# A Review on Deep Learning Based Helmet Detection

Arya K M. and Ajith K K.
Department of Electronics and Communication Engineering,
Government College of Engineering Kannur

***Abstract*— Usage of helmets is one of the most requirements while riding motorcycles. Every individual has to obey the safety protocols set by the government. However, those are not followed properly. So, several smart techniques must be implemented. Construction industries and power sub-stations suffer lots of fatalities because of the carelessness of the workers. So, a surveillance system is needed that can detect helmets and prevent the number of accidents. A computer vision-based automatic video monitoring system will help to develop an automatic helmet detection system. This paper is a review of detecting helmet with the help of different types of deep learning approaches. CNN is an artificial neural network that expects images inputs and is designed to work on images mostly to handle computer vision problems. RCNN, Fast RCNN, Faster RCNN, YOLO, and so on are the famous examples of a deep learning algorithm. These act in distinct ways in different fields such as network architecture, training strategy, and optimization.**

***Index Terms*—CNN, Deep learning, Faster RCNN, Helmet Detection, Object Detection, YOLO v2.**

## I. INTRODUCTION

In the past years, road accidents are at a peak. Mostly youngsters are the victims of such incidents. One of the main reasons behind this scenario is avoiding helmets while riding motorcycles. To have a solution for this issue, video-surveillance cameras are implemented in the roadways to catch the riders without helmets. However, this is not completely efficient due to the inaccuracies with the system. Motorcycles with high speed may not be captured in this system. Video-surveillance camera is only applicable to motorcycles at a moderate speed. Moreover, helmets with different colors, shapes, and sizes are available in today's market. The system operates with some predefined measurements related to the helmets. To solve this issue, a system with more reliability should be designed and implemented. This should be designed to perform custom operations. Investigating measurements that are given by the ministry of service, India Today Data Intelligence Unit (DIU) [1] has discovered that of all the street mishaps that occurred in 2017, bikes were the most exceedingly awful hit. In 2017, more than 48,746 bike clients passed on in street mishaps. Fig.1 shows the statistical image of road accidents in India. Unexpectedly, 73.80 percent were not ready to wear a helmet.

Arya K M. is associated with Department of Electronics and Communication Engineering, Government College of Engineering Kannur, Kerala, India.(Email: aryakmohanan@gmail.com)

Dr. Ajith K K. is a faculty of Department of Electronics and Communication Engineering, Government College of Engineering Kannur, Kerala, India. (Email: ajithkk@gcek.ac.in)

This shows that four bike riders who died in a street mishap consistently did not wear a protective helmet.

Moreover, helmets are very necessary items in all the workplaces. Wearing helmet is also considered as a mandatory rule for the employees but, it is not followed well. Workplaces may be full of hazards. There can be flaws in maintaining workstation. So, employees may be insecure. There are chances to slip and fall on the floor. Such incidents may be dangerous for their lives. It could make injuries to their head which can be considered a serious issue. To avoid this, helmets are always preferred.


Fig. 1. Statistical image of road accident in India, 2017.

Over recent years, several ways to detect the helmet were designed. Chiverton et al. (2012) [2] developed a Histogram of Oriented Gradients (HOG) and Support Vector Machines (SVM) based helmet detection system. HOG is used to extract features of motorcyclists and their heads and SVM for feature classification which obtained 85 % accuracy. Waranusast et al. (2013) [3] used color-based features for helmet identification. However, that approach used only one descriptor for feature extraction. Silva et al. (2013) [4] studied a hybrid descriptor, combined with HOG, Local Binary Patterns (LBP), and Circle Hough Transform (CHT). Random Forest is used as the classifier, then the system became 94.23 % accurate. This suggested that types of descriptors are combined properly, which could make a better performed one. Dahiya et al. (2016) [5] developed a helmet identification system for a motorcyclist.

It compared the performances of three widely used feature descriptors, Scale-Invariant Feature Transform (SIFT), HOG, and LBP using an SVM classifier. For those, features used for identification are from manual designs. Existing video surveillance based methods for helmet detection are passive and require significant human assistance. In general, those designs are practicable due to the influence of human beings even though it cannot be long term efficient. The automatic system is highly desirable for reliable and robust monitoring also for reducing human resources needed. Real-time implementation, occlusion, the direction of motion, temporal changes in conditions like illumination and shadow, quality of the video, and so on are should be considered seriously for helmet detection. These will make helmet detection a challenging task. Using features extracted by deep CNN for detection tasks is better than using hand-engineered features. Recently, deep CNN has become the dominant algorithm in computer vision and pattern recognition tasks.

## II. OVERVIEW OF DEEP LEARNING BASED HELMET DETECTION

Deep learning is a subset of machine learning that very closely tries to mimic the human brain's working using neurons. This technique is focused on building artificial neural networks using several hidden layers. There are varieties of Deep learning networks such as Multilayer Perceptron (MLP), Autoencoders (AE), and Convolutional Neural Networks (CNN), Recurrent Neural Network (RNN). The processing power needed for Deep learning is readily becoming available using GPUs, Distributed computing, and powerful CPUs. Moreover, as the data amount grows, the Deep learning model seems to outperform Machine learning models. Most of the Deep learning network detect object accurately in very fast. RCNN, Fast RCNN, Faster RCNN, SSD, YOLO, and so on are the most familiar Deep learning networks. Object detection includes the operations of object location and classification. CNN has been successfully used in object detection to improve the object location and classification accuracy. There are three tasks that object detection must solve classification, localization, and finding all objects. These three tasks are also found in helmet detection. Compared to simple image classification, there are many more aspects that object detection needs to address. As a result, the computational complexity is significantly higher than that of the image classification. There are mainly three tasks in any deep-learning-based automatic helmet detection system.

- *Data collection:* In real-time helmet detection applications, input data is video, converted into the image frame. Video captured from the front, back, or even from sides.
- *Pre-processing*: Data is converted into a suitable format for further processing. Each algorithm uses a different type of pre-processing techniques. It may change the size, aspect ratio, color, and so on.
- *Feature extraction and detection:* Deep learning techniques extract features from the input image and detect an object. An algorithm may be a single-stage or two stages. Most of the algorithm provides a bounding box and a confidence score for the detected object.

### A. *Real-Time Object Detection With Reduced Region Proposal Network via Multi-Feature Concatenation*

Helmet detection is a real-time application (Helmet detection in traffic monitoring), so inference speed is an important consideration. Faster RCNN [6] is a two-stage object detection algorithm. Faster RCNN utilized Region Proposal Network (RPN), which is used to find all the Region of interest (ROIs), for object detection. So it can be used for helmet detection. VGG-16 backbone based Faster R-CNN will give high accuracy. In Faster RCNN there are several layers of pooling, the feature maps become too small for accurate bounding box regression and classification. Also, many ROIs are generated by RPN. This leads to an increase in the number of computations (So increase detection time) in case of helmet detection. Also, the detection rate is less than 10f/s.

Kuan et al. [7] proposed real-time object detection with reduced RPN (RRPN) via multi-feature concatenation. This proposed method can detect helmet accurately. Faster RCNN network is modified for improving performance (inference time and accuracy) using some useful tricks. The pruning, RRPN, and Assisted multi-feature concatenation, are the three primary modules in this improved faster RCNN.

- *Pruning:* Most of the weights in the neural network are useless. If remove some weights, will reduce the model size. Pruning is motivated by number of connections in the human neural system. At the time of birth, a baby has 50 trillion connections in neurons, it will increases to 1000 trillion after 1 year. Pruning begins to occur and it will reduce to 500 trillion connections after 10 years. The pruning mechanism removes redundant connections in the brain. It is possible to adopt weight pruning on a neural network. Compression of parameter solving the memory and performance issues. Pruning and quantization are the two main methods for removing less significant weights. Quantization requires customized hardware support during run time. In the proposed method pruning is done in both convolution and fully connected layer (FC). Use the absolute sum method and find significant weight, then remove redundant weights. This helps to reduce parameter and amount of computation. Pruning decreases the model size. However, pruning involves deleting some filters, which, in turn, results in loss of accuracy and information loss.
- *Reduced regional proposal network*: After pruning, Faster RCNN uses a reduced RPN to compensate for accuracy loss. Improved RPN improving overall accuracy. There are two methods for improving RPN, change the architecture or input. Reduced RPN is the architecture modification of RPN. Use simple convolution to adjust channel dimension, reduce channel in RPN, it helps to reduce parameters. RRPN consists of a convolution layer with 1x1 kernels and another layer with dilated convolution (convolution applied to the input image in defined gaps). These are simpler convolution, helps to adjust the channel dimensions of feature maps.

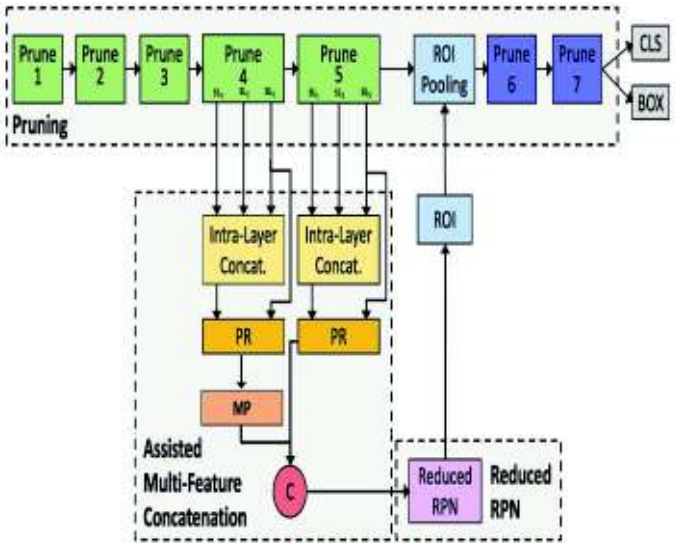


Fig. 2. Overall architecture [7].

- *Assisted multi-feature concatenation:* Intra-layer concatenation and proposal refinement (PR) are used in the assisted multi-feature concatenation. The output of this concatenation module is given as an input of RRPN. There are two types of feature map shapes and characters. Normally the last layer of the neural network has more redundant features, but the positive effect when taking feature maps from all layers. Sub-layers 1, 2, and 3 in both conv4 and conv5 of Faster RCNN are used in the assisted multi-feature concatenation. Then the concatenation result is given to the PR [9] block. PR block converts a feature map to ROI related data and helps to increase detection accuracy. The first PR block output is given to a maximum pooling layer, that makes feature maps of conv4 match with feature map of conv5.

Fig 2 shows the overall architecture of the proposed method. Improved Faster RCNN tested on ZF-Net [10] and VGG16 [11] based framework. Pruning helps to reduce more than 70% parameters. Pruning is both a convolutional layer and FC layers help to reduce parameters dramatically. Moreover, it will compress the network. After pruning, the mean average precision (mAP) values are 58.1% and 66.5% for ZF-Net and VGG16 respectively. After adding RRPN and assisted multi-feature concatenation, mAP can be improved by 2.1%–60.2% for ZF-Net and by 2.6%–69.1% for VGG16. Finally Improved Faster RCNN, compress the parameters of ZF-Net by81.2%, and the parameters of VGG16 are compressed by 73%. This compression helps to make the small model for memory limited systems. The inference speed of ZF-Net became 40 f/s and 27-f/s VGG16.This improved faster RCNN will help to make efficient helmet detection.

B. *Visual and Semantic Knowledge Transfer for Large Scale Semi-supervised Object Detection*

Helmets are various types; motor rider's helmet is different from construction workers. But they have a large similarity. Yuxing et al. [12] proposed a methodology for object detection, which can be used for helmet detection in different real-time applications. The developed method utilized visual as well as semantic similarity. A classifier and detector have some category-specific difference. Motor rider's helmet and construction worker's helmet have visual and semantically similar properties. Almost they have the same appearance. Semantically similar means both of them are related to safety and accident. For example, the difference between the dog detector and the classifier can be used for a cat detector, other than a tree detector. Likewise difference between motor riders helmet detector and the classifier can be used for construction worker's helmet detector and vice versa.

Deep learning wants a huge amount of datasets for training purposes. Pre-processing requires manual annotation of input with bounding boxes, it will become time-consuming, unreliable, and extremely laborious. There are two types of annotations, image level annotation and object-level annotation. In an image-level annotation there is no need for a bounding box, only just annotated each image with a label, it is simple. In object-level annotation, uses bounding boxes and labels. Object categories with both image-level and object-level annotations are called "strong" categories. "Weak" categories mean categories with only image-level annotations. It is possible to finds a transformation between CNN classifiers and detectors of strong categories because they have object-level annotation. Object detector can't directly train on weak categories, training process and fine-tuning requires bounding box annotation. The transformation of strong categories can be applied in weak categories to train a detector in a semi-supervised manner. The main advantage of the proposed method, it reduces the workload and speeds up the training process. This transformation is actually based on visual and semantic knowledge. Yuxing et al utilized Large Scale Detection through Adaptation(LSDA) [13] framework, upon which the proposed approach is based. LSDA combines adaptation methods to a deep convolutional neural network, create a fast and effective large scale detector. Knowledge transfer models improve LSDA. Shape and texture are considering for measuring visual similarity. Consider D be the dataset of K categories to be detected. Set of m (m < K) is strong categories, denoted as B, weak categories denoted as A. K Image classifier is trained D (D = A U B), but only m object detectors (from B) can be learned.

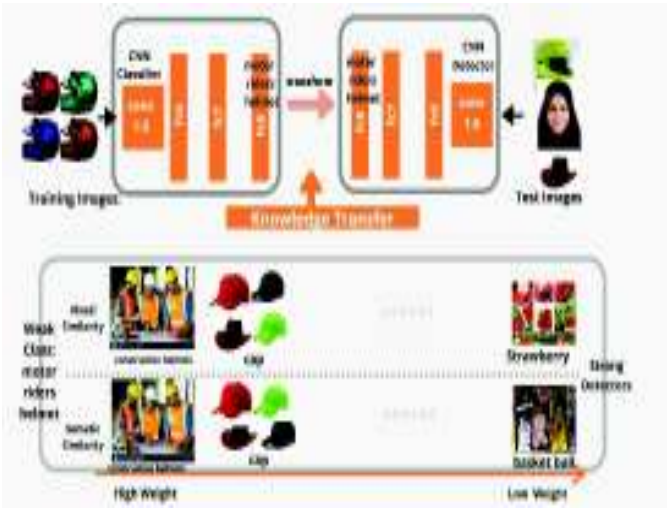


Fig. 3. An illustration of similarity-based knowledge transfer model.

The only set of m is strong categories (bounding box annotations available only for strong categories). The LSDA algorithm learns to convert (K-m) image classifiers (from A) into their corresponding object detectors through different steps (Pre-training, Fine-tuning for classification, Category-invariant adaptation, and Category-specific adaptation). Similarity measurement is used to find the nearest categories and weight them accordingly, convert an image classifier into an object detector. Illustration of similarity-based knowledge transfer model shown in Figure 3.

- *Knowledge Transfer via Visual Similarity:* The visual similarity (denoted Sv) between j and i (j ∈ A and i ∈ B) is modeled, transfer this knowledge for making classifiers into detectors for A.

$$s_v\,(j,i)\;\alpha\;\frac{1}{N}\sum_{n=1}^{N} CNN_{softmax}\,(I_n)i \qquad (1)$$

Equation (1) is used to find visual similarity. A is considered as the validation set, where In is a positive image from category j. N is the number of positive images for category j, and $CNN_{softmax}$(In)i is the ith CNN output of the softmax layer on In, namely, the probability of In being category i ∈ B as predicted by the fine-tuned classification network. Sv (j, i) ∈ [0, 1] is the degree of similarity after normalization on all the categories in B. By using this visual similarity measurement and weight of classifier calculate the weight of detector for j categories in A. That is shown in equation (2). $\Delta Bij$ Is the variance matrix of the last fully connected layer of LSDA after fine-tuning.

$$\forall j \in A: w_{jv}^{d} = w_j^{c} + \sum_{i=1}^{m} s_v(j,i)\Delta_{B_i^j} \qquad (2)$$

Using equation (2) calculated the weight for the detector of weak categories. It is a weighted nearest neighbour scheme.

- *Knowledge Transfer via semantic Similarity :* Semantic similarity is based on the natural language domain. Embedding is a collection of vectors (the word from natural language domain converted into a vector of real numbers). By using synset [14] embedding each category is converted into a word vector, then compute the L2-norm of each pair. ds(j; i) is the semantic distance between j and i categories. Each category j ∈ $A$ is represented as a vector of m ∈ B and calculated the distance using L2-norm. Then calculating semantic similarity, is inversely proportional ds(j, i) and weight for a detector.

$$\forall j \in A: w_{js}^{d} = w_j^{c} + \sum_{i=1}^{m} s_s(j,i)\Delta_{B_i^j} \qquad (3)$$

Table 1 shows that visual and semantic knowledge transfer improving mean average precision for weak categories. This type of knowledge can be used for helmet detection in real-time.

TABLE 1
Mean average precision (mAP) for different method

| Method | Number of Nearest Neighbors | mAP on A: "Weakly labeled" 100 Categories |
|---|---|---|
| LSDA | avg/weighted – 5<br>avg/weighted – 10 | 15.97/16.12<br>16.15/16.28 |
| LSDA with visual transfer | avg/weighted – 5<br>avg/weighted – 10 | 17.42/17.59<br>17.62/18.41 |
| LSDA with semantic transfer | avg/weighted – 5<br>avg/weighted – 10 | 17.32/17.53<br>16.67/17.50 |

### C. *Helmet Use Detection of Tracked Motorcycles Using CNN-Based Multi-Task Learning*

Usually, the handcrafted feature-based binary classifier has some disadvantages in the case of motor cyclist's helmet detection. When there are many motorcycles such a method may be failed. The deep learning-based method automatically develops representation based on input image data. Normally detect motorcyclists and helmet through two separate CNN, it is time-consuming and not suited for real time application. Some technique uses a single CNN. Han et al. [15] proposed a Multi-Task Learning (MTL) framework for patch-based helmet use classification. In that developed methodology pre-trained and fine-tuned RetinaNet first detect active motorcyclists and track them. In the next step, helmet use is identified. Retina Net is a single-stage deep learning network. Given input is fed into backbone CNN to generate multi-scale feature maps. From which multi-scale feature pyramid is generated. At each level of the pyramid, there are two sub-networks. One for regression from anchor box to ground truth box and the other for classify anchor boxes. After non-maximum suppression across the multi-scale feature pyramid, Retina Net arrives at the detection result. Instead of training from scratch, fine-tuning the Retina Net model with pre-trained weights obtained by the COCO dataset. A cropped image patch of tracked motorcyclists is used for the helmet detection step. CNN is used to train for two purposes, visual similarity learning for tracking motorcycles and patch-based helmet classification. MTL helps to learn both tasks simultaneously MTL means that a single model able to learn a different task from a single input. MTL is very useful in helmet detection. Because automatic helmet detection application involves different task like moving vehicle tracking, bike detection then helmet detection in case of the traffic monitoring system. While considering the helmet detection system in construction sites that needs person detection and tracking as well as helmet detection.

X (a) and x (b) are two image patches. That is used as the input to the Siamese network. The network uses an InceptionV3 [16] CNN body with shared weights. The global average pooling layer (GAP) and the final fully-connected (FC) layer of the network is removed. Input, x (a), and x (b) are transformed into feature vectors after passing the output of the CNN body to a GAP layer. By using these feature vectors, the

network predicts helmet use class. MTL significantly decreases the computational cost.


Fig. 4. Architecture for patch-based helmet use classification and visual similarity learning [15].

.

D. *An advanced deep learning approach for safety helmet wearing detection*

Lin Shi et al. [17] proposed an Improved Faster RCNN based methodology for helmet detection in a construction site. In most of the real-time applications helmet appears a small portion compared to the background. The size of the helmet will be very different in each input image. Faster RCNN is a two-stage architecture, can be used to detect helmet, but it can't give better results in different scale input. Original faster RCNN network scale shorter side to 600 pixels by keeping the original aspect ratio of the output image and unchanged. This results in poor generalization performance of different size objects. Faster RCNN can be improved using different strategies. Lin Shi et al applied three methods for improving Faster RCNN, Multiscale training, increasing anchor box, and Online Hard Example Mining (OHEM).

- *Multiscale training:* Improved faster R CNN randomly resize image shorter side to 480 or 600 or 750, and randomly select one of the three for the training process. The multiscale strategy is adopted in the network training stage to increase the robustness of object size.
- *An increasing number of anchor boxes:* The number of anchor boxes is the very important parameters in CNN. It affects the generation of ROIs. Faster RCNN uses 9 anchor boxes, each sliding window produces 9 ROIs of different scales and aspect ratio. Faster RCNN uses 1:1, 2:1, 1:2 scales and 128x 128, 256x256,512x512 aspect ratio. This does not help to detect a small object. Improved Faster RCNN uses an extra 3 anchor with 64x64 (smaller than the default) aspect ratio and the same scale. This will help to detect a small helmet in an input image. A total of 12 anchor boxes are used.
- *OHEM*: Faster RCNN consists of RPN and detection module. RPN module is used to generate ROIs and predict object bounding box and confidence scores. Redundant ROIs are eliminated by RPN. Some ROIs are assigned with a positive label and some with a negative label. The largest ROI for IOU with ground truth box and ROIs with IOU greater than 0.7 are considered a positive label. A negative label means all ROIs with IOU less than 0 .3 with ground truth boxes. RPN generates more than 128 ROIs and randomly selects some of them for the training process. Faster RCNN uses 1:3 as the ratio of positive and negative samples. Random selection or ROIs leads to a model select a large number of invalid samples, model degradation. OHEM help to select hard samples and train the network, improving performance.

The structure of improved faster RCNN is shown in figure 5. Faster RCNN has an ROI pooling network, ROI network. OHEM expands the ROI network into two ROI network. The two network s will share network parameters. One of them is read-only for calculating and sorting the loss value of all regional proposals. Then select 128 regional proposals with large loss value. Another ROI network is the standard ROI network. This network will predict classification results and coordinate of the bounding box for a hard sample selected by the read-only ROI network. This will help to improve the accuracy of object detection.

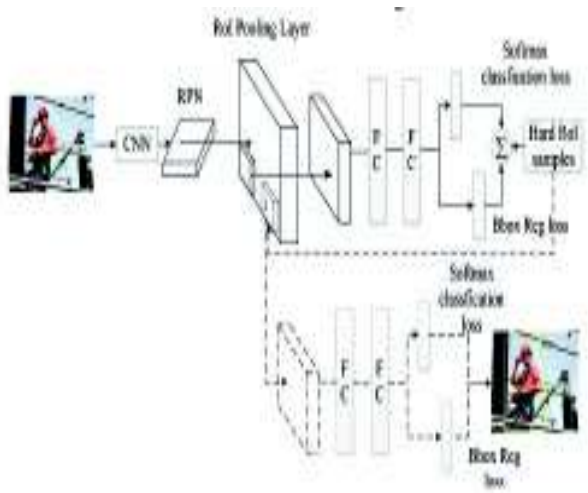


Fig.5. The structure of improved faster RCNN [17].

TABLE II
Comparison of Faster RCNN and Improved Faster RCNN

| Method | Recall % | Precision % |
|---|---|---|
| Faster RCNN | 82.22 | 87.57 |
| Improved Faster RCNN | 90.45 | 91.42 |

Improved Faster RCNN has better precision and recall rate, and significantly improve the performance of helmet detection.

E. *An Automatic Detection of Helmeted and Non-helmeted Motorcyclist with License Plate Extraction using Convolutional Neural Network*

YOLO is one of the single-stage architecture. YOLO has different versions like YOLO v1, YOLO v2, YOLO v 3 [19], and so on. Jimit et al. [18] utilized YOLO v2 architecture for the helmet detection stage. Use YOLOv2 at two different stages one after another to improve the helmet detection accuracy. First

YOLO v2 is trained on COCO data sets, used to detect all classes. By using intermediate processing discard all classes other than the person and cropped a person's image automatically. Cropped images of detected persons are utilized as input to the second YOLOv2 stage which was trained on the data set of helmeted images. Second YOLO v2 was a modified version (Final convolution layer of YOLO v2 has only 30 filters) to detect a single helmet class. Darknet framework based YOLOv2 contains 23 convolutional layers and 5 max-pooling layers, it can learn a general representation of an object. The helmet is different in shape and size, this shape and size are a general representation. So it is very helpful to use YOLO v2 to detect the helmets in real-time with a high confidence score. YOLO v2 can find contextual information about the object (Usually helmets are worn in the head region of a person, this is contextual information.) other than their appearance. Moreover, YOLOv2 predicts class probabilities and bounding boxes directly from input images in one evaluation. Jimit et al developed a methodology using the pre-trained network for helmet detection. Training a neural network with one task, form parameters. These parameters can be used for training another task. By using pre-training, it will help to reduce the amount of training time. YOLO v2 pre-trained on a huge and manifold dataset like the ImageNet, captures features like edges and curves in its early stages. Pre-training is very helped fully in most of the classification problems. After pre-training second YOLO v2 is trained on helmet data sets and tested using helmeted and non -helmeted images. Results obtained using the developed method by Jimit et al for different scenarios are shown in Fig 6.

Fig. 6. Detected helmet in different scenarios [18].

YOLO v2 can distinguish cap and helmet very accurately, even both of them have similar features. It will be differentiated scarves from the helmet. Different colored and shaped helmets will detect accurately. Moreover, it detects the helmet of motorcyclists captured from the side-view camera. YOLO is an effective algorithm for helmet detection. Jimit et al obtain helmet detection accuracy of 94.70 % with precision 0.9463 and recall 0.9486.

## III. CONCLUSION

Different types of deep learning techniques have made great progress in real-time object detection. The performance of a deep learning-based helmet detection system depends to a large extent on the network's ability to learn features. Accurate and fast helmet detection techniques need complex methodology. First, this paper analyzes the object detection method, which can be very effectively used in helmet detection applications. Real-time object detection with reduced region proposal network via multi-feature concatenation, this paper helps to understand how pruning reduces the parameters and complexity of the network. Also, multi-feature concatenation and RRPN increase the accuracy of detection. Knowledge about the visual and semantic similarity between strong and weak categories can be transferred to make a detector of weak categories. These methodologies help to use existing helmet detection systems to develop a new system with higher accuracy and less training time. This idea is discussed in the second paper, visual and semantic knowledge transfer for large scale semi supervised object detection. The real-time helmet detection systems have different tasks. Helmet use detection of tracked motorcycles using CNN-based multi-task learning, this paper very clearly tells why MTL provides fast detection. Most MTL system uses the Siamese network for different tasks. The advanced deep learning approach uses different techniques like multi-scale training, an increasing number of anchor boxes, and OHEM for helmet detection. Automatic detection of a helmeted and non-helmeted motorcyclist with license plate extraction using a convolutional neural network discusses how YOLO effectively detects a helmet from an image. Moreover, YOLO is the most used helmet detection algorithm for different applications. YOLO has a different version from YOLOv1 to YOLOv5. YOLO is one of the best choices for helmet detection applications in real-time. Almost every deep learning approach provides more than 90% accuracy for real-time helmet detection.

## REFERENCES

[1] https://www.indiatoday.in/diu/story/two-wheeler-death-road-accidents-helmets-states-india-1602794-2019-09-24 [Online].
[2] J. Chiverton, “Helmet presence classification with motorcycle detection and tracking,” IET Intelligent Transport Systems, vol. 6, no. 3, pp.259–269, September 2012.
[3] R. Waranusast, N. Bundon, V. Timtong, C. Tangnoi, and P. Pattanathaburt, “Machine vision techniques for motorcycle safety helmet detection,” in 2013 28th International Conference on Image and Vision Computing New Zealand (IVCNZ 2013), pp. 35–40,Nov 2013.
[4] R. Silva, K. Aires, T. Santos, K. Abdala, R. Veras, and A. Soares,“Automatic detection of motorcyclists without helmet,” in Latin American Computing Conference (CLEI), ,pp. 1–7, Oct 2013.
[5] K. Dahiya, D. Singh, and C. K. Mohan, “Automatic detection of bikeriders without helmet using surveillance videos in real-time,” in 2016 International Joint Conference on Neural Networks (IJCNN), pp. 3046–3051, July 2016
[6] S. Ren, K. He, R. Girshick, and J. Sun, ``Faster R-CNN: Towards real time object detection with region proposal networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137_1149, Jun. 2017.
[8] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” in Proc. Adv. Neural Inf. Process. Syst., pp. 1097–1105, 2012.
[9] C. Peng, X. Zhang, G. Yu, G. Luo, and J. Sun, “Large kernel matters—Improve semantic segmentation by global convolutional network,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), pp. 1743–1751, Jul. 2017.
[10] M. D. Zeiler and R. Fergus, “Visualizing and understanding convolutional networks,” in Proc. Eur. Conf. Comput. Vis. Basel, Switzerland: Springer, 2014, pp. 818–833.

[11] A. Zisserman and K. Simonyan, "Very deep convolutional networks for large-scale image recognition," Sep. 2014, arXiv:1409.1556. [Online].Available: https://arxiv.org/abs/1409.1556.

[12] Yuxing Tang, Josiah Wang, Xiaofang Wang, Boyang Gao, Emmanuel Dellandr´ea, Robert Gaizauskas and Liming Chen, "B.Visual and Semantic Knowledge Transfer for Large Scale Semi-supervised Object Detection",2017

[13] J. Hoffman, S. Guadarrama, E. Tzeng, R. Hu, J. Donahue, R. Girshick, T. Darrell, and K. Saenko, "LSDA: Large scale detection through adaptation," in Neural Information Processing Systems (NIPS), 2014.

[14] S. Rothe and H. Sch ¨ utze, "Autoextend: Extending word embeddings to embeddings for synsets and lexemes," in the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (ACL),2015.

[15] Hanhe Lin , Jeremiah D. Deng, Deike Albers et al, " Helmet Use Detection of Tracked Motorcycles Using CNN-Based Multi-Task Learning", IEEE open access journal, 2020.

[16] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, ``Rethinkingthe inception architecture for computer vision,'' in Proc. IEEE Conf.Comput. Vis. Pattern Recognit. (CVPR), pp. 2818-2826. Jun. 2016.

[17] BShoukun Xu, Yaru Wang, Yuwan Gu, "An advanced deep learning approach for safety helmet wearing detection", in 2019 International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData), oct 2019

[18] Jimit Mistry, Aashish K. Misraa, Meenu Agarwal,"An Automatic Detection of Helmeted and Non-helmeted Motorcyclist with License Plate Extraction using Convolutional Neural Network" in 2017 seventh international conference on image processing theory,tools and applications pp. 156-176, March 2018

[19] J. Redmon and A. Farhadi, "YOLO9000: better, faster, stronger," CoRR, vol. abs/1612.08242, 2016. [Online]. Available: http://arxiv.org/abs/1612.0824.