

PROJECT REPORT

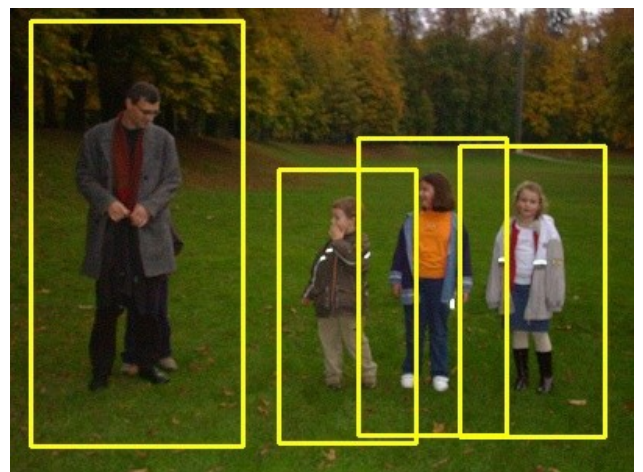
1. PEDESTRIAN DETECTION WITH INPUT AS INDIVIDUAL IMAGES

In this part of the project we have implemented pedestrian detection on a **set of images** and stored the outputs in a folder. We have taken a total of **27 images** as input. The input images are stored in the '**images**' folder. Upon running the code, a folder named '**output**' is automatically created which would hold the output images. For this part we have use **HOG + Linear SVM** inbuilt model, which gives pretty good results.

Steps followed:

- Loading of images.
- Fitting the model and setting up the hyperparameters.
- Getting the locations of the pedestrian in the image.
- Using ***non_max_supression*** method to ensure that there are no overlapping bounding boxes for the same pedestrian.
- Drawing the bounding boxes on the images.
- Making an 'output' folder if it doesn't exist.
- Saving the output images.

SAMPLE SOLUTION



INPUT

OUTPUT

2. REAL-TIME PEDESTRIAN DETECTION WITH WEBCAM

The model used for this is same as before, HOG + Linear SVM. We have just used the webcam's video feed as the input. A video is after all a collection of images put together. Steps followed:

- **VideoStream** library used to get the input video feed.
- Video was read frame by frame and searched for pedestrian.
- Drawing of bounding boxes along with non_max_supression.
- Obtained images shown as output one after another to give a sense of a video in real-time.

SAMPLE SOLUTION SCREENSHOT

