Machine Learning, ICS3206, Course Project 2020

Liam Attard [0299300L]

1 Viola-Jones technical discussion

1.1 Introduction

The Viola-Jones paper (Rapid Object Detection using a Boosted Cascade of Simple Features) describes a machine learning technique that achieves a fast and accurate object recognition method that isn't based on deep learning. This report is still relevant today especially for face detection and has been cited around 22,000 times. The technique used is achieved from the following three steps:

- The image is first represented using a summed-area table called the integral image.
- The principal features of the target object are picked out from a training set using an algorithm based on AdaBoost and generates efficient classifiers.
- The image is then passed through several filters, referred to as a "cascade structure" which is, in essence, a degenerate decision tree.

This paper's detector was tested on 384 by 288 images at 15 frames per second and found to be accurate irrelevant of facial features and ethnicity. The prompt detection of this technique is what makes it ahead of other methods.

1.2 The Integral Image

1.2.1 Haar-Like Features

Haar features are rectangular features maked over an image. To calculate these principal features of an image several computations are performed.

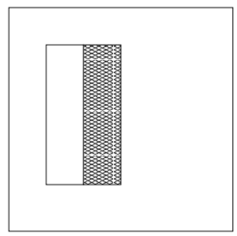

Figure 1: Shows an example of a feature where the sum of pixels in the white rectangle is subtracted from the sum of pixels in the grey rectangle. By performing these calculations on raw image values, the result can take a lot of time so that is why calculating the integral image any rectangular value can be performed quickly and in constant time.

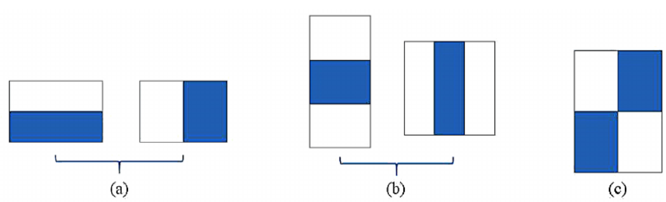

Figure 2: Shows different ways of calculating the features, (a) shows two-rectangle features, (b) shows three-rectangle features and (c) shows a four-rectangle feature.

1.2.2 Calculating The Integral Image

The integral image is found by iterating over each pixel and computing its new value, which is obtained by calculating the sum of pixels above and to the left of it. The original pixel i(x,y)'s integral image ii(x,y) can be found by using the following equation.

$$ii(x, y) = \sum_{(x \leq x, y' \leq y)}$$

1	5
2	4

Original Image

The Matrix below shows an example of an original image alongside its integral image.

1	6
3	12

Integral Image

1.2.3 Integral Image PseudoCode

```
for each pixel(x2,y2):
    for x in x1 to x2:
        for y in y1 to y2:
            sum += Arr[x][y]
```

Where x1 is the row number of the top-left pixel [0,0], y1 is the column number of the top-left pixel [0,0], x2 is the row number of the bottom-right pixel, y2 is the column number of the bottom-right pixel.

1.2.4 Calculating the Sum of Any Pixel Value From the Integral Image

Consider the following matrix:

1	7	4	2	9
7	2	3	8	2
1	8	7	9	1
3	2	3	1	5
3	0	5	6	6

Original Image

The sum of the greyed out area is 1+5+6+6 which is 18. Now consider its integral image:
To Calculate the sum of the greyed out area subtract the summation of the unwanted areas, in this case: 113 - 64 - 71 which is -22 and re-add the areas

1	8	12	14	23
8	17	24	34	45
9	26	**40**	59	**71**
12	31	48	68	85
14	42	**64**	90	113

Integral Image

that have been taken off twice in this case -22+40 which is 18. Although in this case, it didn't make sense to calculate the integral image given how small the size of the original image is, in pictures with a very high resolution, this would save a lot of time.

1.3 Learning Classification Functions Using AdaBoost

Over 180,000 features can be calculated given that the detector is of size 24x24 pixels. This number of features can be in danger of both slow classification and overfitting. In order to avoid such problems a boosting algorithm, AdaBoost, is used to select a small set of critical Haar-like features using weak classifiers. For each of the features the algorithm finds the function which separates them positively (contain a face) and negatively (don't contain a face) in the most optimal way ie. leaving the least possible number of values misclassified. Given a 24 by 24 image x with feature f_j, a weak classifier (h_j), is defined by

$$h_j(x) = 1 \text{ if } p_j f_j < p_j \theta_j$$
$$h_j(x) = 0 \text{ otherwise}$$

where θ_j is the treshold and p_j indicates the direction of the inequality sign.

1.3.1 Resultant Features

For the Viola-Jones algorithm a very aggresive approach which selects only a few number of features was implemented.

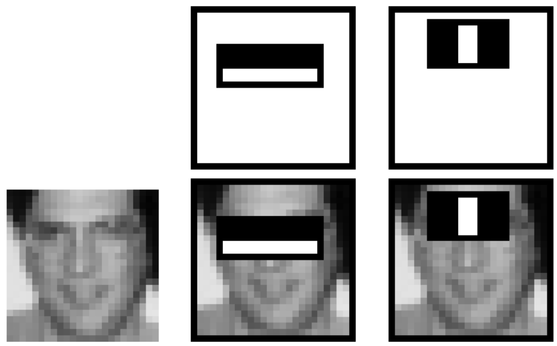

Figure 3: The figure shows how the top two features AdaBoost gave priority to are the darker eye area over the cheeks and the bridge of the nose.

1.4 Cascade of Classifiers

The third part of the algortihm places the previous features in a form of ranking stages where an image would have to pass through all of the stages in order to result positive (ie. contain the object/face). This increases performance by avoiding counting all the features for parts of the image which are definetly negative, if the image fails to pass through the first feature, it fails.