IMAGE PROCESSING

Study of Face Recognition and Learning

PREPARED IN PARTIAL FULFILLMENT OF THE STUDY ORIENTED PROJECT COURSE (MATH F266) DURING SEMESTER-I 2014-15

INSTRUCTOR:BIJIL PRAKASH

By:

Bhavul Gauri

2011B4A7439G

ABSTRACT

This report includes the description about the Study Oriented Project done during 1st Semester of 2014-2015. It navigates through different algorithms for Face Recognition as a part of Image processing, and studies in detail, the most widely used Viola and Jones Algorithm for face detection. Then, in the second part, it presents two learning algorithms - Support Vector Machines and Neural Networks, which are known and shown to work well on the image datasets for Face verification. Adaboost, the core part of Viola Jones face detection, and the most widely known Multiclassification algorithm for SVM, One vs One classification has been implemented as a part of the project.

The aim of the project was to study different ways Face Recognition is being used these days. As a part of future research, this serves as a base, and can be used to further work on betterment of the existing state-of-the-art algorithms presented in this report.

Acknowledgements

I would like to thank BITS Pilani and the Mathematics Department for giving me an opportunity to work on this study project. Moreover, I can't express enough regards to my mentor, Mr. Bijil Prakash, who has guided me all along, and always encouraged me for research in the area, inspiring on the days whenever I felt down.

Contents

1 Introduction

We are in midst of a visually enchanting world, which manifests itself wih a variety of forms and shapes, colors and textures, motion and tranquility. The human perception has the capability to acquire, integrate, and interpret all this abundant visual information around us. It is challenging to impart such capabilities to a machine in order to interpret the visual information embedded in still images, graphics, and video or moving images in our sensory world. It is thus important to understand the techniques of storage, processing, transmission, recognition, and finally interpretation of such visual scenes. Face recognition has become a popular research area in last ten years or so and has been gathering attention of not only computer scientists, but neuroscientists, and psychologists also. The problem of face recognition is very simple to formulate - Given a still image or video images of a scene, machine needs to identify or verify one or more persons in the scene using the database of faces.

1.1 What is Face Recognition

Face recognition is the field of identifying faces. Typically known to be used for different forms of security systems, a facial recognition system identifies a person from a digital image or a video frame. The popular way of doing this is by comparing selected facial features from the image with a database.

1.2 Current Popular Uses

There has been a lot of buzz since the first face recognition device came up. At first, it was used only for security purposes as a biometric. But, these days it’s being used in smartphones, for attendances in classes, to catch criminals and theives and what not. It is believed that face recognition will open up a ton of possibilities in how we interact not just with each other, but with objects as well.

2 Face Recognition - What, How and Why?

2.1 History

Facial recognition algorithms were introduced in 1960s, the first one located the features, such as eyes, nose and mouth on the photographs. In 1970s, Goldstein and Harmon came up with 21 special markers to automate the recognition. And, finally in late 1980s, Kirby and Sirovich applied algebra techniques to face recognition problems. Since then, it's been a popular field of research and people have tried integrating different aspects of computer science and associated different fields of mathematics to come up with algorithms that are faster, and more accurate.

2.2 Face Recognition : a 2 step process!

Face recognition can be broken into two individual and mostly independent steps, each of which is a field of study in itself. Face recognition involves face detection/identification as its first step, and face verification or matching as the second.

The first process only involves processing the image and finding facial features in the image, so as to detect if the image or frame contains a human face or not. However, the second process is related to Machine Learning, and deals with some algorithm that matches the facial features extracted from the image to the database of facial features available for learning. Only after this is done, our machine is able to predict whose face is present in the image.

If we had to categorize, all the face recognition algorithms can be broadly classified into three types :

1. **Holistic Models** : Here, complete face is taken as input for detection.Some of the best examples are Face recognition using eigenfaces, Principle Component Analysis, Linear discriminant analysis, etc.

2. **Feature-based Models** : In this methods local features such as eyes, nose and mouth are first of all extracted and their locations and local statistics (geometric and/or appearance) are fed into a structural classifier. Viola Jones face detection, on which we'll be discussing more in this report is one of the most used examples of this kind.

3. **Hybrid Models** : Hybrid face recognition systems use a combination of both holistic and feature extraction methods. Generally 3D Images are used in hybrid methods. The image of a person's face is caught in 3D, allowing the system to note the curves of the eye sockets, for example, or the shapes of the chin or forehead. Even a face in profile would serve because the system uses depth, and an axis of measurement, which gives it enough information to construct a full face. The 3D system usually proceeds thus: Detection, Position, Measurement, Representation and Matching.

2.3 Face Recognition as a Security Measure

The security of information is becoming very significant and difficult. Security cameras are presently common in airports, Offices, University, ATM, Bank and in any locations with a security system. Face recognition is a biometric system used to identify or verify a person from a digital image. Even in smartphones today, face recognition systems are built in which use the front camera to securely lock or unlock the device.

Early face recognition algorithms used simple geometric models, but the recognition process has now matured into a science of sophisticated mathematical representations and matching proceses. Major advancements and initiatives in the past ten to fifteen years have propelled face recognition technology into the spotlight. Face recognition can be used for both verification and identification (open-set and closed-set).

3 The problem of Face Detection

A face detector has to tell whether an image of arbitrary size contains a human face and if so, where it is. There are many ways to detect a face in a scene - easier and harder ones. Here is a list of the most common approaches in face detection:

1. **Using Color in the image** : The color of skin segments is used to identify the faces and their location in the image. Some algorithms were made for this, one of which even used PCA, but this method never reached ultimate popularity since it failed due to different lighting conditions

2. **Finding faces by Motion** : If we have a video with us, the fact that face is always moving(even if by a very small margin) can be exploited to find the faces. Some researches worked for designing algorithms that did this, but they didn't perform well when other objects were also moving in the background

3. **Finding faces in Unconstrained Images** : This is the part current research is going on. How do you design an algorithm that performs well irrespective of the background, the colors, and the lighting conditions. Various attempts have been made, and numerous algorithms have come out which used techniques ranging from Neural Networks to Hausdorff distance to geometrically locate different features and detect the face in the image.

   However, the breakthrough happened when Viola and Jones algorithm was introduced. It defined 'weak classifiers' using haar features, and showed that on excessive training of these weak classifiers, amazingly accurate results could be achieved. Since then, it's being used most commonly and widely for face detection.

3.1 Viola Jones Algorithm

There are three ingredients working in concert to enable a fast and accurate detection: the integral image for feature computation, Adaboost for feature selection and an attentional cascade for efficient computational resource allocation. The basic principle of the Viola-Jones face detection algorithm is to scan the detector many times through the same image - each time with a new size.

3.1.1 Integral Image and features

The Viola-Jones algorithm uses Haar-like features, that is, a scalar product between the image and some Haar-like templates.

Haar Features

Haar features are similar to the convulation kernels which are used to detect the presence of that feature in the given image. Each feature results in a single value which is calculated by subracting the sum of pixels under white rectange from the sum of pixels under the black rectange.
Viola Jones algorithm uses a 24x24 window as the base window size to start evaluating these features in any given image. The value of any given feature is always simply the sum of the pixels within clear rectangles subtracted from the sum of the pixels within shaded rectangles. If we consider all possible parameters of the haar features, like position, scale and type, we end up calculating 160,000+ features in this small window itself. However, with the use of an image representation called the integral image, rectangular features can be evaluated in constant time, which gives them a considerable speed advantage over their more sophisticated relatives. Because each rectangular area in a feature is always adjacent to at least one other rectangle, it follows that any two-rectangle feature can be computed in six array references, any three-rectangle feature in eight, and any four-rectangle feature in just nine. To compensate the effect of different lighting conditions, all the images should be mean and variance normalized beforehand. Those images with variance lower than one, having little information of interest in the first place, are left out of consideration.

The integral image which allows to calculate them at a very low computational cost. Instead of summing up all the pixels inside a rectangular window, this technique mirrors the use of cumulative distribution functions.

3.1.2 Feature selection using Adaboost

All right, we can get the features using haar-like features as told above. But, which one of them are needed? We know that in base window of 24x24 base resolution, 160000+ features are evaluated. But, only few of them are relevant.
One natural framework for considering this problem of face detection is that of binary classification, in which a classifier is constructed to minimize the misclassification risk. Since no objective distribution can describe the actual prior probability for a given iage to have a face, the algorithm must minimize both the false negative and false positive rates in order to achieve an acceptable performance. This task requires an accurate numerical description of what sets human faces apart from other objects. While there exists It turns out that these characteristics can be extracted with a remarkable committee learn- ing algorithm called Adaboost, which relies on a committee of weak classifiers to form a strong one through a voting mechanism. A classifier is weak if, in general, it cannot meet a predefined classification target in error terms.

Adaboost finds a weighted combination of all these features in the way that each of the selected features are considered okay to be included if they can at least perform better than random guessing (detects more than half the cases). Adaboost constructs a strong classifier as a linear combination of these weak classifiers.
How Adaboost works :

1. Adaboost starts with a uniform distribution of “weights” over training examples.
2. Select the classifier with the lowest weighted error (i.e. a ’weak’ classifier)
3. Increase the weights on the training examples that were misclassified.
4. Repeat the first three steps.
5. At the end, carefully make a linear combination of the weak classifiers obtained at all the iterations.

3.1.3 CASCADING

Even if an image should contain one or more faces, it is obvious that an excessive large amount of the evaluated sub-windows would still be negatives (non-faces). So, the algorithm should concentrate on discarding non-faces quickly and spend more time on 'probable' face regions. Hence, a single strong classifier formed out of linear combination of all best features is not a good way to evaluate on each window because of computation cost. This is why, a cascade classifier is used which is composed of stages, each containing a strong classifier. So, all the features are grouped into several stages where each stage has certain number of features. The job of each stage is used to determine whether a given sub window is definitely not a face OR may be a face. A given sub window is immediately discarded as not a face if it fails in any of the stage.

For training a cascade, we must choose the number of stages in cascade (or the number of strong classifiers), the number of features of each strong classifier, and the threshold of each strong classifier. But, how do we choose this? Finding an optimum combination is extremely difficult. So, Viola and Jones suggested a heuristic algorithm for cascade training.

- select f_i (Maximum Acceptable False Positive rate/stage)
- select d_i (Maximum Acceptable True Positive rate/stage)
- select Ftarget (Taret Overall False Positive rate)

  Until Ftarget is met :

  - Add new stage:
    - Until f_i,d_i rates are met for this stage, keep adding features and train new strong classifier with AdaBoost.

4 The problem of Face Verification

Once the face has been detected in the image/frame, we need to verify if we can identify the face. This is done using learning algorithms. A dataset of faces is fed to the machine, from features of which, it learns those faces. Then, at the time of identification of a new face, it tries to match the features with the features of faces it has learnt. Whichever comes close enough is produced as the predicted face. Various learning algorithms have been implemented to do this job, and everyday new Machine Learning algorithms keep on coming up. However, based on experimental analysis, it's been seen that Support Vector Machines and Neurat Networks work great with image data, i.e. their features. This is why, we'll be venturing into Support Vector Machines in the rest of the report, and will also introduce ourselves to Neural networks. But, we'll leave the detailed study of how Neural networks performs for future findings.

4.1 Support Vector Machines

Support Vector Machines are powerful classification and regression tools, but their compute and storage requirements increase rapidly with the number of training vectors, putting many problems of practical interest out of their reach. The core of an SVM is a quadratic programming problem (QP), separating support vectors from the rest of the training data.

4.1.1 One against All approach

SVM is originally designed for binary classification and the extension of SVM to the multi-class scenario is still an ongoing research topic. The conventional way is to decompose the M -class problem into a series of two-class problems and construct sev- eral binary classifiers. The earliest and one of the most widely used implementations is the one-against-all method, which constructs M SVM classifiers with the ith one separating class i from all the remaining classes. One problem with this method, however, is that when the M classifiers are combined to make the final decision, the classifier which generates the highest value from its decision function is selected as the winner and the corresponding class label is assigned without considering the competence of the classifiers. In other words, the outputs of the decision function are employed as the only index to indicate how strong a sample belongs to the class. The underlying assumption for doing so is that the classifiers are totally trustable and equally reliable, which does not always hold in multi-class cases.

A working code was written as a part of the project which, for a dataset of image properties taken from UCI Machine learning repository, produced a great result of 98 % accuracy.

4.1.2 One against One approach

Another major method is called the one-against-one method. This method constructs k(k-1)/2 classifiers where each one is trained on data from two classes. For the training data from ith and jth classes, we solve a binary classification problem. There are different methods for doing the future testing after all k(k-1)/2 classifiers are constructed. After some tests, we decide to use the following voting strategy - if sign(decision function) says x is in the ith class, then the vote for the ith class is added by one. Otherwise, the jth is increased by one. Then we predict x is in the class with the largest vote. The voting approach described above is also called the “Max Wins” strategy. In case that two classes have identical votes, though it may not be a good strategy, now we simply select the one with the smaller index.

A working code was written as a part of this project which gave a better accuracy than the above specified One-against-all approach. Not only that, it aced in the trainig and testing time area as well, performing faster than one-against-all approach.

5 CONCLUSION

As a part of this project, I studied the popular face recognition methods, and delved deeper into the implementation of the most widely used approach - Viola Jones face detection algorithm. The study of Face "Recognition" felt more complete when the learning algorithm part dived in, and we looked at Support Vector machines and two of the state of the art algorithms for the same. The code for Viola Jones algorithm and One against All as well as One against One was written as a part of the project, but has been ommitted from the report.

The study helped me understand the way face recognition works in machines, in phones, and this could work very well for a future project of designing a new algorithm, or making it useful in some form of a software or an application that exploits these techniques.

6 BIBLIOGRAPHY

1. Analysis of Multiclass Support Vector Machines, Shigeo Abe, Kobe University

2. A comparison of Methods for Multi-class Support Vector Machines, Chih-Wei Hsu and Chih-Jen Lin, National Taiwan univeristy

3. Image processing techniques in Face Recognition, A.Devi and Dr.A.Marimuthu, Dr.SNS Rajalakshmi College of Arts and Science, Coimbatore

4. Robust Real-Time Face Detection, Paul Viola, Microsoft Research and Michael J. Jones, Mitsubishi Electric Research Laboratory