Real-Time Detection of Motorcyclist without Helmet using Cascade of CNNs on Edge-device

Dinesh Singh[1] *Member, IEEE*, C. Vishnu[2] and C. Krishna Mohan[2] *Member, IEEE*

***Abstract*— The real-time detection of traffic rule violators in a city-wide surveillance network is a highly desirable but challenging task because it needs to perform computationally complex analytics on the live video streams from large number of cameras, simultaneously. In this paper, we propose an efficient framework using edge computing to deploy a system for automatic detection of bike-riders without helmet. First, we propose a novel robust and compact method for the detection of the motorcyclists without helmet using convolutional neural networks (CNNs). Then, we scale it for the real-time performance on an edge-device by dropping redundant filters and quantizing the model weights. To reduce the network latency, we place the detector module on edge-devices in the cameras. The edge-nodes send their detected alerts to a central alert database where the end users access these alerts through a web interface. To evaluate the proposed method, we collected two datasets of real traffic videos, namely, *IITH_Helmet_1* which contains sparse traffic and *IITH_Helmet_2* which contains dense traffic. The experimental results show that our method achieves a high detection accuracy of $\approx 95\%$ while maintaining the real-time processing speed of $\approx 22fps$ on Nvidia-TX1.**

I. INTRODUCTION

In recent years, the computer vision techniques have become an important component for the automatic traffic management using video surveillance to obviate traffic congestion, advancement of transportation safety, and improvement of traffic flow [1]–[4]. An intelligent transportation system is the integration of various advanced technologies such as intelligent computing, network communications, visual representation, visual-based analysis, efficient sensor electronics, etc [5]–[7]. Since, motorcycles are an affordable and daily mode of transport, there has been a rapid increase in motorcycle casualties due to the fact that most of the motorcyclists do not wear the helmet which makes it an ever-present danger every day to travel by motorcycle [8]–[10]. In the last couple of years alone, most of the deaths in accidents are due to damage in the head [11]. Because of this fact, wearing a helmet is mandatory according to the traffic rules, violation of which attracts hefty fines in India. Inspite of this fact, a large number of motorcyclists do not follow the traffic rules. The manual strategies to catch these violators have several drawbacks such as interrupted traffic flow, unpleasant weather conditions for police personnel, etc. Existing video surveillance based methods are passive and require significant human assistance. In general, such systems are infeasible due to the involvement of humans, whose efficiency decrease over long duration. Automation of this process is highly desirable for reliable and robust monitoring of these violations as well as reduce the amount of human resources needed significantly. Presently, all major cities across the world already deployed extensive video surveillance network at public places to keep a vigil on a wide variety of threats. Thus the solution for detecting violators using the existing infrastructure is also cost-effective.

To date, several researchers [8]–[10], [12]–[16] have tried to tackle the problem of detection of motorcyclists without helmet by using different methods in computer vision. But they have not been able to accurately identify motorcyclists without helmets under challenging conditions such as occlusion, illumination, poor quality of video, varying weather conditions, etc. Some of the reasons for the poor performance of existing approaches are: (i) the use of not so efficient handcrafted features for object classification, (ii) the consideration of irrelevant objects for the detection of motorcyclists without helmet, and (iii) most of the existing methods are computationally complex and thus not suitable to be used in real-time. The performance of a recognition system depends, to a large extent, on whether it can extract and utilize relevant information. However, extraction of relevant information for the detection of such violators is non-trivial due to a variety of reasons such as scale variations, viewpoint changes, camera quality, change in illumination, etc. Thus it becomes crucial to design efficient representation that can deal with these challenges while preserving categorical information of violator and non-violator classes. Deep networks have gained much attention with state-of-the-art results in complex recognition tasks such as image classification [17], accident detection [18], object recognition [19], [20], tracking [21], [22], detection, and segmentation [23], [24] due to their ability to learn discriminatory features directly from raw data without resorting to manual tweaking. The challenges in the existing recognition systems and the recent advances in deep learning motivate us to design an efficient framework for the detection of motorcyclists driving without helmet in real-time that can handle wide variations in viewpoints and environmental conditions. The main contributions of this paper are as follows:

- Design of a robust and reliable method for detection of moving motorcyclists in real-time using convolutional neural network (CNN) under the various challenging conditions, such as viewpoint, illumination effects, weather change, etc.
- Acceleration of the CNN model used to detect motor-

[1]Dinesh Singh is with RIKEN Center for Advanced Intelligence Project, Japan dinesh.singh@riken.jp. This work was done while D. Singh was with IIT Hyderabad, India

[2]C. Vishnu and C. Krishna Mohan are with IIT Hyderabad, India ckm@iith.ac.in

cyclists in real-time on the limited-resource embedded device.
- Develop light weight but powerful CNN model for efficient classification of head (i.e. violator) and helmet, with very limited set of training samples.
- Use of an Edge-computing framework to overcome the communication overhead and network latency.

II. Related Work

To date, many researchers have proposed several methods [8], [10], [12]–[16] to solve the problem of real-time helmet detection in traffic. Chiu *et al.* [12] proposed a system for the detection of motorcyclists in surveillance videos. This system segments the moving object and then tracks motorcycles and their head regions using a probability-based algorithm to handle the occlusion problem, but it can not handle small variations due to noise and illumination effects. Also, it uses Canny edge detection with a search window of a certain size to detect head. Chiverton *et al.* [13] used edge histogram based features intending to detect motorcyclists. The strength of this method is that it performs well even if there is low-intensity light or low illumination in videos due to the use of edge histograms near the head instead of detecting the features in the head region. Since the edge histograms used circular Hough transforms to compare and classify helmets, it leads to a lot of misclassifications among motorcyclists with a helmet because the objects look like helmet can be categorized as a helmet. Also, helmets of different types may not be classified as helmets. Silva *et al.* [14], [16] proposed a system to cope with the misclassification problem in which vehicles are tracked using Kalman filter. A significant advantage of the Kalman tracking system is the ability to continue to track objects even if they are slightly occluded. But when there are more than two or three motorcyclists appear in the same frame, Kalman filter fails because Kalman filter mostly works well for linear state transitions. But to track multiple objects, we need non-linear functions.

Recently, Dahiya *et al.* [8] proposed a system which first uses the Gaussian mixture model to detect moving objects. This model is robust to slight variations in the background. It uses two classifiers in serial, one for the separating motorcyclist from moving objects and another for separating without-helmet riders from the with-helmet riders. However, it uses only hand engineered features such as SIFT [25], HOG [26], LBP [27] along with kernel SVM [28] in both classifications. Their approach was promising as it performs well for the classification of motorcyclists and non-motorcyclists but results in a low performance for the classification of the helmet and non-helmet riders under harsh conditions. Singh *et al.* [9] proposed a visual big data framework which scales the method in [8] to a city scale surveillance network. Experimental results show that the framework can detect a violator in less than 10 milliseconds. The existing methods suffer from several challenges such as (i) occlusion of objects, (ii) illumination effects, (iii) use of ill-posed traditional representation, and (iv) the use of not so efficient methods for the recognition of $motorcyclists$ and classification of $violators$ (riders without helmet) vs. $followers$ (riders with helmet). For efficient detection of motorcyclists without a helmet, we also need to have good feature representation of the motorcyclists to classify them accurately which is not possible using HOG [26] or LBP [27] or SIFT [25] on images with fewer pixels. Above mentioned issues inspired us to propose a novel method entirely based on deep learning, which uses a CNN based object detector for the detection of motorcyclists and a CNN classifier to extract discriminatory features for further detection of motorcycle rider driving without a helmet.

Since the problem of helmetless motorcyclists detection falls into the category of object recognition, Dhaiya *el al.* [8] and Vishnu *et al.* [10] use the Gaussian mixture model (GMM) for object detection which is computationally fast, but it is not able to detect many moving objects in dense traffic scenarios. Several other methods for object detection such as deformable part models (DPM) [29], R-CNN [30], Fast R-CNN [31], and Faster R-CNN [32] are showing much-improved performance concerning detection accuracy but have significant computational overhead. In this work, we use *you only look once* (YOLO) [33] to detect the motorcyclists in the incoming live video frames from a CCTV camera. In YOLO, a single convolutional neural network (CNN) predicts multiple bounding boxes simultaneously, in a given image with their class probabilities. The full network architecture of YOLO consists of 24 convolutional layers and 2 fully connected layers where it formulates the task of object detector as a single regression problem to avoid a complex pipeline and make it incredibly fast on the modern GPUs like *Titan-X*, *P100*, etc. However, it is not suitable for the real-time detection on the limited-resource edge-devices like Nvidia TX1/TX2, etc. which we are targeting for the deployment of such a system. Hence, we design two CNN models, namely, M-Net for the fast and efficient detection of the motorcyclists on the Nvidia-TX1 and H-Net for the efficient classification of the head and helmet.

III. Detection of Helmetless Motorcyclists

The proposed framework for automatic detection of motorcyclists driving without a helmet is given in the block diagram as shown in Fig. 1. The entire framework consists of four steps: (i) detection of motorcycles using a CNN based object detector, (ii) localization of the upper body part of the person riding the motorcycle, (iii) prediction using H-Net a CNN classifier trained for binary classification of head and helmets, and finally (iv) temporal consolidation of the alert to generate more reliable alerts. The details of the methods used in these steps are discussed in the following subsections.

A. M-Net: An efficient CNN model for motorcyclist detection

The first step of the proposed framework is the detection of motorcyclists in the incoming live video stream. This problem falls into the category of object detection. Existing methods for object detection such as YOLO [33] and SSD [34] are showing much improved detection accuracy but

Fig. 1: Block diagram of proposed framework for the detection of motorcyclists without Helmet. [Best viewed in color]

have significant computational overhead. Here, we present M-Net, an efficient CNN based model for the detection of the motorcyclists suitable for the real-time performance on the resource-limited devices. The architecture of the M-Net is derived from tiny-YOLO architecture. Since tiny-YOLO requires a significant computational time and not as powerful as full YOLO, a series of changes are made to improve the accuracy and the speed of the M-Net. As our objective is to detect motorcyclist only, we restrict M-Net to two classes, namely, motorcyclist (i.e. motorcycle with person riding) and others (i.e. cars, person, cycle, motorcycle with a person) which resulted into less parameters in the last softmax layer as well as bounding box regression. To increase the accuracy of this network for motorcyclist detection, we re-trained it from a large number of images from our dataset where the ground trouth is generated from a full YOLO network. We label a motorcyclist when scores for both the classes (i.e. motorcycle and person) are above a threshold. After successful training of the M-Net, we eliminate substantially similar convolutional filters by using k-means clustering on convolutional filters which results into reduced processing time. Let $\mathcal{W}^l = \{\mathbf{w}_1, \mathbf{w}_2, ..., \mathbf{w}_m\}, \mathbf{w}_i \in \mathbb{R}^d$ be the set of weights for original convolutional filters in the l^{th} convolutional layer. During the filter elimination, the goal is to construct a new set of $k < m$ weights $\bar{\mathcal{W}}^l = \{\bar{\mathbf{w}}_1, \bar{\mathbf{w}}_2, ..., \bar{\mathbf{w}}_k\}, \bar{\mathbf{w}}_i \in \mathbb{R}^d$ with a small discrepancy with $\mathcal{W}^l$ i.e for most $w \in \mathcal{W}^l$, there is a representative $\bar{w} \in \bar{\mathcal{W}}^l$ such that the Euclidean distance between $\mathbf{w}$ and $\bar{\mathbf{w}}$ is small. The average discrepancy loss of $\bar{\mathcal{W}}^l$ with respect to $\mathcal{W}^l$ is defined as

$$\mathcal{L}(\bar{\mathcal{W}}^l, \mathcal{W}^l) = \mathbb{E}\left[\min_{1\le j\le k} \|W - \bar{\mathbf{w}}_j\|^2\right] = \frac{1}{m}\sum_{i=1}^{m} \min_{1\le j\le k} \|\mathbf{w}_i - \bar{\mathbf{w}}_j\|^2, \quad (1)$$

where $\|\cdot\|$ denotes Euclidean norm and the expectation is over W drawn uniformly at random from $\mathcal{W}^l$.

In order to further accelerate the convolutional operations, we ternarize the weights in the convolutional layers of the finally trained network to $\omega \in \{-1, 0, +1\}$ as follows:

$$\omega = \begin{cases} +1, & if\ \bar{w}_i > T_{max}, \\ -1, & if\ \bar{w}_i < T_{min}, \\ 0, & otherwise. \end{cases} \quad (2)$$

The parameters T_{max} and T_{min} along with a scaling factor μ are determined using genetic algorithm by minimizing the following objective function on a validation set $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_1, \cdots, \mathbf{x}_n\}$:

$$\min_{\boldsymbol{\omega}} \frac{1}{n}\sum_{j=1}^{n} \|\bar{\mathbf{w}}^T\mathbf{x}_j - \mu\boldsymbol{\omega}^T\mathbf{x}_j\|^2. \quad (3)$$

The ternarization of the weights accelerates the detection because it reduces the number of multiplication operations in convolutional layers. Also, the ternarization of the weights results into the loss of accuracy in comparison to the original network. However, we recover this loss by again fine-tuning the last fully connected layer and the softmax layer.

In this way, we get the bounding boxes of all the bikes present in the current frame. It can also detect multiple motorcyclists in a video frame as shown in Figure 2.

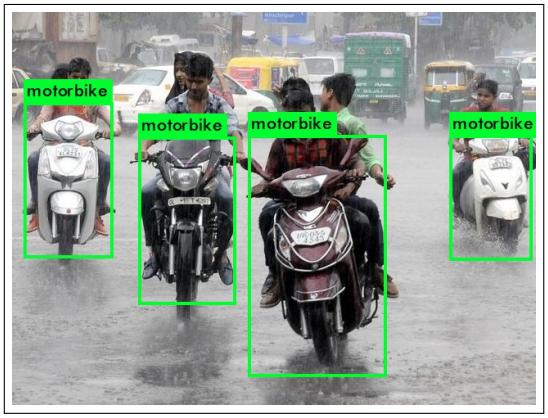


Fig. 2: The multiple motorcycle detection in dense traffic.

B. Localization of the Rider's Head

The output of the previous step is a set of bounding boxes *m-bbox* for the motorcyclists. For each detection of a motorcyclist, we again search for its rider (i.e., a person) and if detected then extend its height slightly up words to guarantee the complete coverage of the rider's head. The upper one-third part of this extended bounding box is the final location for the rider's head and the output will be a bounding box $bbox(x, y, w, h)$, where (x, y) are the coordinates of the center of the bounding box and (w, h) are the width and the height of the bounding box, where all values are the ratio with respect to the size of full input image.

C. Classification of Head and Helmet using CNN

The output of the previous step is a $bbox(x, y, w, h)$ locating the region of image consisting of the motorcyclist's upper body part as shown in Figure 3. The next task

is to ensure whether the detected motorcyclist is wearing a helmet or not. If the resulted faces are clearly visible then one can easily detect violators by applying methods such as Viola–Jones [35], HoG [26], SIFT [25], LBP [27], DeepFace [36], VGG_face CNN descriptor [37], Deep Head Pose [38], etc. to classify them into face vs. non-face categories. However, in spite of a good resolution camera, the faces are not clearly visible due to the size of their appearance while covering the entire road. This makes the task of detection of violators non-trivial and thus all the techniques as mentioned earlier fail to address this task. The deep learning model like the convolutional neural network can be applied to extract hidden information relevant for discriminating the heads from the helmets. As mentioned earlier, most relevant pre-trained deep model VGG16 is also not able to solve this task due to unclear face appearance and tiny images of the head/helmet. The other models have a large number of parameters and thus unable to give a real-time performance and require a significant number of training samples. Also, the two large CNNs in Vishnu *et al.* [10] increased their prediction time and may lead to overfitting as they are trained from the scratch. To address above mentioned issues, we design a simple, fast, and robust convolutional neural network classifier which can be trained from relatively small number of training examples. As we know that the first few layers of the large CNNs extract generic features and can be used for learning variety of tasks thereafter. Thus, we leverage this fact and design a tiny network on top of the activation filters received from the output of an intermediate layer of the detector network. Fig. 4 shows the architecture of the proposed CNN model called H-Net used for the classification of motorcyclist with helmet and without a helmet.

The input to H-Net is a tensor of size $28 \times 28 \times 192$ which is croped from the output of the second covolutional layer of the detector network. Let $bbox(x, y, w, h)$ be the bounding box locating the head of a motorcyclist as computed in the previous step and $O_l(\mathcal{W} \times \mathcal{H} \times \mathcal{C})$ is the output of the l^{th} convolutional layer, where $\mathcal{W}, \mathcal{H}$, and $\mathcal{C}$ are the width, height, and number of channels, respectively. Then the input I to the

Fig. 3: The sample images of the located motorcycle riders with and without a Helmet of various shape and viewpoints.

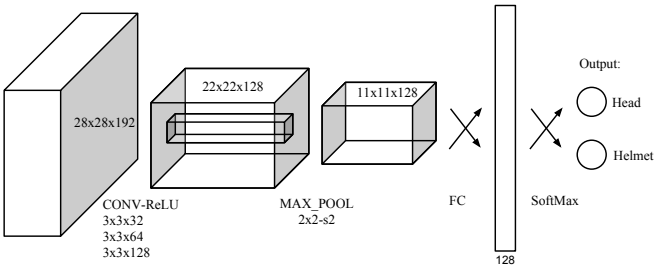


Fig. 4: Architecture of the proposed network H-Net for head vs. helmet classification. (s→stride)

H-Net is determined as

$$I = O_l\Big(\mathcal{W} \cdot (x - \frac{w}{2}) : \mathcal{W} \cdot (x + \frac{w}{2}), \mathcal{H} \cdot (y - \frac{h}{2}) : \mathcal{H} \cdot (y + \frac{h}{2}), 1 : \mathcal{C}\Big),$$
$$I = resize(I, [28 \times 28 \times \mathcal{C}]). \quad (4)$$

H-Net consists of three convolutional layers with rectified linear unit (ReLU) as activation, followed by one 2×2 max-pooling layer, and one fully connected layer of 128 neurons. Finally, the network uses a softmax layer with two classes. The resulting activation of each layer is shown in Fig. 5. It can be observed from the figure that the model gives high activation values corresponds to the helmet while low activation values corresponds to the head. Also, there is an increase in the intensities of the activation values for the deeper layers.

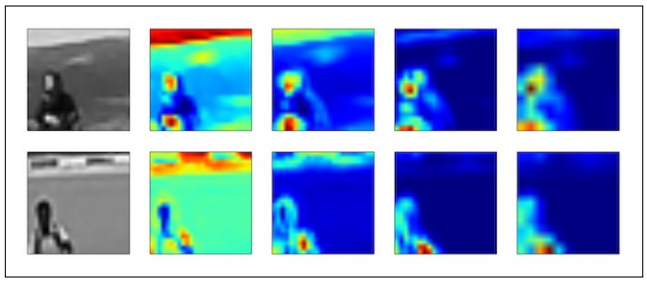
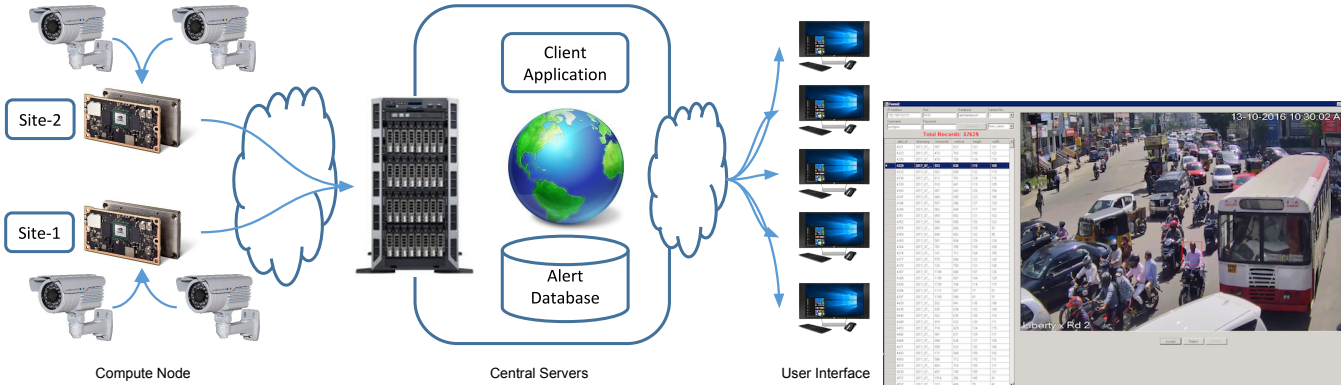


Fig. 5: The activations produced by the various layers of the proposed CNN classifier after training. The top row shows the activation for a rider with a helmet while the bottom row indicates the activation for a rider without a helmet. [Best viewed in color]

D. Temporal Consolidation of the Alerts:

From the previous phase, we obtain decision on each individual frame whether it contains the violator(s) (motorcyclists without helmet) or not. As the proposed approach is applied on continuous video stream, there are multiple alarms raised for a single violator in multiple frames. However, the correlation between continuous frames is completely neglected in the detector module. Thus, we consolidated the alerts generated from the detector module over the continuous frames in order to generate less number of alerts with increased reliability i.e. reducing the number of false



(a) Framework (b) Interface

Fig. 6: The architecture and user interface of the proposed edge computing-based framework for the detection of violators.

alerts. Let y_i be the label for i^{th} frame which is either $+1$ (i.e. at-least one violator is detected) or 0 (i.e. no violator detected). Then for the n frames, the violation alarm is triggered as

$$Alarm = \begin{cases} True, & if \frac{1}{n}\sum_{i=1}^{n} y_i > T_f, \\ False, & otherwise, \end{cases} \quad (5)$$

where the threshold T_f is determined empirically. In our case, the value of $T_f = 0.6$ and $n = 5$ are used.

IV. Edge Computing Framework for Traffic Monitoring

In a city-scale surveillance scenario, the central computing infrastructures are unable to perform in real-time due to large network delay from video sensor to the central computing server. For accurate and real-time detection of traffic violations and incidents in such a scenario, we propose an efficient framework using edge computing for deploying large-scale visual computing applications which reduces the latency and the communication overhead in the camera network as shown in Fig. 6a. The entire system architecture consists of three parts, namely, compute node, central servers, and client interface. The compute nodes are the embedded devices placed in the close vicinity of the cameras at the sites. The detector modules are placed on the computing nodes where they process the live video footage from the cameras in real-time without any delay and send their detected alerts to a central alert database at the central server. As the central servers have better computational resources, we further evaluate the received alerts using a more reliable model. For example, in helmetless motorcyclists detection, the tiny-YOLO is used in the detector module at compute nodes with a relatively low confidence score for the detection of violators and full YOLO with a high confidence score is used at the server for re-evaluation. This re-evaluation helps in the reduction of false alarms. The end users can access the detected alerts from the central alert database through a user interface as shown in Fig. 6b.

V. Experiments

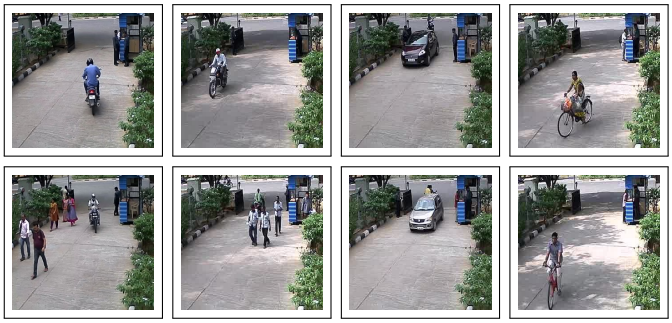
The experiments are conducted on a machine running Ubuntu 16.04 Xenial Xerus having specifications Intel(R) Xeon(R) CPU E5-2697 v2 @ 2.70GHz×48 processor, 128GB RAM with NVIDIA Corporation GK110GL [Tesla K40c]×2 GPUs as central server and Nvidia Jetson TX1 as the compute node. The programs for helmet detection are written in C & CUDA with the help of the various libraries such as opencv for image processing and vision tasks. For training deep models, we use darknet [39], an open source neural network framework written in C and CUDA. Darknet is fast, easy to install, and supports CPU and GPU computation. The original camera images has resolution 1920×1080 pixels.

A. Datasets Used

The performance of the proposed approach is evaluated on two video datasets *IITH_Helmet_1* and *IITH_Helmet_2* containing sparse traffic and dense traffic, respectively. Both the datasets are collected by us because there is no public dataset available till the date to the best of our knowledge. The datasets are made public for future use by the research community at *https://www.iith.ac.in/vigil/resources.html*. The brief descriptions for both the datasets are as follows.

IITH_Helmet_1: This dataset is collected from the surveillance network at Indian Institute of Technology Hyderabad, India (IITH) campus. It is a two-hour surveillance video data collected at 30 frames per second. Fig. 7a presents sample frames from the collected dataset. We have used the first one hour of the video for training and the remaining for testing purpose. The training video contains 42 motorcycles, 13 cars, and 40 humans. Whereas, the testing video contains 63 motorcycles, 25 cars, and 66 humans.

IITH_Helmet_2: This second dataset is acquired from the CCTV surveillance network of Hyderabad city in India. It is a 1.5 hour video collected at 25 frames per second. The sample frames from this dataset are presented in Fig. 7b. The first half an hour of the video is used for training the model and the remaining for testing purpose. The training video



(a) *IITH_Helmet_1*

(b) *IITH_Helmet_2*

Fig. 7: Sample frames from datasets showing the various difficulties like congestion, direction of motion, variety of violations, occlusion, etc.

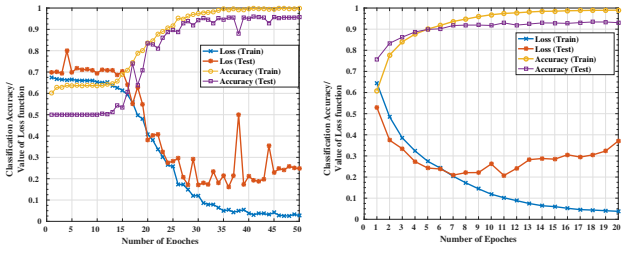
contains 1261 motorcyclists and 4960 non-motorcyclists. Whereas, the testing video contains 2312 motorcycles, and 9112 non-motorcyclists.

B. Evaluation of Classification Accuracy

As explained in the section III-A, the M-Net is more robust and reliable for accurate detection of the motorcyclists irrespective of variations in their appearance and environmental condition as shown in Fig. 2. Also, it can detect multiple motorcyclists in a single frame accurately even in dense traffic. Additionally, these resulted detections correspond to a higher confidence score of 0.75. Reducing the threshold on confidence score would lead to more detections but may increase the false detection. The proposed model successfully detects all the motorcyclists in the case of sparse traffic as the case in *IITH_Helmet_1* dataset while using a low threshold of 0.3 on the confidence score without a single false detection. While the performance of motorcycle detection using GMM as used in [8], [10] is $\approx 98\%$ on *IITH_Helmet_1* dataset. Similarly, on the dense traffic also like the case of *IITH_Helmet_2* dataset, it did not raise any false alarms even on a very low threshold of 0.2. However, due to high occlusion of various vehicles as well as the size of their appearance, it missed certain motorcyclist. This problem occurs since the videos collected in the dataset are unconstrained and the cameras are not placed explicitly for such a task and thus can be solved easily by putting the camera in an appropriate place.

However, the classification of the head vs. helmet is challenging as shown in Fig. 10. Fig. 10a and Fig. 10c depict the 2D visualization of the spread of the extracted train and test samples from *IITH_Helmet_1* dataset, respectively. Here, the pattern of the two classes, namely, head and helmet are overlapping each other showing that the patterns share high inter-class similarity along with intra-class dissimilarities which makes the classification task more complex. Thus, the performance of the previously used methods $GMM + HoG$ [8] achieved only 93.80% on *IITH_Helmet_1* dataset and while score a low performance of 57.78% on *IITH_Helmet_2* dataset as shown in Table I. However, the performance is improved slightly using CNN [10] which achieved 98.63% on *IITH_Helmet_1* dataset and 87.11% on *IITH_Helmet_2* dataset. However, the proposed deep CNN model as explained in section III-C addresses this problem more precisely in comparison to previously used methods. Here, we present an extensive evaluation of the proposed approach. In literature, the performance of the deep models is justified from three observations, namely, 1) the value of the loss function for train and validation sets during training and their convergence, 2) the visualization of the feature maps of intermediate layers, 3) the scatter plots of the final representations.

The training behaviors on both *IITH_Helmet_1* and *IITH_Helmet_2* datasets are shown in Fig. 8a and Fig. 8b, respectively. Both the figures show the epoch-wise change in the value of the loss function and the corresponding classification accuracy of training and test sets. The loss function used is categorical cross-entropy and the optimizer used is adadelta. Due to the limited number of samples for the training in the *IITH_Helmet_1* dataset, there are small changes in the first few epochs though we applied regularization and dropout to address this issue. After 15 epochs, it rapidly optimizes the loss function, and the resulted training reached close to zero, while corresponding classification accuracy almost touched to 1 in less than 50 epochs.



(a) *IITH_Helmet_1* (b) *IITH_Helmet_2*

Fig. 8: Classification accuracy and values of loss function at each epoch for train and test sets while training H-Net.

It can be observed that the test set also follows the structure learned by the model for the discrimination of the two classes from training samples which also presented in the test set. A classification accuracy of 98.70% is achieved on test set. However, after 28 epochs, it starts deviating from this behavior which we considered as the convergence point. The lowest value of loss function and highest accuracy is observed at 36^{th} epoch. While the training on *IITH_Helmet_2* dataset starts optimizing the loss function from first epoch

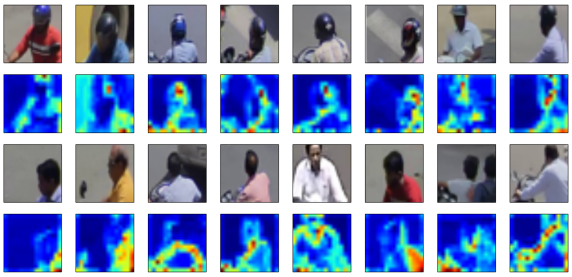


Fig. 9: Sample images and their respective activation maps for the two classes, namely, helmet and head.

onwards and quickly converged within 7 epochs due to the availability of relatively more number of samples than *IITH_Helmet_1* dataset. In spite of substantial variation in and across the two classes, the model is able to learn the discriminate representation.

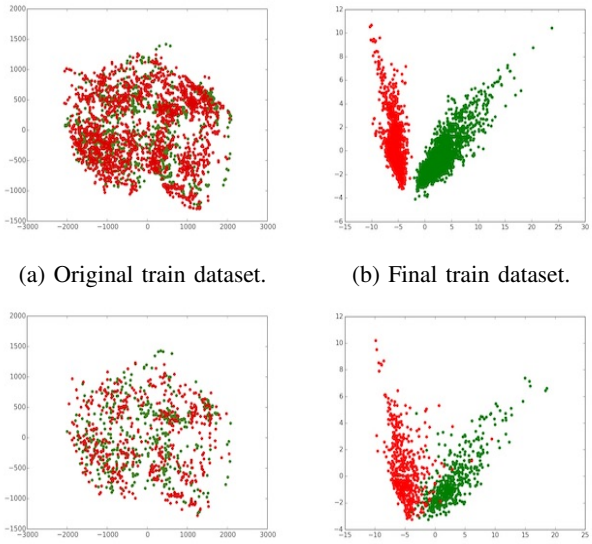
Additionally, the filters for various samples of both the classes as shown in the Fig. 9 for *IITH_Helmet_1* dataset reveal the success of the proposed model. The learned hidden deep structures are of discriminating in nature as well as self-explanatory. The feature maps across the various samples of class helmet contain a consistently high activation values for the pixels correspond to a helmet in the input images. However, for the another class (i.e. head), this structure is clearly different. As can be clearly shown from the two figures that instead of the head region, it produces high activation for the region corresponds to the shoulder.

The final observation is the transformation in the distribution of the train and test datasets from the first input image to the final deep representation (i.e. the output of the last fully-connected layer of the trained CNN model). Fig. 10 shows the scatter plots of the *IITH_Helmet_1* dataset for top two principal components. The sub-figures (a) & (c) show the original (input raw pixels) distributions for train and test sets, respectively. Similarly, sub-figures (b) & (d) show the distributions of the final deep representation for train and test sets, respectively. It can be observed from the scatter plots that the proposed model learns the distribution of the two classes and transforms them into space where they are easy to classify, and the learned weight of the model also follows the same kind of distribution. Thus it can be concluded that the model transforms a complicated and hard to classify distribution of the two classes into a distribution where the points from two categories are easy to identify. This transformation results in a high classification accuracy in comparisons to other approaches.

The comparison with the various recently proposed approaches on both the datasets are presented in the Table I. The proposed system outperforms the existing methods with a margin of $4.90\%, 0.07\%$ on *IITH_Helmet_1* and $36.38, 6.95\%$ on *IITH_Helmet_2* datasets, respectively.

C. Evaluation of Space & Time Requirement

The proposed approach for the detection of the motorcyclist without a helmet can process a real-time stream at



(a) Original train dataset. (b) Final train dataset.

(c) Original test dataset. (d) Final test dataset.

Fig. 10: Scatter plots of *IITH_Helmet_1* dataset showing distributions of the two classes, namely, helmet (green dots) and head (red dots) in train and test datasets.

TABLE I: Performance (%) of the classification of 'Helmet' vs. 'Without helmet' using CNN

Method	IITH_Helmet_1	IITH_Helmet_2
GMM+HOG+SVM [8]	93.80	57.78
GMM+CNN+CNN [10]	98.63	87.11
Proposed Approach:	**98.70**	**94.16**

a speed of 22 fps on Nvidia-TX1 and takes a total of $943MB$ space in device memory. The space requirement on the device for weights of detector CNN model is $889MB$ and an additional memory requirement of $54MB$ for input, intermediate, and output variables. Similarly, the space requirement on the device for weights of H-Net classifier model is $8MB$ and an additional memory requirement of $1MB$ for input, intermediate, and output variables. Thus the proposed framework is highly scalable for processing multiple real-time cameras streams. Table II shows the space and processing speed when processing multiple streams on a single GPU card.

TABLE II: Space & Time requirements of the proposed models.

Stream	Model Size		Processing Speed		
	Detector *(MB)*	Classifier *(MB)*	Detector *(ms)*	Classifier *(ms)*	Joint *(fps)*
1	943	9	40	5	22
2	997	10	59	7	15
3	1051	11	81	10	11
4	1105	12	148	18	6

VI. Conclusion

The proposed framework for real-time detection of motorcyclists driving without helmets is able to perform in

diverse surveillance conditions. The proposed framework recognizes violators accurately as compared to the existing methods. Also, there is a significant reduction in the number of false alarms because of the use of cascaded CNNs. Even with such a high detection rate, our approach processes incoming video stream in real-time because of the elimination of the redundant filters from the convolutional layers and ternarization of the weights which ultimately reduces the number of floating point operation needed. The placement of the detector modules in the vicinity of the capturing devices in a edge-computing framework reduces the communication overhead and solves the issue of network latency. The experimental results show the efficacy of the proposed approach.

VII. Acknowledgment

This work has been conducted as the part of SATREPS project entitled on " Smart Cities For Emerging Countries Based on Sensing, Network, and Big Data Analysis of Multimodal Regional Transport System" funded by JST and JICA.

References

[1] Q. Wang, J. Wan, and Y. Yuan, "Locality constraint distance metric learning for traffic congestion detection," *Pattern Recognition*, 2017.

[2] Y.-B. Lin and C.-P. Young, "High-precision bicycle detection on single side-view image based on the geometric relationship," *Pattern Recognition*, vol. 63, pp. 334 – 354, 2017.

[3] Z. An, Z. Shi, Y. Wu, and C. Zhang, "A novel unsupervised approach to discovering regions of interest in traffic images," *Pattern Recognition*, vol. 48, no. 8, pp. 2581 – 2591, 2015.

[4] X. Li, M. Ye, Y. Liu, F. Zhang, D. Liu, and S. Tang, "Accurate object detection using memory-based models in surveillance scenes," *Pattern Recognition*, vol. 67, pp. 73 – 84, 2017.

[5] B. H. Chen and S. C. Huang, "An advanced moving object detection algorithm for automatic traffic monitoring in real-world limited bandwidth networks," *IEEE Trans. Multimedia*, vol. 16, no. 3, pp. 837–847, April 2014.

[6] N. Li, J. J. Jain, and C. Busso, "Modeling of driver behavior in real world scenarios using multiple noninvasive sensors," *IEEE Trans. Multimedia*, vol. 15, no. 5, pp. 1213–1225, Aug 2013.

[7] S. P. Narote, P. N. Bhujbal, A. S. Narote, and D. M. Dhane, "A review of recent advances in lane detection and departure warning system," *Pattern Recognition*, vol. 73, pp. 216 – 234, 2018.

[8] K. Dahiya, D. Singh, and C. K. Mohan, "Automatic detection of bike-riders without helmet using surveillance videos in real-time," in *IJCNN*, Vancouver, Canada, 24–29 July 2016, pp. 3046–3051.

[9] D. Singh, C. Vishnu, and C. K. Mohan, "Visual big data analytics for traffic monitoring in smart city," in *IEEE ICMLA*, Anaheim, California, 18–20 December 2016.

[10] C. Vishnu, D. Singh, and C. K. Mohan, "Detection of motorcyclists without helmet in videos using convolutional neural network," in *IJCNN*, May 2017.

[11] C. Behera, R. Ravi, L. Sanjeev, and D. T, "A comprehensive study of motorcycle fatalities in south delhi," *J. Indian Academy of Forensic Medicine*, vol. 31, no. 1, pp. 6–10, 2009.

[12] C.-C. Chiu, M.-Y. Ku, and H.-T. Chen, "Motorcycle detection and tracking system with occlusion segmentation," in *Int. Workshop on Image Analysis for Multimedia Interactive Services*, Santorini, Greece, 6–8 June 2007, pp. 32–32.

[13] J. Chiverton, "Helmet presence classification with motorcycle detection and tracking," *IET Intelligent Transport Systems (ITS)*, vol. 6, no. 3, pp. 259–269, 2012.

[14] R. Silva, K. Aires, T. Santos, K. Abdala, R. Veras, and A. Soares, "Automatic detection of motorcyclists without helmet," in *Latin American Computing Conf. (CLEI)*, Puerto Azul, Venezuela, 4–6 October 2013, pp. 1–7.

[15] W. Rattapoom, B. Nannaphat, T. Vasan, T. Chainarong, and P. Pattanawadee, "Machine vision techniques for motorcycle safety helmet detection," in *Int. Conf. Image and Vision Computing New Zealand (IVCNZ)*, Wellington, New Zealand, 27–29 November 2013, pp. 35–40.

[16] R. V. Silva, T. Aires, and V. Rodrigo, "Helmet detection on motorcyclists using image descriptors and classifiers," in *Graphics, Patterns and Images (SIBGRAPI)*, Rio de Janeiro, Brazil, 27–30 August 2014, pp. 141–148.

[17] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems (NIPS)*, Lake Tahoe, Nevada, United States, 3–6 December 2012, pp. 1097–1105.

[18] D. Singh and C. K. Mohan, "Deep spatio-temporal representation for detection of road accidents using stacked autoencoder," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 3, pp. 879–887, 2019.

[19] D. Jeff, J. Yangqing, V. Oriol, H. Judy, Z. Ning, T. Eric, and D. Trevor, "DeCAF: A Deep Convolutional Activation Feature for Generic Visual Recognition," *ICML*, vol. 32, no. 1, pp. 647–655, 2014.

[20] N. Perveen, D. Singh, and C. K. Mohan, "Spontaneous facial expression recognition: A part based approach," in *IEEE ICMLA*, Anaheim, CA, USA, 18–20 Dec 2016, pp. 819–824.

[21] N. Hyeonseob and H. Bohyung, "Learning Multi-Domain Convolutional Neural Networks for Visual Tracking," in *CVPR*, Las Vegas, United States, June 26th - July 1st 2016, pp. 4293–4302.

[22] Z. Kaihua, L. Qingshan, Wu, and Y. Ming-Hsuan, "Robust Visual Tracking via Convolutional Networks without Training," *IEEE Trans. Image Processing*, vol. 25, no. 4, pp. 1779–1792, 2016.

[23] G. Ross, D. Jeff, D. Trevor, and M. Jitendra, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *CVPR*, Colombus, Ohio, 24–27 June 2014, pp. 580–587.

[24] Z. Wang, D. Xiang, S. Hou, and F. Wu, "Background-driven salient object detection," *IEEE Trans. Multimedia*, vol. 19, no. 4, pp. 750–762, April 2017.

[25] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *IJCV*, vol. 60, no. 2, pp. 91–110, 2004.

[26] D. Navneet and B. Triggs, "Histograms of oriented gradients for human detection," in *CVPR*, San Diego, California, 20–26 June 2005, pp. 886–893.

[27] Z. Guo, D. Zhang, and L. Zhang, "A completed modeling of local binary pattern operator for texture classification," *IEEE Trans. Image Processing*, vol. 19, no. 6, pp. 1657–1663, 2010.

[28] C. Cortes and V. Vapnik, "Support vector networks," *Machine Learning*, vol. 20, no. 3, pp. 273–297, 1993.

[29] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," *IEEE TPAMI*, vol. 32, no. 9, pp. 1627–1645, Sept 2010.

[30] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation," in *CVPR*, June 2014, pp. 580–587.

[31] R. Girshick, "Fast R-CNN," in *ICCV*, Dec 2015, pp. 1440–1448.

[32] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," *IEEE Trans. PAMI*, vol. 39, no. 6, pp. 1137–1149, 2017.

[33] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *CVPR*, June 2016, pp. 779–788.

[34] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. E. Reed, C. Fu, and A. C. Berg, "SSD: single shot multibox detector," in *ECCV*, Amsterdam, The Netherlands, October 11-14, 2016 2016, pp. 21–37.

[35] P. Viola and M. J. Jones, "Robust real-time face detection," *Int. J. Comput. Vision*, vol. 57, no. 2, pp. 137–154, 2004.

[36] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, "Deepface: Closing the gap to human-level performance in face verification," in *CVPR*, June 2014, pp. 1701–1708.

[37] O. M. Parkhi, A. Vedaldi, and A. Zisserman, "Deep face recognition," in *BMVC*, 2015.

[38] S. S. Mukherjee and N. M. Robertson, "Deep head pose: Gaze-direction estimation in multimodal video," *IEEE Trans. Multimedia*, vol. 17, no. 11, pp. 2094–2107, Nov 2015.

[39] J. Redmon, "Darknet: Open source neural networks in c," http://pjreddie.com/darknet/, 2013–2016.