NATIONAL CONFERENCE ON EMERGING TRENDS INTELLIGENT COMPUTING AND COMMUNICATION
APRIL 13-14, 2012

Access Authentication in Hadoop with Face Recognition

Mr. Pravin Chavan Sagar Salunke

pravinchavan52@gmail.com sagarsalunke7@gmail.com

Awadhut More Prashant Shinde

moreawadhut@gmail.com pashinde.7@gmail.com,

Abstract—

This paper proposes an enhancement in access security in Apache Hadoop[1] with the help of face recognition. Hadoop is a software framework that supports data-intensive distributed applications under a free license. Hadoop creates clusters of machines and coordinates work among them. With the help of wired or wireless network hadoop cloud computing is connected to the several clients or mobile devices. The services like Software as a service (SaaS), Platform as a service (Paas), Infrastructure as a service (Iaas) are initiated by the hadoop. Through the hadoop cloud computing master is connected to the several slaves.

Built-in Hadoop Distributed File System (HDFS) security features such as Access Control Lists and Kerberos used alone are not adequate for enterprise needs. To overcome from this face recognition can be used for authentication. There are several algorithms are available for face recognition, out of it Eigen based face recognition[2] approach is suggested which uses Principal component analysis method .

This whole system is integrated with Apache Hadoop software stack. In cloud computing the major challenge is Security and authentication. Proposed System provide solution for this.

Keywords—

Apache Hadoop, Eigen based Face Recognition, PCA

I. INTRODUCTION

Apache Hadoop is a powerful open source software platform and which is a software foundation project. It allows thousands of nodes and large amount of data to work on the applications. This Hadoop was inspired by the Google's MapReduce [3]and Google File System(GFS) papers.

Hadoop supports file sytems, accessing this file systems is done via Hadoop Common. To start the Hadoop, Hadoop Common packages has necessary JAR files and scripts. This package also has documentation, source code and a contribution section which includes projects from the Hadoop Community.

A. Hadoop Characteristics

1. It is a scalable system, for incrementally storage of

ISBN: 9788175157262

data, it is efficient.

2. It automatically handles data replication and node failure.

3. It does the hard work – developer can focus on processing data logic.

4. Enable applications to work with large amounts of data in parallel.

The proposed system has the following targets:

1. Extra Authentication to login to Apache Hadoop Cloud Services.

2. On the existing security layer, add one more layer.
3. Integration of Face recognition module in Apache Hadoop.

4. To perform Face Recognition in minimum seconds of time.

Need:

Present system uses Kerberos[5] for authentication, to enhance security in addition to Kerberos, paper suggests Face recognition approach for authentication.

B. A generic face recognition system

An input to face recognition system is an image or video stream and the output is the verification of the user. Some approaches define a face recognition system as a three step process - see Figure 1. From this point of view, the Face Detection and Feature Extraction phases could run simultaneously.

Figure 1: A generic face recognition system.

The first step is the Face detection, which is defined as the process of extracting faces from scenes. So, the system positively identifies a certain image region as a face. The next step is feature extraction which is defined as the obtaining relevant facial features from the data. Finally, the system does recognize the face. In an identification task, the system would report an identity from a database. This phase involves a comparison method, a classification algorithm and an accuracy measure.

Detection methods divided into categories
This classification can be:

- Knowledge-based methods- Ruled-based methods that encode our knowledge of human faces.
- Feature-invariant methods- In this method tries to find invariant feature of a face irrespective of it's angle or position.
- Template matching methods- In this method, input images is compared with the stored patterns of the faces or the features.
- Appearance-based methods- A template matching method whose pattern database is learnt from a set of training images.

This paper proposes Eigenface[6]-based method which comes under appearance based method. With the help of PCA(Principal Component Analysis) Sirovich and Kirby developed a method for representing the images efficiently. This approach uses face as a co-ordinate system. For making the coordinate system some vectors are used these are called as eigen pictures. Later, Turk and Pentland used this approach to develop a eigenface-based algorithm for recognition.

II. EIGENFACE GENERATION

A mathematical process called principal component analysis (PCA)[7] is used for generating a set of eigen faces on a large set of images of different human faces. Informally, eigenfaces can be considered a set of "standardized face component", resulted from statistical analysis of many images of faces. The human face can be considered to be a combination of these standard faces. For example, one's face might be composed of the average face plus 20% from eigenface 1, 65% from eigenface 2, and even -35% from eigenface 3. Remarkably, it does not take many eigenfaces combined together to achieve a fair approximation of

ISBN: 9788175157262

most faces. Also, because a person's face is not recorded by a digital photograph, but instead as just a list of values (one value for each eigenface in the database used), much less space is taken for each person's face.

The eigenfaces that are created will appear as light and dark areas that are arranged in a specific pattern. This pattern is how different features of a face are singled out to be evaluated and scored. There will be a pattern to evaluate symmetry, if there is any style of facial hair, where the hairline is, or evaluate the size of the nose or mouth. Other eigenfaces have patterns that are less simple to identify, and the image of the eigenface may look very little like a face.

The technique used in creating eigenfaces and using them for recognition is also used outside of facial recognition. This technique is also used for handwriting analysis, lip reading, voice recognition, sign language/hand gestures interpretation and medical imaging analysis. Therefore, some do not use the term eigenface, but prefer to use 'eigenimage'.

Eigenfaces figures out the main differences between all the training images, and then how to represent each training image using a combination of those differences.

For example, one of the training images might be made up of:

(averageFace) + (13.5% of eigenface0) **-** (34.3% of eigenface1) + (4.7% of eigenface2) + ... + (0.0% of eigenface199).

Once it has figured this out, it can think of that training image as the 200 ratios: {13.5, -34.3, 4.7, ..., 0.0}.

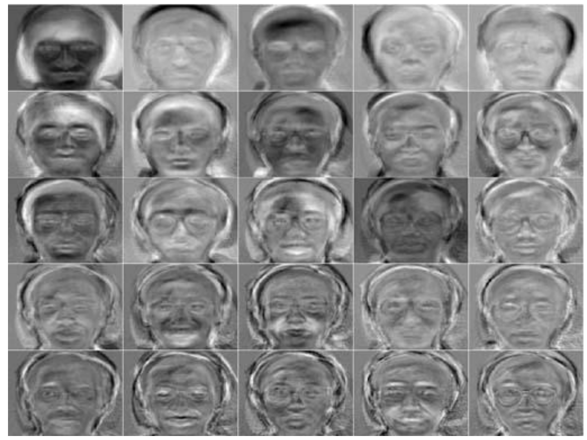

Fig 2: Eigenfaces

A. *Practical implementation*

To achive a set of eigenfaces, following steps are followed:

1. Create a training set of face images. The pictures constituting the training set should have been taken under the same lighting conditions. Then normalisation is performed over images. Images are again sampled to a common pixel resolution ($r \times c$). Every image is treated as one vector, by concatenating the pixel rows in the original image, which results in a single row with $r \times c$ elements. For this implementation, it is assumed that all images of the training set are stored in a single matrix **T**, Here Every row of matrix **T** is an image.
2. Mean is subtracted. Then image **i** has to be calculated which is average image and then subtracted from each original image in **T**.
3. Eigenvectors and eigenvalues are calculated from covariance matrix **S**. Each eigenvector has the equal dimensions i.e. number of components associated with image, as the original images, results in itself as an image. The eigenvectors of this covariance matrix are therefore called eigenfaces. eigenfaces gives directions in which image differs from mean image.It is seen that, this is a computationally expensive step, so as to became practically

ISBN: 9788175157262

applicable eigenfaces stems from the possibility to compute the eigenvectors of **S** efficiently, without ever computing **S** explicitly, as detailed below.

4. Choose the principal components. The *p* x *p* covariance matrix will result in p eigenvectors, each representing a direction in the *r* × *c*-dimensional image space. Only those eigenvectors (eigenfaces) with largest associated eigenvalue are used.

For represetation of both new and existing faces these eigenfaces can now used. To calculate how that new face differs from the mean face it is needed to project a new (mean-subtracted) image on the eigenfaces . The eigenvalues is associated with each eigenface which represent how much the images in the training set vary from the mean image in that direction. But some information is lost by projecting the image on a subset of the eigenvectors, in order to minimize this loss by using only those eigenfaces which have the largest eigenvalues. For example, if working with a 100 x 100 images, then it will result in 10000 eigenvectors. In practical applications, most of the faces are identified using a projection on between 100 and 150 eigenfaces, so remaining all eigenvectors are not necessary hence discarded.

B. *Computing the eigenvectors*

It is infeasible for computation if we performs Principle Component Analysis directly on the covariance matrix of the images. If small, say 100 x 100, greyscale images are used, each image is a point in a 10,000-dimensional space and the covariance matrix **S** is a matrix of 10,000 x 10,000 = 10^8 elements. However the rank of the covariance matrix is limited by the number of training examples: if there are *N* training examples, there will be at most *N-1* eigenvectors with non-zero eigenvalues. If the number of training examples is smaller than the dimensionality of the images, the principal components can be computed more easily as follows.

- Let T be the matrix of preprocessed training examples, where each column contains one mean-subtracted image. The covariance matrix can then be computed as **S** = TT and the eigenvector decomposition of **S** is given by

$$\mathbf{S}\mathbf{v}_i = \mathbf{T}\mathbf{T}^T\mathbf{v}_i = \lambda_i \mathbf{v}_i$$

- However $\mathbf{TT}^T$ is a large matrix, and if instead we take the eigenvalue decomposition of

$$\mathbf{T}^T\mathbf{T}\mathbf{u}_i = \lambda_i \mathbf{u}_i$$

then we notice that by pre-multiplying both sides of the equation with **T**, we obtain

$$\mathbf{T}\mathbf{T}^T\mathbf{T}\mathbf{u}_i = \lambda_i \mathbf{T}\mathbf{u}_i$$

Meaning that, if $\mathbf{u}_i$ is an eigenvector of $\mathbf{T}^T\mathbf{T}$, then $\mathbf{v}_i=\mathbf{T}\mathbf{u}_i$ is an eigenvector of **S**. If we have a training set of 300 images of 100 x 100 pixels, the matrix $\mathbf{T}^T\mathbf{T}$ is a 300 x 300 matrix, which is much more manageable than the 10000 x 10000 covariance matrix. Notice however that the resulting vectors $\mathbf{v}_i$ are not normalised; if normalisation is required it should be applied as an extra step.

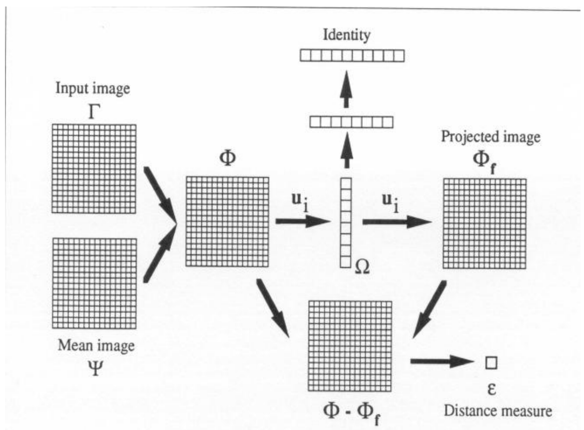

Fig 3: Image Processing for face recognition

C. *Use in Face Recognition*

Behind the creation of eigenfaces had been the source of motivation by facial recognition. For such use. eigenfaces have advantages over all other available techniques like system's speed and efficiency. Eigenfaces are able ot functionally operate on lots of faces in very little time and it is also very fast. Unfortunately, this type of facial recognition have drawback to consider : there is trouble in recognizing faces when they are viewed in differant levels of light or angles. In order to work system well, the faces need to be seen from a frontal view under similar lighting. Face recognition by using eigenfaces have been proving to be quite accurate. By experimenting with system to test it under various different conditions, the following correct recognitions were found : an average of 96% in light variation, 85% in orientation variation and 64% in size variation.(Truk & Pentland, p. 590)

For complementing eigenfaces, another approach named eigenfeatures has been developed. This approach combines facial metrics(measurement of distance between facaial features) with the eigenface approach. Another method, which is competing with the eigenface technique uses 'fisherfaces'. This facial recognition method is less sensitive to variation in lighting and pase of the face than the method usnig eigenfaces. The active appearance model which is a more modern alternative to eigenfaces and fisher faces . This model decouples the face's shape from it's texture: it does an Eigen face decomposition of the face after wrapping it to mean shape. This allows it to perform better on different projections of the face and when the face is tiled.

System Diagram

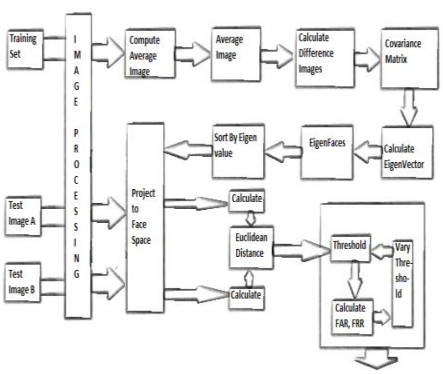

FIG 4: SYSTEM DIAGRAM

Image is captured from the webcamera. Images from the training set are given to the image processing. Then on the particular image average image is captured. After that difference image is calculated, depending upon covariance matrix eigenvector is calculated. Eigenfaces are sort by eigen value and projected on face space. Once the image is projected then euclidean difference of two images are calculated. Depending upon this euclidean difference threshold value, FAR, FRR are calculated. And this threshold value gives the actual result saying the image captured of the object is regarding to that object or not.

III. CRITICISMS

A. Limitations

Face recognition does not work well where poor lighting, sunglasses, long hair or other objects which are partially covers the subject's face and low resolution images. It's another serious disadvantage is that many systems are less effective if facial expressions vary. Even a big smile can render the system less effective. For eg.: Canada now allows only neutral facial expressions in passport photos.

B. Efficiency

Critics of the technology have complained that the London Borough of Newham, as of 2004, never recognized a single criminal, despite several criminals in the system's database are living in the Borough and the system have been running for several years. "Not once, as far as the police know, has Newham's automatic facial recognition system spotted a live target ". This information seems conflicting with claims that the system was credited with a 34% reduction in crime (hence why it was rolled out to Brimingham also). However it can be explained by the notion that when we regularly told to that they are under constant video surveillance with advanced face recognition technology, this fear alone can reduce the crime rate, whether the face recognition system technically works or does not.

Privacy problems

Many citizens' complaints that their privacy is being compromised by the use of surveillance technologies by corporations and the state. Some fear that it could lead to a "total surveillance society," with the government and other authorities having the ability to know the whereabouts and activities of all citizens around the clock. This knowledge could continue to be deployed to prevent the lawful exercise of rights of citizens to criticize those in office, specific government policies or corporate practices.

CONCLUSION:

Built-in Hadoop Distributed File System (HDFS) security features such as Access Control Lists and Kerberos used alone are not adequate for enterprise needs. So this paper proposes security enhancement with Eigen face recognition. The Eigen face approach to face recognition is motivated by information theory. Eigen face approach does provide a practical solution to the problem of face recognition. It is simple and relatively fast. Eigenface method is useful for its ability to compress large datasets into a small number of eigenfaces and weights. One of the major advantages of eigenfaces recognition approach is the ease of implementation. However, a few limitations are demonstrated as well. First, the algorithm is sensitive to head scale. Second, it is applicable only to front views. Third, it shows good performance only under controlled environment, and may fail in natural scenes.

REFERENCES:

[1] Access Security on Cloud Computing Implemented in Hadoop System

http://dl.acm.org/citation.cfm?id=2057349

[2] M. Turk and A. Pentland (1991). "Face recognition using eigenfaces". Proc. IEEE Conference on Computer Vision and Pattern Recognition. pp. 586–591. http://www.cs.ucsb.edu/~mturk/Papers/mturk-CVPR91.pdf.

[3] HDFS Users Guide - Rack Awareness

[4] "Google's MapReduce Programming Model -- Revisited" — paper by Ralf Lämmel; from Microsoft

[5] B. Clifford Neuman and Theodore Ts'o (September 1994). "Kerberos: An Authentication Service for Computer Networks". IEEE Communications 32 (9): 33–8. doi:10.1109/35.312841.

[6]D. Pissarenko (2003). Eigenface-based facial recognition. http://openbio.sourceforge.net/resources/eigenfaces/eigenfaces-html/facesOptions.html.

[7] Jonathon Shlens, A Tutorial on Principal Component Analysis.

ISBN: 9788175157262