Chapter 19
Passenger's Behavior Recognition System Using Computer Vision

Nurhayati Hasan and Muhd Khairulzaman Abdul Kadir

Abstract One of the problems that occurs inside public transport is that passengers often overlook or ignore public transport's rules such as in regards to eat and drink. Eating foods and drinks are not even allowed in public transport in avoiding excessive littering and drink spills which could potentially cause unwanted accidents. This system aims to recognize two actions such as eating and drinking through image processing in real-time environment. It also aims to classify and label the behavior of the passengers. This research is on a passengers behavior recognition system using computer vision (PBRSUCV) as a prototyping model. The method consists of image processing with a faster region-based convolutional neural networks (faster R-CNN) classification project that can be implemented in public transport to solve the stated problem. The system consists of a camera and laptop. The camera is used as a sensor to detect the targeted behaviors while the laptop will be the main device where every image processing takes place. The system was tested in real-time and is able to detect and label the eating and drinking behavior correctly with 99% accuracy on a single and two people in the image frame. It is certain that this system is capable to accurately recognize and classify the targeted behaviors in the public transport without any problem with the help of the faster R-CNN deep neural network.

Keywords Computer vision · Image processing · Recognition system

19.1 Introduction

Public transport is very important for Malaysian citizens as it makes us convenience to go anywhere and helps to reduce traffic congestion in the country. To maintain a delicate environment of the public transport, it is important to have rules in place. The rules which are often overlooked or ignored by public transport passengers are in regard to eating and drinking. Consuming food and drinks are not even allowed

N. Hasan · M. K. A. Kadir (✉)
Universiti Kuala Lumpur, British Malaysian Institute, Gombak, Selangor, Malaysia
e-mail: khairulzaman@unikl.edu.my

A. Ismail et al. (eds.), *Advanced Materials and Engineering Technologies*,
Advanced Structured Materials 162,
https://doi.org/10.1007/978-3-030-92964-0_19

in public transport to avoid excessive littering and drink spills that could potentially cause unwanted incidents and dirty environments.

This research intends to examine a high-definition real-time video of passengers in public transport to recognize their wrong behavior. To achieve such a goal, computer vision and image processing with faster R-CNN have been applied in this system. The objective of the system is to recognize two actions such as eating and drinking through image processing in real-time. It also aims to classify and label the wrong behavior on the passenger.

There are a lot of different computer vision techniques used in the past to identify the presence of human and its behavior. A study regarding extraction of human gait is developed using Haar Cascade classifier technique on OpenCV. It applies background subtraction to crop the unnecessary region of interests (ROI) in a video frame which are then transformed into silhouette form [1–3].

In a study to detect human carrying baggage from video sequences, several human body parts are divided such as head, body, leg, and baggage component. Each feature is then extracted and trained using a support vector machine (SVM) classifier [4].

Histogram of oriented gradient (HOG) feature set is used in the study to identify pedestrians. It applies different machine learning algorithms to ensure a successful detection and analyze the speed of the system. HOG performs better with SVM which gives almost linear separable features [5, 6].

19.2 Materials and Methods

The system's need to input data which requires thousands of images. Both images are taken from high-definition (HD) video recordings, which were then splits into thousands of frames to produce samples of the image. Figure 19.1 shows the flow of process to train the images.

Initially, a virtual environment was set up in Anaconda for TensorFlow-GPU using the anaconda command prompt to run as an administrator and install the other required packages by issuing commands.

Once the TensorFlow Object Detection API is all configured, then the images will be used to train a new classifier for detection which will be included.

To train a good classifier for detection, TensorFlow needs thousands of images of an object. The training images should have random entities in the image together

Fig. 19.1 Flow of image training process

with the target objects and should have a various range of backgrounds and lighting environments, to train a versatile classifier. After that, 20% of the images were moved to the test directory, and 80% of them to the training directory. This is to ensure smooth and accurate image processing in real-time mode.

Finally, it is time to generate the TFRecords with the images labeled, that serve as input data for the TensorFlow training model.

First, the.xml image data will be used to create.csv files that contain all the train and test image data. These generates file called 'train.record' and 'test.record'. This will continue to train the current classifier for object detection. If all is set right, TensorFlow will start the preparation. The setup will take 30 s before commencing the actual preparation [1, 7].

The flowchart as in Fig. 19.2 is designed to provide a better understanding of the PBRSUCV process. This program begins by gathering image samples of selected behaviors. After that, the image samples obtained will be trained using the deep neural network. During the system testing, the image obtained from the live stream would be compared to the training images.

If the observed behaviors are not the same as the qualified images, live videos will continue to be collected until a targeted behavior is observed. The cycle stops until the activity has been correctly identified and the public transit manager has been informed of the identification.

It is also important to determine the suitable hardware and software requirements to make sure the objectives of this research are achieved. Table 19.1 shows the hardware requirement used to build PBRSUCV while Table 19.2 describes the software information being applied.

19.3 Results

The system environment has been tested in real-time mode. The video is being analyze for its performance and accuracy of PBRSUCV detection.

For the first test, the process is done for a single person in real-time video and varying the distance of the subject from camera. The distance of the subject from camera was increased by 0.3 m for each step. After that, the percentage of detection accuracy was recorded and analyzed. The results for the detection are shown in Figs. 19.3, 19.4 and 19.5.

The percentage of accuracy is calculated based on the accuracy recorded on the labeled box of each eating person in the image frame as in Eq. (19.1).

$$\text{Accuracy percentage (\%)} = 100\left[\frac{\text{Total number of people eat or drinks in the frame}}{\text{Total number of people in the frame}} \times 100\right] \tag{19.1}$$

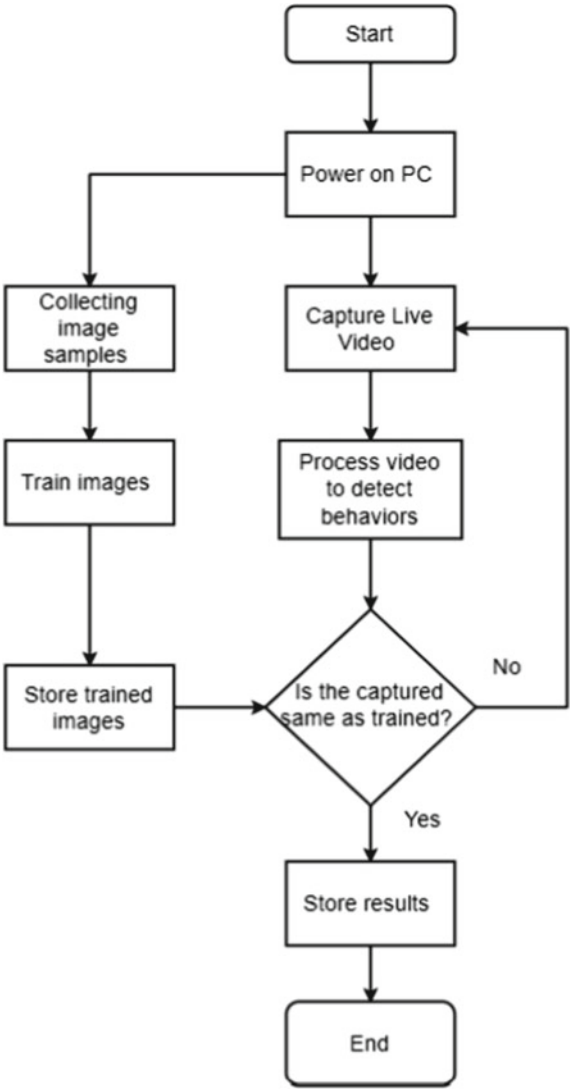

Fig. 19.2 Flowchart of PBRSUCV

The results of percentage accuracy are shown in Tables 19.3, 19.4, and 19.5 for one person, two and three people in a scene of real-time video and then cut into nine frames based on the different distance of the person from the camera.

Table 19.1 Hardware requirements

Hardware	Justification
Laptop	OS: Windows 10 CPU: Processor Intel Core i5 8th Gen GPU: NVIDIA GeForce GTX 1650 RAM: 8 GB DDR4 Storage: 256 GB SSD Display: Full HD (1920 × 1080)
Camera	Dual: 12 MP, f/1.8, 28 mm (wide) Features: Quad-LED dual-tone flash, HDR Video: 4 K@24/30/60 fps, 1080p@30/60/120/240 fps

Table 19.2 Software requirements [8, 9]

Software	Justification
Anaconda	Distribution of the Python language free and open source for scientific computation The release provides software for Windows, Linux and macOS suitable for data-sciences
CUDA	Enable application developers and professional engineers to use a graphics processing unit allowed by CUDA for general purposes
cuDNN	CuDNN provides finely optimized implementations for common protocols such as forward and backward convolution, pooling, normalization, and layers of activation
LabelImg	A graphical tool to annotate files. Annotations are stored in PASCAL VOC format as XML files, the format ImageNet uses
Tensorflow V1.13.1	An end-to-end open source machine learning tool. It has a detailed, dynamic wide range of capabilities, libraries, and community resources that enables researchers to push the state-of-the-art in ML, and developers to easily build and deploy ML powered apps
Python 3.5	A general-purpose, translated programming language

19.4 Discussion

For the results obtained in Tables 19.3, 19.4, and 19.5, the system is showing a high accuracy which is 99% for one and two people. The system also can detect up to 3 people in a time with almost 98% accuracy. However, the accuracy keeps reduced as the number of people and distance from camera is increased.

Currently, in this system the study shows that there is a limitation that can be achieved by using this method. From the experiment that has been conducted in the testing phase, the system recorded quite a low percentage of accuracy for several times with the increase of people in the image frame.

Fig. 19.3 The result of detection percentage for 1 person in real-time video

This limitation of this system can be avoided for future improvements. The accuracy of the detection could reduce due to some factors as follows:

i. The distance of subject from camera being too far away.
ii. Using too low resolution of camera.
iii. Low intensity of light.

Fig. 19.4 The result of detection percentage for 2 people in real-time video

iv. Too many people (crowded) in a place at one time so the detection can be overlapped.

Therefore, the system's efficiency and accuracy can be improved further by using a camera with higher quality and increase the number of samples for the training data.

Fig. 19.5 The result of detection percentage for 3 people in real-time video

19.5 Conclusion

In conclusion, this PBRSUCV is capable to detect and classify the behavior of eating and drinking through image processing in real-time mode. The result of accuracy percentage for the detection also showing high precision and validity rather than errors.

The system can detect people conducting eating or drinking action up to four people at a time with high accuracy by appointing a green box around the mouth,

Table 19.3 The percentage of accuracy result for one person eating in a scene

Distance of people from camera (m)	Total accuracy percentage recorded in a frame	Total number of people eating in a frame	Accuracy percentage (%)
0.3	99	1	99
0.6	99	1	99
0.9	99	1	99
1.2	93	1	93
1.5	80	1	80
1.8	77	1	77
2.1	67	1	67
2.4	0	1	0
2.7	0	1	0

Table 19.4 The percentage of accuracy result for two people eating in a scene

Distance of people from camera (m)	Total accuracy percentage recorded in a frame	Total number of people eating in a frame	Accuracy percentage (%)
0.3	196	2	97.5
0.6	99	1	99
0.9	99	1	99
1.2	89	1	89
1.5	81	1	81
1.8	77	1	77
2.1	77	2	38.5
2.4	0	2	0
2.7	0	1	0

Table 19.5 The percentage of accuracy result for three people eating in a scene

Distance of people from camera (m)	Total accuracy percentage recorded in a frame	Total number of people eating in a frame	Accuracy percentage (%)
0.3	99	1	98
0.6	99	1	95
0.9	99	1	93
1.2	93	1	90
1.5	80	1	86
1.8	77	1	82
2.1	67	1	0

food and hand gesture with label and percentage of accuracy for each person. While the percentage of accuracy is poor for 5–6 people, it can still be increased to the next point by increasing the number and quality of samples for training images.

Generally, errors consist only of 4–6 people, but for single and two people, the system is capable of being detected and classify with 99% precision. It is also found that the resolution of the camera and intensity of light can affect the result of detection other than the number of people in a frame and the distance of camera from the subjects. Therefore, the type of camera used to construct this system is very important in making sure the reliability and efficiency of the system's function.

The system can be updated to the next level making the system more robust and easier for applications to come. The current system can detect accurately only up to four people at a time.

For future enhancement, the efficiency of the system can be increased by adding more samples of training images with high quality. Currently, about 3000 samples of images were used to create the recognition system. Therefore, the samples can be increased up to 5000 images in the future.

Next, the specification of hardware used is also very important in building this PBRSUCV. A high specification of processing machine can speed up the process of training images which can save a lot of time. A higher specification type of camera also plays an important role in the detection process where the better the quality of the real-time video, the higher the accuracy of recognition can be obtained.

Finally, this PBRSUCV also can be upgraded by including the Internet of Things (IoT) platform and alert system. The current function of the system only can detect the eating and drinking behavior which is displayed on the screen. Therefore, these upgrades can give extra features to the detection system where all the data captured can be recorded and analyzed from time to time regularly.

References

1. Ismail AP, Tahir NM (2017) Human gait silhouettes extraction using Haar cascade classifier on OpenCV. In: UKSim-AMSS 19th international conference on computer modelling and simulation (UKSim). https://doi.org/10.1109/uksim.2017.25
2. Piccardi M (2004) Background subtraction techniques: a review. In: 2004 IEEE international conference on systems, man and cybernetics, vol 4 (IEEE Cat. No. 04CH37583). The Hague, pp 309–3104. https://doi.org/10.1109/ICSMC.2004.1400815
3. Abdul KMK (2018) Tracking and indexing moving object in multitude environment. M.S. thesis. Electrical Engineering, Universiti Teknologi Malaysia
4. Wahyono JKH (2017) Detection of human carrying baggage from video sequences. J Intell Fuzzy Syst 32(2):1601–1613. https://doi.org/10.3233/jifs169153
5. Ardiyanto I, Adji TB, Asmaraman DA (2018) On comprehensive analysis of learning algorithms on pedestrian detection using shape features. J Intell Fuzzy Syst 35(4):4807–4820. https://doi.org/10.3233/jifs-18491
6. Dalal N, Triggs B (2005) Histograms of oriented gradients for human detection. In: 2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05), vol 1. San Diego, CA, USA, pp 886–893. https://doi.org/10.1109/CVPR.2005.177

7. Abadi M, Agarwal A, Barham P et al (2020) TensorFlow: large-scale machine learning on heterogeneous distributed systems. http://download.tensorflow.org/paper/whitepaper2015.pdf. Accessed 12 June 2020
8. TensorFlow (2020) Anaconda documentation. https://docs.anaconda.com/anaconda/user-guide/tasks/tensorflow/. Accessed 12 June 2020
9. NVIDIA (2020) NVIDIA CUDNN documentation. https://docs.nvidia.com/deeplearning/sdk/cudnn-developer-guide/index.html. Accessed 28 June 2020