Multimedia Tools and Applications
https://doi.org/10.1007/s11042-020-08627-w

# Automated detection of helmet on motorcyclists from traffic surveillance videos: a comparative analysis using hand-crafted features and CNN

**Linu Shine**[1] · **Jiji C. V.**[1]

Received: 18 April 2019 / Revised: 7 November 2019 / Accepted: 2 January 2020 /
Published online: 12 February 2020


**Abstract**
The higher mortality rate in motorcycle accidents is attributed to negligence in wearing a helmet by two-wheeler riders. Identification of helmetless riders in real-time is an essential task to prevent the occurrence of such events. This paper presents an automated system to identify motorcyclists without a helmet from traffic surveillance videos in real-time. The problem becomes more challenging when computational resources are limited. We have compiled a custom dataset for developing an automated helmet detection algorithm. The proposed system uses a two-stage classifier to extract motorcycles from surveillance videos. Detected motorcycles are further fed to a helmet identification stage. We present two algorithms for classifying riders with and without a helmet, one based on hand-crafted features and the other based on deep convolutional neural network (CNN). Our experiments show that the proposed CNN model gives the best performance in terms of accuracy while the feature-based model gives faster detection. Most importantly, to ensure the light-weightiness of the proposed system all the computations are performed in CPUs only.

**Keywords** Helmet detection · Motorcycle classification · Foreground segmentation · Vehicle tracking

## 1 Introduction

Rising income and an ever-increasing need for mobility fuels an exponential growth in vehicle population across the globe. When a slow evolving traffic infrastructure meets this rising vehicle population, it creates havoc and traffic congestions. In such cases, where time to destination prioritize the list, motorized two-wheelers happen to be the preferred means of transport, especially in low and middle-income countries. The small size, cost and ability

✉ Linu Shine
linushine@cet.ac.in

Jiji C. V.
jijicv@cet.ac.in

[1] Department of Electronics and Communication Engineering, Computer vision Lab, College of Engineering Trivandrum, Kerala, India

to traverse quickly through heavy traffic, catalyses the growth of two-wheelers on the road. The rise in two-wheeler sales across the globe in each year validates this fact [16].

In most developing countries, there are no separate lanes for bicycles or motorized two-wheelers. Since they share common traffic space with fast-moving trucks, buses, and cars, two-wheelers are at a high risk of being involved in accidents. It is no wonder they top the fatality charts in road accidents and they notoriously guard the premier place in accident graphs every year [22]. Negligence in wearing a helmet, the only safety equipment for two-wheelers, escalates the death rate in motorcycle accidents. Even though the usage of a helmet is mandatory by law in many countries, manual checking is employed to check law infringements. Lack of sufficient manpower seems to be a bottleneck in such systems.

The global status on road safety [25] points out that proper enforcement of proven measures can bring down the death rate in road accidents. An automated system, if employed to detect and penalize motorcycle riders without a helmet, will naturally motivate the two-wheeler riders to wear helmets. This will also reduce the requirement for manpower in monitoring the traffic for detecting traffic violations. This motivates us to propose an end-to-end system based on three modules to detect motorcyclists without a helmet from traffic surveillance videos in real-time.

The first module extracts moving objects (mostly vehicles) from videos employing background subtraction. Extracted objects are then passed through the second module which segments out motorcycles from the massive complex mixture of vehicles. The final module extracts the head region and detects whether riders use a helmet or not. The block diagram of the proposed module is illustrated in Fig. 1.

We employ a combination of gradient and texture features to identify riders without a helmet. We also use a convolutional neural network(CNN) for helmet classification and also present a performance comparison between hand-crafted features and CNN. Though a lot many researchers [5, 6, 11, 23, 33] across the globe have already developed such automated systems, we have improved the motorcycle extraction and helmet detection stages with the integration of multiple features.

The contributions of our paper are

- An end-to-end real-time system for helmet detection which includes
  - a two-stage approach for motorcycle classification.
  - a novel concatenated hand-crafted features for helmet identification.
  - a custom CNN model for helmet detection.
- A custom dataset developed for training and testing the proposed system.

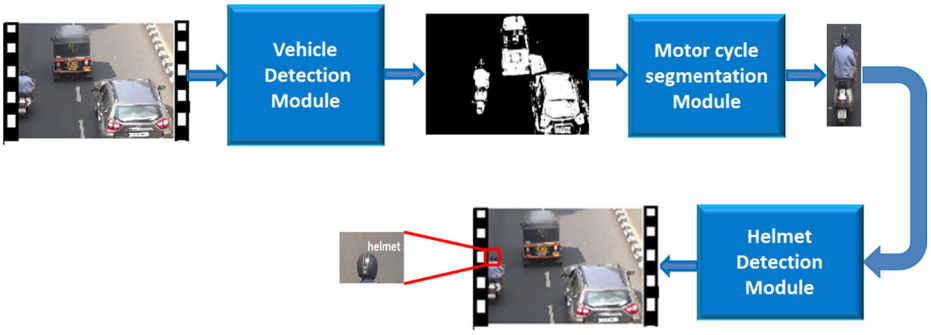



**Fig. 1** Block-level representation of the proposed system for helmet detection

The paper is organized into six sections. Section 2 reviews the most recent studies that have been published in this area. Section 3 details the proposed methodology for automated helmet detection. In Section 4 we describe the experimental setup and the dataset. Section 5 discusses the results obtained during all stages of the system, including metrics for the evaluation. This section also provides a comparative analysis between helmet detection using hand-crafted features and convolutional neural networks. We conclude the discussions with recommendations for future studies in Section 6.

## 2 Related work

Recently, several studies have been conducted for traffic video analysis to determine the flow, density, velocity, etc., of vehicles on the road. This section reviews the current state of the art in fields of moving vehicle detection, motorcycle segmentation, and helmet detection.

### 2.1 Moving vehicle detection

For object segmentation, several state-of-the-art methods based on active contour modelling are available [3, 36, 37]. However, for moving object segmentation, conventional approaches like temporal differencing, background subtraction and optical flow methods [15] are widely used. To be specific, for traffic surveillance systems background subtraction methods are used since surveillance cameras are assumed to be stationary. In this method, the current image is subtracted from a reference background image, which is upgraded during a period. The subtraction leaves just non-stationary or new objects, where it gives the entire outline of the new object.

Background subtraction involves background modelling, background initialization, background maintenance, and foreground identification [2]. Background modelling methods are categorized as basic background modelling [21, 39], statistical background modelling [7, 30], background modelling via clustering [17], neural network background modelling [27], and background estimation [35]. The initialization can be done using temporal statistical methods like randomly selected frames [31] or a mixture of Gaussians [30]. Maintenance rule deals with updating the pixels with blind or a variable learning rate [9] based upon whether it was classified into foreground or background in the previous classification [1]. Foreground classification deals with the classification of pixels into the foreground or background[4].

### 2.2 Motorcycle segmentation

Background subtraction leaves white patches (blobs) in a dark background in correspondence with moving objects in the frame. Vehicle classification can be done either by using parameters of these blobs or by using classifiers. The parameters of the blob, like position, length (L) and width (W) obtained after background subtraction is used to classify vehicles[13, 20]. In [20], key features like length and width from each category of vehicles are applied to a decision-tree method for classification. Such systems are sensitive to camera calibrations, and features should be adjusted when applied to a different road. To overcome this difficulty, Chiu et al [5] suggested the usage of pixel ratio to identify the motorcycle. In certain works [6], the aspect ratio of the bounding box of the blob is used as a measure to detect motorcycles from frames. Area and standard deviation of the hue around the blob centre are also used along with aspect ratio to separate the motorcycle from the rest of the

vehicles [33]. In this method, the model is developed for camera orientation perpendicular to the vehicle movement and hence, chances of occlusion by a larger vehicle are very high which leads to false negatives.

Appearance-based features like hough transform, corners, and tyre characteristics are used to demarcate motorcycles [23]. Some researchers suggest extracting features like the histogram of oriented gradients (HOG) from blob regions and to classify them into motorcycles and non-motorcycles. Silva *et al.* [11] proposed a method in which, the system uses the wavelet transform (WT) descriptor and the random forest classifier. Chiverton [6] also suggested the usage of HOG features [8] to detect a motorcycle with a support vector machine (SVM) classifier [28]. In [34], a hierarchical SVM that uses a hybrid descriptor with local binary patterns (LBP), hu moment invariants (HMI) and color histograms (CH) is used to classify helmets. All these methods rely on hand-crafted features and have stable accuracy compared to the parameter-based approach. The downside of this approach is that it requires a training phase and has a slight computational overhead compared to a parameter-based approach.

### 2.3 Helmet detection

Helmet detection is mainly done using a classifier trained on hand-crafted features. Some literature suggests the use of circular hough transform (CHT) to extract areas corresponding to the head region[29]. Since the human head and helmets have a similar geometric pattern, this method when taken alone for helmet identification degrades the performance of the system. Chiu et al. [5] proposed algorithms to detect occluded motorcycles using the visual length, visual width, and pixel ratio. During the calibration stage of this method, the operator needs to input parameters like helmet radius, camera angle, camera height, etc. So the system needs a recalibration whenever the camera parameters change.

In most of the works, HOG features of the region of interest (ROI) are extracted and trained with SVM classifier to classify riders with and without helmets [5, 6, 23]. Silva et al. [11] proposed a method in which the circular Hough transform (CHT) and the HOG descriptor were the image features, and used the multilayer perceptron (MLP) classifier for helmet detection. The algorithm for helmet detection accomplished an accuracy of 91.37%. The results were obtained with the author's database. In this method, shape-based features alone are considered for helmet identification. In [33], ROI is again divided into 4 regions. Features like arc circularity, average intensity, and average hue in the quadrant are the features used for training the K-Nearest Neighbour(KNN) classifier. The system claims 74% accuracy in detecting helmets. The main advantage of this method is the ability to check the helmet for the pillion rider as well. The side view of the vehicle is taken to detect the presence of a helmet. This is a flaw in the system. The license plate details cannot be extracted since the camera is placed perpendicular to the vehicle movement. The possibility of occlusion of motorcycle riders is also high in such systems

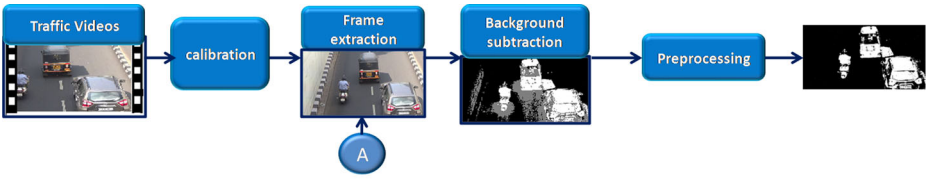
Vishnu et al. [32] proposed a CNN based system for detecting riders without a helmet from surveillance videos with an accuracy of 92.87% with their datasets. They have used AlexNet as their CNN architecture and used a graphic processing unit(GPU) based system for training.

In this paper, we present a fully automated system working in real-time, to segment two-wheelers from surveillance videos with fewer computations. A two stage cascaded classifier is used to extract motorcycles from surveillance videos. Here, we use an effective combination of a parametric based and feature based classifier, thereby combining the

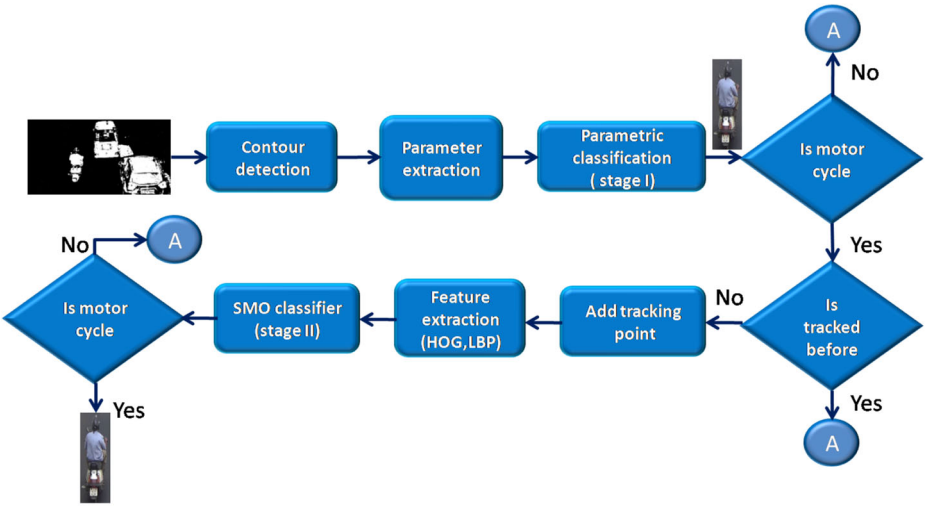
advantages of the two approaches. Head regions are classified into riders with helmets and without helmets using hand-crafted features and also with a custom CNN. The hand-crafted features use a novel combination of shape as well as texture features. The proposed CNN model has less number of trainable parameters when compared with standard models.

## 3 Proposed methodology

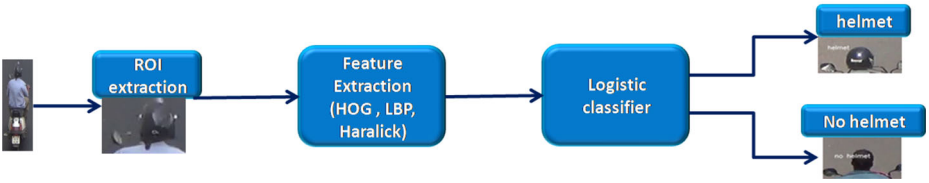
The proposed vision-based system successfully combines three modules sequentially to monitor traffic surveillance videos in real-time and identify motorcyclists travelling without a helmet. The first module comprises a vehicle detection stage, which is followed by the second module for the segmentation of motorcycles and the final module detects the motorcyclists without a helmet. The detailed block diagram of the end-to-end automated helmet detection system is shown in the Fig. 2.




(a) Moving vehicle detection module




(b) Motorcycle segmentation module




(c) Helmet Detection module

**Fig. 2** The proposed end-to-end helmet detection system

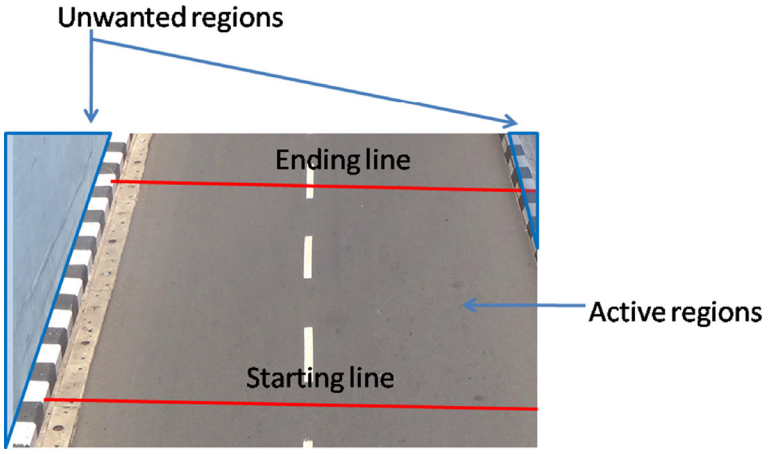



**Fig. 3** Output of calibration stage

The following subsections provide a detailed explanation of each module.

### 3.1 Moving vehicle detection module

This module is employed to extract moving vehicles from input surveillance videos. The block-level representation of the system is shown in Fig. 2a. It consists of a calibration stage and a foreground extraction stage. The calibration stage discards the unwanted regions in the surveillance video (see Fig. 3).

Further, starting and ending lines are drawn to fix regions within which capturing and tracking of vehicles are to be done. These lines help in reducing the computational load and memory required for the system since capturing and analyzing is done only when vehicles are within the periphery of the starting line.

To extract moving vehicles from the active region, we use an improved adaptive Gaussian mixture model as proposed by Zivkovic[40]. This background subtraction method is invariant to lighting changes, repetitive motions, clutter and shows good results even with slow-moving objects. Here, a mixture of Gaussian distributions is used to represent each pixel. The value of a pixel at any time t is denoted by $\Phi_t$. Now a Bayesian decision $\Upsilon$ decides whether a pixel belongs to the background (BG) or foreground (FG)

$$\Upsilon = \frac{P(BG/\Phi_t)}{P(FG/\Phi_t)} = \frac{P(\Phi_t/BG)P(BG)}{P(\Phi_t/FG)P(FG)} \tag{1}$$

The resulting blobs from the background subtraction stage correspond to moving objects. These are passed to the motorcycle classification stage after some preprocessing. There are complex dehazing methods to retrieve the image from a noisy background [38]. But to reduce computational complexity, we first convert the background-subtracted image into a

binary image. The resulting images are further enhanced using morphological closing and opening.

### 3.2 Motorcycle segmentation module

This stage in the proposed system deals with motorcycle classification, segmentation, and tracking. The block-level architecture of this module is shown in Fig. 2b. First, every foreground pixels are gathered into connected regions (blobs) and extracted using a two-level connected component labelling method by Dillen [10],[12]. Further, we used two-stage classification to segment motorcycles from these blobs.

In the first stage parameter based classification is utilized. Parametric features like area, perimeter, aspect ratio and extent of each blob in a frame are calculated. In the proposed approach the camera is assumed to be stationary and hence camera parameters like orientation, resolution, etc. are fixed. The camera view is restricted to 7 meters lengthwise of the road. So the aforementioned parametric features are fixed for each type of vehicle. The parametric feature range for motorcycle is calculated beforehand by trial and error. The extracted parametric features from each blob are compared with the saved parametric feature set of a motorcycle. If the similarity measure between the two lies within a threshold, then a rectangular bounding region, in correspondence with the matching blob, is segmented from the original frame and kept as a motorcycle. Parameter based classification filters out most of the foreground regions corresponding to non-motorcycles with less computational overhead. Here, a tracking point is added to remove multiple instances of the same motorcycles.

The accuracy of an automated helmet detection system depends on how well the system segregates motorcycles from the surveillance videos. So to further eliminate misclassification a second classifier is also employed. The second stage is a feature-based SMO classifier, which uses concatenated HOG and LBP as a feature vector. The gradient (HOG) and texture (LBP) features from the segmented regions are extracted and given to a trained SMO classifier to sieve only motorcycles at the output of this stage. In this system, we need to employ feature extraction and classification only to those blobs whose features are identical to that of motorcycles.

### 3.3 Helmet detection

In this stage, helmet detection is done on the segmented motorcycle regions from the previous stage. The block-level representation of this module is shown in Fig. 2c. The motorcycle rider's head usually appears in the upper part of the image. The top 20% height of the motorcycle image is defined as ROI. Extracted ROI's are converted into grayscale images and further resized into $64 \times 64$ pixels. The use of this small area for helmet detection further reduces the processing time and has better accuracy than using the complete motorcycle image. To further classify these ROI's into helmet/non-helmet, we experimented with both hand-crafted features and CNN features.

#### 3.3.1 Hand-crafted features

HOG, LBP and Haralick features are extracted from the head region. Shape information is obtained from the HOG feature vector while the other two provide texture information. This study proposes the usage of a novel combination of shape as well a texture features to identify helmet.

**Histogram of Oriented Gradients(HOG)** [8]**:** We use a 576 long feature $l_2$-normalized HOG descriptor given by

$$f_1 = \frac{v}{\sqrt{\|v\|_2^2 + \epsilon^2}} \tag{2}$$

where $v$ is the HOG feature descriptor which gives the frequency of orientation of image gradients and $\epsilon$ is a constant.

**Local Binary pattern (LBP):** Local binary pattern [24, 26] is a texture descriptor, which labels the pixels by comparing it with neighbourhood pixels and assigns a unique number using

$$f = \sum_{p=0}^{P-1} s(gr_p - gr_c) \tag{3}$$

$$s(gr_p - gr_c) = \begin{cases} 1, & gr_p \geq gr_c \\ 0, & otherwise \end{cases} \tag{4}$$

where $gr_c$ and $gr_p$ denote the gray values of the central pixel and its neighbour, s is a thresholding function, and P is the number of neighbours. There were 129 uniform LBP patterns in each image which amounts to a feature length of 130.

**Haralick features:** Haralick features are another set of texture indicators using the idea of statistical indicators, originally proposed in [14]. A group of 13 descriptors with unit distance is evaluated from each co-occurrence matrix calculated at $\theta$= 0° ,45° , 90° , 135° . For the grayscale image, haralick feature descriptor had a length of 52. The fourteenth haralick feature is omitted in our calculation.

Most of the studies[5, 6, 29] for helmet detection use feature descriptors capturing the shape information alone. The textures of the head region with a helmet and without a helmet are widely different. If this information is also considered along with the shape information, it can increase the accuracy of the overall system. LBP and Haralick descriptors capture texture information, while HOG descriptors will have the shape information. This motivated us to propose a system that uses an effective combination of feature descriptors that provide shape as well as texture information. Our feature descriptor is formed by the concatenation of HOG, LBP and Haralick features.

The next step is the training of the classifier. Supervised learning is employed here. The feature vectors extracted from the head images are concatenated and placed as row vectors, along with its labels “helmet” and “no helmet”. These concatenated feature vectors are fed to the Logistic classifier which uses the multinomial regression model with ridge estimator[18, 19] along with their class label. The feature-length when texture and gradient features are concatenated is 758. In the classification phase, an unlabelled vector (a query or test point) is classified by assigning a new label predicted by a logistic classifier. The hand-crafted feature based classifier system is found to work efficiently if the camera parameters are fixed. Whenever there is a significant change in camera parameters like zoom, orientation, etc., the accuracy of the system based on classifier is affected. So we have proposed a CNN based system for helmet classification.

### 3.3.2 Convolution Neural Network (CNN)

In the visual classification arena, CNN has transcended many of the classical machine learning techniques. They are similar to neural networks but preserve the spatial dimension of

an image. Deep learning requires tuning of millions of parameters, so a large number of examples are required for training. We have used data augmentation techniques like flipping, rotation, etc. to enhance datasets since CNN is translational and viewpoint invariant. The final dataset consists of 1500 images in each category.

The segmented ROIs are directly fed to the CNN network for classification. Here no hand-crafted features are required as CNN has the capability of learning necessary features for differentiating the classes during the training process. Lower layers of a CNN architecture learn low-level features like corners, oriented edges, etc., from an input image. Higher-order convolution layers are responsible for learning more complex features like texture. The popular CNN models for image classification have complex architectures as they are built to differentiate between thousands of classes. Chances of overfitting are high when pretrained models are used for simple predictions. Moreover, they require huge computational power and need a GPU based system for real-time applications. We plan our end application to be run in real-time on a cost-effective embedded system with less compute power compared to costly GPU based systems. So there is a stringent limitation on the number of computations allowed without trading accuracy. This motivated us to develop our Custom CNN for helmet identification.

Our CNN model operates on a 64×64 RGB image. The flow diagram of our Custom CNN is shown in Fig. 4. Our Custom CNN network consists of 7 convolution layers, 2 maxpool layers, and 2 fully connected layers. Low-level features from the image are captured by convolving 32 filters of size 3 ×3. This is followed by ReLU activation. Feature map size is then reduced to half by using maxpooling. Feature size is halved again after another convolution layer. In the subsequent 3 convolution layers, the number of feature maps is increased to 64. The number of activation maps gets reduced back to 32 in the following 2 convolution layers. All convolution layers use strided convolutions to reduce the feature sizes. Convolution layers are followed by 2 fully connected layers. ReLU activation function is applied after every convolution layer and to the first fully connected layer. A drop out factor of 0.5 is also used to prevent overfitting in the first fully connected layer. The last fully connected layer uses a sigmoid activation function. We have arrived at this configuration by trial and error, considering losses in early epochs and the number of trainable

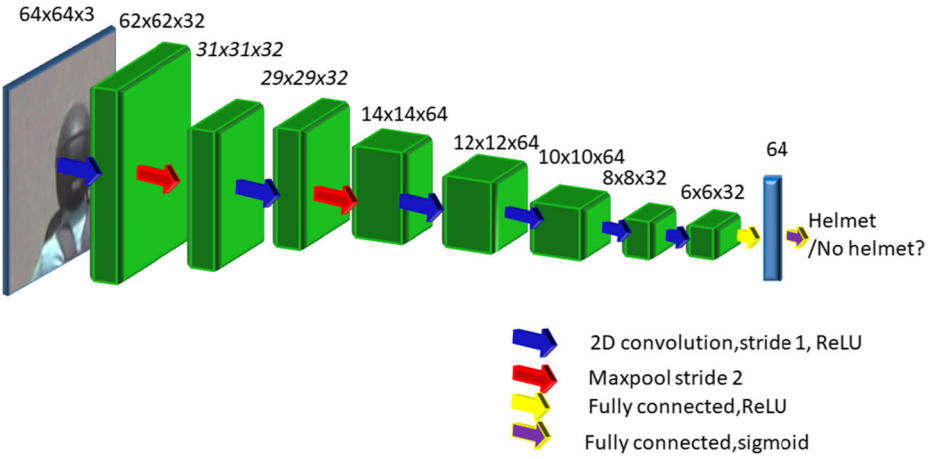



**Fig. 4** Flow diagram of Custom CNN for helmet classification

**Table 1** Architecture of Custom CNN for helmet classification

| Layer | Input | Kernal size | Kernals | Stride | Output | Parameters |
|---|---|---|---|---|---|---|
| Conv1 | 64 x 64 x 3 | 3 x 3 x 3 | 32 | 1 | 62 x 62 x 32 | 896 |
| Pool1 | 62 x 62 x 32 | 2 x 2 x 1 | _ | 2 | 31 x 31 x 32 | _ |
| Conv2 | 31 x 31 x 32 | 3 x 3 x 32 | 32 | 1 | 29 x 29 x 32 | 9248 |
| pool2 | 29x 29 x 32 | 2 x 2 x 1 | _ | 2 | 14 x 14 x 32 | _ |
| Conv3 | 14 x 14 x32 | 3 x 3 x 32 | 64 | 1 | 12 x 12 x64 | 18496 |
| Conv4 | 12 x 12 x 64 | 3 x 3 x 64 | 64 | 1 | 10 x 10x 64 | 36928 |
| Conv5 | 10 x 10 x 64 | 3 x 3 x 64 | 64 | 1 | 8 x 8 x 64 | 36928 |
| Conv6 | 8x 8 x 64 | 3 x 3 x 64 | 32 | 1 | 6 x 6 x32 | 18464 |
| Conv7 | 6x 6 x 32 | 3 x 3 x 32 | 32 | 1 | 4 x 4 x32 | 9248 |
| FC1 | 4 x 4 x32 | 512 | 64 | _ | 64 | 32832 |
| FC2 | 64 | 64 | 2 | _ | 2 | 130 |

parameters. The architecture of the Custom CNN model used for helmet classification is shown in Table 1.

Our network uses only 163,170 trainable parameters compared to 62.3 million parameters in Alexnet which is used in [32]. Therefore the computing power required for our Custom CNN, for both training and testing is less when compared with [32] when training from scratch. This is an important feature when deploying a CNN network on an embedded platform.

## 4 Experimental setup and data set

We have compiled a new dataset for the work since no benchmarked datasets are available. The database used for tests and development of the system is from videos captured on public roads. The experimental setup for video capturing consists of a video camera (Sony HDR-pj380E) mounted on a tripod kept on an overbridge, with the camera focusing traffic through an underpass. Videos were captured with resolution 1250×720 pixels with a frame rate of 25 fps. The dataset comprises of video snippets of duration ranging from 2 minutes to 16 minutes, which amounts to a total of 120 minutes. All the videos taken for this study comprises of the camera focusing on the rear side of the vehicle. This was done to ensure that once the non-use of a helmet is detected, we can also take the registration details of the vehicle from its license plate automatically by using automatic number plate recognition(ANPR) techniques.

The videos were recorded during day time with different lighting conditions at different times and on different days. The dataset consists of a total of 3159 vehicles, of which 1645 are two-wheelers. Here we have considered all motorized two-wheelers as a single category - motorcycle. It includes motorbikes and scooters. The system consists of 199 single riders without a helmet, 974 helmet users and the remaining consists of more than one rider. For a balanced class, we have used an equal number of riders with and without a helmet in the experiment. Images from the sample dataset are shown in Fig. 5. All the images used in the experiment were extracted using the procedure explained above. Twenty percent of videos are kept for testing and the rest is used for training.

**Fig. 5** Sample frames from datasets

All experiments were done in a 64-bit Intel core i7 machine working on 3.4 GHz clock frequency and 8 GB RAM. Usually, a GPU is used for training deep networks. Since the number of trainable parameters is less when compared with the popular CNN network, we could train our Custom CNN with the above mentioned Intel i7 machine without any GPU. We have used python, skicit, Keras, sklearn, opencv, mahotas, weka for developing our system and verification.

### 4.1 Evaluation metrics

The evaluation metrics used in the study are defined as follows.

Let T1 denote the number of cases with helmet predicted correctly, T2 denotes the number of cases without helmet predicted correctly, T3 denotes the number of cases without helmet wrongly detected as with helmet by our algorithm and T4 denotes the number of cases with helmet detected as without helmet by our algorithm.

*Accuracy* (**A**) is the ratio of correctly predicted observations to total observations.

$$A = \frac{T1 + T2}{T1 + T2 + T3 + T4} \tag{5}$$

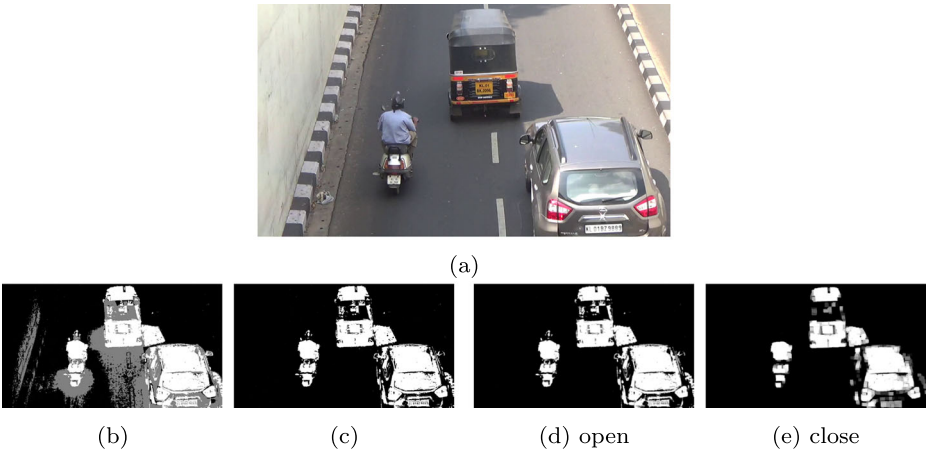
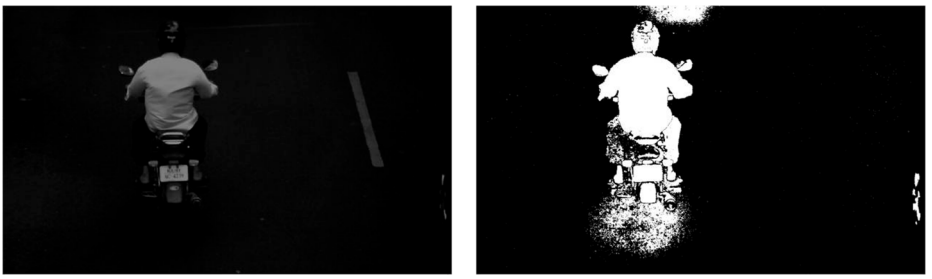



**Fig. 6** **a** Original Image **b** Background subtracted image **c** Binary image **d** Morphological Close **e** Morphological Open



(a) (b)

**Fig. 7** **a** Frame captured in dark-time **b** Background subtracted image

*Precision* (**P**) is the ratio of correctly predicted helmet cases to the total number of cases predicted as helmet.

$$P = \frac{T1}{T1 + T3} \tag{6}$$

*Recall* (**R**) is the ratio of correctly predicted helmet cases to the total number of helmet cases. Also known as True Positive Rate(**TPR**).

$$R = \frac{T1}{T1 + T4} \tag{7}$$

*F1 Score* (**F1**) is the weighted average of precision and recall.

$$F1 = \frac{2RP}{R + P} \tag{8}$$

*Kappa coefficient* (**K**) measures the agreement between the accuracy of the system to the accuracy of a random system, is: where accuracy of random system,

$$Ar = \frac{(T2 + T3)(T2 + T4) + (T1 + T4)(T1 + T3)}{(T1 + T2 + T3 + T4)^2} \tag{9}$$

$$K = \frac{A - Ar}{1 - Ar} \tag{10}$$

*False Positive Rate*(**FPR**) is the ratio of incorrectly predicted cases without a helmet to the total number of cases without a helmet.

$$R = \frac{T3}{T2 + T3} \tag{11}$$

**Fig. 8** Sample results of vehicle segmentation

**Table 2** Performance of SMO classifiers with 10 fold cross validation with various descriptors for motorcycle classification

| Descriptor | A % | K | P | R | F1 | ROC area |
|---|---|---|---|---|---|---|
| HOG[1] | 97 | 0.95 | 0.98 | 0.97 | 0.975 | 0.975 |
| LBP[2] | 97.5 | 0.95 | 0.961 | 0.99 | 0.975 | 0.975 |
| [1]+[2] | **99.5** | **0.99** | **99.5** | **0.99** | **0.995** | **0.99** |

Bold indicates top values in each category

*Receiver Operating Characteristic* (**ROC**) is a plot between TPR and FPR for different cut-off points.

## 5 Results

This section deals with the results of our experimentation. In addition to the results of our proposed system, we have also done a comparative analysis of the performance of descriptors and classifiers. The following metrics were used in evaluating the performance of different classifiers with different combinations of feature descriptors: Accuracy(A), Kappa coefficient(K), F-measure(F)or F1 score, Recall(R), Precision(P), ROC Area. These metrics are calculated while performing a 10 fold cross-validation of the classifiers. The Receiver operating characteristics(ROC) is also plotted. The classifiers used in the study are Naive Bayes classifier, SVM with linear kernel, Stochastic gradient descend(SGD) classifier with SVM as hinge loss function, Sequential minimal optimization algorithm(SMO) for training SVM classifier , Logistic classifier which uses multinomial regression model with ridge

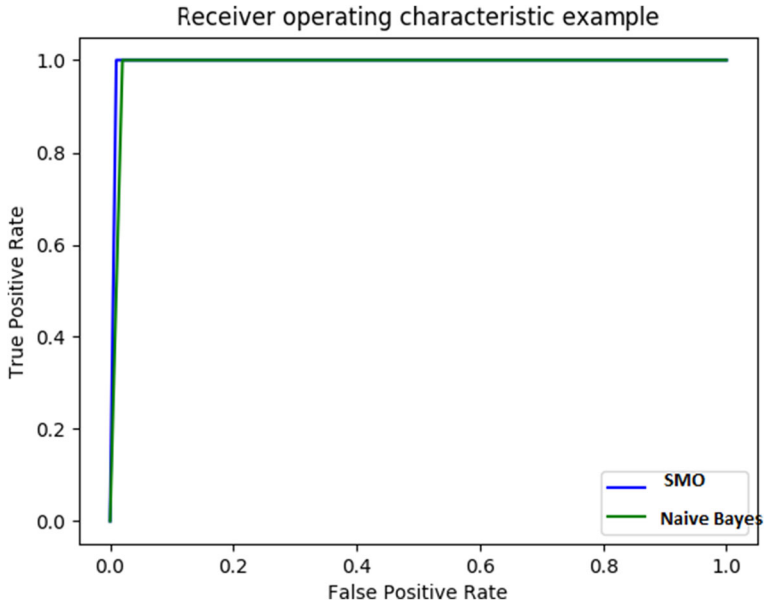



**Fig. 9** ROC of motorcycle

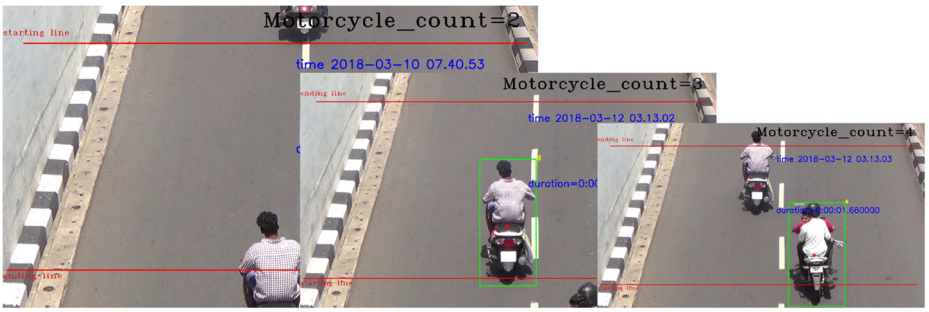



**Fig. 10** Sample results for tracking

estimator and Random forest classifier. For evaluating the performance of these classifiers with a different combination of hand-crafted feature descriptors, a 10 fold cross-validation is used to generate results for comparison. There are two main components in our system: motorcycle detection and helmet identification. The experiments were done separately for each of these components, with the images extracted from videos. This section is divided into results of moving vehicle detection, motorcycle segmentation and results of helmet detection.

## 5.1 Moving vehicle detection

Surveillance videos are first read frame by frame. Background subtraction using AGMM is applied to each frame, to obtain the foreground images. The resulting images are binarized and morphological operations like closing and opening are applied to remove noises. Figure 6 represents the result of background subtraction and the results of post processing. These enhanced images are used to separate the white blobs, which mostly correspond to vehicles in the next stage. Figure 7 shows the output of moving vehicle detection for a frame captured in dark-time.

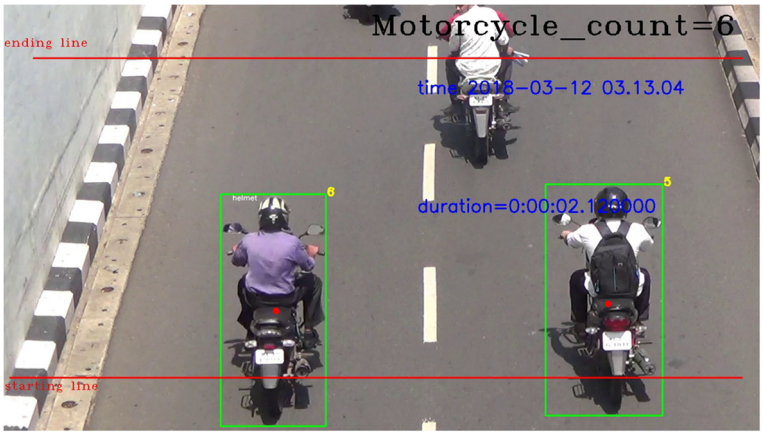



**Fig. 11** Tracking and detecting multiple motorcycles in a single frame

**Table 3** Performance of logistic classifier with 10 fold cross validation with various combination of descriptors for helmet classification

| Descriptor | A % | K | P | R | F1 | ROC Area |
|---|---|---|---|---|---|---|
| HOG[1] | 83.42 | 0.668 | 0.834 | 0.834 | 0.834 | 0.858 |
| LBP[2] | 80.4 | 0.608 | 0.804 | 0.804 | 0.804 | 0.798 |
| Haralick [3] | 92.96 | 0.859 | 0.93 | 0.93 | 0.93 | 0.957 |
| [1]+[2] | 93.72 | 0.874 | 0.93 | 0.93 | 0.93 | 0.978 |
| [1]+[3] | 83.42 | 0.668 | 0.835 | 0.834 | 0.83 | 0.858 |
| [2]+[3] | 86.67 | 0.733 | 0.83 | 0.83 | 0.83 | 0.894 |
| [1]+[2]+[3] | **96.98** | **0.94** | **0.97** | **0.97** | **0.97** | **0.994** |

Bold indicates top values in each category

**Table 4** Performance of classifiers with 10 fold cross validation with concatenated features of HOG,LBP and Haralick descriptor for helmet classification

| classifier | A % | K | P | R | F | ROC Area |
|---|---|---|---|---|---|---|
| Naive Bayes | 89.20 | 0.784 | .893 | 0.892 | 0.892 | 0.936 |
| SVM | 87.94 | 0.759 | 0.887 | 0.879 | 0.766 | 0.879 |
| SGD | 96.48 | 0.93 | 0.955 | 0.955 | 0.955 | 0.965 |
| SMO | 96.73 | 0.934 | 0.967 | 0.967 | 0.967 | 0.967 |
| Random Forest | 94.97 | 0.89 | 0.945 | 0.945 | 0.945 | 0.992 |
| Logistic | **96.98** | **0.94** | **0.97** | **0.97** | **0.97** | **0.994** |

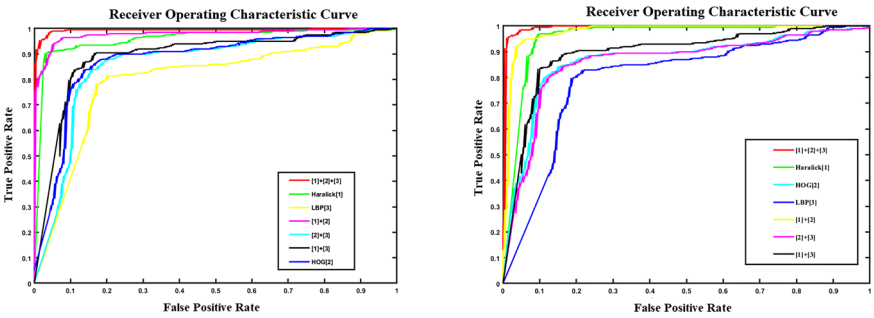
Bold indicates top values in each category

**Table 5** Comparison of proposed method with existing works

| Methods | A % | K | P | R | F | ROC Area |
|---|---|---|---|---|---|---|
| [6, 23] | 85.17 | 0.852 | 0.834 | 0.834 | 0.852 | 0.800 |
| [11] | 91.35 | 0.829 | 0.73 | 0.83 | 0.83 | 0.908 |
| [34] | 92.4 | 0.85 | 0.814 | 0.804 | 0.804 | 0.918 |
| proposed method | **96.98** | **0.94** | **0.97** | **0.97** | **0.97** | **0.994** |

Bold indicates top values in each category


**Fig. 12** Sample helmet classification results




(a) ROC of logistic classifier with helmet (b) ROC of logistic classifier with out helmet

**Fig. 13** Receiver Operating Characteristics

## 5.2 Motorcycle segmentation

Motorcycle detection consists of two phases as described above. In the first phase where blob parameters are used to separate motorcycles, the accuracy obtained is 96.5%. The false positives obtained in this case are noises due to dust movement in wind, reflections from heavy trucks and one or two autorickshaws. The blobs having similar geometric features of motorcycles are extracted from each frame. They are converted to grayscale for feature extraction and classification in the second phase. The results of motorcycle segmentation are shown in Fig. 8. These images are from different roads under different lighting conditions.

Table 2 shows a comparison chart of HOG, LBP and a hybrid descriptor which is a concatenated feature vector of LBP and HOG with SMO classifier.

From the Table 2, it is clear that when gradient and texture features are concatenated and used with SMO classifier performance increases to 99.5% with 10 fold cross-validation. The Kappa coefficient value also lies well above in the excellent range [29]. The ROC curves are also plotted for the motorcycle in Fig. 9.

### 5.2.1 Tracking

This section deals with the results of tracking. Here, multiple detections of the same motorcyclists is avoided. Each figure in Fig. 10 shows the detected vehicles marked using a bounding box (green colour). The number (yellow colour) at the top of each bounding box shows the tracking number of the vehicle. The centroid is shown as a dot (red colour).

The first frame in Fig. 10 shows that the motorcycle count was 2 when a new motorcycle just crosses the starting line. In the second frame, the motorcycle is detected and found to be a new entry, and therefore the motorcycle count is incremented to 3. The last frame in the figure shows that the count remains the same till a new vehicle enters the scene and gets detected. This system is also able to track and detect multiple motorcycles in a single frame and is shown in Fig. 11.

**Table 6** Performance of Custom CNN

| classifier | A % | K | P | R | F | ROC Area |
|---|---|---|---|---|---|---|
| Custom CNN | 99.62 | 0.99 | 1 | 1 | 0 .99 | 1 |

Bold indicates top values in each category

## 5.3 Helmet detection

### 5.3.1 Hand-crafted features

This section deals with the comparison of feature descriptors for identifying helmets.

Table 3 shows the performance of the logistic classifier with different combinations of feature vectors. From the table, it is clear that when a gradient feature (HOG) is concatenated with a texture feature (LBP & Haralick), it improves the classification accuracy. When haralick, HOG and LBP descriptors are concatenated, the Logistic classifier shows an accuracy of 96.98%. Though HOG features capture the shape of the object, it alone cannot differentiate between heads and helmets since both have a near to circular shape. The texture feature (LBP) when combined with HOG, shows a significant improvement in performance. As shown in the ROC curve, the results are further improved by including an additional texture feature (haralick). This is because texture variations of the head and the helmet are effectively captured by haralick and LBP descriptors and results in a superior descriptor when combined with HOG.

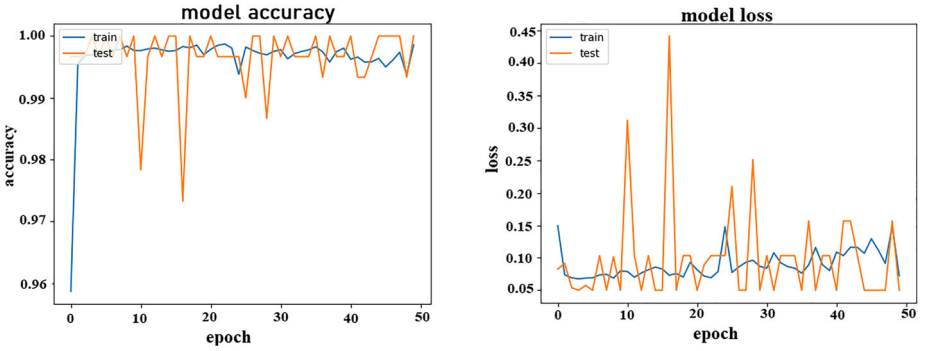
After deciding on the hand-crafted feature combination, we tested the performance of different classification algorithms. From Table 4, it is clear that the Logistic classifier is ahead of all other classifiers in terms of evaluation metrics.

Next, we compared the proposed methodology with existing helmet detection works in Table 5. Here, [11] used CHT and HOG features with Random forest classifier whereas [6] and [23] used HOG + SVM combination. In [34], colour information is also incorporated using color histograms and HMI descriptors along with LBP. When compared with these methods our proposed hand-crafted feature combination shows superior performance.

Figure 12 shows a sample of classified ROI from different traffic scenes.

It is seen that Logistic classifier has the highest performance when all the three features are concatenated and evaluated with the performance of the classifiers as shown in Table 4

The Receiver operating characteristics of the Logistic classifier of helmet and no helmet case is plotted in Fig. 13a and b. The disadvantage with hand-crafted features is that, if there is variation in camera parameters, the accuracy of the system changes accordingly. This can be mitigated if we are using CNN based system.




(a) Accuracy vs Epoch for Custom CNN (b) Loss vs Epoch for Custom CNN

**Fig. 14**

**Table 7** Comparison of standard CNN models with the proposed Custom CNN model for helmet classification (Trainable parameters for VGG-16 and InceptionV3 corresponds to that of fine-tuned layers only)

| Models | A % | Prediction time (in milli seconds) | Trainable parameters (in millions) |
|---|---|---|---|
| Custom CNN | **99.62** | 2 | **0.16** |
| Logistic classifier | 96.98 | **0.9** | ∼ |
| VGG-16 | 97.6 | 230 | 6.42 |
| InceptionV3 | 89.42 | 90.2 | 13.12 |

Bold indicates top values in each category

### 5.3.2 Convolutional neural network

The final augmented dataset consists of a total of 3000 images of helmets and non-helmets. When divided into 10 folds, each fold consists of 300 images. CNN based helmet classification system shows an accuracy of 99.62% for 10 fold cross-validation of data. The evaluation metrics for Custom CNN is provided in Table 6. As shown in the table, the Custom CNN outperforms the hand-crafted features in terms of all evaluation metrics Figs. 14a and b shows that the model converges faster providing a high accuracy (99.62%) and low loss within a few epochs.

Here, we have also compared the performance of two standard CNN classification models: VGG-16 and InceptionV3 with the proposed Custom CNN and the Logistic classifier trained on handcrafted features in Table 7. By employing transfer learning we fine-tuned only the top four layers of both VGG-16 and InceptionV3. All the other layers have pre-trained weights from ImageNet. From Table 7, Custom CNN outperforms all other classifiers in terms of accuracy. Both VGG-16 and InceptionV3 had lower test accuracy due to overfitting. In terms of speed, both the standard CNNs are computationally expensive. The hand-crafted features have a slight advantage over Custom CNN when compared in terms of speed. They can be used in embedded applications in which less computational power and real-time detection is required.

In short, the fully automated system proposed is able to perform motorcycle segmentation and helmet identification at the rate of 27fps from surveillance videos.

## 6 Conclusion

In this paper, we have developed a fully automated system that works in real-time, to detect motorcycle riders without a helmet from surveillance videos. Our system was tested on a custom dataset and evaluation metrics are computed using 10 fold cross-validation. The system can detect motorcycles with an accuracy of 99.5% with sequential minimum optimization (SMO) classifier. For helmet detection, experimental results show 96.98% accuracy with the Logistic classifier and an accuracy of 99.62% with the Custom CNN classifier. The small prediction time and comparatively good accuracy indicate that for embedded applications, hand-crafted features is a better choice than Custom CNN. But Custom CNN has an upper hand in terms of accuracy and other evaluation parameters. An extension to this work would be to develop a fully automatic penalizing system for not wearing a helmet, after incorporating an accurate vehicle license plate recognition stage.

## References


1. Bouwmans T (2012) Background subtraction for visual surveillance: a fuzzy approach. Handb Soft Comput Video Surveill 5:103–138
2. Bouwmans T (2014) Traditional and recent approaches in background modeling for foreground detection: an overview. Comput Sci Rev 11:31–66
3. Chen X, He F, Yu H (2019) A matting method based on full feature coverage. Multimed Tools Appl 78(9):11,173–11,201
4. Chiranjeevi P, Sengupta S (2017) Interval-valued model level fuzzy aggregation-based background subtraction. IEEE Trans Cybern 47(9):2544–2555
5. Chiu CC, Ku MY, Chen HT (2007) Motorcycle detection and tracking system with occlusion segmentation. In: 2007. WIAMIS'07. Eighth international workshop on Image analysis for multimedia interactive services. IEEE, pp 32–32
6. Chiverton J (2012) Helmet presence classification with motorcycle detection and tracking. IET Intell Transp Syst 6(3):259–269
7. Cuevas C, Martinez R, Berjón D, Garcia N (2017) Detection of stationary foreground objects using multiple nonparametric background-foreground models on a finite state machine. IEEE Trans Image Process 26(3):1127–1142
8. Dalal N, Triggs B (2005) Histograms of oriented gradients for human detection. In: 2005. CVPR 2005. IEEE computer society conference on Computer vision and pattern recognition. IEEE, vol 1, pp 886–893
9. Di Stefano L, Mattoccia S, Mola M (2004) A change-detection algorithm enabling intelligent background maintenance. In: International conference image analysis and recognition. Springer, pp 437–445
10. Dillencourt MB, Samet H, Tamminen M (1992) A general approach to connected-component labeling for arbitrary image representations. J ACM (JACM) 39(2):253–280
11. e Silva RR, Aires K, de MS Veras R (2017) Detection of helmets on motorcyclists. Multimedia Tools and Applications 77:1–25
12. Fisher R, Perkins S, Walker A, Wolfart E (2003) Connected component labeling. website: http://homepages.inf.ed.ac.uk/rbf/HIPR2/label.htm
13. Gupte S, Masoud O, Martin RF, Papanikolopoulos NP (2002) Detection and classification of vehicles. IEEE Trans Intell Transp Syst 3(1):37–47
14. Haralick RM, Shanmugam K et al (1973) Textural features for image classification. IEEE Transactions on systems, man, and cybernetics SMC-3(6):610–621
15. Huang SC (2011) An advanced motion detection algorithm with video quality analysis for video surveillance systems. IEEE Trans Circ Syst Video Technol 21(1):1–14
16. https://www.techsciresearch.com/report/global-two-wheeler-market/1416.html/. Accessed: 2010-09-30
17. Kim K, Chalidabhongse TH, Harwood D, Davis L (2004) Background modeling and subtraction by codebook construction. In: 2004 international conference on Image processing (ICIP). IEEE, vol 5, pp 3061–3064
18. Krishnapuram B, Carin L, Figueiredo MA, Hartemink AJ (2005) Sparse multinomial logistic regression: Fast algorithms and generalization bounds. IEEE Trans Pattern Anal Mach Intell 27(6):957–968
19. le Cessie S, van Houwelingen J (1992) Ridge estimators in logistic regression. Appl Stat 41(1):191–201
20. Leelasantitham A, Wongseree W (2008) Detection and classification of moving thai vehicles based on traffic engineering knowledge. In: 2008. ITST 2008. 8th international conference on ITS Telecommunications. IEEE, pp 439–442
21. Li X, Ng MK, Yuan X (2015) Median filtering-based methods for static background extraction from surveillance video. Numer Linear Algebra Appl 22(5):845–865
22. McCarthy M, Walter L, Hutchins R, Tong R, Keigan M (2007) Comparative analysis of motorcycle accident data from ots and maids. Published Project Report PPR 168
23. Mukhtar A, Tang TB (2015) Vision based motorcycle detection using hog features. In: 2015 IEEE international conference on Signal and image processing applications (ICSIPA). IEEE, pp 452–456
24. Ojala T, Pietikäinen M, Harwood D (1996) A comparative study of texture measures with classification based on featured distributions. Pattern Recogn 29(1):51–59
25. Organization WH (2018) Global status report on road safety 2018: Summary. Tech. rep., World Health Organization
26. Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, Blondel M, Prettenhofer P, Weiss R, Dubourg V, et al. (2011) Scikit-learn: Machine learning in python. J Mach Learn Res 12(Oct):2825–2830
27. Ramirez-Quintana JA, Chacon-Murguia MI (2015) Self-adaptive som-cnn neural system for dynamic object detection in normal and complex scenarios. Pattern Recogn 48(4):1137–1149

28. Schölkopf B, Burges CJ, Smola AJ et al (1999) Advances in kernel methods: support vector learning. MIT Press, Cambridge
29. Silva R, Aires K, Veras R, Santos T, Lima K, Soares A (2013) Automatic motorcycle detection on public roads. CLEI Electron J 16(3):4–4
30. Stauffer C, Grimson WEL (1999) Adaptive background mixture models for real-time tracking. In: 1999. IEEE computer society conference on Computer vision and pattern recognition. IEEE, vol 2, pp 246–252
31. Teknomo K, Fernandez P (2015) Background image generation using boolean operations. arXiv:1510.00889
32. Vishnu C, Singh D, Mohan CK, Babu S (2017) Detection of motorcyclists without helmet in videos using convolutional neural network. In: 2017 international joint conference on Neural networks (IJCNN). IEEE, pp 3036–3041
33. Waranusast R, Bundon N, Timtong V, Tangnoi C, Pattanathaburt P (2013) Machine vision techniques for motorcycle safety helmet detection. In: 2013 28th international conference of Image and vision computing New Zealand (IVCNZ). IEEE, pp 35–40
34. Wu H, Zhao J (2018) An intelligent vision-based approach for helmet identification for work safety. Comput Ind 100:267–277
35. Yamamoto A, Iwai Y (2009) Real-time object detection with adaptive background model and margined sign correlation. In: Asian conference on computer vision. Springer, pp 65–74
36. Yu H, He F, Pan Y (2018) A novel region-based active contour model via local patch similarity measure for image segmentation. Multimed Tools Appl 77(18):24,097–24,119
37. Yu H, He F, Pan Y (2019) A novel segmentation model for medical images with intensity inhomogeneity based on adaptive perturbation. Multimed Tools Appl 78(9):11,779–11,798
38. Zhang S, He F, Ren W, Yao J (2018) Joint learning of image detail and transmission map for single image dehazing. Vis Comput 35:1–12
39. Zhiwei H, Jilin L, Peihong L (2004) New method of background update for video-based vehicle detection. In: 2004. Proceedings. the 7th international IEEE conference on Intelligent transportation systems. IEEE, pp 580–584
40. Zivkovic Z, Van Der Heijden F (2006) Efficient adaptive density estimation per image pixel for the task of background subtraction. Pattern Recogn Lett 27(7):773–780



**Linu Shine** received her B.Tech. degree in electronics and communication from Cochin University of Science and Technology, Kerala, India, in 2002 and M.Tech degree in Electronic Design Technology from Indian Institute of Science Bangalore, India, in 2012 . She is currently as Assistant Professor in College of Engineering Trivandrum, Kerala, India, and also pursuing Ph.D. degree in electronics and communication at Computer Vision Lab, at same college.

Her research interests include computer vision and machine learning and its application in the field of intelligent transportation.

**Jiji C. V.** received his B. Tech in Electronics and Communication from T K M College of Engineering (University of Kerala) in 1988; M. Tech. in Communication Engineering from Indian Institute of Technology, Mumbai in 1997; PhD from the Department of Electrical Engineering, Indian Institute of Technology, Mumbai in 2007.

He is currently a Professor with Dept. of Electronics and Communication, College of Engineering Trivandrum, Kerala, India. He teaches Random Processes and Applications, Digital Filter Design and Applications, Advanced Digital Signal Processing, Multirate Systems and Wavelets, Digital Image Processing, Estimation and Detection Theory. His research areas are computer vision, image processing, computational photography, signal processing.