for the lane detecting subsystem.

Future Work

The camera is fixed for the onboard car. So, the image resolution will be fixed for any video. The space between the two-lane line also fixes for any road. The missing one lane line can be found by mathematical and geometrical calculations. Calculating the angel and the width between the lane lines, possible to find the beside the lane. In case of the road without a lane mark, the system will alert the driver.

6.0.3 Maintenance sign

Very often maintenance sign is not seen on road like other mandatory signs but once maintain sing is detected then the driver has to do many things. For example, a driver needs to control speed and or need to follow an alternative road. So, It is also the most important detection subsystem. Few issues are related to maintenance sign detection which is given below. Feature work will be also given below with some suggestions.

Noise In any case of noise removing is the biggest challenge for any kind of image processing technique. Here is also the same issue. Moreover, we cannot set the region of interest directly. The maintenance sign contains triangular. Finding triangular in every image is computationally expensive. For solving this issue hug data is required for training the system or need more filtering to make a clear input image like the training data.

Data set Only 1500 images have been used for training and testing. All the data from the Institute of Neuroinformatik, German [40]. This is not sufficient for training and test by comparing the MNIST data set. The most popular data set for handwriting detection is MNIST Dataset. The number of images is 60 thousand for only 10 classes. Using KNN and python tool 96.20% accuracy has been found for MNIST Dataset[41]. In this project, only 1500 images have been used for each class. Compare to that, this number is so less. For this reason, prepossessing has been done for getting more accuracy still needs more data for more accuracy.

Feature work

There are two main factors for not getting higher accuracy. The first one is the data set and the second one in the noisy input image. From the complication section, it is very clear that the image data set is not rich enough. The highly rich data set is required for getting more accuracy. For solving the noisy issue. The image can be filtered using some other sophisticated algorithm which will eliminate the noises.

Suggestion YOLO is a deep learning framework. Several issues have been found for training, It is resource hunger and so hard to training but these approaches provide better performance using GPU. An experiment has done but it took lone time to train. It takes 3 days for training only 30 images of two classes, using Toshiba Satellite E 8050 computer. It contains Processor, Intel(R) Core(TM) i5-3230M CPU @ 2.60GHz, 2600 Mhz12 GB RAM, 2 Core(s), 4 Logical Processor(s).

Summary: All the limitations and their possible solutions to this project have been presented separately with future work.

7 Conclusion

The most frequently used transport means road transport. Every household wants a car for convenient transportation. As a result, the number of transport is increasing and this is leading to increasing the road accident. So, for keeping the road safe, this project has been developed. This project has been developed for alerting the driver. The driver can watch the potholes, lane line and maintenance sign. The system will alert the driver but the output of this system can be used for the controlling unit of the vehicle. In addition to that, some other module is possible to integrate with this system. The Traffic maintenance sign detection subsystem can possible to extend by adding class. For example, Road Maintenance Sign Detection has done in this project but adding more other classes of signs images are possible to detect. So, it is a scale-able system. Lane detection is 100% correct but needs to decrease the processing time. The pothole is also correctly detecting in nearly real-time.

Bibliography

[1] M. R. Endsley, "Toward a theory of situation awareness in dynamic systems," in *Situational Awareness*, pp. 9–42, Routledge, 2017.

[2] J. Nine, S. Saleh, O. Khan, and W. Hardt, "Traffic light sign recognition for situation awareness using monocular camera," 07 2019.

[3] "Opencv." `https://opencv-python-tutroals.readthedocs.io/en/latest/py_tutorials/py_imgproc/py_canny/py_canny.html/`. Accessed: 01-10-2019.

[4] "Knn algorithm." `https://www.kdnuggets.com/2016/01/implementing-your-own-knn-using-python.html/`. Accessed: 26-09-2019.

[5] "Kernel 2d to 3d." `https://medium.com/datadriveninvestor/machine-learning-basics-support-vector-machines-358235afb523/`. Accessed: 1-10-2019.

[6] J. S. Miller and W. Y. B. Glueck, "Us department of transport (lni)," 2014.

[7] D. T. A. Ghebreyesus, "Global status report on road safety 2018," 2018.

[8] Y. Kumar and Y. Jain, "Research aspects of expert system," *International Journal of Computing & Business Research*, 2012.

[9] H. Bahşi, V. Dieves, T. Kangilaski, P. Laud, L. Mõtus, J. Murumets, I. Ploom, J. Priisalu, M. Seeba, E. Täks, *et al.*, "Mapping the information flows for the architecture of a nationwide situation awareness system:(poster)," in *2019 IEEE Conference on Cognitive and Computational Aspects of Situation Management (CogSIMA)*, pp. 152–157, IEEE, 2019.

[10] J. Nine, S. Manoharan, and W. Hardt, "Concept of the comprehension level of situation awareness using an expert system," 08 2019.

[11] Y. Erdani, "Developing backward chaining algorithm of inference engine in ternary grid expert system," *International Journal of Advanced Computer Science and Applications*, vol. 3, no. 9, 2012.

[12] B.-J. Park, C. Yoon, J.-W. Lee, and K.-H. Kim, "Augmented reality based on driving situation awareness in vehicle," in *2015 17th International Conference on Advanced Communication Technology (ICACT)*, pp. 593–595, IEEE, 2015.

[13] Y. Jo and S. Ryu, "Pothole detection system using a black-box camera," *Sensors*, vol. 15, no. 11, pp. 29316–29331, 2015.

[14] A. Danti, J. Y. Kulkarni, and P. Hiremath, "An image processing approach to detect lanes, pot holes and recognize road signs in indian roads," *International Journal of Modeling and Optimization*, vol. 2, no. 6, p. 658, 2012.

[15] B. Dorj and D. J. Lee, "A precise lane detection algorithm based on top view image transformation and least-square approaches," *Journal of Sensors*, vol. 2016, 2016.

[16] J. Lin and Y. Liu, "Potholes detection based on svm in the pavement distress image," in *2010 Ninth International Symposium on Distributed Computing and Applications to Business, Engineering and Science*, pp. 544–547, IEEE, 2010.

[17] K. Vigneshwar and B. H. Kumar, "Detection and counting of pothole using image processing techniques," in *2016 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC)*, pp. 1–4, IEEE, 2016.

[18] A. Akagic, E. Buza, and S. Omanovic, "Pothole detection: An efficient vision based method using rgb color space image segmentation," in *2017 40th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, pp. 1104–1109, IEEE, 2017.

[19] X. An, M. Wu, and H. He, "A novel approach to provide lane departure warning using only one forward-looking camera," in *International Symposium on Collaborative Technologies and Systems (CTS'06)*, pp. 356–362, IEEE, 2006.

[20] C. Y. Low, H. Zamzuri, and S. A. Mazlan, "Simple robust road lane detection algorithm," in *2014 5th International Conference on Intelligent and Advanced Systems (ICIAS)*, pp. 1–4, IEEE, 2014.

[21] J. He, H. Rong, J. Gong, and W. Huang, “A lane detection method for lane departure warning system,” in *2010 International Conference on Optoelectronics and Image Processing*, vol. 1, pp. 28–31, IEEE, 2010.

[22] G. Deng and Y. Wu, “Double lane line edge detection method based on constraint conditions hough transform,” in *2018 17th International Symposium on Distributed Computing and Applications for Business Engineering and Science (DCABES)*, pp. 107–110, IEEE, 2018.

[23] S. B. Wali, M. A. Abdullah, M. A. Hannan, A. Hussain, S. A. Samad, P. J. Ker, and M. B. Mansor, “Vision-based traffic sign detection and recognition systems: Current trends and challenges,” *Sensors*, vol. 19, no. 9, p. 2093, 2019.

[24] H. Fleyeh, R. Biswas, and E. Davami, “Traffic sign detection based on adaboost color segmentation and svm classification,” in *Eurocon 2013*, pp. 2005–2010, IEEE, 2013.

[25] C. Rahmad, I. F. Rahmah, R. A. Asmara, and S. Adhisuwignjo, “Indonesian traffic sign detection and recognition using color and texture feature extraction and svm classifier,” in *2018 International Conference on Information and Communications Technology (ICOIACT)*, pp. 50–55, IEEE, 2018.

[26] W. Garage, “Opencv wiki,” 2006.

[27] “Cuda gpu port.” `https://opencv.org/cuda/`. Accessed: 23-09-2019.

[28] “Opencl announcement.” `https://opencv.org/opencl/`. Accessed: 23-09-2019.

[29] M. P. RAVI and A. ASHOKKUMAR, “Analysis of various image processing techniques,” *International Journal of Advanced Networking & Applications (IJANA) Volume*, vol. 8, p. 86–89, 2017.

[30] D. Cardani, “Adventures in hsv space,” *Laboratorio de Robótica, Instituto Tecnológico Autónomo de México*, 2001.

[31] G. C. Shapiro, Linda G. & Stockman, “Embedded real-time video processing system on fpga,” in *Computer Vision*, (New Jersey), pp. 279–325, Prentice Hall, 2002.

[32] S. Ravi and A. Khan, "Morphological operations for image processing: understanding and its applications," in *Proc. 2nd National Conference on VLSI, Signal processing & Communications NCVSComs-2013*, 2013.

[33] J. Shin, E. Lee, K. Kwon, and S. Lee, "Lane detection algorithm based on top-view image using random sample consensus algorithm and curve road model," in *2014 Sixth International Conference on Ubiquitous and Future Networks (ICUFN)*, pp. 1–2, IEEE, 2014.

[34] B.-J. M. Webb-Robertson, M. M. Matzke, and C. S. Oehmen, "Dimension reduction via unsupervised learning yields significant computational improvements for support vector machine based protein family classification," in *2008 Seventh International Conference on Machine Learning and Applications*, pp. 457–462, IEEE, 2008.

[35] Y. Zhong, "The analysis of cases based on decision tree," in *2016 7th IEEE International Conference on Software Engineering and Service Science (ICSESS)*, pp. 142–147, IEEE, 2016.

[36] V. John, K. Yoneda, Z. Liu, and S. Mita, "Saliency map generation by the convolutional neural network for real-time traffic light detection using template matching," *IEEE transactions on computational imaging*, vol. 1, no. 3, pp. 159–173, 2015.

[37] T. Guo, J. Dong, H. Li, and Y. Gao, "Simple convolutional neural network on image classification," in *2017 IEEE 2nd International Conference on Big Data Analysis (ICBDA)(*, pp. 721–724, IEEE, 2017.

[38] M.-T. Duong, T.-D. Do, and M.-H. Le, "Navigating self-driving vehicles using convolutional neural network," in *2018 4th International Conference on Green Technology and Sustainable Development (GTSD)*, pp. 607–610, IEEE, 2018.

[39] "Language statistic." `https://www.tiobe.com/tiobe-index/`. Accessed: 15-10-2019.

[40] "Data sign." `http://benchmark.ini.rub.de/?section=gtsrb&subsection=dataset/`. Accessed: 02-10-2019.

[41] T. Makkar, Y. Kumar, A. K. Dubey, Á. Rocha, and A. Goyal, "Analogizing time complexity of knn and cnn in recognizing handwritten digits," in *2017 Fourth*

International Conference on Image Information Processing (ICIIP), pp. 1–6, IEEE, 2017.

Selbstständigkeitserklärung

Add this Document: ou can also link to bare URLs without an additional description: `https://www.tu-chemnitz.de/studentenservice/zpa/formulare/allgemeineformular` `abschlussarbeit_selbststaendigkeitserklaerung.pdf`

//