INTERNATIONAL JOURNAL OF INTEGRATED ENGINEERING
ISSN: 2229-838X e-ISSN: 2600-7916

IJIE

Vol. 16 No. 3 (2024) 100-106
https://publisher.uthm.edu.my/ojs/index.php/ijie

Detecting and Extracting Illegal Signs from Video

Nur Syakira Suhaimi[1], Vik Tor Goh[1]*, Timothy Tzen Vun Yap[2], Hu Ng[3]

[1] *Faculty of Engineering, Multimedia University, 63100 Cyberjaya, MALAYSIA*
[2] *School of Mathematical & Computer Sciences, Heriot-Watt University, 62200 Putrajaya, MALAYSIA*
[3] *Faculty of Computing and Informatics, Multimedia University, 63100 Cyberjaya, MALAYSIA*

*Corresponding Author: vtgoh@mmu.edu.my
DOI: https://doi.org/10.30880/ijie.2024.16.03.010

Article Info

Received: 27 November 2023
Accepted: 5 February 2024
Available online: 30 April 2024

Keywords

YOLO, frame extraction, optical character recognition, illegal signs

Abstract

This project focuses on developing an automated system to detect illegal signs in urban environments from videos. The system utilizes computer vision and machine learning techniques, specifically the YOLOv5 object detection framework, to accurately identify and locate illegal signs in video frames. It incorporates a verification process using Optical Character Recognition (OCR) to differentiate between legal and illegal signs based on the extracted text information. The system is designed as a user-friendly web application, allowing users to upload videos or images for analysis and receive comprehensive results. The system can achieve a detection accuracy of up to 78.6%. With this system, authorities can effectively manage and regulate illegal signs in urban areas, contributing to better urban landscapes.

1. Introduction

Illegal signs have long been a problem impacting both the community and law enforcement agencies. These signs are a nuisance and often advertise illegal services or businesses such as money lending, prostitution, and the sale of adult toys. The culprits are unperturbed by the consequences of being caught due to the low fines or penalties. Consequently, this problem continues to persist despite the authorities' various efforts to remove them, which include documenting the telephone numbers displayed on these signs and having them terminated.

However, the process of documenting the large volume of illegal signs is tedious and time-consuming. To address this issue, this project aims to develop a machine learning-based system that can efficiently assist the authorities in detecting and recognizing illegal signs in video recordings. Furthermore, by leveraging the widespread availability of mobile phones, the system can also be utilized by the public to contribute videos of illegal signs, thereby reducing the need for extensive manpower from the authorities. The focus lies in training a custom model using YOLOv5, an object detection framework, to accurately identify various types of illegal signs.

The system will automatically extract specific video frames containing illegal signs, eliminating the time-consuming and labor-intensive manual review process. Through performance evaluation, we seek to assess the system's accuracy and effectiveness in detecting and extracting illegal signs, providing valuable insights for potential improvements, and ensuring the system meets the desired standards of accuracy and efficiency.

By streamlining the detection and documentation of illegal signs, this system aims to assist authorities in saving valuable resources, reducing costs, and preserving the aesthetic integrity and safety of our cities. This system has the potential to revolutionize the identification, mitigation, and prevention of illegal signs, paving the way for vibrant and well-maintained urban environments.

2. Related Work

The most widely used system for object detection is YOLO-based methods. These approaches utilize the You Only Look Once (YOLO) family of object detection methods for real-time detection. YOLO divides the input image into a grid and predicts bounding boxes and class probabilities for objects within each grid cell. Papers like [1], [2], and [3] have demonstrated real-time processing capabilities. This is due to the fast detection speeds offered by YOLO-based methods, enabling real-time or near-real-time applications [4].

However, there are some disadvantages to this approach, such as low precision and accuracy in detecting small objects, as observed in [5] and [6]. In comparison, CNN-based methods have higher accuracy in detecting and classifying objects, such as washroom signs, traffic signs, and manga images, as shown in [7], [8], and [9]. However, the performance of CNN-based methods may be influenced by image quality, including lighting conditions or image noise.

Other methods employed include color segmentation, shape recognition, and neural network classification, as used in [10]. This approach offers the advantage of accurately detecting signs even in ambiguous cases and has low computational time. However, it struggles with non-persistent detection of signs in subsequent frames.

3. Methodology

This chapter provides an overview of the methodology employed in the project for the automated identification of illegal signs in uploaded videos. The approach involves utilizing machine learning techniques and frame extraction to train a custom model using YOLOv5. The custom model is designed to accurately detect and extract various types of illegal signs, enabling the development of a robust system for sign detection and verification.

3.1 Data Collection

The YOLOv5 model needs to be trained on a large dataset. This data collection process involves downloading photographs of illegal signs from the internet, manually creating images of illegal signs and modifying them to include authentic backgrounds and capturing photographs of any discovered illegal signs.

Furthermore, a few videos were captured for evaluation purposes. While a total of 16 videos were recorded for evaluation, 74 photos of illegal signs were collected for the dataset. Additionally, videos and images of road signs were obtained from the internet to test the system's ability to distinguish between legal and illegal signs. This resulted in a total of 7 road sign images and 11 road sign videos. Additionally, unrelated images and videos were gathered to assess the system's functionality, consisting of 4 images and 1 video.

3.2 Image Labelling and Augmentation

To perform this task, image labeling and augmentation are carried out online on a website called Roboflow. Since the project's objective is to detect only illegal signs, a single class named "illegal-signs" is set in Roboflow. Subsequently, 74 photos from the dataset are uploaded to the website and manually labeled by selecting the area in each image that contains the illegal signs.

The images are then pre-processed by automatically orienting and resizing them to a size of 640 by 640 pixels. Data augmentation is applied to expand the dataset, using shearing at angles of 8° horizontally and 30° vertically. As a result of this data augmentation method, the dataset has grown to a total of 140 images. The dataset is divided into three parts: 71% (99 photos) for training, 19% (27 images) for validation, and 10% (14 images) for evaluation.

3.3 Model Training

Model training is a crucial aspect of this project as it ensures accurate detection of illegal signs. YOLOv5 was selected as the model due to its proven ability to generate precise bounding boxes and the availability of extensive documentation for training and utilization. Google Colab was chosen as the training platform due to its convenient features such as pre-installed libraries, cloud storage, multi-device compatibility and access to free GPU, and TPU resources.

The training process involves feeding the labeled and augmented dataset file from Roboflow into the YOLOv5 training script in Google Colab. Batch size and epoch values are determined through trial and error. The batch size is chosen considering training time, memory usage, and accuracy trade-offs, where larger batch sizes increase time and memory requirements but may yield higher accuracy. Epochs represent the number of iterations required to train the model using the entire dataset, with an appropriate number of epochs necessary to avoid underfitting or overfitting issues.

The trained model is evaluated using an evaluation script to measure inference time and metrics such as mAP (mean average precision), as well as testing with a separate pool of images. If the evaluation results are unsatisfactory, adjustments are made by modifying batch size, epochs, or augmenting the dataset to improve

Int. Journal of Integrated Engineering Vol. 16 No. 3 (2024) p. 100-106

accuracy. If the evaluation meets expectations, the best-trained model is saved for future use. The chosen configuration for this project includes a batch size of 16 and an epoch value of 500, striking a balance between training time, memory usage, and achieving high-quality results.

3.4 Object Detection

The system utilizes the YOLOv5 model, a highly accurate and efficient deep learning framework, for object detection. Visual Studio Code is used to write the program, facilitating seamless integration with the YOLOv5 framework, while OpenCV is employed for frame extraction. Various parameters and options are employed during object detection, such as specifying the weights file, defining the input source, and enabling the saving of outputs such as bounding box coordinates, class labels, confidence scores, and cropped images.

The system's script employs the YOLOv5 model and subprocess functionality to perform object detection, setting parameters and flags to accurately identify objects of interest in uploaded videos. Important information, such as bounding box coordinates, class labels, confidence scores, and cropped images, is saved for further analysis and processing. A confidence threshold of 0.58 is set to determine valid detections, and objects surpassing this threshold are included in the final results. This comprehensive approach to object detection showcases the system's effectiveness across diverse domains.

3.5 Frame Extraction

The frame extraction process is a crucial step in the object detection pipeline implemented within the YOLOv5 model. It involves identifying and extracting frames from video recordings that contain objects of interest based on predetermined criteria such as bounding box area and confidence value. Each frame of the video is analyzed, and any cropped frames containing detections of illegal signs are saved in their respective folders. The dimensions of the bounding box provide insights into the size and shape of the detected object. The system also considers the confidence value and calculates the area of the bounding box as a quantitative measure of the object's size within the frame.

Throughout the frame extraction process, the system keeps track of the frame with the largest bounding box area and the highest confidence value encountered so far, achieved through conditional checks. It updates relevant variables if a new bounding box area surpasses the previous maximum, and the associated confidence value exceeds the current maximum confidence. Once all frames have been processed, the system identifies the frame with the highest confidence and largest bounding box area. If such a frame exists, it is extracted and saved. Additionally, a cropped version of the frame containing the object is also set aside, offering a closer view for detailed analysis or further processing, particularly for Optical Character Recognition (OCR) software.

3.6 Optical Character Recognition (OCR)

EasyOCR was chosen as the preferred Optical Character Recognition (OCR) tool due to its lightweight nature and extensive documentation. To accommodate the usage of both English and Malay languages, which are commonly found on illegal signs in Malaysia, the OCR was initially configured to read words from these two languages. By passing the cropped image as input, EasyOCR extracted the text from the image and stored it in a regular text file. This file serves as a reference for the extracted text from the illegal sign images, allowing for easy access and analysis of the OCR output.

3.7 Illegal Sign Verification

To determine whether a video contains an illegal sign, the system utilizes a whitelist of predefined words, which consists of words associated with legitimate signs commonly found in public spaces. Examples of these words include "Awas," "Selangor," "Jalan," "Bandaraya," and "Danger." The system employs a comparison process to identify if any of these legal words are present in the OCR result obtained from the image.

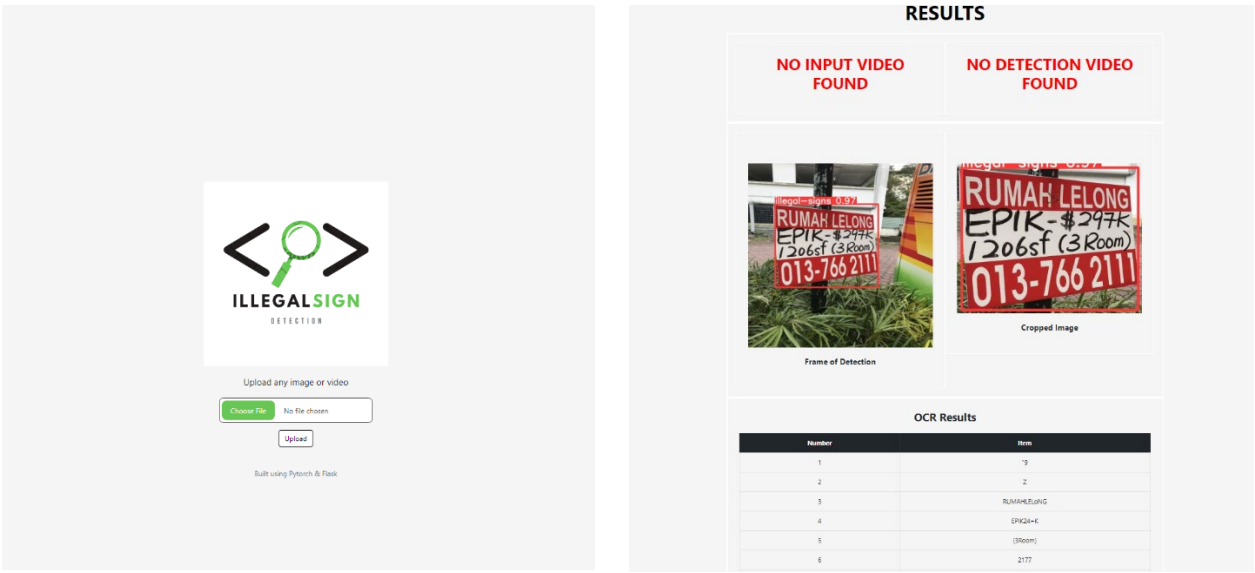
If a match is found, the loop breaks since there is no need to continue checking the remaining items. Consequently, the input is labeled as a non-illegal sign, and only the original video will be displayed on the result page. If none of the words in the whitelist are found, the input is labeled as an illegal sign. By employing this mechanism and comparing the OCR result against the whitelist, the system effectively distinguishes between illegal and non-illegal signs.

3.8 Web Application

To create a user-friendly interface for the system, the Flask framework was chosen. Flask allows for efficient development of web applications and provides fast results. The initial page shown in Fig. 1(a), serves as the entry point where users can upload a video file containing potential illegal signs. After submitting the file, the backend

initiates the object detection process, which involves executing the YOLOv5 model, extracting frames, performing OCR, and verifying illegal signs.

Upon completion of the object detection process, the system displays the results on the second page, shown in Fig. 1(b). This page summarizes the findings and presents various generated elements, including the original video, the detection video, the frame of detection, the cropped image, and the OCR results (if the input is a video). The integration of Flask and the web pages creates an interactive and accessible web application that enables users to upload files, perform object detection, verify illegal signs, and easily view and access the processed results.



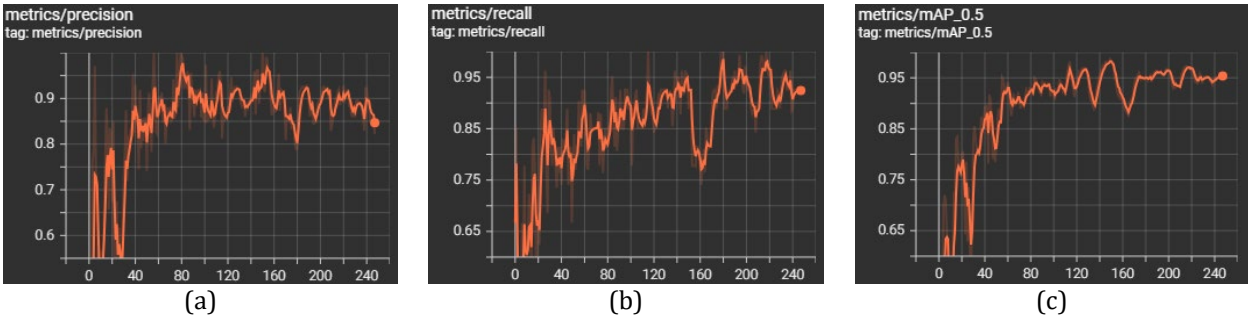
(a) (b)

Fig. 1 *User interface of web application (a) initial landing page; (b) output of sign detection*

4. Results and Discussions

4.1 Training and Evaluation of the Custom YOLOv5 Model

The trained model has achieved a precision of 92.8%, a recall of 96.1%, and an mAP50 of 97.7%. Fig. 2 displays the precision, recall and mAP50 graphs. These results are considered good enough, considering the small dataset size consisting of only 140 images. During the testing phase of YOLOv5, precision, recall, and mAP50 serve as important evaluation metrics for assessing the performance of the object detection model.



(a) (b) (c)

Fig. 2 *Graph showing (a) precision; (b) recall; (c) mAP50 of the trained model*

4.2 Sign Detection and Extraction Accuracy

To evaluate the system's overall accuracy, a test dataset consisting of 16 illegal sign videos, 11 non-illegal sign videos, and 1 non-related video was used. The test involved inputting the quantities of each scenario into the web application and observing the results to categorize them as true positive, false positive, false negative, or true negative. The evaluation results can be seen in Table 1 and Table 2.

Int. Journal of Integrated Engineering Vol. 16 No. 3 (2024) p. 100-106

Table 1 *Detection outcomes*

Scenarios	Number of Tested Items	True Positives	False Positives	False Negatives	True Negatives
Illegal Sign Videos	16	10	3	3	0
Non-Illegal Sign Videos	11	0	0	0	11
Non-related Videos	1	0	0	0	1
Total	**28**	**10**	**3**	**3**	**12**

Table 2 *Confusion matrix*

		Actual Values	
		Positive	**Negative**
Predicted Values	**Positive**	True Positive (TP) = 10	False Positive (FP) = 3
	Negative	False Negative (FN) = 3	True Negative (TN) = 12

Next, a confusion matrix will be used to help evaluate the system's precision, recall and accuracy. The formulas for these metrics are shown in Eq. 1, Eq. 2, and Eq. 3, respectively. Based on these equations, the values are calculated and presented in Table 3.

$$Precision = \frac{True\ Positive\ (TP)}{True\ Positive\ (TP)+False\ Positive\ (FP)} \quad (1)$$

$$Recall = \frac{True\ Positive\ (TP)}{True\ Positive\ (TP)+False\ Negative\ (FN)} \quad (2)$$

$$Accuracy = \frac{True\ Positive\ (TP)+True\ Negative\ (TN)}{True\ Positive\ (TP)+True\ Negative\ (TN)+False\ Positive\ (FP)+False\ Negative\ (FN)} \quad (3)$$

Table 3 *The performance of the illegal sign detection system*

Precision	Recall	Accuracy
0.769	0.769	0.786

The performance results are acceptable but can be further improved. One possible reason for the limited results shown in Table 3 is the small training dataset, which consists of only 140 images, including both real and synthetic images. We anticipate better performance by increasing the dataset size, and this will be the focus of our future work.

4.3 Character Recognition Accuracy

Character recognition accuracy is assessed by comparing the output of the OCR to the original document from which the same text was extracted. The number of correctly detected characters is counted over the total number of characters to determine the character-level accuracy. A total of 10 photos of illegal signs will be tested, including head-on and skewed images. The mean accuracy will be used to determine the overall accuracy of the system. Some of the test results are shown in Fig. 3.

For the sign in Fig 3(a), the OCR result is shown in Fig 3(b), with a total of 51 characters. Out of these, only 3 characters are correctly detected, resulting in an accuracy of 0.06. This low accuracy can be attributed to the sign being sideways, as the OCR can only read from left to right, not top to bottom. On the other hand, for the sign in Fig. 3(c), the OCR result is displayed in Fig 3(d), with a total of 26 characters. Among these, 18 characters are correctly detected, yielding an accuracy of 0.6923. By analyzing the data, the average accuracy of the OCR system can be calculated. Table 4 provides the accuracy of each character detection from each sign.

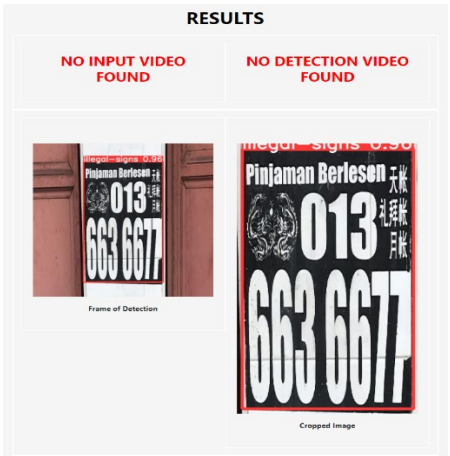
Table 4 *OCR accuracy result*

Sign Number	Accuracy
1	0.692
2	0.980
3	0.060
4	0.800
5	0.944
6	0.272
7	0.551
8	0.765
9	0.950
10	0.711
Average	**0.673**

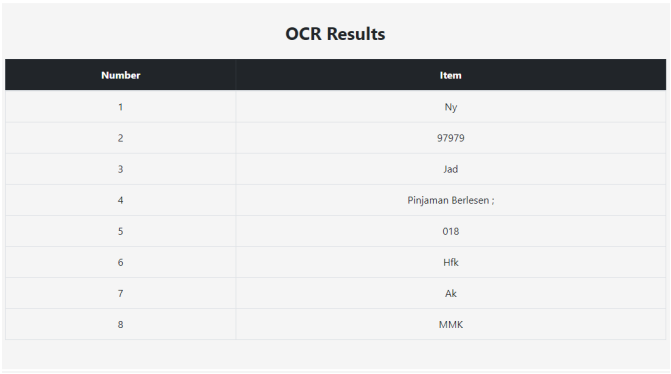
From the results in the Table 4, the average accuracy of the OCR system is determined to 67.3%. Some problems that contribute to this value include images with handwritten words, which are more challenging for the OCR library to read. Additionally, some images are not correctly oriented and appear blurry. However, most of the images are correctly read by the OCR due to frame extraction, which selects the highest-quality frame. Although character recognition is not the focus of this project, the results are satisfactory and acceptable.

(a)

(b)



(c)



(d)

Fig. 3 *Results of OCR*

5. Conclusion

The project has successfully developed an automated system that utilizes machine learning and frame extraction techniques to detect illegal signs. By employing YOLOv5 and OCR, the system accurately identifies and extracts frames containing illegal signs from videos. The incorporation of a verification process based on predefined legal words enhances the system's ability to distinguish between illegal and non-illegal signs. The evaluation results validate the system's effectiveness in accurately detecting illegal signs and extracting relevant information. Overall, this project showcases the potential of machine learning and YOLOv5 in addressing the challenge of illegal sign detection, providing a practical solution to enhance efficiency and resource management in sign detection and regulation.

Acknowledgement

The researchers sincerely appreciate and express gratitude for financial support from the Ministry of Higher Education, Malaysia, under the Fundamental Research Grant Scheme with grant number FRGS/1/2022/ICT07/MMU/03/1.

Conflict of Interest

Authors declare that there is no conflict of interests regarding the publication of the paper.

Author Contribution

The authors confirm contribution to the paper as follows: ***study conception and design:*** *Nur Syakira Suhaimi, Vik Tor Goh;* ***data collection:*** *Nur Syakira Suhaimi;* ***analysis and interpretation of results:*** *Timothy Tzen Vun Yap, Hu Ng;* ***draft manuscript preparation:*** *Nur Syakira Suhaimi, Vik Tor Goh. All authors reviewed the results and approved the final version of the manuscript.*

References

[1] Kuan-Ying, S., Ming-Fei, C., Po-Cheng, T., & Cheng-Han, T. (2022). Establish a dynamic detection system for metal bicycle frame defects based on YOLO object detection. *Proceedings of IET International Conference on Engineering Technologies and Applications (IET ICETA 2022)*, pp. 1–2.

[2] Ponika, M., Jahnavi, K., Sridhar, P. S. V. S., & Veena, K. (2023). Developing a YOLO based object detection application using OpenCV. *Proceedings of 7th International Conference on Computing Methodologies and Communication (ICCMC 2023)*, pp. 662–668.

[3] Jaison, B., Anjali, J. G., Jeevitha, J., & Devi, C. P. (2022). You Only Look Once (YOLO) object detection with COCO using machine learning. *Proceedings of IEEE International Interdisciplinary Humanitarian Conference for Sustainability (IIHC 2022)*, pp. 1574–1578.

[4] Madey, A. S. A., Yahyaoui, A., & Rasheed, J. (2021). Object detection in video by detecting vehicles using machine learning and deep learning approaches. *Proceedings of International Conference on Forthcoming Networks and Sustainability in AIoT Era (FoNeS-AIoT 2021)*, pp. 62–65.

[5] Zhang, J., Huang, M., Jin, X., & Li, X. (2017). A real-time Chinese traffic sign detection algorithm based on modified YOLOv2. *Algorithms*, 10(4), 127.

[6] Bayhan, E., Ozkan, Z., Namdar, M., & Basgumus, A. (2021). Deep learning based object detection and recognition of unmanned aerial vehicles. *Proceedings of 3rd International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA 2021)*, pp. 1–5.

[7] Chakraborty, D., & Chiracharit, W. (2020). Washroom sign detection using convolutional neural network in natural scene images. *Proceedings of 17th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON 2020)*, pp. 706–709.

[8] Jung, S., Lee, U., Jung, J., & Shim, D. H. (2016). Real-time traffic sign recognition system with deep convolutional neural network. *Proceedings of 13th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI 2016)*, pp. 31–34.

[9] Yanagisawa, H., Yamashita, T., & Watanabe, H. (2018). A study on object detection method from manga images using CNN. *Proceedings of International Workshop on Advanced Image Technology (IWAIT 2018)*, pp. 1–4.

[10] Broggi, A., Cerri, P., Medici, P., Porta, P. P., & Ghisio, G. (2007). Real time road signs recognition. Proceedings of IEEE Intelligent Vehicles Symposium (IEEE IV 2007), pp. 981–986.