ISBN: 978-1-948012-16-4

Contemporary Issues in Computing (CIC)

DOI: <http://doi.org/10.26480/cic.01.2020.48.50>

CrossMark

CNN BASED OBJECT DETECTION SYSTEM: A REAL TIME APPLICATION

P Rohini^a, L Koteswara Rao^b, P Sree Lakshmi^c^aDepartment of CSE, Faculty of Science and Technology, ICAI Foundation for Higher Education, Hyderabad^bDepartment of ECE, Koneru Lakshmaiah Education Foundation, Hyderabad, India^cDepartment of IT, BVRIT Hyderabad, India

This is an open access article distributed under the Creative Commons Attribution License CC BY 4.0, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

ARTICLE DETAILS

Article History:

Received 26 October 2020

Accepted 27 November 2020

Available online 03 December 2020

ABSTRACT

In many countries, two-wheeler is the most common means of transport. However, because of less protection, this is considered as a highly risky mode of transport. As a solution to this, bike-riders can use helmet to safeguard the head. Upon observing the role of Helmet during the two-wheeler ride, many governments made the helmet compulsory for the riders. Many government functionaries follow the manual strategies to identify the violators. Manual means of identification is a time consuming process. Helmet is one of the main protection unit for a motor bicyclist. However, many to conform to the law of wearing helmet. Here, to detect the motorcyclists who are violating the helmet laws, a system using image processing and convolutional neural network is implemented. The system consists of motorbike detection, helmet vs no helmet classification and motorbike licence plate recognition. The motorbikes are detected using the feature vector HOG. Once the motorbike is detected, by means of convolutional neural network, it is determined whether the motorcyclist is driving with helmet or without helmet. Whenever any motorcyclist is identified without a helmet, license plate of the motorcycle is detected using TensorFlow object detection.

KEYWORDS

CNN, object detection, Helmet.

1. INTRODUCTION

According to the statistics, in India about 28 two-wheeler riders die daily on Indian roads in 2016 because of not wearing helmets. In the year 2017, 31 out of 100 people died in road accidents which shows increased rate from 21.6 deaths per 100 accidents in 2005. Also, it is proved that one of every five bike riders who died on roads were not wearing helmet. There are existing methods which use specialized sensors in the ergonomics of the motorbike to check the presence of helmet (Chiverton, 2012; Silva et al., 2013; Silva et al., 2014; Doungmala et al., 2016; Li et al., 2019). But it is impossible to convince every user for installation of sensors on the already existing bikes. Also, the accuracy and integrity of these sensors is questionable. Apart from this, systems that use video processing have very high computational costs. The technologies that were used to build the system very expensive hence making it an economically non-viable choice.

This paper tries to mitigate the aforementioned problems by proposing a potential solution using continuous real time video processing of the traffic video feed. The proposed solution will provide a completely free of cost system once installed. Also, the software was built using free and open source technologies hence it has an overall software cost equal to zero. Every vehicle on the street will be evaluated and a database of the offenders will be generated real time. Hence given that every rider not wearing helmet is prosecuted, there will be an increased awareness in the public.

2. AN OVERVIEW OF THE DIFFERENT R-CNN BASED ALGORITHMS

2.1 R-CNN

R-CNN extracts a bunch of regions from the given image using selective search, and then checks if any of these boxes contains an object. We first extract these regions, and for each region, CNN is used to collect specific features. Finally, collected features are used in object detection. Unfortunately, R-CNN becomes rather slow due to these multiple steps involved in the process. The problem the R-CNN system tries to solve is to locate objects in an image (object detection). You could start with a sliding window approach. When using this method, you just go over the whole image with different sized rectangles and look at those smaller images in a brute-force-method. The problem is you will have a giant number of smaller images to look at. To our luck, other smart people developed algorithms to smartly choose those so-called region proposals.

Well, this is one fundamental issue with this R-CNN system. You can't train the whole system in one go (This will be solved by the fast R-CNN system). Rather, you will need to train every part independently. That means that the Alex Net was trained before on a classification task. After the training, they removed the last SoftMax layer. Now the last layer is the fully connected 4096-dimensional one. This means that our features are 4096 dimensional.

Another important thing to keep in mind is that the input to the Alex Net is always the same (227, 227, 3). The image proposals have different shapes

Quick Response Code	This paper was presented at	Access this article online	
	International Conference on Contemporary Issues in Computing (ICCIC-2020) - Virtual IETE Sector V, Salt Lake, Kolkata From 25th-26th July 2020	Website: www.intelcomp-design.com	DOI: 10.26480/cic.01.2020.48.50

though. Many of them are smaller or larger than the required size. So, we will need to resize every region proposal.

Algorithm	Features	Prediction time / image	Limitations
CNN	Divides the image into multiple regions and then classifies each region into various classes.	–	Needs a lot of regions to predict accurately and hence high computation time.
R-CNN	Uses selective search to generate regions. Extracts around 2000 regions from each image.	40-50 seconds	High computation time as each region is passed to the CNN separately. Also, it uses three different models for making predictions.
Fast R-CNN	Each image is passed only once to the CNN and feature maps are extracted. Selective search is used on these maps to generate predictions. Combines all the three models used in R-CNN together.	2 seconds	Selective search is slow and hence computation time is still high.
Faster R-CNN	Replaces search method by Region proposal network (RPN) which makes the algorithm much faster.	0.2 seconds	Object proposal takes time and as there are different systems working one after the other, the performance of systems depends on how the previous system has performed.

2.2 Faster R-CNN

From the literature, it is observed that Faster R-CNN fixes the problem of Selective search by replacing it with Region Proposal Network (RPN). We first extract feature maps from the input image using ConvNet and then pass those maps through a RPN to return object proposals. Subsequently, the maps are classified to predict the bounding boxes. **Faster R-CNN** makes further progress than Fast R-CNN. Search Selective process is substituted by **Region Proposal Network (RPN)**. As the name revealed, RPN is a network to propose regions. For instance, after getting the output feature map from a pre-trained model (VGG-16), if the input image has 600x800x3 dimensions, the output feature map would be 37x50x256 dimensions.

Both of the above algorithms i.e R-CNN and Fast R-CNN use Selective Search method to find out region proposals. It is observed that the Selective search is a slow and time-taking process affecting the overall performance of network. Therefore, we came up with an object detection algorithm which eliminates the problems of selective search method. Proposed method enables the network learn about region proposals.

Resembling the Fast R-CNN, image is given as input to a convolutional network and convolutional feature maps are extracted. Replacing the Selective search algorithm, a separate network is employed to predict region proposals. Predicted region proposals are reshaped by means of a RoI pooling layer. Subsequently, classification of image within the proposed region and prediction of offset values for bounding boxes is done.

3. IMPLEMENTING FASTER R-CNN FOR HELMET DETECTION

3.1 Algorithm

Step 1: Gather the data

Step2: Label the data

Step 3: Creating a label map

Step 4: Installing TensorFlow-GPU

Step 5: Create virtual environment

Step 6: Configure environment variable

4. EXPERIMENTAL RESULTS

4.1 Algorithm is Implemented to Detect the Helmet from Various Backgrounds. Results of the Same are Depicted in Fig. 1

Figure 1: Helmet detection from three different backgrounds

5. COMPARISON BETWEEN YOLO AND FASTER R-CNN

We can see that YOLO and Faster RCNN both share some similarities. They both use an anchor box-based network structure, both use bounding box regressions. Things that differ YOLO from Faster RCNN is that it makes classification and bounding box regression at the same time. Judging from the year they were published; it makes sense that YOLO wanted a more elegant way to do regression and classification. YOLO however does have its drawback in object detection. YOLO has difficulty detecting objects that are small and close to each other due to only two anchor boxes in a grid predicting only one class of object. It doesn't generalize well when objects in the image show rare aspects of ratio. Faster RCNN on the other hand, does detect small objects well since it has nine anchors in a single grid, however it fails to do real-time detection with its two-step architecture.

6. CONCLUSIONS

In this paper, we devised a CNN based method for object detection. Identification of riders who wear the helmet is done effectively.

REFERENCES

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

- J. Chiverton. 2012. Helmet presence classification with motorcycle detection and tracking, in IET Intelligent Transport Systems, 6(3), 259-269.
- Li, W., Liu, K., Yan, L. *et al.* 2019. FRD-CNN: Object detection based on small-scale convolutional neural networks and feature reuse. *Sci Rep* 9, 16294. <https://doi.org/10.1038/s41598-019-52580-0>
- Silva, K.A., T. Santos, K. Abdala, R. Veras and A. Soares. 2013. Automatic detection of motorcyclists without helmet, 2013 XXXIX Latin American Computing Conference (CLEI), Naiguata, 1-7.
- Silva, R.R.V., K.R.T. Aires and R.d.M.S. Veras. 2014. Helmet Detection on Motorcyclists Using Image Descriptors and Classifiers, 2014 27th SIBGRAPI Conference on Graphics, Patterns and Images, Rio de Janeiro, 141-148