

Car detection using OpenCV

Leyla MURADKHANLI

Baku Higher Oil School, Baku, Azerbaijan
E-mail: leyla.muradkhanli@bhos.edu.az

Abstract. The article provides an overview of car detection using OpenCV, a popular open-source computer vision library. It covers the different steps involved in car detection, including object detection using Haar Cascade classifiers, feature extraction, classification and tracking. The article highlights the usefulness of car detection systems in fields such as traffic monitoring, security systems, and autonomous driving.

Keywords: *computer vision; machine learning; car detection; OpenCV; Haar Cascade classifier*

1. INTRODUCTION

Car detection is an active area of research and development in the field of computer vision and artificial intelligence. Car detection using computer vision technique involves automatically detecting and locating cars in images or video frames. Car detection has many applications, such as traffic control, autonomous driving, and security and surveillance systems.

Car detection algorithms typically involve using machine learning techniques, such as Haar Cascades, histogram of oriented gradients (HOG), or convolutional neural network (CNN), to train a classifier on a large dataset of positive and negative images [1, 2]. Positive images contain cars, while negative images do not. The trained classifier is then used to detect cars in new images or video frames.

In this article OpenCV based on Haar Cascade classifier was used to detect cars in videos.

2. HAAR CASCADE CLASSIFIER FOR CAR DETECTION

Car detection refers to the process of using machine learning algorithms to automatically detect and locate vehicles in images or videos. This is an important task in many applications, including autonomous driving, traffic control, and surveillance.

Car detection algorithms typically use a combination of image processing and machine learning techniques. One common approach is to use a Haar Cascade classifier, which is a machine learning algorithm that can be trained to recognize specific object patterns, such as the shape and features of cars.

Haar Cascade is a machine learning object detection algorithm used to identify objects in images and videos. It was developed by Viola and Jones in 2001 and is based on the Haar wavelet transform.

The Haar Cascade algorithm works by training a classifier on a large dataset of positive and negative images. Positive images contain cars, while negative images do not. The algorithm uses the positive and negative images to learn the features that distinguish cars from the background.

Once the classifier is trained, it can be used to detect cars in new images or video frames. This is done by sliding a window over the image and applying the classifier to each sub-region of the image. If the classifier detects a car in a sub-region, it is labeled as a positive detection.

3. CAR DETECTION STEPS

Car detection involves several steps (Figure 1):

1. Preprocessing

The first step is to preprocess the input image or video frame. This may involve resizing the image, converting it to grayscale, and applying filters to enhance the image quality.

2. Object detection

The next step is to apply a car detection algorithm to the preprocessed image. One popular algorithm for car detection is the Haar Cascade classifier, which can be trained to detect cars in images.

3. Postprocessing

Once the car detection algorithm has been applied, postprocessing steps may be applied to refine the results. This may involve removing false positives or performing additional filtering to improve the accuracy of the detections.

4. Visualization

Finally, the detected cars can be visualized in the original image or video frame, along with any additional information such as bounding boxes or labels.

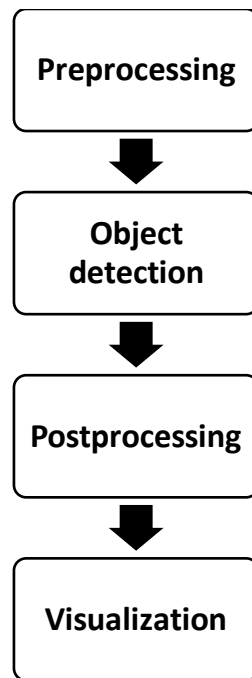


Fig. 1. Car detection steps

4. IMPLEMENTATION IN PYTHON

OpenCV is a popular open-source computer vision library that provides a wide range of tools and functions for image and video processing. It includes built-in support for Haar Cascade classifiers, making it easy to implement car detection using this algorithm. OpenCV also provides tools for postprocessing and visualization, making it a powerful tool for car detection applications.

There are the following steps in OpenCV Python program to detect cars in video frame :

Step 1. Import libraries of python OpenCV

```
import cv2
```

Step 2. Capture frames from a video

```
cap = cv2.VideoCapture('video.mp4')
```

Step 3. Create cascade classifier.

Trained vehicle.xml classifier describes some features of some object we want to detect.

CascadeClassifier function, the predefined function of OpenCV, to train the images from the pre-trained XML file (Cascade file – car). `car_cascade = cv2.CascadeClassifier('cars.xml')`

Step 4. Running loop to detect

loop runs if capturing has been initialized.

Reads frames from a video

```
ret, frames = cap.read()
```

Step 5. Convert to gray scale of each frames

```
gray = cv2.cvtColor(frames, cv2.COLOR_BGR2GRAY)
```

Step 6. Detects cars of different sizes in the input image

Detect multiple objects, i.e. cars, using detectMultiScale.

```
cars = car_cascade.detectMultiScale(gray, 1.1, 1)
```

Step 7. Draw a rectangle in each cars

Step 8. Display frames in a window

Figure 2 and Figure 3 show car detection results of video1 and video2.

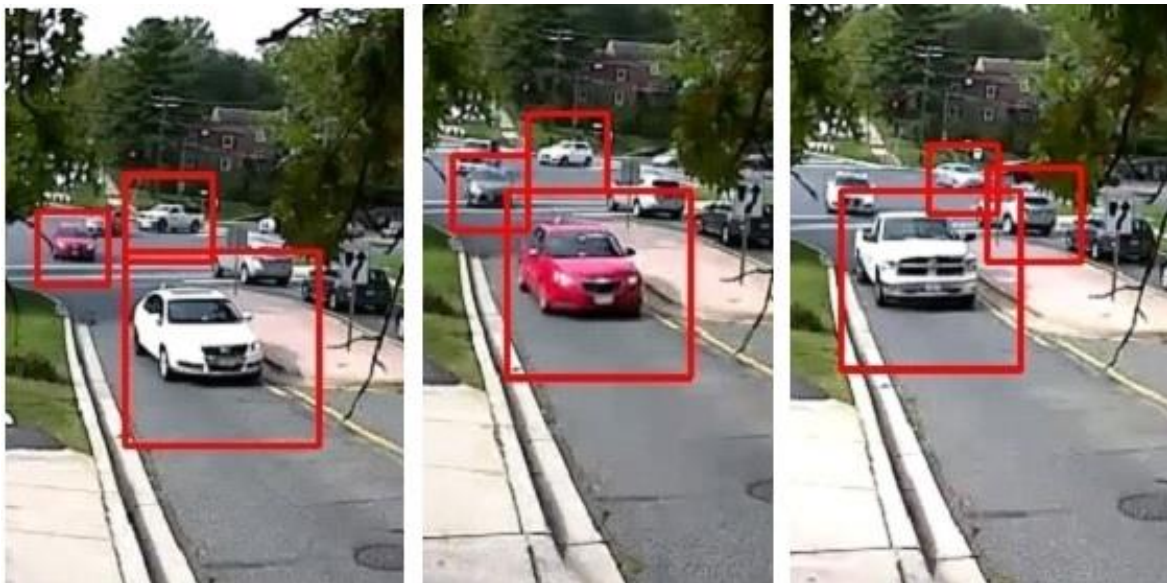


Fig. 2. Car detection result of video1

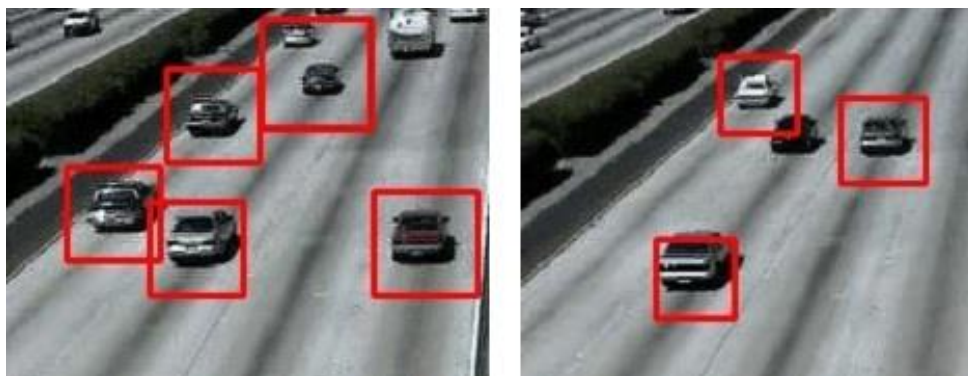


Fig. 3. Car detection result of video2

5. CONCLUSION

This article proposes car detection method based on Haar Cascade classifier using OpenCV. By combining object detection, feature extraction, classification, and tracking techniques, highly accurate and reliable car detection systems can be developed using OpenCV. These systems have a wide range of applications in fields such as traffic monitoring, security systems, and autonomous driving.

Car detection using computer vision is a well-researched topic with a wide range of applications. Car detection can be used in the following applications :

- Security and surveillance

Car detection can be used in security and surveillance systems to monitor parking lots, streets, and highways for any suspicious activity.

- Traffic control and management

Car detection can help in managing traffic flow and reducing traffic congestion in urban areas. It can also be used to detect and prevent traffic violations such as illegal parking, wrong-way driving, and over-speeding.

- Autonomous driving

Car detection is a crucial component of autonomous driving systems. It can be used to detect other vehicles on the road and help in collision avoidance.

- Parking assistance

Car detection can be used to assist drivers in parking their vehicles, by detecting empty parking spaces and guiding the driver to park in the correct location.

- Transportation and logistics

Car detection can be used in transportation and logistics industries to track the movement of vehicles and monitor the delivery of goods.

Car detection has significant potential to improve the efficiency and safety of transportation systems, reduce traffic congestion, and enhance security and surveillance in public spaces. There is still ongoing research into improving the accuracy and robustness of car detection systems.

References

1. Jinlong Chen, Yiming Jiang, Minghao Yang, " Research on Vehicle Detection Algorithm based on Haar-like and CNN " Proceedings of 2020 the 10th International Workshop on Computer Science and Engineering (WCSE 2020), Shanghai, China, 19-21 June, 2020, pp. 125-131.
2. Miftahul Hasanah, Gulpi Qorik Oktagalu Pratamasunu, Ratri Enggar Pawening "Automatic Car Detection Using Haar Cascade Classifier and Convolutional Neural Network for Traffic Density Estimation", Indonesian Journal of Artificial Intelligence and Data Mining (IJAIDM), Vol. 4, No. 1, March 2021, pp. 11 – 18.