Visual Computing : Assignement 2

Mohamed Kerroumi
mohamed.kerroumi@student.ecp.fr

In this assignment, I implemented a system that outputs bounding boxes (bb) of pedestrians using classic methods of computer vision (No Deep Learning).

The approach I used to address the problem of pedestrian detection is first of all background subtraction to remove the static part in the sequence of images, then I combined contour detection and a pre-trained HOG + Linear SVM model of OpenCv to detect pedestrian, I will detail each step in the following sections.

1. Approach

1.1. Background Subtraction

Background subtraction is a major preprocessing step in many vision based applications and since the input images used in the assignment are taken by a static camera, I thought it might be very useful for pedestrian detection to extract the moving foreground from the static background. Several algorithms were introduced for this purpose. I used BackgroundSubtractorMOG2 function available on OpenCV to do this task, MOG2 is a Gaussian Mixture-based Background/Foreground Segmentation Algorithm.

Figure 1: Input image before and after background subtraction

1.2. Histogram of Oriented Gradients and Linear SVM

The Histogram of Oriented Gradients descriptor is heavily used in object recognition tasks, Dalal and Triggs suggested in their seminal 2005 paper the Histogram of Oriented Gradients for Human Detection, they demonstrated that the Histogram of Oriented Gradients (HOG) image descriptor and a Linear Support Vector Machine (SVM) could be used to train highly accurate human detectors. OpenCV provides a pre-trained HOG + Linear SVM model that can be used to perform pedestrian detection in both images and video streams. I applied this model on input images after background subtraction to find bounding boxes which may contain humans, I ran into the problem of detecting multiple bounding boxes around the same object. So I applied non-maximum suppression to remove redundant and overlapping bounding boxes, as seen on the figure 2.

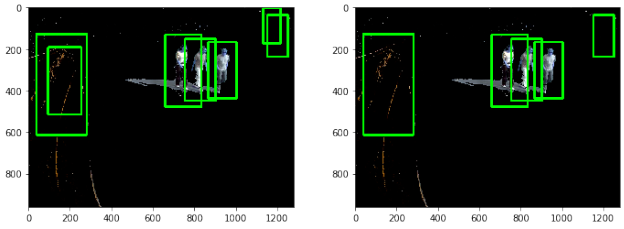

Figure 2: Image before and after non-maximum suppression.

1.3. Contour Detection

As you can see on figure 2, when we applied HOG + Linear SVM model on images without background to find bounding boxes with pedestrian, we detected some noisy boxes that didn't contain any human, so one way to suppress those boxes is to use contour detection.
The idea is to use the function **findContours** in OpenCV to detect contours in the image without background, and to keep only contours with an $Area > 3$ because those contours are more probable to contain human bodies, then we draw bounding boxes on those contours, an example of this is shown in the figure below.

We found the list of bounding boxes corresponding to each input image.
Then for a given input image, for each bounding box detected by the HOG + SVM model, we kept this bounding box only if it overlapped with at least one of the bounding boxes returned by the contour detector for this image, this

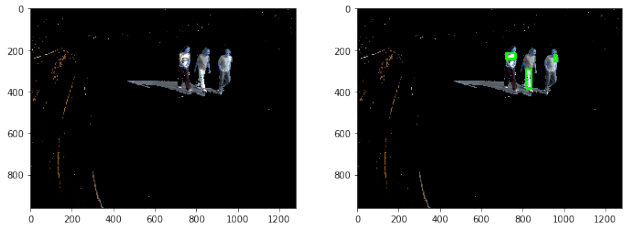

Figure 3: Contour detection on image without background

way we removed the negative bounding boxes. the results of applying this approach are below :

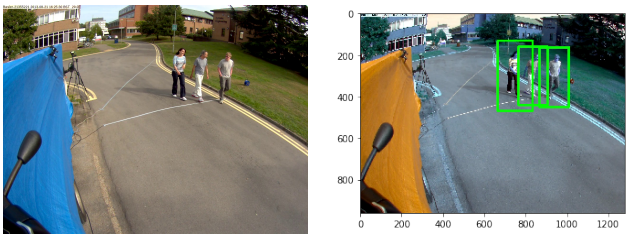

Figure 4: Input image with final bounding boxes

2. Results and Conclusion

After applying this approach to all the input images, I used as stated in the assignment the **Intersecion Over Union** as a metric to evaluate the accuracy of the approach I used to detect bounding boxes of pedestrian. I got $IoU = 19.48\%$, I could further increase this metric if I trained the SVM classifier on the sequence of images given for the assignment and also if I found the best hyper-parameters (threshold of the probability of bounding boxes to keep for the SVM classifier , threshold of the area of contours that may contain humans), but the code takes 2h to run, so due to lack of time I kept this model.

References

[1] *Histograms of Oriented Gradients for Human Detection.* Navneet Dalal and Bill Triggs, *INRIARhàone-Alps*. (2005)

[2] *An Improved Adaptive Background Mixture Model for Real-time Tracking with Shadow Detection.* P. KaewTraKulPong and R. Bowden , *Vision and Virtual Reality group*. (2001)