LAB 4 REPORT

A PREPRINT

Daniele cecca
Artificial Intelligence for Science and Technology
Milano Bicocca University
Supervised Learning

April 12, 2024

1 Introduction

Face detection is a really important task in computer vision. One method to solve this task is by using The Viola-Jones (VJ) algorithm. In this paper, we discuss the implementation of the VJ algorithm for face detection, focusing on methodologies employed to enhance specificity

2 Viola Jones' Algorith - VJ

Viola-Jones algorithm is a machine-learning technique for object detection proposed in 2001 by Paul Viola and Michael Jones in their paper "Rapid object detection using a boosted cascade of simple features". The algorithm was primarily conceived for face detection. Despite it has lower accuracy than modern face detection methods based on Convolutional Neural Networks (CNNs), the Viola-Jones algorithm is still an efficient solution for resource-constrained devices.

Given a grayscale image, the algorithm analyzes many windows of different sizes and positions and tries to detect the target object by looking for specific image features in each window.

The Viola Jones algorithm has four main steps

- Selecting Haar-like features
- Creating an integral image
- Running AdaBoost training
- Creating classifier cascades

3 Methodology

To enhance the specificity of the VJ algorithm, various methodologies are employed during training.

3.1 Data Augmentation

One crucial aspect is data augmentation, particularly for the negative class (non-face images). Augmentation techniques such as flipping, rotation, and mirroring are applied to increase the diversity of negative samples. This augmentation helps the algorithm better discriminate between faces and non-face regions, leading to improved specificity.

```
for ii=1:size(neg,1)
    im=imread([neg(ii).folder filesep neg(ii,1).name]);
    imwrite(im,[outdir filesep neg(ii,1).name]);

    [pp,ff,ee]=fileparts(neg(ii).name);
```

```
    im_flip=fliplr(im);
    imwrite(im_flip,[outdir filesep ff '_flip' ee]);

    im_ud = flipud(im); % Upside-down version
    imwrite(im_ud,[outdir filesep ff '_ud' ee]); % Save upside-down version

    for nrot=1:10 % Rotate the image multiple times
        imr = imrotate(im, 35*nrot, 'crop'); % Rotate by multiples of 35 degrees
        imwrite(imr,[outdir filesep ff '_r' num2str(nrot) ee]);
    end
end
```

3.2 Tuning of Hyperparameters

Additionally, parameter tuning significantly contributes to specificity improvement. Specifically, attention is directed towards the optimization of the following parameters during the training phase:

- **False Alarm Rate**: Acceptable false alarm rate at each stage, specified as a value in the range (0 1]. The false alarm rate is the fraction of negative training samples incorrectly classified as positive samples. Lower values for FalseAlarmRate increase complexity of each stage. Increased complexity can achieve fewer false detections but can result in longer training and detection times. Higher values for FalseAlarmRate can require a greater number of cascade stages to achieve reasonable detection accuracy.
- **Number of Cascade Stages**: Number of cascade stages to train, specified as a positive integer. Increasing the number of stages may result in a more accurate detector but also increases training time. More stages can require more training images, because at each stage, some number of positive and negative samples are eliminated. *(In our case due to the low number of images the training will end at the 5th stage)*

```
trainCascadeObjectDetector('myFaceDetector.xml',positiveInstances, ...
    negativeImages, 'FalseAlarmRate',0.05, 'NumCascadeStages',20);
```

4 Conclusion

The optimized VJ algorithm demonstrates improved specificity in face detection tasks. By augmenting the negative class and fine-tuning training parameters, the algorithm achieves better discrimination between faces and non-face regions. Evaluation on test images yields promising results, with an average precision of 0.60. These findings highlight the effectiveness of the methodologies employed in enhancing the specificity of the Viola-Jones algorithm for face detection applications.

However, it's worth noting that with more computational power at our disposal, further enhancements could be explored. For instance, increasing the number of cascade stages , augmenting even more the negative samples and augmenting positive samples, similar to what was done for negative samples, could potentially yield even better results. These more complex solutions might require additional computational resources but could lead to further improvements in detection accuracy and specificity.

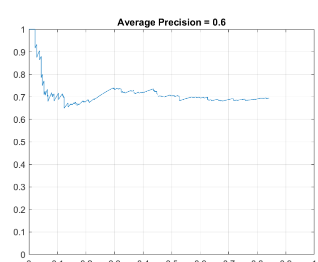

Figure 1: Result Average Precision

References

[1] Train Cascade Object Detector Model - MATLAB TrainCascadeObjectDetector. *MathWorks United Kingdom*. Available at: `uk.mathworks.com/help/vision/ref/traincascadeobjectdetector.html`. Accessed 12 Apr. 2024.

[2] The Viola-Jones Algorithm | Baeldung on Computer Science. Baeldung. Available at: `www.baeldung.com/cs/viola-jones-algorithm`. Accessed 12 Apr. 2024.