Human Gesture Recognition using Principle Component Analysis

Mohit Arora[1], Somya Chaudhary[2]
[1]Assistant Professor, Shri USB College, Abu Road
[2]Assistant Professor, Siddhi Vinayak Group of College, Alwar
[2]somyaa.chaudhary@gmail.com

Abstract- Gesture recognition is a topic in computer science and language technology with the goal of interpreting human gestures via mathematical algorithms. Gesture recognition can be seen as a way for computers to begin to understand human body language, thus building a richer bridge between machines and humans. Gestures can originate from any bodily motion or state but commonly originate from the face and hand.

The basic idea of PCA is to reduce the number of variables used to span a dataset, thereby reducing the dimension of the classification problem. The procedure is to transform the dataset into a feature space where the variables are uncorrelated and the first principal component holds the highest variance, the second principal component the second highest variance etc. When the dataset has been transformed most of the principal components contain very little information and can therefore be ignored.

Keyword- PCA, Eigenvalues, Eigenfaces,Eigenvectors

I.INTRODUCTION

An idea behind PCA is as follows: Imagine that every pixel in an image is a variable and say that an image consist of N pixels. The image can then be represented as a point in a N-dimensional space. Since image of different gesture from the human face look very alike, they will not be randomly distributed in the N-dimensional space, but instead they will be located in a much smaller subspace, which may even have fewer dimensions than the number of gesture classes. This subspace is found by PCA.

When PCA is used in a system to classify images, it is separated into two parts: an on-line and an on-line part. The o-line part is performed in order to and the transformation matrix and generate a classifier, all based on a set of training images. The on-line part uses the transformation matrix and the classifier computed off-line to transform and classify any new images.

The transformation maps images from image space to a low-dimensional feature space, in which gesture classes make up clusters. Thus, the approach is very pixel-oriented, introducing sensitivity towards the position, the orientation and the scaling/size of the hand in the image, e.g., two images showing the same gesture but at different positions/rotation/scaling will not be transformed to the same point in feature space. To overcome this problem every image is subjected to a normalization procedure.

II.EIGEN VALUE

Eigen Values and Eigen vectors are derived from German word “, which means "proper" or "characteristic." An eigenvalue of a square matrix is a scalar that is usually represented by the Greek letter ג (pronounced lambda). Eigenvector is a vector which can not be zero. Eigenvector is denoted by small letter x. All eigenvalues and eigenvectors satisfy the equation Ax = ג x for a given square matrix, *A*.

III.EIGENFACES

Eigenfaces are a set of eigenvectors used in the computer vision problem of human face recognition. The approach of using eigenfaces for recognition was developed by Sirovich and Kirby (1987) and used by Matthew Turk and Alex Pentland in face classification. It is considered the first successful example of facial recognition technology. These eigenvectors are derived from the covariance matrix of the probability distribution of the high-dimensional vector space of *possible faces of human beings*.

A)Eigenface generation

To generate a set of eigenfaces, a large set of digitized images of human faces, taken under the same lighting conditions, are normalized to line up the eyes and mouths. They are then all resample at the same pixel resolution. Eigenfaces can be extracted out of the image data by means of a mathematical tool called principal component analysis (PCA). Here are the steps involved in converting an image of a face into eigenfaces:

1. Prepare a training set. The faces constituting the training set **T** should be already prepared for processing.
2. Subtract the mean. The average matrix **A** has to be calculated and subtracted from the original in **T**. The results are stored in variable **S**.
3. Calculate the covariance matrix.
4. Calculate the eigenvectors and eigenvalues of this covariance matrix.
5. Choose the principal components.

There will be a large number of eigenfaces created before step 5, and far fewer are really needed. Select from them those that have the highest eigenvalues. For instance, if we are working with a 100 x 100 image, then this system will create 10,000 eigenvectors. Since most individuals can be identified using a database with a size between 100 and 150, most of the 10,000 can be discarded, and only the most important should remain.

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 and medical imaging. Therefore, some do not use the term eigenface, but prefer to use 'eigenimage'. Research that applies similar eigen techniques to sign language images has also been made.

B)Use in facial recognition

Facial recognition was the source of motivation behind the creation of eigenfaces. For this use, eigenfaces have advantages over other techniques available, such as the system's speed and efficiency. Using eigenfaces is very fast, and able to functionally operate on lots of faces in very little time. Unfortunately, this type of facial recognition does have a drawback to consider: trouble recognizing faces when they are viewed with different levels of light or angles. For the system to work well, the faces need to be seen from a frontal view under similar lighting. Face recognition using eigenfaces has been shown to be quite accurate. By experimenting with the system to test it under variations of certain conditions, the following correct recognitions were found: an average of 96% with light variation, 85% with orientation variation, and 64% with size variation.

IV.PRINCIPAL COMPONENT ANALYSIS

Principal component analysis (PCA) is a vector space transform often used to reduce multidimensional data sets to lower dimensions for analysis.

PCA was invented in 1901 by Karl Pearson. PCA involves the calculation of the eigenvalue decomposition of a data covariance matrix or singular value decomposition of a data matrix, usually after mean centering the data for each attribute. PCA is the simplest of the true eigenvector-based multivariate analyses. Often, its operation can be thought of as revealing the internal structure of the data in a way which best explains the variance in the data. If a multivariate dataset is visualized as a set of coordinates in a high-dimensional data space (1 axis per variable), PCA supplies the user with a lower-dimensional picture, a "shadow" of this object when viewed from it’s (in some sense) most informative viewpoint.

PCA is mathematically defined as an orthogonal linear transformation that transforms the data to a new coordinate system such that the greatest variance by any projection of the data comes to lie on the first coordinate (called the first principal component), the second greatest variance on the second coordinate, and so on. PCA is theoretically the optimum transform for a given data in least square terms.

PCA can be used for dimensionality reduction in a data set by retaining those characteristics of the data set that contribute most to its variance, by keeping lower-order principal components and ignoring higher-order ones. Such low-order components often contain the "most important" aspects of the data. However, depending on the application this may not always be the case.

REFERENCES

[1] R.C. Gonzales and R.E.Woods. Digital Image processing. Prentice Hall, second edition, 2002.
[2] “Face recognition using eigenfaces”, M. Turk and A. Pentland
[3] “Eigenfaces for recognition”, M.Turk and A. Pentland “Stellar Spectral Classification using Principal Component Analysis and artificial neural networks”, Harinder P Singh, Ravi K Gulati and Ranjan Gupta
[4] Smith I L.: A tutorial on Principal Components Analysis. Student tutorial. 2002.