Traffic Signal Recognition System Based on Color and Time for Visually Impaired

P. Kamakshi
pk.it@kitsw.ac.in
Kakatiya Institute of Technology & Science Warangal, India

Abstract

Nowadays, a blind man finds it very difficult to cross the roads. They should be very vigilant with every step they take. To resolve this problem, Convolutional Neural Networks(CNN) is a best method to analyse the data and automate the model without intervention of human being. In this work, a traffic signal recognition system is designed using CNN for the visually impaired. To provide a safe walking environment, a voice message is given according to light state and timer state at that instance. The developed model consists of two phases, in the first phase the CNN model is trained to classify different images captured from traffic signals. Common Objects in Context (COCO) labelled dataset is used, which includes images of different classes like traffic lights, bicycles, cars etc. The traffic light object will be detected using this labelled dataset with help of object detection model. The CNN model detects the color of the traffic light and timer displayed on the traffic image. In the second phase, from the detected color of the light and timer value a text message is generated and sent to the text-to-speech conversion model to make voice guidance for the blind person. The developed traffic light recognition model recognizes traffic light color and countdown timer displayed on the signal for safe signal crossing. The countdown timer displayed on the signal was not considered in existing models which is very useful. The proposed model has given accurate results in different scenarios when compared to other models.

Keywords:
Convolutional Neural Networks, traffic signal, countdown timer, text-to-speech

1. Introduction

Recognition of traffic lights is a big challenging task for visually impaired people. All over the nation there are almost 2.2 billion blind people according to the national survey[1]. They are not able to cross the traffic signal safely and need help from others to cross the traffic signals. Now-a-days the traffic on roads is rapidly increasing compared to past and it is very difficult for even an ordinary person to cross road.

Many times, the people with disabilities are considered as traffic weak and they face many traffic accidents but there are no protection areas in surroundings of disabled people in India. As per the data from other developed countries, only 7 among 631 impaired offices were in incapacity insurance regions. [3] Further, only 57% of disabled people are able to use the latest developed technologies to cross the roads. Though there are acoustic signals installed at the traffic signals in some developed countries, the blind people are unable to locate them easily.

While crossing the roads, blind person carries some audio devices and tactile information to analyse the traffic signals to ensure a safe walk. Although there are many systems implemented in this view point, like push button and audio push buttons but they are not present at traffic signal areas. To avoid these challenges the mobile application serves best to give the accurate output.

Many solutions proposed are included in literature. Most of them make use of smart glasses, smart sticks, cloud managed applications, IoT devices like raspberry Pi and auditory function, GPS enabled mobile devices etc. These devices are very expensive and difficult to maintain them properly.

In this paper, traffic signal detection model is developed to help blind people while crossing the traffic signals. When compared to existing models this model performs efficiently because signal color as well as countdown timer is also considered . The countdown timer helps the blind people to know the change of signals in real time. This paper includes related work, system overview, proposed model, implementation, results and conclusion.

Manuscript received April 5, 2023
Manuscript revised April 20, 2023
https://doi.org/10.22937/IJCSNS.2023.23.4.7

2. Related Work

Research in supporting advancements for street crossing has focussed on traffic light detection and crossing direction. Most of the traffic signal recognition systems were implemented using image recognition. Yu, Yang, et al.[7] implemented a traffic signal recognition system based on road vehicle cooperative technology. The system can identify multiple traffic lights at intersection. It uses auxiliary points in order to increase the fault tolerance. Algorithm is implemented using street vehicle agreeable technology. The vehicle acquires the situation with traffic light by speaking with Traffic-Light Road Side Unit (TLRSU).

Cruz, Jerome Paul N., et al.[8] presented an object recognition tool using Artificial Neural Networks(ANN). The optimum lighting required for image is 670 lumens and optimum distance is 40cm. Optimum accuracy upto 99.99940181%. It includes three major steps: Image processing, ANN processing and Interpretations. In first process, the image is detected in terms of values. These values are sent to the second step i.e., to ANN model to recognize the image in the form of values. These values are finally sent to the third step at which they are converted into linguistic terms and helps the blind to cross the road safely at signal.

Park, Huijin, et al.[3]in their work stated that the visually impaired people are facing many challenges to cross the roads safely these days. They implemented Crosswalk lights recognition system which gives the voice guidance to the blind person. The system developed using IoT devices like raspberry Pi. The accuracy of the system is 92.7% at day and 67.3% at night. Model contains the server, raspberry Pi which is equipped with camera to record the video and then sent to the server address. The server will detect the crosswalk light using google object detection API, then colour is classified as red, green or yellow. The signal will be sent to raspberry pi and voice to blind.

Ihejimba, Chikadibia, et al.[4] developed a cloud based IoT using edge computing and mobile cloud which provides highly scalable and highly available traffic signal notification for visually impaired and blind(VIB). The system includes Mobile app, IoT device and Cloud platform.

Angin, Pelin, et al.[5] proposed a mobile cloud computing approach which used GPS services and cloud infrastructure. This model not only uses GPS signals but also the Wi-Fi access points and cell tower triangulation to provide the accurate results. The user uses the glasses in which the camera module is integrated in order to get relevant images.

Ahmetovic, et al.[9] developed Zebra crossing spotter for the protection of blind people which includes a computer vision-based technique that mines stored spatial image databases for detection of zebra crosswalks in urban cities. This is can work efficiently with large image datasets.

Bai, et al.[10] proposed cloud and vision-grounded route framework for the eyeless. The finish of the model isn't just to give route, yet additionally to cause eyeless man to believe to be a typical individual. The incorporates a head protector formed with sound system cameras toward the front, android-grounded cell phone, wed activity and cloud registering stage. The cloud working out stage is the center of the framework, incorporates object revelation and acknowledgment, OCR (Optical Character Recognition), discourse handling, vision-grounded SLAM (Simultaneous Localization and Mapping) and way arranging, which are completely grounded on profound learning calculation.

Al-Nabulsi, et al. [11] proposed an algorithm which is capable to detect the traffic lights. This algorithm is mainly useful for color blindness people. They used the 22 different types of traffic light signals of Jordan and Kuwait. The traffic light is detected by comparing the candidate traffic light with some sample traffic light images by usage of correlation. This algorithm is useful when the person is not a deaf.

3. Proposed Work

In this paper, the traffic signal recognition system based on color and time for visually impaired is proposed. It consists of two phases, in the first phase the Common Objects in Context (COCO) labelled dataset is used, which includes images of different classes like traffic lights, bicycles, cars etc. The traffic light object will be detected using this labelled dataset with help of object detection model. Then the CNN model detects the color of the traffic light and timer displayed on the traffic image. After detecting the color and timer of the signal a voice guidance is sent to the visually impaired person. In the existing systems only color is considered. There is a disadvantage by considering only the color because when the image is captured the color may be green and the time may be less than three seconds. The voice will generate a signal to cross the road but the signal may change to red which leads to risk for the visually impaired person. In the same case our model will generate a voice signal, “The signal color is green and should cross within t seconds”. As the person cannot cross within t seconds so he will wait. As another example, if the color is red and the time tis less than three seconds then the person can wait for t+2 seconds and cross the road without any risk or recapturing the image.

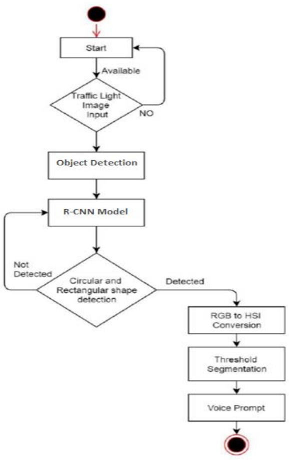

Figure 1: Flow chart of the proposed model

3.1 Coco (Common Objects In Context) Dataset

The MS COCO dataset [15] (Microsoft Common Objects In Context) is considered for experiment because it has huge scope object revelation, division, significant point disclosure and inscribing dataset. At present the dataset comprises of 328k pictures. As shown in table 1 COCO dataset is advantages than other datasets,

The COCO dataset has the annotations for

- Object Detection
- Captioning
- Key-points Detection
- Stuff Image Segmentation

Using annotations, the model identify the traffic signal object from the complete captured image. After identifying the traffic signal portion of light and countdown time will be cropped. This cropped image is further used by CNN model to detect the color and time leftover.

Datasets Name	Perspective	Number of Images	Type	Conditions (Day/Night, etc.)
GSV dataset	GSV	657,691	Zebra	Crosswalk lines may disappear, Crosswalks are partially covered, shadows affect the illumination of the road, different styles of zebra crosswalks
IARA	Vehicle	12,441	Zebra	Capture during the day
GOPRO	Vehicle	11,070	Zebra	N/A
Coco	Pedestrian	328000	Traffic lights	Capture during day and night

Table 1:Comparison of different Datasets

3.2 Convolutional Neural Network (CNN) Algorithm

The convolutional neural network (CNN) is a class of Machine learning brain organizations. CNNs addresses an enormous advancement in picture handling. CNN is most commonly used to take apart visual pictures and will continually work in backend for picture class. CNN has three layers

Input Layers
Hidden Layers
Convolutional Layers
Pooling Layers
Fully Connected Layers
Output Layers

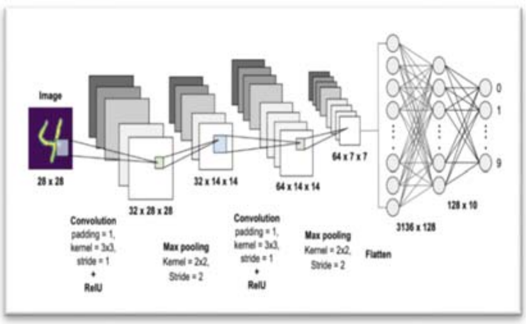

Figure 2: CNN Model

A CNN convolves (consolidates one capacity with one more by shaping their intricacy) learned highlights with input information and utilizations 2D convolutional layers. When contrasted and other picture class calculations, CNN utilizes authentically little pre-handling. This means that they can learn the filters automatically without intervention of any other hand.

If only CNN is used then explicitly multiple bounding boxes need to be drawn around the desired region in the image. In turn , the obtained output may be inaccurate and the computations may get slower. In order to overcome this problem R-CNN algorithm is used which makes computations faster.

In R-CNN selective search is used to extract upto 2000 regions from the image. These regions are called as region-proposals.

The selective search algorithm is given below.

- Produce starting sub-division, create numerous applicant regions.
- Use of Greedy calculation to join comparative area into bigger ones recursively.
- Use the created regions to deliver the last up-and-comer area proposition.

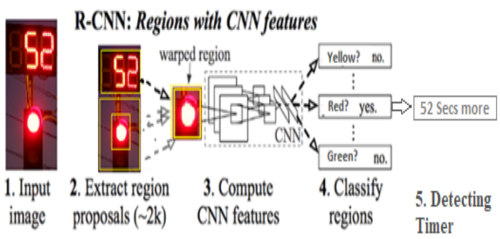

Figure 3: R-CNN model for traffic signal recognition

3.3 Proposed Traffic Signal Recognition System Design

The overall system design of our model is shown in figure 4. The system consists of three blocks:

1) Front end
2) API
3) Back end

Front end

The front part acts as user interaction. It is designed using scripting languages. It includes the functionalities like capturing the image and sending it to back-end processing. The implemented system provides complete voice guidance for capturing the image and uploading the image.

API

The API(Application Programming Interface) is mainly used to carry the request to the back-end and send the response back to the front-end part of application.

Back end

The source code gets implemented at the back-end part. It contains different methods of traffic light application like main method, detect_traffic_light, detect_red_and_yellow etc. The uploaded traffic light image will be given as an input to the main function

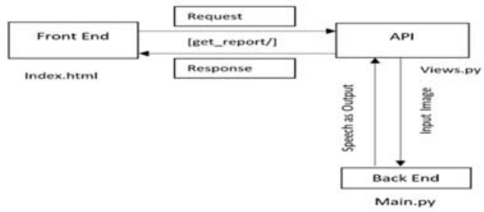

Figure 4: Overview of Proposed Method System Design

The following represents the steps to detect traffic light and recognize the status of the light and countdown time. Then the light color and time is converted from text to speech to give the voice guidance for visually impaired person.

1: read the captured image

2: call detect_traffic_lights function by passing image as a parameter, which returns status of the traffic signal(color and countdown time) to result.

3: return the result to view.py and call the text_to_speech method which gives voice guidance

The detect traffic lights function will loads the object detection model, tensor flow model and label map.

detect_traffic_lights(image)

1: set OS path to input images directory
2: load object detection model
3: load tensor flow model
4: load label map
5: define a numpy array variable as img_arr
6: **assign** the value to img_arr by calling image_to_numpy_array method
7: **define** stop_flag as a boolean variable and assign value to it by calling **read_traffic_light_obj(image)**method
8: **define** the wait_time variable as numerical value and assign it by calling image_to_text_converter() method

8: **if** stop_flag is true **then**
return ‘please stop for’ + wait_time + ‘seconds’
else then
return ‘Should cross within’ + wait_time + ‘seconds’

The *read_traffic_light*function will read the captured image. This captured image is sent to tensorflow model to crop the light color and countdown time.

read_traffic_light_obj(image)
1: **define** tf_obj variable and assign the value to it by calling object_detection model, by passing traffic light object class as a parameter to it.
2: **define** crop_img and assign the traffic image part by calling tensorflow model
3: **define** flag variable as a Boolean and assign value by calling detect_red_yellow method
4: **if** flag **then**
return true
else then
return false

The detect_red_yellow function is used to detect the color of the traffic image using the threshold value. A range of variables are defined as red and yellow. The color of input image is detected by comparing the threshold value of original red and yellow colors. Function will return true if the detected color is either red or yellow otherwise it returns false.

Detect_red_and_yellow()
1: **define** a range variable for red and yellow colors
2: **define** a threshold and assign the rgb values of red and yellow to it.

3: Compare the threshold values with rgb value of input image.

If rate > threshold then

return true // *detected as red color*

else return false // *detected as green color*

4. Experimentation and Results

The model was initially trained using COCO dataset. The dataset contains different classes of objects like cars, traffic lights and person etc as shown in figure 5. Among these different classes, our model mainly uses traffic light objects.

Figure 5: Labelling different classes using COCO dataset.

When the visually impaired person captures the traffic signal using smart glasses and gives it as input to the model. The input image is captured as shown in figure 6.

Figure 6: Traffic Signal Image

Once the image is given to the object detection model, the model using the labelled dataset crops the part of traffic lights. The cropped image is shown in below figure 7.

Figure 7: Cropped Traffic signal image

The cropped traffic signal image is then sent to detect whether the color is red, yellow or green using the threshold values. The timer attached to the traffic signal will also be read by the model to give the exact results as shown in figure. 8.

Figure 8: Detected traffic signal and timer

Once the signal and timer is detected by the proposed model, the result is sent to text to speech model to give the appropriate voice guidance to the visually impaired person.

On comparing our model with other models like Roters[16] and Cheng[17]the results shown were very accurate and have greater recall and precision. The comparison of our model performance is shown in table 2.

Class	Measure	Proposed Method	Roters	Che
Go	Recall (%)	**97.7**	55.3	57.3
	Precision (%)	93.5	**100**	97.6
Stop	Recall (%)	**98.9**	52.4	90.3
	Precision (%)	90.2	**100**	98.3

Table 2: Comparison of Our Proposed Model with other models

The accuracy of proposed system is calculated in light of the results shown. The mathematical calculation of accuracy is done using below formula.

Accuracy

$$Accuracy = \frac{True_{positive} + True_{negative}}{True_{positive} + True_{negative} + False_{positive} + False_{negative}}$$

Accuracy is a metric that by and large portrays how the model performs across all classes. It's helpful when all classes are of equivalent importance. It's determined as the rate between the quantity of right forecasts to the absolute number of expectations.

Precision

The precision is determined as the rate between the quantity of Positive examples accurately arranged to the all out number of tests named Positive (either accurately or wrongly). The precision estimates the model's delicacy in characterizing an example as sure

$$Precision = \frac{True_{positive}}{True_{positive} + False_{positive}}$$

Recall

The recall is determined as the rate between the quantity of Positive examples accurately delegated Positive to the all out number of Positive examples. The review estimates the model's capacity to descry Positive examples. The high level the review, the more sure examples identified.

$$Recall = \frac{True_{positive}}{True_{positive} + False_{negative}}$$

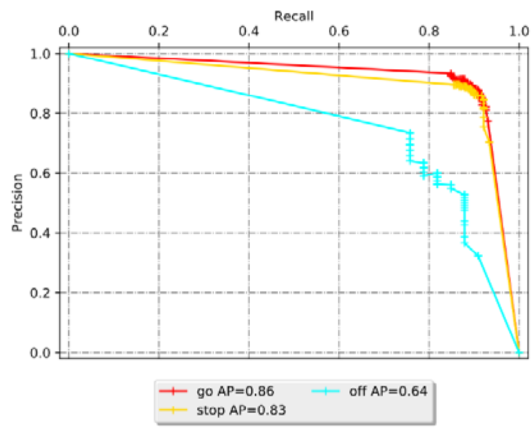

Figure 9: Precision vs Recall graph

The Figure 9 shows the results of different signals like red, yellow and green and their precision and recall in various scenarios.

5. Conclusions

In this paper, a Traffic Signal Recognition model is presented which is designed using R-CNN Model. To provide a safe environment for the visually impaired, a voice message is given according to light state and timer state at that instance. The model is trained using COCO dataset. The traffic light object will be detected using this labelled dataset with help of object detection model. The developed traffic light recognition model identifies traffic light color and countdown timer displayed on the signal for safe road crossing. The countdown timer displayed on the signal was not considered in existing models which is very useful. This model compared to other models gives accurate results in different scenarios such as during day and night.

References

[1] https://www.who.int/news-room/fact-sheets/detail/blindness-and-visual-impairment

[2] S. Sankaran et al., "A Survey Report on the Emerging Technologies on Assistive Device for Visually Challenged People for Analyzing Traffic Rules," 2020 International Conference on Communication and Signal Processing (ICCSP), 2020, pp. 0582-0587, doi: 10.1109/ICCSP48568.2020.9182335.

[3] Park, Huijin, et al. "Implementation of Crosswalk Lights Recognition System for the Blind's Safety." 2019 IEEE Eurasia Conference on IOT, Communication and Engineering (ECICE). IEEE, 2019.

[4] Ihejimba, Chikadibia, and Rym Z. Wenkstern. "DetectSignal: A Cloud-Based Traffic Signal Notification System for the Blind and Visually Impaired." 2020 IEEE International Smart Cities Conference (ISC2). IEEE, 2020.

[5] Angin, Pelin, and Bharat K. Bhargava. "Real-time mobile-cloud computing for context-aware blind navigation." International Journal of Next-Generation Computing 2.2 (2011): 405-414.

[6] Bhargava, Bharat, PelinAngin, and Lian Duan. "A mobile-cloud pedestrian crossing guide for the blind." International Conference on Advances in Computing & Communication. 2011.

[7] Yu, Yang, et al. "Design and Implementation of Traffic-Light Signal Recognition System at Intersection." 2020 IEEE 3rd International Conference on Computer and Communication Engineering Technology (CCET). IEEE, 2020.

[8] Cruz, Jerome Paul N., et al. "Object recognition and detection by shape and color pattern recognition utilizing Artificial Neural Networks." 2013 International Conference of Information and Communication Technology (ICoICT). IEEE, 2013.

[9] Ahmetovic, Dragan, et al. "Zebra crossing spotter: Automatic population of spatial databases for increased safety of blind travelers." Proceedings of the 17th International ACM SIGACCESS Conference on Computers & Accessibility. 2015.

[10] Bai, Jinqiang, et al. "A cloud and vision-based navigation system used for blind people." Proceedings of the 2017 International Conference on Artificial Intelligence, Automation and Control Technologies. 2017.

[11] Al-Nabulsi, Jamal, AbdelwadoodMesleh, and Adnan Yunis. "Traffic light detection for colorblind individuals." 2017 IEEE Jordan Conference on Applied Electrical Engineering and Computing Technologies (AEECT). IEEE, 2017.

[12] P. S. Swami and P. Futane, "Traffic Light Detection System for Low Vision or Visually Impaired Person Through Voice," 2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA), 2018, pp. 1-5, doi: 10.1109/ICCUBEA.2018.8697805.

[13] El-Taher, Fatma El-Zahraa, et al. "A systematic review of urban navigation systems for visually impaired people." Sensors 21.9 (2021): 3103.

[14] Ghilardi, Marcelo C., et al. "Real-time detection of pedestrian traffic lights for visually-impaired people." 2018 International Joint Conference on Neural Networks (IJCNN). IEEE, 2018.

[15] https://paperswithcode.com/dataset/coco

[16] Roters, Jan, Xiaoyi Jiang, and Kai Rothaus. "Recognition of traffic lights in live video streams on mobile devices." IEEE Transactions on Circuits and Systems for Video Technology 21.10 (2011): 1497-1511.

[17] Cheng, Ruiqi, et al. "Real-time pedestrian crossing lights detection algorithm for the visually impaired." *Multimedia Tools and Applications* 77.16 (2018): 20651-20671.