International Journal of Scientific Research in Science, Engineering and Technology
Print ISSN: 2395-1990 | Online ISSN : 2394-4099 (www.ijsrset.com)
doi : https://doi.org/10.32628/IJSRSET23102117

Design and Development of Automatic Detection System for Motercyclists without Helmet using Machine Learning

K. N. Hande[1], Pooja Takliker[2], Lekha Bhoyar[3] , Pranjali Bawankar[4] , Piyush Navghare [5]
[1]Department of Computer Science and Engineering, Priyadarshini Bhagwati College of Engineering Nagpur, Maharashtra, India
[2,3,4,5]B.E Student, Department of Computer Science and Engineering, Priyadarshini Bhagwati College of Engineering, Nagpur, Maharashtra, India

ARTICLEINFO

Article History:
Accepted: 05 April 2023
Published: 27 April 2023

Publication Issue
Volume 10, Issue 2
March-April-2023

Page Number
670-678

ABSTRACT

The use of two-wheelers as a mode of transport is increasing rapidly, but unfortunately, many riders neglect to wear helmets, which can lead to accidents and fatalities. To address this issue, many countries have implemented laws mandating the use of helmets for two-wheeler riders, and the police force often discourages this behavior by issuing traffic violation tickets. However, the process of issuing these tickets is often manual and tedious, which can lead to delays and errors. To solve this problem, a proposed system is to automate the process of detecting riders who are not wearing helmets. The system would use image processing algorithms to extract the license plate number of the rider, which would then be used to issue a traffic violation ticket. The image processing algorithm would consist of five parts, including image procurement, preliminary processing, fringe detection and segmentation, feature extraction, and recognition of character number plates using suitable machine learning algorithms. This automated system would not only make the process of issuing traffic violation tickets faster and more efficient, but it would also increase the compliance of two-wheeler riders with helmet laws. This would lead to a reduction in accidents and fatalities caused by not wearing helmets, ultimately making roads safer for everyone

Keywords: Automatic Detection System for Motercyclists without Helmet using Machine Learning , Moving Object Detection ,License Plate extraction ,Haar Cascade Classifier , Image thresholding ,Open CV2

I. INTRODUCTION

Two-wheelers are a popular mode of transportation in many countries, but unfortunately, many riders do not wear helmets, which increases the risk of accidents and injuries. To address this issue, governments have made it a punishable offense to ride a bike without a helmet, but the current methods of catching violators are often manual and require significant human assistance. To solve this problem, an automated system for detecting violators without helmets is highly desirable. The current video surveillance-based methods are passive and require significant human assistance, which can be inefficient over long durations. Automating this process would enable reliable and robust monitoring of these violations while reducing the need for human resources. Additionally, many countries are already adopting systems involving surveillance cameras at public places, which means that the solution for detecting violators can use the existing infrastructure, making it cost-effective. By automating the process of detecting violators without helmets, the system can significantly reduce the number of accidents and injuries caused by not wearing helmets. This would ultimately make the roads safer for everyone and improve the overall safety of two-wheeler riders. According to the World Health Organization (WHO), wearing a helmet while riding a two-wheeler can reduce the risk of head injuries by as much as 69% and reduce the risk of fatalities by up to 42%. Despite this, many riders continue to neglect wearing helmets, either due to lack of awareness or because they find them uncomfortable. To enforce helmet laws and increase compliance, many countries have implemented manual methods for detecting violators. However, these methods are often inefficient and can be prone to errors. Automated systems can be a more reliable and efficient solution. An automated system for detecting violators without helmets can be implemented using surveillance cameras that are already in place in public places. The system would use image processing algorithms to detect riders without helmets and issue traffic violation tickets automatically. This would significantly reduce the need for human resources, making the system cost-effective and efficient. The automated system can also be integrated with existing traffic management systems to provide real-time monitoring of helmet violations. This would enable authorities to identify areas with high violation rates and take appropriate measures to improve compliance, such as increasing awareness campaigns or increasing the number of surveillance cameras in those areas. Overall, an automated system for detecting violators without helmets is a highly desirable solution that can significantly reduce the number of accidents and fatalities caused by not wearing helmets. It would also improve the efficiency and reliability of enforcement, making the roads safer for everyone.

II. OVERVIEW

The use of helmets while riding two- and three-wheeled motor vehicles is crucial in preventing head injuries, which are the leading cause of death and major trauma for such users. Motorcycle travel carries a much higher risk of injury or death than driving a car, with the risk of a fatal crash being 28 times greater in 2016. About 75% of accidents involve motorcycles and passenger vehicles, while the remaining 25% are motorcycle accidents. Proper helmet use can significantly reduce the risk of fatal injuries, with a reduction of up to 42%, and the risk of head injuries, with a reduction of up to 69%. Despite this, many riders neglect to wear helmets, leading to a high number of deaths and injuries on the roads. In India, the highest rate of deaths due to motorcycle riders without helmets is reported in Tamil Nadu, accounting for 24% of accidents. Uttar Pradesh follows with 4,406 deaths (12.25%), followed by Maharashtra with 4,369 deaths and Madhya Pradesh with 3,183 deaths. Meghalaya and Mizoram were the only two states where no motorcycle users without helmets died. This highlights the urgent need for better enforcement of helmet laws and increased awareness campaigns to promote the use of helmets among motorcycle riders. Automated systems for detecting violators without helmets, as discussed in the previous section, can be a reliable and efficient solution to this problem, ultimately saving countless lives and reducing the number of injuries on the roads.

III. ABOUT PRESENT SYSTEM

The present system for detecting helmet use by motorcyclists substantially relies on traditional styles, which involve moving object discovery as the first step. The stir segmentation system is used to prize moving objects from surveillance vids, and common styles include optic inflow, frame difference, and background deduction. The coming step involves using hand- designed point descriptors similar as LBP, overeater, and SIFT to prize the features of motorcycles and other vehicles. Eventually, motorcycles are classified using double classifiers like SVM and KNN. The problem of detecting helmet use by motorcyclists is divided into two way The first step is to member and classify the vehicle images, aiming to descry all the moving objects in the scene. The alternate step is helmet discovery, which uses a mongrel descriptor to prize image features and a support vector machine classifier to classify an image into a helmet and anon-helmet. Dahiya etal. habituated background subtraction and object segmentation to detect bike riders from surveillance video, and then determined whether the bike rider was using a helmet or not using visual features and a binary classifier. Talaulikar etal. also used a background deduction fashion to identify moving vehicles and PCA to decide features. While these traditional styles have been used for detecting helmet use by motorcyclists, they bear significant mortal backing and are frequently unresistant. Automated systems, as bandied before, can significantly reduce the quantum of mortal coffers demanded and give dependable and robust monitoring of helmet use violations.

IV. LITERATURE SURVEY

1. Real-Time Helmet Detection System for Bike Riders Using Deep Learning. 2019 Deep Learning, CNN, Faster R-CNN Vivek Khetan, Himanshu Agarwal, Yuvraj Singh Parihar.
2. License Plate Recognition Using Convolutional Neural Network. 2018 Convolutional Neural Network (CNN) Sukhdeep Kaur, Mandeep Singh.
3. A novel method for motorcycle rider's helmet detection based on Faster R-CNN 2019 Faster R-CNN, Deep Learning, CNN Xiaoyan Huang, Yan Feng, Xiufen Ye.
4. License Plate Recognition using Deep Learning: A Survey 2021 Deep Learning, CNN, Faster R-CNN, YOLO, SSD Abdur Rehman, Akram Fida, Arslan Shaukat, Irfan Mehmood, Muhammad Iqbal, Muhammad Qasim Mehmood .
5. Real-time Vehicle Detection and License Plate Recognition Using Deep Learning. 2020 Deep Learning, CNN, YOLOv3, OpenCV Saad Bin Sajjad, Syed Wasi Haider Naqvi, Zulfiqar Habib, Muhammad Asif, Hasan Ahmed.
6. A Deep Learning Approach to Motorcycle Detection for Helmet Detection 2020 CNN, Faster R-CNN, YOLOv3, RetinaNet Jiaming Yang, Xiaoxiang Guo, Xiaozheng Huang, Xiaoyu Liu, Xingbin Liang, Xianghan Wang.

V. PROPOSED METHODOLOGY

5.1. Moving Object Detection :

Moving object Discovery The first task in helmet identification is to descry a moving vehicle. It's the first step before performing more sophisticated functions similar as tracking or categorization of vehicles. Rather than incontinently recycling the entire videotape, the illustration starts by carrying an original videotape frame in which the moving objects are segmented from the background. Processing only the original many frames helps to take the way needed to reuse the videotape. The focus sensor needs a certain number of videotape frames to initialize the Gaussian admixture model. The focus segmentation process isn't perfect and frequently includes undesirable noise. Next, we find bounding boxes of each connected element corresponding to a moving vehicle. Generally, further than one blob is detected piecemeal from moving vehicles similar as climbers, trees, tykes and other small noises. All the blobs that correspond of lower than n number of pixels are discarded(in our case n is 150 pixels). But there are a lot of gaps in the blob, that is, it isn't one coherent blob. We use morphological opening to remove the noise, and to fill gaps in the detected objects, which makes the blob more coherent. Once the blob is set up, the raw image is uprooted that's hidden behind the blob.

5.2. Vehicle Classification :

Vehicle classification is a task of identifying the type of vehicle from the image or video data. In this case, the goal is to classify the extracted moving vehicles into two categories: two-wheelers or four-wheelers. The training dataset consists of 1000 images for each class, and synthetic images are used to augment the training data to increase its diversity.The images contain the vehicle surrounded by various objects, such as trees, footpath, buildings, and other noise objects. These objects can add complexity to the classification problem, making it challenging for the classifier to distinguish between vehicles and other objects. However, the use of synthetic images and image augmentation can help to address this challenge.To test the classifier's effectiveness, a different set of images is used. Although this dataset may not be representative of real-world moving objects, it is sufficient to evaluate the performance of various machine learning algorithms.The images are converted to grayscale, and the raw pixel values are used as input features to the classifier. Various machine learning algorithms, ranging from classical algorithms to modern deep neural networks, are used to classify the vehicles into two categories. The classifier's performance is evaluated based on metrics such as accuracy, precision, recall, and F1 score.

5.3. Helmet Detection :

The next step in the process is to detect whether the rider of the two-wheeler is wearing a helmet. To accomplish this, the same approach used in vehicle classification is applied. However, instead of using the entire two-wheeler image, only the head region of the rider is cropped and used as input to the helmet detector.By using cropped images of the head region, the focus is solely on the helmet detection task, making it easier to train and test the effectiveness of various machine learning algorithms. The same number of images where the rider is wearing a helmet and where the rider is not wearing a helmet were used to maintain class balance.Several machine learning classifiers are used to select the best one for this task. The classifier's performance is evaluated based on metrics such as accuracy, precision, recall, and F1 score. The ultimate goal is to achieve high accuracy and precision in detecting whether the rider is wearing a helmet or not.

5.4. License Plate extraction :

The license plate extraction process can be broken down into several steps. The first step is to detect the presence of a license plate in the image using object detection algorithms such as Haar cascades, HOG, or deep learningbased methods. Once the license plate is detected, the next step is to localize the plate by finding its position and orientation in the image. This can be done using techniques such as template matching or edge detection. After localizing the plate, the license plate extraction step involves cropping the image to extract the license plate region. This can be done by defining a rectangular bounding box around the plate's location or using more sophisticated techniques such as polygonal region selection.Once the license plate region has been extracted, it can be pre-processed to enhance its quality and remove any noise or distortion. This typically involves techniques such as thresholding, image resizing, and contrast enhancement. Finally, the extracted license plate can be passed to an OCR engine to recognize the characters on the plate. The accuracy of the OCR depends on various factors such as the quality of the image, the font used on the plate, and the language of the characters. Therefore, it is important to ensure that the license plate extraction process is robust and accurate to obtain reliable OCR results.

VI. SYSTEM ARCHITECTURE

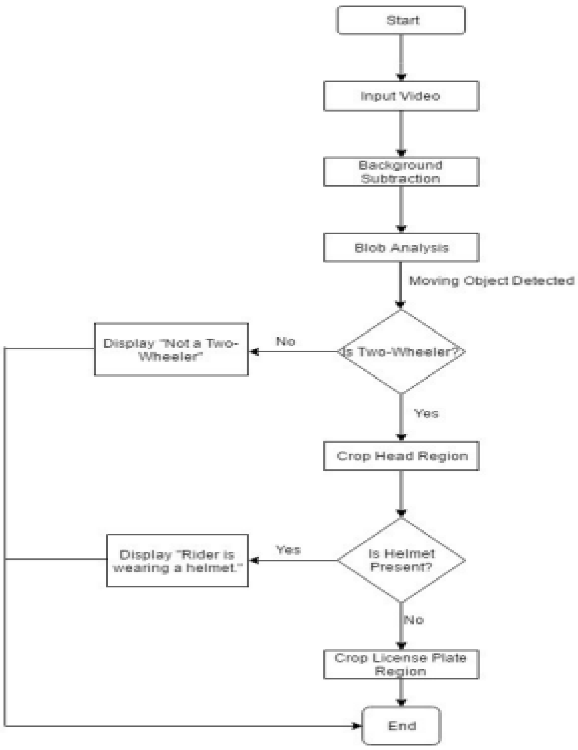

Fig.1. Flowchart of the proposed method

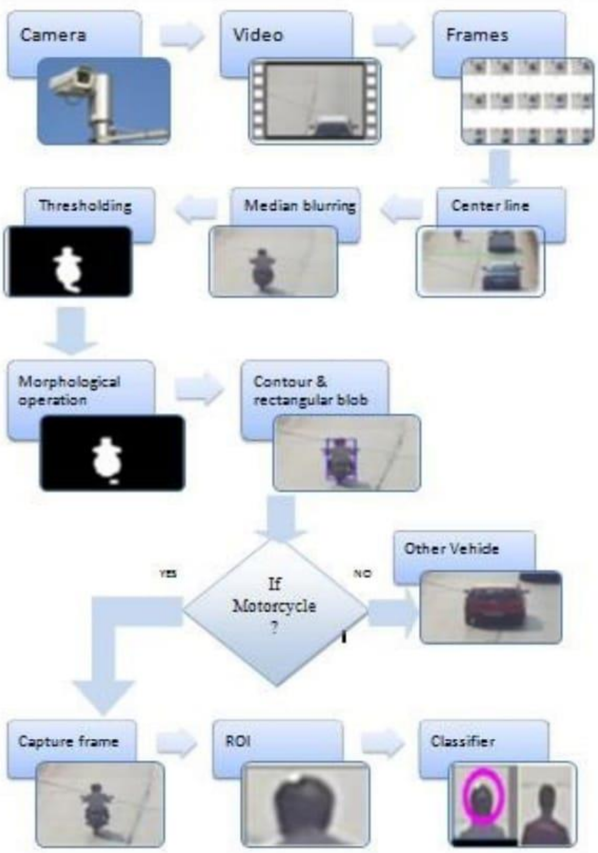

Fig. 2 Implementation of Purposed Method.

VII. ALGORITHM

The technique involves capturing a video stream through a webcam or recording a video of traffic on the road. The first step is image acquisition, where the camera captures the traffic on the road, and the frames are extracted from the video for further processing. It's essential to ensure that the camera is fixed at a constant distance and angle from the road, with a camera setting that covers only the road, to avoid capturing unnecessary information. The next step is preprocessing techniques, where noise removal and binarization take place. Noise removal is necessary to eliminate any unwanted information from the captured frames that could affect the accuracy of the helmet detection. Binarization is the process of converting the color image to a binary image, where only two values are present (black and white) to simplify the image and make it easier to process. After preprocessing, the technique involves vehicle classification, where motorcycles are classified based on their aspect ratio and area of the contour of a particular vehicle. The aim is to identify the motorcycles in the video stream and process them further for helmet detection. Finally, helmet detection involves extracting the head part (ROI) of the rider and giving its features to the classifier. The classifier has been trained on a set of pictures of helmets to match trained features with the ROI and determine whether the motorcyclist is wearing a helmet or not.

7.1. Haar Cascade Classifier :

The algorithm works by breaking down an image into smaller sub-regions, and for each sub-region, it applies a set of classifiers that evaluate whether the sub-region contains the object of interest or not. Each classifier is essentially a function that looks for specific patterns or features in the image that are indicative of the presence of the object. During the training phase, the algorithm learns which patterns or features are most useful for detecting the object of interest. It does this by using the positive training images to identify the features that are commonly present in those images, and the negative training images to identify features that are likely to occur in regions without the object. Once the training is complete, the resulting Haar Cascade Classifier can be used to detect the object of interest in new images. The algorithm applies the classifier to each sub-region of the image, and if enough of the classifiers indicate the presence of the object, the algorithm considers the sub-region to contain the object. Haar Cascade Classifier is commonly used for object detection in computer vision applications, such as face detection, pedestrian detection, and bike detection. As you mentioned, OpenCV provides several pre-trained cascading classifiers that can be used for these applications.

7.2 Open CV2 :

OpenCV is a widely-used open source computer vision and machine learning library that provides a wide range of functions and algorithms for image and video processing, object detection and recognition, and other computer vision tasks. It was originally developed by Intel in 1999, but is now maintained by the OpenCV community. The Viola-Jones algorithm, which is used in OpenCV for object detection, is a popular algorithm that uses Haar features and a cascade of classifiers to detect objects in images. Haar features are simple rectangular features that are computed at various positions and scales in an image, and are used to detect patterns and edges. The cascade of classifiers is trained on a set of positive and negative samples, and progressively reduces the false positive rate by applying more complex classifiers to regions that are likely to contain the object. To use OpenCV for object detection, we need to first train a cascade classifier using a set of positive and negative samples. The positive samples are images that contain the object you want to detect, and the negative samples are images that do not contain the object. The classifier is trained by iteratively applying a set of features to each sample and adjusting the weights of the features until the classifier can accurately distinguish between the positive and negative samples. Once the cascade classifier is trained, you can use it to detect the object in new images. The algorithm works by sliding a window over the image and applying the classifier to each sub-window to determine whether it contains the object. If the classifier detects the object, the algorithm outputs the location and size of the detected object. OpenCV also provides a number of other functions and algorithms for image and video processing, including filtering, segmentation, feature detection and extraction, optical flow, and camera calibration. These functions and algorithms can be used in a wide range of applications, from robotics and autonomous vehicles to medical imaging and video surveillance.

7.3. Image thresholding :

Image thresholding is a technique used in image processing and computer vision to create binary images from grayscale or color images. The idea behind thresholding is to separate pixels in the image based on their intensity values. In thresholding, a threshold value is chosen and each pixel in the image is compared to this value. If the intensity value of a pixel is above the threshold value, it is set to one, otherwise it is set to zero. Thresholding is a powerful tool for image segmentation, which is the process of dividing an image into meaningful parts or regions. It can be used to isolate objects or regions of interest in an image, remove noise or background, or create binary masks for further processing. OpenCV provides several thresholding functions, including cv. threshold(), cv. Adaptive .Threshold(), and cv.threshold2(). These functions take as input a grayscale image and a threshold value or parameters and return a binary image. The simplest form of thresholding is the binary thresholding, which is implemented using the cv. threshold() function in OpenCV. This function takes three arguments: the input image, the threshold value, and the maximum value. All pixels in the input image with intensity values greater than the threshold are set to the maximum value, and all other pixels are set to zero. There are several other types of thresholding, such as adaptive thresholding, Otsu's thresholding, and triangle thresholding, which are used in different situations depending on the characteristics of the input image and the desired output.

Fig. Image Thresholding

7.4 You Look Only Once :

YOLO is an object detection algorithm that uses CNNs to descry objects in real- time. It divides the image into a grid and predicts bounding boxes and class probabilities for each grid cell. The algorithm uses regression to predict the coordinates of the bounding boxes and class probabilities for each object in each grid cell. YOLO is known for its speed and accuracy, and has been widely used in various computer vision applications. The tiny YOLO is a lightweight version of the YOLO algorithm, designed for real-time object detection on resource-constrained devices, while YOLOv3 is the latest and most accurate version of the YOLO algorithm. Residual blocks: YOLO uses residual blocks, which are a special type of neural network layer that helps to alleviate the vanishing gradient problem that can occur in deep neural networks. Residual blocks allow for the network to learn and capture more complex features of the image, which can lead to better object detection performance. Bounding box regression: YOLO predicts bounding boxes around detected objects by using regression. Specifically, the algorithm predicts the bounding box coordinates (x, y, width, height) for each object detected in the image. The bounding box is represented by a set of four values, with (x, y) representing the coordinates of the top left corner of the box, and (width, height) representing the width and height of the box. Intersection Over Union (IOU): YOLO uses IOU to evaluate the accuracy of its object detection. IOU measures the imbrication between the predicted bounding box and the ground verity bounding box for each detected object. If the IOU value is above a certain threshold (e.g. 0.5), then the object detection is considered to be accurate.

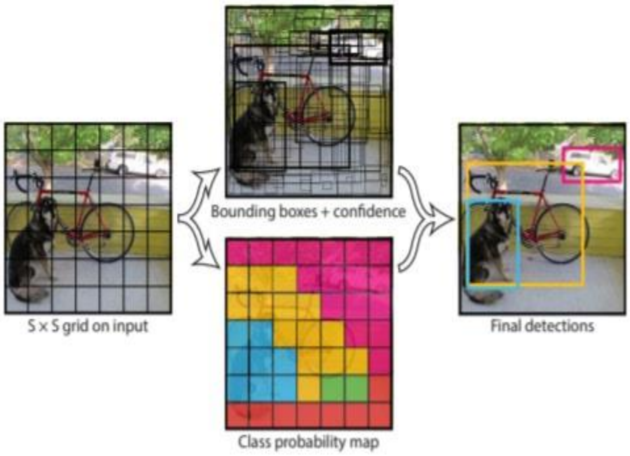

Fig. Yolo v3 image

VIII. EXPERIMENT RESULT

Fig. Output 1

Fig. Output 2

Fig. Output 3.

IX. CONCLUSION

In conclusion, the system for detecting helmet and number plate violation in motorcycles is an important step towards improving road safety. By using computer vision techniques such as CNN, the system is able to accurately detect whether a motorcyclist is wearing a helmet or not and recognize the license plate of the motorcyclist. This system can be integrated with existing traffic surveillance systems and used by law enforcement agencies to monitor and enforce helmet and number plate regulations. Additionally, the system can be further improved by increasing the training dataset and improving the image quality. Other computer vision techniques such as image segmentation and feature extraction can also be explored to improve the accuracy and efficiency of the system. With continued development and refinement, this system has the potential to significantly reduce the number of accidents and fatalities caused by helmet and number plate violations.

X. REFERENCES

[1]. J.Chiverton, "Helmet presence Classification with Motorcycle detection and tracking," in IET Intelligent Transport Systems, vol. 6, no 3, pp. 259-269, September 2012.

[2]. R. Silva, K. Aires, T. Santos, K. Abdala, R. Veras and A. Soares, "Automatic detection of motorcyclists without helmet," 2013 XXXIX Latin American Computing Conference (CLEI), Naiguata, 2013, pp. 1-7.

[3]. R. R. V. e. Silva, K. R. T. Aires and R. d. M. S. Veras, "Helmet Detection on Motorcyclists Using Image Descriptors and Classifiers," 2014 27th SIBGRAPI Conference on Graphics, Patterns and Images, Rio de Janeiro, 2014, pp. 141-148.

[4]. P. Doungmala and K. Klubsuwan, "Helmet Wearing Detection in Thailand Using Haar Like Feature and Circle Hough Transform on Image Processing," 2016 IEEE International Conference on Computer and Information Technology (CIT), Nadi, 2016, pp. 611-614.

[5]. K. Dahiya, D. Singh and C. K. Mohan, "Automatic detection of bike-riders without helmet using surveillance videos in real-time," 2016 International Joint Conference on Neural Networks (IJCNN),Vancouver, BC, 2016, pp. 3046-3051.

[6]. C. Vishnu, D. Singh, C. K. Mohan and S. Babu, "Detection of motorcyclists without helmet in videos using convolutional neural network," 2017 International Joint Conference on Neural Networks (IJCNN), Anchorage, AK, 2017, pp. 3036-3041.

[7]. N. Dalal and B. Triggs, “Histograms of oriented gradients for human detection,” in 2005 IEEE Computer Society Conference on Computer

Vision and Pattern Recognition (CVPR'05), vol. 1, June 2005, pp. 886–893 vol. 1.

[8]. R. R. V. e. Silva, K. R. T. Aires, and R. d. M. S. Veras, "Helmet detection on motorcyclists using image descriptors and classifiers," in 2014 27th SIBGRAPI Conference on Graphics, Patterns and Images, Aug 2014, pp. 141–148.

[9]. A. Hirota , N. H. Tiep, L. Van Khanh, and N. Oka, Classifying Helmeted and Non-helmeted Motorcyclists. Cham: Springer International Publishing, 2017, pp. 81–86 10. SahilBadla ., "Improving The Efficiency of Tesseract OCR Engine"., San Jose State University SJSU ScholarWorks, spring 2014.

Cite this article as :

K. N. Hande, Pooja Takliker, Lekha Bhoyar, Pranjali Bawankar, Piyush Navghare, "Design and Development of Automatic Detection System for Motercyclists without Helmet using Machine Learning ", International Journal of Scientific Research in Science, Engineering and Technology (IJSRSET), Online ISSN : 2394-4099, Print ISSN : 2395-1990, Volume 10 Issue 2, pp. 670-678, March-April 2023. Available at doi : https://doi.org/10.32628/IJSRSET23102117

Journal URL : https://ijsrset.com/IJSRSET23102117