Image Recognition and Test Analysis Based on PCA Method

12112610 王越

1. Abstract

In a Yale face database, there are 15 individuals, each with 11 images. The requirement is to select one image per individual to create a sample library and derive a PCA feature library from this sample library. Subsequently, select any image from the Yale image database and identify its identity.

Each image can be regarded as a matrix composed of pixel values or expanded into a vector representation. We employ the PCA method to project the original image space onto a principal component space, significantly reducing the dimensionality of the original space. Subsequently, we determine the similarity between images using a specific metric based on the projection of images onto this principal component space. In this study, we utilize the most common distance measurement methods.

2. Introduction

In the 21st century, there has been a significant decline in crime rates compared to previous eras, and the widespread adoption of social surveillance has played an undeniable role in this process. With the advancement of facial recognition technology, our facial databases are becoming increasingly enriched, allowing us to search for corresponding individual information in our database based on the results captured by any camera, thereby providing conditions for further enhancing societal security. Our research is based on this application scenario, where we utilize publicly available datasets from the internet to simulate a database and a test set for image recognition tasks.

Our core methodology employs PCA grayscale feature extraction to transform images into a feature space for subsequent similarity comparison. Additionally, to assess the effectiveness of our algorithm, we have devised several image perturbation methods based on real-world scenarios, such as image cropping, adding Gaussian noise, image pixelation, and contour extraction to alter styles. We analyze the results of these perturbations and examine the strengths and weaknesses of our model.

3. Methods

3.1 PCA principle

The K-L transform, also known as the Karhunen-Loève transform, utilizes the orthogonal matrix composed of normalized orthogonal eigenvectors of the covariance matrix of the original data as the transformation matrix. It performs orthogonal transformation on the original data, achieving data compression in the transformed domain. This transformation possesses characteristics such as decorrelation and energy concentration. Under the mean square error criterion, it is a transformation that minimizes distortion, effectively removing correlations between the original data. PCA, on the other hand, involves selecting the

eigenvectors corresponding to the k largest eigenvalues of the covariance matrix to form the K-L transform matrix.
The number of principal components to retain depends on the percentage of cumulative variance retained in the total variance, indicating the extent to which the first few principal components summarize the information. In practice, a rough percentage can be set to determine how many principal components to retain; if retaining an additional principal component contributes only marginally to the increase in cumulative variance, it is typically not retained. Assuming that an image of a human face consists of N pixels, it can be represented as an N-dimensional vector Γ. In this way, a training sample set can be represented as Γi (i = 1, ..., M). The orthogonal eigenvectors of the covariance matrix C are the basis vectors that constitute the face space, known as eigenfaces.
Arranging the eigenvalues in descending order: $\lambda 1 \geq \lambda 2 \geq ... \geq \lambda r$, their corresponding eigenvectors are denoted as μk. In this manner, each human face image can be projected into a subspace spanned by u1, u2, ..., ur. Therefore, each human face image corresponds to a point in this subspace. Similarly, any point in the subspace corresponds to an image. With such a reduced-dimensional subspace formed by "eigenfaces," any facial image can be projected onto it to obtain a set of coordinate coefficients. These coefficients indicate the position of the image in the subspace, serving as the basis for facial recognition. The computation of coordinates for each image in the database within this subspace results in a set of coordinates, which becomes the search space for the subsequent recognition and matching process.
Calculating the coordinates of a new input image within the subspace, employing the minimum distance method, involves traversing the search space to obtain the coordinate vector with the smallest distance from the input image. The corresponding facial image associated with this vector is identified as the recognition match result.

3.2 perturbation analysis

To evaluate and enhance our core PCA (Principal Component Analysis) algorithm, we aim to simulate real-world scenarios by conducting extensive tests that replicate the common challenges encountered in practical image acquisition. Typically, images obtained from sources such as surveillance cameras are characterized by a range of issues, including incomplete image capture, noise interference, low resolution due to suboptimal pixel quality, and stylization in human images. Based on the specific attributes of our PCA algorithm, we have elected to focus on the challenge posed by incomplete images. To address this, we have adopted an interpolation approach to reconstruct entire images. This strategy is intended to augment our software and enhance its robustness in handling real-world data variabilities.

3.3 experiment steps

Each individual chooses 4 images from a pool of 60 images to serve as training samples. Each image, which has dimensions of 128x128 pixels, is transformed into a column vector denoted as Xi (16384x1). These column vectors are then organized into a data matrix.

$$X = (X1,X2,\dots,Xn)\ (n=60)\ (1)$$

Subsequently, we calculate, in succession, the mean vector, the data matrix after centering, and the covariance matrix.

$$\mu = 1/n \sum \boldsymbol{Xi}\ (2)$$

$$C=(X1-\mu,X2-\mu,\dots,Xn-\mu)\ (3)$$

$$\Sigma = 1/n CCT\ (4)$$

First, it is essential to compute the eigenvalues of the covariance matrix, select the top k largest eigenvalues, and derive the corresponding eigenvectors (commonly known as 'eigenfaces'). These k eigenvectors are then arranged as columns in a transformation matrix with dimensions of 16384*k.

$$W=(e1,e2,\dots,ek)\ (5)$$

Subsequently, we proceed to compute the projection of each image (a k-dimensional column vector) and calculate the projection of the target face to be recognized (a k-dimensional column vector), denoted as Z.

$$Yi=W\ T\ (Xi-\mu)\ (6)$$

$$chZ=WT\ (Z-\mu)\ (7)$$

Finally, we perform an exhaustive search, j donating the j th person.

$$Yj=\min\|Yi-chZ\|\ (8)$$

4. Results

4.1 Incomplete Images & Results After Image Reconstruction

We resized the images to 60x128 dimensions, and subsequently filled the cropped areas to restore them to 128x128 dimensions. This process resulted in the generation of significant black regions within the images, which had a substantial impact on methods relying on PCA (Principal Component Analysis) for pixel values. Our findings have empirically confirmed this phenomenon.

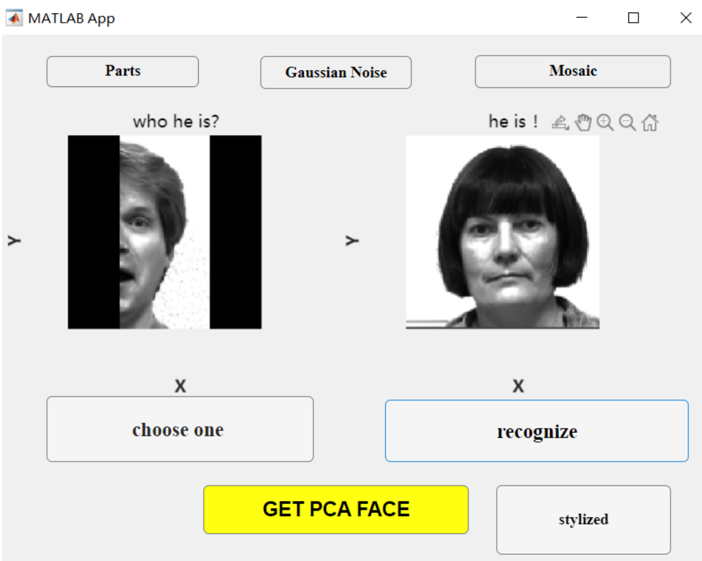

Fig 1

It is evident that incomplete images of this nature can have a profound impact, with much of the impact stemming from the misjudgments arising from alterations in pixel values, which do not adequately reflect the real-world context. Consequently, we introduced an image interpolation reconstruction feature to restore these portions of the images as faithfully as possible before proceeding with the recognition process.

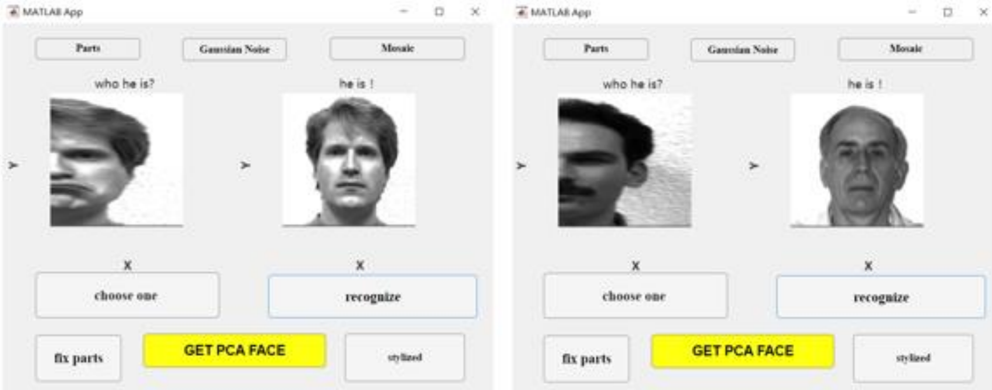

Fig 2

We can observe that with such improvements in place, there is a noticeable enhancement in our results. However, there still exists a considerable prevalence of misclassifications.

4.2 Gaussian Noise

To simulate a reduction in image resolution, we initially employed Gaussian noise as a tool to degrade the resolution.

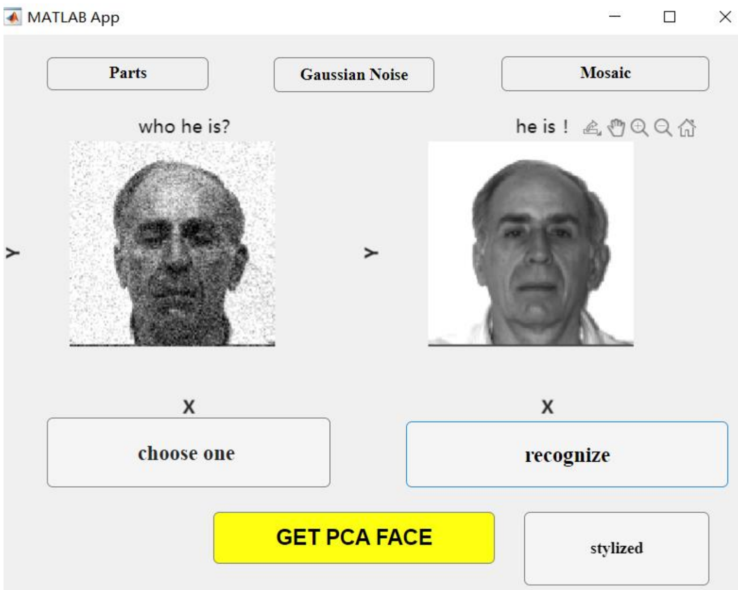

Fig 3

We can observe that our algorithm exhibits high accuracy, except in cases where the shadowing is extremely severe.

4.3 Pixelation

However, in real-life scenarios, we often encounter resolution degradation in the form of pixelation, which is commonly observed when enlarging an image.

In this experimental phase, we observed that pixelation, due to its relatively unaltered pixel value distribution, results in a high level of recognition accuracy.

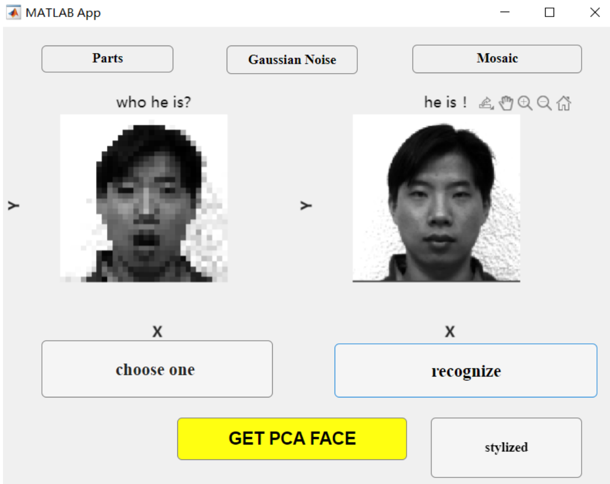

Fig 4

4.4 Contour Extraction

In everyday life, particularly in criminal investigations, there are often situations where a preliminary sketch of a suspect's portrait is created. We aim to simulate the style of such rough sketches, and in our practical implementation, we extracted the contours of the photographs for assessment. We observed that this operation significantly alters the pixel distribution, resulting in poor recognition performance. This suggests that our PCA algorithm is no longer suitable for recognition tasks in such scenarios. To enhance performance, it becomes necessary to incorporate criteria beyond pixel values, such as texture.

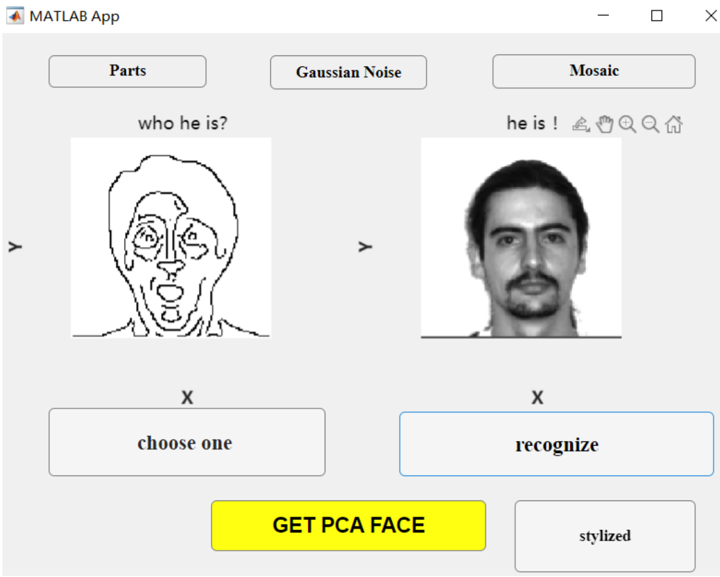

Fig 5

5. Conclusion

Due to the reliance on grayscale-related calculations in our PCA method, it exhibits heightened sensitivity to variations in shading while lacking consideration for contour and texture details. This disparity is particularly evident in the results. In cases where noise or pixelation is introduced, situations that do not substantially alter the overall grayscale relationships, our recognition performance remains reasonable, simulating scenarios where image resolution is low in real-life. However, when dealing with incomplete images, even those containing only contours, recognition performance is markedly diminished. After conducting experiments following image reconstruction, we observed improved results, further affirming our hypothesis.

In subsequent research efforts aimed at enhancing recognition performance, I believe it is imperative to incorporate considerations for texture features. Employing machine learning techniques, for instance, could potentially lead to significant improvements. Currently, our recognition criteria solely rely on pixel values, which is overly simplistic and lacks robustness. Our experiments indicate that this algorithm struggles to handle complex real-world situations. We hypothesize that the inclusion of texture features in the assessment will yield favorable results, and we intend to validate this in future experiments.