Face detection using multi-scale HOG algorithm

Saul Gómez
Universidad de Los Andes
sc.gomez11@uniandes.edu.co

Diego Valderrama
Universidad de Los Andes
df.valderrama@uniandes.edu.co

Abstract

Detection of objects and classify them into different class in a same image is a complex task in computer vision. The multi-scale histograms of oriented gradients (HOG) extracte features of the image that allow classify objects in different scales into different classes. The current study evaluates the performances of multi-scale HOG using 4 sliding windows. It was obtained an average precision of 0.769. The performance also was compared with Viola & Jones method.

1. Introduction

Object recognition is one of the fundamental challenges in computer vision [2]. Within this field of research, the object detection task has been widely studied by researchers. However, the detection of real-world objects of interest, such as faces and people, represents a complex task given that have challenging problems, among which we can highlight the difficulty modeling the objects, the significant variety in color and texture, and the fact that the backgrounds against which the objects lie are unconstrained [6]. Since object recognition is closely related to this task, it is possible to address the object detection problem as a binary classification problem, where it is necessary to find possible candidates in an image to later classify them. Nonetheless, in contrast to the case of pattern classification where we need to decide between well-defined classes, the detection problem requires us to differentiate between the object class and the rest of the world [6]. Consequently, it is necessary to implement a system that possess a model of the object class that has high interclass and low intraclass variability [5]. Additionally, a robust object detection system should be able to detect objects in uneven illumination, objects which are rotated into the plane of the image, and objects that are partially occluded or whose parts blend in with the background [5].

Now, face detection is the step stone to all facial analysis algorithms, including face alignment, face modeling, face relighting, face recognition, face verification/authentication, head pose tracking, facial expression [9]. Although face detection is one of the visual tasks which humans can do effortlessly, in computer vision terms, this task could become a very complex challenge [3]. In general, the main idea in this problem is to detect and localize whether or not an unknown number of faces in a given arbitrary image and return the coordinates where each face is located [3], [9], [8]. Due to its wide usefulness, face detection is a well studied problem in computer vision, reaching modern face detectors that can easily detect near frontal faces [4]. Notwithstanding, many of the current face recognition techniques assume the availability of frontal faces of similar sizes, which produce a bias on the system, since in reality, this assumption may not hold due to the varied nature of face appearance and environment conditions where a face can occur in a complex background and in many different positions [3]. Thus, recognition systems that are based on standard face images are likely to mistake some areas of the background as a face [3]. The difficulty associated with face detection can be attributed to many variations in scale, location, orientation, pose changes, exaggerated facial expressions, extreme or poor illumination conditions, occlusions, that can lead to large visual variations in face appearance and significantly reduce the robustness of the detector [9], [4]. In addition to this, it is also necessary to take into account the large visual variations of human faces in the cluttered backgrounds and the large search space of possible face positions and face sizes [4].

Henceforth, it is necessary to develop a method that can overcome the challenges mention above joint with an appropriate image representation in order to achieve an accurate face detection algorithm that provides efficiency and reproducible results regardless of the database being used or the problem addressed.

2. Methods

2.1. Multi scale HOG strategy

Multi scale HOG strategy combine spatial pyramids for images and HOG descriptor commonly used in classification vision problem in which is necessary differentiate objects from different classes [1]. As in HOG method, this method also uses a sliding window to extract the HOG descriptor for each of the windows that run through the original image. However, since pyramids are used, the descriptor is extracted for each of the levels of the pyramid maintaining the size of the sliding window in each level [1]. This allows to obtain a more robust descriptor that can be used to characterize objects that are in several scales in a same image [1].

2.2. Algorithm

Our method consists in a multi-scale HOG descriptor to detect faces. We obtained the positive features vector using a HOG descriptor for all the training images of the dataset. The set of negative images was used to obtain 20.000 negative features in a random way in order to train a SVM with the positive and negative features of the dataset. Subsequently, a multi-scale HOG descriptor of 4 sliding window (27x27, 39x39,51x51 and 63x63 pixels) was applied in all test set. For each window, we got the HOG vector and assign a them to one class (face or no face) ussing a threshold for our classifier. If the class assigned was face, we saved the coordinates of the bounding box. Later, we made a non-maximum suppression in order to remove the overlapping detections. Finally, we evaluate our detection method with the groundtruths of the test set and also with other extra test set that consist in 20 natural images.

On the other hand, it is important to clarify that exists 2 fundamental hyperparameters in our method. The first one, is the threshold that we use to assign a feature vector of the sliding window to face class. The step of the sliding window is other fundamental hyperparameter because with a lower steps we can obtain more information that could be a face that with a higher steps we could ignored. The rest of the hyperparameter as number of orientations (9), hog cell size (6) and template size (36x36) were kept constant.

2.3. Experiments

As mentioned above, the number of steps and the threshold are two fundamental hyperparameters of our method. For this reason, we made some experiments to obtained the values of these hyperparameter that improve the performance of our method. We defined 3 thresholds (0, 0.5 and 1) and for each one we change the number of steps in 3 or 6. Then, we select the hyperparameters with wich the average precision is higher.

2.4. Evaluation

Given that this is a detection problem, the method used to evaluate the performance was precision-recall curves. To obtained these metrics, we used the code proposed by Viola & Jones with our classifier [7]. Furthermore, we also get the performance using Viola & Jones method.

3. Results

3.1. Test set

Table 1 shows the average precision and the time of processing for all of the experiments made. We found that the best threshold for our method is 0.5 and we also found that a higher number of steps improve the performance of our method. This may have happened since with a higher step some background in the image have been ignored. For this reason, we select 0.5 and 6 for the threshold and number of steps, respectively.

Table 1. Optimum hyperparameter values

Threshold	Step	Average Precision	Time (min)
0	3	0.768	110.1
0.5	3	0.767	110
1	3	0.75	109.8
0	6	0.759	80
0.5	6	0.769	79.8
1	6	0.759	80.2

Fig 8 and 6 shows the precision-recall curve for our method and viola & Jones method, respectively. These images shows that Viola & Jones method has better accuracy and better precision than our method. One reason of these results is the difference between the classifiers because we used a single classifier and Viola & Jones used a cascade of classifiers that allow to get better results [7].

3.2. Extra-test set

Figure 1 shows examples of detections using our method and Viola & Jones method in the extra test set. As you can see, with our method we get many false positives. The reason of this situation can be the size of the window because the sliding window used are very small compared to the size of the faces. On the other hand, Fig 1(c), 1(d) show that Viola & Jones method works better than ours but this method also presents difficulties when there are a large number of close faces in the same image.

4. Discussion

First, as shown in the figures, the method implemented with the optimal parameters presents a high average precision (AP). However, the accuracy of the algorithm is reduced as it is not in the ability to correctly discriminate structures at the background and tends to classify them as a face detection. As stated in table 1, when the method iterates the sliding window over the image with a smaller step, it is able to analyze a greater amount of pixels, which provides an advantage for the detection of faces reflected in a greater precision. In the opposite case, increasing the number of steps results in a bias that prevents the correct extraction of the shape representation of each pixel, which

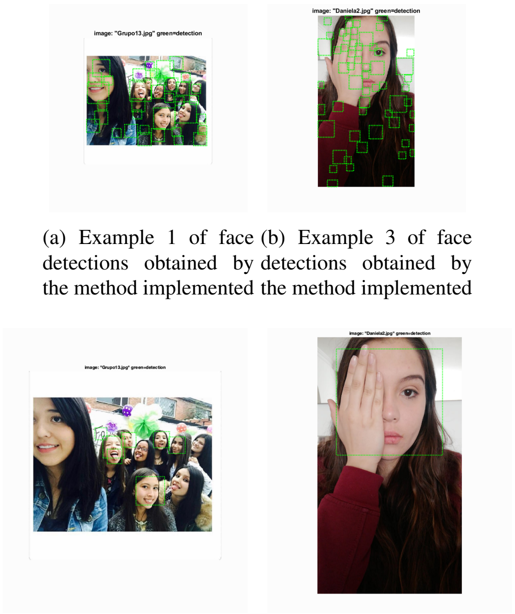
image: "Grupo13.jpg" green=detection
image: "Daniela2.jpg" green=detection
image: "Grupo13.jpg" green=detection
image: "Daniela2.jpg" green=detection

(a) Example 1 of face detections obtained by the method implemented

(b) Example 3 of face detections obtained by the method implemented

(c) Example 1 of face detections obtained by the Viola & Jones method

(d) Example 2 of face detections obtained by the Viola & Jones method

Figure 1. Examples of face detections obtained by the method implemented and by Viola & Jones method in extra test set

results in a reduction in the accuracy of the method. Likewise, while a high number of steps represents better efficiency and more computational savings, if the task requires finer detection it will be convenient to decrease the number of steps in order to obtain more accurate results.

On the other hand, it was expected that increasing the threshold value would reduce the recall of the algorithm, since the classifier model needs a higher level of detection confidence to classify it as true positive. However, this can only be seen in the figure 4 for the experimental run using a threshold of 1 with steps of 3 pixels, where the recall was seriously reduced. In the case of other experimental runs (See figures 2, 3, 5), the recall remained invariant to the changes in the threshold.

Now, in figure 9 can be observed examples of the method's performance in face detection in several images of the test set. As can be noticed in figure 9(a), the algorithm presents a perfect performance detecting the face of a single person in a clear backgroung even in the presence of noise due to shape representation of the image that allows to ignore the salt and pepper noise. Similarly, as seen in the figure 9(f), the algorithm is able to detect multiple faces present in the same image. On the other hand, from those observed in the figures 9(b), 9(d), 9(g) and 9(h) it is possible to determine a pattern present in the detection of false positives mainly due to the cluttered background. The representation of the shape of the multiple structures present in the images present a sufficiently high level of confidence that leads the classifier to confuse them with a face. Yet, as shown in figure 9(b) the algorithm detects easily hand-draw faces. This can be explain since this kind of faces are not affected by different factors such as illumination, occlusion or changes of the point of view, which allows the algorithm to detect them.

Furthermore, the method tends to classify structures that belong to the person, such as clothing or body parts, as faces generating more false positives and decreasing accuracy. Also, it can be evidenced a pattern in the false negative results (see figure 9(d)). People with dark skin prevent the algorithm from recognizing the shape patterns learned in the training set, so it is unable to detect these types of faces. Also, when the resolution of the image does not allow a person's face to be viewed in detail, the developed method cannot detect it. Finally, an inherent error of the method associated with the supplied groundtruth could also be evidenced. As can be seen from the figure 9(h), only a single face on the entire image was labeled, however, as can be seen the method manages to detect the faces of different people regardless of their position or head rotation, which are still classified as false positives.

As mentioned in the results section, one of the main differences between our method and the method proposed by Viola & Jones is the classifier. These researchers proposed a cascade of classifiers based on AdaBoost whose complexity is increasing in order to have a better selection criteria to classify the detections as positive or negative [7]. Other differences between our method and the method proposed in [7] are related with the representation of the image and the processing time. In [7], the researchers presented and used the concept of ”Integral image” that eliminate the need to compute a multi-scale pyramid for the detection of faces at different scales in a same image. This image representation decreases the processing time required by the cascade of classifiers for face detection, allowing 15 frames per second to be processed [7].

5. Conclusions

In conclusion, despite the different errors obtained in the results for the detection of faces by the algorithm developed, the presented performance is acceptable to give solution to the problem of detection of faces reaching an AP close to 80% exceeding the randomness. However, there are some factors that may improve this performance, between which we can find to incorporate a greater variety of sliders windows in order to detect the features of the faces whose dimensions are extremely large and occupy most of the image. Also, considering a training set with images that share more similarity with those of the evaluation set would provide better learning to the algorithm. In addition, this would also allow hard negative mining, with the aim that the method recognizes as negative the structures that generate more confusion.

References

[1] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *international Conference on computer vision & Pattern Recognition (CVPR'05)*, volume 1, pages 886–893. IEEE Computer Society, 2005.

[2] P. F. Felzenszwalb, R. B. Girshick, D. A. McAllester, and D. Ramanan. Object detection with discriminatively trained part based models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32:1627–1645, 2009.

[3] E. Hjelmås and B. K. Low. Face detection: A survey. *Computer Vision and Image Understanding*, 83:236–274, 2001.

[4] H. Li, Z. L. Lin, X. Shen, J. Brandt, and G. Hua. A convolutional neural network cascade for face detection. *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5325–5334, 2015.

[5] A. Mohan, C. Papageorgiou, and T. A. Poggio. Example-based object detection in images by components. *IEEE Trans. Pattern Anal. Mach. Intell.*, 23:349–361, 2001.

[6] C. Papageorgiou, M. Oren, and T. A. Poggio. A general framework for object detection. In *ICCV*, 1998.

[7] P. Viola and M. J. Jones. Robust real-time face detection. *International journal of computer vision*, 57(2):137–154, 2004.

[8] M.-H. Yang, D. J. Kriegman, and N. Ahuja. Detecting faces in images: A survey. *IEEE Trans. Pattern Anal. Mach. Intell.*, 24:34–58, 2002.

[9] C. Zhang and Z. Zhang. A survey of recent advances in face detection. 2010.

Images

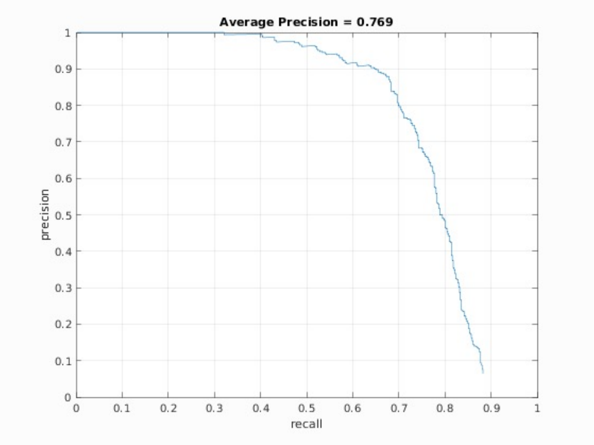

Figure 2. Precision-Recall curve using six steps and 0.5 threshold in test set.

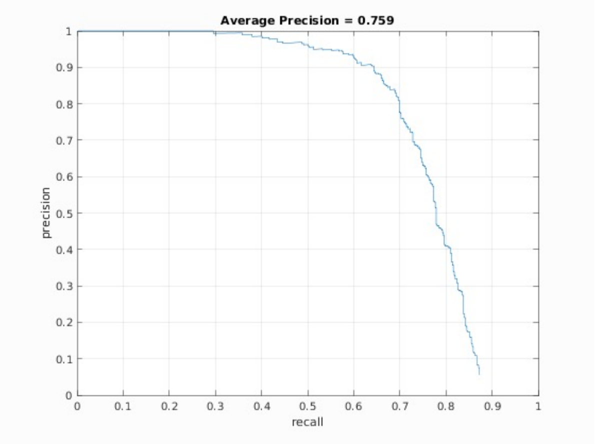

Figure 3. Precision-Recall curve using six steps and 1 for threshold in test set.

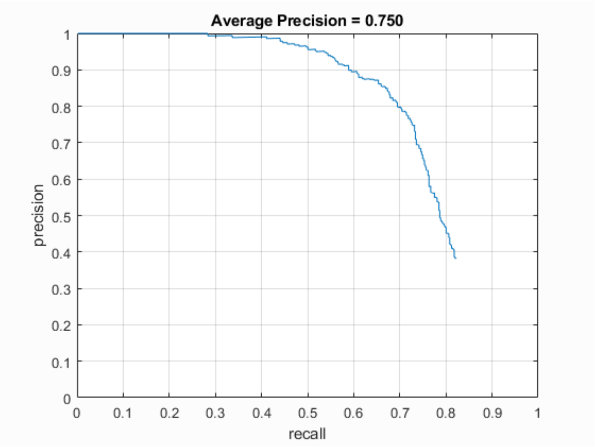

Figure 4. Precision-Recall curve using three steps and 1 threshold in test set.

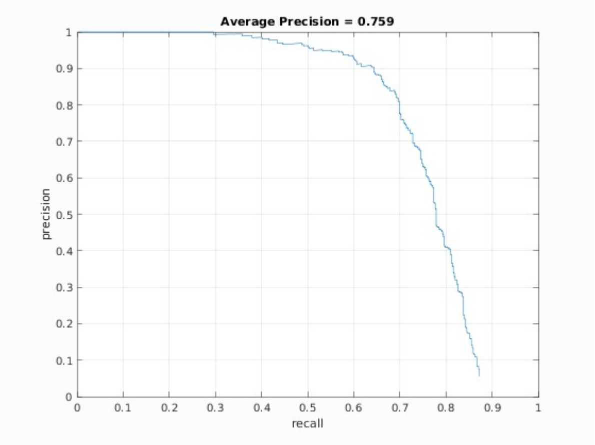

Figure 5. Precision-Recall curve using six steps and 0 threshold in test set.

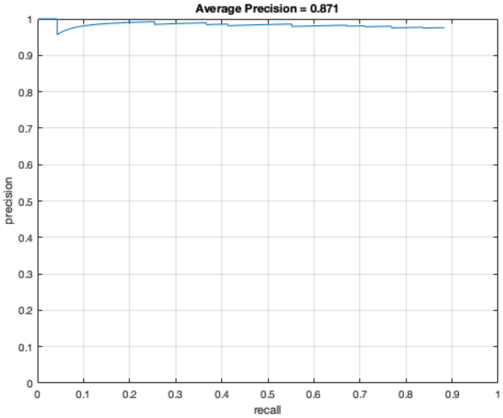

Figure 6. Precision-Recall curve using Viola & Jones method in test set.

Figure 7. Precision-Recall curve using Viola & Jones method in extra test set.

Figure 8. Precision-Recall curve using our method in extra test set.

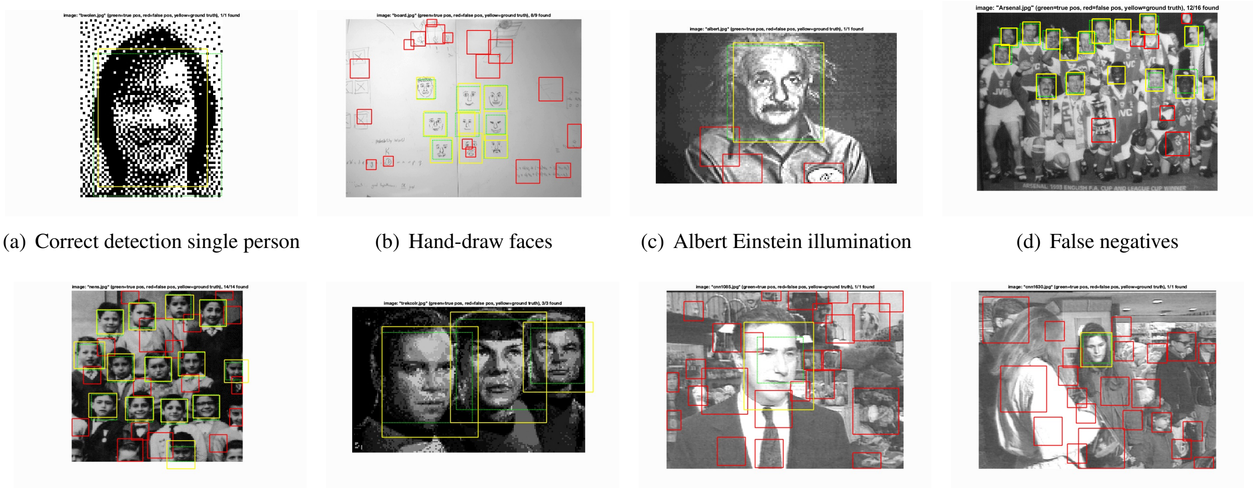
image: "bwolen.jpg" (green=true pos, red=false pos, yellow=ground truth), 1/1 found
image: "board.jpg" (green=true pos, red=false pos, yellow=ground truth), 8/9 found
image: "albert.jpg" (green=true pos, red=false pos, yellow=ground truth), 1/1 found
image: "Arsenal.jpg" (green=true pos, red=false pos, yellow=ground truth), 12/16 found
ARSENAL 1993 ENGLISH F.A. CUP AND LEAGUE CUP WINNER
image: "nens.jpg" (green=true pos, red=false pos, yellow=ground truth), 14/14 found
image: "trekcolr.jpg" (green=true pos, red=false pos, yellow=ground truth), 3/3 found
image: "cnn1085.jpg" (green=true pos, red=false pos, yellow=ground truth), 1/1 found
image: "cnn1630.jpg" (green=true pos, red=false pos, yellow=ground truth), 1/1 found

(a) Correct detection single person

(b) Hand-draw faces

(c) Albert Einstein illumination

(d) False negatives

(e) Group of children

(f) Faces in group of people image

(g) Cluttered background

(h) Inherent error

Figure 9. Examples of face detections obtained by the method implemented