ISTANBUL TECHNICAL UNIVERSITY
FACULTY OF COMPUTER AND INFORMATICS

RULE BASED COLLISION AVOIDANCE SYSTEM FOR VEHICLES

Graduation Project

Bayram Barış Sarı
150130123

Çiğdem Duvacı
150120301

Department: Computer Engineering
Division: Computer Engineering

Advisor: Prof. Dr. Uluğ Bayazıt

July 2018

Özgünlük Bildirisi

1. Bu çalışmada, başka kaynaklardan yapılan tüm alıntıların, ilgili kaynaklar referans gösterilerek açıkça belirtildiğini,
2. Alıntılar dışındaki bölümlerin, özellikle projenin ana konusunu oluşturan teorik çalışmaların ve yazılım/donanımın tarafımdan yapıldığını bildiririm.

İstanbul, 30.07.2018

Bayram Barış Sarı　　　　Çiğdem Duvacı

Acknowledgments

We would like to thank our advisor Prof. Dr. Uluğ Bayazıt for his advices and support through the project.

We would like to thank Istanbul Technical University, Faculty of Computer and Informatics for giving us the opportunity to improve our vision, and for educating us as a good computer engineer.

We would like to thank Res. Assis. Yusuf Hüseyin Şahin for his support when we had a trouble about optical flow vectors. He helped us to correct the parameters of Farneback function, and that's how we managed to continue our project.

We would like to thank our friend, Ozan Yakar for helping us to correct the mathematical equations.

Lastly, we would like to thank our families and our friends for their support throughout this long period of our graduation project.

Rule Based Collision Avoidance System For Vehicles

(Summary)

With the development of technology, cameras have become an essential part of our lives day by day. Cameras have been widely used in almost every field of our lives. The images were only stored on the film earlier but they are now taken digitally and stored in electronic media. Therefore, images can be processed in various ways in electronic environment. In this way, images are used in a wide variety of fields and purposes. For example, video recordings are used to capture criminals, medical imaging devices to detect diseases, as well as vehicle license plate identification and speed detection, face recognition technology in traffic, and many more. By virtue of these technological improvements, the dependency to the camera has increased.

Understanding the events leading up to a car crashes is meaningful in the prevention of the traffic accidents. Some researchers are expressed that most of the accidents occur because of the faults related to drivers. With that objective, vehicle related crash avoidance technologies were needed to improve. Nowadays, number of the traffic accidents are reduced by means of these technologies such as forward collision warning systems, intelligent engine brake systems, trace control systems, lane departure warning systems, blind spot warning systems and further.

Collision avoidance systems are aimed at reducing the effects and the number of accidents in roads. Therefore, it is aimed to reduce the rate of the number of casualties experienced and to decrease injuries and loss of life along the year.In these systems, it is significant to receive the data correctly. The data taken according to the ambient conditions with various sensors have been processed and a certain number of accident has been prevented using these technologies.

The objective of this project is to design a system that can warn the driver in the event of dangerous situations that may be encountered while driving. A camera located on the front of the vehicle to take images for the detection of motion in surrounding area on road. If the danger is detected by system, driver will be warned to avoid any accident.

For this purpose, two toy car models were used for shooting dangerous and non-dangerous situations. The images were taken with a mobile phone camera and the data set was created. After all data were collected, the data were transferred to the computer environment and the images were subjected to necessary pre-processing. OpenCV and Numpy libraries have been used predominantly in Python for processing images in the PC environment.As a result of the investigations, motion vectors are obtained by using Farneback optical flow function.Under certain assumptions, the states that the movement indicates a danger and the cases that it does not indicate are determined as a rule. After that, the collision field on the image is determined. The speed at which the predicted moving vehicle is estimated by the current optical flow has been used in various calculations and it has been determined that the vehicle will enter the collision zone. Driver have been warned in dangerous situation in order that he/she could take precautions.

Araçlar İçin Kural Tabanlı Çarpışma Önleme Sistemi

(Özet)

Teknolojinin gelişmesiyle, kameralar gün geçtikçe yaşamımızın ayrılmaz bir parçası haline gelmiştir. Hayatımızın hemen hemen her alanında kameralar yaygın olarak kullanılmaya başlanmıştır. İlk zamanlarda, film üzerine kaydedilerek saklanan görüntülerin yerini şimdilerde sayısal olarak kaydedilip elektronik ortamlarda saklanabilen görüntüler almıştır. Sayısal ortamdaki görüntüler bilgisayar ortamına çeşitli şekillerde işlenebilmektedir. Bu sayede görüntüler çok sayıda çeşitli alanda ve amaçta kullanılmaktadır. Örneğin, video kayıtlarından suçluların yakalanmasından, medikal görüntüleme cihazları aracılığıyla hastalıkların tespitine, bunun yanı sıra trafikte araç plaka tespiti ve hız tespiti, yüz tanıma teknolojisi gibi daha birçok alanda yararlanılmaktadır. Bu teknolojik gelişmeler sayesinde kameralara olan bağlılık artmıştır.

Kameralar günlük hayatımızın her alanında çeşitli amaçlar doğrultusunda yer almaya başlamıştır. Aynı zamanda teknolojinin getirdiği yeniliklerden biri de günlük hayatta sık sık kullandığımız ulaşımımızı kolaylaştıran araçlarımızdır. Son zamanlarda, trafikteki araç yoğunluğunun artmasıyla trafik kazalarının yaşanması kaçınılmaz hale gelmiştir. Trafik kazalarına sebep olan olayların anlaşılması, trafik kazalarının önlenmesi açısından önem kazanmaktadır. Bazı araştırmacılar, kazaların birçoğunun sürücülerin hataları yüzünden meydana geldiğini belirtmektedir. Bu sebeple, araçla ilgili çarpışma önleyici teknolojilerin geliştirilmesi gerekiyordu. Günümüzde öndeki araca çarpma uyarı sistemleri, akıllı motor freni sistemleri, takip kontrol sistemleri, şerit izleme uyarı sistemleri, kör nokta uyarı sistemleri ve daha ileri teknolojiler trafik kazalarının önlenebilirliğine katkıda bulunmaktadır.

Çarpışma önleme sistemleri sayesinde trafikte oluşan kazaların ve etkilerinin azaltılması amaçlanmaktadır. Dolayısıyla, yıl içinde yaşanan kazaların ve yaralıların, can kaybı sayısında azalma olması hedeflenmektedir. Bu sistemlerde verinin en doğru bir şekilde elde edilmesi çok önemlidir. Bu yüzden, çeşitli sensörler ile ortam koşullarına göre alınan veri işlenerek kazaların önüne belli bir oranda geçilmiştir. Bu tür sistemler, sürücü müdahalesine gerek duymadan veya sürücüyü önlem alması için uyarı gönderecek şekilde tasarlanır.

Bu projenin amacı, araç hareket halinde iken aracın ön kısmında bulunan kameradan alınan görüntüler kullanılarak izlenilen yol üzerinde karşılaşılabilecek tehlikeli durumların öncesinde sürücüyü uyarabilen bir sistem tasarlamaktır. Böylelikle, sürücü uyarılarak herhangi bir kaza durumundan kaçınılması hedeflenmektedir.

Bu doğrultuda, iki oyuncak araba model olarak kullanılarak tehlikeli olan ve tehlikeli olmayan durumlar için çekimler yapıldı. Görüntüler telefon kamerası ile çekilerek veri seti oluşturuldu. Bu veriler toplandıktan sonra veriler bilgisayar ortamına aktarılarak görüntüler gerekli ön işlemlerden geçirilmişlerdir. Bilgisayar ortamında görüntülerin işlenmesi için Python üzerinde openCV ve Numpy kütüphaneleri ağırlıklı olarak kullanılmıştır. Hareket tespiti ve tahmini için optik akış(optical flow) methodlarından yararlanılarak hareket vektörlerinin oluşturulması gerekmiştir. Yapılan araştırmalar sonucunda Farneback optik akış fonksiyonundan yararlanılarak hareket vektörleri elde edilmiştir. Daha sonra, bazı varsayımlar altında hareketin tehlike belirttiği ve be-

lirtmediği durumlar kural olarak kararlaştırılmıştır. İlk olarak, görüntü üzerindeki çarpışma alanı belirlenmiştir. Tehlike olarak öngörülen hareketli aracın o anki optik akış ile hesaplanan hızı, çeşitli hesaplamalarda kullanılmıştır ve aracın çarpışma bölgesine girip girmeyeceği tespit edilmiştir. Tehlikeli durumumlarda sürücünün önlem alması için uyarı gönderilmiştir.

Contents

1 Introduction

Vehicles have an important place in people's life. People need to use vehicles for traveling, going to work, going to school, visiting someone and such many more purposes. The most used vehicle by people is car. Along with the increased use of cars on the roads, traffic accidents have become now inevitable. Some studies show that every year millions of people are injured, killed or disabled in car accidents. Understanding the events leading up to a car crashes is meaningful in the prevention of the traffic accidents. According to "Transportation Accident Statistics 2013 Report" published by the Ministry of Transport, Maritime Affairs and Communication, traffic accidents which are result in an average of 2866 physical damage and 442 deaths and injuries occur in a day. 88.7 percent of traffic accidents were driven by flaws of drivers, 9 percent of them caused by pedestrians while only 1 percent of the accidents were caused by road defects. Researches are proved that the reason behind the most of the car accidents are the faults related to drivers. Alan Amici, who is a vice president of automotive engineering at TE, explains "Recent NHTSA research shows that approximately 94 percent of accidents are caused by human error," [1]. As the technology improves, various in-vehicle technologies and sensors begin to gain ground for preventing these accidents.

Cars today have many semi-autonomous features like adaptive cruise control, assisted parking and self-braking systems. These features make use of radars, ultrasound sensors, thermal imaging cameras, laser scanner as well as the front camera to guide the driver and the car. It is believed that cars with advanced safety features cause to decimate in the rate of the deaths and injuries were caused by the accidents.

In this project, a single camera is used to record a video for estimation of the movement of the vehicles in order to help avoid various risks that may arise during driving. It is possible to verify the data manually and analyze the data using computer vision methods autonomously. The methods are implemented on PC environments. The motion of the vehicles will be detected from the images using computer vision techniques initially. The images captured by the camera will be processed by the computer vision techniques. The method based on optical flow is used to detect moving object.

After all, depending on the movement of the object, the system detects the critical situations involving other vehicles or pedestrians ahead of or to the side of the vehicle. Rule based system is created, and used to determine the state of the road. The system will inform the driver as early as possible to prevent any accidents which might occur before the moment of danger so that the driver can take action to avoid a collision. In the Figure 1.1, the flowchart of the collision award system is given.

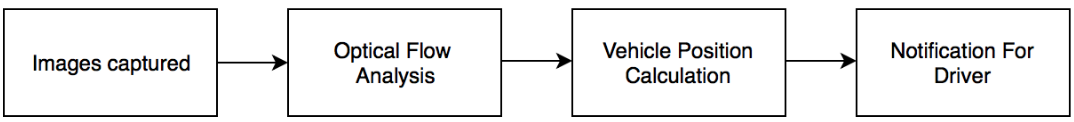

Figure 1.1: The flowchart of the collision avoidance system

With such technology, road traffic fatality rate is expected to lessen considerably. These technologies lead to decline in the number of the accidents as well.

In this report, description of the project and its plan, literature review and previous studies, design and implementation, test and evaluation details are explained. In evaluation, the success of the project is examined for enhance the development results in the future.

2 Project Description and Plan

The section contains the general description and the plan for the project.

2.1 Project Description

The purpose of the project is to classify the image frames from data collected by camera according to level of danger to cause a collision. The system identify the dangerous conditions and notify the driver.

2.1.1 Time Plan

This section contains the work breakdown structure and the time plan of the project. The GANTT diagram of the project can be seen in the Figure 2.1 and Figure 2.2 below.

# of WBS	Task Name	Start Date	Due Date
1	Project Conception and Initiation	05/10/17	27/04/18
1.1	Literature Search	05/10/17	27/04/18
1.1.1	Research of Motion Detection in Computer Vision	16/11/17	27/12/17
1.2	Analysis of Software Requirements	16/12/17	22/01/18
1.3	Interim Report Preparation	12/01/18	09/02/18
1.4	Video Dataset Collection	22/01/18	23/03/18
2	Implementation	09/02/18	13/07/18
2.1	Classification of Image Processing System Design	09/02/18	30/03/18
2.2	Collecting Data Using Image Processing Technique	02/04/18	06/07/18
2.3	Development of the System	02/04/18	13/07/18
2.4	Integration	28/06/18	13/07/18
3	Testing	09/04/18	20/07/18
4	Final Report Preparation	02/07/18	29/0718

Figure 2.1: Tasks of the project

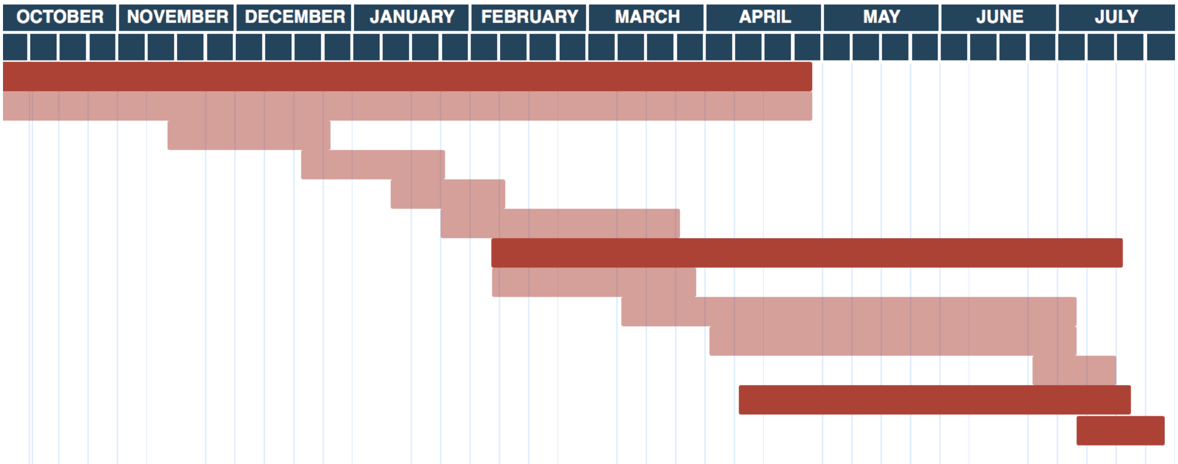

Figure 2.2: Time plan of the project

2.1.2 Work Breakdown Structure

Work Breakdown Structure where main tasks are enlisted and subtasks that build the main task is given in Figure 2.3 below.

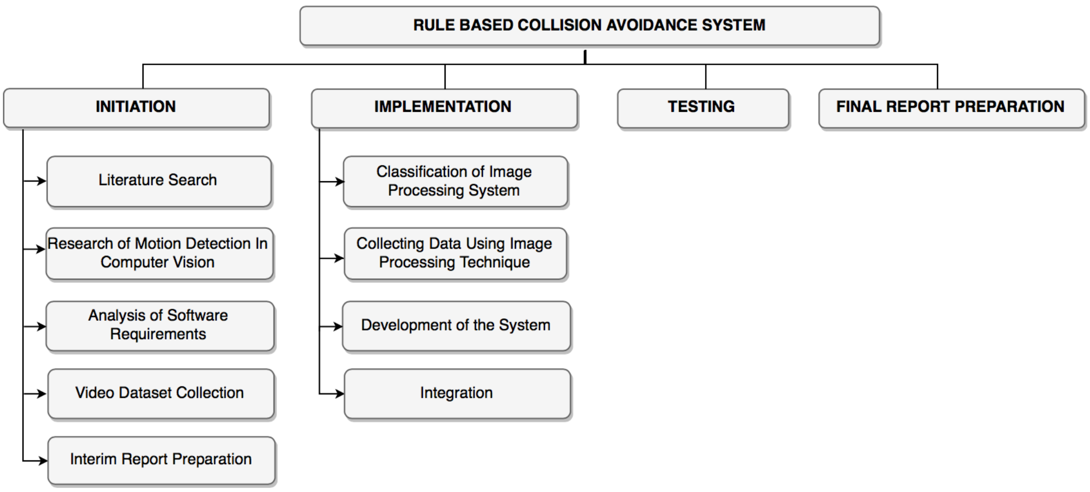

Figure 2.3: Work breakdown structure of the project

3 Background

In this section, general information is given about previous studies of collision assist for vehicles, computer vision, optical flow techniques and machine learning algorithms used along the project.

3.1 Previous Studies of Collision Avoidance System for Vehicles

Recently, the number of the vehicles on the roads have been increased considerably as the global economy grows. Due to this reason, the number of traffic accidents has increased proportionally compared to the increased in the number of the vehicles in the roads. The research and development of the intelligent driver assistance systems and autonomous cars have been come to the forefront for the safety issues on the roads. Large numbers of studies have been done on that score for years. These studies purposed many various approaches and solutions for the problems. Consequently, many types of driver assistance systems have been developed and introduced into the market in order to prevent of the possible traffic accidents and to reduce the risks.

As mentioned previously, taking necessary precautions is crucial to save lives and minimize the number of the people injured on-road accidents. So, distinctive methods and sensors are used to detect the possible crashes. These safety features on vehicles work throughout driving to keep the driver and passengers in safe. They alert the drivers to the dangers ahead and help them to take action before the collision. There are many types of sensors widely used in driver assistance systems. The technology systems anticipate possible collisions with any kind of obstacles, recognize them using such as radars, cameras, laser scanners or ultrasonic sensors. These sensors vary according to their purpose of usage. According to the massive data obtained from the sensors, the driver is warned by the system to prevent or mitigate collision occurring while on the road.

As stated in statistical data by the United Nations, more than 1 million people lost their lives in traffic accidents each year in the world. Researches have shown driver inattentions is one of the prominent factors which cause traffic accidents [2]. It has been expressed that the most common sensors used in collision avoidance systems are radars and cameras which are monitoring the road ahead or behind in the traffic. The driver is given notice of dangerous situations. The reason why radars are used frequently despite their cost is that they have more precise, more robust and higher sensing capabilities [3]. An affordable camera vision can anticipate vehicles, pedestrians, traffic lanes in close-range, it is feasible to use for collision avoidance systems on that account [4]. Since camera vision system and passive infrared is not suitable to measure the distance between the vehicle and object ahead. In addition to being budget-friendly, ultrasonic radars can produce highly accurate results in a short range of distance. As a result of this, more than one sensor can be used to complement each other in such systems [5].

3.2 Computer Vision

In computer vision, an image frame or more than one image frames can be identified with different techniques in order to obtain any kind of useful information according to relevant needs through their transferred data in electronic environment. Computer vision techniques are currently used in a quite broad range of field. Application areas can be listed as face recognition technology, security systems, surveillance systems, understanding satellite images in military industry, radiology in medicine and so on. One of the applications is the motion detection and estimation across video frames that can be processed and analyzed using utility of optical flow. Majority of the technologies on vehicle collision warning and detection systems have offered using passive camera sensors. Naveen Onkarappa offered to carry out an optical flow method which can be applied on driver assistance systems for object recognition [6]. Elisa Martinez introduced a system for vehicle collision detection using optical flow combined with time-to-contact (TTC) which is an estimation of time to hit by other object [7]. However, the method which Elisa introduce cannot provide detection of object.

3.2.1 Optical Flow

One of the first method used in motion detection is optical flow which provide more accurate and robust results than many other methods. Optical flow analysis has been developed over 30 years. It is also widely used in industrial, military, and UAVs [8][9][10]. Many motion detection methods are based on optical flow methods. The motion can be determined by the change of the position of every single pixel in the two consecutive image frames. The apparent velocity, which occurs on the two-dimensional plane resulting from the object motion being perceived by the visual sensor, is called optical flow and the optical flow field is approximated to the velocity field by the methods used [11]. There are several assumptions while working with optical flow.

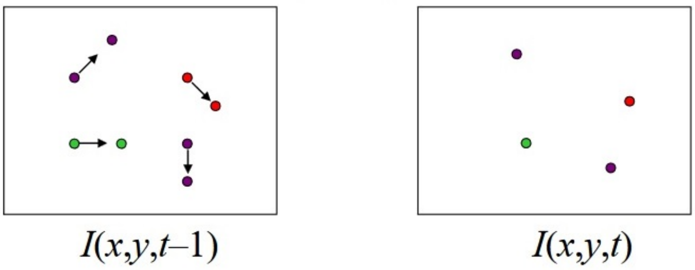

Figure 3.1: Pixel motion from image I(x,y,t-1) to I(x,y,t)[12]

First assumption is brightness constancy. The brightness is assumed stationary over the short time period. The other assumption is the motion between frames should be small[12].

This equation can be found after the figure 3.1 is interpreted:

$$\mathrm{I}(\mathrm{x}, \mathrm{y}, \mathrm{t}+1) = \mathrm{I}(\mathrm{x}+\mathrm{u}\delta \mathrm{t}, \mathrm{y}+\mathrm{v}\delta \mathrm{t}, \mathrm{t}+\delta \mathrm{t}) \qquad [13]$$

In this equation, δt denotes the time difference between two frames, uδt and vδt denote the motion of the pixel at (x,y)[13].

There are two unknown variables. It is not possible to solve this equation while we have only one equation. In order to solve the optical flow problem, new methods and solution are invented and developed. In our case, Farneback method is used to estimate optical flow.

3.2.2 Farneback's Method

Optical flow methods can be divided into two classes, as local and global methods. Local methods compute the vectors by use of specific pixels from the image, whereas all pixels on image are considered in the global methods. Optical flow is computed using the Gunnar Farneback's algorithm which is a global optical flow method. Detection of the direction and speed of a moving object in a video frame or an image can be archived using Farneback method.Farneback's two frames optical flow based on polynomial expansion of a neigbourhood of every pixels in the image [14]. The Farneback method is fast and linear. In addition to that, it has the smallest average error when it is compared with other optical flow algorithms [15].

Gunnar Farneback expressed quadratic polynomials which generates the local signal model represented in a local coordinate system. The formula below represents the approximation of each pixel neighbourhood by polynomial where A is a symmetric matrix, b a vector and c a scalar.

$$\mathrm{f(x)} \sim \mathrm{x^T A x + b^T x + c} \qquad [15]$$

The parameters are computed by a Gaussian weighted least squares approximation of the signal. By computing the neighborhood polynomials on two subsequent images, the displacement d can directly be acquired under ideal translation. Image $f_1(x)$ is taken at time t and $f_2(x)$ at time (t + dt). If $f_1(x)$ and $f_2(x)$ is determined, d[d_1,d_2] may be calculated. Here, d implies the flow of the pixels. In equations below, $f_2(x)$ is constructed using f_1(x - d) where d is global displacement.

$$\begin{aligned} \mathrm{f_2(x)} &= \mathrm{f_1(x-d) = (x-d)^T A_1 (x-d) + b_1^T (x-d) + c_1} \qquad [16] \\ &= \mathrm{x^T A_1 x + (b_1 - 2A_1 d)^T x + d^T A_1 d - b_1^T d + c_1} \\ &= \mathrm{x^T A_2 x + b_2^T x} \end{aligned}$$

Thus, the coeffients in the two polynomial gives the formulas below as assumption of brightness constancy is fulfilled.

$$\mathrm{A_2 = A_1,}$$

$$\mathrm{b_2 = b_1 - 2A_1 d}$$

We can obtain d ,which is the value of displacement, from the equation for b_2. As shown in equation below:

$$d = -\frac{1}{2}A_1^{-1}(b_2 - b_1)$$

4 Design and Implementation

This section contains the design ad implementation steps of the project.

4.1 Data Collection

In the data collection part, two consecutive image sequences, which are recorded by Samsung Galaxy 8, were captured from the top of the car. Shutter speed is adjusted 1/180. ISO value is set a lower range between 50 and 300 where exact numbers depends on the lighting of the environment. Manuel focus is selected. Since the object is moving and the motion needs to be captured by camera without noise and blur, the settings of the camera is of paramount importance. Shots were made using several different scenes and environments. The camera that detects the moving objects on the road ahead was mounted on the windscreen of the car, but could not get sufficient depth for the calculation of motion vector. Hence, the camera was repositioned exactly 35 cm above the roof of the vehicle and the shots were made from the scratch. Lastly, the image frames were received as input arguments by optical flow function.

Figure 4.1: First Frame

Figure 4.2: Second Frame

4.2 Optical Flow Calculation

In the optical flow calculation part, the frames were resized from 4032x1960 to 2016x980 and converted to gray-scale before computation of the dense optical flow. For all of the computations on images, openCV and Numpy libraries, which offer variety of image processing functions, were relied on heavily on Python implementation environment. *calcOpticalFlowFarneback* function is used to extract the optical flow vectors per each pixel. The goal of using the function is to track the motion of moving objects with its direction and speed. The detected motion between pixels is denoted by the arrowed lines and stationary points denoted by points over the image. In the Figure 4.3 below,

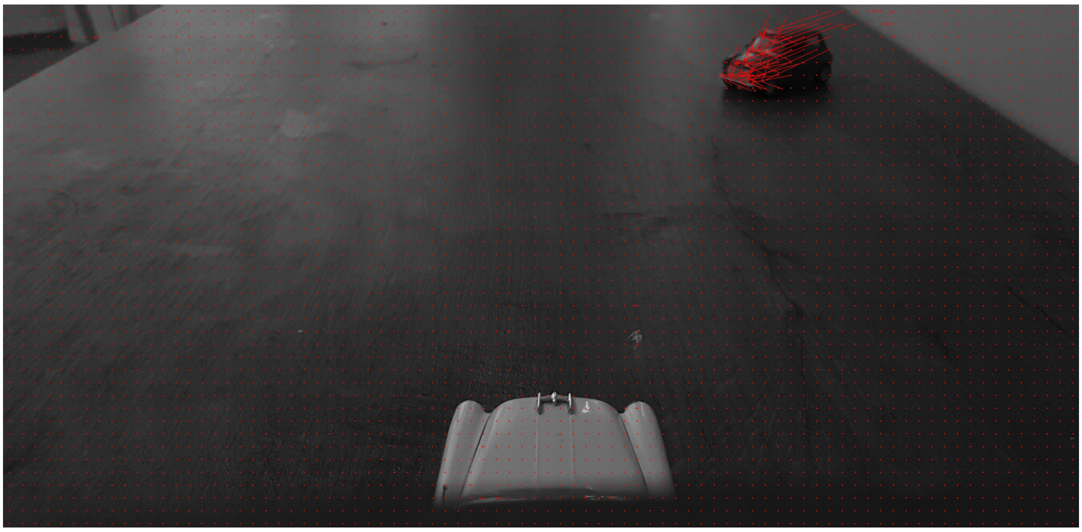

Figure 4.3: Output of calcOpticalFlowFarneback function

calcOpticalFlowFarneback function computes the optical flow using some input parameters. The parameters of the function should be well-selected in order to have good results. These parameters are explained below [17].

- **pyr_scale:** This parameter specifies the image scale (<1) to build pyramids for each image. pyr_scale set as 0.5. This value allows each layer of the parameter to be half the size of the previous layer.
- **levels:** This parameter specifies the number of pyramid layers. This value set as 7. That means 6 extra layers were produced to use in pyramid.
- **winsize:** Window size where the flow is computed. The robustness increase while the winsize set larger values, however the motion field becomes indistinct. This value set as 15.
- **iterations:** Number of iterations the algorithm does at each pyramid level. This value set as 10.
- **poly_n:** This parameter is the size of the pixel neighborhood in order to find polynomial expansion. This value set as 7.
- **poly_sigma:** This parameter is standard deviation of the Gaussian that is used to smooth derivatives used polynomial expansion. For poly_n=7, this value set as 1.5 for better results.

The time unit of the optical flow vectors is frame for this project. Each time, the speed of the vector is calculated as pixel/frame. Afterwards, collision area was determined and drawn on the output image as well. In the next section, the states of collision will be explained.

4.3 Collision Zone

As mentioned previously, the output of Farneback optical flow was obtained. Subsequently, the output is utilized to compute whether there will be a collision or not. However, for the decision of collision or non-collision, it is necessary to create the collision zone, and make a few assumptions first.

The first assumption is the vehicle that has the implemented system goes straight with a constant speed, 10 cm/s. Since it moves with the same speed of the camera, there is no optical flow vectors which are output of Farneback function. However, it has to be considered for the creation of orange zone because its slope depends on where the vehicle is going to be after 3 frames. For finding this position, the constant speed has to be converted to pixel/frame for each time before being drawn. Because of the perspective of the camera, its value decreases gradually in pixel/frame unit. Additionally, the second assumption is the second vehicle that might cause a collision is not in the collision zone in the first frame. The last assumption is that the approaching vehicle goes with the same speed in pixel/frame unit. In order to devise the collision zone, perspective calculation is made, and therefore the constant of speed in pixel/frame will not change the threat.

When the researches that are related to this project are interpreted, it is possible to obtain that at least two cameras are necessary due to finding the depth of the objects in scenery[18][19]. Unfortunately, these kind of shots couldn't be made for this project because of lack of technical conditions. For depth, shots were made when camera was 35 cm above the car. The example scenery for this project is as follows:

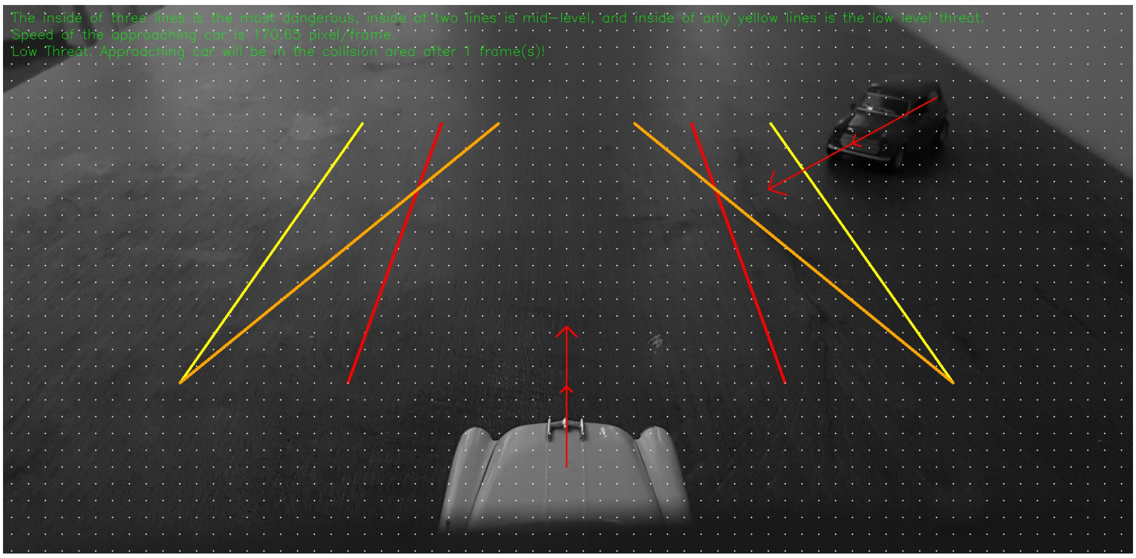

Figure 4.4: Collision Zone

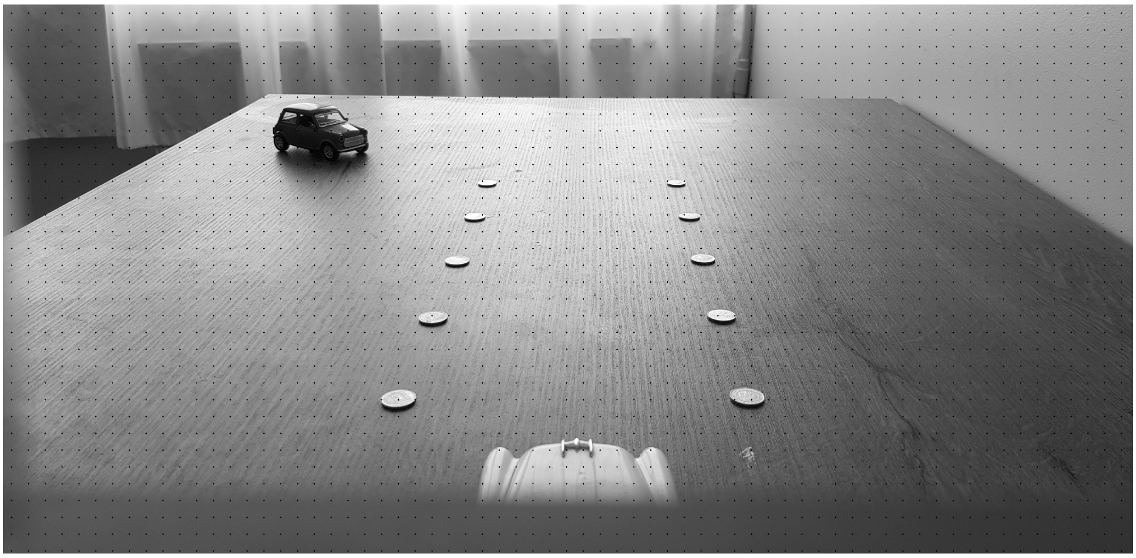

Figure 4.5: 5 Pair of Coins for Perspective Calculation

Before devising the collision zone, researches are made for finding approximately width of a lane, and the ratio of the average width of the cars to width of a lane. For a safe, comfortable driving, 3.7 meters width is the best width for a lane based on many surveys [20][21], and when the average width of a car is considered, it is about 1.7 meters[22]. Thus, the ratio between them is $\frac{1.7}{3.7} = 0.4594$. Since the width of the vehicle used in the experiment was 8 cm, it was found that the width of the lane should be 18 cm. In total 5 pair of coins are placed to represent the lane where the vehicle that has the system in. Each pair are placed 10 cm intervals, and the horizontal distance between a pair coin is 18 cm, i.e. $\frac{10}{0.4594}$. The borders of the lane is drawn in red. For finding the slope of the lines, a perspective calculation is made. For each vertically pair of coins , the real distance is known, 10 cm, yet the distance in the frame changes when the vertically pair of coins are further away. The graph below shows the calculations for the right line of red zone:

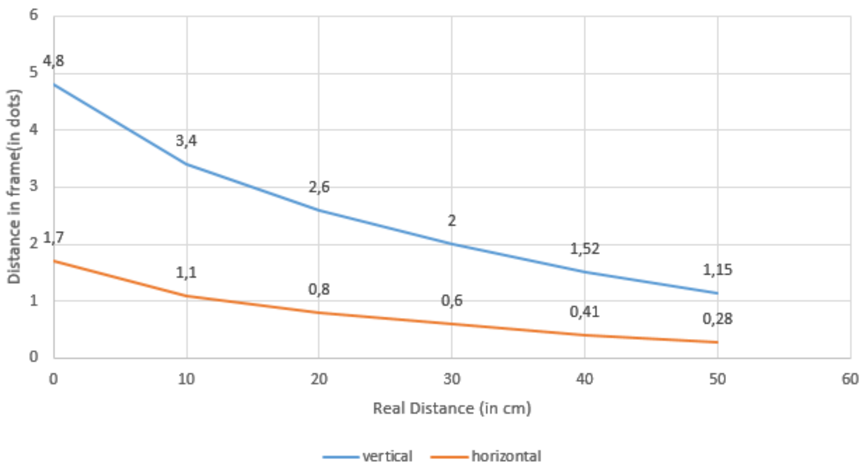

Figure 4.6: Distances between vertically pair of coins

In this graph, it has seen that, the distance between two coins decreases when the pair of coins are far away. For instance, the real distance, i.e. 10 cm, is represented in 4.8 dots, since the distance between two dots is 30 pixels, it is equal to 4.8x30 = 144 pixels. On the other hand, the distance between two coins that are the furthest from the camera is equal to 1.52 dots, or 1.52x30 = 45.6 pixels. When the average slope is computed:

$$\text{avg.slope} = \frac{\frac{1.7}{4.8} + \frac{1.1}{3.4} + \frac{0.8}{2.6} + \frac{0.6}{2} + \frac{0.41}{1.52}}{5} = 0.32$$

This calculation is made for the left line of the red zone, and it is determined that there was a slight slip when shooting, for this reason the slope is found to be -0.40. The averages of 0.32 and 0.40 are taken and the slope is accepted as 0.36. As a result, two lines with -0.36 and 0.36 slopes are drawn in red.

It is also thought that considering only the lane of the vehicle as a collision zone is not a sufficient approach to successfully avoid a collision, and therefore half width of a lane from each side of the lane are considered also in the collision zone. The borders of this area are drawn in yellow. For finding the slope of the yellow lines, the intersection point between yellow and red lines is found, then the vertical and horizontal distances from bottom x,y points of the left yellow line is calculated. Thus the slope is found, and since the other yellow line has the same value with minus sign, two yellow lines are drawn.

Orange lines are also calculated and drawn for more accurate system. Since it is not a proper approach to classify the whole lane as dangerous after 3 frames of time have passed, the start points of the yellow lines and the intersection point on red lines that the vehicle will reach after 3 frames is found. Then, the slope is found and the orange lines are drawn. The figure 4.7 explains the calculation better:

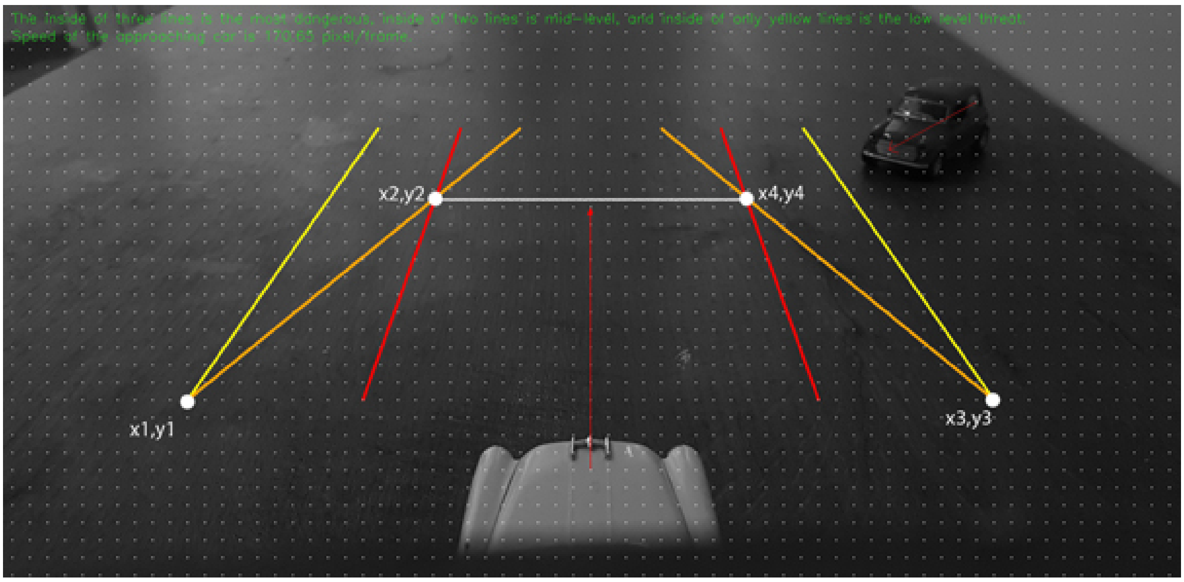

Figure 4.7: Points for orange line calculations

The slope of orange lines is found with the use of (x1,y1) and (x2,y2) pair of points.

White line is drawn according to the position of the vehicle after 3 frames.

As a result,the orange lines divide the threats to three levels; low, middle, and high level. Because when the real-life implementations of vehicle autonomous systems are considered, it is clear that systems takes into account the environment in a conic shape. Taking linear the lane, and its surroundings are not a good approach, and therefore it is not the stated approach in this project.

The devised collision zone is as follows:

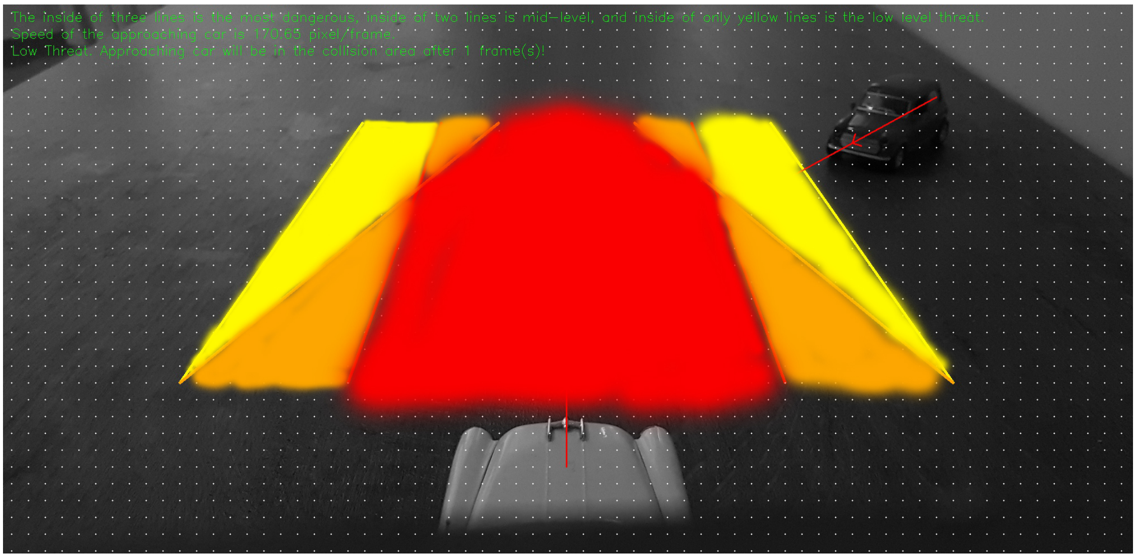

Figure 4.8: Collision Zone

In the figure 4.8, the red zone signs the highest level threat, the orange zone signs the middle level threat, and lastly, the yellow zone signs the low level threat.

For each color of lines, the equations below is computed to understand whether the approaching vehicle enters the collision zone or not.

$$\mathrm{XX}_1 = \frac{(y' - y_3)(x_4 - x_3)}{y_4 - y_3} + x_3 \qquad \mathrm{XX}_2 = \frac{(y' - y_1)(x_2 - x_1)}{y_2 - y_1} + x_1$$

The found x points, XX_1 and XX_2, are the points on the lines that divide the collision zone from the rest of the frame. If the x coordinate of the speed vector is between these two found x points, i.e. $XX_1 < x' < XX_2$, it means that the vehicle will enter the collision zone. These equations and x points can be understood better when the Figure 4.9 is checked:

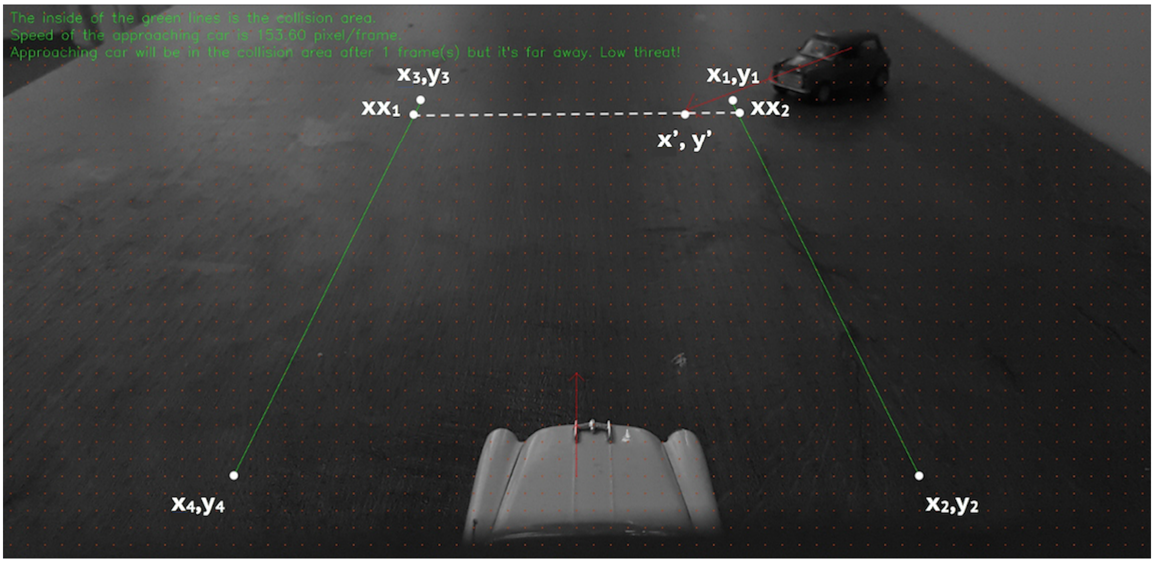

Figure 4.9: Checking the inequality: $XX_1 < x' < XX_2$

Note: This implementation was the previous version of this project. However, it's more easy to show the inequality in this figure, and that's why it is used.

After it is clear that the vehicle will enter the collision zone after a few frames, the level of warning depends on the color zone it enters. If it is between orange and red lines, it means that the situation is too serious because it will be in the most dangerous area. If it is in orange zone but not in red zone, it means that it is a mid level of threat. If it is not between orange or red lines, it means that it is only a low threat.

5 Testing and Evaluation

5.1 Experiments

In this part, more information about the experiments, i.e. collision scenarios, experimental environment, and evaluation criteria, are going to be explained.

5.1.1 Collision Scenarios

When real-life car accidents are considered, it is obvious that there is a lot of possible different accident scenarios. For instance a vehicle can crash into a still object, or when two cars face on a two-way road, it might cause head on collisions, or when a vehicle goes straight, the other approaching uncontrolled from the right or left can cause side-impact collisions. There is also many interesting but not common car accidents that might be an example scenarios. However, this project takes into account of only a few prevalent scenarios that capture thorough most of the possibilities of the car accidents.

As it has explained in the previous parts, after the algorithm had been implemented, one car that approaches from the right corner was tested as first collision case. That car had different route and different pace for testing different scenarios. Then, one car that approaches from the left corner of the screen was tested, and finally cars that go straightly from left or right side was tested. These were instances of collision scenarios. For non-collision situations, one car that goes straight the same direction or the opposite direction with the car that has the system in different lane was tested. Each time, one of the most important thing was to know how far a car was moving.

5.1.2 Experimental Environment

Python's OpenCV and Numpy libraries has been used for implementing all the algorithm. For each scenario, the required scene is created with 2 toy cars being on a large table and a large number of frames are taken by Samsung Galaxy S8. They are taken with pro mode of the camera for avoiding too many details because optical flow vectors of Farneback algorithm are too sensitive to any movements. Shutter is decreased, ISO is arranged due to the sunlight, and finally manual focus is arranged. More than 100 pair of frames are tested and the best 20 pairs are taken for the evaluation of the project.

5.1.3 Evaluation Criteria

In this project, an algorithm is implemented to give warning when there is a possible collision between two cars, and the main objective of this project is to predict the accident before it happens and to make the necessary warning to the driver. Since it is a rule-based systems, finding accuracy for the example scenarios is not enough. Due to this reason, system evaluation has been made based on the following questions:

- **The calculation of the system:** During the design part of the project, it is thought that a calculation up to 5 frames of time passes is enough. However, even though it does not seem from the examples in results section, system might do the calculations until 8 frames of time.
- **Level of dangers of the system:** The implemented system has three different level of threats, it might be seen in Figure 4.8: low level, mid-level and high level of threats.
- **The boundaries of the collision zone:** The collision zone is formed to cover 2 lanes. It covers the width of the lane that the vehicle is going to and the half width of the right and left of that lane. On the other hand, the height of the collision zone is determined based on where the vehicle will be after 8 frames.
- **The sensitivity of the system to the environment:** The environment of the system is crucial for this project. Because in the first part of it, it is necessary to discover convenient optical flow vectors to proceed. Therefore, the shots are made in pro mode to reduce this sensitivity. If the examples in results section are examined, it will be observed that there are very small changes in the environment, with the exception of light.

5.2 Results

In this section, there are 7 examples that the system is successful and 2 examples that the system fails. The results and the success rate are explained in detail after the examples.

Figure 5.1: First Frame

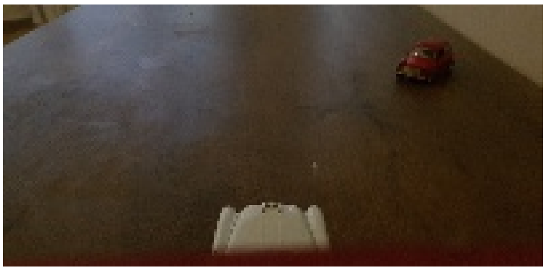

Figure 5.2: Second Frame

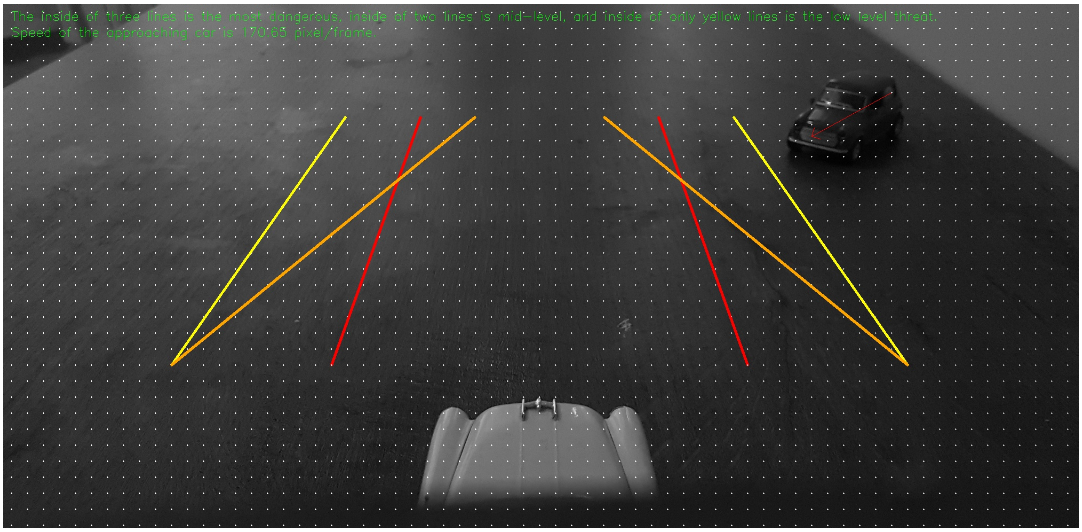

Figure 5.3: Optical Flow Vectors

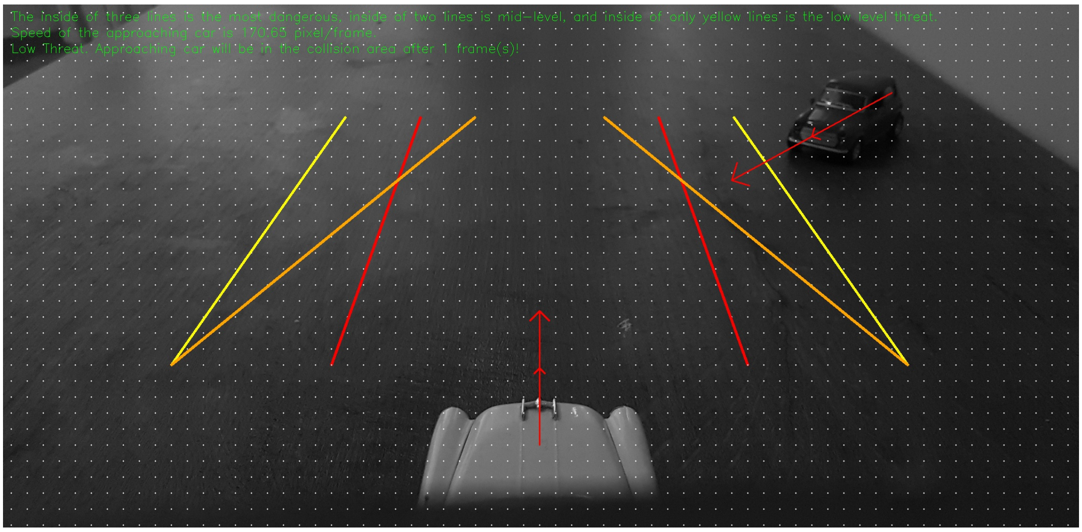

Figure 5.4: Low Level Threat

In this example, the car that approaches from right corner goes with the speed of 170.62

pixels/frame, and it will possibly enter the collision zone after 1 frame. As it has seen obviously in figure 5.4, it will be in the yellow lines but not in orange and red lines. Hence, the system give a low level of warning to the driver.

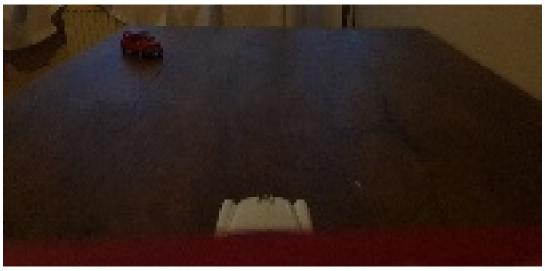

Figure 5.5: First Frame

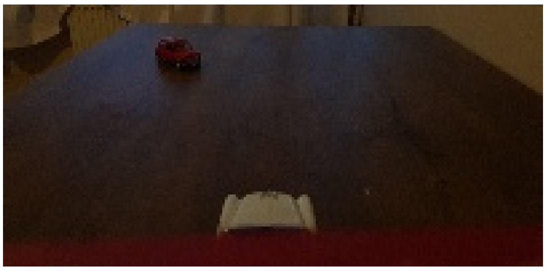

Figure 5.6: Second Frame

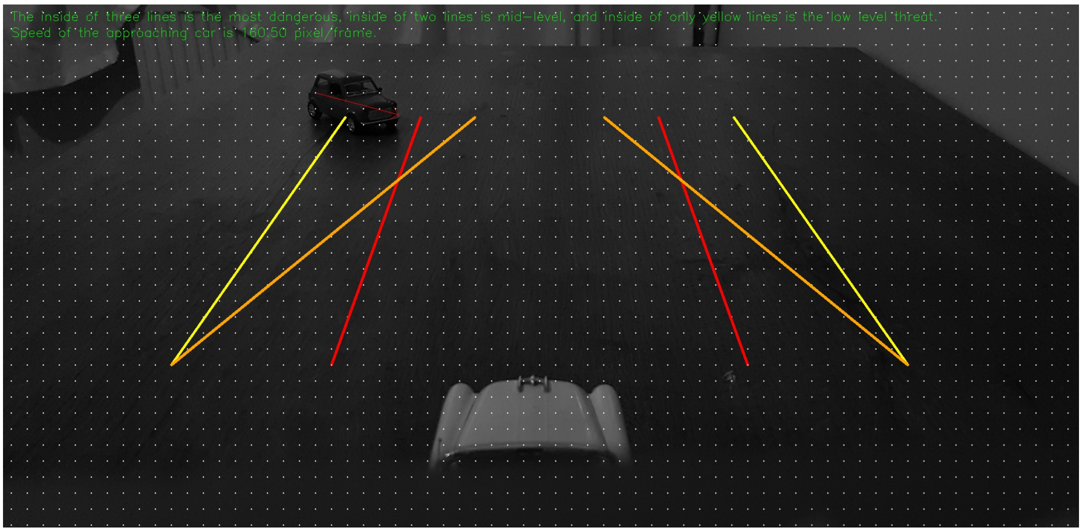

Figure 5.7: Optical Flow Vectors

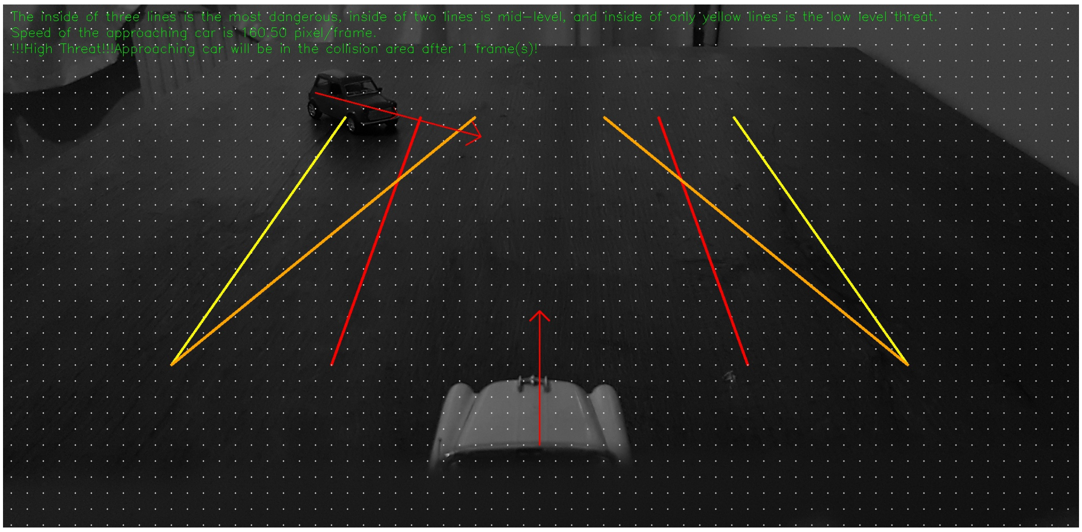

Figure 5.8: High Level Threat

In this example, one vehicle approaches from left corner with 160.5 pixels/frame speed. Just after one frame passes, it enters the collision zone. Due to the finding that it will be between the left three x points and right three x points, i.e. it will be in the intersection of three zones, it will be in the most dangerous zone. So, it is a high level threat situation.

Figure 5.9: First Frame

Figure 5.10: Second Frame

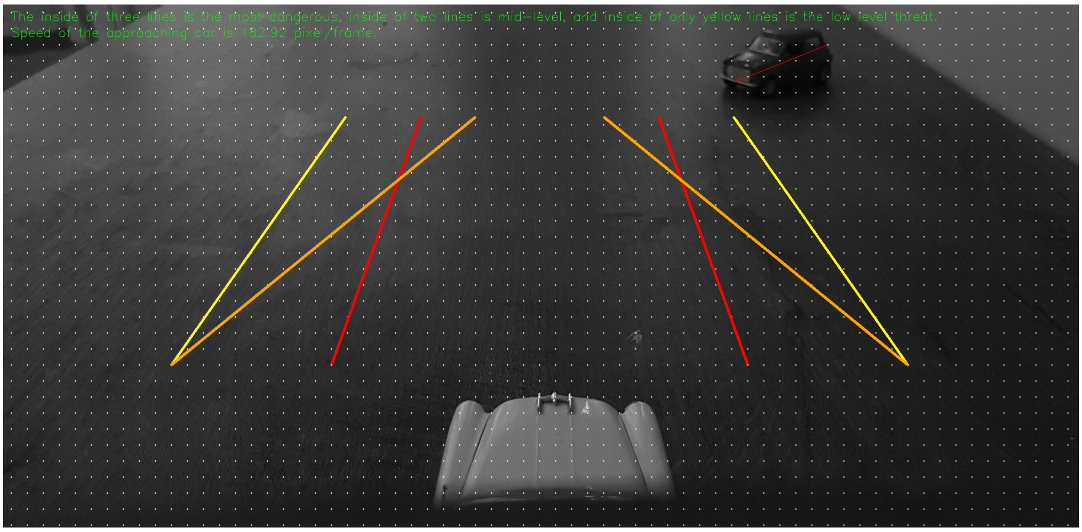

Figure 5.11: Optical Flow Vectors

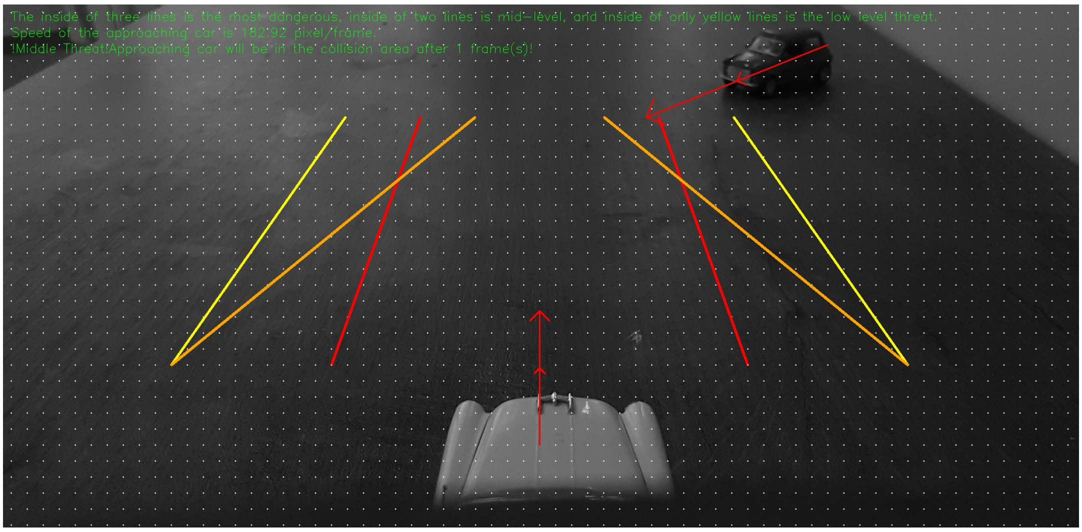

Figure 5.12: Mid-level Threat

This example is used to test whether the system gives the mid-level warning when the speed vector of the vehicle is between the yellow and red lines but not between the orange lines. The approaching vehicle has the speed of 182.9 pixels/frame. After 1 frame passes, it will be in the collision zone, in particular, in the of orange zone.

The scenarios above are examples of a scenario that one car approaches from left or right corner to the middle. They are used to test head-on collision scenarios.

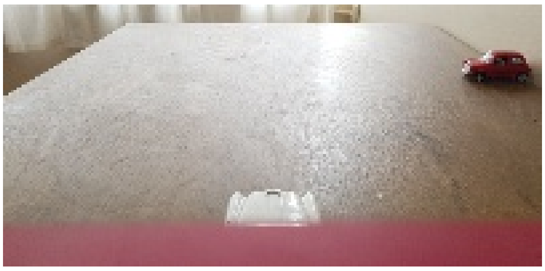

Figure 5.13: First Frame

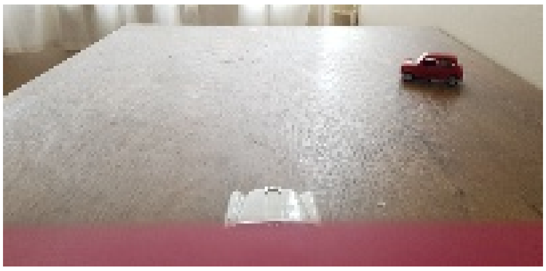

Figure 5.14: Second Frame

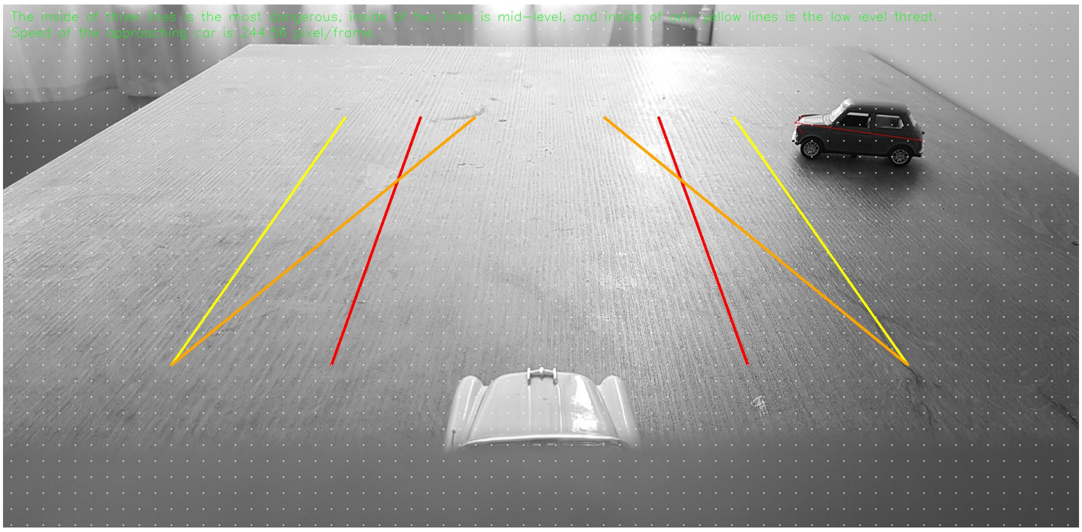

Figure 5.15: Optical Flow Vectors

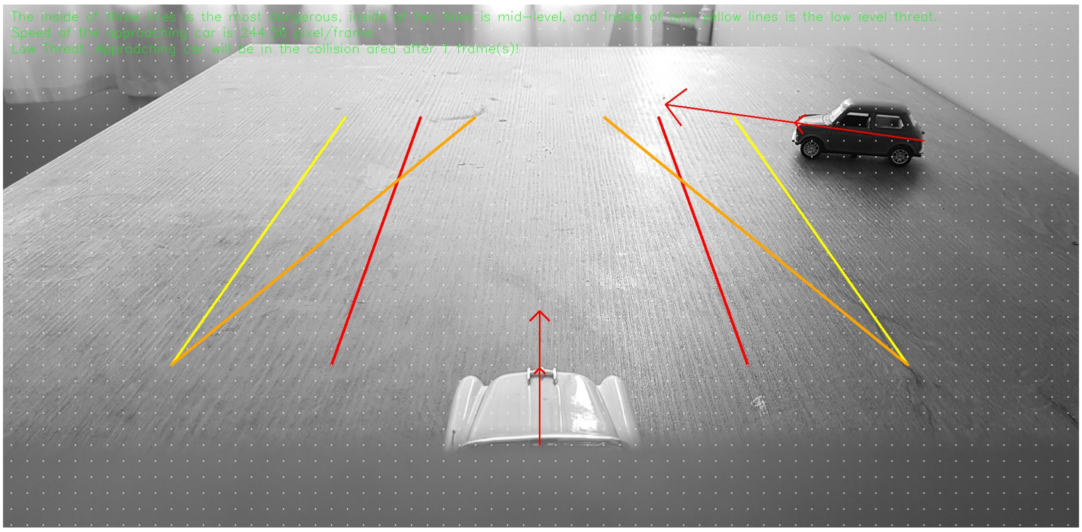

Figure 5.16: Low Level Threat

This is another example of collision scenarios. One vehicle goes from right side to left side with a high speed, about 244.56 pixels/frame. It enters the collision zone but it is a low level warning situation because it will enter the yellow zone of collision zone.

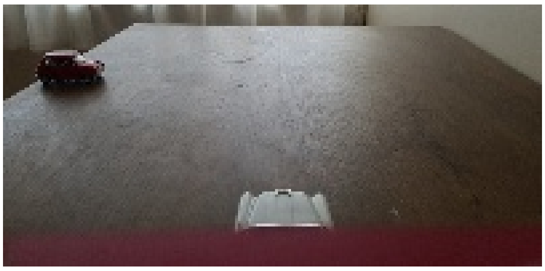

Figure 5.17: First Frame

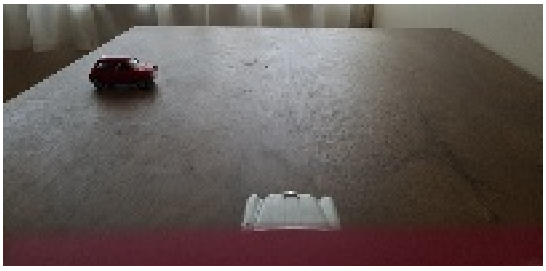

Figure 5.18: Second Frame

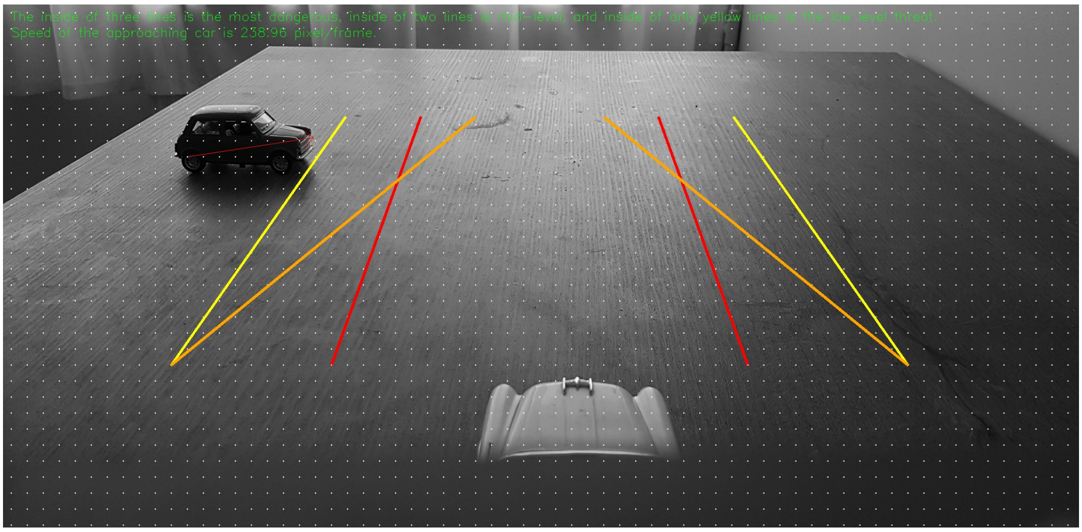

Figure 5.19: Optical Flow Vectors

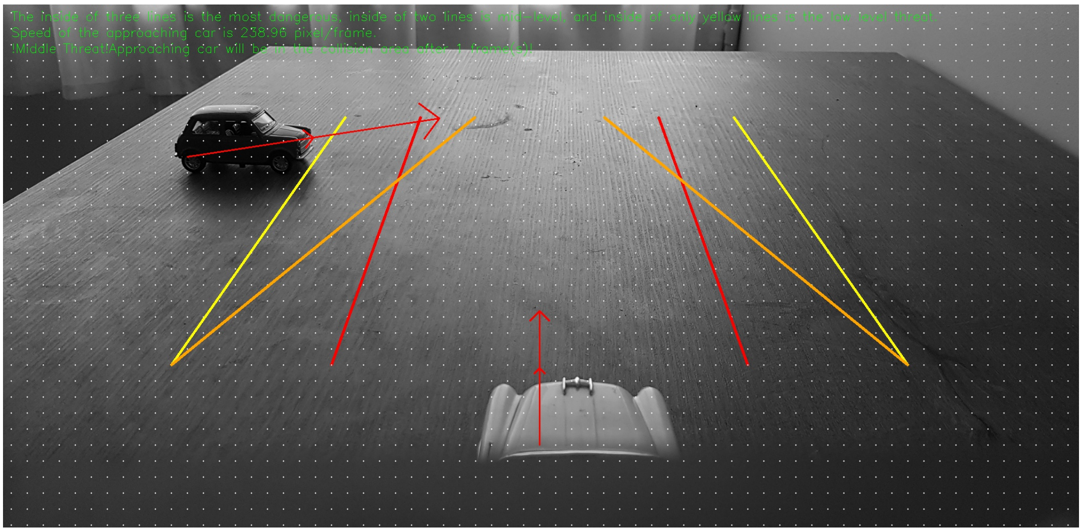

Figure 5.20: Mid-Level Threat

This scenario is similar with the previous one, only the side of the vehicle changes. The speed of the vehicle is approximately 239 pixels/frame. However, the approaching vehicle starts its move closer than the previous example, and the result of that, it will enter the mid-level threat zone, i.e. orange zone after one frame of time passes.

These two scenarios above are used to test side impact collision scenarios. One car approaches from the right or left side, yet it is possible to say that the route of it is not through the route of the car that has the implemented system because system gives only low and mid-level warnings.

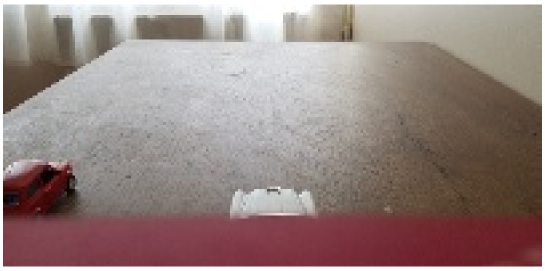

Figure 5.21: First Frame

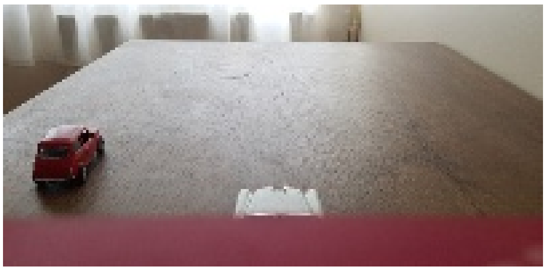

Figure 5.22: Second Frame

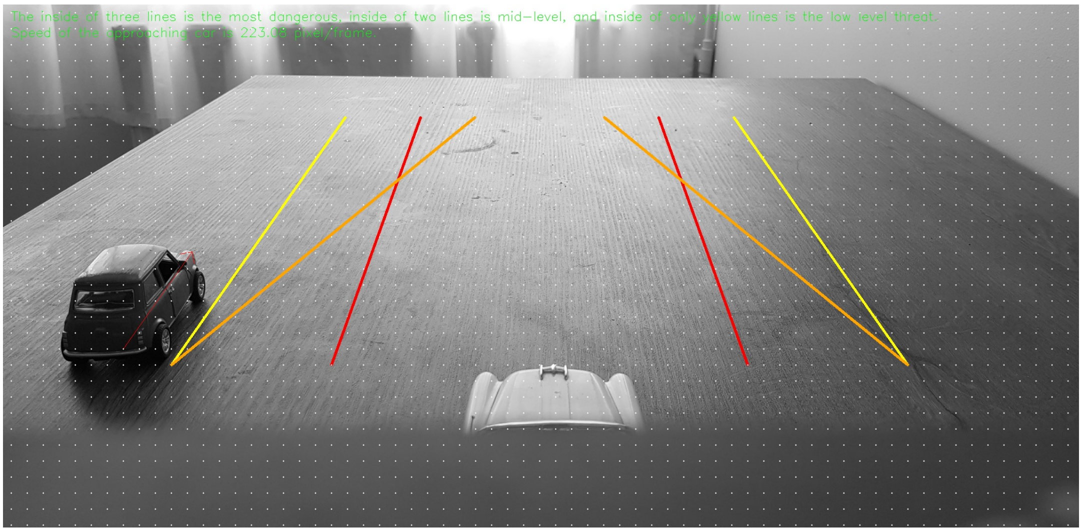

Figure 5.23: Optical Flow Vectors

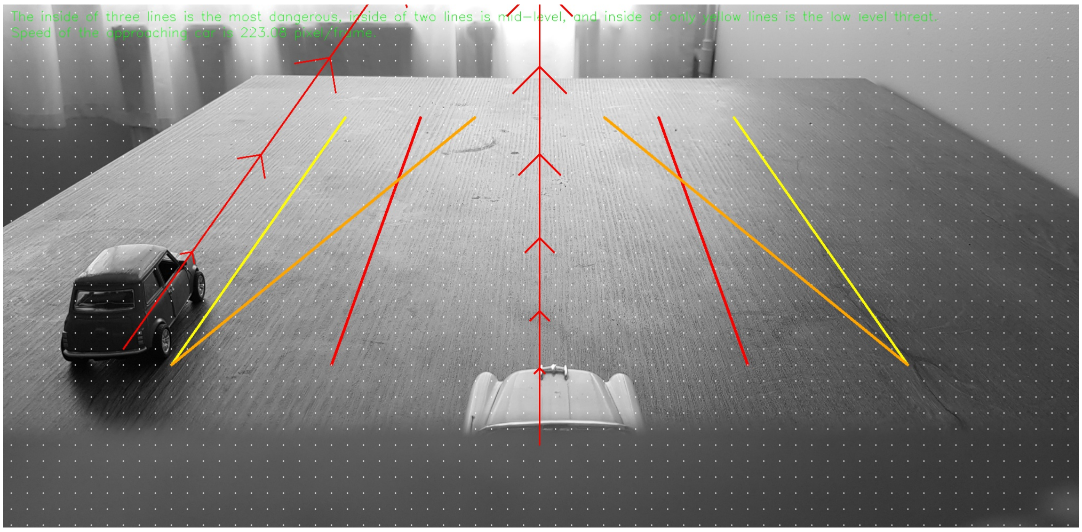

Figure 5.24: Non-collision Situation

In this example, two vehicles go in the same direction with different speeds, 223.8 pixels/frame and 10 cm/s speed respectively from left. As it has seen obviously, even after 6 frame of time passes, the roads of the vehicles will not intersect. Thus, they will protect their lanes, and there will not be a collision situation. For this reason, the system doesn't give any warning.

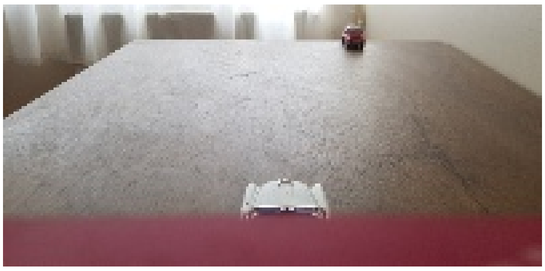

Figure 5.25: First Frame

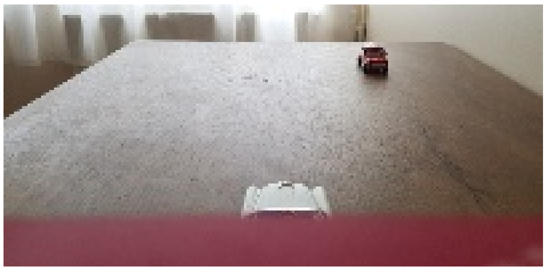

Figure 5.26: Second Frame

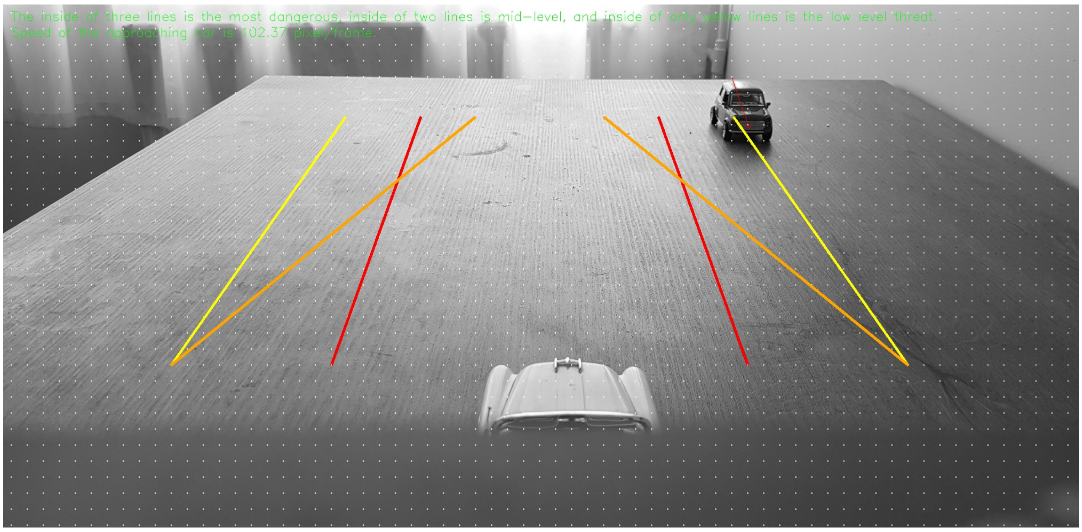

Figure 5.27: Optical Flow Vectors

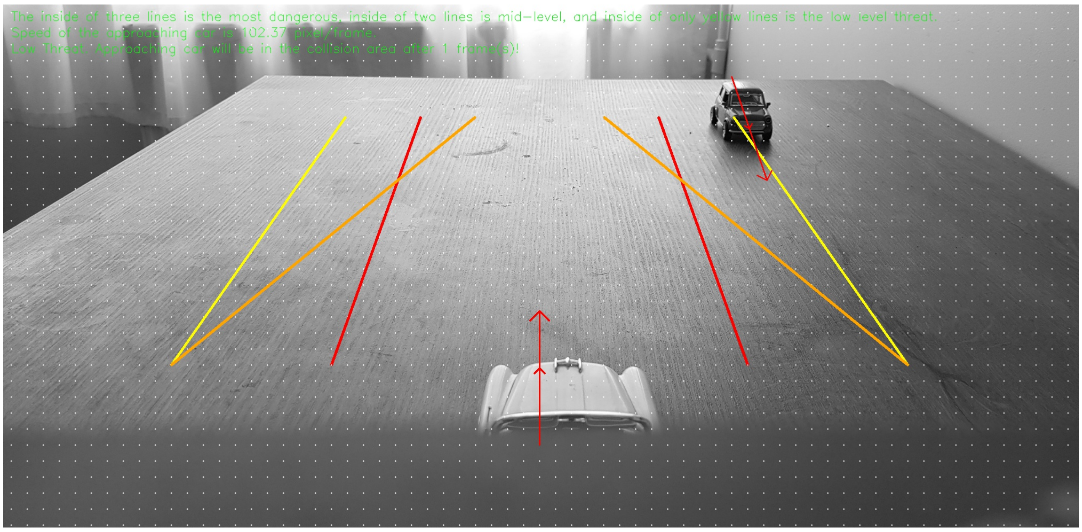

Figure 5.28: Low Level Threat

In this example, two vehicles have the same direction, however their moves are the opposite direction. It is thought that this is another example scenario of the non-collision scenarios, yet the results say it is a low threat collision situation because the approaching vehicle does not protect it's lane, it's coming to the left, towards to the other vehicle, and that's why it enters the yellow zone after one frame of time passes.

It would be non-collision situation if the approaching vehicle was going straightly in a different lane.

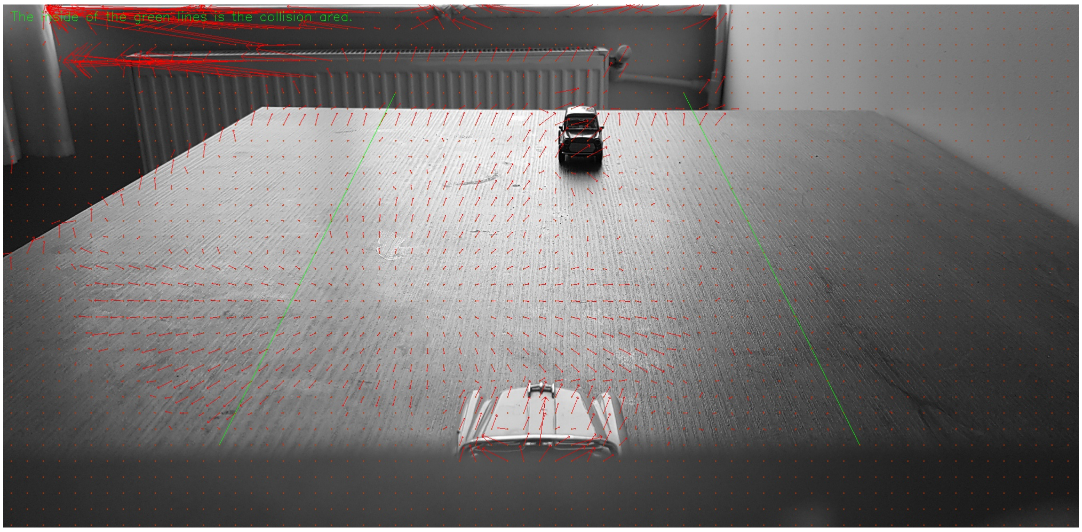

Figure 5.29: Disrupted Optical Flow Vectors

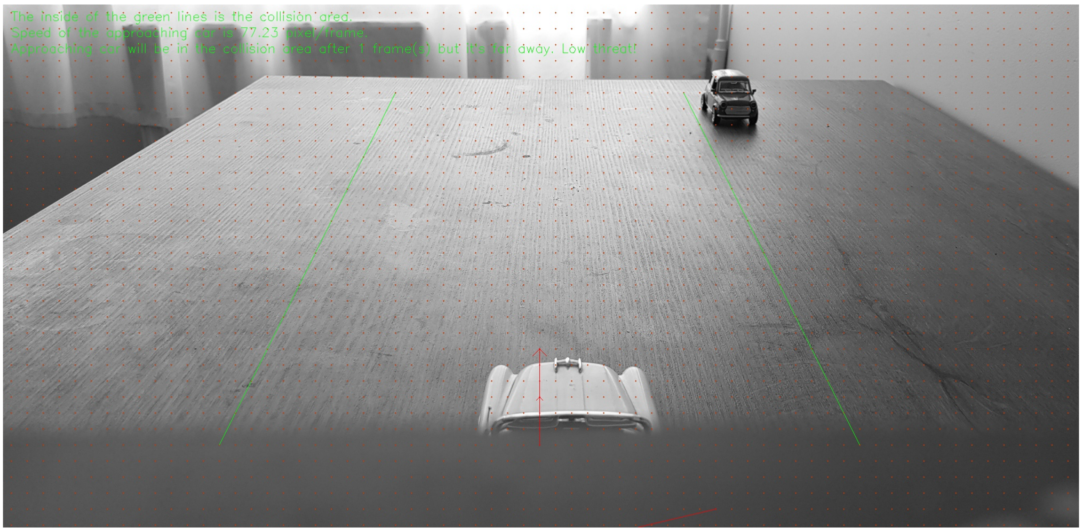

Figure 5.30: Not Recognized Optical Flow Vectors

These two figures above, 5.29 and 5.30, demonstrate the samples when the system fails. Due to optical flow vectors, it has seen that the moving object is not recognized, and therefore it is not possible to acquire correct outputs. Extraction of optical flow or velocity vectors for independently moving objects might be extremely challenging if conditions are unsatisfactory. Either the external noise or ill-posed vision system can lead to the failure of the algorithm or distorted results. Unwanted external conditions should be eliminated by diverse stable differentiation. Global motion models are more suitable for a reliable extraction of the optical flow in cases where non-stationary camera is used. On the other hand, global motion models come to mean that will be less efficiency, stability and simplicity [23]. Other issue is that the project is implemented using single camera, there is no way to sense the depth using single camera. Thus, the camera is positioned a slight different position before the shootings. The lighting of the environment while filming is crucial for the optical flow. The lighting should be stable during the movement of the car because Farneback's method compares the changes over all pixels in the consecutive images. In order to get the motion without blur and noise, camera settings were calibrated and stabilized for the specific environment. All these modification were done to reduce or remove the effects of uncontrolled system environment.

To sum up, Figure 5.4, 5.8 and Figure 5.12 are used to determine whether the algorithm performs the same calculations, and whether gives appropriate results when a vehicle enters collision zone from left or right corner. On account of this, the geometric calculation that used to find x points on the left and on the right boundary lines of the each zone are approved. As it has explained before, to find out whether the vehicle enters the collision zone between a few frames, y value of the optical flow vector is fixed, and this equation 4.3 is used for the yellow lines that separating the collision zone from the rest of the frame. If x value of the optical flow vector is between the found x values, it means that the vehicle will be in the collision zone. Moreover, Figure 5.16, Figure 5.8 and Figure 5.20 are used to diversify the collision scenarios. According to the part of the collision zone that the vehicle enters in, these scenarios are ensued low threat, mid-level threat or high threat.

In addition to these collision scenarios, non-collision scenarios have to be tested, and that's why, the scenarios like Figure 5.24 are created. Eventually, a few more samples are tested where the system finds improper optical flow vectors, and thus fails.

All the obtained outcomes are enlisted below:

Table 5.1: Results of the System for 20 Samples

Low Threat	**Middle Threat**	**High Threat**	**Non-collision**
6	5	3	6

Table 5.2: After All Level of Threats Merged

s height	**Collision**	**Non-collision**
Collision	13	1
Non-collision	1	5

It is clearly seen that the system successfully works for most of the scenarios, 17 out of 20 samples. Dataset might seem like too small, but it should not be forgotten that this system is not based on machine learning algorithms, it is a rule-based system. Testing each scenario only a few times are more than enough to prove that the rules of the system .

6 Conclusion and Future Works

Most of the companies that produce vehicles have many features for safety. Many of them are equipped with self-breaking system which starts to work when an unexpected occasion happens, so it prevents the collision. Assisted parking system is another example, driver can see the behind of the car with a camera, and sensor helps when the car approaches too much to an object. These kind of technological improvements for vehicles are surprising the people, yet the competition between the companies accelerates and there will be smarter cars in the near future.

Hence, this project is not only about to warn the driver when there is a possible collision, but also about to be aware of technological developments that mentioned above. When the outcomes of the project are checked, it has seen that our rule-based collision system covers most of the collision scenarios as planned before. Conversely, many issues were encountered during the implementation of the project, and referring them for future works is necessary.

In almost all of the aspects of the life, things do not occur as planned because many problems arise out of nowhere. At the beginning of the second semester, the plan was recording many videos, finding optical flow vectors of a moving object, then using the coordinates of these vectors as an input for a convolutional neural network. Three channels of an image, i.e. an input, would be like this: all zeros as Red, the coordinates of X vectors as Blue, and the coordinates of Y vectors as Green. So, it was not an meaningful picture for people but it was expected that it was meaningful for CNN network. Experiments had been done for approximately 3 months, but the success rate of the CNN were not sufficient to continue with this approach even though 3 different datasets,each of them had more than 30 possibly collision and 30 possibly not collision examples, were created.

It was decided to devise a rule-based system when the Deep learning algorithm had failed to succeed. This approach states that with the help of certain rules based on physic and geometry, and with a few assumptions, it is possible to create a system that warn people when there is possibility of a collision. For this system, many rules had to be defined, and these followings are thought to be the main rules

1. Optical flow vectors are always found between two frames.
2. The noise has to be deleted first from the optical flow vectors, and if there is less than 5 vectors left, it is not considered as a moving object.
3. Moving object has many optical flow vectors, however the biggest one represents the movement. The rest of the vectors are deleted.
4. After the optical flow vectors are filtered, if more than one moving object is found, the system performs the process for the fastest object.
5. The vehicle that the system is running goes straight with constant speed which is 10 cm/s. Each time, this speed is converted to pixels/frame unit, and then is drawn to the figures.

6. Collision zone that has 3 level of danger zone is created. The red zone embodies the most dangerous collisions, the orange zone embodies the mid-level dangers, and the yellow zone embodies the low level collisions. The ratio of the width of the red zone to the vehicle that this system has is the same the ratio of the width of a lane to the width of an average car.

7. It is found how many frames are necessary for the moving object to enter the collision zone.

8. If the moving object does not enter the collision zone, there is no warning. If it enters the red zone, the system gives high level warning; if it enters the orange zone, the system gives mid level warning; if it enters the yellow zone, the system gives low level warning.

These rules were tested only for a small dataset. Relying on these results are not enough to claim that the system will work well with a large-scale dataset and give correct results. Since there are many more collision scenarios in real-life, the dataset and these rules should be expanded. Furthermore, the optical flow function must be enhanced because the environment in which the shot is made may vary, and this function should not be affected by these changes. These improvements will advance the rule-based system.

References

[1] S. Singh, "Critical reasons for crashes investigated in the national motor vehicle crash causation survey," *Traffic Safety Facts Crash•Stats. Report No. DOT HS 812 115)*, 2015. [Online]. Available: https://crashstats.nhtsa.dot.gov/Api/Public/ViewPublication/812115

[2] N. C. for Statistics and Analysis, "Speeding: 2014 data," *Traffic Safety Facts. Report No. DOT HS 812 265*, 2016. [Online]. Available: https://crashstats.nhtsa.dot.gov/Api/Public/ViewPublication/812265

[3] H. L. Bloecher, J. Dickmann, and M. Andres, "Automotive active safety x00026; comfort functions using radar," in *2009 IEEE International Conference on Ultra-Wideband*, Sept 2009, pp. 490–494.

[4] E. Dagan, O. Mano, G. P. Stein, and A. Shashua, "Forward collision warning with a single camera," in *IEEE Intelligent Vehicles Symposium, 2004*, June 2004, pp. 37–42.

[5] A. Amditis, A. Polychronopoulos, I. Karaseitanidis, G. Katsoulis, and E. Bekiaris, "Multiple sensor collision avoidance system for automotive applications using an imm approach for obstacle tracking," in *Proceedings of the Fifth International Conference on Information Fusion. FUSION 2002. (IEEE Cat.No.02EX5997)*, vol. 2, July 2002, pp. 812–817 vol.2.

[6] N. Onkarappa and A. D. Sappa, "An empirical study on optical flow accuracy depending on vehicle speed," in *2012 IEEE Intelligent Vehicles Symposium*, June 2012, pp. 1138–1143.

[7] E. Martinez, M. Diaz, J. Melenchon, J. A. Montero, I. Iriondo, and J. C. Socoro, "Driving assistance system based on the detection of head-on collisions," in *2008 IEEE Intelligent Vehicles Symposium*, June 2008, pp. 913–918.

[8] J. Qi, N. Yu, and X. Lu, "A uav positioning strategy based on optical flow sensor and inertial navigation," in *2017 IEEE International Conference on Unmanned Systems (ICUS)*, Oct 2017, pp. 81–87.

[9] F. Valette, F. Ruffier, S. Viollet, and T. Seidl, "Biomimetic optic flow sensing applied to a lunar landing scenario," in *2010 IEEE International Conference on Robotics and Automation*, May 2010, pp. 2253–2260.

[10] V. Agrawal, M. Frueh, O. Bringmann, and W. Rosenstiel, "Evaluating optical flow vectors under varying computer-generated snow intensities and pixel density for autonomous vehicles," in *2018 48th Annual IEEE/IFIP International Conference on Dependable Systems and Networks Workshops (DSN-W)*, June 2018, pp. 247–254.

[11] J. L. Barron and N. A. Thacker, "Tutorial: Computing 2d and 3d optical flow," in *2008 IEEE Intelligent Vehicles Symposium*, 2005.

[12] [Online; Date Accessed: 26.07.2018]. [Online]. Available: http://vision.stanford.edu/teaching/cs231a_autumn1112/lecture/lecture13_optical_flow_cs231a.pdf

[13] S. Rakshit and C. H. Anderson, "Computation of optical flow using basis functions," *IEEE Transactions on Image Processing*, vol. 6, no. 9, pp. 1246–1254, Sep 1997.

[14] [Online; Date Accessed: 26.07.2018]. [Online]. Available: https://dspace.cc.tut.fi/dpub/bitstream/handle/123456789/25188/husseini.pdf?sequence=1&isAllowed=y

[15] G. Farneback, "Two-frame motion estimation based on polynomial expansion," in *Proceedings of the 13th Scandinavian Conference on Image Analysis*, ser. SCIA'03. Berlin, Heidelberg: Springer-Verlag, 2003, pp. 363–370. [Online]. Available: http://dl.acm.org/citation.cfm?id=1763974.1764031

[16] S. Husseini, "A survey of optical flow techniques for object," p. 6, aug 2017.

[17] [Online; Date Accessed: 26.07.2018]. [Online]. Available: https://docs.opencv.org/3.0-beta/modules/video/doc/motion_analysis_and_object_tracking.html

[18] Z. Jiang and W. Zeng, "Automatic vehicle parking system based on binocular vision and path planning," 2008, pp. 69–72.

[19] L. Liang, Z. Lei, and X. Jin, "The simulation of an auto-parking system," in *2011 6th IEEE Conference on Industrial Electronics and Applications*, June 2011, pp. 249–253.

[20] S. Liu, J. Wang, and T. Fu, "Effects of lane width, lane position and edge shoulder width on driving behavior in underground urban expressways: A driving simulator study," vol. 13, no. 10, 2016.

[21] P. Melo, A. Lobo, A. Couto, R. Frias, and C. Rodrigues, "Road cross-section width and free-flow speed on two-lane rural highways," vol. 2301, no. 1, 2012, pp. 28–35.

[22] [Online; Date Accessed: 26.07.2018]. [Online]. Available: https://www.automobiledimension.com/mini-car-dimensions.html

[23] K. Pauwels and M. M. V. Hulle, "Optic flow from unstable sequences containing unconstrained scenes through local velocity constancy maximization," *Procedings of the British Machine Vision Conference 2006*, 2006.