Bachelor of Science in Computer Science & Engineering

Helmet Detection of Motorcyclists in Real Time Using Cascade Classifier

by

Angkon Kumar Roy

ID: 1504100

Department of Computer Science & Engineering

Chittagong University of Engineering & Technology (CUET)

Chattogram-4349, Bangladesh.

May, 2021

Helmet Detection of Motorcyclists in Real Time Using Cascade Classifier

Submitted in partial fulfilment of the requirements for
Degree of Bachelor of Science
in Computer Science & Engineering

by

Angkon Kumar Roy
ID: 1504100

Supervised by

Muhammad Kamal Hossen
Associate Professor
Department of Computer Science & Engineering

Chittagong University of Engineering & Technology (CUET)
Chattogram-4349, Bangladesh.

The thesis titled '**Helmet Detection of Motorcyclists in Real Time Using Cascade Classifier'** submitted by ID: 1504100, Session 2019-2020 has been accepted as satisfactory in fulfilment of the requirement for the degree of Bachelor of Science in Computer Science & Engineering to be awarded by the Chittagong University of Engineering & Technology (CUET).

Board of Examiners

______________________________________ Chairman

Muhammad Kamal Hossen
Associate Professor
Department of Computer Science & Engineering
Chittagong University of Engineering & Technology (CUET)

______________________________________ Member (Ex-Officio)

Dr. Md. Mokammel Haque
Professor & Head
Department of Computer Science & Engineering
Chittagong University of Engineering & Technology (CUET)

______________________________________ Member (External)

Dr. Abu Hasant Mohammad Ashfak Habib
Professor
Department of Computer Science & Engineering
Chittagong University of Engineering & Technology (CUET)

Declaration of Originality

This is to certify that I am the sole author of this thesis and that neither any part of this thesis nor the whole of the thesis has been submitted for a degree to any other institution.

I certify that, to the best of my knowledge, my thesis does not infringe upon anyone's copyright nor violate any proprietary rights and that any ideas, techniques, quotations, or any other material from the work of other people included in my thesis, published or otherwise, are fully acknowledged in accordance with the standard referencing practices. I am also aware that if any infringement of anyone's copyright is found, whether intentional or otherwise, I may be subject to legal and disciplinary action determined by Dept. of CSE, CUET.

I hereby assign every rights in the copyright of this thesis work to Dept. of CSE, CUET, who shall be the owner of the copyright of this work and any reproduction or use in any form or by any means whatsoever is prohibited without the consent of Dept. of CSE, CUET.

Signature of the candidate

Date:

Acknowledgements

The success and final outcome of this thesis needed a lot of help and encouragement from a lot of people, and I consider myself extremely lucky to have earned it all the way through. It's been a satisfying experience both professionally and socially. Most of what I've done has been made possible by such supervision and support.

Above all, I am grateful to God for allowing me to finish this thesis. Following that, I'd like to thank my honorable thesis advisor Muhammad Kamal Hossen, Associate Professor, Department of Computer Science and Engineering, Chittagong University of Engineering and Technology (CUET) from the bottom of my heart for his encouragement, inspiration, and unwavering support during my thesis study. I value his probing questions, which forced me to think about things from different perspectives, as well as his unwavering support during the process. I appreciate the constant inspiration, guidance, and advice provided by the department.

Finally, I'd like to express my gratitude to my father and mother for their unwavering love, encouragement, inspiration, and support throughout my life and academic career.

Abstract

For regular commutes in Bangladesh, the motorcycle is a common mode of transportation. The number of motorcycle accidents has significantly increased over the years. In 2020, 4,891 road deaths occurred in Bangladesh. On the road, motorcycle deaths account for 25% of all fatalities. A motorcycle rider must wear a helmet and stop riding bikes with three or more wheels, according to the Motor Vehicle Act. Since the helmet is the most essential piece of motorcycle safety equipment, many riders refuse to wear it. If the passenger should not wear a helmet, a motorcycle accident may be fatal. The traffic control team tried to manage the situation manually, but in the real world, it was unsuccessful. The ideal solution will be to develop a smart helmet recognition system that can detect this sort of issue without requiring human interference. Our proposed model recognizes moving objects in real-time from surveillance video using background subtraction. The helmet images were trained using a cascade classifier. The motorcycle was detected using the Haar Cascade Classifier. Both helmet and non-helmet riders are classified using a machine learning methodology known as the LBP Cascade classifier approach.

Keywords: moving object, motorcycle detection, helmet detection, Haar Cascade, LBP Cascade.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Introduction

Motorcycles are widely used as a means of transportation in a number of countries. Their key advantages over other vehicles are their low premiums and low maintenance costs. Over the last decade, the number of motorcycle accidents has increased. According to Bangladesh Road Transport Authority (BRTA) [1], the number of motorcycles saw an approximately threefold increase from 7 lakhs to 22 lakhs in two years (2016 to 2018). As many as 6,686 people lost their lives and 8,600 were injured in a total of 4,891 road accidents in 2020 in Bangladesh [2]. The figures were revealed in the Bangladesh Passengers Welfare Association's (BPWA) 2020 annual road accident monitoring survey. On average, 18 people were killed in road accidents around the world every day. The bulk of those injured are motorcyclists who were not wearing a helmet at the time of the accident.

The most critical piece of motorcycle safety equipment is the helmet. Wearing a helmet protects the motorcyclist from crashes. Despite the fact that many countries expect riders to wear a helmet, many do not or do so improperly. According to a field survey conducted in Dhaka, 57 percent of all riders wore helmets, with motorcycle drivers wearing helmets at a rate of 90 percent. The number of pillion riders who wore helmets ranged from 3% to 5%. Since serious head injuries are common in fatally injured motorcyclists, wearing a helmet is important. Helmets are around 37% effective in reducing motorcycle deaths and 67% effective in preventing head injuries [3]. Helmets provide another layer of protection to the wearer's head, guarding against the more severe forms of traumatic brain injury. A helmet tends to reduce the risk of serious head and brain injuries by minimizing the impact of a force or collision on the head.

A mechanism for deciding whether or not a motorcyclist is wearing a helmet is proposed in the proposed scheme. To do so, we'd separate the vehicles into two groups: motorcycles and other vehicles. Then, for each motorcycle, we'll look at the top half of the ground. The top of the motorcyclist's helmet, whether he or she is wearing one, should be neatly squared. The completion of the scheme would help in the tracking of motorcycle riders' helmet use.

1.2 Framework/Design Overview

Our main aim is to relieve the stress that comes with maintaining motorcyclist traffic control. In previous years, numerous crashes occurred, resulting in several deaths and significant injuries. Motorcyclists will be required to wear helmets, which will reduce the number of casualties.

The main goal of this proposed solution is to automate the whole system while still identifying bikes with helmeted passengers. The whole procedure is broken down into the following basic steps:

- Detect moving objects from real time video.
- Detect motorcycles by classification of the vehicles.
- Identify area of interest of the detected objects.
- Develop a system that will detect helmet on motorcyclists using Cascade Classifier.

Figure 1.1: Framework overview.

1.3 Difficulties

Vehicle tracking, detection, and classification, like any traffic surveillance system, is challenging due to the presence of light, blurriness, and pollution of a narrow path in a remote area but vulnerable to real-world problems. This demand makes the task incredibly challenging due to the growing accuracy requirements. The Intelligent Transportation System and safety driver assistance systems are hot topics of study in the area of transportation and traffic control. As the number of cars on the road grows and development on the Intelligent Transportation System (ITS) gains momentum, road traffic video monitoring is becoming increasingly necessary. Motorcycle activity surveillance provides a foundation for collecting critical data like helmet use, helmet classification, and traffic law enforcement, among other items. The major difficulties in detection of helmets are enlisted as follows:

- Fast moving objects.
- Sudden change in object's direction of motion and velocity.
- Change of lighting of the detection region
- Complex backgrounds.
- Vehicle classification.
- Region of interest detection.

1.4 Applications

The organisers and administrative department are now concentrating their efforts on ensuring peace in various heavily populated areas. According to recent studies, reluctance of motorcyclists to wear helmets is one of the driving factors in the violation of environmental security. A string of recent tragic accidents have highlighted the importance of having an accurate motorcycle helmet warning system in order to take effective safety action in the event of a crisis. Motorcycle injuries claimed the lives of 1,026 people in the last ten months [4]. The most of them lacked a helmet. The number of crashes and casualties over time support the argument that traffic monitoring can be increased if a motorcycle and helmet detection system is established. Helmet detection of motorcyclists also help in various spheres as follows:

- Reduce traffic violation

- Ensuring safety of motorcyclists
- Automated system
- This system recognizes motorcycle among other vehicles
- Detects helmet on motorcyclists.

1.5 Motivation

One of the most significant research areas in today's world is intelligent traffic surveillance systems. Vehicle tracking is critical to making everyone's life safer and more relaxed. Autonomous vehicles will be on the streets in the immediate future. A self-surveillance framework will also be implemented. After all, each and every life is important. In the event of a collision, a motorcycle rider would be protected by wearing a helmet.

This region has seen a lot of progress in the past. We have established a robust helmet detection system in this paper that can detect moving objects using background subtraction, detect motorcycles within moving objects, and finally check whether a motorcycle rider is wearing a helmet or not. We have used cascade classifiers to achieve our goals. By using a proper algorithm with less computing complexity and processing time, the system improves the identification accuracy of motorcycles and helmets on them.

1.6 Contribution of the thesis

Thesis or research analysis is done to meet a certain set of objectives, such as the quality of findings or the time it takes to process data. The thesis's fields of contribution can be outlined as follows:

- This framework detects moving objects by comparing two consecutive video frames.
- Proper image pre-processing steps have been introduced for reduction of whole processing time.

- Detection of motorcycles using Haar Cascade.
- Detection of helmets using LBP Cascade.

1.7 Thesis Organization

The rest of this thesis report is organized as follows:

- Chapter 2 gives a quick rundown of the work that has been done so far in relation to "Helmet detection of motorcyclists in real time".
- The suggested system's solution is explored in Chapter 3. This chapter delves into the whole work process.
- The appraisal of performance and progress, as well as the system's review, are seen in Chapter 4.
- Chapter 5 contains an outline of this research analysis as well as a more detailed roadmap for the proposed structure.

1.8 Conclusion

The current framework is outlined in this part. This chapter provides an overview of helmet detection of motorcyclists in real time using cascade classifiers, as well as the challenges. This section also contains information about the work's motivation as well as the contributions made. In the following part, we'll look at the issue's past and current situation.

Chapter 2

Literature Review

2.1 Introduction

The aim of this research is to look at the challenges that might arise when conducting helmet detection research, as well as to provide a comprehensive overview of various detection methods. By providing a concise summary of previous articles, this chapter discusses a variety of attribute representation and classification methods used by various researches and the performances of these researches on different datasets.

Based on the series of acts, the object detection problem can be split into two main sections. For motorcycle and helmet identification, these are attribute representation and classification approaches. This chapter provides a concise overview of all of these processes.

2.2 Related Literature Review

We'll go through some of the more common inventions of recent years, as well as their drawbacks, in this section. Many techniques have been suggested for moving target identification, motorcycle detection, and helmet detection. Moving object detection and recognition are important computer vision issues that serve as the basis for future target movement testing. To distinguish between the vehicle and its surroundings, vehicle tracking employs technology such as optical image detection.

In [5] , moving objects are separated and classified as a motorcycle or other moving objects using the K-Nearest Neighbor (KNN) classifier based on features derived from their area properties. This unit, though, has a weakness in that it only detects bikes from side. Classification of vehicles in classes like car, motorcycles, trucks etc is done in [6]. The descriptors SURF, HAAR, HOG, and LBP are used to remove vehicle features in this article. Multilayer Perception, Support Vector Machines, and Radial-Bases Function Networks were used as classifiers for classification.

Another technique is presented for detecting helmet automatically in [7]. For feature extraction, a hybrid descriptor based on the Local Binary Pattern, Histogram of Oriented Gradients, and Hough Transform descriptors is suggested.

In [8], they suggested a Haar cascade-based vehicle classification scheme. In their paper, they show how to identify a vehicle from a video using Haar features and a cascade classifier. Until being labelled with the classifier, the images were converted to grayscale.

In [9], Stoimenovic Milos has implemented AdaBoost and SVM (Support Vector Machine) algorithms for real-time object detection in images, as well as details on how to train the classifier. Various attribute extraction techniques are explored in this chapter, including Sobel edge detection, Canny edge detection, and Hough transformation [10].

In [11], The study focuses on deep learning strategies for distinguishing between bikers who wear helmets and those who do not. By building bounding boxes, CNN and SSD also allowed real-time segmentation and classification of photographs. VGG19, VGG16, Mobile Nets, and Google Nets are CNN models that differentiate between images with and without a helmet (Inception v3). Python TensorFlow is used to measure the stability of a neural network. To develop a single net that creates structures quicker, the SSD paradigm would be combined with other net architectures such as Google Net and Mobile Nets. The other three CNN networks were outperformed by Mobile Nets.

In [12],The entirely enhanced real-time structure separates motorcyclists with and without helmets from caution using HOG as a classifier and LBP as a feature vector. The collected pictures are enhanced using an adaptive Gaussian mixture after each frame from footage is analyzed and vehicles moving are replaced by extracting redundant regions. The classification is done after the requisite pre-processing steps have been completed in order to minimize the capita. Two-stage sorting is used to get the best results and prevent misclassification. Finally, the ROI of the top 20% of the image is measured and the images are converted to grayscale to create the biker's head. The segmented photographs are then sent to CNN, which classifies them as either helmeted or unhelmeted bikers. In [13], Chaitat et al used YOLO classifier for the detection. But it uses two separate system along with GPU for which the implementatio becomes costly.

In [14], biker helmet wearers and non-helmet wearers are classified in real time using HOG as a feature vector and SVM as a classifier. Context simulation with background subtraction is done first, and involves attribute extraction and classification to separate moving objects like cars, humans, and bikes from non-moving objects like roads, trees, and buildings, eliminating effort in subsequent phases. In the following point, the biker will be classified. A set of functions and kernels are combined to verify classification outputs.

In [15], Varon et al used object recognition in vehicular flow, binary classification for motorcycle recognition, contour construction of the rider's head position, and classification using different deep learning methods in their research. With this process, approximately ten thousand images were used in the training period. As a result, the deep learning solution for this device needed a large number of photographs.

In [16], Padmini used SVM to develop a technology that detects bikes and helmets. After detecting history from the video and executing some morphological operations, they used SVM to detect motorcycles. They then used SVM and the HOG attribute descriptor to classify the helmet.

2.2.1 Gaussian Blur

In Machine Learning, the Gaussian Filter is one of the most widely used blur filters. It makes use of the "kernel convolution" strategy. This filter operates by measuring a value for each pixel (similar to the mean, but with more bias in the middle). The regular distribution, which is formed like a bell curve, is used to construct the filter. The theory is that pixels closest to the middle have a higher weight, whereas pixels farther apart have a lower weight.

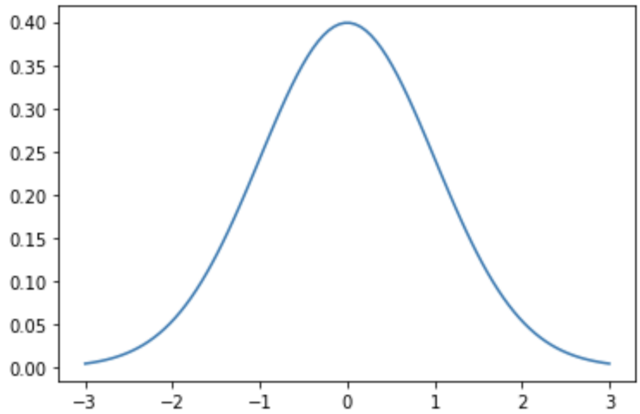

Figure 2.1: Grayscale conversion.

Figure 2.1 a normal distribution with a mean of 0 and a standard deviation (and variance, which is standard deviation squared) of 1 is an example.

A sample Gaussian filter would be as such:

1	2	1
2	4	2
1	2	1

Figure 2.2: (3,3) Gaussian filter.

2.2.2 Haar Feature

A Haar function is a collection of calculations performed on adjacent rectangular regions in a detection window at a specific location. The estimating process involves rounding up the pixel intensities in each region and then subtracting the totals. The following are some examples of haar characteristics in fig. 2.3.

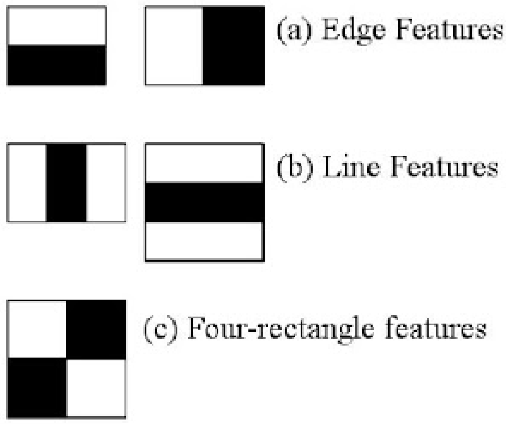

Figure 2.3: Types of haar features.

Integral images are used to speed up the estimation of these Haar features. It splits the screen into sub-rectangles and produces array references for each of them, rather than measuring at each pixel. This are then used to compute the Haar characteristics.

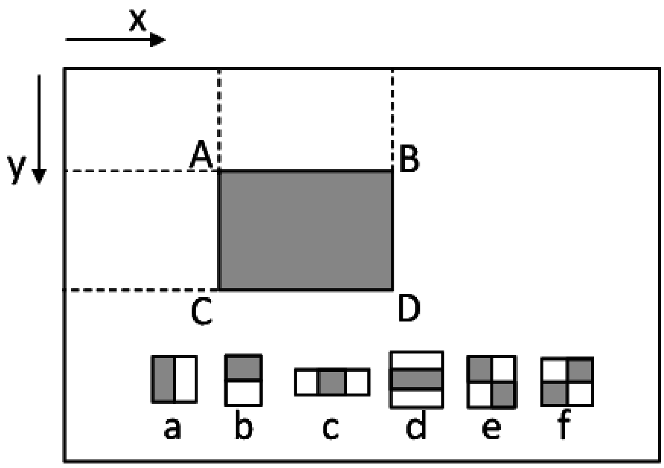

Figure 2.4: Illustration for how an integral image works.

When doing object detection, keep in mind that almost all Haar features are irrelevant since the only features that matter are those of the object. To represent an object, however, we must pick the best features from hundreds of thousands of Haar features.

2.2.3 Local Binary Pattern

For extracting texture features, the LBP is a valuable method. This method is widely used in facial recognition and pattern recognition approaches. The LBP operator transforms an image into a sequence or image of integer labels that define the image's appearance at a small scale.

The operator can also be extended to function in a variety of different sized communities. By using circular neighborhoods and bilinearly interpolating the pixel values, any radius and number of pixels in the neighborhood can be obtained. For neighbourhoods, we'll use the notation (P,R), which denotes P sampling points on a circle with a radius of R. Figure 2.5 shows an example of different circular neighborhoods.

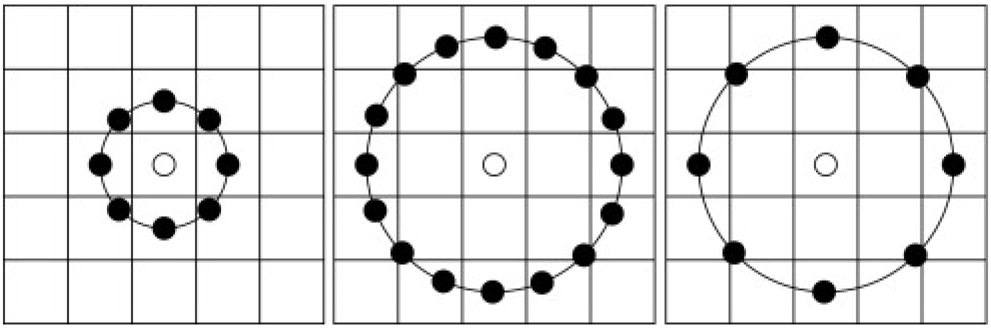

Figure 2.5: Three neighborhood examples used to define a texture.

The so-called uniform pattern is a useful extension to the original operator for shortening the function vector and introducing a simple rotation invariant descriptor. This hypothesis was motivated by the fact that certain binary patterns occur more often in texture images than others. The LBP is said to be uniform if the binary series has no more than two 0-1 or 1-0 transformations. The sequence 00010000 (two transitions) is universal, but 01010100 (six transitions) is not. Each uniform pattern has its own bin in the LBP histogram, whereas all nonuniform patterns are assigned to a single bin for computation. By using uniform patterns, the length of the feature vector for a single cell is reduced from 256 to 59.

2.2.4 Adaboost

The AdaBoost algorithm, which stands for Adaptive Boosting, is a Boosting technique used as an Ensemble Method of Machine Learning. The weights are reassigned to each instance, with higher weights allocated to instances that were incorrectly labelled. Adaptive Boosting is the term for this. Boosting is a technique used in supervised learning to minimize bias and heterogeneity. It is based on the principle of learners' sequential development. With the exception of the first, each subsequent learner is derived from previous learners. To put it another way, poor students are turned into strong students. There is a small difference in how the Adaboost algorithm works since it works under the same principle as boosting.

It generates n number of decision trees during the data training period. When the first decision tree/model is built, the record that was incorrectly labelled in the previous model takes precedence. As input, only these documents are sent to the second model. The cycle will be repeated until a number of foundation learners has been determined.

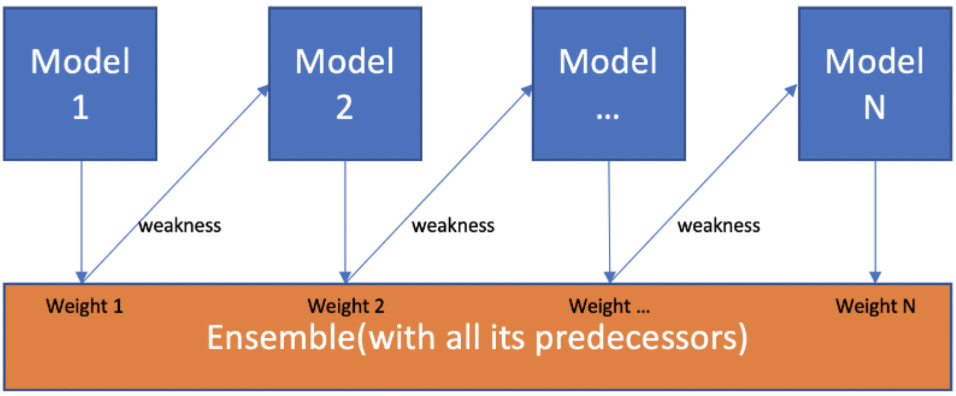

Figure 2.6: Adaboost technique.

Figure 2.6 After the first model is completed and the errors from the first model are registered by the algorithm, the record that was incorrectly classified is given as the input for the next model. This operation is carried out again and again before the criterion is met. The errors from the previous model were used to construct a n number of models, as seen in the diagram. In this way, boosting works. Models 1, 2, 3,..., N are individual configurations known as decision trees. Both boosting templates work on the same principle.

2.2.5 Cascade Classifier

Machine learning-based feature-based cascade classifiers entail training a cascade function with a large number of positive and negative images. Cascade classifiers are created by dividing strong classifiers into phases using the AdaBoost classifier. The term "cascade" refers to how the final classifier is made up of a sequence of simpler classifiers that are applied to the field of concern before the selected object is discarded or moved. The cascade classifier divides the classification function into two levels: planning and recognition. The training stage's function is to gather samples that can be classified as positive or negative. The cascade classifier uses many supporting functions to construct a training dataset and measure the prominence of classifiers. To train the cascade classifier, we'll need a collection of positive and negative samples.

The classifier was initially educated on a small number of positive and negative sets, which were random images of the same dimension of varying sizes. The classifier returns a “1” if the region supposedly recognizes the face; otherwise, it returns a “0". The key goal of the cascade classifier is to identify points of interest at different scales and improve the classifier's accuracy without adjusting the input image scale.

2.3 Conclusion

Several related works were discussed in the preceding subsection. They were all constrained in every way. There were just a few datasets in some of them. Some of them were not focused on detecting motorcycle helmets, but others were. Many of the aforementioned flaws have been attempted to be addressed in this new scheme. The technique is discussed in the following chapter.

Chapter 3

Methodology

3.1 Introduction

Finding a motorcycle rider's helmet is a difficult task. When an item moves, it creates a number of different lighting effects. As a consequence, the presence of several frames in a video is altered. The frame's shifting points should be separated from the rest of the scene. There may be some noise in the picture. Moving vehicles must also be classified into the proper vehicles even though they are separated. In our case, it's a motorcycle. The primary goal is to determine the motorcycle's classification. It detects a helmet in the classified motorcycle picture. The pixel intensities can be reduced to sense the helmet. A model based on Cascade classifiers and various image processing techniques has been created to solve these issues. The phases of the recommended solution are detailed in this chapter.

3.2 Diagram/Overview of Framework

Our primary focus is to design and implement a system that can easily detect helmet on motorcyclists on roads. This system is achieved using machine learning technique. That is called Cascade Classifier.

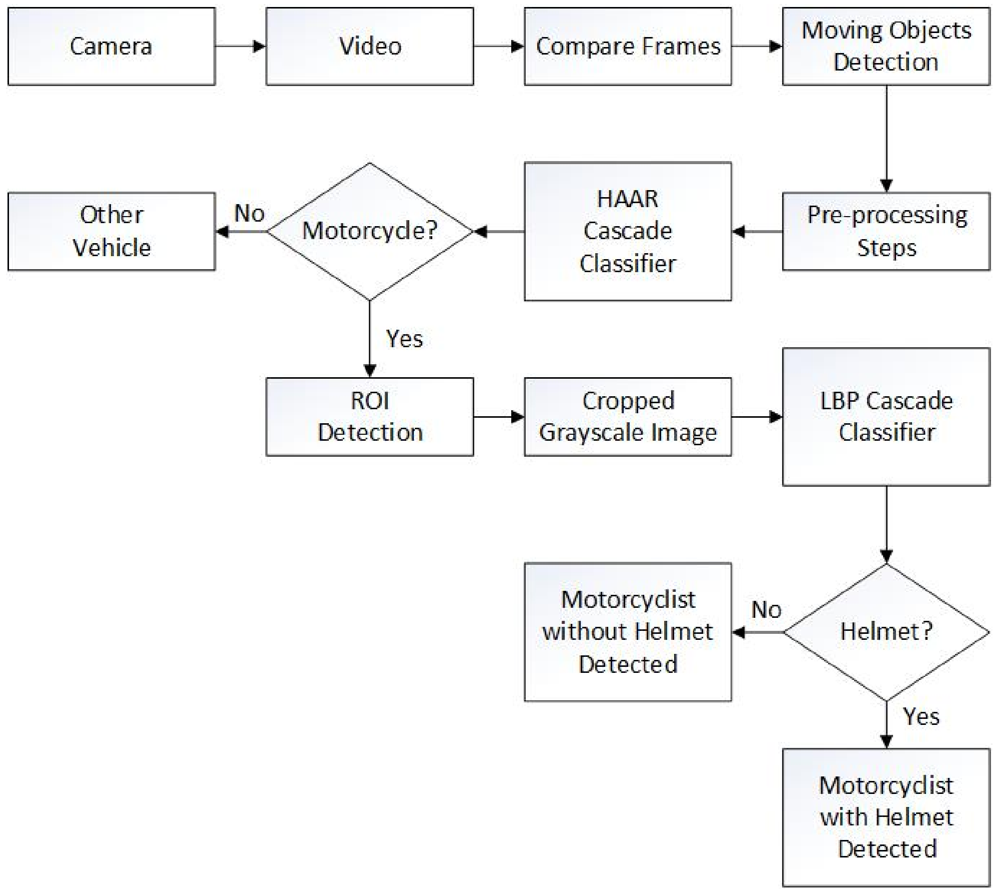

Figure 3.1: Main methodology.

Figure 3.1 shows the steps of the proposed detection method that will help to detect the requisite object practically. In the beginning of the whole steps, a camera is used to capture video which will be used for the process. From the video, exact frames are separated. Then sequential two frames are compared for background subtraction process. The frames give a comparison of pixels of the objects that are moving. The pixel difference of the two images indicates the moving objects in the frame. Then the moving objects are extracted from the frame and their coordinates are saved into the temporary storage for vehicle classification step. In image preprocessing steps, several methods are used such as grayscale conversion, blurring, thresholding, dilation and resizing. After these steps, the coordinates of moving objects are used to crop image. This image is used for classification of vehicles. HAAR Cascade classifier is used to detect motorcycle. After that, region of interest is selected from the classified image. Then, resizing is applied for the preparation of the helmet detection process.

The features are extracted using LBP. Then the image is checked using another trained cascade classifier whether there are helmet or not.

3.3 Detailed Explanation

3.3.1 Moving Object Detection

The first task is to detect the moving objects from the captured video. Moving objects can be cars, pedestrians, heavy vehicles, motorcycles etc. We can also find birds, insects and other things as moving objects. But they are ignored because of the reduction of processing data. So smaller moving objects will not be in our consideration.

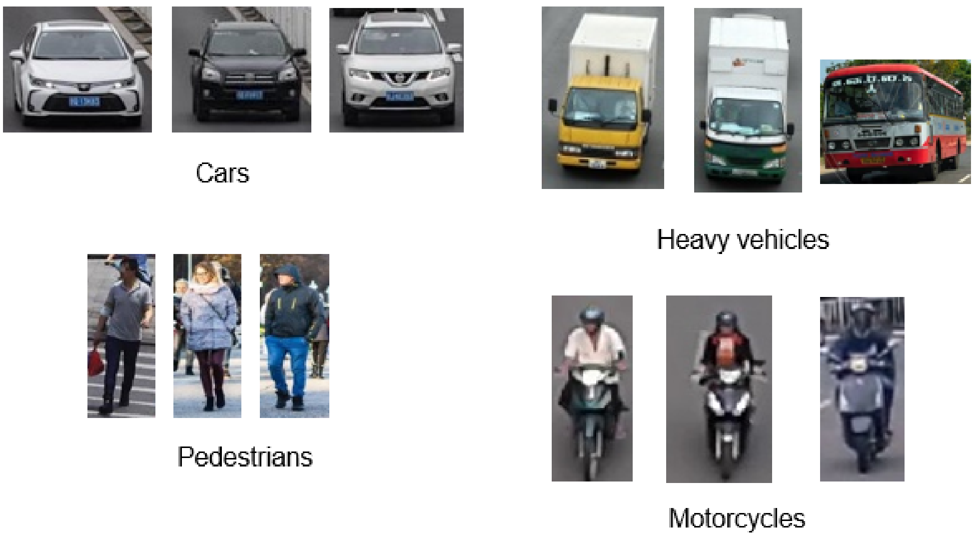

Figure 3.2: Moving objects.

Figure 3.2 shows some samples of moving objects in the frame. This process is implemented by comparing two frames and their pixel values from the captured video. The two frame will create two image arrays. Using these two image arrays will help us to extract the pixels of the objects that are moving. If the object in the image is moved, then its pixel value is changed. This measurement was accomplished by comparing the two image arrays. Then the coordinates of the moving objects are saved into storage for further process. Figure 3.3 shows pixel difference of two consecutive frame.

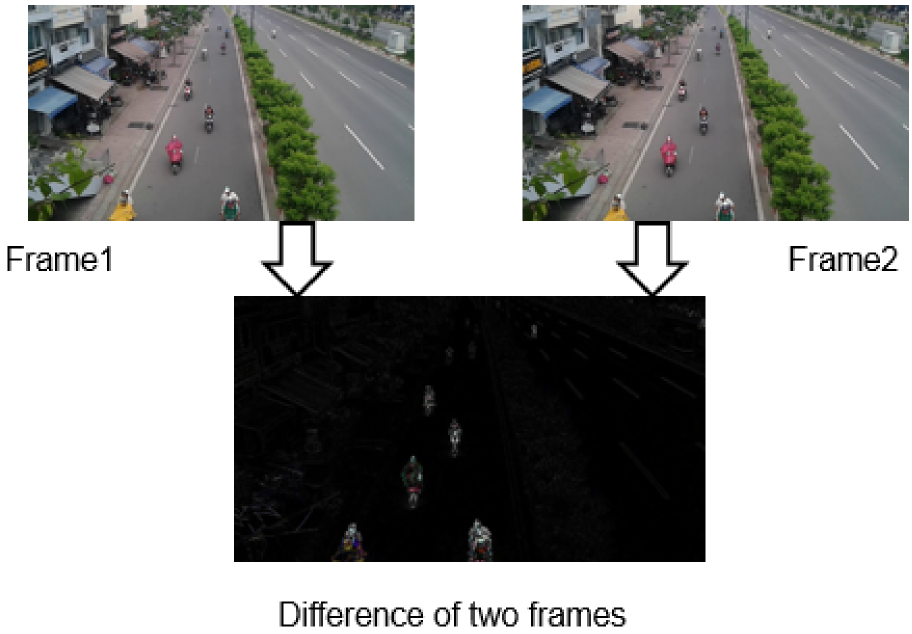

Figure 3.3: Difference of two consecutive frame.

3.3.2 Image Preprocessing

The triumph of the detection process greatly rely on a good preprocessing method. In this step, functions are carried out to prepare an image for further procedures. Unwanted noise is also removed in this step. Conversion of grayscale, Gaussian blur, thresholding and dilation is used in this step which are shown below:

3.3.2.1 Grayscale conversion

A monochromatic collection of colors ranging from black to white, grayscale is a monochromatic collection. As a consequence, a grayscale image has no color and just grayscale shades. Grayscale content can also be included in color files, which can be stored as grayscale (or black and white) images. This is because any pixel has a luminance value, regardless of color. Luminance, also known as brightness or intensity, is a color scale that ranges from black (zero intensity) to white (high intensity) (full intensity). Most image file formats support a minimum of 8-bit grayscale, which provides $2 \wedge 8$ or 256 levels of luminance per pixel. Some formats support 16-bit grayscale, which provides $2 \wedge 16$ or 65,536 levels of luminance. This technique strips each pixel of all color information, leaving only the luminance. Since digital images are represented by a mixture of red, green, and blue (RGB) colors, each pixel has three distinct luminance values. When extracting color from an image, these three values must be merged into a single value. This can be accomplished in a variety of ways. One option is to average the luminance values of each pixel. Another choice is to keep the luminance values of the red, green, or blue channels.

The image was converted to grayscale from RGB. The picture is reduced to a single eight-bit representation. Figure 3.4 shows the resulted grayscale image. The gray image will assist in defining and clarifying the edges. This would be more difficult if the image was in color. This will also speed up the processing of the image.

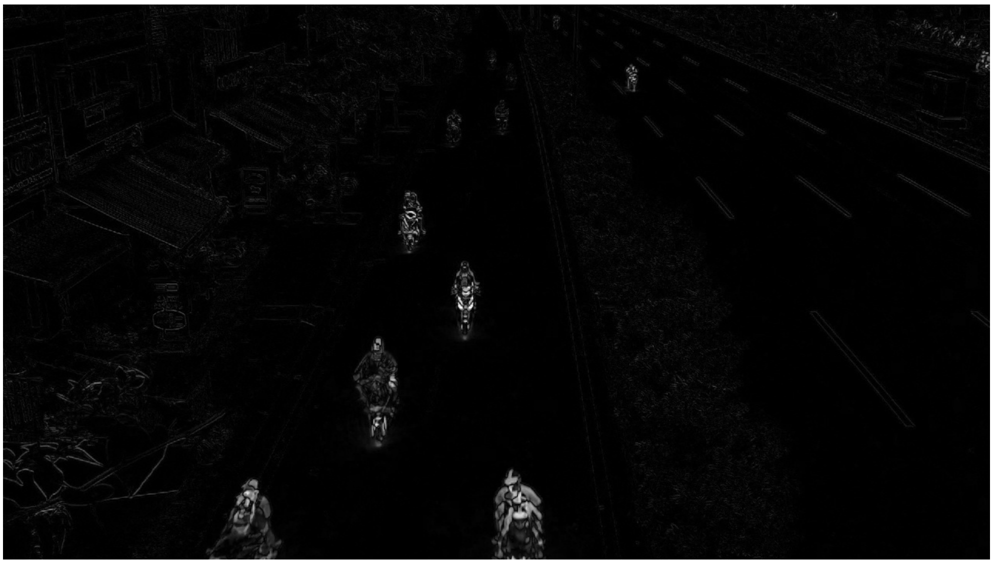

Figure 3.4: Grayscale conversion.

3.3.2.2 Gaussian Blur

Blurring is used to make the image seamless and translucent. The blurring effect is intended to balance out sudden changes in pixel intensities. The image has been blurred using the Gaussian blurring method. The picture is blurred using a function called Gaussian Blur. In this case, a Gaussian kernel is used instead of a box filter. The kernel's width and height should have been specified earlier (5,5)

$$\frac{1}{273}\begin{bmatrix} 1 & 4 & 7 & 4 & 1 \\ 4 & 16 & 26 & 16 & 4 \\ 7 & 26 & 41 & 26 & 7 \\ 4 & 16 & 26 & 16 & 4 \\ 1 & 4 & 7 & 4 & 1 \end{bmatrix}$$

Figure 3.5: Gaussian kernel(5,5).

The sigmaX and sigmaY standard deviations in the X and Y directions were also specified. Provided that they were both zeros, the kernel size was used to calculate them. Gaussian blurring is widely used because of its high performance in removing Gaussian noise from images and its ability to discern image edges. Figure 3.6 shows example of image blurring technique.

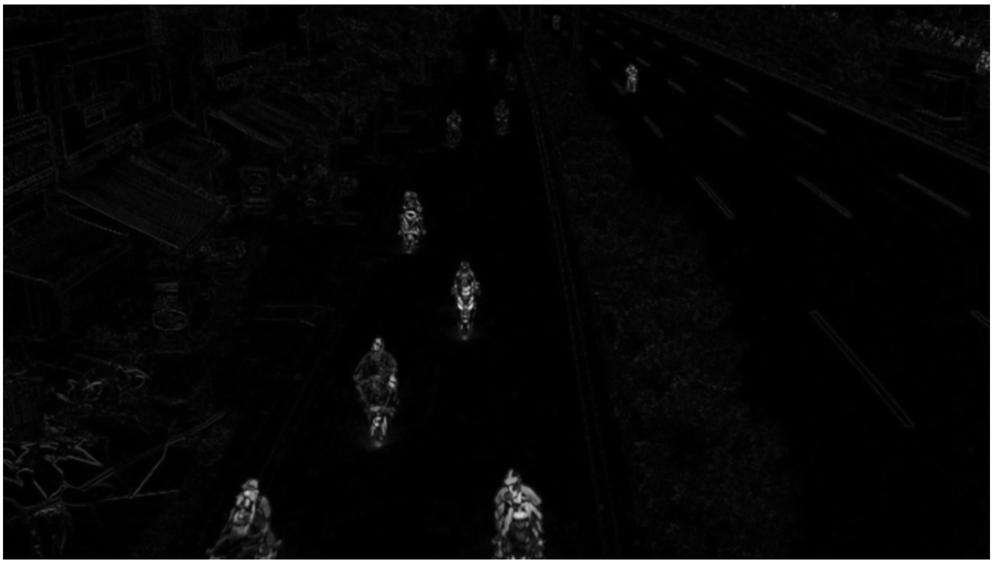

Figure 3.6: Image blurring.

3.3.2.3 Thresholding

Thresholding is an image segmentation technique that alters the pixels of an image to make them easier to understand. The method of transforming a color or grayscale image into a binary image, which is simply black and white, is known as thresholding. Thresholding is most widely used to choose points of significance in a photograph while ignoring the parts we don't want to use. In this image segmentation process, we change the pixel values of the blurred image to make it easier to read. The method of converting an image to a binary image is called thresholding. In the graphic, there will only be two values: 0 and 1. Converting a multi-tone image to a bi-tonal image makes subsequent processing even easier.

We used binary thresholding to determine the edges of the picture sharply. The minimum threshold value was taken 20 and maximum value was taken 255. We will get a binary image from grayscale image after thresholding is applied. We can see resultant thresholded image in fig. 3.7.

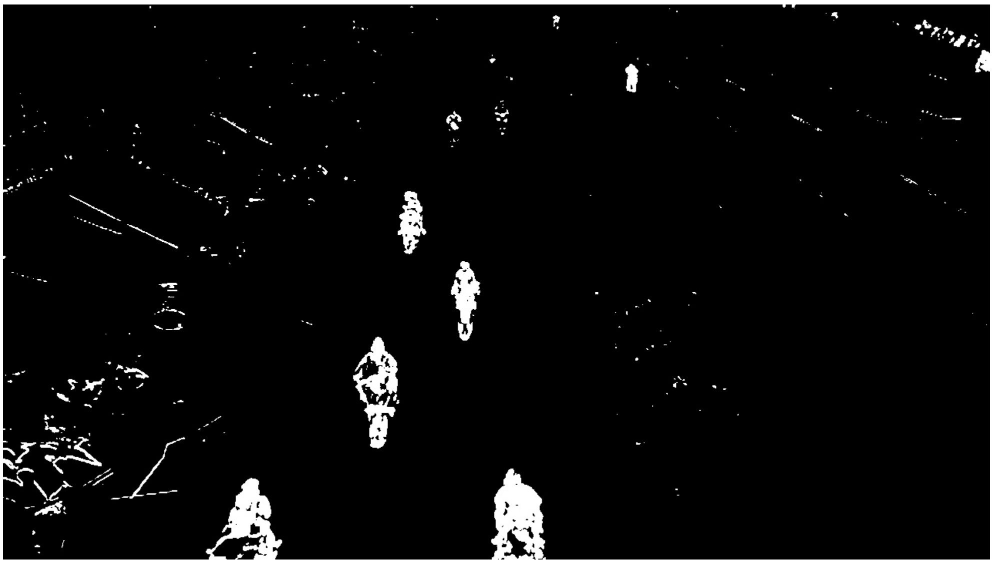

Figure 3.7: Thresholding of image.

3.3.2.4 Dilation

After completing thresholding process, there might be some holes lying in the middle of respected objects. For reducing the holes, dilation method is applied. During this process, the further noise which lies on the moving objects will be reduced. The dilation method on the binarized file is used to expand pixel values and apply pixels to object boundaries. This will remove any remaining noise from the picture. Noise is generated on the images as a result of the variations in intensities. Figure 3.8 shows the outpur result after the dilation process. Noises are responsible for distortion of images. The dilation process will remove remaining noise and is prepared for vehicle classification.

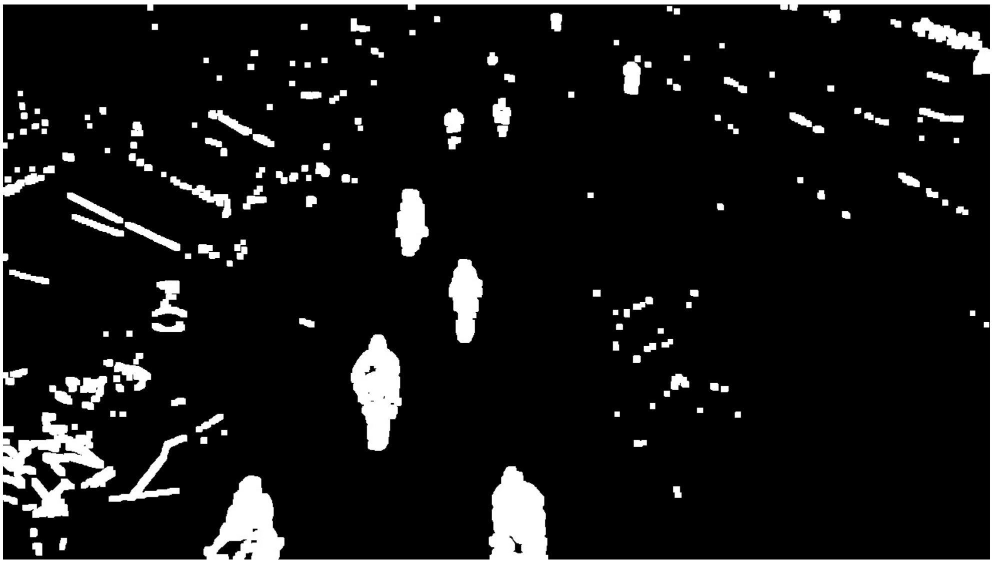

Figure 3.8: Dilation process.

3.3.3 Detection of Motorcycle

The following process is done by using Haar cascade classifier. The features of image is extracted using haar like features. Haar cascade classifier extracts features from an image using haar like feature extraction. For the detection process, first we need to train the classifier. Then with the trained model we can detect motorcycle in the image.

3.3.3.1 Training The Cascade

The cascade classifier is made up of a series of moves, each with a group of weak learners. Boosting is used to teach poor students, and it provides a highly accurate classifier based on the cumulative prediction of all weak students. Adaboost essentially chooses the appropriate characteristics and instructs the classifiers on how to use them. By merging "weak classifiers," it produces a "solid classifier" that the algorithm can use to detect objects. Haar features were used to remove the image's features in this case.

Weak learners are generated by moving a window over the input image and measuring Haar features for each subsection of the image. This difference is compared to a threshold for discriminating between non-objects and objects that has been taught. Since these are "weak classifiers," a large number of Haar features are needed to create a solid, accurate classifier. The last step combines these weak learners into a strong learner using cascading classifiers.

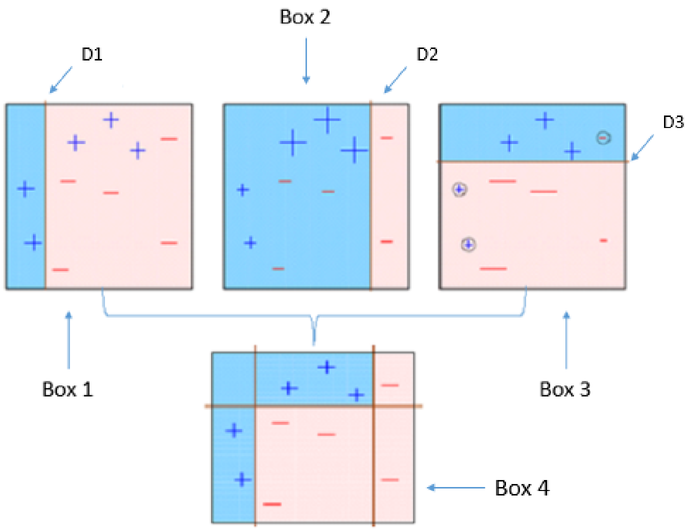

Figure 3.9: Representation of a boosting algorithm.

About 1817 images of motorcycles were used as positive image for the cascade training. Around 2517 negative images were used which does not contains any motorcycle. In our study, we used the opencv createsamples feature to produce positive samples for opencv traincascade. The output file from this feature is fed into opencv traincascade, which trains the detected motorcycle. Negative samples are drawn from random images that do not include the bikes that need to be identified. According to the model, the files were pre-processed and resized to 64x64 dimensions. Production time for the resized image would be reduced. After ten iterations of poor classifier planning, we had a successful classifier. The efficient classifier is saved in xml format for model verification.

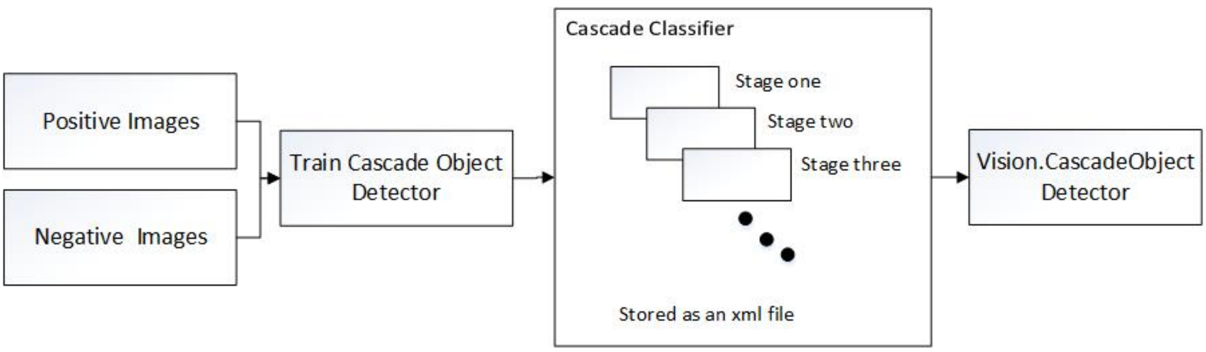

Figure 3.10: A flowchart of training the cascade.

3.3.3.2 Detection with Haar Cascading

Now after converting the image using the preprocessing steps, we tried to locate the exact features in the extracted moving objects coordinates. The coordinates were previously stored from moving object detection step. Using the motorcycle classifier which is an object loaded with XML file, we used an inbuilt function with it called the detectMultiScale. This role aided us in finding the characteristics of the new picture. It will use all of the features from the motorcycle classifier object to detect the features of the new picture. The parameters that will be transferred to this function are as follows:

- The gray scale variable(binary in our case).
- ScaleFactor - Parameter specifying how much the image size is reduced

The scale factor is the basic building block of our scale pyramid. In more depth, our model is assigned a defined scale during planning, which is visible in the XML. This means that if a motorcycle of this size is visible in the picture, it would be recognized. By rescaling the input file and making it detectable by the algorithm, we will reduce the size of a larger motorcycle to a smaller one. To increase the likelihood of detecting a comparable size with the model for identification, we use a small step for resizing, i.e. reduce the size by 5%. A fair possible value for this is 1.05. This also implies that the algorithm takes longer to run because it is more rigorous. We increased it to 1.59 for quicker detection since there is only one object to identify in the picture. As a result, we chose 1.59 as the scaleFactor in our case because it was the best fit for the image we were dealing with.

The "minNeighbors" parameter determines how many neighbors each candidate rectangle must have in order to be kept. This parameter will have an effect on the performance of the detected bikes. The higher the value, the less detections there are, but the higher the accuracy. 3 6 is a good value for it. In our case, since we have only one object in the image, we have taken 1 as the minNeighbors and this has worked perfectly for the image that we have used.

The function detectMultiScale returned four values from the previous step: the detected motorcycle feature's x-coordinate, y-coordinate, width(w), and height(h).

Based on these four principles, we'll draw a rectangle around the motorcycle in the main picture. As a result, the coordinates are saved in case they are needed again. If the motorcycle is not found in the image, the image will be ignored and we will not use this image for further helmet detection.

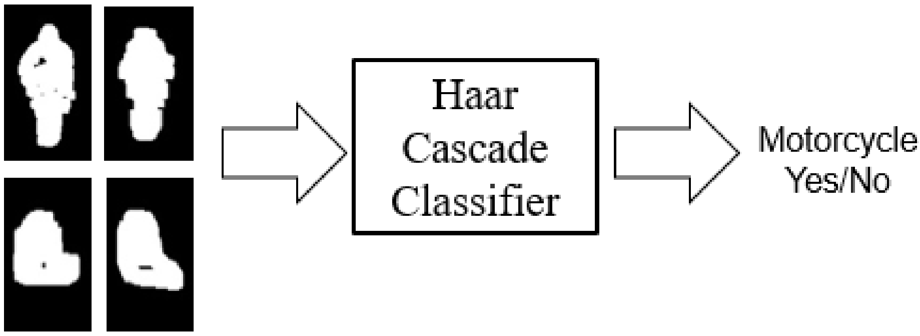

Figure 3.11: Detection of motorcycle.

After detecting motorcycles from the moving objects, a rectangle is created around the region. This detected image will be passed down for ROI detection.

3.3.4 Measuring ROI

After detecting motorcycle using the cascade classifier, region of interest is measured where helmet will be detected. We have seen that the helmet occurs at the top 25% of the classified motorcycle image. So we have used the top 25% height of the image. The image was cropped according to the process and passed it to the next step for classification of image.

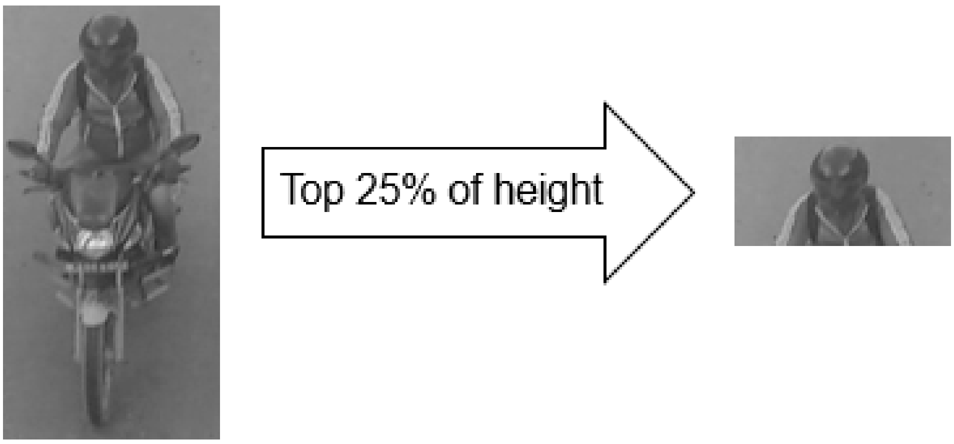

Figure 3.12: ROI measurement.

3.3.5 Cropping Grayscale image

After measuring the region of interest the region was cropped from the previous grayscale converted frame. This image will further used for helmet detection in the frame. The grayscale image will passed into the cascade model to detect whether there exist a helmet or not.

3.3.6 Helmet Detection

This step is done by using LBP cascade classifier. The features of image is extracted using LBP like features. LBP cascade classifier extracts features from an image using LBP like feature extraction. The helmet detection process is divided into two parts, Training the cascade and detection with the trained model.

3.3.6.1 Training The Cascade

According to the basic principle behind the LBP operator, two complementary steps can be used to reflect two-dimensional surface textures: local spatial patterns and gray scale comparison. The LBP operator has been extended to function in a variety of sized neighborhoods. Any radius and number of pixels in the neighborhood can be achieved by using a circular neighborhood and bilinearly interpolating values at non-integer pixel coordinates. Figure 3.13 shows the LBP features of a grayscale sample image. The complementary comparison measure may be the gray scale variation of the local neighborhood.

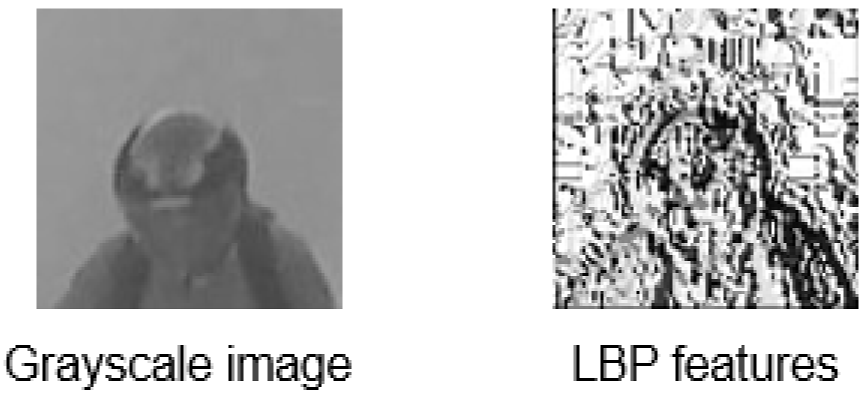

Figure 3.13: LBP features of a grayscale image.

The cascade classifier is made up of a series of moves, each with a group of weak learners. Boosting is used to teach poor students, and it provides a highly accurate classifier based on the cumulative prediction of all weak students.

For the cascade preparation, approximately 1109 helmet photographs were used as supportive images. A total of 2204 negative photographs were used, none of which had a helmet. The files were pre-processed and resized to 64x64 dimensions according to the model. It will take less processing time to process the resized image. After 10 iterations of weak classifier training, we got a strong classifier.The strong classifier is saved as XML file for the model testing.

3.3.6.2 Detection with LBP Cascade

The previous phase's grayscale image is now clipped and sent to the model for recognition of a helmet in the image. For the helmet classifier, which is an entity loaded with XML data, we used an inbuilt function called detectMultiScale. This role aided us in finding the characteristics of the new picture. It will use all of the features from the cascade classifier object to detect the features of the new image.

The scale factor is the basic building block of our scale pyramid. In more depth, our model is assigned a defined scale during planning, which is visible in the XML. This means that if a motorcycle of this size is visible in the picture, it would be recognized. By rescaling the input image and making it detectable by the algorithm, we will reduce the size of a larger image to a smaller one. To increase the likelihood of detecting a comparable size with the model for identification, we use a small step for resizing, i.e. reduce the size by 5%. A fair possible value for this is 1.05. This also implies that the algorithm takes longer to run because it is more rigorous. We increased it to 1.25 for quicker detection since there is only one object to identify in the image. As a result, we chose 1.25 as the scaleFactor in our case because it was the best fit for the image we were dealing with.

The "minNeighbors" parameter specifies the minimum number of neighbors each candidate rectangle should have in order to be retained. The efficiency of the detected bikes would be affected by this parameter. The higher the value, the less the detections but the higher the accuracy. 36 is a reasonable number. In our case, since we have only one object in the image, we have taken 1 as the minNeighbors and this has worked perfectly for the image that we have used.

The x-coordinate, y-coordinate, width(w), and height(h) of the detected feature of the helmet were returned by the function detectMultiScale from the previous point. Based on these four principles, we'll draw a rectangle around the observed helmet in the main picture. The process is now over.

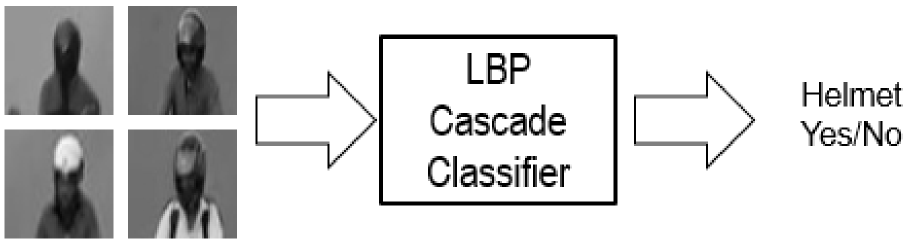

Figure 3.14: Detection of helmet.

After detecting helmet from the cropped image, a rectangle is created around the region in the main frame. The final processed frame is given as output which contains detected motorcycles and riders with helmets.

3.4 Conclusion

The technique for cascade classifier-based helmet detection is discussed in this chapter. Two attribute extraction methods and two classifier structures were used in this situation. The following parts go into the experimental effect review of the proposed technique.

Chapter 4

Results and Discussions

4.1 Introduction

In the previous chapter, the methodology for building a framework for detection of helmet of motorcyclist using cascade classifier technique has been discussed in details.

This framework is implemented using computer vision technique such as opencv. The full process was done using personal computer which provides 8GB of RAM and core i7 processor. We have tried to bring the best possible results by following various techniques.

4.2 Dataset Description

To boost prediction accuracy, a high-quality dataset is needed. We went to great lengths to find the best dataset for our system. There are two categories of dataset:

- Motorcycle dataset.
- Helmet dataset.

For training purpose for the motorcycle detection system 1,817 images have been used in the positive training folder. 2,517 images were used as negative images. Negative images does not contain any positive image. Thus no motorcycle is present in the negative images. The positive image belong to only one class that is motorcycle. Minimum false alarm rate was achieved during the cascade training. The positive and negative datasets were resized into 64x112 dimension. The images were pre-processed and converted into binary images for training according to the methodology. Figure 4.1 shows some sample motorcycle images from dataset.

Figure 4.1: Motorcycle dataset.

For training purpose for the helmet detection system 1,109 images have been used in the positive training folder and 2,204 images were used as negative images. The negative images are same as that were used in the motorcycle training. Negative images do not contain any helmet. Thus no helmet is present in the negative images. The positive image belong to only one class that is helmet. Figure 4.2 shows some sample helmet images from dataset. Minimum false alarm rate was achieved during the LBP cascade training. The positive and negative images were converted into 64x64 dimension. Figure 4.3 shows some sample negative images from dataset. The images were converted into grayscale images for the purpose of training.

Figure 4.2: Helmet dataset.

Figure 4.3: Negative sample dataset.

4.3 Impact Analysis

It is impossible to overestimate the role of effect analysis in the application of any new method. It's crucial to think about whether it will help or hurt society and human ethics. The impact analysis is divided into two parts, each of which is discussed below:

4.3.1 Social and Environmental Impact

The proposed structure would make keeping track of helmet offences simpler for the authority. By asking motorcyclists to wear helmets, the number of people killed or injured in motorcycle accidents will be reduced. It would guarantee the safety of motorcyclists. The traffic authority can also conduct an urgent check to see whether there has been a violation of the helmet code.

4.3.2 Ethical Impact

The system that has been proposed is fully ethical. It advises riders to wear helmets when riding their motorcycles. It is for their own protection. It would have little impact on them or their surroundings. Motorcyclists will be required to wear helmets, guaranteeing their safety.

4.4 Evaluation of Framework

The main purpose of this proposed methodology is to detect helmet among the motorcyclists by detecting motorcycles and checking whether there is helmet or not in the image. The whole process is done by using cascade classifiers. The main framework is divided into two major steps. But before that, we had to detect the moving objects by comparing two consecutive frames. Then among the moving objects we detected motorcycle using Haar Cascade with a validation accuracy of 95.07%. Then the system detects helmet among the detected motorcycle images using LBP Cascade with a validation accuracy of 93.61%. Lastly the system draws a rectangle around the detected helmet.

By comparing with other frameworks we can see that, definitely this proposed framework is a better choice. [16] uses HOG and SVM method for the whole process, where HOG is sensitive to image rotation. Also the SVM does not work well when there is noise in the image. [11] focuses on deep learning strategies to classify the bikers with and without a helmet. It is a complex method and hard to implement. From these functions of the above mentioned framework it is clear that the proposed methodology for helmet detection of motorcyclists using cascade classifiers is a better approach.

4.5 Evaluation of Performance

In order to test the system's effectiveness, four videos were used to evaluate its reliability. The videos' measurements, volume, and frame rate differ. The details of them are shown below in the table:

Table 4.1: Description of testing videos.

Parameter	**Video1**	**Video2**	**Video3**	**Video4**
Frame width	854	1280	1920	1920
Frame height	480	720	1080	1080
Video length	61s	90s	16s	12s
No of frames	1829	2703	480	360

The traffic sequence videos we have tested for checking result table 4.1, shown some information about those videos such as videos frame height and width, number of frames. Based on the outcome of the research video, we will determine the accuracy and effectiveness of our proposed scheme. Correct correspondence is the number of points which the proposed method detects and tracks correctly. And the term for correspondence is the total number of match points.

$$Accuracy = \frac{Number - of - corrected}{Number - of - correspondence} * 100\% \tag{4.1}$$

The testing system is divided into two sections:

- Performance in detection of motorcycles.
- Performance in detection of helmets.

4.5.1 Performance in Detecting Motorcycles

The performancce measurement of motorcycles are shown below:

Table 4.2: Performance in detecting motorcycle.

		Predicted Motorcycles	
Actual Motorcycle		Class=Yes	Class=No
	Class=Yes	167	8
	Class=No	7	131

Table 4.3: Quantitative measurement for motorcycle detection.

Parameter	**Value**
Accuracy	0.9521
Precision	0.9598
Recall	0.9543

As shown in table 4.2, we use traffic sequences of video to check the Haar Cascade-based motorcycle detection system, with the detection results as shown above. We can see that the system has a good detection result and it can also detect fast moving motorcycles. When a new vehicle enters the scene, a new detecting object is seen, a new number is distributed and a detecting window in that vehicle initialized. For moving destinations such as vehicles quickly, this has a strong detection effect for motorcycles. This can detect motorcycles that randomly join the monitoring scenario.

4.5.2 Performance in Detecting Helmet

The performance measurement of helmet are shown below:

Table 4.4: Performance in detecting helmet.

		Predicted Helmet	
Actual Helmet		Class=Yes	Class=No
	Class=Yes	142	9
	Class=No	6	76

As shown in table 4.4, we use traffic sequences of video to check the LBP Cascade-based helmet detection system, with the detection results as shown above. We can see that the system has a good detection result. When a new motorcycle is detected, then ROI is measured and in that region it is checked whether there is

Table 4.5: Quantitative measurement for helmet detection.

Parameter	**Value**
Accuracy	0.9356
Precision	0.9594
Recall	0.9403

helmet detected or not. For multiple detected motorcycles in one frame, this has a strong detection effect for helmet. After detection process, a bounding box is drawn around the detection area. The fig. 4.4, fig. 4.5 and fig. 4.6 shows output sample from video1,video2 and video3.

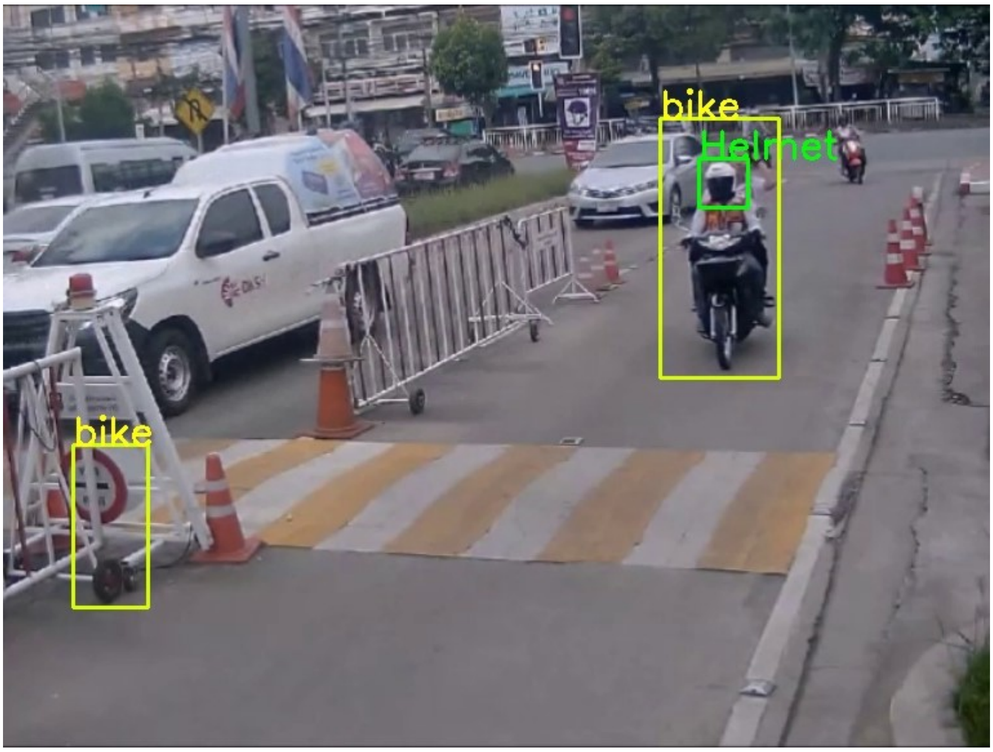

Figure 4.4: Final output sample of video1.

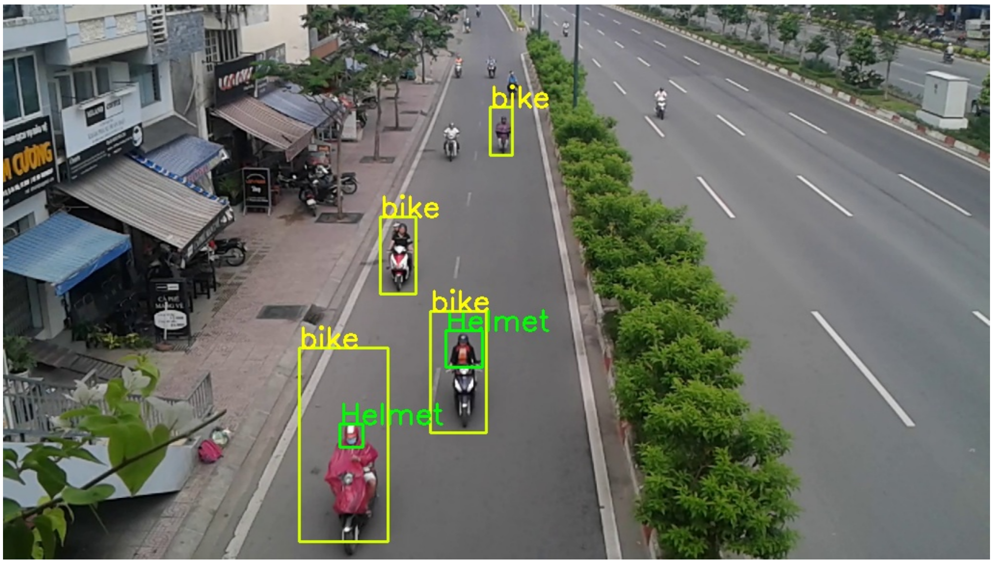

Figure 4.5: Output frame sample of video2.

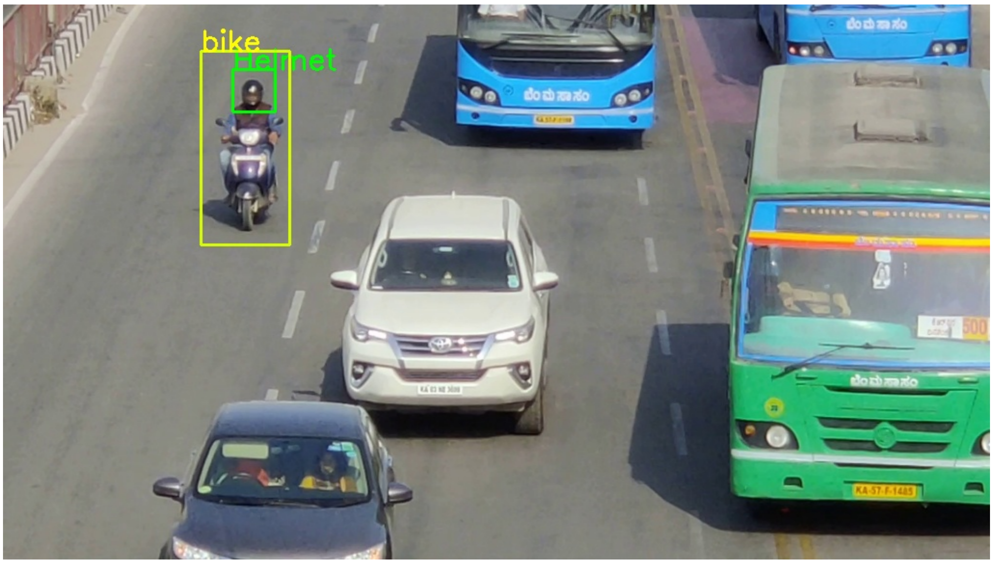

Figure 4.6: Output frame sample of video3.

4.5.3 Comparison with other Existing Method

Table 4.6: Description of testing for helmet.

Method	Accuracy
Padmini et al [16]	87.6%
Varon et al [15]	86.62%
Our method	93.56%

From the Table 4.6, we can see the comparison of other helmet detection system using existing methods with our proposed methods. From [16] where we can see that HOG feature extraction method used with SVM. It has accuracy higher accuracy rate but not high as ours. In [15], image processing techniques along with deep learning techniques have been used to achieve that goal. But their accuracy is not high as ours also. Applying image processing techniques with Haar Cascade and LBP Cascade gives the best result for achieving our goal.

4.6 Conclusion

This chapter examines the outcomes and effectiveness of the proposed strategy. Furthermore, several outputs have been shown to demonstrate its efficacy. The following chapter examines the framework's inference.

Chapter 5

Conclusion

5.1 Conclusion

The bulk of human operations have been outsourced thanks to technological advancements in recent years. With this aim in mind, CCTV cameras have been installed in virtually all city infrastructures. By detecting motorcyclists wearing helmets at different times of the day, these cameras can be used to provide security to traffic monitoring. If the helmet is to be monitored via these cameras, a diligent job of a person sitting in front of the camera all day is not needed.

The proposed method tackles the difficulties of identifying targets as their appearance varies due to changes in lighting, robust object movement, and the problem of the target object being blurred by other surrounding objects.

With the aim of solving the above problems, a proper cascade classifier approach is proposed as the detection process. The cascade classifier is a computationally simple algorithm that matches target and candidate models in each image to solve the problem of target motion uncertainty. The combinations of Haar cascade and LBP cascade proved to be efficient in solving the detection problem as mentioned before.The combination of Haar and LBP cascades, as previously stated, proved to be successful in solving the detection problem.

It's worth noting that our proposed algorithm employs background subtraction to detect moving objects. This is done by comparing two consecutive frames, which is both computationally simple and effective at separating moving objects. The aim of this research is to improve the accuracy of motorcyclist helmet detection in order to improve traffic safety systems on roads with cameras.

5.2 Future Work

Since, regardless of the effectiveness of various algorithms, through rapid advances in the field of computer vision, detecting targets remains a task with no limits. In this thesis, target objects are defined using rectangular regions. The use of background pixels as foreground pixels poses a problem if the rectangle is very large. A smaller rectangle, on the other side, results in a lack of knowledge about the target. In this respect, there is still room for study into developing a system for segmenting the foreground from the background based on the different shapes of the target area.

The proposed methodology can be extended to tracking of license plate. The license plate will show the license number of motorcyclists and it will be stored in the database. So the traffic surveillance authority can fine the motorcyclist. The methods for retrieving license plate numbers, on the other hand, are likely to result in increased difficulty as used in real-world situations.

Also the system may be tested and implemented during the night. At night their is darker region around the detection area. So it is a challenge detecting at night.

Finally, the research should be extended to include a warning system that alerts audiences when riders are not wearing helmets. Since there have been many dangerous accidents that have resulted in severe casualties in the recent past.

References

[1] S. C. S. Correspondent, *Growing number of motorbikes to blame, nischa survey finds*, Oct. 2018. [Online]. Available: https://www.thedailystar.net/country/road-accident-in-bangladesh-2018-survey-blames-rising-of-bikers-1641166 (cit. on p. 1).

[2] *Report: 18 people killed every day on average in road accidents in 2020*, Jan. 2021. [Online]. Available: https://www.dhakatribune.com/bangladesh/2021/01/09/report-18-people-killed-every-day-on-average-in-road-accidents-in-2020. (cit. on p. 1).

[3] *Fatality facts 2019: Motorcycles and atvs*. [Online]. Available: https://www.iihs.org/topics/fatality-statistics/detail/motorcycles-and-atvs (cit. on p. 1).

[4] *1,026 lives lost in motorcycle accidents in 10 months*, Nov. 2020. [Online]. Available: https://www.dhakatribune.com/bangladesh/2020/11/12/1-026-lives-lost-in-motorcycle-accidents-in-10-months (cit. on p. 3).

[5] R. Waranusast, N. Bundon, V. Timtong, C. Tangnoi and P. Pattanathaburt, 'Machine vision techniques for motorcycle safety helmet detection,' in *2013 28th International Conference on Image and Vision Computing New Zealand (IVCNZ 2013)*, 2013, pp. 35–40. DOI: 10.1109/IVCNZ.2013.6726989 (cit. on p. 6).

[6] R. Silva, K. Aires, R. Veras, T. Santos, K. Lima and A. Soares, 'Automatic motorcycle detection on public roads,' *CLEI Electronic Journal*, vol. 16, no. 3, pp. 4–4, 2013 (cit. on p. 7).

[7] R. Silva, K. Aires, T. Santos, K. Abdala, R. Veras and A. Soares, 'Automatic detection of motorcyclists without helmet,' in *2013 XXXIX Latin American Computing Conference (CLEI)*, 2013, pp. 1–7. DOI: 10.1109/CLEI.2013.6670613 (cit. on p. 7).

[8] M. K. Hossen, M. N. A. Siddiquee, A. C. Roy and M. S. A. Chowdhury, 'A video-based vehicle detection and classification system using cascade haar classifier,' *Southeast Asian Journal of Sciences*, vol. 6, no. 1, pp. 56–65, 2018 (cit. on p. 7).

[9] M. Stojmenovic, 'Algorithms for real-time object detection in images,' *Handbook of Applied Algorithms: Solving Scientific, Engineering and Practical Problems*, pp. 317–346, 2007 (cit. on p. 7).

[10] A. Aichert, 'Feature extraction techniques,' in *Camp medical seminar ws0708*, 2008, pp. 1–8 (cit. on p. 7).

[11] N. Boonsirisumpun, W. Puarungroj and P. Wairotchanaphuttha, 'Automatic detector for bikers with no helmet using deep learning,' in *2018 22nd International Computer Science and Engineering Conference (ICSEC)*, 2018, pp. 1–4. DOI: 10.1109/ICSEC.2018.8712778 (cit. on pp. 7, 32).

[12] L. Shine and C. V. Jiji, 'Automated detection of helmet on motorcyclists from traffic surveillance videos: A comparative analysis using hand-crafted features and cnn,' *Multimedia Tools and Applications*, pp. 1–21, 2020 (cit. on p. 7).

[13] A. Chairat, M. N. Dailey, S. Limsoonthrakul, M. Ekpanyapong and D. Raj K.C., 'Low cost, high performance automatic motorcycle helmet violation detection,' in *2020 IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2020, pp. 3549–3557. DOI: 10.1109/WACV45572.2020.9093538 (cit. on p. 8).

[14] K. Dahiya, D. Singh and C. K. Mohan, 'Automatic detection of bike-riders without helmet using surveillance videos in real-time,' in *2016 International Joint Conference on Neural Networks (IJCNN)*, 2016, pp. 3046–3051. DOI: 10.1109/IJCNN.2016.7727586 (cit. on p. 8).

[15] M. A. V. Forero, 'Detection of motorcycles and use of safety helmets with an algorithm using image processing techniques and artificial intelligence models,' in *MOVICI-MOYCOT 2018: Joint Conference for Urban Mobility in the Smart City*, 2018, pp. 1–9. DOI: 10.1049/ic.2018.0001 (cit. on pp. 8, 35, 36).

[16] V. L. Padmini, G. K. Kishore, P. Durgamalleswarao and P. T. Sree, 'Real time automatic detection of motorcyclists with and without a safety helmet,' in *2020 International Conference on Smart Electronics and Communication (ICOSEC)*, 2020, pp. 1251–1256. DOI: 10.1109/ICOSEC49089.2020.9215415 (cit. on pp. 8, 32, 35).