Available online at www.sciencedirect.com
ScienceDirect
ICT Express 6 (2020) 121–124
www.elsevier.com/locate/icte

Real-time Bhutanese license plate localization using YOLO

Yonten Jamtsho, Panomkhawn Riyamongkol*, Rattapoom Waranusast

Naresuan University, Phitsanulok 65000, Thailand

Received 6 September 2019; accepted 8 November 2019
Available online 20 November 2019

Abstract

The Automatic License Plate Recognition (ALPR) is one of the intelligent transportation systems which provides a safe and secure mode of transportation. In ALPR technology, recognition accuracy entirely depends on the performance of the localization phase. This paper presents the real-time Bhutanese license plate (LP) localization using YOLO (You Only Look Once). The vehicle detection was performed before the LP localization to eliminate the false positives generated by the signboards as they look similar to LPs. A single convolutional neural network gave an overall mean average precision of 98.6% with the training loss of 0.0231 for vehicle and LP.

Keywords: ALPR; YOLO; Bhutanese license plates; Vehicle detection; False positive

1. Introduction

Nowadays, Intelligent Transport System (ITS) is regarded as one of the best intelligent systems in the field of transport sectors. The main objective of the ITS is to provide a safe and secure model of transportation by the use of emerging technologies [1] to control traffics in cities and highway. Among many ITS technologies, the ALPR system plays a vital role in providing better services to the people whereby reducing traffic congestion and increasing the processing time. The ALPR system is mostly used in a automatic toll fee collection, managing parking space and cross-country border control. As the number of road users increases day by day around the world, there has been a problem of traffic congestion in cities and highways. Similarly, as Bhutan heads towards digitizing of transport sectors, it has become essential to use the ITS to automate the conventional way of managing traffic. Bhutan is the only country in the world that does not have single traffic lights. The police personnel physically check the LP details, which is time-consuming and need more human resources. Therefore, it has become utmost important to develop a real-time application to detect the LP for the development of an ALPR system for Bhutan.

The typical ALPR system consists of three stages: LP localization, character segmentation and character recognition. Among three stages, LP localization is one of the challenging tasks since the inaccurate localization will hamper the accuracy of the character segmentation followed by character recognition [2]. Therefore, the purpose of this study is to develop a system that localizes the Bhutanese LP after the detection of vehicle from the real-time video using the single convolutional neural network.

1.1. Bhutanese license plate

The ALPR technology tends to be region-specific with the variation of the LP. Similarly, the ALPR technology already deployed in another country does not work in the Bhutanese context due to different vehicle laws and orders. Apart from this, Bhutanese LPs vary from color to its size based on the type of ownership. There are eight types of Bhutanese LPs and some of them are shown in Fig. 1. The first upper portion contains the Bhutanese scripts and the second portion contains the actual LP number.

The remainder of this paper is organized as follows: Section 2 discusses the related work; Section 3 explains the proposed YOLO approach. The experimental results of the study are discussed in Section 4, followed by the conclusion in Section 5.

* Corresponding author.
E-mail addresses: yontenj61@email.nu.ac.th (Y. Jamtsho), panomkhawnr@nu.ac.th (P. Riyamongkol), rattapoomw@nu.ac.th (R. Waranusast).
Peer review under responsibility of The Korean Institute of Communications and Information Sciences (KICS).

https://doi.org/10.1016/j.icte.2019.11.001

Fig. 1. Sample license plates of Bhutan.

2. Related work

2.1. Vehicle detection

Vehicle detection is one of the problems related to object detection, where we intend to find the position of the objects. Vehicle detection from the videos is vital for the development of the self-driving car and also to design an intelligent traffic system [3]. To detect the vehicle from the video or image, many researchers have proposed methods like background subtraction [4], feature-based [5] and CNN [6]. In [4], the background subtraction method was proposed to extract the vehicle by subtracting the moving vehicle from the stored static image. The difference in the image was used as the vehicle regions after the thresholding. This method is not adaptive to changing lighting conditions. Trainable object detection from the images and videos were proposed in [5], where it was based on learning the labeled training data which extract the objects. The Region-CNN [7] method was proposed in [6] to identify the vehicle using the region proposal. This method cannot be deployed in the real-time application since it takes 47 s for an image to detect objects.

2.2. LP localization

For LP localization, many researchers had used traditional approaches like edge-based, color-based and feature-based methods. In edge-based, one of the distinguishing features of the LP is its rectangular shape with a known aspect ratio. Due to the color transition between the car body and the LP, the Vertical Sobel operator was applied to find the vertical edges in [8] and [9]. After finding the vertical edges, the width to height ratio was used to find the correct LPs. The edge-based approach is more sensitive to noise and unwanted edges [2]. In some country, the color information is used to differentiate the types of LPs. In a color-based approach, the color information is used to distinguish the colors of the car and LPs. For Korean LPs, Deb and Jo [10] proposed HSI color for the detection of the candidate regions and was verified by using position histogram. The HSI model is more sensitive to noise. The features using HOG algorithm with sliding window technique was proposed for Brazilian LPs with the SVM classifier to quantify the LP region.

2.3. You Only Look Once (YOLO)

YOLO is a single convolutional neural network that predicts the bounding boxes with the class probabilities from the single scan [11]. Instead of selecting the interesting regions from the image, YOLO takes the problem of object detection as the regression problem where the object detection and classification take place in a single neural network. This type of algorithm is mostly used in the real-time application. The original YOLO architecture consists of 24 convolutional layers, followed by two fully connected layers. In YOLO, the input image is divided into MxM grid cells. Grid cell that contains the center of the object is responsible for predicting the object. The output tensors of YOLO model will be a vector of $M*M*(B*5 + C)$ where B represents the predicted bounding boxes and the confidence score by each grid cell, C is the class probabilities for the predicting bounding boxes. Each bounding box B contains 5 components: bb_x, bb_y, bb_w, bb_h and the box confidence score C. The coordinates (bb_x, bb_y) represents the center of the object with respect to grid cell location and offsets (bb_w, bb_h) represents the width and height of the bounding box with respect to image dimensions. YOLO predict multiple bounding boxes per grid cell but those bounding boxes having highest Intersection Over Union (IOU) with the ground truth is selected, which is known as non-maxima suppression.

3. Proposed framework

3.1. System overview

The typical overview of the proposed system is shown in Fig. 2. The input to the system is a video frame and the output is localized LP. Before directly localizing the LP from the video frame, the vehicle detection has been performed first to eliminate the false positive. There is a chance that some objects like signboard might also get detected due to the presence of similar characteristics with the license plates. Therefore, the license plate is only extracted if it is enclosed inside the bounding box of a vehicle; otherwise, the bounding box information of the LP is discarded. The YOLO single convolutional neural network is used to achieve both the localization of the vehicle and the license plate at the same time.

3.2. Vehicle and LP localization

Since Bhutanese LPs contain different background and foreground colors, it is challenging to use traditional approaches like edge-based and color-based. These methods require good quality images and it is still expensive to deploy in the real-time scenario. License plate localization is one of the tough tasks since it has to tackle different types of external factors such as extreme weather conditions and different illumination. The Bhutanese license plate varies in shapes and sizes and can be confused with other similar shape in the frame. Therefore, the YOLO (version 2) based object detector is proposed to handle the different Bhutanese license plates.

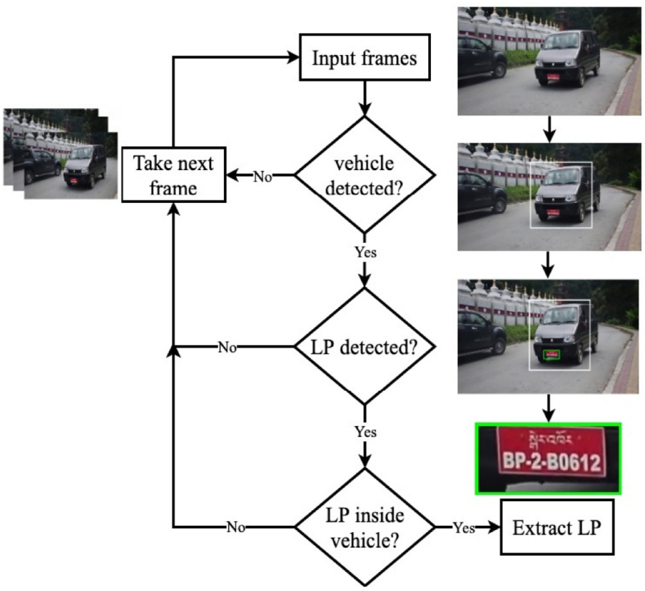

Fig. 2. System overview.

3.3. Data annotation

Before training the YOLO model, we start by generating the bounding box for each class. In this study, two classes (Vehicle and Plate) were annotated per image using the *RectLabel* [12] software. Since the *RectLabel* tool gives the actual starting coordinates ($\mathrm{x_{min}}, \mathrm{y_{min}}$) and the ending coordinates ($\mathrm{x_{max}}, \mathrm{y_{max}}$) for each bounding box, the bounding box offsets need to be normalized between 0 and 1 to fit into YOLO format (*x, y, w, h*) using Eqs. (1)–(4)

$$\mathrm{x} = \frac{(\mathrm{x_{min}} + \mathrm{x_{max}})}{2*\mathrm{W}} \tag{1}$$

$$\mathrm{y} = \frac{(\mathrm{y_{min}} + \mathrm{y_{max}})}{2*\mathrm{H}} \tag{2}$$

$$\mathrm{w} = \frac{(\mathrm{x_{max}} - \mathrm{x_{min}})}{\mathrm{W}} \tag{3}$$

$$\mathrm{h} = \frac{(\mathrm{y_{max}} - \mathrm{y_{min}})}{\mathrm{H}} \tag{4}$$

where W and H are the width and height of the image.

3.4. Algorithm for localization of LP inside the vehicle

In the typical ALPR, it is crucial to check the presence of a vehicle to reduce the number of false positives generated by the signboard. In this study, we will automatically localize the LP inside the enclosing vehicle bounding boxes using the single convolutional neural network. The YOLO model predicts center (center*X, centerY*) coordinates of the bounding box followed by boxes' width (*w*) and height (*h*). Therefore, Eqs. (5) and (6) was used to derive the top and left corner of the bounding box for both vehicle and plate.

$$\mathrm{x_{min}} = \mathrm{centerX} - (\mathrm{w}/2) \tag{5}$$

$$\mathrm{y_{min}} = \mathrm{centerY} - (\mathrm{h}/2) \tag{6}$$

After finding the starting coordinates of the bounding box, the ending coordinates (x_{max}, y_{max}) can be computed using the following Eqs. (7) and (8):

$$\mathrm{x_{max}} = \mathrm{x_{min}} + \mathrm{w} \tag{7}$$

$$\mathrm{y_{max}} = \mathrm{y_{max}} + \mathrm{h} \tag{8}$$

To check the presence of the LP inside the vehicle, the following pseudo code is used.

Input: *P and V are the 2D-array of bounding box coordinates*
  N *← length (V)*
  M *← length (P)*
  for *i ← 0 to N* ***do***
    for *j ← 0 to M* ***do***
      $p_{min} \leftarrow P[i][0];$ $q_{min} \leftarrow P[i][1]$
      $p_{max} \leftarrow P[i][2];$ $p_{min} \leftarrow P[i][3]$

      $x_{min} \leftarrow V[j][0];$ $y_{min} \leftarrow V[j][1]$
      $x_{max} \leftarrow V[j][2];$ $y_{max} \leftarrow V[j][3]$

      if $(p_{min} > x_{min}\ \&\ q_{min} > y_{min})\ \&\ (p_{max} < x_{max}\ \&\ q_{max} < y_{max})$ ***then***
        $PLATE \leftarrow (p_{min}, q_{min}, p_{max}, q_{max})$
        break

---

where:

- P and V represents the 2D arrays of bounding box coordinates for PLATE and VEHICLE
- (p_{min}, q_{min}) and (p_{max}, q_{max}) are the starting and ending coordinates of license plate
- (v_{min}, v_{min}) and (v_{max}, v_{max}) are the starting and ending coordinates of vehicle
- Since YOLO predicts only four coordinates, 0 to 3 indices are used to access the coordinates.

4. Experimental results

The datasets for the study were all generated from the video frame. The datasets consist of 1014 license plate images found in Bhutan. After the annotation of all the required classes and normalizing into YOLO format, the modified version of YOLOv2 Darknet [11] based on Alexey's implementation [13] was proposed. The YOLO model is already trained on 20 classes, which include vehicle, but in our proposed method, we intended to detect PLATE and VEHICLE classes using one convolutional neural network. We had to modify the number filters in the last layer to match the number of classes. The formula to calculate the filters is shown in Eq. (9).

$$\mathrm{Filters} = (\mathrm{C} + 5) * \mathrm{A} \tag{9}$$

where A is an anchor boxes which predicts 5 bounding boxes offset (A = 5) and, C is the number of classes. The number of filters was set to 35. Training of YOLO needs powerful graphical processing unit (GPU), Nvidia CUDA and cuDNN.

Table 1
Tabulation of Average Precision (AP) and average IOU for each epoch.

Epoch	Plate (%)	Vehicle (%)	Avg IOU
1000	98.25	99.01	81.75
2000	98.51	98.95	83.48
3000	98.04	98.79	83.61
4000	98.52	98.83	83.92
5000	**98.52**	**98.82**	**83.99**
6000	98.52	98.82	83.94
7000	98.52	98.73	83.93

Fig. 3. False positives generated by signboard.

(a) (b)

Fig. 4. (a) LP detected inside the vehicle, (b) No LP detected.

Since we are interested in two classes, the YOLO model was trained on Google Colaboratory for 7000 epochs with 64 batch size and the image dimensions of 608 × 608. Google Colaboratory is a free online cloud service that provides Tesla K80 GPU with 12 GB RAM and contains all the deep learning libraries. Table 1 shows the average IOU and precision for plate and vehicle classes generated by each epoch.

After 7000 epochs, the epoch with highest average IOU was selected as the weights to be implemented in the proposed real-time application. From Table 1, we can select the weights of 5000th iteration since it has 83.99% overlap of predicted bounding box with the ground truth. The overall mean average precision (mAP) was 98.6% with the training loss of 0.0231 for 64 batch size with 8 subdivision. According to darknet documentation [13], the model is said to be performing well if the average training loss is below 0.05. Figs. 3 and 4 show results obtained from the experiments.

In Fig. 4(a), the model has localized LP inside the vehicle bounding box, whereas in Fig. 4(b), the model has not predicted LP.

5. Conclusion

The purpose of this study was to solve the problem of Bhutanese license plates localization. The detection of the vehicle was done before the localization of license plates in order to eliminate the false positive. In the study, the single convolutional neural networks were proposed to handle different types of license plate found in Bhutan. The proposed method gave an overall mAP of 98.6% with 0.0231 training loss, which is best to be deployed in the real-time application. In future, a greater number of datasets need to use for the training.

Declaration of competing interest

The authors declare that there is no conflict of interest in this paper.

References

[1] P.I. Reji, V.S. Dharun, License plate localization: A review, Int. J. Eng. Trends Technol. 10 (13) (2014).
[2] S. Du, M. Ibrahim, M. Shehata, W. Badawy, Automatic license plate recognition (ALPR): A state-of-the-art review, IEEE Trans. Circuits Syst. Video Technol. 23 (2) (2013) 311–325.
[3] X. Hu, et al., SINet: A scale-insensitive convolutional neural network for fast vehicle detection, IEEE Trans. Intell. Transp. Syst. 20 (3) (2019) 1010–1019.
[4] L. Vasu, An Effective Step to Real-Time Implementation of Accident Detection System using Image Processing, Master of Science, Oklahoma State University, USA, 2010.
[5] C. Papageorgiou, A trainable system for object detection in images and video sequences, 2000.
[6] A.A. Yilmaz, M.S. Guzel, I. Askerbeyli, E. Bostanci, A vehicle detection approach using deep learning methodologies, 2018, ArXiv180400429 Cs.
[7] R. Girshick, J. Donahue, T. Darrell, J. Malik, Rich feature hierarchies for accurate object detection and semantic segmentation, 2013, ArXiv13112524 Cs.
[8] D. Zheng, Y. Zhao, J. Wang, An efficient method of license plate location, Pattern Recognit. Lett. 26 (15) (2005) 2431–2438.
[9] M. Sarfraz, M.J. Ahmed, S.A. Ghazi, Saudi Arabian license plate recognition system, in: 2003 International Conference on Geometric Modeling and Graphics, 2003. Proceedings, 2003, pp. 36–41.
[10] K. Deb, K. Jo, HSI color based vehicle license plate detection, in: 2008 International Conference on Control, Automation and Systems, 2008, pp. 687–691.
[11] J. Redmon, A. Farhadi, YOLO9000: Better, faster, stronger, 2016, ArXiv Prepr. ArXiv161208242.
[12] RectLabel - Labeling images for bounding box object detection and segmentation, 2019, RectLabel. [Online]. Available: https://rectlabel.com. [Accessed: 10-Apr-2019].
[13] Alexey, Windows and Linux version of Darknet Yolo v3 & v2 Neural Networks for object detection (Tensor Cores are used): AlexeyAB/darknet, 2016.