Assignment 4
Face Detection Using Viola-Jones Detector

April 12, 2024

Simone Vaccari
915222

Alessio De Luca
919790

1 Objective

The primary goal of our experiment is to develop a face detection system. This involves training a cascade object detector using positive instances (images containing faces) and negative instances (images without faces).

2 Procedure

In order to perform our task, we were initially given a training set containing 6713 images of faces (positive examples) and 274 negative examples, i.e. images in which no face appears.

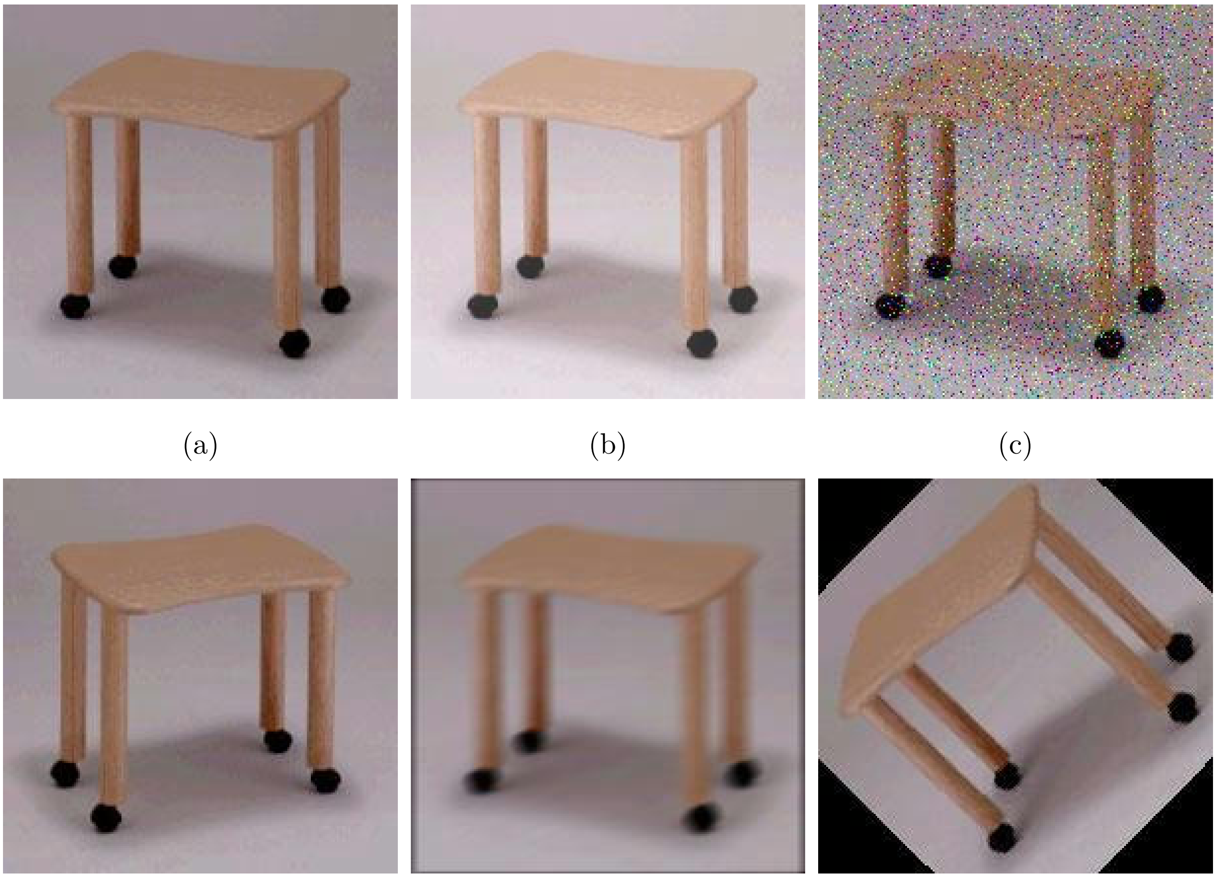

Figure 1: example of negative sample (left) and positive sample (right)

Given the very low number of negative examples with respect to the positive ones, the first step that needs to be performed is the **augmentation** of the negative class. We therefore applied different transformations to every non-face image, in order to generate new training images.

These included:

- Vertical mirroring (1 new image per original negative sample)
- Horizontal mirroring (1 new image per original negative sample)
- Rotation of random degree (10 new images per original negative sample)
- Brightness adjustment (6 new images per original negative sample)
- Salt & Pepper noise addition (6 new images per original negative sample)
- Camera motion addition (6 new images per original negative sample)

Examples of such transformations can be seen in Figure 2, where a random image of negative class has been selected.



(a) (b) (c)

(d) (e) (f)

Figure 2: Representation of: (a) original image, (b) brightened, (c) Salt & Pepper noised, (d) vertical mirrored, (e) motion blurred, (f) rotated

With this preliminary step, we were able to extend the negative class to about 8500 examples.

At this point we could train our **Viola-Jones detector**. This type of detector combines the following concepts:

- Haar-like features, e.g. two-rectangle, three-rectangle, four-rectangle features;
- Integral images, an intermediate representation that allows faster computations;
- AdaBoost algorithm;
- Cascade classifier, a multi-stage classifier that consists of multiple AdaBoost classifiers, that consequently evaluate an input, forwarding it only if it's considered positive by that classifier;

To train the detector we set the following parameters:

- *NegativeSamplesFactor* equal to 2, specifying that the number of negative images used at each stage is twice the number of positive images;
- *NumCascadeStages*, which determines the number of training stages, set equal to 10;
- *FalseAlarmRate* equal to 0.01, specifying the fraction of negative training samples incorrectly classified as positive ones;
- *TruePositiveRate* (the minimum fraction of correctly classified positive training samples) equal to 0.99;
- HOG (histogram of oriented gradients) as *FeatureType*, since using Haar-like features necessitate a large amount of memory;

Once trained, the detector was applied to a test set of 130 images, and detected faces were annotated using bounding boxes. The results were then evaluated against ground truth data to compute average precision.

3 Results

When training our detector we notice an increase in the amount of time required at each training stage. This is due to the fact that the complexity of the considered features increases in later stages, leading to more computationally expensive classifiers.

To visualize the precision of our detector we plotted the precision-recall curve, reported in Figure 3, showing the tradeoff between precision and recall (sensitivity) for different thresholds. The resulting average precision, computed using an overlap threshold of 0.2, is of 0.68.

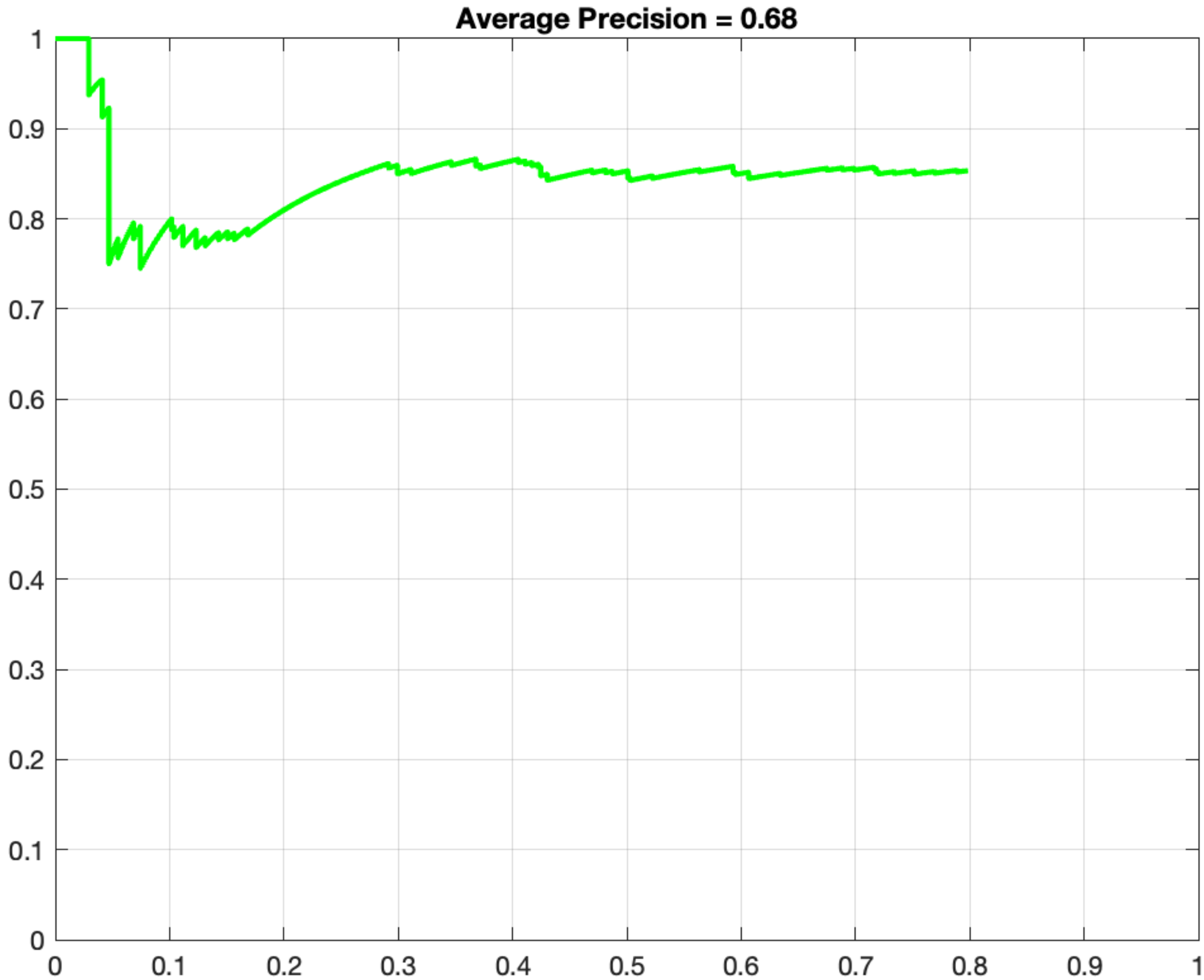


Figure 3: Precision-Recall curve with Average Precision

In the following figures we can visualize in yellow the detections made by our detector in the test images, while in green we highlighted the true faces to be detected.

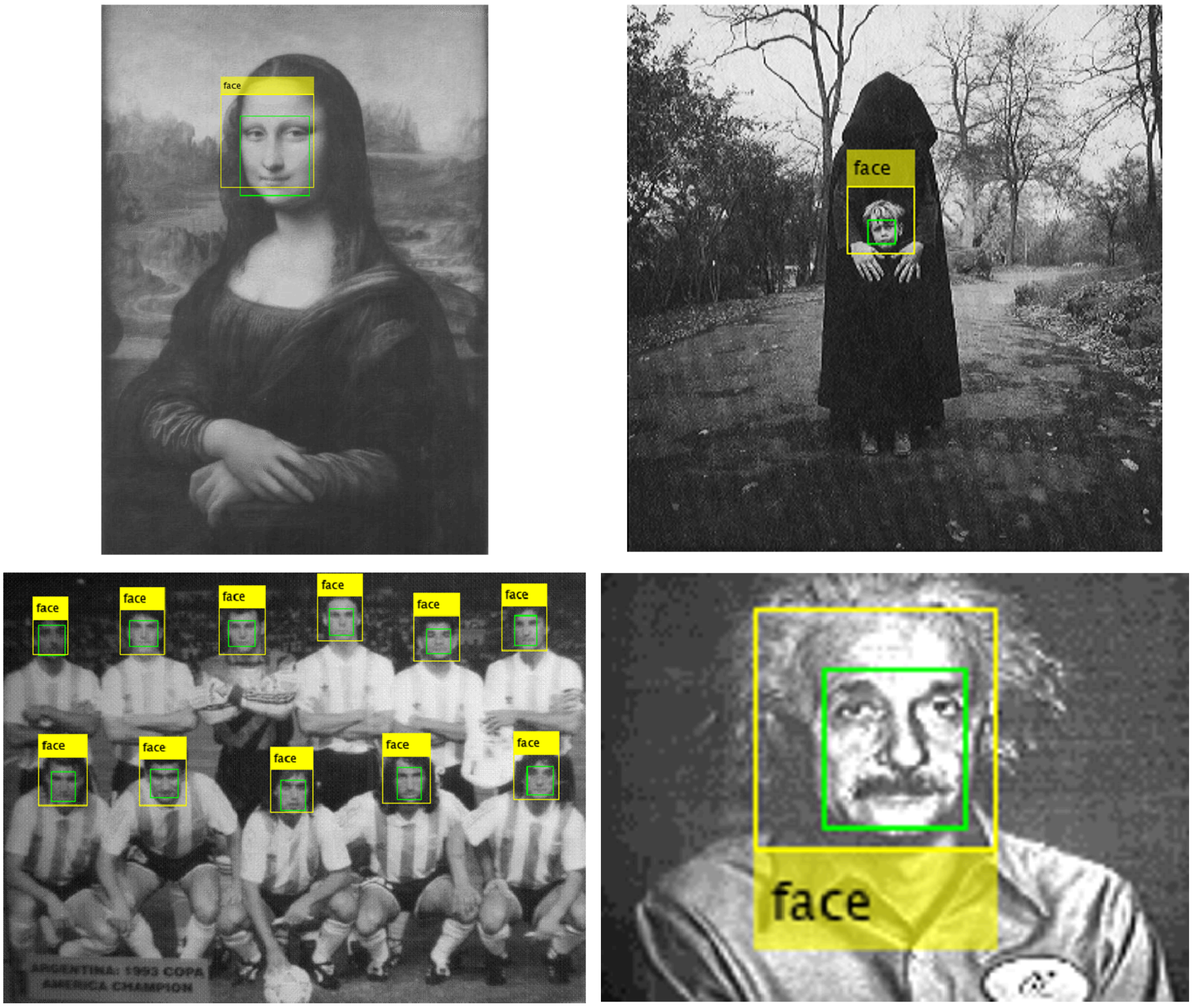


Figure 4: Images with 100% true faces detected

Figure 5: Images with most true faces detected and False Positives or False Negatives

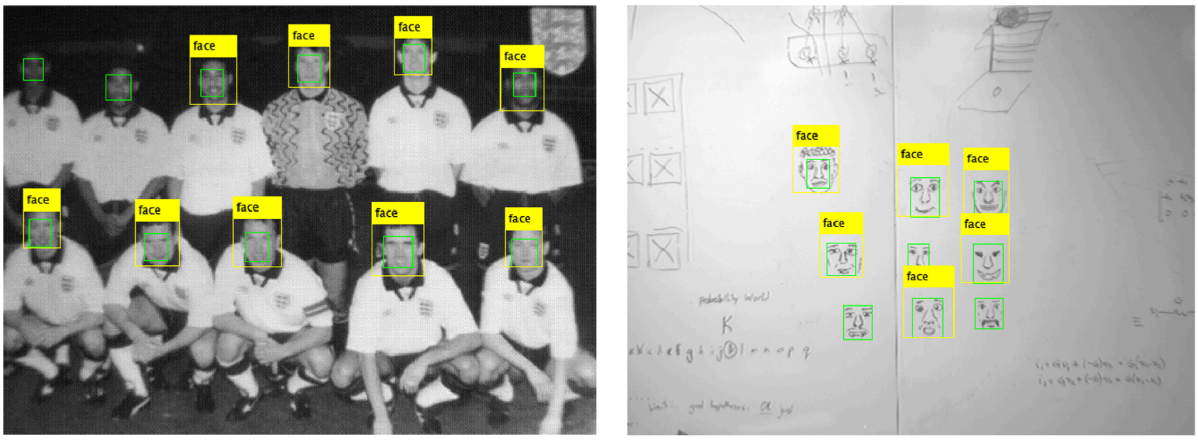


Figure 6: Images with some true faces not detected

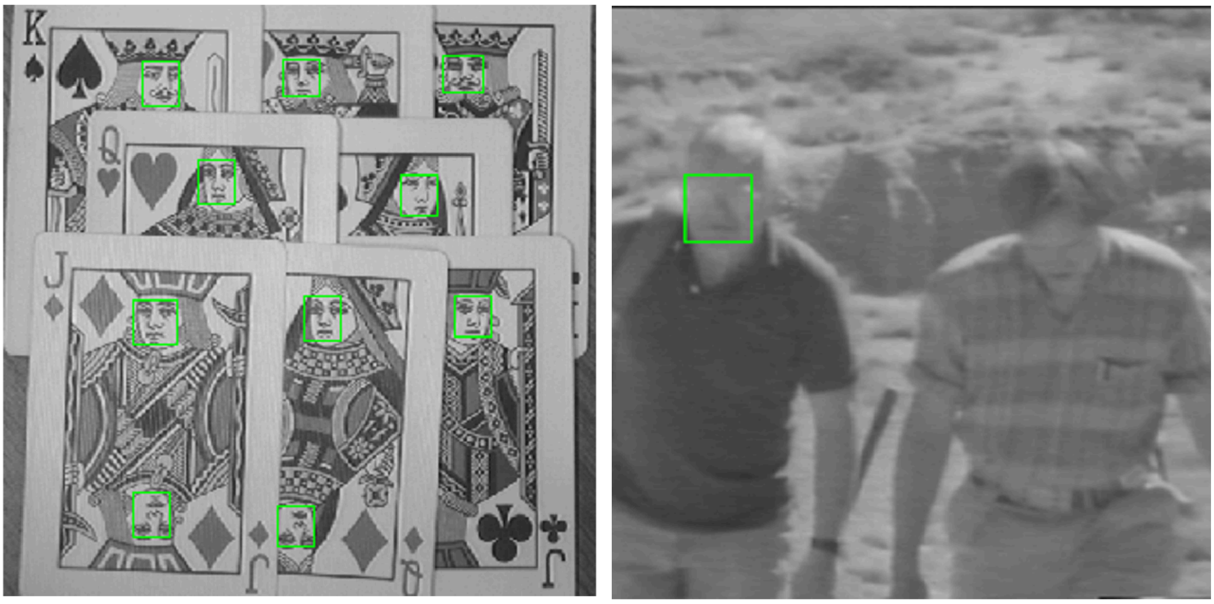


Figure 7: Images with no true faces detected

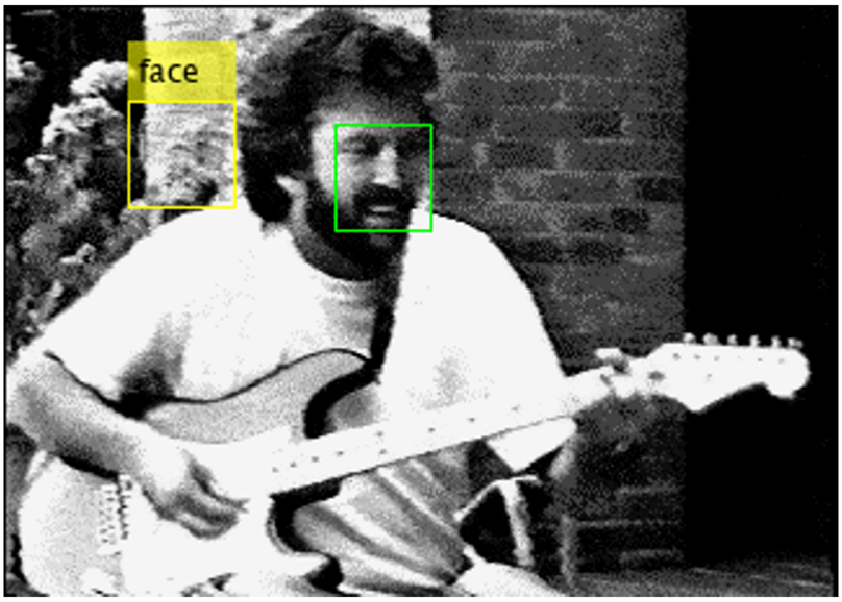


Figure 8: Image with profile face not recognized

4 Conclusions

By analyzing the outcomes, it becomes evident that our detector excels in identifying faces within group portraits, yet encounters challenges with drawings or images affected by blurring. It seems to encounter the biggest difficulty in detecting faces in darker regions of the images, while it performs much better with well-illuminated subjects facing the camera directly. On the other hand, it encounters difficulty in accurately detecting profiles, as can be seen in Figure 8. Nevertheless, the Viola-Jones Detector shows very good properties for what concerns computational time and is still considered one of the most promising tools for face detection.