FORMULATION OF A NEW IMAGE PROCESSING MECHANISM FOR AUGMENTED REALITY BASED NAVIGATION IN AUTONOMOUS VEHICLES FOR STATIONARY OBSTACLES

A Thesis

Submitted in partial fulfillment of the requirements for the award of the degree of

DOCTOR OF PHILOSOPHY

in

Computer Applications

By

Supriya Sharma

41400201

Supervised By
Dr. Lovi Raj Gupta

Co-Supervised By
Dr. Mithilesh Kumar Dubey

LOVELY PROFESSIONAL UNIVERSITY

PUNJAB

2020

DECLARATION

I declare that this thesis entitled "Formulation of New Image Processing Mechanism for Augmented Reality based Navigation in Autonomous Vehicles for Stationary Obstacles" has been prepared by me under the guidance of Dr. Lovi Raj Gupta, Executive Dean, Lovely Faculty of Technology and Sciences, Lovely Professional University and Dr. Mithilesh Kumar Dubey, Associate Professor, School of Computer Applications, Lovely Professional University. No part of this thesis has been formed the basis for the award of any degree or fellowship previously.

Supriya Sharma
School of Computer Applications,
Lovely Professional University,
Jalandhar Delhi G.T. Road (NH-1),
Phagwara, Punjab, 144411
India.

Date :

CERTIFICATE

We certify that Ms Supriya Sharma has prepared her thesis entitled "Formulation of new Image Processing mechanism for Augmented Reality based Navigation in Autonomous Vehicles for Stationary Obstacles" for the award of Ph.D. Degree of Lovely Professional University under our guidance. She has carried out the work at the Department of Computer Applications, Lovely Professional University, Phagwara (India).

Supervisor:

Dr. Lovi Raj Gupta
Executive Dean
Lovely Faculty of Technology and Sciences
Lovely Professional University
Phagwara, Punjab-14441, India

Date:

Co- Supervisor :

Dr. Mithilesh Kumar Dubey
Associate Professor
School of Computer Applications
Lovely Professional University
Phagwara, Punjab-14441, India

Date:

PREFACE

Augmented-reality based autonomous navigation system is the upcoming area which has recently attracted much attention of the research community. Many software and automobile companies are into this work. Advancements in areas such as automation, augmented reality, navigation has made augmented-reality based autonomous navigation system possible. GPS, sensors and vision-based approaches have been proposed for implementing the existing autonomous navigation systems. However, these techniques were unable to achieve the human way of navigation. In this work, we have introduced a novel image processing based mechanism for augmented reality autonomous navigation systems. This will minimize the gap between the human way of navigating and the computer-assisted navigation. The system will navigate autonomously and intelligently with the help of prominent markers the way as human navigates on the road. Without GPS and sensors, the system will navigate from one place to another with the help of a monocular camera. As the system will navigate autonomously in the outdoor environments, it will also detect the obstacles encountered on the path. Therefore, in this work, the focus is on the implementation of the novel image processing based localization mechanism named as advanced navigation marker identification and categorization system (ANMIC) and a real-time stationary obstacle detection system based on the geometric parameters. The performance of the proposed approach has shown that it is able to achieve high accuracy and improve the performance quite substantially compared with the other existing techniques.

The ANMIC is a context dependent, multi-perspective, multi-level concept based on image capturing. The field related to autonomous vehicle among map navigation has historically been informed by knowledge from narrow functional areas. While some effort towards producing a broader perspective of the concept has been made, nonetheless, autonomous vehicle in navigation continues to be largely

eclectic with little development of concept within the consensus on its conceptualization and research methodological bases. The work proposes the method of image capturing, its categorization for run time video. MATLAB has been used to implement the concept. In the present work, an algorithm is proposed which creates image categorization mechanism that leads to detection of landmarks from run time video; vehicle navigation is further facilitated by the detected markers using optical flow approach; categorization mechanism detects the non-trivial markers by removing triviality step by step.

Real-time obstacle detection in the urban environments is the challenging task which has attracted much interest from the research community. Detection of obstacles while navigating on the road can help in the reduction in the number of road accidents and enabling the path planner to choose the most efficient and safe route towards the desired destination. Sensor-based techniques have already been used for obstacle detection in the previous works. However, this work focuses on the concept of vision-based stationary obstacle detection for the navigation of an un-manned ground vehicle. An algorithm based on image processing techniques has been implemented to achieve the desired results. We have considered three parameters: immovability, line of sight and geometric conditions for the obstacle detection purpose. These obstacles are detected at a particular distance using the camera calibration process. The work is classified into three main stages: 1. Movable and stationary objects are distinguished from the real-time scene; 2. Acknowledged stationary objects are checked for their presence within the line of sight of the vehicle; 3. The resultant objects satisfy certain geometric thresholds. A monocular moving camera has been used to capture the real-time scene, and, the experimental implementation has been tested successfully in the outdoor environments using the MATLAB tool. The results obtained in the comparison to the state of the art techniques show that the proposed algorithm is highly efficient and accurate for the detection of stationary obstacles encountered in the real-time environment without having any prior knowledge of the obstacles.

ACKNOWLEDGEMENT

First and foremost, I would like to express my sincere gratitude to my supervisor Dr. Lovi Raj Gupta for always providing me his valuable time and support. Through his immense knowledge and patience, I was able to accomplish all my tasks on time.

I wish to thank my co-supervisor Dr. Mithilesh Kumar Dubey for providing his precious time and his enormous knowledge. He has been actively interested my work and has always been available to advise me. I am very grateful for his patience, motivation and enthusiasm.

Sincere thanks to Mr. Saleem for his continuous motivation which made this journey easy.

A warm thanks to Mr. Rahul for his endless sacrifices and selfless support that made everything possible on this path.

I also want to mention here my gratitude towards my parents and my younger brother Mr. Vishal who went through a lot of ups and downs but still continued to shower their endless love and care which always boosted my morale to carry forward this journey and make my efforts visible.

(Supriya Sharma)

CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ACRONYMS

AR	—	Augmented Reality
GPS	—	Global Positioning System
HMD	—	Head Mounted Display
HUD	—	Head Up Display
IMU	—	Inertial Measurement Unit
LIDAR	—	Light Detection and Ranging
LoS	—	Line of Sight
MATLAB	—	Matrix Laboratory
RFID	—	Radio Frequency Identification
ROI	—	Region of Interest
SIFT	—	Scale-Invariant Feature Transform
SURF	—	Speeded Up Robust Features

CHAPTER – 1

INTRODUCTION

Automotive industry is advancing by leaps and bounds. Every day, there is an evolution in the field of autonomous navigation, and still, there is a lot more to be done. According to the WHO report published in the year 2018 [1], "Approximately 1.35 million people die every year as a result of road traffic crashes. Road traffic injuries are the leading cause of death for children and young adults aged 5-29 years". The major causes of these road accidents are human errors such as driver fatigue, emotional imbalance, or illness. As the autonomous navigation systems are unaffected by the aforementioned reasons, so these systems will result in reduced number of road accidents as well as congestions. Various pros of these systems for society got the heed to the various software and automotive companies to work on it.

1.1 AUTONOMOUS NAVIGATION SYSTEM

An autonomous navigation system can be defined as a system in which a vehicle navigates from a source to destination with or without human intervention [2][3][4][5]. This system [6] is diversified into five levels ranging between zero level automation to a fully automated system. These systems rely on a hybrid approach with the help of GPS, sensors, image processing techniques, and many more in which the GPS and satellite data are instructive for providing the location information in the form of co-ordinates [7][8][9][10], and sensors are beneficial for the detection of various obstacles coming in the path. So, an autonomous vehicle takes input from various sources and makes its conclusion to run on a distinct path. Various technologies discussed in [11] that are contributing to the execution process of these systems are image processing, sensors, satellite data, visual odometry, computer vision, and augmented reality.

1.2 AUGMENTED REALITY

Augmented reality (AR) is a technology that envelops the virtual information on the real-world environment [12][13]. AR is the trending area of research that aims to enhance the real world by overlaying computer-generated data on top of it. It is a dissimilarity of the virtual environment or virtual reality [14] that adjuncts reality rather than completely replacing it. AR takes either the digitally combined items or genuine inputs from clients as signals, voice commands, and eye gaze and creates overlays on a real-world scenario that can be seen by a patron. It includes subsidiary layer of information generally as computer-supported graphics to amalgamate additional points of interest in the physical (real) world around us. AR-technology can be diversified as marker-less approach and marker based AR approach [15][16][17]. Figure 1.1 represents the persistent scale showing all the possible variations of real and virtual objects.

FIGURE 1.1: *Reality-virtuality continuum*

- **Markerless Augmented Reality:** It is the process of overlaying the virtual data on the 3D scene irrespective of having any prior knowledge of the real-world environment.
- **Marker-based Augmented Reality:** It is the process of overlaying the computer-generated information on the 3D scene with the help of markers. These markers are pasted in the environment so that the AR application withdraws the information only by scanning the previously-stored markers.

1.3 APPLICATIONS OF AUGMENTED REALITY

Augmented reality technology is contributing to almost every field of society. Its tremendous benefits to society are countless. The application areas of augmented reality are as follows:

- **Education and Training:** In the education field, the trainers are trying to use visualization techniques to provide the user with a better elaboration of the concept. AR big bang theory is a particular illustration of such applications in which the notion of big bang theory is elucidated by a user-friendly and an interactive platform.

- **Military**: AR is used in designing military training by exhibiting the real battlefield scene, augmented with the computer-generated information on it.
- **Gaming**: Augmented reality provides the gaming platform a more interactive and realistic environment with its visualisation and tracking techniques. Today, there are various leisure activities like AR car racing, and more are getting the attention of the users for a better gaming adventure. [18]
- **Entertainment**: AR has been put in the application in the entertainment industry as well, including games, videos. This technology has made the videos more fascinating and user-friendly.
- **Medical**: AR has proven to be very fruitful in the medical industry too. It superimposes the medical information of a patient on the patient itself for providing a superior diagnosis.
- **Tourism**: AR also intensified the globetrotting experiences where the nearby details of the surroundings are easily displayed on the tourists' system for making them familiar with the new locations [19][20][21].

Therefore, augmented reality is contributing to almost every area, from entertainment to education. In the military, head-mounted display (HMD) units have been proven beneficial. This technology has also supported in the medical field, especially in ultrasound imaging and surgeries. In archaeology, it is used for enhancing the perception of the real-world environment. In teaching and learning areas, it helps to understand even the complex concepts easily with the help of visualization techniques. AR has proven to be a big game-changer in the field of navigation also [22][23][24][25][26].

1.4 AUGMENTED REALITY: ROLE IN AUTONOMOUS NAVIGATION SYSTEMS

The major strength of augmented reality is its intuitive depiction of information where the user perceives virtual and real objects as co-existing in the same space [27]. There has been a lot of effort being put in this field, which is going on even today and still a lot more will be required afterwards. AR navigation systems are for indoor navigation as well as for the outdoor navigation purpose.

- **Indoor navigation:** Augmented reality is useful in case of indoor navigation. As GPS signals do not work efficiently within the buildings; therefore, the AR systems use Wi-Fi, Bluetooth, and RFID for locating the user's current locale. The navigation system uses the pre-built databases for localization process to implement the pasted markers detection process.
- **Outdoor navigation:** AR navigation systems extract precise information about road layout, traffic signs, traffic signals, signboards and everything else around it from the real-world environment [28]. Vision systems do not only perceive the outside information of the locomotive to detect and track the roads and avoid hitting obstacles or pedestrians but simultaneously it also constantly discerns information about the cabinet of the vehicle to monitor the vigilance of the driver and even predict his/her intent of driving [30][31][32][33][34].

The typical architecture for the augmented reality based autonomous navigation system given by Nartz [10] is explained in Figure 1.2.

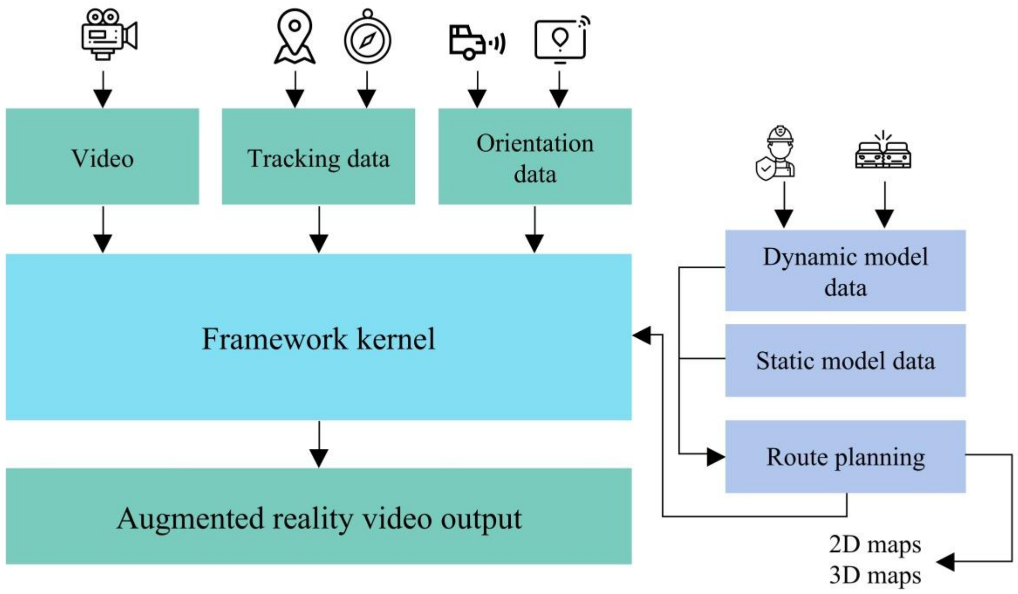

FIGURE 1.2: *Basic architecture of Augmented Reality Navigation System.*

In figure 1.2, the architecture of the augmented reality based navigation system by Nartz depicts that the data is extracted from various input channels like GPS, sensors, gyros, compass, camera etc. and this information is collectively used for the navigation of the vehicle. GPS receiver is used for the vehicle‘s localization. Wheel sensors are used to keep the track of vehicle by using GPS signal’s orientation information in case of GPS failure. This information is gathered with the help of trackers like compass and gyros. This system has also been prepared for indoor navigation. The virtual 3D information is generated by using the static model data like 2D and 3D maps, dynamic model data such as men at work along with the route planning algorithm. The information extracted from the sensors and the maps is overlaid on the captured image of the real world environment from the camera. This information is then outputted to the augmented reality display unit.

1.5 NEED FOR THE AUGMENTED REALITY BASED AUTONOMOUS NAVIGATION SYSTEM

Recent advancement in every field motivated the technological revolution in the automotive industry too. If one has to navigate from one place to another, they have to rely on the driver. Non-drivers (old aged, disabled, non-drivers, etc.) are not able to navigate freely. The automotive industry would be of great help to beginners.

This will also lessen the accident rate as these systems are unaffected by factors like driver fatigue, disequilibrium, or illness [35][36]. Musk says Tesla's 'Autopilot' cuts accident rates by 50% more.

1.6 AUGMENTED REALITY AUTONOMOUS NAVIGATION SYSTEM

This section comprehends various roles of the augmented reality based navigation system. The generalized components here and there around which the augmented reality navigation system revolves are environmental conditions, driver’s behaviour, and vehicle’s state. So, this section describes all the current efforts in the detection of the various constituents of the augmented reality based autonomous navigation system.

These navigation systems while navigating from one place to another; also suspect the environment for a safer navigation experience. Lane detection, pedestrian detection, obstacle detection, traffic scene detection, evaluation of driver’s behaviour, and localization of the vehicle are the basic components of the AR navigation system [37][38]. Figure 1.3 explains the distinctive components in the augmented reality based autonomous navigation system.

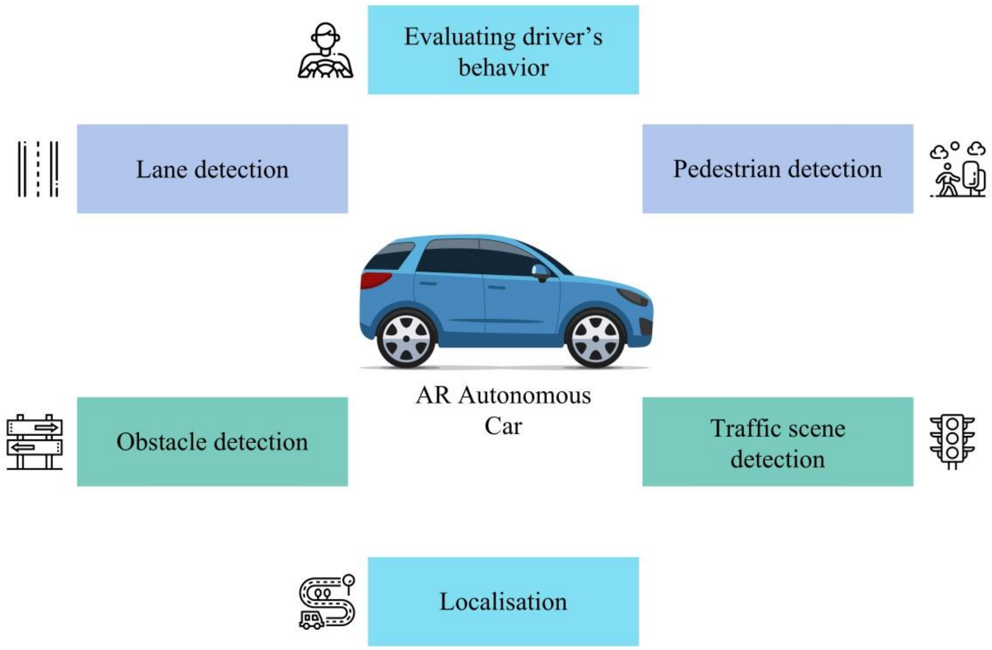

FIGURE 1.3: *Components of AR Navigation system*

1.6.1 Lane Detection

Lane detection is the process of detecting the lanes on the road. Utmost road accidents are a cause of high speed, overtaking, sudden change of lane, low visibility or due to bad weather, and so forth. A lot of work has been done to work out these problems. According to Dang [39], the classic lane detection system follows a process with the following steps:

- A frame is captured from the camera
- Image is converted to grayscale image
- Threshold process
- Lane mark or lane pixels are used
- Final filter

In the driving assistant system, lane detection is primarily done by the optical approach. However, the appearance of the lane is too simple to extract the distinct features which lead to the wrong detections. Therefore, sensors, along with the deep learning techniques, have been used to predict the road. Radar, along with the map data, provides the longitudinal information, and sensor data is used for the road course estimation [40]. LaneNet, a deep neural network technology, is used for lane detection in [41][42]. Convolution neural network has also been implemented to get the desired results for the same [43].

A faster lane detection algorithm also exists that has no adverse impact on the accuracy of the detection process. The results are achieved by using the Rasberry Pi 2 as a controller along with the camera with the reduced resolution [40].

Gaussian filtering, RANSAC algorithms are conjointly used for lane detection purpose [44], that have detected all the lanes in the still images of the street in an urban environment in figure 1.4.

FIGURE 1.4: *Lane detection: green colour for highlighting the lines on the lanes*

1.6.2 Driver's behaviour

As far as safety is concerned, the driver plays a major role in the navigation field, as the majority of the accidents occur due to the errors committed by human drivers. There is a greater chance of collisions if the driver is tired, sick, or distracted due to infinite reasons. So, for the safe navigation, it was realized that monitoring the surroundings only is not important but monitoring the state of the driver is also important for assuring safety. 25% of the road accidents are due to the driver's distraction, according to [45]. Therefore, for evaluating the driver's behaviour, vision-based approaches have been introduced. Some researches have been done on the vehicle's driver behaviour, and on analysing the neighbouring driver's reaction to avoid the accidents [46]. An algorithm for analysing the driver's behaviour has been proposed by [47] [48]. Their work deals with the features for the evaluation process of the neighbouring driver’s behaviour. Those features are: Left move, right move, left to right, background of the driver. Based on these features, the alertness of the driver is assessed, and this results in averting the accident rate.

In the paper [49], sensors are combined with HUD to keep track of the driver’s attention. Here, HUD technology is implemented in autonomous cars to keep a check on the driver’s eyes. The main hurdle faced in this work is that if a lot of information is being displayed on the windshield, it may result in the distraction for the driver.

1.6.3 Pedestrian Detection

In traffic-related injuries, the pedestrian accident is the second major factor. To reduce these types of accidents, sensor-based approaches are used. A lot of research on pedestrian detection has been done until now [50]. According to a study, “Video sensors are primarily used for the detection of people as they have no interference with the environment”. Therefore, a stereo vision approach is used for this purpose. The pedestrian detection system can be divided into two steps: pedestrian recognition and ROI.

The pedestrian detection is not the only contributor for avoiding the collision; the pedestrian dynamics and their behaviour also needs to be studied [51]. The visible and the non-visible sensors are used for predicting the collisions between the pedestrian and the vehicle [33][52].

1.6.4 Traffic Scene Detection

Traffic signs seem to be really helpful for assuring road safety as they provide the necessary information for assisting the user in navigating securely on the road. Traffic lights, traffic signs, and route boards are there to follow for safe navigation. So, they need to be detected. Quality work has been done in this field until now. Traffic sign recognition is nearly impossible to be processed by using sensors [53], and they are very crucial for navigation purposes. They can act as markers for the decision making process. Usually, colour and shape models are used to categorize the road signs. A vision-based algorithm for the traffic scene detection has been proposed in which the results for lane detection, traffic light, and sign detection are achieved efficiently. Some autonomous cars, like, Waymo [54], use cloud computing for analysing the traffic data.

Colour-based approach has been used in the traffic sign detection and recognition process. Blobs are extracted from the input image, then the colour segmentation process is carried out on these blobs using linear support vector machines (SVM), and this is known as the classification process. The identification of these signs is performed by pattern recognition [55]. There are other approaches supporting this field in which Hough transformation is used for the detection process, and neural network technology is best suited for the classification process, whereas at the final step, recognition of the traffic signs is done by Kalman filter. The average processing time of this system is 30 milliseconds per frame [56][57][58][59][60][61][62].

The traffic signs are recognised by extracting their geometric features also. The shape of the sign, along with its background color, has been used for this purpose. Image processing techniques are also applied in the recognition process [63].

1.6.5 Localization

Landmarks can act as an external reference point that can be easily observed in navigation. Humans often use landmarks for navigation. Anything that is easily recognizable and helpful for the navigation process can qualify as a landmark. For example, hoardings, buildings, road signs, etc. are some of the potential landmarks. Current navigation systems can be made more effective if landmarks are added as cues for navigation, especially in unfamiliar destinations. Though route guidance systems are totally dependent on the precision of the map database and availability of GPS signal, yet they are very functional [64].

AR for navigation and information browsing in an urban environment has been reviewed for the tourist guidance system. In the information browsing unit, the surrounding data is fetched. In navigation mode, the target and destination are set, according to which the shortest path is computed in a known network of possible routes. The system is interactive and reacts to the user’s movements. Information is displayed on the series of waypoints that are visualized as popup bubbles, appears to be standing in the environment, and the user can click on those bubbles to check the related information. The information can be images, text shown to the user in a heads up display. This work tries to give an outlook on possible future user interfaces for location-based services [65].

In [64], GPS, along with the map data, is used for calculating the various possible routes (especially the shortest route) to a specified destination. Navigation is accomplished with the help of visual and auditory information in the form of map overview and turns by turn instructions to the drivers, respectively.

This paper describes the ‘4’ levels of tasks in the AR-based autonomous navigation, i.e., scene capture, scene identification, scene processing, and scene visualization [66]. This work also tried to display the useful navigation information extracted from the various technologies (sensors, laser, infra-red, or GPS) in a way in which there is no distraction to the driver [67].

Numerous works have been done for indoor navigation also. Vision-based techniques are very helpful in which the sensors and the map data have been used to detect the landmarks for the localization process [68]. Images have been used for the marker detection, image sequence matching, and the location recognition purpose. Pre-training of the markers for the navigation is required in this work [69].

The image segmentation method is also helpful for landmark detection. Landmarks such as stop lines, arrows, and traffic signboards are essential in order to accomplish autonomous driving. Therefore, the features of the potential landmarks have been extracted and are then compared with the previously built database, which carries information on segmentation direction, feature descriptors, etc. The major contributor here in this work is pre-built databases [70].

1.6.6 Obstacle Detection

Obstacle detection is the most important task while navigating in a real-time environment. Safety always goes along with the navigation process. In the urban environment, there are lots of hurdles on the way to reach destinations. There are different approaches used in various works to detect the obstacles accurately, but an obstacle detection process is still a challenging task. While navigating the autonomous vehicle on the road, the sensor-based approaches are there to detect the obstacles [71] [72]. There are various techniques used for the detection of obstacles other than sensors such as Shape-based [73], deep learning, Vision-based, template matching, Background subtraction, and many more.

Some of the work has been done on the vehicle pose estimation and shape estimation. They are extracting the information from the 3D scene. Here, the focus is on the object detection as it is also a challenging task [74].

The obstacle detection approaches are also helpful in the video surveillance system. The work has been done in this field also. A vehicle detection algorithm has been proposed for the video surveillance system in [75]. They have fragmented their work into two steps. The first step is to detect the object, and then perform tracking of the object by using optical flow and background subtraction. In this way, the vision-based techniques have been used.

In most of the work, LIDAR sensors are primarily used in the obstacle detection process, but there will still be some dead zone for it. For example, the object with less height than the LIDAR's detection height will be disregarded. Therefore, the camera would be the compliment in this case for the detection of the Obstacle [39].

Today, the latest autonomous cars in the market, like Waymo, are using the IoT Sensors like radar, camera, LIDAR, ultrasonic sensors together for the monitoring and collision warning systems [76][77].

Table 1.1: Summary of AR Autonomous Navigation System

S. No.	Application	Approach
1	Pedestrian Detection	Sensors + Vision
2	Lane Detection	Sensors +Vision based
3	Driver's Behaviour	Vision-based
4	Obstacle Detection	Sensor + Vision
5	Localization	GPS + Sensors + map data
6	Traffic Scene Detection	Vision based

Table 1.1 explains the various application areas of the AR-based navigation system and the approaches used for them. It shows that the sensor-based and vision-based approach is best suited for the Pedestrian Detection, Obstacle Detection and localization process. Vision-based approaches are best suited for Traffic Scene detection. Sensors are necessary for the lane detection process. So, the above table 1.1 depicts the various approaches used in their corresponding applications till now.

In our work, the main emphasis is on the localization and obstacle detection tasks utilizing image processing techniques. In this work, the image processing based categorization mechanism is proposed. This mechanism captures images from the real world and stores these images in the image repository. Images are categorized, and potential markers are identified. It also detects the obstacles ahead to appraise the system to act accordingly.

Organization of the thesis

The thesis is organized in chapters. A brief outline of the chapters is given below.

Chapter 1 discusses about the augmented reality, autonomous navigation, role of the augmented reality based autonomous navigation system and its technological contribution.

Chapter 2 discusses the previous work done in the field of the localization process, as well as the obstacle detection process. Along with it, the technological classification has also been presented.

Chapter 3 classifies the research gaps and defines the objectives with their scope.

Chapter 4 proposes an effective and efficient image categorization mechanism for the marker detection in addition with bookmark identification.

Chapter 5 proposes an approach to build a new image repository for the augmented reality based navigation.

Chapter 6 proposes an algorithm for the stationary obstacle detection.

Chapter 7 concludes the proposed work with future directions which have also been discussed.

CHAPTER – 2

REVIEW OF LITERATURE

2.1 INTRODUCTION

Various technologies that are contributing to the autonomous navigation systems are image processing, sensors, satellite data, visual odometry, and augmented reality. Augmented reality navigation systems superimpose the virtual navigation information on the real-time environment [45][78]. There are various applications of these systems, such as vehicle navigation, pedestrian detection, lane detection, obstacle detection, and many more.

The primary AR models created by computer graphics pioneer Ivan Sutherland and his scholars at Harvard University and the University of Utah showed up in the 1960s in which they used a see-through to present 3D illustrations [13]. Researchers at U.S. Air Force’s Armstrong Laboratory, the NASA Ames Research Centre, the Massachusetts Institute of Technology, and the University of North Carolina at Chapel Hill continued this research during the 1970s and 1980s. Sony Walkman (1979), digital watches, and personal digital organizers were introduced in this era. In the 1990s, as PCs turned out to be little enough to be worn consistently, wearable computing [8, 9] emerged, and later, the HUD's started getting utilized for this purpose [79][80][81]. Today, According to the [43], Wayray is the only developer in the world that is integrating augmented reality into the cars and this company also builds HUD.

2.2 EXISTING WORK DONE IN LOCALIZATION PROCESS

The autonomous driving system comprises tasks such as scene understanding, tracking, object detection, landmark detection, and many more. Landmarks can act as external reference points that can be easily observed in navigation. Humans often use landmarks for navigation. Anything that is easily noticeable and worthwhile for the navigation process can be considered as a landmark. For example, hoardings, buildings, road signs, etc. are some of the potential landmarks.

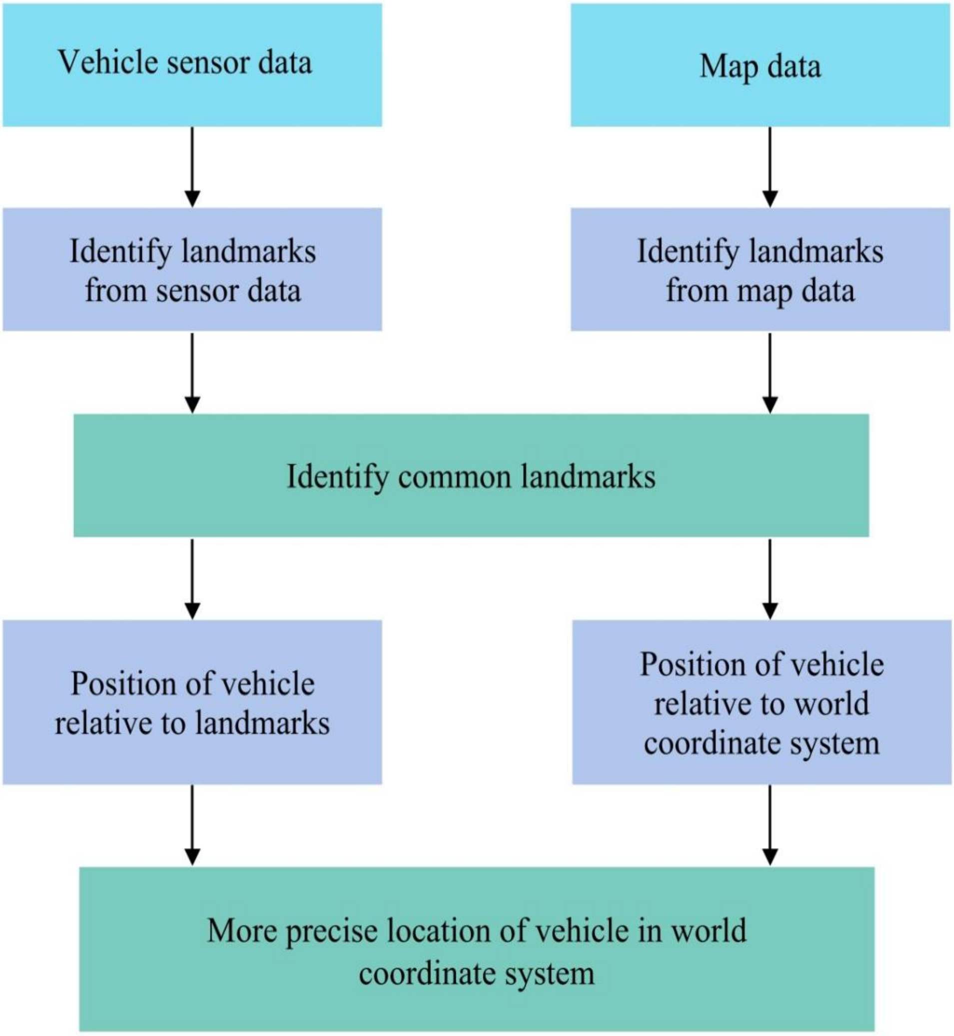

FIGURE 2.1: *Determination of vehicle location in world coordinate system*

The number of motor vehicles in the world has escalated to over one billion, which has grown almost ten times since the 1960s [82]. Taking into consideration, the growing need of swift transportation system among citizens, even more automobiles are likely to be available in the future. As technology has evolved a lot in the software field, it is now possible to use the calculating power of computers to enhance the driving experience and more effectively manage traffic flow. Information technology has simplified several occupations in the last 50 years, such as word processing, statistical calculation, and data retrieval. Nonetheless, it has been a dream to do so autonomously with a machine, because driving a car is much more complicated than just these simple tasks. Researchers have begun significant work in the advancement of similar technologies to make this dream a reality.

Numerous researches have been carried out in relation to autonomous driving techniques, and several systems have been developed for the purpose of using sensors like GPS, cameras, IMU, maps, etc. for navigation of vehicles in unknown areas [83]. In the present work, the use of a camera has been taken in place of GPS and other sensors. For this purpose, undoubtedly, there are already map-based and map-less systems for navigation. In the case of map-based systems, we have to build the database of the markers, for navigation to be performed. In the latter case, i.e., maples navigation, there is no need for a pre-built database for navigation purposes. The sum time database is made while navigation and matching is done so as to navigate safely. In map less navigations, we have two approaches: Appearance-based navigation and optical flow [84]. Optical flow takes note of the motion of the object or the features adjoining to the sequence of the images. In the proposed mechanism, landmarks are used to confirm the path for navigating correctly and safely [85].

Landmarks are marked as movable and non-movable. Movable marks are the ones that can change its position according to seasonal conditions, but non-movable marks remain for a long time. This research work is based on the image processing categorization mechanism related to identified landmarks, which are immobile, and those prove helpful for navigation of vehicles in unknown territories. There has recently been a considerable amount of research reported in the navigation identification and categorization systems for autonomous vehicle area. This study proposes advanced navigation marker identification and categorization systems. A review of related literature was undertaken with the primary focus on the image categorization system, defining the research variables as well as the conceptual-listed relationships between them. The navigation process can be performed using various techniques like sensor-based navigation, IMU-based, RFID-based, GPS-based and vision-based approaches. The vision-based approaches for autonomous navigation purposes are trending these days [86], however a lot of work is yet to be done in this field. Vision-based tracking has its variety of application in video surveillance, video monitoring as well as autonomous navigation.

Given below are some of the researchers who put efforts in this area of work. Starting with E. Rose *et al.* [87] presented a system that used augmented reality for annotating real-world objects. They performed a matching of real objects with the virtual model of the real-world objects so as to annotate them with information from the corresponding model visually. Any user related inquiry about the real-world object is translated into a query on the model and feedback is produced that can augment the user's view of the real model. Different techniques of camera calibration, object calibration, and tracking are used in many AR applications.

R.T. Azuma [13], presented the role of augmented reality in various applications. Here, 3D objects are integrated into a real-time environment with the help of AR technology. But, there are two main challenges that come into existence with AR technology: Registration and Sensing.

U. Neumann *et al.* [88], introduced the architecture for the detection and tracking in unprepared environments that uses natural features for AR applications. This architecture comprises feature selection, motion estimation, and evaluation in a closed-loop for robust 2D tracking. The biggest barrier in building this system is a less accurate long-range sensor

T. Hollerer *et al.* [89], in their work, developed an interface for indoor and outdoor users to access and manage the information registered with the real world. In this, indoor users can communicate with and provide information to the outdoor users to guide them.

W.Narzt *et al.* [10] worked for the enhancement of traffic safety and introduced a system for the visualization of the information.

G. Reitmayr *et al.* [90] has made a system for urban environment that uses AR for navigation and information browsing tasks to guide the user to reach his destination and present information of surroundings respectively.

D. Robertson *et al.* [91] developed a prototype for navigation in an urban environment using matching of images with a database consisting of the front view of the buildings.

D. Bradely *et al.* [92] presented a virtual-navigation system using panorama images in which the dynamic view of the image is taken into account and also flattened the need for full 3D modeling.

W. Narzt *et al.* [93] in their work they presented an innovative visualization model for the AR-based navigation system in which the car is used as an AR apparatus. The virtual information is superimposed on the real-world scene with the help of the car's windshield.

I.C. Seyong *et al.* [94] has proposed a recognition algorithm which uses edge and block-based approach to detect the POI for various applications by removing the background objects with the help of edge segments of small-sized blocks within an image frame and the input here is the camera's captured video.

P. Jun e*t al.* [95] has developed an AR-based guidance system in which the users are instructed in the museum where location and the multimedia information is provided to the users as per their interests.

K. Jongbae *et al.* [96] originated a system in which a combination of marker, color information, and prior knowledge is used to identify the specific location. For this work, an adaptive thresholding method and location model are used to detect markers to increase performance and image sequence matching to reduce the execution time, respectively. Here, the printed papers are used as markers.

W. Wu *et al.* [64] has introduced a prototype-based system in which landmarks are used as external reference points to provide navigation information as the landmarks are easily identifiable.

A. Mulloni *et al.* [97], introduced the indoor navigation system in which fiducial markers are detected by camera phones to determine the user location.

A.B. Rad *et al.* [98] gave detailed information about a system for navigation with the help of fiducial markers. In this, navigational instructions and location information is coupled with each AR marker to recognize a location by using the pre-constructed image database and location model. The main objective of this work is location positioning [99][100].

P. Barrie *et al.* [101] has established a feature-based indoor navigation system in which fiducial markers are used, which, when detected by the algorithm, generate the route information and gives the instructions for taking two steps ahead only. In this work, the main drawback is frequent scans, which consumes a lot of time.

Z. Toth *et al.* [102], in their work, used a database of digital images to identify obstacles, non- uniform road, and crossed roundabouts. Their approach provides easy and efficient navigation instructions. They have used some software and network topologies for this work.

A. Taneja *et al.* [103] used images captured by the mobile along with images from google street view that is used for route determination for a moving vehicle. In this, by comparing images from both sources, continuous localization is performed.

Kemsaram et al. [104] discovered an algorithm for image tracking, which has a key role in autonomous surveillance and monitoring. He developed a hybrid autonomous visual tracking algorithm that relies on cam-shift and Extended Kalman Filter Estimator, which is based on micro-aerial vehicles. This algorithm recognizes and tracks the moving target continuously. The simulation is done on the MATLAB tool for identifying the performance of the proposed algorithm. The result demonstrates that the proposed algorithm is very efficient and precise for tracking ground moving targets.

Reiser et al. [105] presented their work with the sensor-based approach. They, during the first run without any prior knowledge, made their system to navigate with the help of sensor-based nodes and then geo-referenced RSSI data has been used for mapping process.

Ort et al. [106] proposed an algorithm in which the localization process of the automated vehicle has been achieved using the sensor technology. Sensor view, local perception system, vehicle odometry are collectively worked for the navigation process effectively. The performance results have been evaluated in the rural environment and found to be satisfactory.

Passot et al. [107] have discussed the various methods for the navigation process. In the first run, the system detects the initial position and builds the navigable route. Then, during the next runs, the system again detects the position and senses the navigable environment and builds the navigable map. The robot can also detect the errors associated with the created map.

The below table 3.1 is showing the different approaches used for the autonomous navigation for the on-road vehicles. This reviews how the technology has been improved year-wise using various combinations of sensors. Several experiments have been made to make the vehicle navigate cautiously on the road in the real-time scenario.

Table 2.1 represents the literature survey in which the various author's work has been characterized on the basis of certain parameters.

TABLE 2.1: Literature Review based on technologies used for localization process

Authors	**Year**	**Approach used**	**Sensor Used**	**Algorithm**
Passot et.al. [1]	2019	The system builds the navigational route by detecting the initial position and by sensing the environment during its runs.	Camera / LIDAR and sonar sensor.	Segmentation, edge detection or shape recognition techniques.
Paolillo et.al. [108]	2018	For automated navigation, they have used a camera, LIDAR, IMU, and visual odometry.	Camera and IMU	Sensor-based reactive framework.
Jian et. al. [41]	2018	Landmark is detected by using sensor data and map data. The position of the vehicle relative to the landmark is determined.	LIDAR and Camera	NA
Huang et.al. [109]	2017	They provided an overview of various approaches for object detection and tracking for developing a navigation system. The work is basically done for maritime navigation. Methods of	Camera, CMOS sensor and a fish-eye lens.	Histogram of oriented gradients and linear SVM.

Contd. …

Authors	Year	Approach used	Sensor Used	Algorithm
		computer vision research have also been highlighted, and their performance is also evaluated		
Fernández et.al. [110]	2016	Performance of several methods in the localization of a mobile vehicle. Local feature descriptors are used like SURF-Harris.	177 full spherical panorama images	SURF descriptor and HOG.
Kaneko et.al. [111]	2016	This method is based on image characterization by segmentation point, road landmarks such as stop lines, arrows, traffic signboards are essential for recognition in order to accomplish autonomous driving. Furthermore, already defined features of landmarks have been extracted, and it is then compared with the previously built database. The major gap is pre-built databases.	2D camera, 30 frames per second, fixed with ROI in front of the vehicle	Segmentation point extraction, canny edge detector.
Bagyaveeresawaran et.al. [112]	2016	The algorithm uses the monocular images from the lander camera as input and produces position and motion estimation. The algorithm used for image matching is the Scale-Invariant Feature Transform (SIFT) algorithm, which uses the moon craters for feature extraction and feature matching.	KITTI dataset is used. IMU, sensors (LIDAR)	SIFT algorithm.
Lima et.al. [113]	2016	Dynamic objects are detected. The driver's behavior is detected. Visual servoing and image-based dynamic window approach is used.	Laser sensor with 180º coverage.	Visual servoing and dynamic window approach.

Contd. …

Authors	Year	Approach used	Sensor Used	Algorithm
X. Wei et.al. [114]	2016	The motion approach is used for scene detection for autonomous drone navigation. Recognition of landmark with the histogram of oriented gradients features and linear SVM methods on each frame of the video stream.	NA	Linear-SVM, RBF-SVM, and HIK-SVM.
Kemsaram et.al. [104]	2016	An algorithm for image tracking for autonomous surveillance and monitoring. A hybrid autonomous visual tracking algorithm using cam-shift and an Extended Kalman Filter Estimator, which is based on micro-aerial vehicles. This algorithm recognizes and tracks the moving target continuously.	Camera	Visual tracking algorithm based on Extended Kalman Filter and Cam Shift approach
Litman et.al. [115]	2014	Merit and demerits of autonomous vehicle is discussed and its cost. Geometric contextual reasoning used for recognition of obstacles or any object. Assumption: The location of the object should be lying on the supporting surface. Novel relation among object location and pose in the image has been presented for solving the optimization problem. The proposed method reduces the false detection rate and false alarm.	NA	NA
Bonin et.al. [116]	2008	A survey for map based and maples navigation. Appearance-based localization, Optical Flow, Ground detection is used.	NA	Visual navigation techniques.

Contd. …

Authors	Year	Approach used	Sensor Used	Algorithm
Y.-H. Lu et.al. [117]	2004	Proposed some problems and challenges in image-based navigation awareness. Two systems have been developed as testbeds. First, it comprises mobility, localization awareness, image processing, etc. in the second; some images are used in the database to help in-vehicle navigation, which creates some issues. The solution to these issues is to use GPS to compare and search the location in the database	Location-aware image database and Image enhanced navigation	Short-range visual lane marking detector and a dead reckoning system.
DeSouza et.al. [118]	2002	Optical flow and appearance-based methods have been used in unstructured environments for navigation purpose	NA	Optical flow and Appearance-based methods.
Giachetti et.al. [119]	1998	Optical flow can successfully be used by a vision system for assisting a driver in a vehicle moving in the usual streets.	Television (TV) camera mounted on a car	Correlation-based and correction of the optical flow technique.
A. Ohya et. al. [120]	1998	A vision-based navigation method for mobile robot navigation is presented. This method is specifically used to avoid obstacles. Stationary obstacles are evaded with a single camera, and moving obstacles are recognized by ultrasonic sensors	Single camera and ultrasonic sensors	Self-localization and retroactive position correction system.

Therefore, there are a number of image processing techniques used for the localization of the vehicle. Here, SURF, SIFT, Canny edge detector, histogram gradients, template matching, background subtraction, are a few exemplars of the above-used techniques.

Now, the various sensors are also used for localization purposes. Odometry, vision-based sensors are used for the tracking process. The below table represents the sensor-based review of the localization process in the tabular form.

TABLE 2.2: *Literature Review based on sensors used for localization process*

Ref No.	Strategy	GPS	Sensor	Compass	Gyroscope	IMU	RFID	Dataset	Camera
[121]	The speed of the vehicle and distance travelled is calculated with the optical flow approach.	×	×	×	×	×	×	×	√
[122]	Optical flow and background subtraction are used for the traffic surveillance system.	×	×	×	×	×	×	×	√
[123]	The optical flow approach is helpful in the fast tracking method.	×	×	×	×	×	×	×	√
[124]	Cell phones are used for indoor navigation using GPS.	√	×	×	×	×	×	×	√
[125]	Crosswalks on the road are recognized to solve the registration problem.	×	×	×	×	×	×	×	√
[126]	A hybrid approach is used for tracking in un-prepared environments.	√	×	√	×	√	×	×	√
[127]	Corner detection, along with the color feature extraction, is used for tracking.	×	×	×	×	×	×	×	√
[128]	Geo-tagged photos are used for navigation purposes.	√	×	×	×	×	×	√	√
[129]	Online creation of a map of natural landmarks	×	×	×	√	×	×	×	√
[130]	Fiducial markers are used for indoor navigation using computer vision.	×	×	×	×	×	×	×	√

Contd. …

Ref No.	Strategy	GPS	Sensor	Compass	Gyroscope	IMU	RFID	Dataset	Camera
[131]	Robot's odometry is used for its localization	×	√	√	√	×	×	×	√
[132]	Images from the web are used for localization from mobile devices.	√	×	×	×	×	×	√	√
[133]	A robust localization system is presented for a complex multilevel indoor environment.	×	√	×	√	√	×	√	√
[134]	Interior features are used for determining user's location in indoor navigation	×	×	×	×	×	×	√	√
[135]	The probabilistic localization method is used for complex indoor environments.	×	√	×	×	√	×	×	√

GPS [136][137], sensors, compass, gyroscopes [138][139][140], IMU are used for locating the vehicle in a real-world environment. There are various sensors available in the market to locate the current position of the vehicle [141][142][143][144]. Navigation is of different types, i.e., Vision-based navigation, Inertial based navigation, GPS based navigation. Some of these are implemented in indoor environments, whereas other are for outdoor environments. [145][146][147][148] [149].

2.3 EXISTING WORK DONE IN OBSTACLE DETECTION PROCESS

Lots of research has been done for the detection of an obstacle in the real-time environment, but this is still a challenging task. For the detection of obstacles, various approaches have been used till now. Vision-based and sensors based are few of them. Mostly, for the detection of real-time obstacles, sensor-based approach is preferred.

Therefore, an obstacle detection system has been proposed by Prasad et al. [150], which took the image as an input and gave complete real-time information. In this, image is processed and compared with images in the database using homography. In case matching is successful, the information correlated to that image will be obtained and sent using the MQTT protocol. This way, sensors, along with images, are used for the detection purpose.

Khandelwal et al. [151] have presented a marker-less approach for the tracking and detection of the obstacles present in the real-world environment in which at first, the input is taken from the camera, and then, pre-processing of the image is done. Further interesting regions are detected and labeled as objects. After the segmentation of the detected objects, the virtual information is superimposed on the real-world environment. Histograms of oriented gradients are used as a feature detector. The frame rate is 24 fps, and the resolution of the image is 640*480 pixels. So this work is done for the marker-less augmented reality applications.

Kim et al. [152] have presented an approach in which there is no usage of specific markers for the tracking process. Image processing based techniques are used. For feature extraction, the Gaussian pyramid is used. Here, a fast corner detection algorithm is used to acquire coordinates of feature points. For tracking of the camera, feature points, edge size, and the direction information as descriptors are used. In this way, the generated information is overlaid on the real-world scene [153][154].

According to Borse et al. [155] work, sensors and image processing based techniques are used to track the moving objects. A camera and a range sensor are used for detection purposes. This system works in five stages. In the beginning, the image is captured from the real-time video. Objects are detected from a particular frame. Out of those objects, obstacles are detected. A decision is taken to avoid those obstacles which leads to the navigation phase. Infrared sensors are used to detect the obstacles.

Rodríguez et al. [156] developed a real-time method for the detection of moving objects. Optical flow is used for the detection of moving pixels in the consecutive frames. A monocular camera has played a major role in this tracking and detection process for the unmanned aerial vehicle.

Thakoor et al. [157] presented an approach for the detection of the objects using the moving camera. This detection process works in three consecutive frames: backward frame, the frame of interest, and the forward frame. Optical flow and background estimation is calculated for these three frames, and then the difference is calculated. Any change in this result leads to get the shape of the moving object.

Wekel et al. [158] have presented a system for the detection of moving objects without the need for sensors. The algorithm used in this system is independent of the geometry. Stereo-matching and ground-motion estimation techniques are applied for the classification of the segmented images. The map is generated from the top view of the robot's field of view. This whole system works with the help of a camera. Harris corner detection, along with the Lucas-Kanade, is applied for feature detection and tracking progress, respectively. RANSAC algorithm is also used for detection purposes.

Lan et al. [159] presented an obstacle detection method for a real-time traffic surveillance system. This system works for the unwanted objects visible on the road like illegally parked vehicles, accident vehicles, and rejected objects. This is a vision-based approach using a Gaussian mixture model and selective updating of GMM. Here, static obstacles are detected using the relative object speed [160].

Barandiaran et al. [161] carried out a survey for the detection of interest points using computer vision-based techniques. In this paper, the focus is on geometric transformations. According to this survey, there are various point detectors available like, HARRIS, GFTT (Good Feature To Track), SIFT (Scale Invariant Feature Transformation), SURF (Speed Up Robust Feature), FAST, MSER (Maximally Stable External Regions, STAR, ORB (Oriented BRIEF). The choice of feature detector is very important in the detection process. As per the evaluated results, ORB is best in terms of accuracy and efficiency.

Kalkofen et al. [162] described the various aspects of the object detection for a real-world environment. Features of the object play a vital role in the detection process. There are various features from which important information can be extracted. Those features are edge features, salient features, area-based features, etc. In this paper, various techniques are discussed for the detection of obstacles in AR visualization. The sensor-based approach can be used; vision-based approaches like background subtraction, optical see-through approach are some of the examples for techniques used in this paper.

Arvind et al. [163] commenced an approach for the detection of dynamic obstacles present in the run-time environment. The reinforcement learning technique is used to train the system for detection purposes. MLP-SARSA is a reinforcement learning policy for sensing the environment and its extractions [164].

Kakoty and Guzzi et al. [165] revealed an approach for the navigation of the system in an unknown dynamic environment. Sonar sensors, along with other sensors, are used to build the map for navigation. The system moves safely on the road by avoiding the static obstacles present in the scene. This approach is inspired by human pedestrian behaviour [166].

Hua et al. [167] have proposed the neural network approach for the detection of real-time obstacles. They stated that the remote sensing and memory reasoning are the key components of the detection process. Also, deep learning is playing a major role in various works done for the autonomous navigation process. It acquires the object from the environment, followed by reducing the complexity of the same, and then it extracts features of that particular object [168][169][170].

Yu et al. [171] presented obstacle detection and tracking method for an urban environment. Object segmentation, followed by the Kalman filter, is applied to take an estimation of parameters of the obstacle detected for navigation in the dense traffic urban environment.

Heras et al. [172] proposed vision-based obstacle detection for autonomous navigation. Background subtraction technique is used for differentiating the moving and static obstacles from the consecutive frames. In this approach, a camera is used for detection purposes.

Chan et al. [173] proposed an image processing method for navigation purposes. In this, two points are selected from the inputs given by two cameras. Five sets of feature points are made and processed to get the distorted homography of the two images that lead to the results.

The above literature shows that there are various techniques used for the detection of obstacles from the real-time environment. The obstacles can be stationary, as well as moving [174][175]. Vision [176] and sensor-based approaches [177] have been used to detect and track the objects in the scene. In vision-based, the detection has been achieved with the help of a single camera, however sometimes, two cameras have also been used. Geometric features [178] are also helpful for detection purposes. Deep learning, neural networks, machine learning are the trending approaches used today for the obstacle detection process [179][180][181][182]183]. The literature below in table 3.3 is presenting the technological contribution in terms of different approaches used until now in tabular form.

TABLE 2.3: *Literature Review based on the technological work done on obstacle detection*

Author	**Year**	**Approach Used**	**Platform Used**	**Hardware Used**
Arvind et al.[163]	2019	Reinforcement learning technique is used to train the system.	NA	Ultrasonic Radar
Hasan et al. [185]	2018	The shortest path planning algorithm is used.	MATLAB	NA
Khalilullah et al. [186]	2018	Single camera and deep learning approach.	Wheel Chair robot.	a PC, a CCD DFK21F04 camera and two AC motors. The PC is equipped with 2.67 GHz Intel Core i5 CPU and 4.00 GB RAM
Song et al. [187]	2018	Techniques like stereo matching, obstacles segmentation, Hough transform, deep learning, and vision-based algorithms are fused together	NA	Intel Core i7 processor (3.1GHz, 4 cores), NVIDIA GeForce Graphics Card (GTX 650) and 4GB RAM.Bumblebee-xb3.
Song et al. [188]	2018	UV-disparity obstacle detection algorithm to get dynamic and relative dynamic objects	Hongqi car, IN2BOT, I ROBOT.	Intel Core i5 processor (2.4GHz), 4GB RAM. Bumblebee-xb3 for stereo images with and a Delphi ESR mmw-radar.
Tsai et al. [189]	2018	Dense optical flow method, SVM, and (SURF).	MATLAB	NA
Zhang et al. [190]	2018	Disparity maps, vanishing points, and weighted graphs are used.	KITTI dataset	Stereo Camera
Gupta et al. [191]	2017	Object detection and object classification approaches.	Open CV Python	NA
Kaneko et al. [192]	2017	Inverse perspective mapping and image abstraction and geodesic distance computation	iRobot Create2 robot, monocular front web camera.	Monocular web camera and the height is 34.5mm above the ground is used.

Contd. …

Author	Year	Approach Used	Platform Used	Hardware Used
Lee et al. [193]	2016	Sensor-based detection algorithm.	Forward-viewing camera, wheel encoders, gyroscope.	
Anvaripour et al. [194]	2015	Shape-based pre-trained histogram generic model	NA	NA
Yu et al. [195]	2014	Discriminative shape-based model and geometric parameters are used.	UIUC Car and Weizmann–Shotton horses	NA
Ess, A. et al. [196]	2010	Self-localization, obstacle detection, and object category recognition		Forward-looking AVT Marlin F033C cameras.
Bonin et al. [116]	2008	Visual navigation techniques	NA	NA
Murphy et al. [197]	2006	Geometric parameters	NA	NA
Nistér et al. [198]	2006	Estimation of real-time ego-motion of a single camera.	PerceptOR.	Synchronized analog cameras with a horizontal field of view of 50◦, and image fields of 720 × 240 resolution
Manduchi et al. [199]	2005	Sensor processing algorithm	The code ran at 1.5 Hz to 0.5 Hz on 320×240 images, on a 900 Mhz Pentium III PC.	A color stereo camera, and a single-axis ladar.
Ohya et al. [200]	1998	Single camera vision and ultrasonic sensors are used to detect stationary and moving obstacles.	YAMABICO robot.	YAMABICO robot.

In the above table, the work of the various authors for the detection of the obstacles has been presented chronologically. The various approaches used and their hardware and software platforms are also discussed. The key techniques of their work have also been described. From the previous works, we found that very little work has been done in the vision-based obstacle detection. A lot of work is yet to be done in this area. In this work, we have proposed the visual-based stationary obstacle detection algorithm using the single 12 MP camera with the resolution of the 4000x3000 pixels whose frame rate is 30 fps.

CHAPTER – 3

OBJECTIVES

Based on the literature review discussed in the previous chapter, following gaps are identified from the existing works:

3.1 RESEARCH GAPS

- Current navigation systems mainly present procedural/paced navigation information to enable a driver to locate a turn.
- Also, the existing navigation systems have not reached their full potential. Two main kinds of navigation interfaces in existing systems are voice commands and the abstract map images on display. Both are useful but not the best.
- Existing navigation systems can increase the driver's cognitive load because the driver has to map the information provided by the navigational system to the real environment outside the windshield.
- In order to perceive the navigation information, the driver has to move his/her attention away from the road.

Following are the objectives framed on the basis of the above mentioned research gaps:

3.2 OBJECTIVES

- To identify and propose an efficient and effective image categorization mechanism.
- To build a new image repository for Augmented Reality based navigation.
- To utilize images for Obstacle detection and bookmark identification.

3.3 SCOPE OF THE PROPOSED WORK

In this era, everyone in the society requires things that can work for them and make their life better and comfortable. The automotive industry is working in that direction rigorously. In the present research, the best image processing mechanism will be formulated for the AR-based navigation of the autonomous vehicle so that the automatic driving along with the safe driving comes into play. This research will be beneficial for old age people, physically handicapped, and non- drivers. They will not be required to drive themselves but provide them with the source and destination locations. Throughout the world, the researchers are primarily focussing on autonomous vehicles that are capable of moving on their own. This is not just merely to support the needy set of people, including old age and physically challenged but largely to reduce the menace and fatality created by erroneous human driving.

3.4 PROPOSED METHODOLOGY

This section explains the proposed methodology for the implementation of the work. The bookmarks are detected and identified, along with the obstacle detection process. Here, the work has been done for the detection of stationary objects.

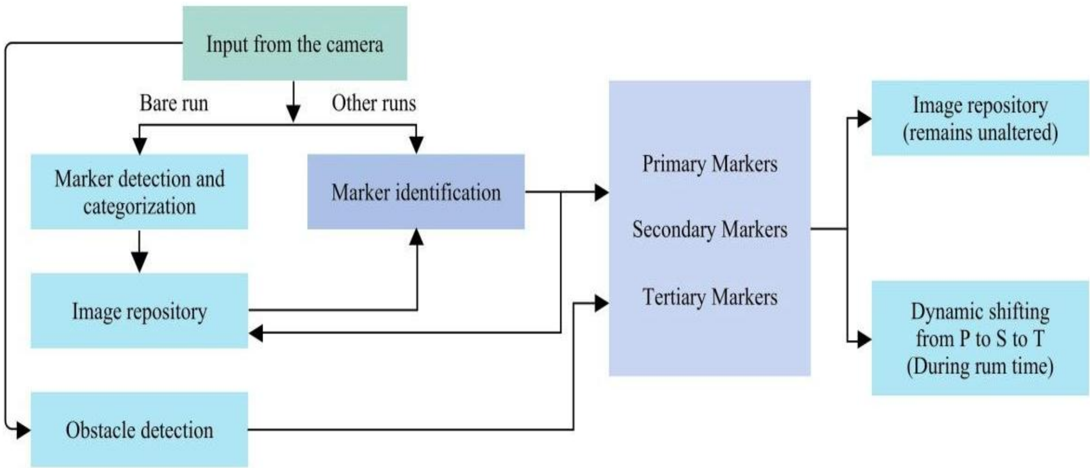

FIGURE 3.1: *Proposed Methodology*

3.4.1 *Steps for the proposed methodology*

Step 1: Input to the image processing categorization module is given by camera.

Step 2: In the first run, which is the bare run, the system performs two steps: capture and store the images into the repository. We assume that during this run, the system will have a zero level of understanding of the path.

Step 3: In the next run, the images are categorized based on certain factors like shape, texture, etc. and potential marker images are identified.

Step 4: The next run would be the supportive run that uses the stored images for navigation.

Step 5: After the identification of potential markers and obstacles, output would be sent to the core algorithm part for further processing and decision making.

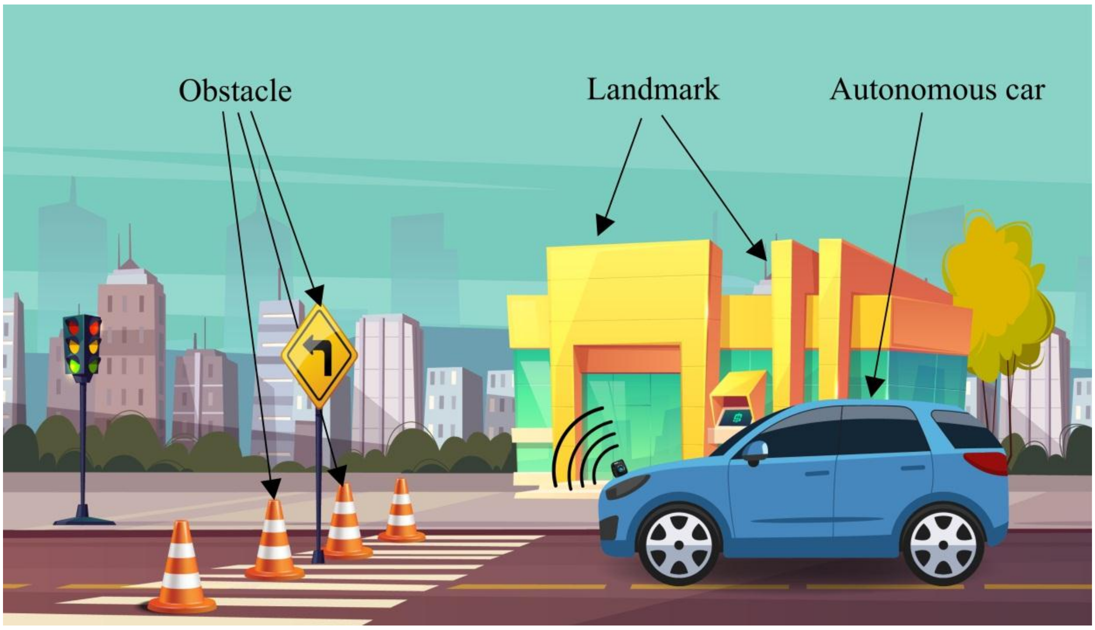

FIGURE 3.2: *Real world scene of the AR autonomous vehicle*

In the above figure 3.2, stationary landmarks and the real time stationary obstacles are highlighted in the environment. The buildings are generally detected as the markers and the objects present on the road are considered as the obstacles.

3.5 PRESUMPTIONS

Following are the presumptions taken in the proposed work:

- The system is moving with a constant speed of 20km/hr.
- Monocular camera is used for the input in the real-time environment.
- Camera is mounted on the bonnet of the car.

CHAPTER – 4

IDENTIFICATION AND PROPOSING AN EFFICIENT AND EFFECTIVE IMAGE CATEGORIZATION MECHANISM

4.1 INTRODUCTION

Automated vehicle is a vehicle that can navigate from source to destination on its own without human intervention. They are generally built on the idea to mimic the human way of navigation from one place to another. Just like humans use landmarks as reference points to move from one place to another, similarly, autonomous vehicles require that too. A landmark is any object that can be easily seen and recognized from a distance. It should be something that is fixed and cannot be replaced by any other object. Landmark can also be named as marker. The choice of marker is an essential step while navigation. Here, in this objective, run time videos are captured, and image processing techniques are applied to those videos to detect the markers by removing the different levels of triviality.

Landmarks are marked as movable and non-movable markers. Movable markers are the ones that can change their position but non-movable markers remains immobile for a long time. This research work is based on the image processing categorization mechanism related to identify landmarks, which are non-movable, and those prove helpful for navigation of vehicles in unknown territories.

4.2 METHODOLOGY AND IMPLEMENTATION

This section deals with the detection, categorization and identification of immovable markers. Advanced Navigation Marker Identification and Categorization System (ANMIC) detects three types of markers by removing the step by step triviality from the real-world scene even for an unknown terrain. During the process, this system catches the identifiable markers based on the categorization mechanism. Such images of the markers are stored in the video for use in the actual run. The system thus matches the features of the markers, and therefore the navigation takes place safely.

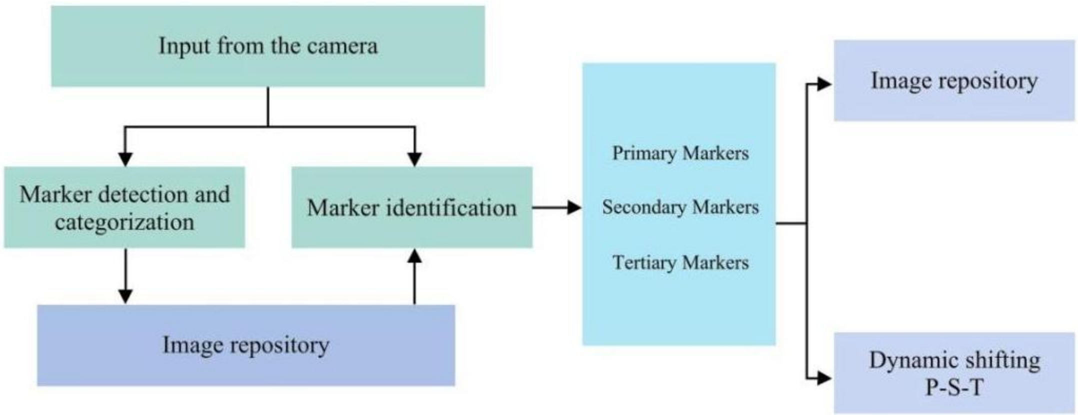

FIGURE 4.1: *ANMIC Methodology*

Figure 4.1 shows the methodology of the proposed algorithm (ANMIC) used for the marker detection and identification.

While capturing the images in the video during bare run, frames are generated on which optical flow techniques are recorded. Optical flow helps to differentiate between movable and non-movable objects as presented in Figure 4.2. In this process, the movable objects are set aside, and non-movable objects take up a distinct place. Thus the trivializations are removed to make the system more efficient with the quality markers.

The non-movable markers are further classified into a selection bases in which the region of interest (ROI) are selected based on certain parameters. Some threshold conditions are applied on these selected regions to get the potential markers from the frames. Then, these potential markers are further categorized into three types of markers; primary, secondary, and tertiary. This categorization is done on the basis of prominent factors such as movability, size, area and visibility conditions.

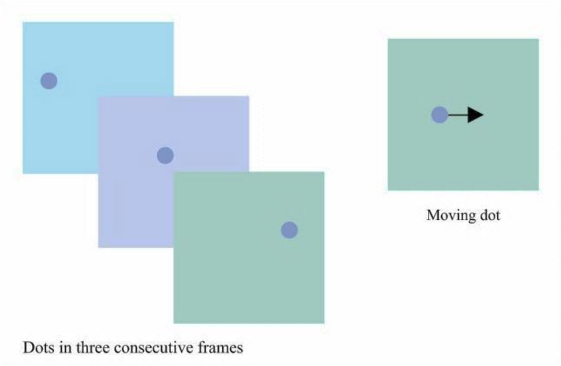

FIGURE 4.2: *Optical flow Concept*

4.2.1 Categorization Process

This section discusses the process of categorization of the markers in the ANMIC. First, the video is made of a specific path (A to B); so that the frames are generated from the video; second, the movable and non-movable objects are differentiated and split-up with the help of the optical flow approach. Finally, the frames with non-movable objects are converted into black and white frames to work with some similarities of the region for proper identification. The processing of peculiar points to the area is identified to confirm them as markers.

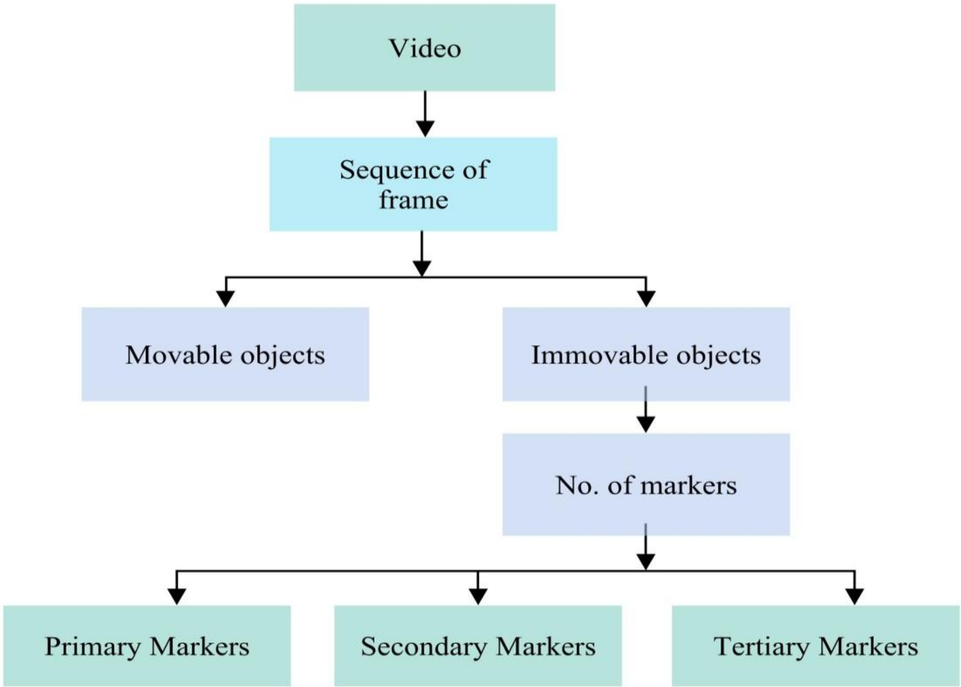

FIGURE 4.3: *Categorization of markers in ANMIC*

Figure 4.3 explains the flow chart followed for the categorization of the markers. The flow chart shows the step by step process followed to identify non-trivial markers. The steps pursued for the categorization are as follows:

1) Objects are categorized into movable and immovable objects.
2) Out of all, objects which are immovable and are satisfying the certain thresholding conditions will be considered as non-trivial markers.
3) Finally, the non-trivial markers are further categorized into primary, secondary, and tertiary markers.

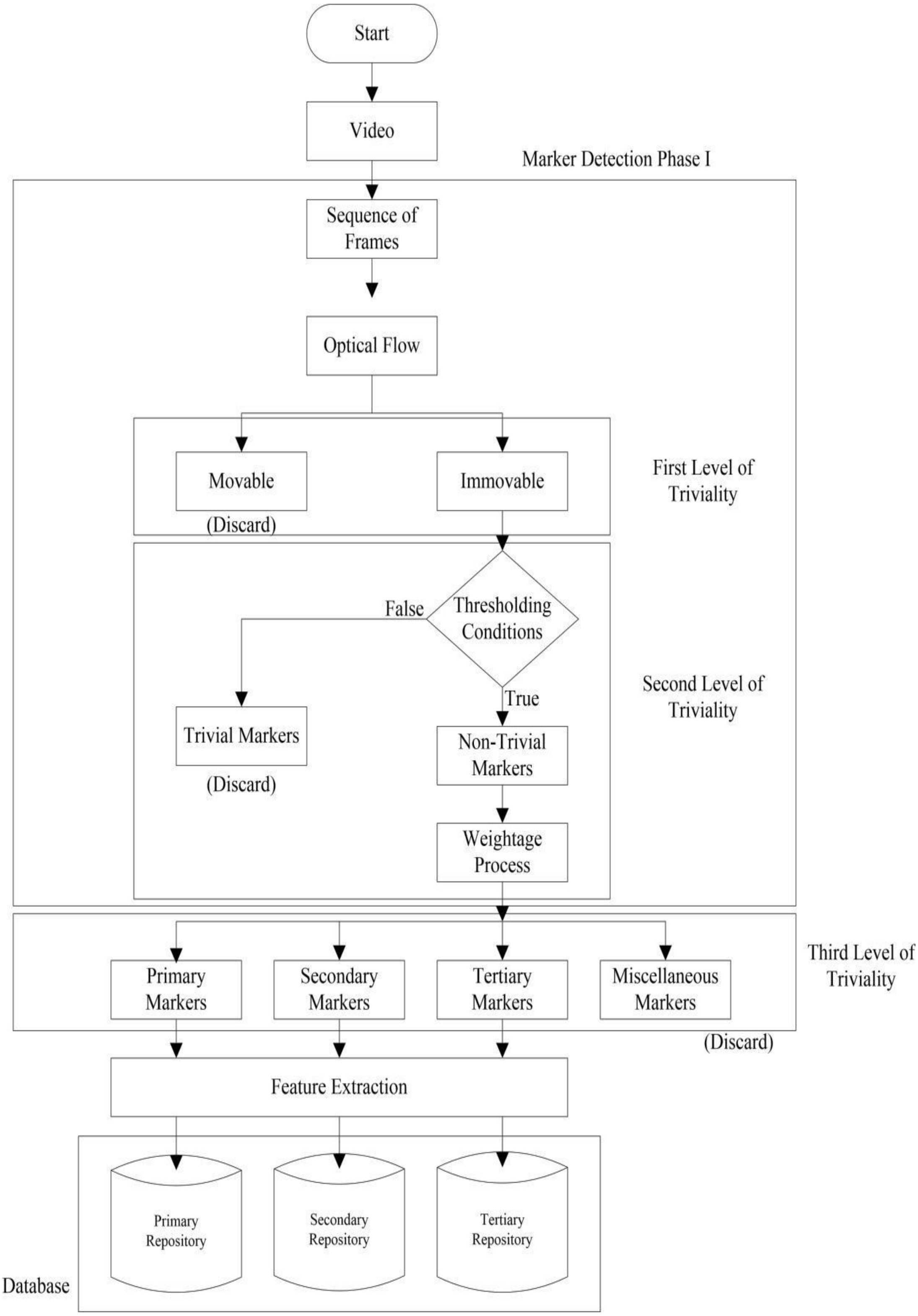

FIGURE 4.4: *Algorithm for identification, and categorisation*

To confirm the identification of the markers, upper and lower limits of the objects are perceived. Lower limits are set to discard the trivial objects to make the markers more precise. Ground and sky can be taken as higher values and can be discarded easily.

Figure 4.4 explains the step by step removal of the triviality for the marker detection process. During the bare run, from capturing the video to the detection of markers includes:

- Categorization of the markers
- Feature extraction of the markers
- Storing the features of the markers

Categorization of the markers leads to the detection of markers from the video, based on the prominence. On its bare run from a path A to B, video is captured, which leads to generation of frames at some specific intervals. From these frames, the moving objects and non-moving objects are differentiated first as a useful marker which is always immovable, with the following characteristics:

a) It should not change its position for a long time.
b) It should be easily recognizable from the long distance.
c) It should be more high spotted than the nearby objects.

So, the objects that fulfill the above set of conditions can be termed as non-trivial markers. The non-trivial markers are then prioritized as the primary, secondary, and tertiary. The features of the markers are extracted and stored in their respective repositories for future use in the navigation process.

This can be explained with the help of the following equations

$$\mathrm{L} < \mathrm{A} < \mathrm{M1} \qquad \dots (4.1)$$

$$\mathrm{L} < \mathrm{A} < \mathrm{M2} \qquad \dots (4.2)$$

Where

‘L’ is the lower limit (Variable).

‘A’ be the area value of objects/regions.

M1 (upper limit) is the maximum area value that is obtained by using max() function limit which is the next maximum value after M1.

The algorithm for the categorization of the markers from the runtime video is also explained below:

4.2.1.1 Algorithm for Marker Detection and Categorization and their storage process in the repository.

Let, L be the black and white image

ar, cr, g be the variables for the area, centroid, and count of the markers detected.

L1, A1, A2 be the lower limit, upper limit 1, upper limit 2 of the threshold limits of the area values.

Step 1: Start

Step 2: Initialize ar= 0; cr= 0;g= 0;

Step 3: Set cr= regionprops(L, ‘Centroid’), ar= regionprops(L, ‘Area’);

Step 4: Set k=length of cr elements

Step 5: For w=1 to k

Define the matrix with ar (area) and cr (co-ordinates);

End for

Step 6: Sort rows of matrix;

Step 7: For p=1 to k

If L1<a<U1 && L1<a<U2 then

Increment g;

End if

End for

Step 8: For counter=1 to g

If counter==1 then

Marker is stored in Primary Repository

Else if counter==2 then

Marker is stored in Secondary Repository

Else if counter==3 then

Marker is stored in Tertiary Repository

Else

Miscellaneous Marker

End If

End For

Step 9: End

TABLE 4.1: *Geometric properties of Immovable Objects*

S.no.	**Region No.**	**Area**	**Centroid**	
			X (co-ordinate)	**Y (co-ordinate)**
1	33	10734	110.3333	103.3333
2	59	50170	111.5	91
3	2	20	113	24
..	..	..	..	..
N	N	1	112	119

Table 4.1 shows all the geometrical properties for all immovable objects present in the frame.

The column region number shows the label number of the particular object. It defines the arbitrary numbers assigned to all regions in the frame. One can fetch the desired property by only using this label number/region number. If one wants to fetch the area value of the desired region, then one can use the region number to do the same. Column marked as area is the area value of the particular region or the object. The last column defines the centroid that contains the x and y coordinates of the specific region. This can be termed as the center of the mass of the particular region or the location of the immovable object.

TABLE 4.2: *Non-trivial markers in the frame*

S. no.	**X co-ordinate of Marker**	**Y co-ordinate of Marker**	**Categorized Marker (Area)**
1	102.8233	29.8127	Primary Marker (-1415)
2	158.5996	-22.1767	Secondary Marker (-1109)
3	201.591	60.6368	Tertiary Marker (-939)

Table 4.2 presents the non-trivial markers found in a particular frame.

These are the markers that have gone through the three levels of triviality. This categorization has been done based on the weightage of the area value. Once the non-trivial markers are categorized into primary, secondary, and tertiary markers, they are stored into their respective repositories.

Values can be decided in reference to the variables which stand distinct in between the upper and lower values that are region occupied by the ground and the sky.

TABLE 4.3: *Threshold value of markers*

S. N.	**X**	**Y**	**Area**	**Region**
1	161.2937	42.055762	269	42
2	137.4968	116.92722	316	30
3	240.7872	45.469231	390	62
4	**264.2367**	**128.97472**	**1424**	**63**
5	**265.6339**	**45.001273**	**3141**	**61**
6	**197.9201**	**59.335582**	**5781**	**38**
7	71.03485	51.48956	12740	2
8	164.5524	218.55365	44474	3

Table 4.3 shows the threshold values of the markers picked from the backend where x and y are the coordinates of the markers of the particular marker. Region number represents the region number within the frame. ‘Area’ represents the area value of the particular marker within the frame.

As the proposed categorization mechanism detects and categorizes the markers from the real-time environment, it is possible that a situation may arise when there will be two markers present at parallel spaces fulfilling the criteria to be categorized into same type of markers for example for the type of primary marker. This kind of condition is known as tie-breaking condition. The probability 'P' of the condition of the tie-breaking between two markers is significantly low as the markers are detected and categorized on the basis of various parameters, size of the marker, area of the marker, its visibility from the distance, centroid etc.

When the markers are present at parallel spaces (tie-breaking condition), then there will be a case of an isosceles triangle present in the real-time environment. This type of case is very rarely present in the real-time scene.

$$P = -\frac{1}{23}\text{; of the cases}$$

To deal with such kind of tie-breaking condition, the marker present on the driver side of the system will be considered as the potential marker by default.

4.2.2 Matching process

During the run-up process, markers are identified to validate the path to check whether the system is on the right path or not. For this, an algorithm is developed to detect the markers. Figure 4.5 shows how the matching should take place and help navigate safely in the run time environment. In this algorithm, the non-trivial marker is detected by applying the categorization mechanism, and then matching is done on a non-trivial marker. These features of the first encountered run time marker will be matched with the primary feature vector file. This will show our system on the right path. But if the primary markers are not matched, then it will look for the secondary match at an average speed, if the secondary marker is not found, then it will go for the tertiary marker at a very slow speed. If the tertiary marker is also not found, then the system is concluded to be wrongly set. As such, the validation system of the path is performed.

In the next runs, from the run time video, the run time markers are detected. Their features are extracted using the feature extraction process. For the validation purpose, the system will look into the repository for matching those features. If the run time features are matched with the features present in the repository, then the system will move forward and look for the next marker.

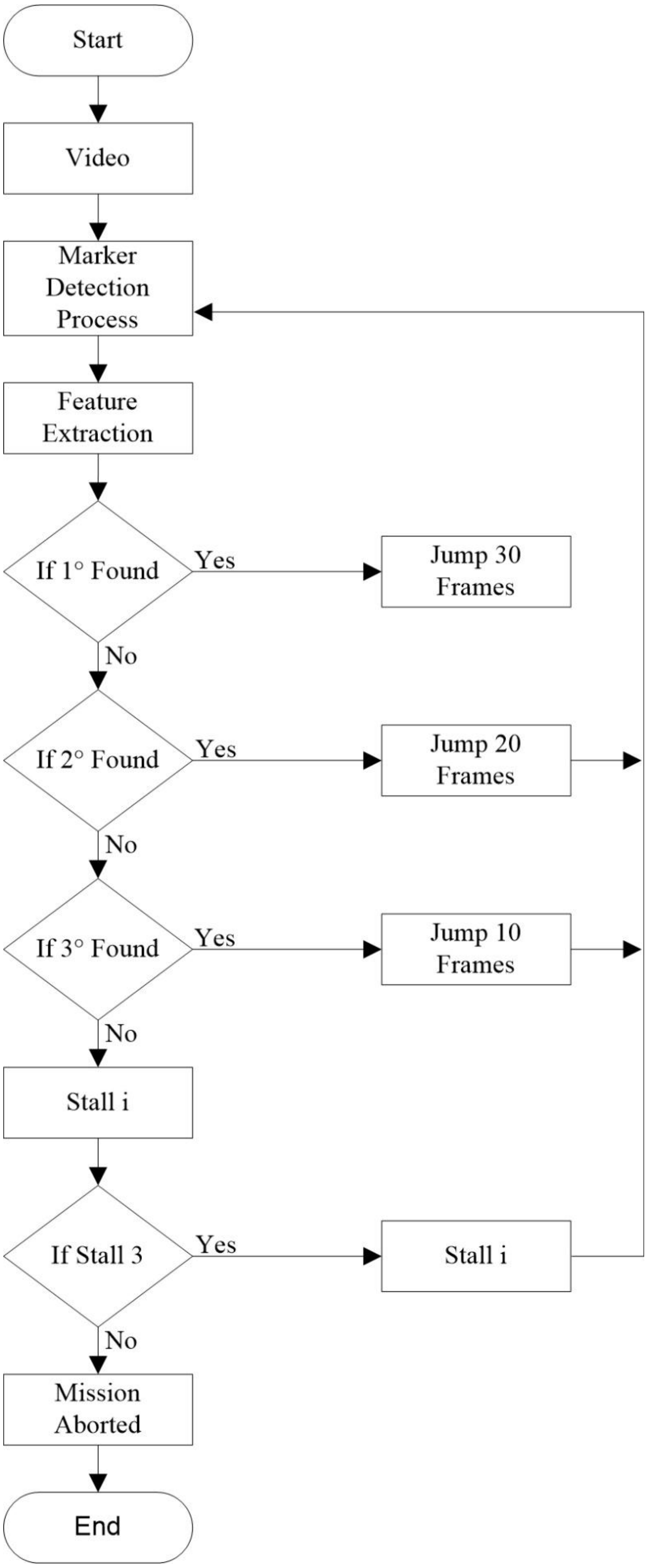

FIGURE 4.5: *Matching process of ANMIC*

4.2.2.1 Pseudo-code for matching process

With stall 3,

If the primary marker is found,

Speed = Normal

If the secondary marker is found

Speed = Average

In the absence of all this, the system appears defective. So it is better to stop and plan otherwise.

TABLE 4.4: *Identification of markers*

S. No.	**Captured Marker**	**Identified marker**	**Matching Status**
1	Marker 1	Primary Marker 1	Found
2	Marker 2	Primary Marker 2	Found
3	Marker 3	Secondary Marker 3	Found
..	..	..	..
N	Marker n	Primary Marker n	Found

Table 4.4 shows the identification of the markers during the matching process (run-time).

The captured marker is detected marker during the run time. During the next runs, when the system will navigate from A to B, then the run time markers are detected and matched with the markers stored in the repository made in bare run.

When the marker is detected during the run time, the system will look for that marker features in the repositories. if the marker is found there, then the system will validate its path to see that whether it is on the desired route or not.

Matching status tells us that run time marker features are matched with the markers present in the repositories, to reflect the “Found” status which is else “NOT FOUND.”

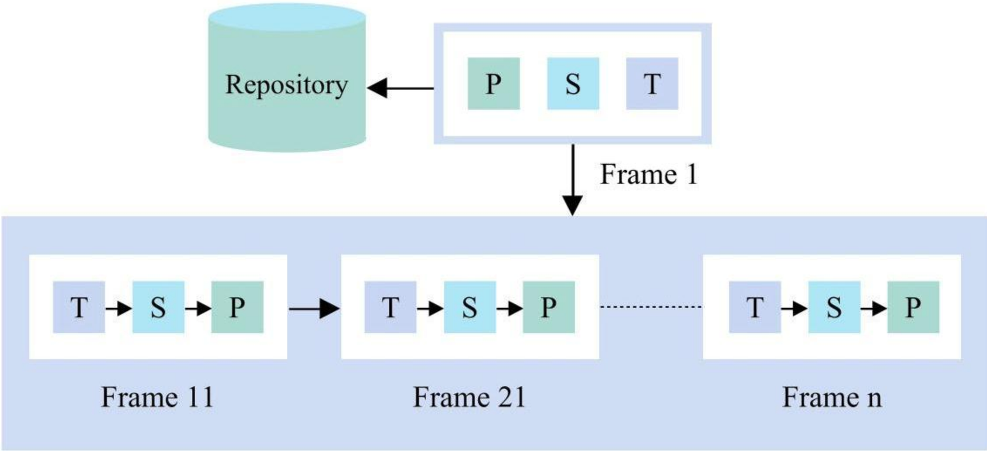

FIGURE 4.6: *Process Diagram explaining the dynamicity of markers during run time*

Figure 4.6 reflects that at the start of the route, primary, secondary, and tertiary markers are highlighted in a particular frame. When the system moves, with the change in the frames, the priority of the markers will also get changed. In Frame 1, there are three markers present, i.e., primary, secondary, and tertiary, that are already stored in the image repository as primary, secondary, and tertiary, respectively. After a gap of 10 frames, in frame 11, the Secondary marker will become the new primary and tertiary will become the new secondary. This process will repeat itself till frame N. This alteration in priority of the markers is for the run time only, and the image repository remains unaltered.

Here, Figure 4.6 explains this concept with the help of the diagram. Here,

P – Primary Marker

S – Secondary Marker

T- Tertiary Marker

4.3 RESULTS AND DISCUSSION

The proposed method has been applied to the video frames to navigate safely in the unknown environment without using a pre-built database in MATLAB tool. Figures 4.7 to 4.11 show the results generated through the application MATLAB 2017b after the categorization of the markers from a video.

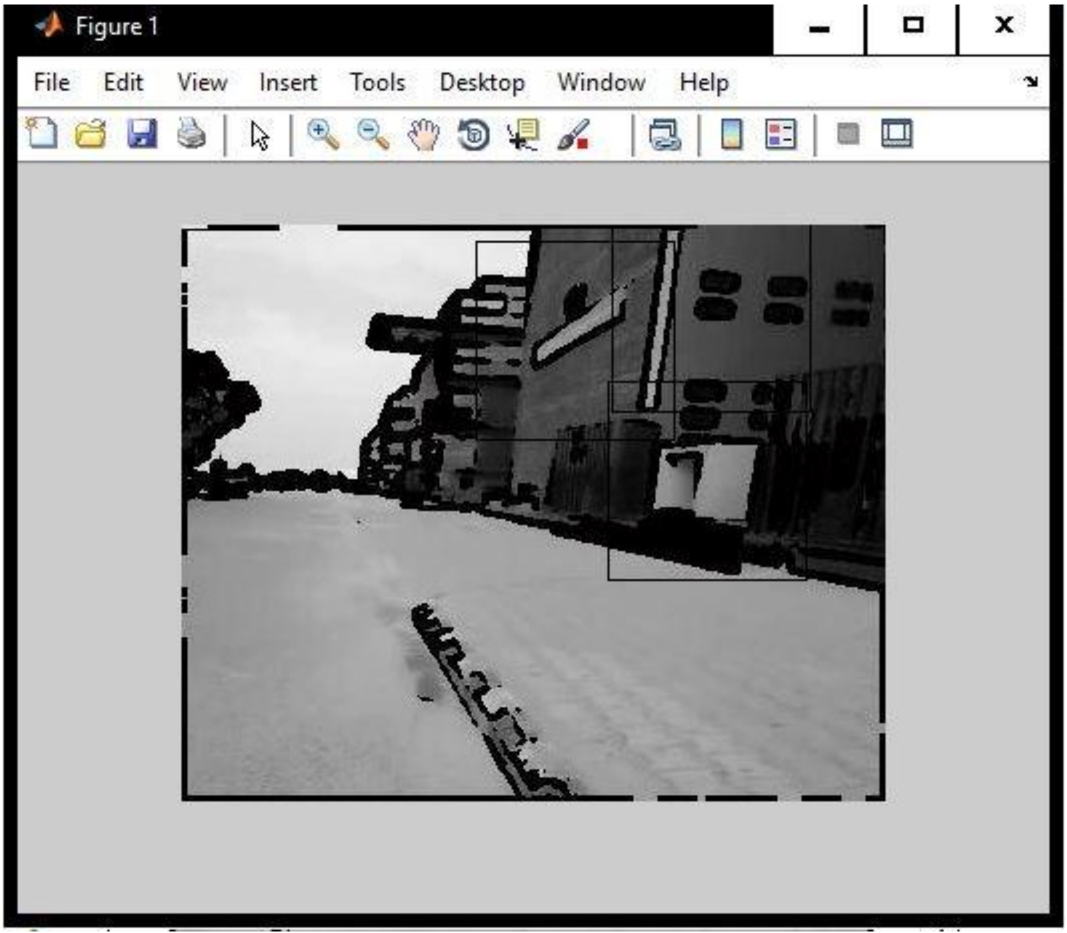

FIGURE 4.7: *Results after categorizing the markers*

Figure 4.7 shows the results of the detection of markers after the categorization process. These are the immovable objects satisfying the threshold conditions to be considered as the non-trivial markers.

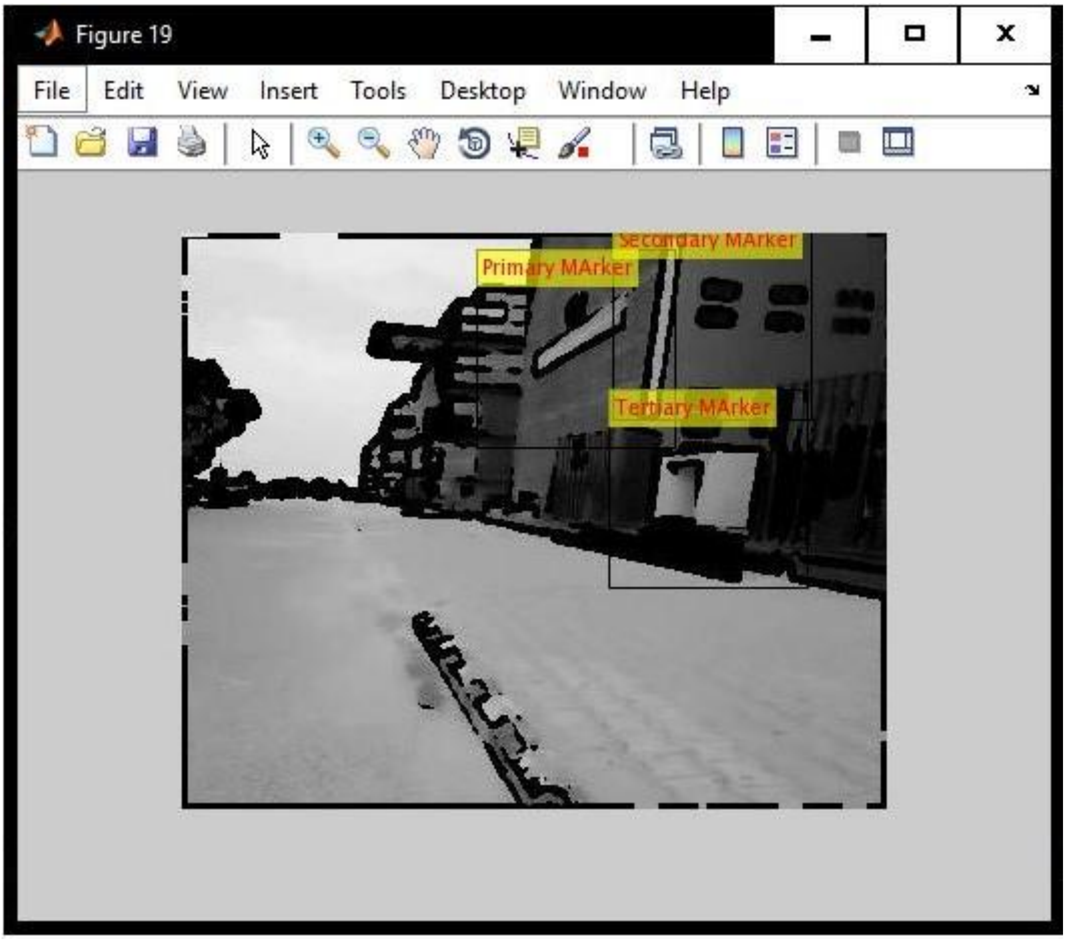

FIGURE 4.8: *Results after prioritizing the markers*

Figure 4.8 underlines the results after prioritizing the non-trivial markers into Primary, secondary, and tertiary. These accented markers are stored in their respective repositories.

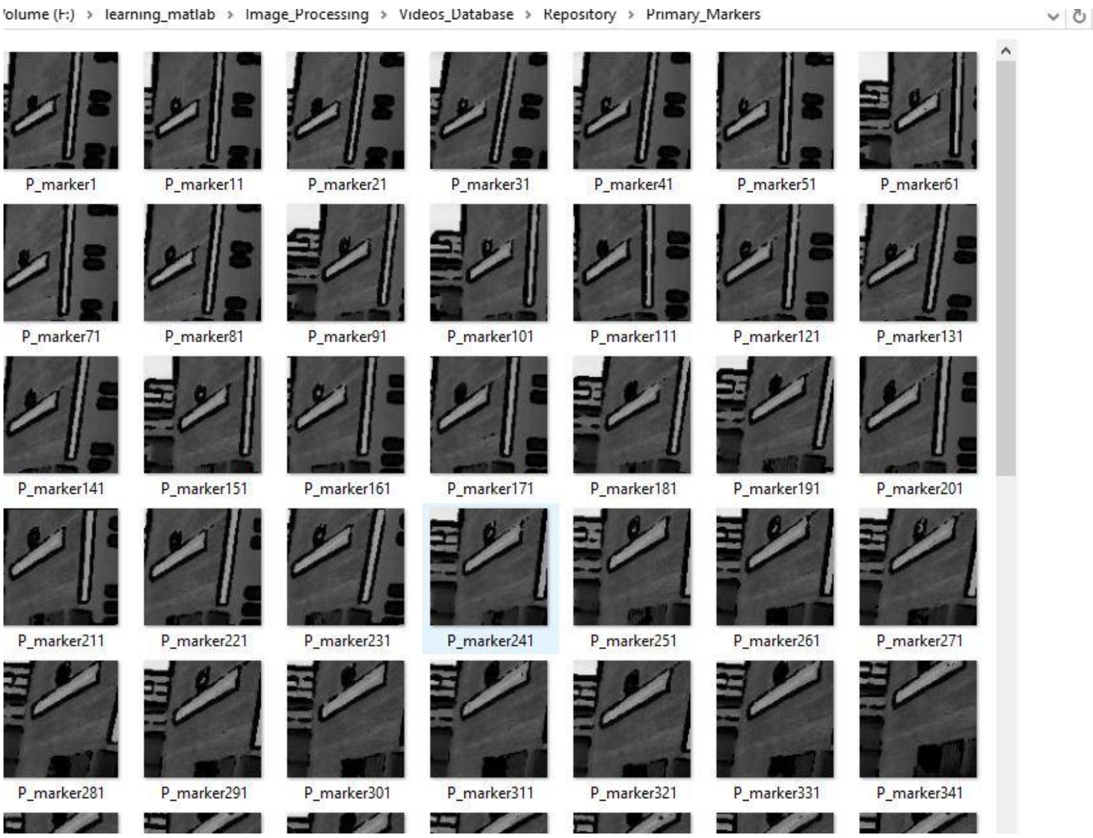

FIGURE 4.9: *Results showing the primary markers storage in the Primary Repository*

Figure 4.9 shows the primary repository in which all the primary markers are stored after the categorization process during the Bare Run. This repository will help the system to navigate safely for a specific path during the next runs.

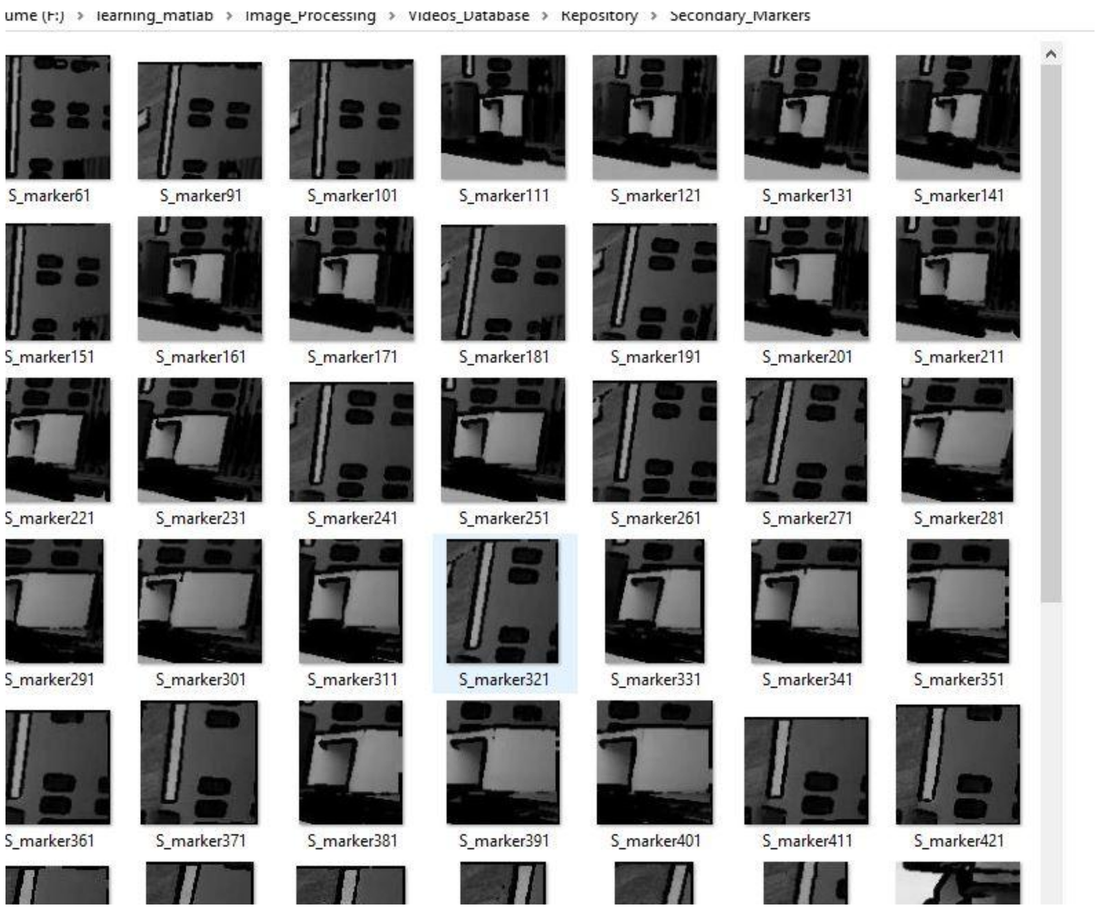

FIGURE 4.10: *Results showing the secondary markers storage in the Secondary Repository*

Figure 4.10 depicts the secondary repository in which all the secondary markers are stored after the categorization process during the Bare Run. This repository will help the system to navigate safely for a specific path during the next runs. This is also known as a supportive repository as these markers can play their roles if the primary marker is not present during the next run at a specific point.

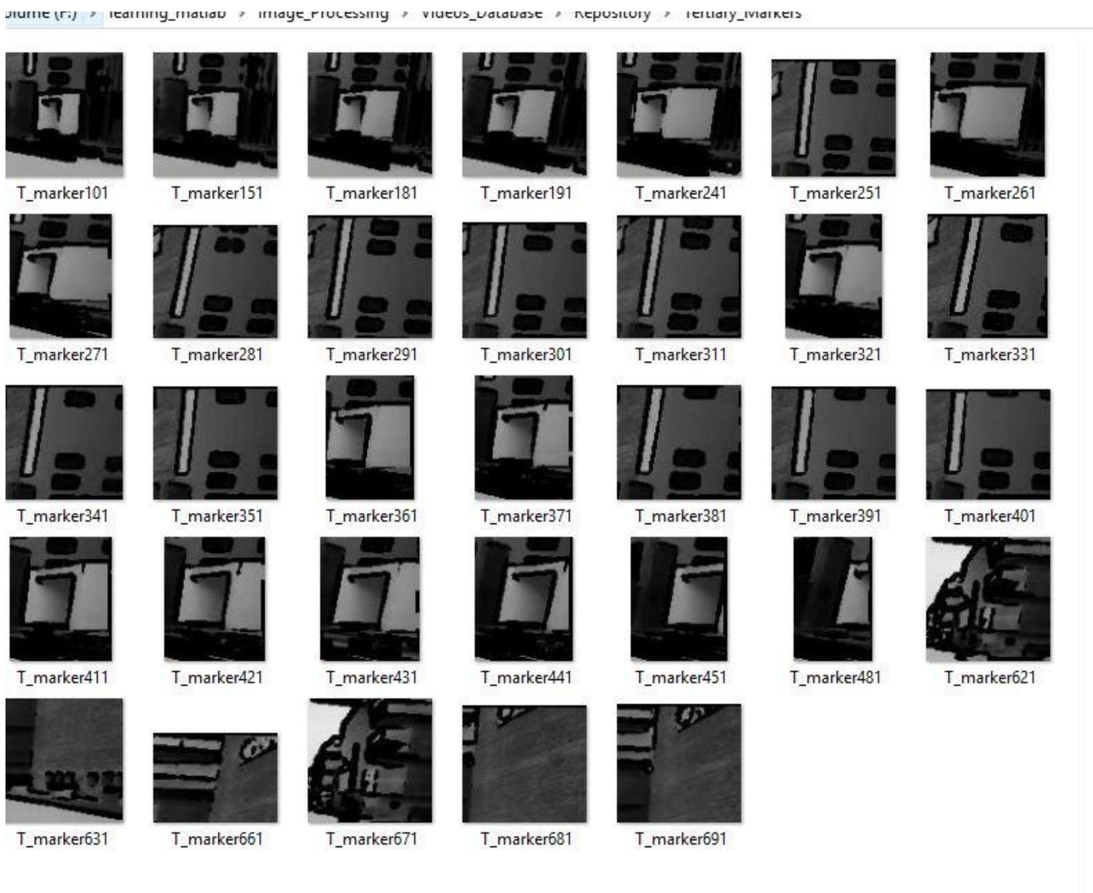

FIGURE 4.11: *Results showing the tertiary markers storage in the Tertiary Repository*

Figure 4.11 shows the tertiary repository in which all the tertiary markers are stored after the categorization process during the bare run. This repository will help the system to navigate safely for a specific path during the next runs. This is also known as Backup Repository, as these markers can play their role if the primary, as well as the secondary markers, are not present during the next run at a specific point.

4.4 COMPARATIVE ANALYSIS

Vision-based navigation has been performed with the help of camera, GPS and sometimes sensors. A lot of work has been done by using the stereo cameras and pre-built datasets. There are various existing benchmarked datasets available such as KITTI, Cityscapes, IDD, Daimler and many more. The existing datasets are prepared for the structured environments but in reality, in Indian roads, the roads are unstructured so, to implement the navigation algorithm, the dataset is made. In the existing works, color, shape, corners, edges features are extracted from the video frames to get the desired navigational information.

However, this research focuses on the utilization of the monocular camera for excavating the meaningful information from the run-time video similarly done with the use of stereo-cameras. Moving ahead, given below is the qualitative analysis of our approach with the existing approaches based on various factors:

- **Prominency:** This parameter gives the in depth information about the major factor on the basis of which the research work has been conducted. Such as, Royer et al. [207] in their work have used 3D maps as major prominent factor for navigating in real-world environment. Shintazo et al. [208] and Yebes et. al. [209] have used the color component as the major factor in their work. However, our proposed approach has focused on the human way of thinking while navigating on the road. Therefore, size of the marker, its visibility from a large distance is the prominence factors in this work.

- **Novelty:** This parameter gives the information about the uniqueness in the proposed work. Here, Royer et al. [207] have worked on a monocular camera and prepared their own dataset. They have proposed a 3 step navigation algorithm for the urban outdoor environments. Natural landmarks are detected using '2' lenses with the interest point selection techniques. But in our proposed approach three types of the natural landmarks are detected at a single time named as the main, supportive and backup markers which are best in case of lost position of the existing marker.

- **Prebuilt-datasets:** Pre-built datasets are the datasets that are required for training the data or for evaluating the experimental results for the proposed work. Yebes et al. [209] and Oh et al. [210] have evaluated results on KITTI dataset. Royer et al. [207] has evaluated their results on the Cityscapes datasets. As these datasets are totally structured, Zhang et al. [211] has proposed the navigational algorithm for the road detection and the road following. No prior knowledge of the environment is required. The main challenge in this work is this is for the small lane roads. But in our proposed approach, the algorithm is designed to navigate in the urban areas.
- **Type of camera:** A camera is used as the basic contributor in the vision based navigation processes. Yebes et al. [209] made use of the stereo images to detect the landmarks for the road navigation. Royer et al. [207] has used the monocular camera for the navigation task even in the outdoor environments.
- **Feature Selection:** Feature selection is the important process while identifying the landmarks or objects from the real-world environment. Yebes et al. used the color features for the detection of the information from the real time images. Royer et al. [207] used the Harris corners for the feature selection from the input image to acquire the meaningful navigational information. Schilling et al [209] has used the geometric as well as the visual feature classification for the autonomous navigation. So, this has used the multi-sensors for implementing the proposed algorithm. Shinzato et al. [208] has used the features like energy, entropy, average and variance from the different color images. However, in our proposed approach, the very primary characteristics like area of the interest points, size, visibility from the far distance have been chosen that are independent of the color of the images.
- **Environment:** This parameter describes the surrounding for which the research has been conducted. Shinzato et al. [208] have done the navigation work for the urban environments. In our proposed work, the navigation operation is performed on the urban outdoor environments.
- **Sensors:** This parameter is associated with various devices that are required to collect the input data for the implementation of the proposed methodology. These devices are known as sensors. For landmark based navigation, Oh et al.

[210] have also used various types of sensors like LIDAR and charge couple device to detect the cars, cyclist and pedestrians. Sometimes, sonar sensors and compass are also used for the navigation task. But in our proposed work, we have used a single camera for the detection of the landmarks for the navigation purpose.

The above discussed qualitative comparative analysis is shown in Table 4.5 below :

TABLE 4.5. *Comparative analysis of different models with proposed model*

References / Parameters	**Royer et al. [207]**	**Shinzato et al. [208]**	**Yebes et al. [209]**	**Oh et al. [210]**	**Zhang et al. [211]**	**Proposed method**
Prominency	3D map	Color and texture	Color	Fusion of multiple sensors	Color	Geometric parameters
Novelty	Natural landmarks	ANN	Stereo data for KITTI object challenge	KITTI	Small lane detection	Human way of navigation
Prebuilt-datasets	Cityscapes	-	KITTI	No	No	No
Type of camera	Monocular	Camera	Stereo	-	Monocular	Monocular
Feature Selection	Corners	Energy, entropy, average	Color	-	Road borders	Movability, size, area, centroid.
Environment	Outdoor	Urban	Urban	-	Outdoor	Urban
Sensors	Two lenses	Digital camera	Stereo camera	Lidar + charged couple device	Camera	Single camera

4.4.1 Platform Description

The mobile robot used for the experiments is a four wheeled driving system on which the front facing monocular camera has been placed. The camera is placed in the front of the device. The main specification of the camera is 12 MP camera with the resolution of the 4000x3000 pixels and recorded at the 1920 x 1080, 30fps.

TABLE 4.6: *Platform Description*

Sensor type	Monocular Camera
Resolution	4000X 3000
Format	JPEG or BMP
Frame rates	30 fps
Focal Length	28 mm
Max CCD format	1/3”
Max Aperture	3.6171875
Video compression	H.265/H.264

4.4.2 System Architecture

The system architecture consists of the maker detection and marker localization system. The marker detection is used to detect the markers present on the unstructured roads, where the grayscale images can be very helpful for the detection purpose irrespective of the color of the markers. Optical flow is implemented to estimate the mobility of the objects on the road. Lucas-Kannade technique has been used while distinguishing the movable and non-movable objects for getting the more efficient results.

4.4.3 Dataset

We have made 10 videos from Amritsar highway to the Ludhiana highway, and Lovely Professional university campus, Phagwara for each. The number of frames generated from these videos depends on the length of the video. The frame rate is 30fps. The proposed algorithm has been applied on these frames to get the anticipated results.

4.4.4 Quantitative comparative analysis

Performance metrics

Pixel accuracy: Per class pixel accuracy is determined by calculating the percent of pixels present within the image.

$$Accuracy = \frac{TP+TN}{(TP+TN+FP+FN)} \qquad \dots (4.3)$$

This metric is not constantly consistent as it might prompt deluding results at some point.

Instance segmentation: This method is based on the calculation of marker instead of the bounding box.

TP: A true positive is perceived when an expected-actual marker pair has an IoU score which surpasses some predefined limit.

FP: A false positive demonstrates an expected marker had no related ground truth object marker.

FN: A false negative demonstrates a ground truth object marker had no related predicted object marker.

Precision: Every predicted marker is compared with the actual detected marker correspond to each input.

$$Precision = \frac{TP}{(TP+FP)} \qquad \ldots (4.4)$$

Recall: Recall is the part of the aggregate sum of significant predictions that were really retrieved.

$$Recall = \frac{TP}{(TP+FN)} \qquad \ldots (4.5)$$

However, in order to calculate the prediction and recall, there is a need to define what constitutes a positive detection**.** For this, IoU score between each (prediction, actual) marker pair has been calculated and then it has been determined which marker pairs have an IoU score exceeding a defined threshold value.

Intersection over Union:

The Intersection over Union (IoU) metric, is also known as Jaccard index, is essentially a method to measure the overlap percentage between the actual marker and predicted marker. The IoU metric measures intersected region between actual marker and predicted marker divided by the union of the actual marker and predicted marker.

$$IoU = \frac{Actual \cap Predicted}{(Actual \cup Predicted)} \quad \dots (4.6)$$

Mean IoU is calculated to compute the global semantic segmentation prediction by taking average of all the actual and predicted marker pair IoU score.

TABLE 4.7: *Actual Primary markers*

Frame No.	**x1**	**y1**	**x2**	**y2**	**x3**	**y3**	**x4**	**y4**
Frame 1	136	139	196	140	196	5	143	6
Frame 11	133	142	183	153	194	2	141	9
Frame 21	147	143	196	156	224	4	161	13
Frame 31	149	141	196	157	227	4	166	10
Frame 41	131	143	182	156	202	4	140	8
Frame 51	121	142	170	154	180	3	123	6
Frame 61	116	142	170	155	176	4	119	4
Frame 71	133	147	182	157	193	4	135	10
Frame 81	146	143	195	149	212	3	156	5
Frame 91	151	136	208	98	218	5	159	6
Frame 101	138	141	193	4	198	4	141	6
Frame 111	137	136	193	143	193	5	136	2
Frame 121	159	136	209	148	220	4	161	3
Frame 131	171	140	223	154	244	3	186	4
Frame 141	168	136	219	149	240	2	183	4
Frame 151	143	140	202	98	207	2	146	3
Frame 161	158	132	219	92	226	1	163	2
Frame 171	169	138	235	92	241	1	175	2
Frame 181	163	138	229	96	242	1	179	1
Frame 191	171	128	236	90	250	2	186	3
Frame 201	149	137	213	93	218	4	153	3

TABLE 4.8: *Actual Secondary markers*

Frame	**x1**	**y1**	**x2**	**y2**	**x3**	**y3**	**x4**	**y4**
Frame 111	205	141	240	142	240	107	205	111
Frame 121	224	140	260	142	264	109	226	112
Frame 131	238	145	276	148	282	115	240	117
Frame 141	236	140	276	144	281	110	240	111
Frame 151	211	147	255	149	258	110	212	113
Frame 161	228	138	274	141	278	102	229	103
Frame 171	241	142	291	147	295	103	244	103
Frame 181	235	144	284	152	292	106	240	108
Frame 191	246	92	304	99	349	2	260	4
Frame 201	226	107	280	102	349	2	228	3
Frame 211	230	99	289	96	351	2	232	3
Frame 221	-	-	-	-	-	-	-	-
Frame 231	251	89	318	99	337	2	268	2
Frame 241	240	89	312	99	350	3	250	3
Frame 251	237	132	303	143	309	99	240	101
Frame 261	266	89	339	97	347	2	278	2
Frame 271	281	92	348	99	350	2	295	2
Frame 281	263	134	343	149	352	95	272	97
Frame 291	254	91	351	101	350	1	268	2
Frame 301	250	136	343	144	350	95	255	98

TABLE 4.9: *Actual Tertiary markers*

Frames	**x1**	**y1**	**x2**	**y2**	**x3**	**y3**	**x4**	**y4**
Frame 191	243	136	291	143	300	103	248	105
Frame 201	223	141	277	149	279	108	226	111
Frame 211	228	135	282	144	284	102	229	106
Frame 221	250	139	304	150	309	106	253	107
Frame 231	248	136	307	141	315	101	254	104
Frame 241	238	136	302	142	308	103	243	107
Frame 251	237	85	307	92	323	3	248	3
Frame 261	262	137	339	144	346	100	265	104
Frame 271	276	140	350	148	349	104	282	106
Frame 281	267	85	351	94	350	1	285	1
Frame 291	252	143	338	148	345	103	257	105

TABLE 4.10: *Predicted Primary markers*

Frame	**x1'**	**y1'**	**x2'**	**y2'**	**x3'**	**y3'**	**x4'**	**y4'**
Frame 1	135	140	197	113	210	5	143	11
Frame 11	133	136	221	108	249	23	142	16
Frame 21	150	139	240	114	272	27	167	15
Frame 31	153	136	243	101	271	3	173	12
Frame 41	136	138	202	115	218	3	146	11
Frame 51	124	138	241	113	227	6	131	7
Frame 61	119	139	162	116	174	2	123	5
Frame 71	135	143	227	115	263	6	142	9
Frame 81	146	140	243	97	264	4	160	6
Frame 91	153	133	249	107	268	4	164	5
Frame 101	205	143	239	146	238	114	205	199
Frame 111	140	131	209	101	245	3	137	4
Frame 121	159	132	256	103	276	5	165	5
Frame 131	173	138	276	93	298	6	189	4
Frame 141	170	133	275	79	296	5	185	4
Frame 151	145	133	246	87	262	2	150	3
Frame 161	160	128	264	87	281	3	168	2
Frame 171	172	132	279	78	297	2	180	1
Frame 181	166	131	279	77	296	1	183	1
Frame 191	171	129	235	88	250	3	188	2
Frame 201	149	136	212	96	218	4	155	4

TABLE 4.11: *Predicted Secondary markers*

Frames	**x1'**	**y1'**	**x2'**	**y2'**	**x3'**	**y3'**	**x4'**	**y4'**
Frame 111	205	138	240	140	240	107	205	111
Frame 121	226	139	260	143	264	110	226	113
Frame 131	240	145	277	149	282	115	244	118
Frame 141	236	140	275	144	280	110	240	113
Frame 151	213	143	252	145	256	112	215	114
Frame 161	229	135	272	139	277	103	235	106
Frame 171	245	138	290	145	294	104	245	108
Frame 181	237	141	284	146	291	108	240	109
Frame 191	249	101	285	79	309	2	261	3
Frame 201	226	105	260	88	277	6	229	6
Frame 211	227	101	261	94	278	4	232	4
Frame 221	-	-	-	-	-	-	-	-
Frame 231	254	100	297	69	316	4	267	3
Frame 241	241	102	279	73	298	3	249	3
Frame 251	237	135	303	143	309	98	241	102
Frame 261	270	99	301	94	317	1	278	3
Frame 271	284	102	323	72	349	4	297	4
Frame 281	264	134	342	148	352	96	269	100
Frame 291	260	102	299	72	315	2	267	3
Frame 301	253	135	341	138	348	97	255	98

TABLE 4.12: *Predicted Tertiary markers*

Frames	**x1'**	**y1'**	**x2'**	**y2'**	**x3'**	**y3'**	**x4'**	**y4'**
Frame 191	244	136	293	145	300	103	149	108
Frame 201	224	141	274	150	278	109	225	111
Frame 211	250	139	283	145	284	102	234	104
Frame 221	249	141	304	150	309	106	252	108
Frame 231	249	136	304	139	314	103	255	105
Frame 241	239	138	301	141	305	104	243	108
Frame 251	241	97	275	64	289	5	247	41
Frame 261	291	142	336	143	343	101	273	104
Frame 271	278	141	349	147	350	104	282	108
Frame 281	272	95	315	62	339	4	286	2
Frame 291	283	147	336	153	344	103	264	106

TABLE 4.13: *Absolute Error in Primary Markers*

Frames	**A**	**B**	**C**	**D**	**E**	**F**	**Absolute error**	**MAPE**
	x2-x1'	**y3-y2'**	**x2-x1**	**y3-y2**	**A*B**	**C*D**	**E/F**	**(Absolute error/N)*100**
Frame 1	61	-108	60	-135	-6588	-8100	0.813	0.90
Frame 11	50	-106	50	-151	-5300	-7550	0.701	0.77
Frame 21	46	-110	49	-152	-5060	-7448	0.679	0.75
Frame 31	43	-97	47	-153	-4171	-7191	0.580	0.64
Frame 41	46	-111	51	-152	-5106	-7752	0.658	0.73
Frame 51	46	-110	49	-151	-5060	-7399	0.683	0.75
Frame 61	51	-112	54	-151	-5712	-8154	0.700	0.77
Frame 71	47	-111	49	-153	-5217	-7497	0.695	0.77
Frame 81	49	-94	49	-146	-4606	-7154	0.643	0.71
Frame 91	55	-102	57	-93	-5610	-5301	1.058	1.1
Frame 101	-12	-142	55	0	1704	–	–	–
Frame 111	53	-96	56	-138	-5088	-7728	0.658	0.73
Frame 121	50	-99	50	-144	-4950	-7200	0.687	0.76
Frame 131	50	-90	52	-151	-4500	-7852	0.573	0.63
Frame 141	49	-77	51	-147	-3773	-7497	0.503	0.55
Frame 151	57	-85	59	-96	-4845	-5664	0.855	0.95
Frame 161	59	-86	61	-91	-5074	-5551	0.914	1.01
Frame 171	63	-77	66	-91	-4851	-6006	0.807	0.89
Frame 181	63	-76	66	-95	-4788	-6270	0.763	0.84
Frame 191	65	-86	65	-88	-5590	-5720	0.977	1.08
Frame 201	64	-92	64	-89	-5888	-5696	1.033	1.14

TABLE 4.14: *Absolute error in Secondary markers*

Frames	A	B	C	D	E	F	Absolute error	MAPE
	x2-x1'	y3-y2'	x2-x1	y3-y2	A*B	C*D	E/F	(Absolute error/N)*100
Frame 111	35	-33	35	-35	-1155	-1225	0.942	1.57
Frame 121	34	-34	36	-33	-1156	-1188	0.973	1.62
Frame 131	36	-34	38	-33	-1224	-1254	0.976	1.62
Frame 141	40	-34	40	-34	-1360	-1360	1	1.66
Frame 151	42	-35	44	-39	-1470	-1716	0.856	1.42
Frame 161	45	-37	46	-39	-1665	-1794	0.928	1.54
Frame 171	46	-42	50	-44	-1932	-2200	0.878	1.46
Frame 181	47	-40	49	-46	-1880	-2254	0.834	1.39
Frame 191	55	-77	58	-97	-4235	-5626	0.752	1.25
Frame 201	54	-86	54	-100	-4644	-5400	0.860	1.43
Frame 211	62	-92	59	-94	-5704	-5546	1.028	1.71
Frame 221	0	0	0	0	0	0	0	0
Frame 231	64	-67	67	-97	-4288	-6499	0.659	1.09
Frame 241	71	-70	72	-96	-4970	-6912	0.719	1.19
Frame 251	66	-44	66	-44	-2904	-2904	1	1.66
Frame 261	69	-92	73	-95	-6348	-6935	0.915	1.52
Frame 271	64	-70	67	-97	-4480	-6499	0.689	1.14
Frame 281	79	-53	80	-54	-4187	-4320	0.969	1.61
Frame 291	91	-71	97	-100	-6461	-9700	0.666	1.11
Frame 301	90	-43	93	-49	-3870	-4557	0.849	1.41

TABLE 4.15: *Absolute error in Tertiary markers*

Frames	A	B	C	D	E	F	Absolute error	MAPE
	x2-x1'	y3-y2'	x2-x1	y3-y2	A*B	C*D	E/F	(Absolute error/N)*100
Frame 191	47	-42	48	-40	-1974	-1920	1.028	3.42
Frame 201	53	-42	54	-41	-2226	-2214	1.005	3.35
Frame 211	32	-43	54	-42	-1376	-2268	0.606	2.02
Frame 221	55	-44	54	-44	-2420	-2376	1.018	3.39
Frame 231	58	-38	59	-40	-2204	-2360	0.933	3.11
Frame 241	63	-38	64	-39	-2394	-2496	0.959	3.19
Frame 251	66	-61	70	-89	-4026	-6230	0.646	2.15
Frame 261	48	-43	77	-44	-2064	-3388	0.609	2.03
Frame 271	72	-43	74	-44	-3096	-3256	0.950	3.16
Frame 281	79	-61	84	-93	-4819	-7812	0.616	2.05
Frame 291	55	-50	86	-45	-2750	-3870	0.710	2.36

Table 4.7, 4.8, 4.9, 4.10, 4.11, 4.12 represents the actual and predicted marker data for primary, secondary and tertiary markers. Some of the frames may not have any marker present e.g Frame 221. Figure 4.12, 4.13, 4.14, 4.15, 4.16, 4.17 illustrates the performance comparison of the proposed algorithm using IoU score and statistics inculding accuracy, precision, recall, F-score, TPR, FPR rate, mean absolute error of the primary, secondary and tertiary markers respectively.

Figure 4.12 depicts that the average IoU score of the primary markers is 65%. Figure 4.13 shows that the average IoU score of the secondary markers is 66%. Figure 4.14 shows that the average IoU score of the tertiary markers is 71%. Figure 4.15 shows the statistics of the primary markers where accuracy is 96%, mean absolute percentage error is 1.25%, Precision is 100%, Recall value is 96.6%, F-score is 1, TPR rate is 0.966 and FPR rate is 0. Figure 4.16 shows the statistics of the secondary markers where accuracy is 86.6%, mean absolute percentage error is 3%, Precision is 80.9%, Recall value is 100%, F-score is 0.89, TPR rate is 1 and FPR rate is 0.30. Figure 4.17 shows the statistics of the tertiary markers where accuracy is 92%, mean absolute percentage error is 4%, Precision is 86.3%, Recall value is 100%, F-score is 0.92, TPR rate is 1 and FPR rate is 0.13. The above statistics is represented in the tabular form in Table 4.16.

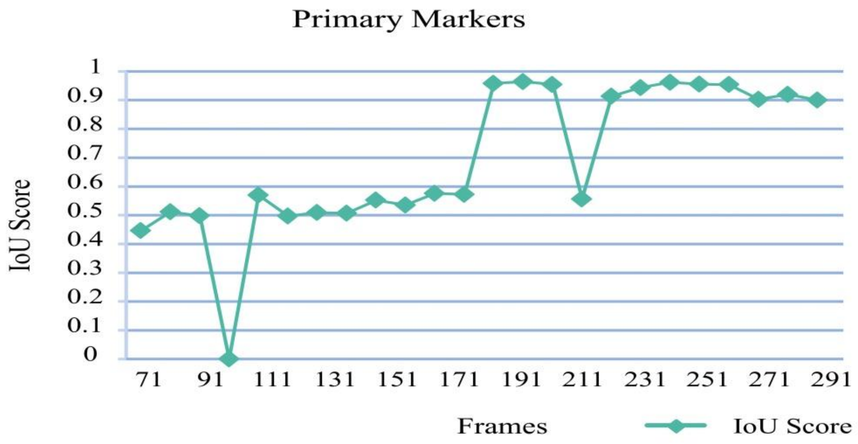

FIGURE 4.12: *IoU Score of the Primary markers*

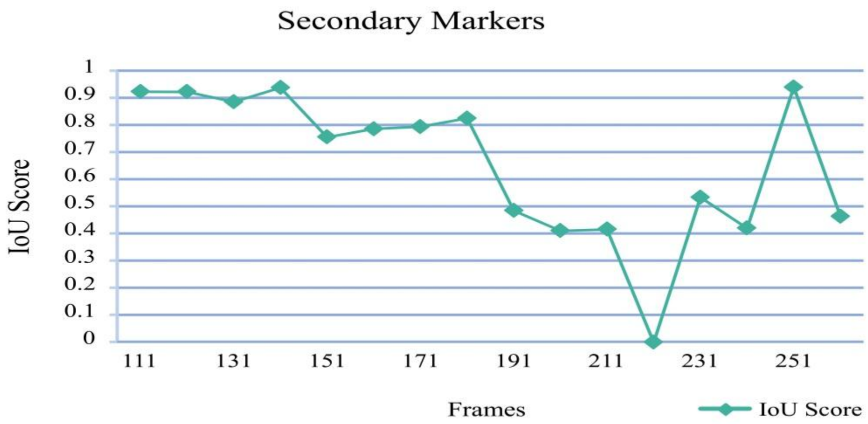

FIGURE 4.13: *IoU Score of Secondary markers*

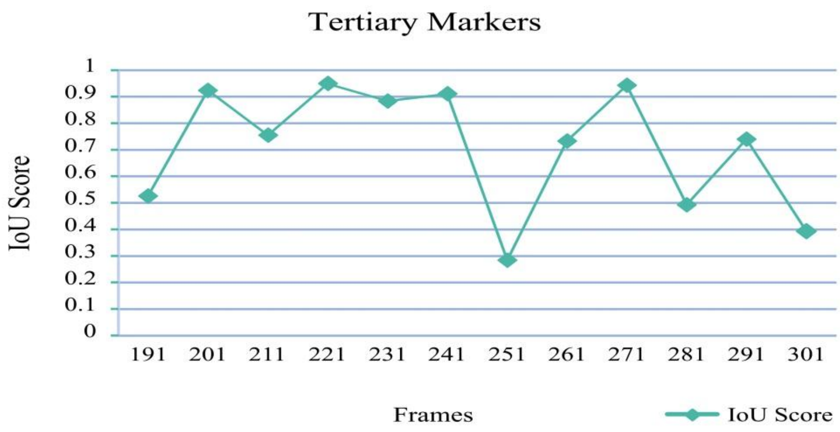

FIGURE 4.14: *IoU Score of Tertiary markers*

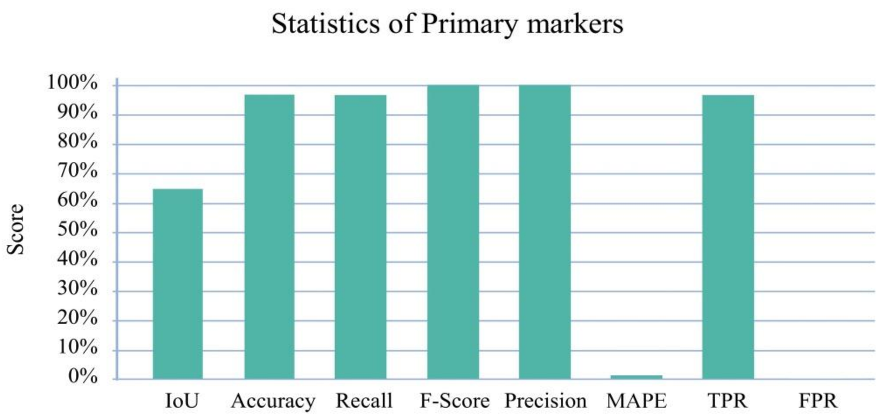

FIGURE 4.15: *Statistics of Primary markers*

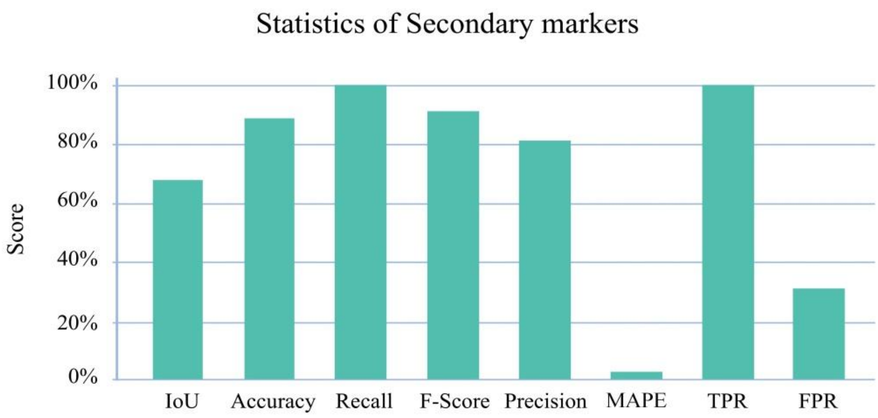

FIGURE 4.16: *Statistics of Secondary markers*

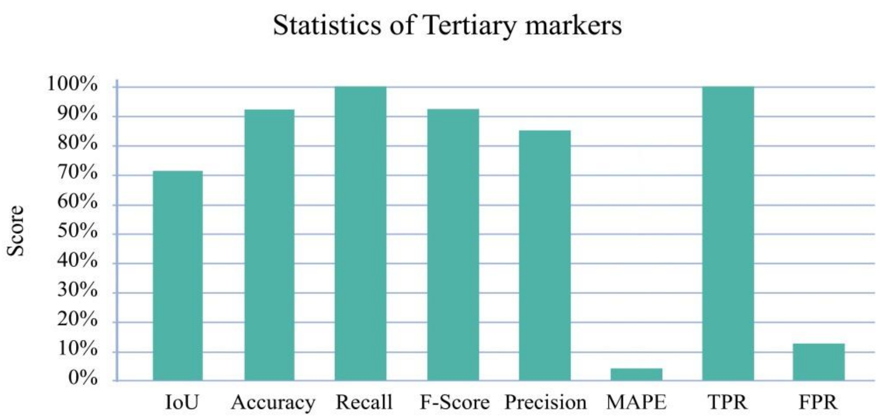

FIGURE 4.17: *Statistics of Tertiary markers*

The statistics from the Figure 4.14 to 4.17 is presented in the below table 4.16.

TABLE 4.16: *Statistics of Primary, Secondary and Tertiary markers*

Markers / Score	Primary	Secondary	Tertiary
IoU	0.65	0.66	0.71
Accuracy	0.96	0.86	0.92
MAE	0.80	0.82	0.81
Precision	1	0.80	0.86
Recall	0.96	1	1
F-Score	1	0.89	0.92
TPR	0.96	1	1
FPR	0	0.30	0.13

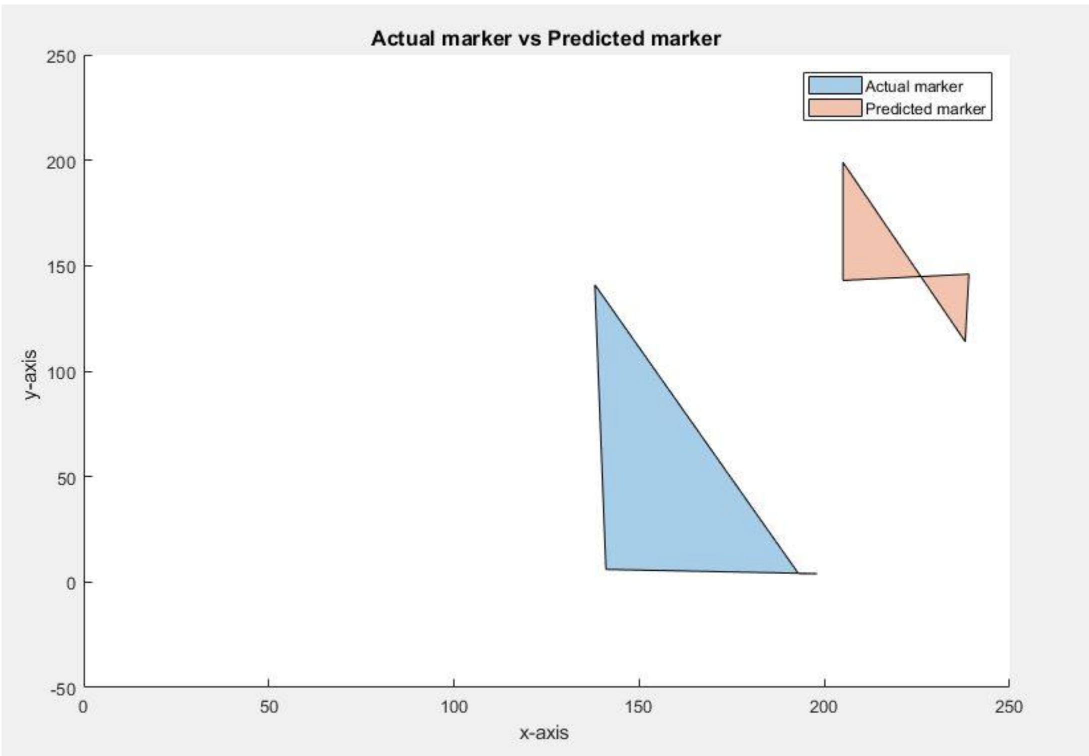

FIGURE 4.18: *Poor result of matched marker with the IoU score=0*

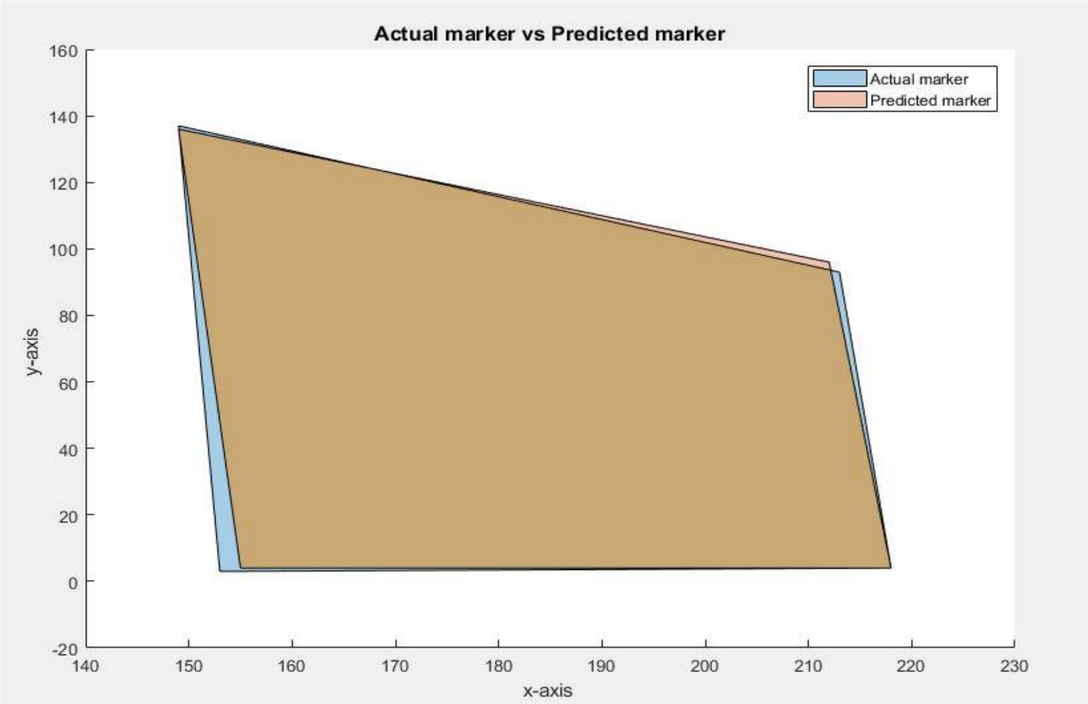

FIGURE 4.19: *Good result of matched marker with the IoU score=0.96*

Figure 4.18 and 4.19 shows the outcomes of the intersected regions of the predicted markers with the actual markers within the frames. Poor results shows that the actual marker that is detected by the proposed algorithm is the incorrectly detected marker whereas the good marker result shows that the marker has been detected successfully in the particular frame.

TABLE 4.17: *Comparison results of Frame detection rate of proposed work with existing works*

Method	**Real frames**	**Corrected recognition frames**	**False recognition frames**	**Detection rate**
Fu et al.[212]	856	845	92	89.14%
Lee et al.[213]	856	-	-	95.4%
Ikeda et al. [214]	856	779	17	89.23%
Chen et al. [182]	856	841	13	96.78%
Proposed method	856	840	-	96.6%

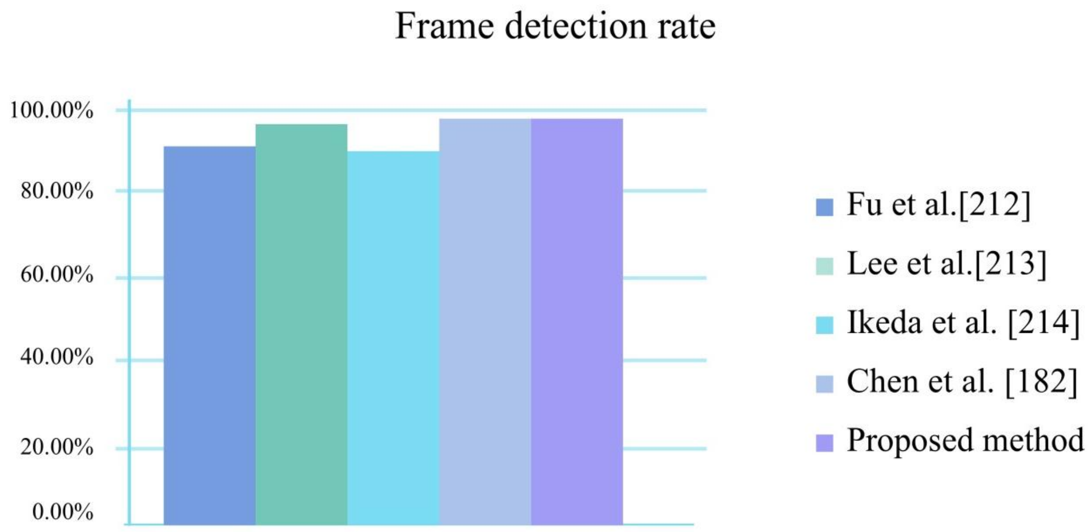

FIGURE 4.20: *Comparative analysis with state of the art*

Table 4.17 and Figure 4.20 shows the comparative analysis of the proposed algorithm with the existing approaches. Figure 4.20 shows that the in [11], the frame detection rate was 89.14%. In [17], the frame detection rate was 95.44%. In [28], the frame detection rate was 89.23%. In [182], the frame detection rate was 96.78%. In proposed work, the frame detection rate was 96.6.%.

Worst case complexity: $O((m*n)^2N)$ where m*n is the size of the frames and N is the number of frames in the real time video.

Best case complexity: O(mn*N log(m*n)) where m*n is the size of the frames and N is the number of frames in the real time video.

Worst case is when the system is not able to locate the marker during the run in any repository i.e. primary, secondary and tertiary. Whereas the best case would be when the system will be able to find the primary marker in the first attempt.

4.5 CONCLUSION

An approach for marker detection has been prescribed, and it aims to help the autonomous system to navigate safely in an unknown environment. A unique concept of categorizing markers into primary, secondary, and tertiary markers will result in better navigation, less computational time, and fast matching process. The three levels of triviality have been removed step by step to make it an easier and an effective categorization process for the detection of the markers.

CHAPTER – 5

BUILDING A NEW IMAGE REPOSITORY FOR AUGMENTED REALITY BASED NAVIGATION

In this objective, the concept of self-created and self-refined repository for the augmented reality based autonomous navigation system has been proposed. This repository comprises of the human-understandable markers detected during the bare run for an unknown environment. The system will then navigate autonomously with the help of these markers later. The novelty of the work is in the extraction of three types of markers that will aid navigation; main markers, supportive markers, and backup markers. After storing these markers, this self-created repository refines itself during the next runs on the basis of marker usability.

5.1 INTRODUCTION

For autonomous navigation, the system has to commute from one place to another without any human intervention. This automated navigation can take place with the support of certain navigation instructions. These instructions are prescribed in various forms, such as audio, video, images, etc. For vision-based instructions, the augmented reality technology is currently being used. Unlike virtual reality, the augmented reality technology superimposes the virtual information on the real word environment [1]. This virtual information can be helpful for the autonomous navigation system. This information may include information about the current location (markers, landmarks, traffic signboards, obstacles, warnings, etc.).

In the 1980s, the main focus was on complex electro mechanical systems. The emphasis was on sensors, actuators, and electronic control units. The starting point of the automation or the entry-level of the automation was focused on the ABS (automated braking system) and ESC (electronic stability control units). This era was related to partially automation [2].

Then generation by generation, new actuator, sensors, and interfaces were introduced. The significant change in this field was introduced by Defense Advanced Research Projects Agency (DARPA) two challenges (in 2004-2007). Here, the main focus was on computation and information processing rather than vehicle underlying dynamics and control system.

However, with no relevant environmental sensing in these systems, they all sit at the lower control level of driving. The system still requires the driver to supervise the navigation systems. So, at this level, there was no sign of fully automation. Now, in today’s era, the main emphasis is on intelligent vehicles. Various companies like Waymo (Google), Uber, Apple are working on self-driving car technologies. Car companies like Volkswagen, Ford, Tesla, BMW are also incorporating these self-driving features [3][4]. Wayray is the only developer which is integrating augmented reality into cars today.

In this work, the concept of self-created and self-refined marker-based image repository has been proposed for any unknown environment. The human-understandable markers are stored in the repository along with the navigation instructions to aid the navigation process.

5.2 METHODOLOGY AND IMPLEMENTATION

5.2.1 Self-created/ Auto-generated marker repository

Human understandable markers lead to the navigation purpose. Markers can be landmarks, hoardings, traffic signs, bridges, etc. A useful marker must have the following characteristics:

- Easy to locate
- Visible from distance
- Larger in size
- Not changing position frequently.

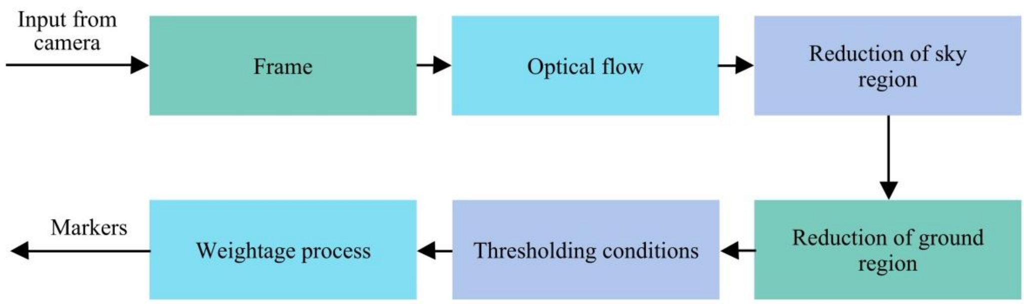

FIGURE 5.1: *Block Diagram of the detection of markers.*

The detection process of the marker is shown in the above Figure 5.1. During the bare run, the system navigates from source to destination for a specific path (i.e., from A to B) for the first time. It automatically detects the markers from the real-world environment. The markers are stationary and prominent. With the help of the optical flow approach, reduction of sky region, and reduction of the ground region, the non-trivial objects are detected, and then they undergo some threshold conditions for further process. Then, these potential markers undergo weightage process for categorization. Therefore, based on immovability, size, area along with the weightage process, these detected and categorized markers are stored in their respective repositories.

These markers get categorized into three types of markers as shown in Figure 5.2.

- Main Markers
- Supportive Markers
- Backup Markers

Main Markers: These are also known as primary markers. These are the markers that are solely responsible for the navigation task.

Supportive Markers: These are also known as secondary markers. These are the markers those are present in the system and available only when required. When main markers are not present due to any reason (i.e., visibility conditions, marker lost, etc.), then these markers come into play.

Backup Markers: These are also known as the tertiary markers. These will rarely get used but are present to deal with the case of emergencies.

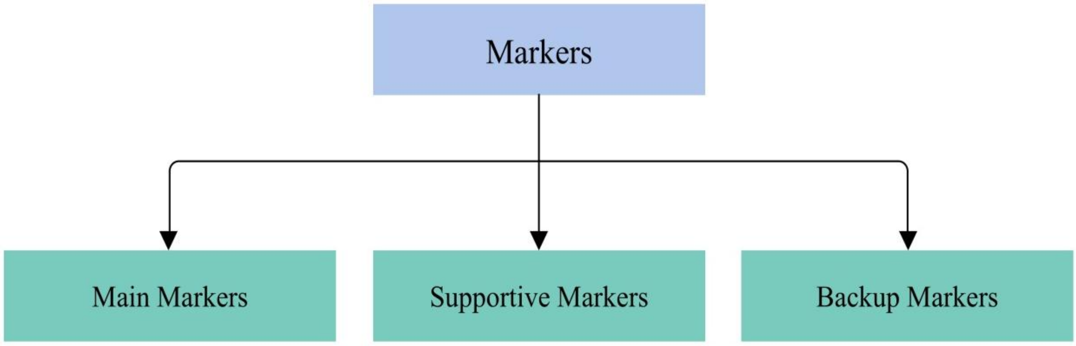

FIGURE 5.2: *Types of markers*

The system distinguishes between the stationary and movable detected objects present in a particular video frame with the help of the optical flow approach. By removing the sky region and ground region, the trivial objects from the frame are eliminated. Then, the detected objects get checked for the prominence factor. Based on the weightage process (size and area), the markers are categorized as main, supportive, and backup markers from the set of non-trivial markers.

TABLE 5.1: *Categorization of markers based on Weightage Process*

S. No.	X	Y	Area	Region
1	257.43	40.92	429	71
2	266.55	128.12	600	72
3	**311.13**	**125.69**	**2680**	**75**
4	**280.12**	**48.924**	**2693**	**70**
5	**203.54**	**58.28**	**6431**	**39**
6	64.33	50.60	12197	1
7	164.28	217.76	44941	2

Table 5.1 shows the weightage process used for the categorization of markers as main, supportive, and backup markers. The 5th row corresponds to the main marker having the highest area of the marker, 4th row represents the supportive marker having the second highest area value, and the 3rd row represents the backup marker, having the lowest of the primary as well as secondary marker, in a particular video frame. This way, the three types of markers are detected and categorized for any unknown environment. These markers are required for validating the path in the

future runs. If one marker is lost for any reason, the rest of the markers (supportive or tertiary) validate the path.

5.2.2 Labeling of repository

Once the markers are detected and categorized from the real time environment, these markers are labeled with the metadata to assist the augmented reality (AR) based navigation system. This metadata consists of the required information for the navigation process and is also helpful in performing the auto-refinement process of the repositories. These detected markers are stored on the cloud infrastructure. NOSQL database is used to store the information corresponding to these markers which includes information like id, url, title, hit-count, run-count and location etc. This information can also be used by AR based applications for a better navigation experience. For an example, in an AR based navigation system, information about the detected markers is augmented on the windshield of the vehicle in the form of pop-up bubbles for user's assistance. If the user wants to get more information about the detected marker, then, he can get the detailed information for that marker by clicking on the pop-up bubble overlaid on the windshield of the vehicle. Fig. 5.3 and Fig. 5.4 shows the labeled markers that are also termed as AR tags.

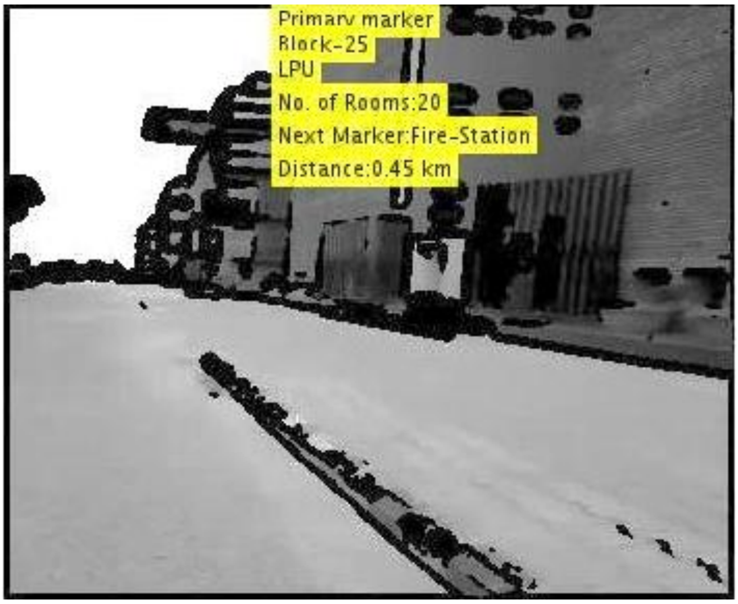

FIGURE 5.3: *AR tag after identifying the marker for a particular frame*

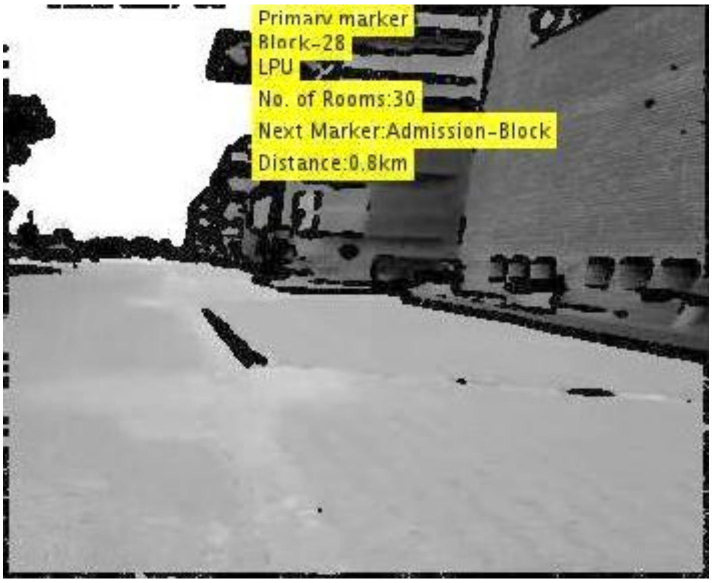

FIGURE 5.4: *AR tag after identifying the marker for next frames*

5.2.3 Auto refinement of repository

Based upon certain parameters, the system automatically checks if the marker in their respective repositories is useful or not. The futile markers automatically get deleted and get stored in the temporary marker repository for a limited amount of time.

If the particular marker is not helping in the navigation task after a few runs, the corresponding marker automatically gets diminished from the respective repository based on the following equation.

$$RF = \frac{H}{R} \times 100 \qquad \dots (5.1)$$

If RF < N

Perform auto-deletion.

Here, No. of hit count= H

No. of run count= R

N, the tolerance factor

RF, the refinement factor is the factor that concludes the formula for the auto-deletion of the trivial markers.

When the value of 'RF', refinement factor is less than the 'N', tolerance factor i.e. the occurrence of the particular marker present in the repository in the real-world environment is very less during the runs, then this implies that the particular marker is of no use. Therefore, it should be deleted from the repository.

Thus, the repository automatically refines itself after a few runs on the basis of its hit count. This leads to the optimization of the repository. Repository can also be termed as an intelligent repository that will help the navigation task to get uncomplicated and effective.

5.3 RESULTS AND DISCUSSION

The proposed approach was applied to a particular video of an unknown path (i.e., from A to B). The main, supportive, and backup markers were then detected, as shown below in Figure 5.5.

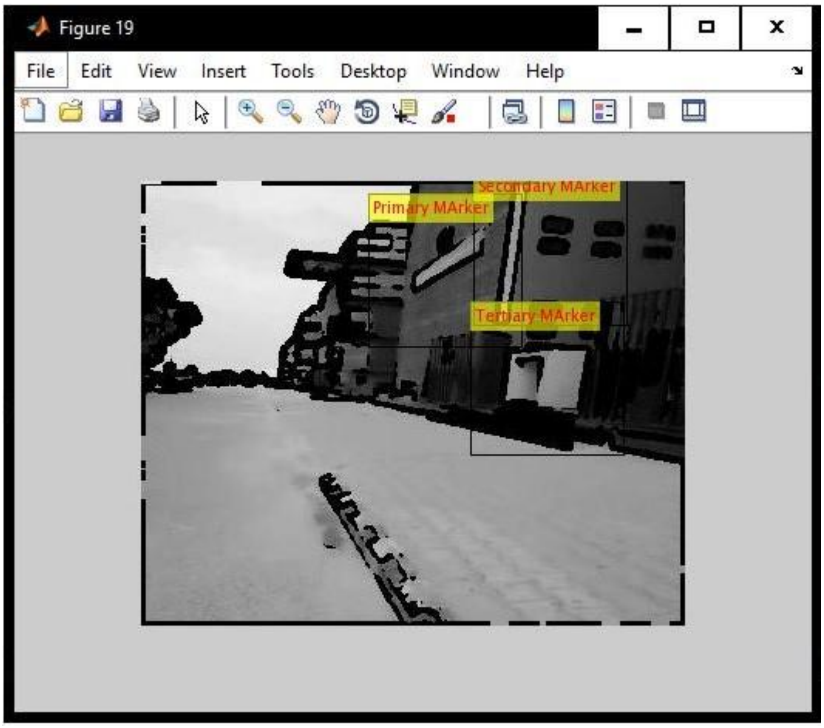

FIGURE 5.5: *Categorized markers: Primary, Secondary as well as Tertiary markers*

These markers, after the categorization process, were stored in their respective repositories. The main markers got stored in the primary repository; supportive markers got stored under the secondary repository. Similarly, the backup markers got stored in the tertiary repository. The primary repository is shown below in Figure 5.6.

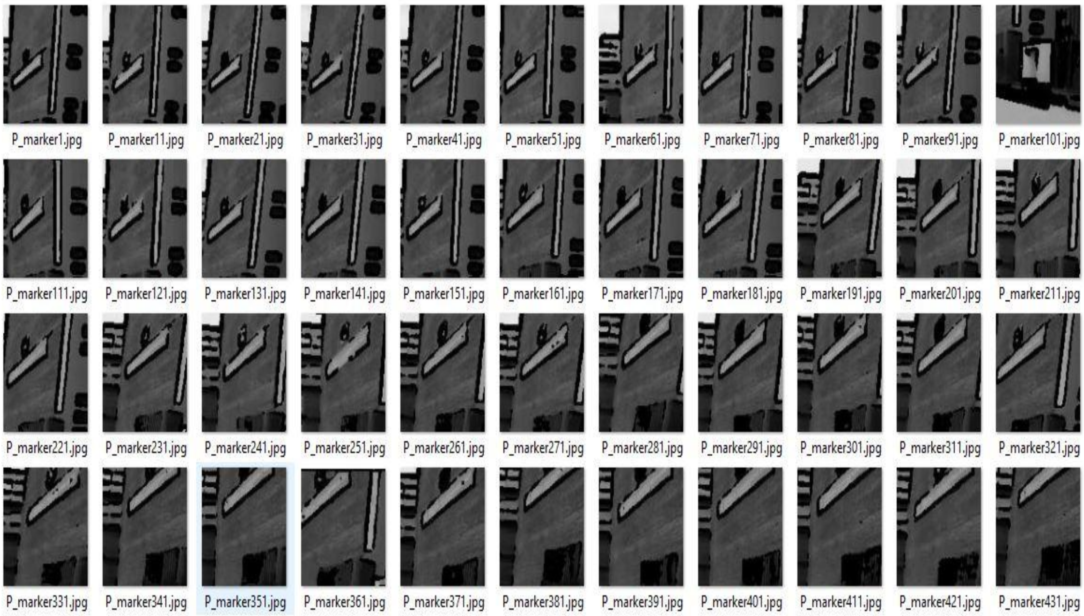

FIGURE 5.6: *Primary repository consisting of all main markers after bare run*

The markers stored during the bare run got auto refined after the next runs. The markers that were not getting used after few runs got deleted from their respective repositories. Figure 5.7 shows the refined primary repository after a particular set of runs.

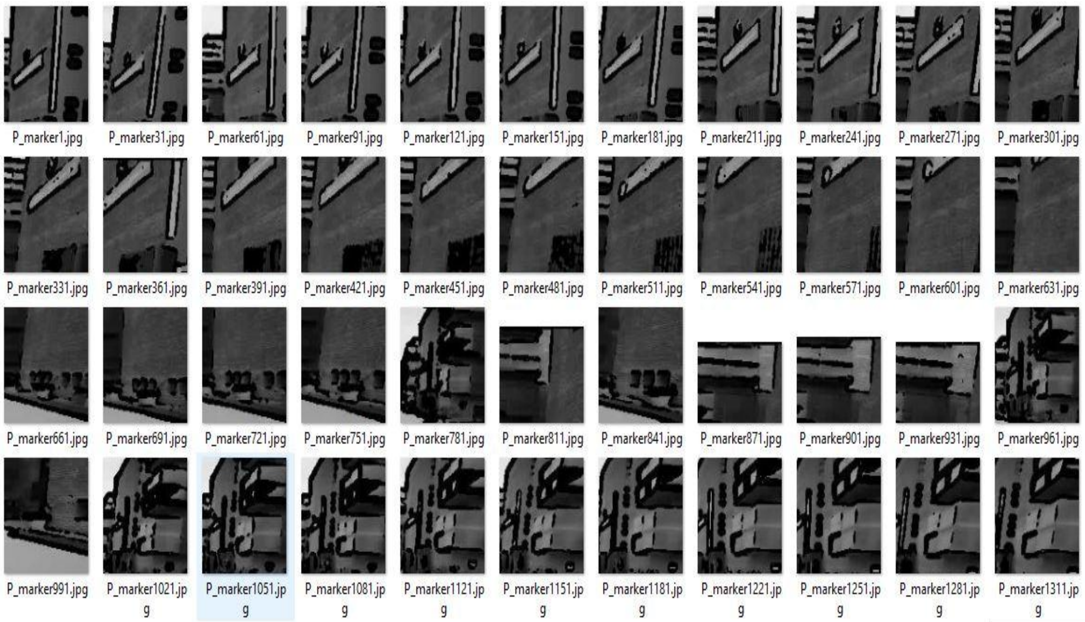

FIGURE 5.7: *Auto-refined primary repository*

Figure 5.5 and Figure 5.7 comprises of the primary markers detected and stored in the bare run to be refined during the next runs, respectively. In Figure 5.7, there is a reduction in the number of primary markers, and this is achieved by auto-deleting the unused markers during the next runs. Secondary and tertiary markers underwent the same process. Thus, self-created repositories got auto-refined that benefitted the repositories in not getting bulked in size. This reduced the space complexity and made the matching process faster for the next runs.

5.4 CONCLUSION

This chapter abridges the concept of self-created and self-refined marker repository. Without any human intervention, this self-created repository will store the human-understandable markers along with their navigation information. This navigation information will help to achieve a better navigation experience with the help of AR technology. Novelty of this repository lies in its three types of markers that are main markers, supportive markers, and backup markers and the labeled information of these markers. This repository will refine itself by removing the meaningless markers in further runs based on a few conditions. The focus of the future work will be on the addition of new markers if introduced on the same path.

CHAPTER – 6

UTILIZATION OF IMAGES FOR OBSTACLE DETECTION AND BOOKMARK IDENTIFICATION

6.1 INTRODUCTION

Real-time obstacle detection plays a major role in the autonomous navigation field. While navigating safely, the system has to detect various upcoming obstacles in the way to reduce collisions or road accidents. This is one of the indispensable works which has to be taken care of while the navigation process takes place. Obstacles can be dynamic and stationary, as well. There are various technologies available in the previous works to detect both types of obstacles. Here, in this work, the stationary obstacle detection algorithm has been proposed. According to various studies, an efficient obstacle detection system must be potential of satisfying the following conditions:

1) Detection of obstacles at the correct time.
2) Detection of correct obstacles.

There are various approaches used for the detection of obstacles. Mostly, Vision-based and sensors based approaches are preferred for this purpose. This work presents a vision-based obstacle detection algorithm in which the stationary obstacles are detected from the real world environment on the basis of visual and geometric parameters. Videos of the real-time scene are captured using a 12 MP camera with the resolution of the 4000x3000 pixels and are recorded at the 1920 x 1080, 30fps. The results are generated in the MATLAB tool.

6.2 METHODOLOGY AND IMPLEMENTATION

The proposed system detects the stationary obstacles in the real-time environment by removing the step by step triviality. The detection process is explained with the help of figure 6.1, given below:

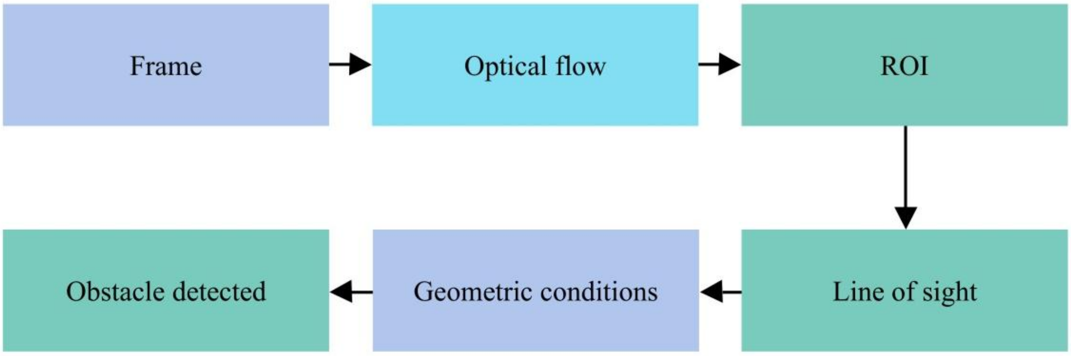

FIGURE 6.1: *Proposed architecture of the obstacles detection*

Figure 6.1 shows the architecture of the proposed algorithm used for the research presented in this work. Input from the camera was taken in the form of a video, and frames were generated from that video at a particular set of intervals. Movable and stationary objects were distinguished with the help of the optical flow technique applied to each frame. Stationary objects were considered for further processing. Then, the regions of interests (ROI) from that particular frame containing the stationary objects were selected, and their required properties like area value and centroid were fetched. All stationary objects were then checked to see that whether they fall within the line of sight or not (line of sight is that area in the real-time scene in which if anything is encountered, would be considered as the potential obstacle and the geomertically, its area is based on the height and the width of the vehicle). The potential stationary objects were then passed through predesigned geometrical conditions to qualify them as the obstacles. This way, the real-time stationary obstacles were detected for the autonomous navigation system. The detailed detection process is elaborated in Figure 6.2.

Figure 6.2 elaborates on the detailed process of detection of the stationary obstacle from the real-time video. First, the video was made for a specific path, i.e., from A to B; the frames were generated from the video. Secondly, the movable and non-movable objects were set separated with the help of the optical flow approach. Finally, the frames with stationary objects were converted into black and white frames to work in relation to some similarities of the region for proper identification.

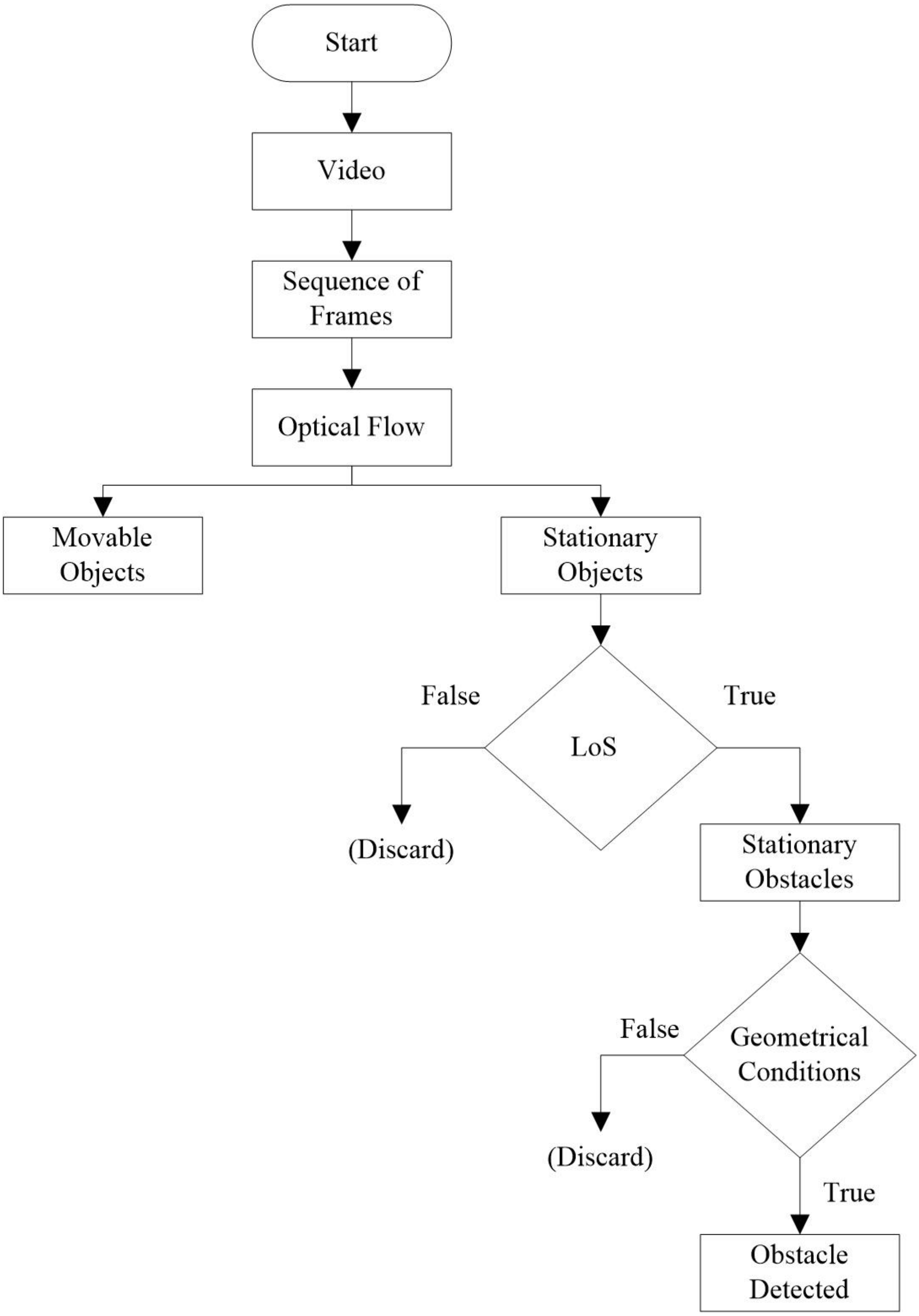

FIGURE 6.2: *Flowchart for the detection of stationary obstacle*

The stationary objects were then checked whether they lied within the line of sight (LOS) frame or not. As the system was navigating on the center of the road with a constant speed, so whenever any object was encountered in the center or the LoS, the system had to do some processing to check various parameters of that object to verify if the system could cross the encountered object or it should stop. For this, the objects present within that LoS frame, undergone certain geometric conditions. The steps followed for the detection of the obstacle process are as follows:

1) Objects were categorized into movable and stationary objects.
2) Out of those, stationary objects which were fulfilling the line of sight (LoS) parameter, were considered as the potential stationary obstacles.
3) In the final stage, the potential stationary obstacles had to pacify certain geometrical conditions further to be considered as the final stationary obstacle.

The centroid and the coordinates of the corners of all the objects were compared with the coordinates of the line of sight (LoS) frame to confirm if the selected stationary objects after applying the optical flow approach lie in the line of sight or not. There can be an uncountable number of stationary objects present in a particular scene.

The following example elaborates the above concept from Figure 6.2, as Let, there be ‘6’ objects present in a particular video frame. For some of the objects, their centroid lies within the LoS frame, but in some cases, the centroid lies outside the line of sight frame. So, to deal with both of the cases, the following example in Figure 6.3 has shown the process. Object 1 lies outside the line of sight; object 2, 3,4, and 5 partially lie in the line of sight, whereas Object 6 lies entirely within the line of sight.

Here,

XL and YL are the lower limits of the line of sight frame.

XH and YH are the upper limits of the line of sight frame.

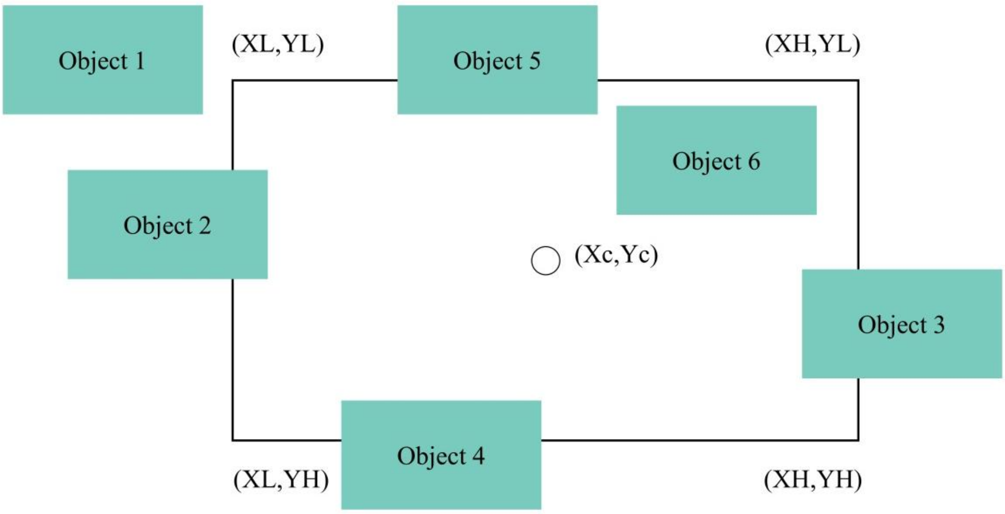

FIGURE 6.3: *Example showing the various cases of the object present within or outside the LoS*

For an object to lie within this line of sight, the object's centroid or its extreme coordinates must lie within the extreme coordinates of the line of the sight frame.

In Figure 6.3, six objects with their three broad categories listed as:

- Object lying entirely within the LoS
- Object lying partially within the LoS
- Object lying outside the LoS

Explanation for one of the above cases, where,

(XL, YL), the top-left point of LoS
(XL, YH), the bottom-left point of LoS
(XH, YL), top-right point of LoS
(XH, YH), bottom-right point of LoS.

To check whether object 1 lies within the LoS or not.

Keeping track of the centroid of the object, if it lies within any part of the line of sight frame, then it must be considered as an obstacle, otherwise, all the corners of the object need to be checked to see whether they lie in the LoS or not. Now, there are two cases: At first, only the centroid of the object gets checked against the line of sight frame, whereas in the second case, all the four corners of the particular object are checked to qualify as an obstacle. The below example in Figure 6.4 explains the same. In this illustration, the detected stationary object is not present in the frame. Therefore, to check its presence in the LoS, there can be two cases.

(Xc,Yc), centre of the line of sight frame.

***Case 1:* If the object's centroid lies in the LoS.**

Point x (Xo, Yo) lies in the LoS.

(Xo, Yo) lies within the XL, YL, XH, YH.

$$XL \leq X_o \leq XH \quad \ldots (6.1)$$

$$YL \leq Y_o \leq YH \quad \ldots (6.2)$$

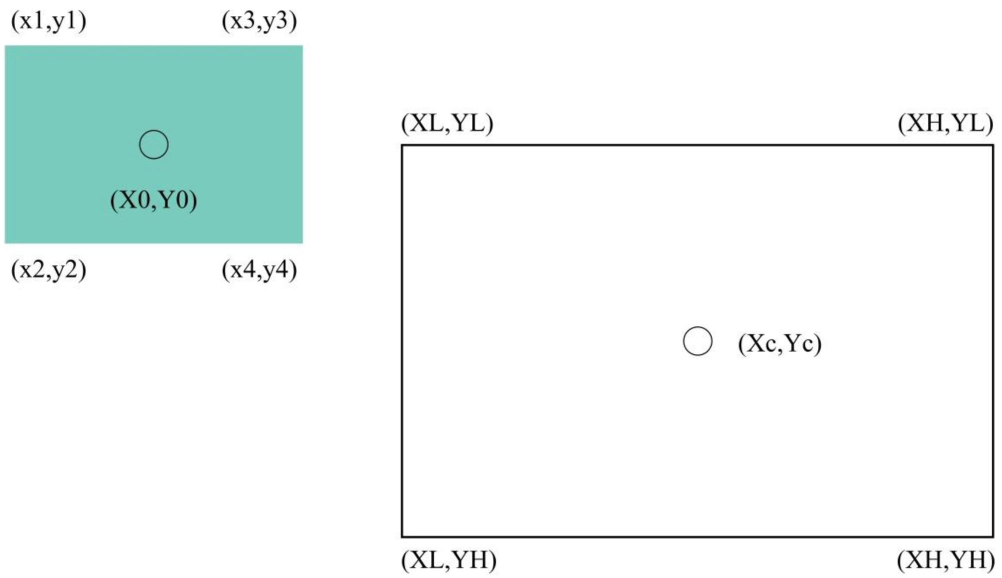

FIGURE 6.4: *Example to show the object lies outside the LoS*

***Case 2:* If the object's centroid does not lie in the LoS.**

Then the lower and upper limits of the object's coordinates (x1,y1), (x2,y2), (x3,y3), and (x4,y4) get checked if they lie within the LoS.

For every i = 1:4

$$XL \leq xi \leq XH \quad \dots (6.3)$$

$$YL \leq yi \leq YH \quad \dots (6.4)$$

Table 6.1 explains the above example (Figure 6.4) in the tabular form where the 'Object 1' is checked if it is qualified as an obstacle or not. If the result is true 'T' against all the columns, then it must be an obstacle; otherwise, it needs to be discarded, which means the result is false 'F', which means, lets say, for an object 1, neither of its centroid nor its any corner point lies in the LoS, so this has not been considered as an obstacle for the navigation process. For an object to be qualify as an obstacle, it has mandatory either of its corners or it centroid must lie within the LoS of the vehicle.

If the centroid of any object lies within the LoS, then the particular object would be considered as the potential obstacle, but if centroid does not lie in the LoS, then the corner points of the object have to be checked for their presence in the LoS frame. Table 6.1 shows the presence of the object in the line of sight frame to be qualified as a potential obstacle.

TABLE 6.1: *Validation Table for the stationary objects in LoS*

Object	**1**	**2**	**3**	**4**	**5**	**6**
Centroid (Xo,Yo)	F	F	F	F	T	F
Point 1 (X1,Y1)	F	F	T	T	T	F
Point 2 (X2,Y2)	F	F	F	T	T	T
Point 3 (X3,Y3)	F	T	T	F	T	F
Point 4 (X4,Y4)	F	T	F	F	T	T
Result	F	T	T	T	T	T

After the system detects a potential obstacle, it has further a qualification corner to be detected as an obstacle in the final round. Height and Width of the detected object are checked against the surface height of the vehicle. For this, the detected object must appease some geometric conditions. The cases that define the presence of an obstacle based on certain geometric parameters are as follows:

Case 1*:* If the object has more width and less height than the width and surface clearance of the vehicle, respectively, the detected object is not considered as an obstacle.

Case 2*:* If the object has less height and width than the surface clearance and width of the vehicle, then it is not an obstacle.

Case 3: If the object has more height than the surface clearance of the vehicle and less width than the width of the vehicle, then it is regarded as an obstacle.

Case 4*:* If the object has more height and width than the surface clearance and width of the vehicle, respectively, then it is an obstacle.

Let,

H be the height of the object detected in the LoS

S_v be the surface clearance of the vehicle

W be the width of the object

W_v be the width of the vehicle

LoS be the Line of Sight

In Figure 6.5, pictorial representation of the above-discussed cases has been presented. The system checks various parameters related to the height and width of the encountered potential obstacle against the surface height and width of the vehicle for the final round of the detection process.

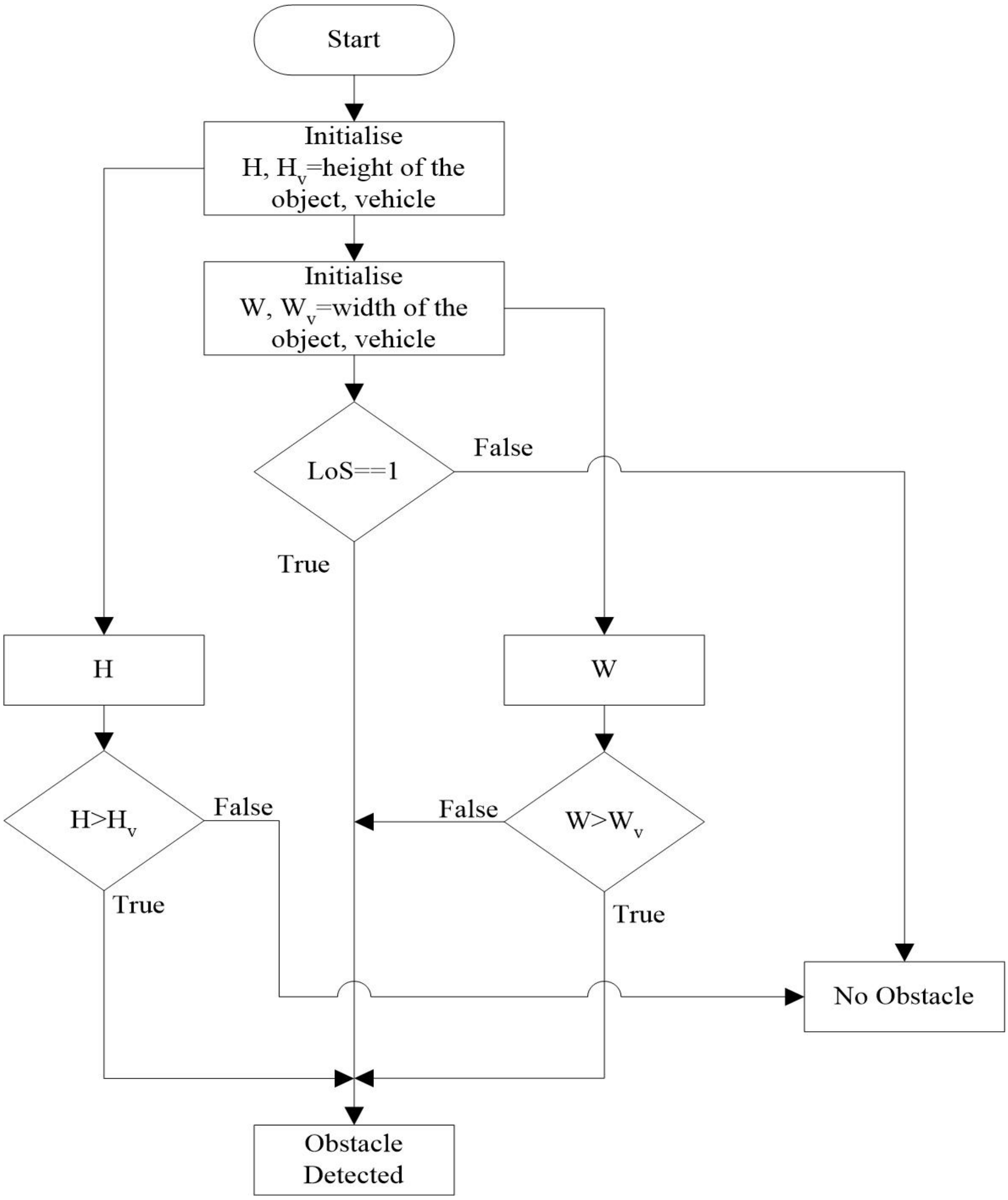

FIGURE 6.5: *Flowchart of the Obstacle Detection Process*

The above discussed geometric properties are in tabular form.

Let us assume,

W_{object} be the width of the object.

W_V be the width of the vehicle.

H_{object} be the height of the object.

SH be the surface height of the object

Table 6.2: *Obstacle Detection Cases*

W_{object}	H_{object}	**Obstacle**
$>W_v$	$>S_v$	T
$> W_v$	$< S_v$	F
$< W_v$	$> S_v$	T
$< W_v$	$< S_v$	F

In Table 6.2, there are only two true cases of the obstacle detection process, when the 'H_{object}' of the detected object is greater than the surface height of the vehicle.

The surface height discussed in the detection process is also known as Ground Clearance of the vehicle that is also known as Ride Height. Ground Clearance is the amount of space between the base of an automobile tire and the lowest point (axle). The minimum and maximum value of the ground clearance of the various car models is shown in Figure 6.6.

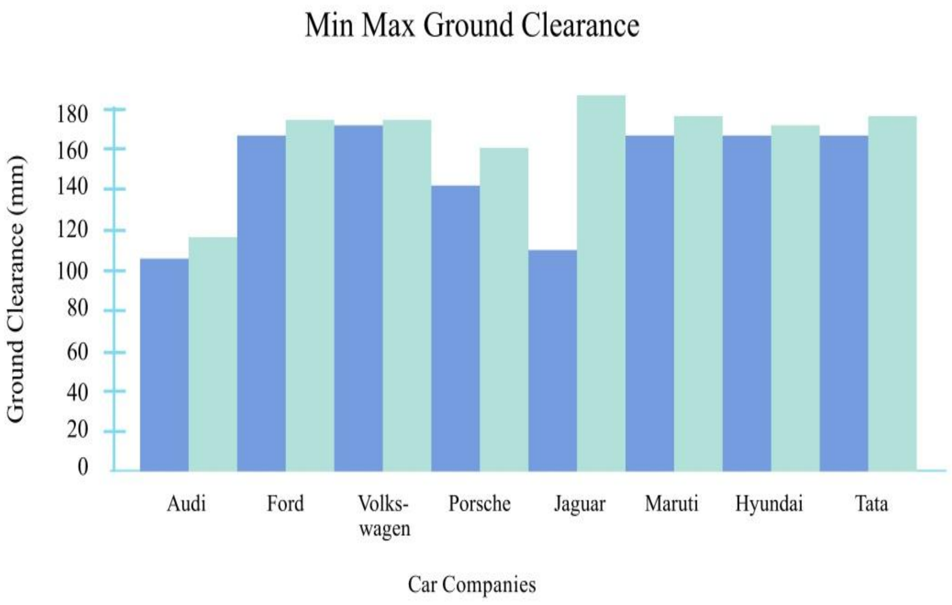

FIGURE 6.6: *Min Max value of Ground Clearance*

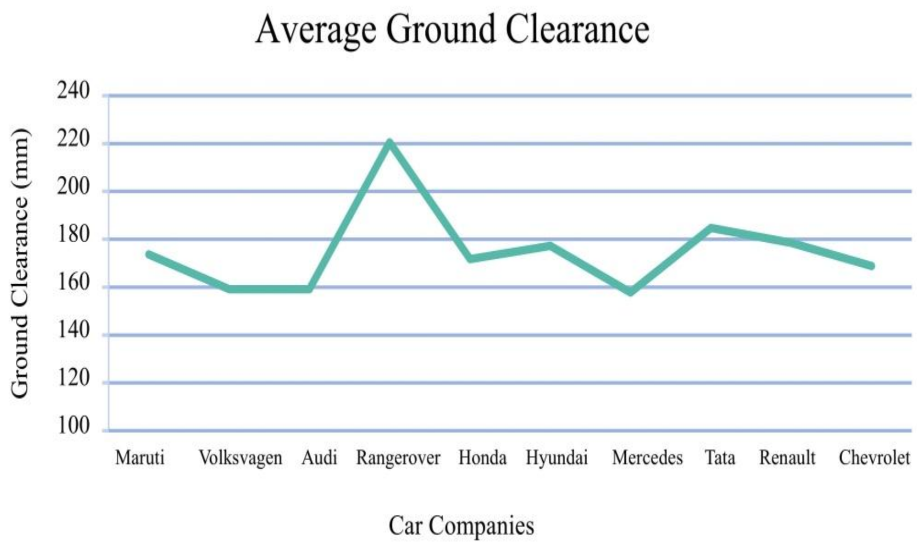

FIGURE 6.7: *Average Ground Clearance of Cars*

The above statistics unveil the average ground clearance value considered for the surface height of the vehicle. According to the statistics showed in Figure 6.7, clearance value for day-to-day car models is 160-168 m.

Conversion of world co-ordinate system to pixel format:

As the ground clearance values are in the world coordinate system, and all the other parameters are in the pixel format. So, to evaluate further work, camera calibration has to be performed. The world coordinate systems have been mapped with the pixel format using the below equation.

$$1\ cm = k \times\ pi \qquad \dots (6.3)$$

As the system is moving in the real-time environment with a constant speed, therefore, in order to calculate the surface height in terms of pixel format, it has to calculate the pixel height of the known object whose surface height in terms of the world coordinate system is already known. Then, by taking a constant distance and speed, the system calculates the pixel height of the detected obstacle with respect to the known object. This is also known as the camera calibration process in which the world coordinates are mapped on the two-dimensional plane. The average surface height of vehicle = 160mm = 16cm

At a particular distance and at a constant speed, the height of the known object is 60cm in world coordinate system and its pixel height is 20pi. The unit conversion has been performed below:

60cm=20pi

1cm=20/60=1/3;

*16cm=(1/3)*16=5.33pi.*

$\therefore$ *1cm=5.33pi.*

Therefore, the final obstacles are successfully detected after satisfying three parameters: movability, line of sight, and geometric conditions. No prebuilt dataset is used in this process.

6.3 RESULTS AND DISCUSSION

The proposed method has been applied to the video frames to navigate safely in an unknown environment without using the pre-built database in MATLAB. Figures 6.8 to 6.11 show the outcomes formulated through the application.

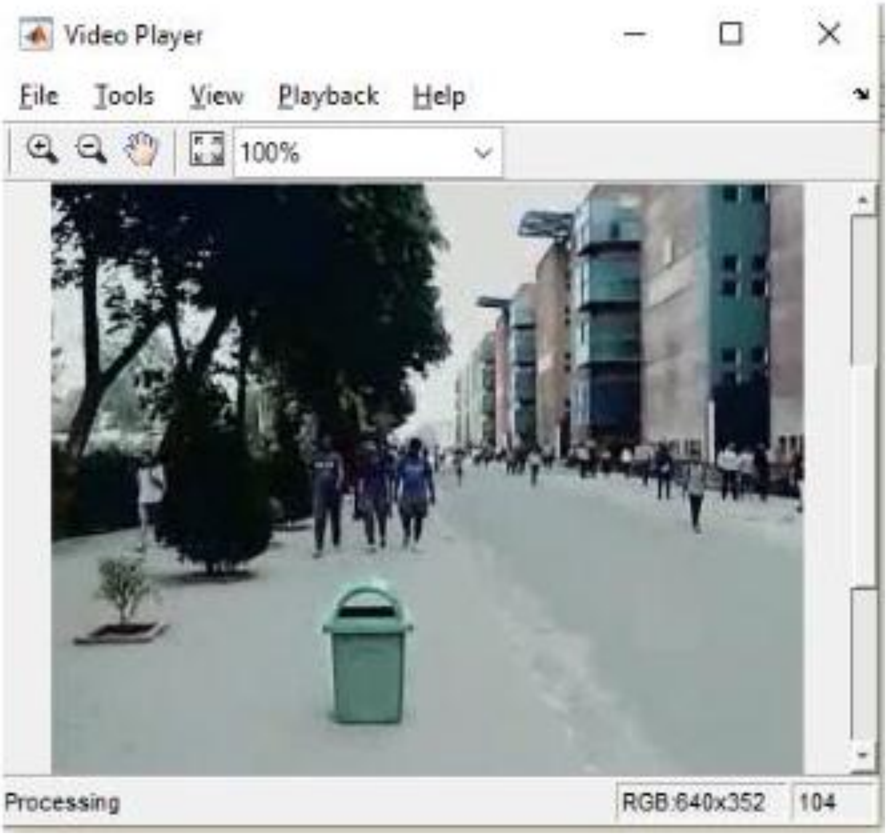

FIGURE 6.8: *Real-time video captured of a path*

Figure 6.8 shows the real-time video of a specific path. The video is captured in the real-time scene, including some obstacles present in the environment. The optical flow technique is applied to the frames extracted from the video to differentiate between movable and stationary objects.

FIGURE 6.9: Detected stationary objects in the real-time path

Figure 6.9 shows the results of the detection of all stationary objects after the optical flow approach. The stationary objects are labeled here. The black regions indicate the movable objects while the remaining objects come under the stationary regions. Once the system is left with stationary objects, it looks for the line of sight frame to check whether the candidate object lies in the LoS frame or not. The line of sight frame is highlighted in the below Figure 6.10.

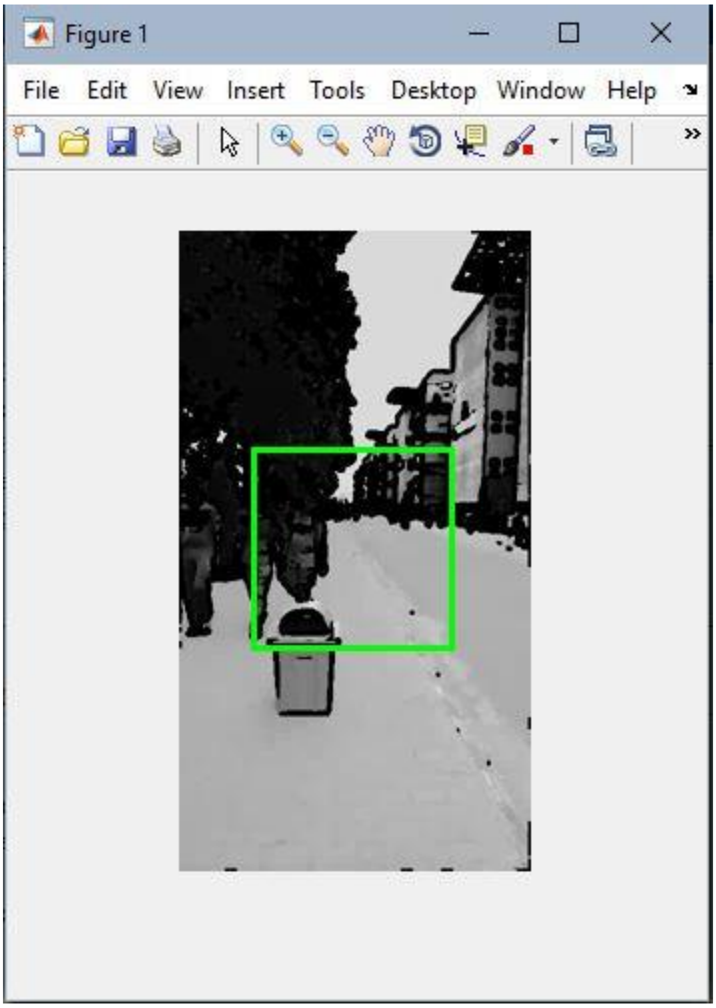

FIGURE 6.10: *Line of sight region in the particular frame*

Figure 6.10 shows the line of sight area in a particular frame. The stationary object residing within this area is considered a potential obstacle. The system looks for the centroid and the corner points of every candidate objects to check for this LoS condition. There can be several objects present in this area, but before considering them as an obstacle, the candidate objects have to fulfill some geometrical conditions discussed in the above section. The resultant objects are then detected as stationary obstacles.

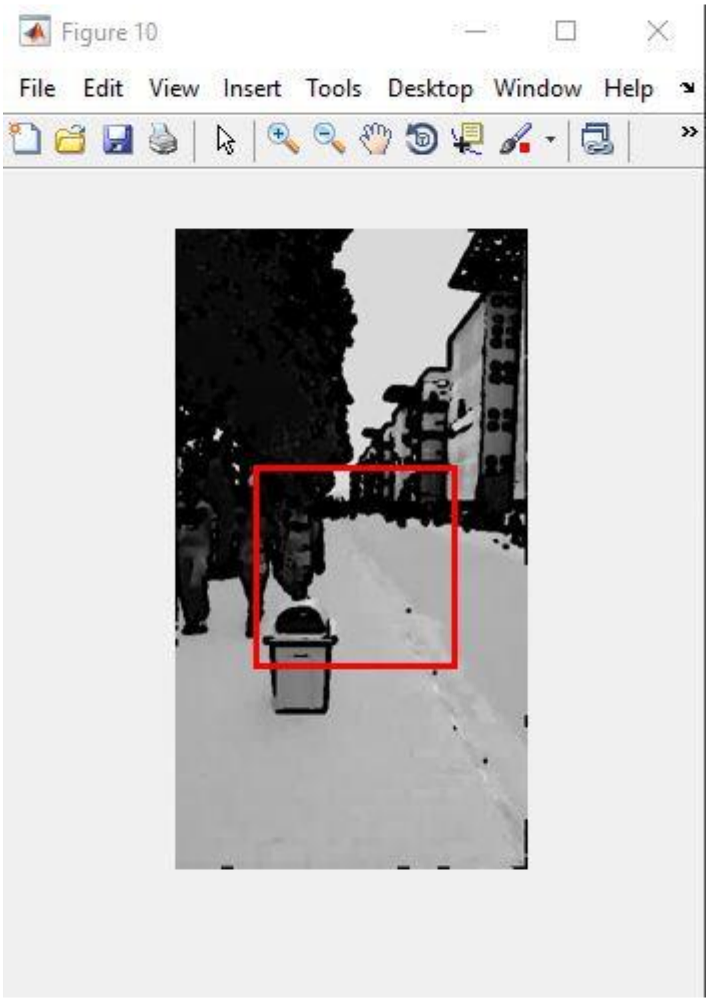

FIGURE 6.11: *Detection of final stationary obstacle*

Figure 6.11 depicts the final stationary obstacle after fulfilling certain geometric conditions. The results are generated in MATLAB 2017b version.

6.3.1 Comparative analysis

The comparison of the proposed approach with the existing approaches has been done in this section. Obstacle detection is a tedious task that depends upon various parameters. However, there are certain novel aspects for describing the efficacy of the proposed work in the automotive industry. Especially, four arduous studies namely, Anvaripour et al. [194], Lee et al. [193], Prasad et al. [154], Song et al. [188] are pondered for comparison purposes. These studies are compared on the basis of different parameters in Table 6.3.

- **Applicability Domain**

An overview with reverence to the stationary obstacle detection for which a specific research has been carried out is discussed using this parameter. In contrast, it elucidates the applicability domain of the proposed work.

- **Major Contribution**

This parameter gives the thorough information about the major contributions made in the proposed field till now. This means, it defines the significant characteristic of the proposed work in automotive industry.

- **Target Achieved**

This parameter explains the end results achieved after implication of the proposed methodology. In other words, it describes the type of output generated after the execution of the proposed algorithm.

- **Baseline Sensing Technology Used**

The basic sensors significantly needed for the implementation of the proposed research work have been stated in this parameter. Different works with their particular specification have been explained here.

- **Detection Model Used**

This parameter gives information about the particular model of detection that has been involved in various studies for successful determination of vulnerability.

- **Knowledge based**

This parameter describes about the prior knowledge required for the execution of the algorithms introduced in the proposed works. However, some systems use this prior knowledge in their learning phase.

- **Feature Extraction Mechanism**

Data acquisition requires an effective technique of extraction of features for overall system efficiency. The comparison of the proposed model with other related studies focused on the features extraction process has been discussed.

- **Data Mining Technique**

Obstacle detection process involves constant extraction of data from the real-world and the repository. Subsequently, analysing the proposed model on the basis of data mining techniques engaged for the extraction of data becomes vital.

- **Intelligent Data Storage**

Data Storage indicates the storage requirements of the results achieved after implementing the proposed methodology. This parameter describes that how and where the output data will be stored for the upcoming use.

- **Assumption**

Assumptions made while implementing the proposed methodology have been discussed and the conclusions of the research are based on these assumptions.

- **Decision Making Model**

This parameter discusses the type of output generated after implementing the proposed work which leads to the decision making process. This means, it describes the expected input requirements of the decision making model.

- **Heterogeneous Datasets**

This parameter is associated with the form of data that was sensed in a particular study. Obstacle detection is a complex parameter; therefore it is necessary to have efficient decision making to evaluate different parameters.

TABLE 6.3: *Comparative analysis of different models with proposed model*

Parameters / References	**Anvaripour et al. (2015) [194]**	**Lee et al. (2016) [193]**	**Prasad et al. (2017) [150]**	**Song et al. (2018) [188]**	**Arvind et al. (2019) [163]**	**Proposed Model**
Applicability Domain	Outdoor environments	Indoor environments	Maritime environment	Outdoor environments	Dynamic environment	Real-time Outdoor environment
Major Contribution	Detection of objects using local shape information	Inverse perspective mapping method is used	Object detection and tracking techniques	Real-time obstacle detection method for collision warning for vehicle active safety system	Obstacle detection	Detection of real-time on-road stationary obstacles using a single camera
Target Achieved	Object Detection	Stationary Objects	-	-	Obstacles	Detection of stationary obstacle based on ground clearance and width
Baseline Sensor Used	Camera	Forward viewing mono-camera and wheel odometry and gyroscope	Electro-optical sensor	Stereo camera and mmw radar	Ultrasonic radar	Monocular Camera
Detection Model Used	Shape-based model	Sensor-based model	No	No	Reinforcement learning	Proposed model

Contd. …

Parameters / References	**Anvaripour et al. (2015) [194]**	**Lee et al. (2016) [193]**	**Prasad et al. (2017) [150]**	**Song et al. (2018) [188]**	**Arvind et al. (2019) [163]**	**Proposed Model**
Knowledge-based	Prior training of dataset	-	-	-		No prior training required
Feature Extraction Mechanism	Boundary fragment extraction method	Markov based obstacle segmentation	-	UV disparity obstacle detection algorithm.	Reinforcement and Q learning	Geometric features extraction method
Data Mining Technique	-	-	-	-	-	Image data mining
Intelligent Data Storage	Local	-	-	-	-	Cloud storage
Assumption	-	Ground is flat.	-	-	-	-
Decision-Making Model	Not applicable	Not applicable	Not applicable	Danger or Potential danger is highlighted.	-	Graphical warnings based
Heterogeneous Datasets	Yes	Yes	Yes	Yes	-	Yes
Matching Parameters	Histogram	No	No	Visual odometery	-	Histogram

Figure 6.12 and 6.13 illustrates the performance comparison of the proposed algorithm using IoU score and statistics inculding accuracy, precision, recall, F-score, TPR, FPR rate, mean absolute error of the detected obstacle.

Figure 6.12 depicts that the average IoU score of the detected obstacles is 83%. Figure 6.13 shows the statistics of the detected obstacle where accuracy is 96%, mean absolute percentage error is 95%, Precision is 90.7%, Recall value is 91.6%, F-score is 0.91, TPR rate is 0.916.

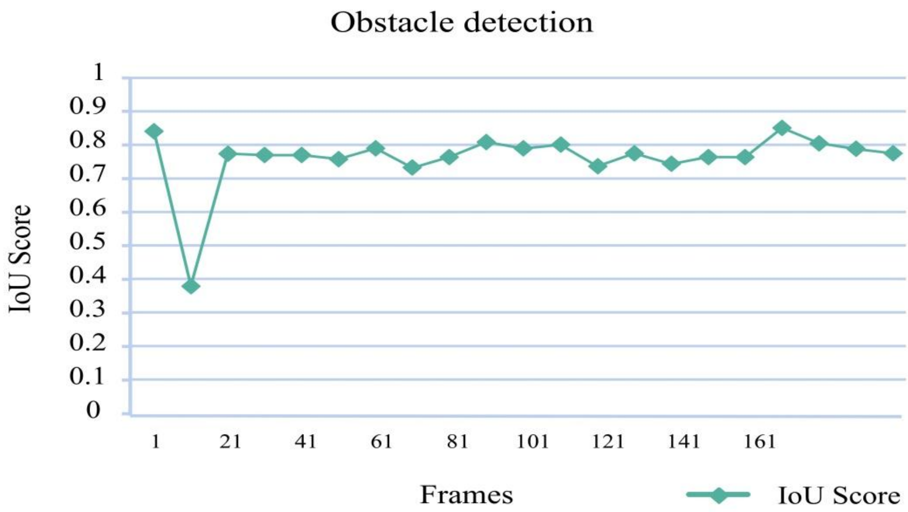

FIGURE 6.12: *IoU score of detected obstacles*

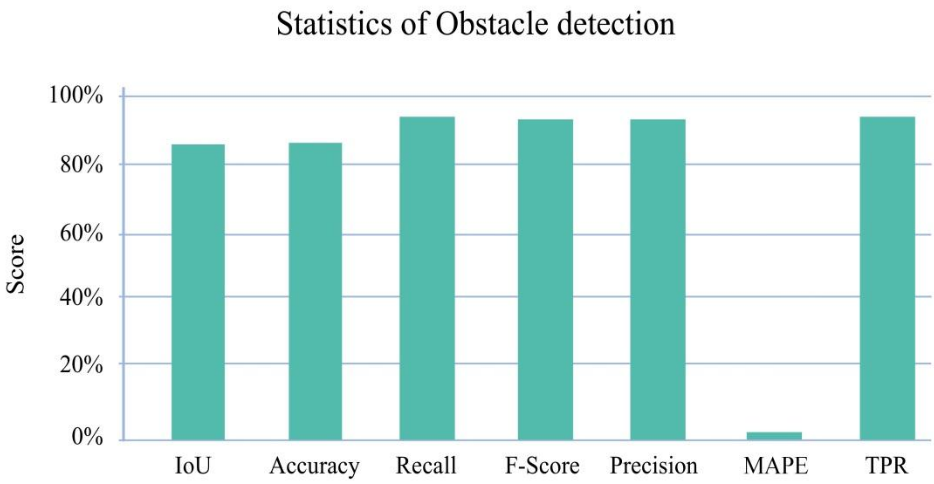

FIGURE 6.13: *Obstacle detection statistics*

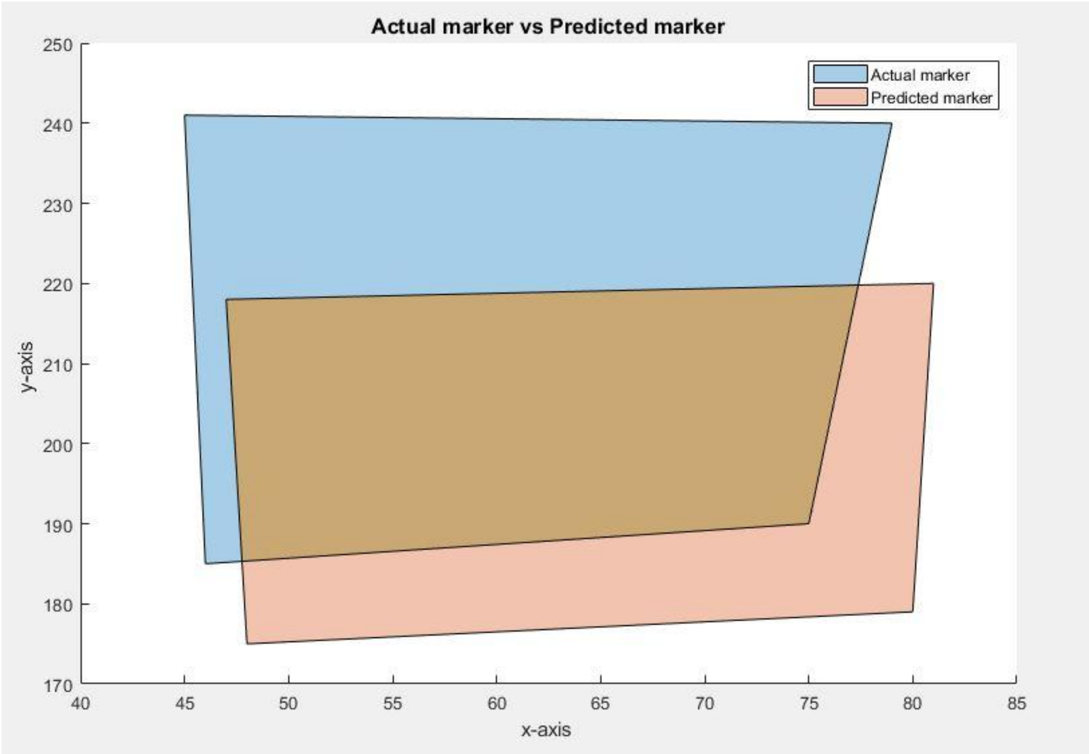

FIGURE 6.14: *Poor result of detected obstacle with IoU score=0.41*

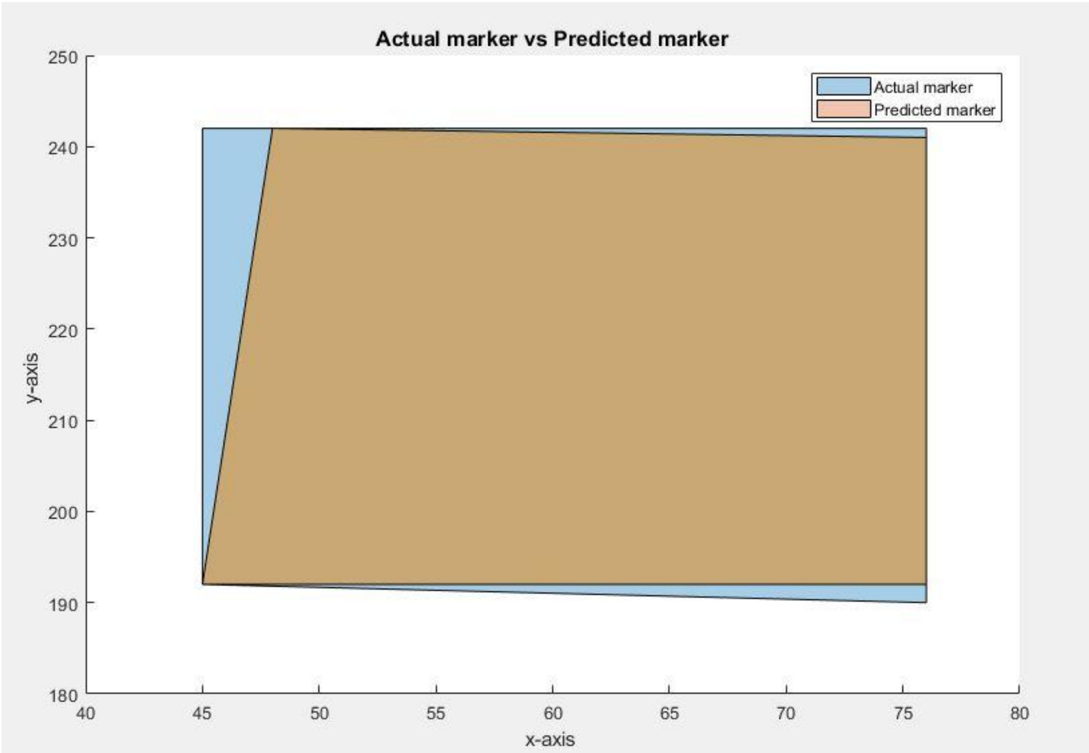

FIGURE 6.15: *Good result of detected obstacle with IoU score=0.92*

Figure 6.14 and 6.15 specifies the observations of the intersected regions of the predicted obstacle with the actual detected obstacle within the frames. Poor results show that the actual obstacle that is detected by the proposed algorithm is the wrong detected obstacle whereas the good detected obstacle result states that the obstacle has been detected successfully in the particular frame.

Worst case complexity: O($(m*n)^2N$**)** where where m*n is the size of the frames and N is the number of frames in the real time video.

Best case complexity: O(mn*N log(m*n)**)** where m*n is the size of the frames and N is the number of frames in the real time video.

6.4 CONCLUSION

This work proposes a vision-based approach for the stationary obstacle detection process. Three main parameters have been taken for this purpose: immovability, line of sight, and certain geometric conditions. Stationary objects present within the line of sight (LoS) of the vehicle and the contents satisfying the given geometric conditions are considered to be obstacles. This approach will help to detect obstacles by discarding unnecessary detections of each and every object coming in the way. This will also reduce the computational time in the detection of obstacles. In most of the previous works, sensors have been paid heed for the detection of obstacles, but here, in this work, the main emphasis is on vision-based obstacle detection even in an unknown environment. Here, no prebuilt dataset has been used. The results are verified on the MATLAB 2017b version and so, the future work will help to recognize the obstacles encountered in the navigation path as this work is limited to the detection process based on ground clearance and width only.

CHAPTER – 7

CONCLUSION AND FUTURE WORK

In this chapter, the conclusion and the future scope of the proposed work is discussed. Section 7.1 presents the conclusion and section 7.2 presents the future work of the proposed work.

7.1 CONCLUSION

An image processing mechanism for augmented reality-based autonomous navigation system has been proposed in this work. This system comprises three main tasks: 1) Bookmark identification and categorization of markers into primary, secondary and tertiary markers, 2) Creation of self-refining marker repository for AR applications, 3) Obstacle detection for safer navigation.

For the bookmark identification, the markers are detected, categorized and identified from the real-world environment. The main focus of this work is on the detection of stationary objects. The novelty of the work lies in its camera-based detection of the three types of markers, primary, secondary, and tertiary to support effective and efficient navigation even in an unknown environment. No pre-trained data is used for the learning process. The system will learn on its owns during the runs, starting from bare run, maturing steadily to completely autonomous run.

The self-created and self-refined repository has been made by the system automatically, and this will help the system to navigate in the next runs. The navigational information is labeled along with the markers in the repository. Three types of repositories have been made consisting of primary, secondary and tertiary markers. These repositories refine itself during the further runs automatically depending on the usability of the respective marker. The information stored along with the markers can assist the AR applications for better understanding of the surroundings.

For assuring the safety, run-time obstacles are also detected. The obstacles are detected with the use of a camera only. Geometric parameters, Line of Sight (LoS), and camera calibration process are responsible for the detection purpose.

This system can be associated with Google’s work, and even the cab drivers can use the markers as the reference points for location sharing with the customers to make their system more user-friendly.

7.2 FUTURE WORK

In this work, labelling of the markers is done on the cloud system. Therefore, the execution time of the algorithms depends upon the request and the response time of the servers and the resource availability. The future work will focus on optimizing the load balancing within the cloud infrastructure on concurrent requests and also on performing all the computations at the device level rather than depending upon the request and response time of the cloud infrastructure which will result in faster execution of the proposed algorithms.

In some cases, the proposed system can detect a non-obstacle as an obstacle. So, in future, to resolve this ambiguity, one more repository named as pre-tagged non-obstacle e.g. A speed breaker or rumble strips viewed in the bare run, can be created. All the identified objects which are not actual obstacles will be stored in this repository. By this, the differentiation among the markers, obstacles and the identified objects can be done successfully.

REFERENCES

[1] World Health Organization, 2018. *Global status report on road safety 2018: Summary* (No. WHO/NMH/NVI/18.20). World Health Organization.

[2] Schmalstieg, D., Langlotz, T. and Billinghurst, M., 2011. Augmented Reality 2.0. In *Virtual realities* (pp. 13-37). Springer, Vienna.

[3] Azuma, R.T., 1997. A survey of augmented reality. *Presence: Teleoperators & Virtual Environments*, *6*(4), pp.355-385.

[4] Van Krevelen, D.W.F. and Poelman, R., 2010. A survey of augmented reality technologies, applications and limitations. *International journal of virtual reality*, *9*(2), pp.1-20.

[5] Nakka, V. and Kabirdas, A., 2011. Design and realization of augmented reality based navigation assistance system. *Intr. J. of computer science and information technologies*, *2*(6), pp.2842-2846.

[6] Czech, P., Turoń, K. and Barcik, J., 2018. Autonomous vehicles: basic issues. *Zeszyty Naukowe. Transport/Politechnika Śląska*.

[7] Mann, S., 1997. Wearable computing: A first step toward personal imaging. *Computer*, *30*(2), pp.25-32.

[8] Wu, W., Blaicher, F., Yang, J., Seder, T. and Cui, D., 2009, October. A prototype of landmark-based car navigation using a full-windshield head-up display system. In *Proceedings of the 2009 workshop on Ambient media computing* (pp. 21-28).

[9] Burnett, G.E., 1998. *'Turn right at the King's Head': drivers' requirements for route guidance information* (Doctoral dissertation, Loughborough University of Technology).

[10] Narzt, W., Pomberger, G., Ferscha, A., Kolb, D., Müller, R., Wieghardt, J., Hörtner, H. and Lindinger, C., 2004, June. A new visualization concept for

navigation systems. In *ERCIM Workshop on User Interfaces for All* (pp. 440-451). Springer, Berlin, Heidelberg.

[11] Khandelwal, P., Swarnalatha, P., Bisht, N. and Prabu, S., 2015. Detection of features to track objects and segmentation using grabcut for application in marker-less augmented reality. *Procedia Computer Science*, *58*, pp.698-705.

[12] Jacob, R.J., 2006, April. What is the next generation of human-computer interaction?. In *CHI'06 Extended Abstracts on Human Factors in Computing Systems* (pp. 1707-1710).

[13] Zhou, F., Duh, H.B.L. and Billinghurst, M., 2008, September. Trends in augmented reality tracking, interaction and display: A review of ten years of ISMAR. In *2008 7th IEEE/ACM International Symposium on Mixed and Augmented Reality* (pp. 193-202). IEEE.

[14] Billinghurst, M., Clark, A. and Lee, G., 2015. A survey of augmented reality.

[15] Arth, C., Grasset, R., Gruber, L., Langlotz, T., Mulloni, A. and Wagner, D., 2015. The history of mobile augmented reality. *arXiv preprint arXiv:1505.01319*.

[16] Azuma, R., Baillot, Y., Behringer, R., Feiner, S., Julier, S. and MacIntyre, B., 2001. Recent advances in augmented reality. *IEEE computer graphics and applications*, *21*(6), pp.34-47.

[17] Guo, Y., Du, Q., Luo, Y., Zhang, W. and Xu, L., 2008. Application of augmented reality GIS in architecture. *ISPRS VIII*.

[18] Schmalstieg, D., 2005, October. Augmented reality techniques in games. In *Fourth IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR'05)* (pp. 176-177). IEEE.

[19] Mekni, M. and Lemieux, A., 2014. Augmented reality: Applications, challenges and future trends. *Applied Computational Science*, pp.205-214.

[20] Gleue, T. and Dähne, P., 2001, November. Design and implementation of a mobile device for outdoor augmented reality in the archeoguide project.

In *Proceedings of the 2001 conference on Virtual reality, archeology, and cultural heritage* (pp. 161-168).

[21] Neumann, U. and Majoros, A., 1998, March. Cognitive, performance, and systems issues for augmented reality applications in manufacturing and maintenance. In *Proceedings. IEEE 1998 Virtual Reality Annual International Symposium (Cat. No. 98CB36180)* (pp. 4-11). IEEE.

[22] Jani, B.Y., Dahale, P., Nagane, A., Sathe, B. and Wadghule, N., 2015. Interior Design in Augmented Reality Environment. *International Journal of Advanced Research in Computer and Communication Engineering*, *4*(3), pp.286-288.

[23] Yu, D., Jin, J.S., Luo, S., Lai, W. and Huang, Q., 2009. A useful visualization technique: a literature review for augmented reality and its application, limitation & future direction. In *Visual information communication* (pp. 311-337). Springer, Boston, MA.

[24] Wagner, D. and Schmalstieg, D., 2003, October. First steps towards handheld augmented reality. In *Seventh IEEE International Symposium on Wearable Computers, 2003. Proceedings.* (pp. 127-135). IEEE.

[25] Friedrich, W., Jahn, D. and Schmidt, L., 2002, September. ARVIKA-Augmented Reality for Development, Production and Service. In *ISMAR* (Vol. 2002, pp. 3-4).

[26] Kahn, S., 2013. Reducing the gap between Augmented Reality and 3D modeling with real-time depth imaging. *Virtual Reality*, *17*(2), pp.111-123.

[27] Dünser, A., Grasset, R. and Billinghurst, M., 2008. *A survey of evaluation techniques used in augmented reality studies* (pp. 1-27). Human Interface Technology Laboratory New Zealand.

[28] Kray, C., Elting, C., Laakso, K. and Coors, V., 2003, January. Presenting route instructions on mobile devices. In *Proceedings of the 8th international conference on Intelligent user interfaces* (pp. 117-124).

[29] Bhorkar, G., 2017. A survey of augmented reality navigation. *arXiv preprint arXiv:1708.05006*.

[30] Murphy-Chutorian, E. and Trivedi, M.M., 2010. Head pose estimation and augmented reality tracking: An integrated system and evaluation for monitoring driver awareness. *IEEE Transactions on intelligent transportation systems*, *11*(2), pp.300-311.

[31] Trivedi, M.M., Gandhi, T. and McCall, J., 2007. Looking-in and looking-out of a vehicle: Computer-vision-based enhanced vehicle safety. *IEEE Transactions on Intelligent Transportation Systems*, *8*(1), pp.108-120.

[32] Gavrila, D.M., 2001. Sensor-based pedestrian protection. *IEEE Intelligent Systems*, *16*(6), pp.77-81.

[33] Gandhi, T. and Trivedi, M.M., 2006, September. Pedestrian collision avoidance systems: A survey of computer vision based recent studies. In *2006 IEEE Intelligent Transportation Systems Conference* (pp. 976-981). IEEE.

[34] McCall, J.C. and Trivedi, M.M., 2006. Video-based lane estimation and tracking for driver assistance: survey, system, and evaluation. *IEEE transactions on intelligent transportation systems*, *7*(1), pp.20-37.

[35] Okuno, A., Fujita, K. and Kutami, A., 1992. Visual navigation of an autonomous on-road vehicle: Autonomous cruising on highways. In *Vision-based vehicle guidance* (pp. 222-237). Springer, New York, NY.

[36] Lin, C.F., Ulsoy, A.G. and LeBlanc, D.J., 2000. Vehicle dynamics and external disturbance estimation for vehicle path prediction. *IEEE Transactions on Control Systems Technology*, *8*(3), pp.508-518.

[37] Kim, J., Park, C. and Kweon, I.S., 2011. Vision-based navigation with efficient scene recognition. *Intelligent Service Robotics*, *4*(3), pp.191-202.

[38] Cho, H., Sung, M. and Jun, B., 2016. Canny text detector: Fast and robust scene text localization algorithm. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 3566-3573).

[39] Dang, L., Tewolde, G., Zhang, X. and Kwon, J., 2017, September. Reduced resolution lane detection algorithm. In *2017 IEEE AFRICON* (pp. 1459-1464). IEEE.

[40] Limmer, M., Forster, J., Baudach, D., Schüle, F., Schweiger, R. and Lensch, H.P., 2016, November. Robust deep-learning-based road-prediction for augmented reality navigation systems at night. In *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)* (pp. 1888-1895). IEEE.

[41] Baek, I. and He, M., 2018. Vehicles Lane-changing Behavior Detection. *arXiv preprint arXiv:1808.07518*.

[42] Wang, Z., Ren, W. and Qiu, Q., 2018. LaneNet: Real-time lane detection networks for autonomous driving. *arXiv preprint arXiv:1807.01726*.

[43] Guan, H., Xingang, W., Wenqi, W., Han, Z. and Yuanyuan, W., 2016, May. Real-time lane-vehicle detection and tracking system. In *2016 Chinese Control and Decision Conference (CCDC)* (pp. 4438-4443). IEEE.

[44] Aly, M., 2008, June. Real time detection of lane markers in urban streets. In *2008 IEEE Intelligent Vehicles Symposium* (pp. 7-12). IEEE.

[45] Martinez, R., 1997. *Traffic Safety Facts (1996): A Compilation of Motor Vehicle Crash Data from the Fatality Analysis Reporting System and the General Estimates System*. DIANE Publishing.

[46] Cheung, E., Bera, A. and Manocha, D., 2018. Efficient and safe vehicle navigation based on driver behavior classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops* (pp. 1024-1031).

[47] Uchida, N., Tagawa, T. and Sato, K., 2017. Development of an augmented reality vehicle for driver performance evaluation. *IEEE Intelligent Transportation Systems Magazine*, *9*(1), pp.35-41.

[48] Cheung, E., Bera, A., Kubin, E., Gray, K. and Manocha, D., 2018, October. Identifying driver behaviors using trajectory features for vehicle navigation. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (pp. 3445-3452). IEEE.

[49] Tonnis, M., Sandor, C., Klinker, G., Lange, C. and Bubb, H., 2005, October. Experimental evaluation of an augmented reality visualization for directing a car driver's attention. In *Fourth IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR'05)* (pp. 56-59). IEEE.

[50] Gavrila, D.M., 2001. Sensor-based pedestrian protection. *IEEE Intelligent Systems*, *16*(6), pp.77-81.

[51] Rasouli, A. and Tsotsos, J.K., 2019. Autonomous vehicles that interact with pedestrians: A survey of theory and practice. *IEEE transactions on intelligent transportation systems*.

[52] Jimison, D., Sambasivan, N. and Pahwa, S., 2007. Wigglestick: an Urban pedestrian mobile social navigation system. In *Georgia Tech Graduate Symposium, Georgia Tech, Atlanta*.

[53] Kwon, S., Ding, D., Park, J., Kim, Y.D., Kim, J. and Jung, W., 2016. Design and Implementation of Traffic-Scene Understanding System for Autonomous Vehicle. *Bulletin of Networking, Computing, Systems, and Software*, *5*(1), pp.73-74.

[54] Templeton, B., Waymo LLC, 2017. *Methods and systems for transportation to destinations by a self-driving vehicle*. U.S. Patent 9,665,101.

[55] Rajesh, R., Rajeev, K., Suchithra, K., Lekhesh, V.P., Gopakumar, V. and Ragesh, N.K., 2011, July. Coherence vector of oriented gradients for traffic sign recognition using neural networks. In *The 2011 International Joint Conference on Neural Networks* (pp. 907-910). IEEE.

[56] Garcia-Garrido, M.A., Sotelo, M.A. and Martin-Gorostiza, E., 2006, September. Fast traffic sign detection and recognition under changing lighting conditions. In *2006 IEEE Intelligent Transportation Systems Conference* (pp. 811-816). IEEE.

[57] Kastrinaki, V., Zervakis, M. and Kalaitzakis, K., 2003. A survey of video processing techniques for traffic applications. *Image and vision computing*, *21*(4), pp.359-381.

[58] Sharma, M., Shukla, A. and Shastri, S., 2012. Robot navigation using image processing and isolated word recognition. *International Journal on Computer Science and Engineering*, *4*(5), p.769.

[59] Feng, J., 2014. *Traffic sign detection and recognition system for intelligent vehicles* (Doctoral dissertation, University of Ottawa).

[60] Li, H., Sun, F., Liu, L. and Wang, L., 2015. A novel traffic sign detection method via color segmentation and robust shape matching. *Neurocomputing*, *169*, pp.77-88.

[61] Tagunde, G.A., Uke, N.J. and Banchhor, C., 2012. Detection, classification and recognition of road traffic signs using color and shape features. *Int. J. Adv. Technol. Eng. Res*, *2*(4), pp.202-206.

[62] Solanki, D.S. and Dixit, G., 2015. Traffic sign detection using feature based method. *International Jornal of Advanced Research in Comuter Science and Software Engineering,(), Feb. pp*.

[63] Sharma S, Rani K.N., Gupta L.R., 2016. Traffic road sign Detection and Recognition using Geometric shapes and Background color: Laying a foundation to use Augmented Reality (A.R.) in Autonomous vehicle Navigation and Decision making. Research Journal of Recent Sciences, 5, pp.17-20.

[64] Wu, W., Blaicher, F., Yang, J., Seder, T. and Cui, D., 2009, October. A prototype of landmark-based car navigation using a full-windshield head-up display system. In *Proceedings of the 2009 workshop on Ambient media computing* (pp. 21-28).

[65] Reitmayr, G. and Schmalstieg, D., 2004. *Collaborative augmented reality for outdoor navigation and information browsing* (pp. 31-41). na.

[66] Mahmood, A., Butler, B. and Jennings, B., 2018. Potential of augmented reality for intelligent transportation systems. *arXiv preprint arXiv: 1806.04724*.

[67] Taneja, A., Ballan, L. and Pollefeys, M., 2014, November. Never get lost again: Vision based navigation using streetview images. In *Asian Conference on Computer Vision* (pp. 99-114). Springer, Cham.

[68] Jian, Y.D. and Ni, K., Faraday Future Inc, 2018. *System and method for vehicle localization assistance using sensor data*. U.S. Patent Application 15/679,019.

[69] Kim, J. and Jun, H., 2008. Vision-based location positioning using augmented reality for indoor navigation. *IEEE Transactions on Consumer Electronics*, *54*(3), pp.954-962.

[70] Huang, Y. and Yang, J., 2008, January. Study of robot landmark recognition with complex background. In *ICMIT 2007: Mechatronics, MEMS, and Smart Materials* (Vol. 6794, p. 67942U). International Society for Optics and Photonics.

[71] Gordon, T.J. and Lidberg, M., 2015. Automated driving and autonomous functions on road vehicles. *Vehicle System Dynamics*, *53*(7), pp.958-994.

[72] Gupta, B., Chaube, A., Negi, A. and Goel, U., 2017. Study on Object Detection using Open CV-Python. *International Journal of Computer Applications*, *162*(8), pp.17-21.

[73] Anvaripour, M. and Ebrahimnezhad, H., 2015. Accurate object detection using local shape descriptors. *Pattern Analysis and Applications*, *18*(2), pp.277-295.

[74] Wang, R., Yang, N., Stueckler, J. and Cremers, D., 2019. DirectShape: Photometric Alignment of Shape Priors for Visual Vehicle Pose and Shape Estimation. *arXiv preprint arXiv:1904.10097*.

[75] Limmer, M., Forster, J., Baudach, D., Schüle, F., Schweiger, R. and Lensch, H.P., 2016, November. Robust deep-learning-based road-prediction for augmented reality navigation systems at night. In *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)* (pp. 1888-1895). IEEE.

[76] Konrardy, B., Christensen, S.T., Hayward, G.L. and Farris, S., State Farm Mutual Automobile Insurance Co, 2018. *Autonomous vehicle application*. U.S. Patent 10,134,278.

[77] Dey, N.S., Mohanty, R. and Chugh, K.L., 2012. On Road Navigation System Using Spatial and MotionImage Processing for Automatic Navigation System. *International Journal of Innovation, Management and Technology*, *3*(1), p.80.

[78] Hirzer, M., 2008, October. Marker detection for augmented reality applications. In *Seminar/Project Image Analysis Graz* (pp. 1-2).

[79] Pfannmueller, L., Kramer, M., Senner, B. and Bengler, K., 2015. A comparison of display concepts for a navigation system in an automotive contact analog head-up display. *Procedia Manufacturing*, *3*, pp.2722-2729.

[80] Piekarski, W., Avery, B., Thomas, B.H. and Malbezin, P., 2004, March. Integrated head and hand tracking for indoor and outdoor augmented reality. In *IEEE Virtual Reality 2004* (pp. 11-276). IEEE.

[81] Newman, J., Ingram, D. and Hopper, A., 2001, October. Augmented reality in a wide area sentient environment. In *Proceedings IEEE and ACM International Symposium on Augmented Reality* (pp. 77-86). IEEE.

[82] Chen, C.Y., Chiang, S.Y. and Wu, C.T., 2016. Path planning and obstacle avoidance for omni-directional mobile robot based on Kinect depth sensor. *International Journal of Embedded Systems*, *8*(4), pp.343-351.

[83] Schmidt, J., Niemann, H. and Vogt, S., 2002, December. Dense disparity maps in real-time with an application to augmented reality. In *Sixth IEEE Workshop on Applications of Computer Vision, 2002.(WACV 2002). Proceedings.* (pp. 225-230). IEEE.

[84] Park, J.S., Bae, B.J. and Jain, R., 2012. Fast natural feature tracking for mobile augmented reality applications.

[85] Chao, H., Gu, Y. and Napolitano, M., 2014. A survey of optical flow techniques for robotics navigation applications. *Journal of Intelligent & Robotic Systems*, *73*(1-4), pp.361-372.

[86] Köhler, J., Pagani, A. and Stricker, D., 2011. Detection and identification techniques for markers used in computer vision. In *Visualization of Large and Unstructured Data Sets-Applications in Geospatial Planning, Modeling and Engineering (IRTG 1131 Workshop)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik.

[87] Rose, E., Breen, D., Ahlers, K.H., Crampton, C., Tuceryan, M., Whitaker, R. and Greer, D., 1995. Annotating real-world objects using augmented reality. In *Computer graphics* (pp. 357-370). Academic Press.

[88] Neumann, U. and You, S., 1999. Natural feature tracking for augmented reality. *IEEE Transactions on Multimedia*, *1*(1), pp.53-64.

[89] Höllerer, T., Feiner, S., Terauchi, T., Rashid, G. and Hallaway, D., 1999. Exploring MARS: developing indoor and outdoor user interfaces to a mobile augmented reality system. *Computers & Graphics*, *23*(6), pp.779-785.

[90] Reitmayr, G. and Schmalstieg, D., 2004. *Collaborative augmented reality for outdoor navigation and information browsing* (pp. 31-41). na.

[91] Robertson, D.P. and Cipolla, R., 2004, September. An Image-Based System for Urban Navigation. In *Bmvc* (Vol. 19, No. 51, p. 165).

[92] Bradley, D., Brunton, A., Fiala, M. and Roth, G., 2005, October. Image-based navigation in real environments using panoramas. In *IEEE International Workshop on Haptic Audio Visual Environments and their Applications* (pp. 3-pp). IEEE.

[93] Jung, C.R., Saldanha, J.S. and da Silveira Jr, L.G., 2009, June. Augmented Reality with Automatic Camera Calibration for Driver Assistance Systems. In *Proceedings of the XI symposium on virtual and augmented reality* (pp. 29-36).

[94] Kim, Y., Lee, K., Choi, K. and Cho, S.I., 2006, September. Building recognition for augmented reality based navigation system. In *The Sixth IEEE International Conference on Computer and Information Technology (CIT'06)* (pp. 131-131). IEEE.

[95] Lee, D.H. and Park, J., 2007, December. Augmented reality based museum guidance system for selective viewings. In *Second Workshop on Digital Media and its Application in Museum & Heritages (DMAMH 2007)* (pp. 379-382). IEEE.

[96] Kim, J. and Jun, H., 2008. Vision-based location positioning using augmented reality for indoor navigation. *IEEE Transactions on Consumer Electronics*, *54*(3), pp.954-962.

[97] Mulloni, A., Wagner, D., Barakonyi, I. and Schmalstieg, D., 2009. Indoor positioning and navigation with camera phones. *IEEE Pervasive Computing*, *8*(2), pp.22-31.

[98] Mohareri, O. and Rad, A.B., 2011, April. Autonomous humanoid robot navigation using augmented reality technique. In *2011 IEEE International Conference on Mechatronics* (pp. 463-468). IEEE.

[99] Kaneko, K., Harada, K., Kanehiro, F., Miyamori, G. and Akachi, K., 2008, September. Humanoid robot HRP-3. In *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems* (pp. 2471-2478). IEEE.

[100] Messom, C.H., 2009. Stereo Vision Controlled Humanoid Robot Tool-kit.

[101] Kasprzak, S., Komninos, A. and Barrie, P., 2013, July. Feature-based indoor navigation using augmented reality. In *2013 9th international conference on intelligent environments* (pp. 100-107). IEEE.

[102] Mélykúti, Z.T.T.L.G. and Barsi, Á., IMAGE-BASED DRIVER'S GUIDANCE SYSTEM.

[103] Taneja, A., Ballan, L. and Pollefeys, M., 2014, November. Never get lost again: Vision based navigation using streetview images. In *Asian Conference on Computer Vision* (pp. 99-114). Springer, Cham.

[104] Kemsaram, N., Thatiparti, T.V.R., Guntupalli, D.R. and Kuvvarapu, A., 2016. A hybrid autonomous visual tracking algorithm for micro aerial vehicles. *International Journal of Engineering Sciences & Research Technology*, *5*(8), pp.524-535.

[105] Reiser, D., Paraforos, D.S., Khan, M.T., Griepentrog, H.W. and Vázquez-Arellano, M., 2017. Autonomous field navigation, data acquisition and node location in wireless sensor networks. *Precision Agriculture*, *18*(3), pp.279-292.

[106] Ort, T., Paull, L. and Rus, D., 2018, May. Autonomous vehicle navigation in rural environments without detailed prior maps. In *2018 IEEE International Conference on Robotics and Automation (ICRA)* (pp. 2040-2047). IEEE.

[107] Passot, J.B., Smith, A., Szatmary, B., Gabardos, B.I., Griffin, C., Rambouts, J., Sinyavskiy, O. and Izhikevich, E., Brain Corp, 2019. *Systems and methods for training a robot to automomously travel a route*. U.S. Patent Application 16/168,368.

[108] Paolillo, A., Gergondet, P., Cherubini, A., Vendittelli, M. and Kheddar, A., 2018. Autonomous car driving by a humanoid robot. *Journal of Field Robotics*, *35*(2), pp.169-186.

[109] Huang, Y.P., Sithole, L. and Lee, T.T., 2017. Structure from motion technique for scene detection using autonomous drone navigation. *IEEE Transactions On Systems, Man, And Cybernetics: Systems*.

[110] Fernández, L., Payá, L., Reinoso, O., Jiménez, L.M. and Ballesta, M., 2016. A study of visual descriptors for outdoor navigation using google street view images. *Journal of Sensors*, *2016*.

[111] Kaneko, A.M. and Yamamoto, K., 2016, March. Landmark recognition based on image characterization by segmentation points for autonomous driving. In *2016 SICE International Symposium on Control Systems (ISCS)* (pp. 1-8). IEEE.

[112] Bagyaveereswaran, V., Rajagopal, N.N. and Anitha, R., 2016. Image Based Autonomous Navigation for a Lander. *International Journal of Applied Engineering Research*, *11*(4), pp.2424-2428.

[113] de Lima, D.A. and Victorino, A.C., 2016. A hybrid controller for vision-based navigation of autonomous vehicles in urban environments. *IEEE Transactions on Intelligent Transportation Systems*, *17*(8), pp.2310-2323.

[114] Wei, X., Phung, S.L. and Bouzerdoum, A., 2016. Visual descriptors for scene categorization: experimental evaluation. *Artificial Intelligence Review*, *45*(3), pp.333-368.

[115] Litman, T., 2014. Autonomous vehicle implementation predictions implications for transport planning. Victoria, BC, Canada: Victoria Transp. *Policy Inst*.

[116] Bonin-Font, F., Ortiz, A. and Oliver, G., 2008. Visual navigation for mobile robots: A survey. *Journal of intelligent and robotic systems*, *53*(3), p.263.

[117] Lu, Y.H. and Delp III, E.J., 2004, January. An overview of problems in image-based location awareness and navigation. In *Visual Communications and Image Processing 2004* (Vol. 5308, pp. 102-109). International Society for Optics and Photonics.

[118] DeSouza, G.N. and Kak, A.C., 2002. Vision for mobile robot navigation: A survey. *IEEE transactions on pattern analysis and machine intelligence*, *24*(2), pp.237-267.

[119] Giachetti, A., Campani, M. and Torre, V., 1998. The use of optical flow for road navigation. *IEEE transactions on robotics and automation*, *14*(1), pp.34-48.

[120] Ohya, I., Kosaka, A. and Kak, A., 1998. Vision-based navigation by a mobile robot with obstacle avoidance using single-camera vision and ultrasonic sensing. *IEEE transactions on robotics and automation*, *14*(6), pp.969-978.

[121] Indu, S., Gupta, M. and Bhattacharyya, A., 2011. Vehicle tracking and speed estimation using optical flow method. *Int. J. Engineering Science and Technology*, *3*(1), pp.429-434.

[122] Paygude, S.S., Vibha, V. and Manisha, C., 2013. Vehicle detection and tracking using the opticalflow and background subtraction. In *Proc. of Int. Conf. on Advances in Computer Science and Application*.

[123] Park, J.S., Bae, B.J. and Jain, R., 2012. Fast natural feature tracking for mobile augmented reality applications.

[124] Patel, A.Z., Mateen, A.M.A., Ali, S.A., Javed, K.A. and Pat, H.M.N., 2015. Implementation of Path Navigator for Smart Devices using Augmented Reality. *International Journal of Advanced Research in Computer Science*, *6*(2).

[125] Yamaguchi, Y., Nakagawa, T., Akaho, K., Honda, M., Kato, H. and Nishida, S., 2007, July. AR-Navi: An in-vehicle navigation system using video-based augmented reality technology. In *Symposium on Human Interface and the Management of Information* (pp. 1139-1147). Springer, Berlin, Heidelberg.

[126] Azuma, R., Lee, J.W., Jiang, B., Park, J., You, S. and Neumann, U., 1999. Tracking in unprepared environments for augmented reality systems. *Computers & Graphics*, *23*(6), pp.787-793.

[127] Kim, B.G., Choi, J.S. and Kim, J.T., 2014. Feature Points Extraction for Camera Tracking in Augmented Reality System. *International Journal of Multimedia and Ubiquitous Engineering*, *9*(9), pp.269-280.

[128] Hile, H., Vedantham, R., Cuellar, G., Liu, A., Gelfand, N., Grzeszczuk, R. and Borriello, G., 2008, December. Landmark-based pedestrian navigation from collections of geotagged photos. In *Proceedings of the 7th international conference on mobile and ubiquitous multimedia* (pp. 145-152).

[129] Davison, A.J., Reid, I.D., Molton, N.D. and Stasse, O., 2007. MonoSLAM: Real-time single camera SLAM. *IEEE transactions on pattern analysis and machine intelligence*, *29*(6), pp.1052-1067.

[130] Klopschitz, M. and Schmalstieg, D., 2007, November. Automatic reconstruction of wide-area fiducial marker models. In *2007 6th IEEE and ACM International Symposium on Mixed and Augmented Reality* (pp. 71-74). IEEE.

[131] Thompson, S., Kagami, S. and Nishiwaki, K., 2006, December. Localisation for autonomous humanoid navigation. In *2006 6th IEEE-RAS International Conference on Humanoid Robots* (pp. 13-19). IEEE.

[132] Yeh, T., Tollmar, K. and Darrell, T., 2004, June. Searching the web with mobile images for location recognition. In *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004.* (Vol. 2, pp. II-II). IEEE.

[133] Hornung, A., Wurm, K.M. and Bennewitz, M., 2010, October. Humanoid robot localization in complex indoor environments. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems* (pp. 1690-1695). IEEE.

[134] Kasprzak, S., Komninos, A. and Barrie, P., 2013, July. Feature-based indoor navigation using augmented reality. In *2013 9th international conference on intelligent environments* (pp. 100-107). IEEE.

[135] Hornung, A., Oßwald, S., Maier, D. and Bennewitz, M., 2014. Monte Carlo localization for humanoid robot navigation in complex indoor environments. *International Journal of Humanoid Robotics*, *11*(02), p.1441002.

[136] Do, Q. and Jain, L., 2010. Application of neural processing paradigm in visual landmark recognition and autonomous robot navigation. *Neural Computing and Applications*, *19*(2), pp.237-254.

[137] Busquets, D., Sierra, C. and De Màntaras, R.L., 2003. A multiagent approach to qualitative landmark-based navigation. *Autonomous Robots*, *15*(2), pp.129-154.

[138] Tovar, B., La Valle, S.M. and Murrieta, R., 2003, September. Optimal navigation and object finding without geometric maps or localization. In *2003 IEEE International Conference on Robotics and Automation (Cat. No. 03CH37422)* (Vol. 1, pp. 464-470). IEEE.

[139] Weiss, S.M., Kapouleas, I. and Shavlik, J.W., 1990. An empirical comparison of pattern recognition, neural nets and machine learning classification methods. *Readings in machine learning*, pp.177-183.

[140] Rabbi, I., Ullah, S. and Khan, S.U., 2012. Augmented Reality Tracking Techniques: A Systematic Literature Review Protocol. *Journal of Computer Engineering IOSRJCE*, *2*(2).

[141] Chang, C.K., Siagian, C. and Itti, L., 2010, October. Mobile robot vision navigation & localization using gist and saliency. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems* (pp. 4147-4154). IEEE.

[142] Emharraf, M., Saber, M., Rahmoun, M. and Azizi, M., 2016. Online local path planning for mobile robot navigate in unknown indoor environment. In *Proceedings of the Mediterranean Conference on Information & Communication Technologies 2015* (pp. 69-76). Springer, Cham.

[143] Yong-Xu, Q., Jia-Min, L., Hui, Q., Bo, Y. and Chang-Xu, J., 2013, November. Campus navigation system based on mobile augmented reality. In *2013 6th International Conference on Intelligent Networks and Intelligent Systems (ICINIS)* (pp. 139-142). IEEE.

[144] Arth, C., Grasset, R., Gruber, L., Langlotz, T., Mulloni, A. and Wagner, D., 2015. The history of mobile augmented reality. *arXiv preprint arXiv:1505.01319*.

[145] Al-Muteb, K., Faisal, M., Emaduddin, M., Arafah, M., Alsulaiman, M., Mekhtiche, M., Hedjar, R., Mathkoor, H., Algabri, M. and Bencherif, M.A., 2016. An autonomous stereovision-based navigation system (ASNS) for mobile robots. *Intelligent Service Robotics*, *9*(3), pp.187-205.

[146] Muthukumar, N., Ramkumar, K., Srinivasan, S., Saravanakumar, G. and Subathra, B., 2017. Path-planning in autonomous electric vehicle using nonlinear state estimation and behaviour-based controllers. *International Journal of Vehicle Systems Modelling and Testing*, *12*(3-4), pp.304-315.

[147] Mohamed, A., Ren, J., Lang, H. and El-Gindy, M., 2018. Optimal path planning for an autonomous articulated vehicle with two trailers. *International Journal of Automation and Control*, *12*(3), pp.449-465.

[148] Yang, G., Hou, Z.G. and Liang, Z., 2006. Distributed visual navigation based on neural Q-learning for a mobile robot. *International journal of vehicle autonomous systems*, *4*(2-4), pp.225-238.

[149] Vivacqua, R.P.D., Bertozzi, M., Cerri, P., Martins, F.N. and Vassallo, R.F., 2017. Self-localization based on visual lane marking maps: An accurate low-cost approach for autonomous driving. *IEEE Transactions on Intelligent Transportation Systems*, *19*(2), pp.582-597.

[150] Prasad, D.K., Rajan, D., Rachmawati, L., Rajabally, E. and Quek, C., 2017. Video processing from electro-optical sensors for object detection and tracking in a maritime environment: a survey. *IEEE Transactions on Intelligent Transportation Systems*, *18*(8), pp.1993-2016.

[151] Khandelwal, P., Swarnalatha, P., Bisht, N. and Prabu, S., 2015. Detection of features to track objects and segmentation using grabcut for application in marker-less augmented reality. *Procedia Computer Science*, *58*, pp.698-705.

[152] Kim, B.G., Choi, J.S. and Kim, J.T., 2014. Feature Points Extraction for Camera Tracking in Augmented Reality System. *International Journal of Multimedia and Ubiquitous Engineering*, *9*(9), pp.269-280.

[153] Gouaillier, D., Hugel, V., Blazevic, P., Kilner, C., Monceaux, J., Lafourcade, P., Marnier, B., Serre, J. and Maisonnier, B., 2009, May. Mechatronic design of NAO humanoid. In *2009 IEEE International Conference on Robotics and Automation* (pp. 769-774). IEEE.

[154] Faessler, M., Fontana, F., Forster, C., Mueggler, E., Pizzoli, M. and Scaramuzza, D., 2016. Autonomous, vision-based flight and live dense 3D mapping with a quadrotor micro aerial vehicle. *Journal of Field Robotics*, *33*(4), pp.431-450.

[155] Borse, H., Dumbare, A., Gaikwad, R. and Lende, N., 2012. Mobile robot for object detection using image processing. *Global Journal of Computer Science and Technology*.

[156] Rodríguez-Canosa, G.R., Thomas, S., Del Cerro, J., Barrientos, A. and MacDonald, B., 2012. A real-time method to detect and track moving objects (DATMO) from unmanned aerial vehicles (UAVs) using a single camera. *Remote Sensing*, *4*(4), pp.1090-1111.

[157] Thakoor, N., Gao, J. and Chen, H., 2004, September. Automatic object detection in video sequences with camera in motion. In *Proceedings of Advanced Concepts for Intelligent Vision Systems*.

[158] Wekel, T., Kroll-Peters, O. and Albayrak, S., 2008, October. Vision based obstacle detection for wheeled robots. In *2008 International Conference on Control, Automation and Systems* (pp. 1587-1592). IEEE.

[159] Lan, J., Jiang, Y., Fan, G., Yu, D. and Zhang, Q., 2016. Real-time automatic obstacle detection method for traffic surveillance in urban traffic. *Journal of Signal Processing Systems*, *82*(3), pp.357-371.

[160] Yadav, P. and Jahagirdar, A., 2015. A static object detection in image sequences by self organizing background subtraction.

[161] Barandiaran, I., Congote, J., Goenetxea, J. and Ruiz, O.E., 2012. Evaluation of interest point detectors for image information extraction. In *KES* (pp. 2170-2179).

[162] Kalkofen, D., Sandor, C., White, S. and Schmalstieg, D., 2011. Visualization techniques for augmented reality. In *Handbook of Augmented Reality* (pp. 65-98). Springer, New York, NY.

[163] Arvind, C.S. and Senthilnath, J., 2019, May. Autonomous RL: Autonomous Vehicle Obstacle Avoidance in a Dynamic Environment using MLP-SARSA Reinforcement Learning. In *2019 IEEE 5th International Conference on Mechatronics System and Robots (ICMSR)* (pp. 120-124). IEEE.

[164] van Dam, G. and Van Kampen, E.J., 2020. Obstacle avoidance for quadrotors using reinforcement learning and obstacle-airflow interactions. In *AIAA Scitech 2020 Forum* (p. 2249).

[165] Kakoty, N.M., Mazumdar, M. and Sonowal, D., 2019. Mobile robot navigation in unknown dynamic environment inspired by human pedestrian behavior. In *Progress in Advanced Computing and Intelligent Engineering* (pp. 441-451). Springer, Singapore.

[166] Guzzi, J., Giusti, A., Gambardella, L.M., Theraulaz, G. and Di Caro, G.A., 2013, May. Human-friendly robot navigation in dynamic environments. In *2013 IEEE International Conference on Robotics and Automation* (pp. 423-430). IEEE.

[167] Hua, X., Wang, X., Rui, T., Wang, D. and Shao, F., 2019. Real-Time Object Detection in Remote Sensing Images Based on Visual Perception and Memory Reasoning. *Electronics*, *8*(10), p.1151.

[168] Song, J., Gao, S., Zhu, Y. and Ma, C., 2019. A survey of remote sensing image classification based on CNNs. *Big Earth Data*, *3*(3), pp.232-254.

[169] Lenskiy, A.A. and Lee, J.S., 2011. Machine Learning for Visual Navigation of Unmanned Ground Vehicles. In *Computational Modeling and Simulation of Intellect: Current State and Future Perspectives* (pp. 81-101). IGI Global.

[170] Pandey, S.K. and Janghel, R.R., 2019. Recent deep learning techniques, challenges and its applications for medical healthcare system: A review. *Neural Processing Letters*, *50*(2), pp.1907-1935.

[171] Yu, Q., Araújo, H. and Wang, H., 2005. A stereovision method for obstacle detection and tracking in non-flat urban environments. *Autonomous Robots*, *19*(2), pp.141-157.

[172] Heras Evangelio, R., 2014. Background subtraction for the detection of moving and static objects in video surveillance.

[173] Chan, S.W. and Lin, C.T., Industrial Technology Research Institute, 2019. *Image processing method and image system for transportation*. U.S. Patent 10,192,287.

[174] Alturki, R. and Gay, V., 2019. Augmented and virtual reality in mobile fitness applications: a survey. In *Applications of Intelligent Technologies in Healthcare* (pp. 67-75). Springer, Cham.

[175] Zhao, J., Liang, B. and Chen, Q., 2018. The key technology toward the self-driving car. *International Journal of Intelligent Unmanned Systems*.

[176] García, F., Prioletti, A., Cerri, P. and Broggi, A., 2018. PHD filter for vehicle tracking based on a monocular camera. *Expert Systems with Applications*, *91*, pp.472-479.

[177] Gordon, T.J. and Lidberg, M., 2015. Automated driving and autonomous functions on road vehicles. *Vehicle System Dynamics*, *53*(7), pp.958-994.

[178] Sun, D., Roth, S. and Black, M.J., 2010, June. Secrets of optical flow estimation and their principles. In *2010 IEEE computer society conference on computer vision and pattern recognition* (pp. 2432-2439). IEEE.

[179] Kirstein, S., Wersing, H. and Körner, E., 2005, September. Online learning for object recognition with a hierarchical visual cortex model. In *International Conference on Artificial Neural Networks* (pp. 487-492). Springer, Berlin, Heidelberg.

[180] Chopkar, T.A. and Lahade, S., 2016. Real time detection of moving object based on FPGA. *IOSR J. Electron. Commun. Eng.(IOSR-JECE)*, *11*(1), pp.37-41.

[181] Kaneko, N., Yoshida, T. and Sumi, K., 2017. Fast Obstacle Detection for Monocular Autonomous Mobile Robots. *SICE Journal of Control, Measurement, and System Integration*, *10*(5), pp.370-377.

[182] Chen, T.S., Tsai, H.W. and Peng, J.J., 2016. Prediction-based object tracking in visual sensor networks. *Wireless Personal Communications*, *87*(1), pp.145-163.

[183] Adam, M.S., Anisi, M.H. and Ali, I., 2017. Object tracking sensor networks in smart cities: Taxonomy, architecture, applications, research challenges and future directions. *Future Generation Computer Systems*.

[184] Cornacchia, M., Kakillioglu, B., Zheng, Y. and Velipasalar, S., 2018. Deep Learning-Based Obstacle Detection and Classification With Portable Uncalibrated Patterned Light. *IEEE Sensors Journal*, *18*(20), pp.8416-8425.

[185] Hasan, S. and Jumaa, S.S., 2018. Retracted: Distance estimation by computer vision and shortest path planning using single camera. *Journal of Fundamental and Applied Sciences*, *10*(6S), pp.216-221.

[186] Khalilullah, K.I., Ota, S., Yasuda, T. and Jindai, M., 2018. Road area detection method based on DBNN for robot navigation using single camera in outdoor environments. *Industrial Robot: An International Journal*.

[187] Song, W., Yang, Y., Fu, M., Li, Y. and Wang, M., 2018. Lane detection and classification for forward collision warning system based on stereo vision. *IEEE Sensors Journal*, *18*(12), pp.5151-5163.

[188] Song, W., Yang, Y., Fu, M., Qiu, F. and Wang, M., 2017. Real-time obstacles detection and status classification for collision warning in a vehicle active safety system. *IEEE Transactions on intelligent transportation systems*, *19*(3), pp.758-773.

[189] Tsai, C.C., Chang, C.W. and Tao, C.W., 2018. Vision-Based Obstacle Detection for Mobile Robot in Outdoor Environment. *Journal of Information Science & Engineering*, *34*(1).

[190] Zhang, Y., Su, Y., Yang, J., Ponce, J. and Kong, H., 2018. When Dijkstra meets vanishing point: a stereo vision approach for road detection. *IEEE transactions on image processing*, *27*(5), pp.2176-2188.

[191] Gupta, B., Chaube, A., Negi, A. and Goel, U., 2017. Study on Object Detection using Open CV-Python. *International Journal of Computer Applications*, *162*(8), pp.17-21.

[192] Kaneko, N., Yoshida, T. and Sumi, K., 2017. Fast Obstacle Detection for Monocular Autonomous Mobile Robots. *SICE Journal of Control, Measurement, and System Integration*, *10*(5), pp.370-377.

[193] Lee, T.J., Yi, D.H. and Cho, D.I., 2016. A monocular vision sensor-based obstacle detection algorithm for autonomous robots. *Sensors*, *16*(3), p.311.

[194] Anvaripour, M. and Ebrahimnezhad, H., 2015. Accurate object detection using local shape descriptors. *Pattern Analysis and Applications*, *18*(2), pp.277-295.

[195] Yu, H., Chang, Y., Lu, P., Xu, Z., Fu, C. and Wang, Y., 2014. Accurate object detection with a discriminative shape model. *Optik*, *125*(15), pp.4102-4107.

[196] Ess, A., Schindler, K., Leibe, B. and Van Gool, L., 2010. Object detection and tracking for autonomous navigation in dynamic environments. *The International Journal of Robotics Research*, *29*(14), pp.1707-1725.

[197] Murphy, K., Torralba, A., Eaton, D. and Freeman, W., 2006. Object detection and localization using local and global features. In *Toward Category-Level Object Recognition* (pp. 382-400). Springer, Berlin, Heidelberg.

[198] Nistér, D., Naroditsky, O. and Bergen, J., 2006. Visual odometry for ground vehicle applications. *Journal of Field Robotics*, *23*(1), pp.3-20.

[199] Manduchi, R., Castano, A., Talukder, A. and Matthies, L., 2005. Obstacle detection and terrain classification for autonomous off-road navigation. *Autonomous robots*, *18*(1), pp.81-102.

[200] Ohya, I., Kosaka, A. and Kak, A., 1998. Vision-based navigation by a mobile robot with obstacle avoidance using single-camera vision and ultrasonic sensing. *IEEE transactions on robotics and automation*, *14*(6), pp.969-978.

[201] Kümmerle, R., Ruhnke, M., Steder, B., Stachniss, C. and Burgard, W., 2015. Autonomous robot navigation in highly populated pedestrian zones. *Journal of Field Robotics*, *32*(4), pp.565-589.

[202] Leibe, B., Leonardis, A. and Schiele, B., 2008. Robust object detection with interleaved categorization and segmentation. *International journal of computer vision*, *77*(1-3), pp.259-289.

[203] Geronimo, D., Lopez, A.M., Sappa, A.D. and Graf, T., 2009. Survey of pedestrian detection for advanced driver assistance systems. *IEEE transactions on pattern analysis and machine intelligence*, *32*(7), pp.1239-1258.

[204] Buch, N., Velastin, S.A. and Orwell, J., 2011. A review of computer vision techniques for the analysis of urban traffic. *IEEE Transactions on Intelligent Transportation Systems*, *12*(3), pp.920-939.

[205] Busquets, D., Sierra, C. and De Màntaras, R.L., 2003. A multiagent approach to qualitative landmark-based navigation. *Autonomous Robots*, *15*(2), pp.129-154.

[206] Brkic, K., 2010. An overview of traffic sign detection methods. *Department of Electronics, Microelectronics, Computer and Intelligent Systems Faculty of Electrical Engineering and Computing Unska*, *3*, p.10000.

[207] Royer, E., Lhuillier, M., Dhome, M. and Lavest, J.M., 2007. Monocular vision for mobile robot localization and autonomous navigation. *International Journal of Computer Vision*, *74*(3), pp.237-260.

[208] Shinzato, P.Y., Grassi, V., Osorio, F.S. and Wolf, D.F., 2012, June. Fast visual road recognition and horizon detection using multiple artificial neural networks. In *2012 IEEE Intelligent Vehicles Symposium* (pp. 1090-1095). IEEE.

[209] Yebes, J.J., Bergasa, L.M. and García-Garrido, M., 2015. Visual object recognition with 3D-aware features in KITTI urban scenes. *Sensors*, *15*(4), pp.9228-9250.

[210] Oh, S.I. and Kang, H.B., 2017. Object detection and classification by decision-level fusion for intelligent vehicle systems. *Sensors*, *17*(1), p.207.

[211] Zhang, H., Hernandez, D.E., Su, Z. and Su, B., 2018. A low cost vision-based road-following system for mobile robots. *Applied Sciences*, *8*(9), p.1635.

[212] Fu, H., Xiang, M., Ma, H., Ming, A. and Liu, L., 2011, October. Abandoned object detection in highway scene. In *2011 6th International Conference on Pervasive Computing and Applications* (pp. 117-121). IEEE.

[213] Lee, J.T., Ryoo, M.S., Riley, M. and Aggarwal, J.K., 2007, September. Real-time detection of illegally parked vehicles using 1-D transformation. In *2007 IEEE Conference on Advanced Video and Signal Based Surveillance* (pp. 254-259). IEEE.

[214] Ikeda, H., Kaneko, Y., Matsuo, T. and Tsuji, K., 1999, October. Abnormal incident detection system employing image processing technology. In *Proceedings 199 IEEE/IEEJ/JSAI International Conference on Intelligent Transportation Systems (Cat. No. 99TH8383)* (pp. 748-752). IEEE.

[215] Sharma S., Gupta L.R. and Dubey M.K., 2019, December. A Review on Augmented Reality based Autonomous Navigation System. In *International Conference on Advances in Computing, Communication and Automation*. IEEE.

[216] Sharma S., Gupta L.R. and Dubey M.K., 2019. Autonomous Vehicle Drive through Advanced Navigation Marker Identification and Categorization System: ANMIC. *International Journal of Advanced Science and Technology*.

[217] Sharma S., Gupta L.R. and Dubey M.K., 2019. Intelligent Marker Repository for Augmented reality based Autonomous Navigation. *International Journal of Control and Automation.*

LIST OF PUBLICATIONS

1. Sharma, S., Gupta, L.R. and Dubey, M.K., 2019. Autonomous Vehicle Drive through Advanced Navigation Marker Identification and Categorization System: ANMIC. *International Journal of Advanced Science and Technology*, *27*(1), pp.338-352. (Scopus Indexed)
2. Sharma, S., Gupta, L.R. and Dubey, M.K., 2019. Intelligent Marker Repository for Augmented reality based Autonomous Navigation. *International Journal of Control and Automation*, *12*(5), pp.271-277. (Scopus Indexed)
3. Sharma, S., Gupta, L.R. and Dubey, M.K., 2019, November. A Review on Augmented Reality based Autonomous Navigation System. In *International Conference on Advances in Computing, Communication and Automation*. IEEE. (Scopus Indexed)
4. Sharma, S., Gupta, L.R. and Dubey, M.K., 2020. Image Processing Mechanism for Augmented Reality based Navigation in Autonomous Vehicles. In *International Conference on Intelligent Engineering and Management*. IEEE. (Accepted) (Scopus Indexed)
5. Sharma, S., Gupta, L.R. and Dubey, M.K., 2020. A Review on Emerging Technologies in Augmented Reality Navigation Systems. *Our Heritage*, 68(30), pp.11730-11744. (UGC Listed)
6. Sharma, S, Kalia, N. R. and Gupta, L.R., 2016. Traffic Road Sign Detection And Recognition Using Geometric Shapes And Background Color: Laying A Foundation To Use Augmented Reality (A.R.) In Autonomous Vehicle Navigation And Decision Making. *Research Journal of Computer and Informational Technology Sciences,5,pp.17-20.*
7. Sharma, S., Gupta, L.R. and Dubey, M.K., 2020. Real Time Stationary Obstacle Detection Algorithm based on Geometric Parameters for Navigation of Autonomous Vehicle, *Multimedia Tools and Applications.* (Communicated).
8. Sharma, S., Gupta, L.R. and Dubey, M.K., 2020. Marker Dataset for Augmented Reality Based Applications. *Information Sciences*. (Communicated).